



Guide de l'utilisateur

AWS Ground Station



AWS Ground Station: Guide de l'utilisateur

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

Les marques et la présentation commerciale d'Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n'est pas d'Amazon, d'une manière susceptible de créer une confusion parmi les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce que c'est AWS Ground Station ?	1
Cas d'utilisation courants	1
Étapes suivantes	2
Comment AWS Ground Station fonctionne	3
Intégration par satellite	3
Composition du profil de mission	3
Planification des contacts	5
Exécution du contact	6
Jumeau numérique	9
Comprendre les composants AWS Ground Station de base	9
Profils de mission	11
Configs	14
Groupes de points de terminaison de flux de données	24
AWS Ground Station Agent	32
Mise en route	34
Inscrivez-vous pour un Compte AWS	34
Création d'un utilisateur doté d'un accès administratif	34
Ajoutez AWS Ground Station des autorisations à votre AWS compte	36
satellite embarqué	38
Vue d'ensemble du processus d'intégration des clients	38
(Facultatif) Dénomination des satellites	39
Satellites de diffusion publics	42
Planifiez les voies de communication de votre flux de données	42
Livraison de données asynchrone	43
Livraison synchrone des données	43
Planifiez votre télémétrie	44
Création de configurations	45
Configurations de livraison de données	46
Configuration de télémétrie (facultatif)	46
Configurations satellites	46
Création d'un profil de mission	47
Comprendre les prochaines étapes	48
AWS Ground Station Succursales	49
Trouver la région AWS pour l'emplacement d'une station au sol	49

AWS Ground Station régions AWS prises en charge	51
Disponibilité des jumeaux numériques	51
Antennes dédiées	51
Antennes de visualisation sur une station au sol	52
Exemple : répertorier les antennes d'une station au sol	52
Afficher les réservations de stations au sol	53
Réservations d'annonces	53
Types de réservation	54
Exemple de code	54
AWS Ground Station masques de site	56
Masques spécifiques au client	56
Impact des masques de site sur les temps de contact disponibles	57
AWS Ground Station Fonctionnalités du site	57
Comprendre comment AWS Ground Station utilise les éphémérides	62
Données d'éphémérides par défaut	63
Fournir des données d'éphémérides personnalisées	63
Présentation de	63
Exemple : utilisation d'éphémérides fournies par le client avec AWS Ground Station	64
Fournir des données d'éphémérides TLE	64
Fournir des données d'éphémérides OEM	71
Fournir des données sur les éphémérides d'altitude azimutale	80
Réservez des contacts avec des éphémérides personnalisées	91
Présentation de	91
Flux de travail de réservation de contacts	91
Flux de travail 1 : Répertorier les contacts disponibles puis réserver	92
Workflow 2 : réservation par contact direct	96
Surveillance des changements d'état des contacts	100
Bonnes pratiques et considérations	102
Comprendre quelle éphéméride est utilisée	103
Éphémérides TLE et OEM	103
Éphémérides d'élévation d'azimut	104
Effet des nouvelles éphémérides sur les contacts précédemment programmés	104
Obtenir l'éphéméride actuelle d'un satellite	105
Exemple de GetSatelliteretour pour un satellite utilisant une éphéméride par défaut	106
Exemple GetSatellite de satellite utilisant une éphéméride personnalisée	106
Liste des éphémérides d'élévation d'azimut	107

Revenir aux données d'éphémérides par défaut	108
Inversion des éphémérides TLE et OEM	108
Gestion des éphémérides d'élévation d'azimut	109
Travailler avec des flux de données	110
AWS Ground Station interfaces de plan de données	110
Utilisation de la diffusion de données entre régions	111
Installation et configuration d'Amazon S3	111
Configuration et configuration d'Amazon VPC	112
Configuration VPC avec agent AWS Ground Station	113
Configuration VPC avec un point de terminaison de flux de données	116
Configurer et configurer Amazon EC2	118
Logiciel commun fourni	119
AWS Ground Station Images de machines Amazon (AMIs)	119
Travailler avec la télémétrie	120
Comment fonctionne la télémétrie	120
Types de télémétrie disponibles	121
Disponibilité par région	121
Configuration de la télémétrie	122
Étape 1 : créer des AWS ressources prérequisées	122
Étape 2 : Création d'un TelemetrySinkConfig	124
Étape 3 : ajouter la télémétrie à votre profil de mission	124
Étape 4 : Planifier un contact	124
Étapes suivantes	125
Comprendre les données de télémétrie	125
Vue d'ensemble des formats de données	126
Télémétrie de pointage	126
Télémétrie de suivi	128
Lecture de données depuis le flux Kinesis Data Streams	130
Versionnement et évolution des schémas	131
Travailler avec des contacts	132
Comprendre le cycle de vie des contacts	133
AWS Ground Station statuts des contacts	135
Conservation des données de contact	136
Comprendre la facturation des contacts	137
Définitions de bande passante	137
Modes de planification	137

CancelContact	138
Scénario 1 : contact unique	138
Scénario 2 : contact à arrêt unique	139
Scénario 3 : double exemplaire	139
Scénario 4 : double court	140
Scénario 5 : plusieurs doublons	141
Scénario 6 : arrêts multiples	143
Scénario 7 : station au sol à antennes multiples sans doublon	144
Scénario 8 : station au sol à antennes multiples avec contacts dupliqués	145
Mettre à jour les contacts et la gestion des versions des contacts	146
Comment fonctionne la gestion des versions des contacts	146
Mettre à jour un contact	146
Statuts des versions de contact	148
Exemples de code	149
Considérations	153
AWS Ground Station jumeau numérique	154
AWS Ground Station Antennes dédiées	155
Qu'est-ce qu'une antenne dédiée	155
Visibilité améliorée des réservations	156
Ressources connexes	157
Surveillance	158
Automatisez avec des événements	159
AWS Ground Station Types d'événements	160
Chronologie des événements de contact	160
Événements Ephemeris	163
Enregistrez les appels d'API avec CloudTrail	164
AWS Ground Station Informations dans CloudTrail	164
Comprendre les entrées du fichier AWS Ground Station journal	165
Afficher les statistiques avec Amazon CloudWatch	167
AWS Ground Station Métriques et dimensions	167
Affichage des métriques	173
Sécurité	180
Gestion de l'identité et des accès	180
Public ciblé	181
Authentification par des identités	181
Gestion de l'accès à l'aide de politiques	183

Comment AWS Ground Station fonctionne avec IAM	184
Exemples de stratégies basées sur l'identité	190
Résolution des problèmes	193
AWS politiques gérées	195
AWSGroundStationAgentInstancePolicy	196
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy	197
Mises à jour des politiques	197
Utilisation de rôles liés à un service	199
Autorisations de rôle liées au service pour Ground Station	199
Création d'un rôle lié à un service pour Ground Station	200
Modification d'un rôle lié à un service pour Ground Station	200
Supprimer un rôle lié à un service pour Ground Station	200
Régions prises en charge pour les rôles liés au service Ground Station	201
Résolution des problèmes	201
Chiffrement des données au repos pour AWS Ground Station	201
Création d'une clé gérée par le client	203
Spécification d'une clé gérée par le client pour AWS Ground Station	204
AWS Ground Station contexte de chiffrement	205
Chiffrement au repos pour les données d'éphémérides TLE et OEM	205
Chiffrement au repos pour les éphémérides d'élévation d'azimut	214
Chiffrement des données pendant le transit pour AWS Ground Station	223
AWS Ground Station Streams d'agents	224
Flux de points de terminaison de flux de données	224
Exemples de configurations de profil de mission	225
JPSS-1 - Satellite de diffusion public (PBS) - Évaluation	225
Satellite de diffusion publique utilisant la livraison de données Amazon S3	227
Voies de communication	227
AWS Ground Station configurations	229
AWS Ground Station profil de mission	230
Assemblage	231
Satellite de diffusion public utilisant un point de terminaison de flux de données (bande étroite)	232
Voies de communication	232
AWS Ground Station configurations	239
AWS Ground Station profil de mission	240
Assemblage	241

Satellite de diffusion public utilisant un point de terminaison de flux de données (démodulé et décodé)	243
Voies de communication	243
AWS Ground Station configurations	250
AWS Ground Station profil de mission	254
Assemblage	254
Satellite de diffusion publique utilisant AWS Ground Station l'agent (large bande)	257
Voies de communication	257
AWS Ground Station configurations	268
AWS Ground Station profil de mission	270
Assemblage	270
Résolution des problèmes	274
Résoudre les problèmes liés aux contacts qui fournissent des données à Amazon EC2	274
Étape 1 : vérifier que votre instance EC2 est en cours d'exécution	275
Étape 2 : Déterminer le type d'application de flux de données utilisé	275
Étape 3 : vérifier que l'application de flux de données est en cours d'exécution	275
Étape 4 : Vérifiez que le flux d'applications de votre flux de données est configuré	277
Étape 5 : Assurez-vous que vous disposez d'un nombre suffisant d'adresses IP disponibles dans le sous-réseau de votre ou de vos instances de réception	279
Résoudre les problèmes liés aux contacts défaillants	280
Cas d'utilisation ayant échoué pour un point de terminaison Dataflow	280
AWS Ground Station Cas d'utilisation de l'agent ayant échoué	281
Résoudre les problèmes liés aux mises à jour des contacts	282
Erreurs de validation synchrones	282
Codes de défaillance asynchrones	285
Vérifier l'état d'une mise à jour	287
Résoudre les problèmes liés aux contacts FAILED_TO_SCHEDULE	288
Les paramètres spécifiés dans votre Antenna Downlink Demod Decode Config ne sont pas pris en charge	288
Étapes générales de résolution des problèmes	289
Le dépannage DataflowEndpointGroups n'est pas dans un état SAIN	289
Résoudre les problèmes liés aux éphémérides non valides	290
Comprendre les erreurs de validation des éphémérides	290
Erreurs de validation courantes pour les éphémérides TLE	291
Erreurs de validation courantes pour les éphémérides OEM	291
Erreurs de validation courantes pour les éphémérides d'élévation d'azimut	292

Étapes de résolution des problèmes	293
Référence complète du code d'erreur	294
Résoudre les problèmes liés aux contacts qui n'ont reçu aucune donnée	298
Configuration de liaison descendante incorrecte	298
Manœuvre du satellite	299
AWS Ground Station panne	299
Résoudre les problèmes de télémétrie	300
Problèmes de configuration courants	300
Problèmes de transmission de données télémétriques	303
Problèmes liés au format des données	304
Obtenir de l'aide	305
Quotas et limites	307
Modalités du service	308
Historique du document	309
AWS Glossaire	315
.....	cccxvi

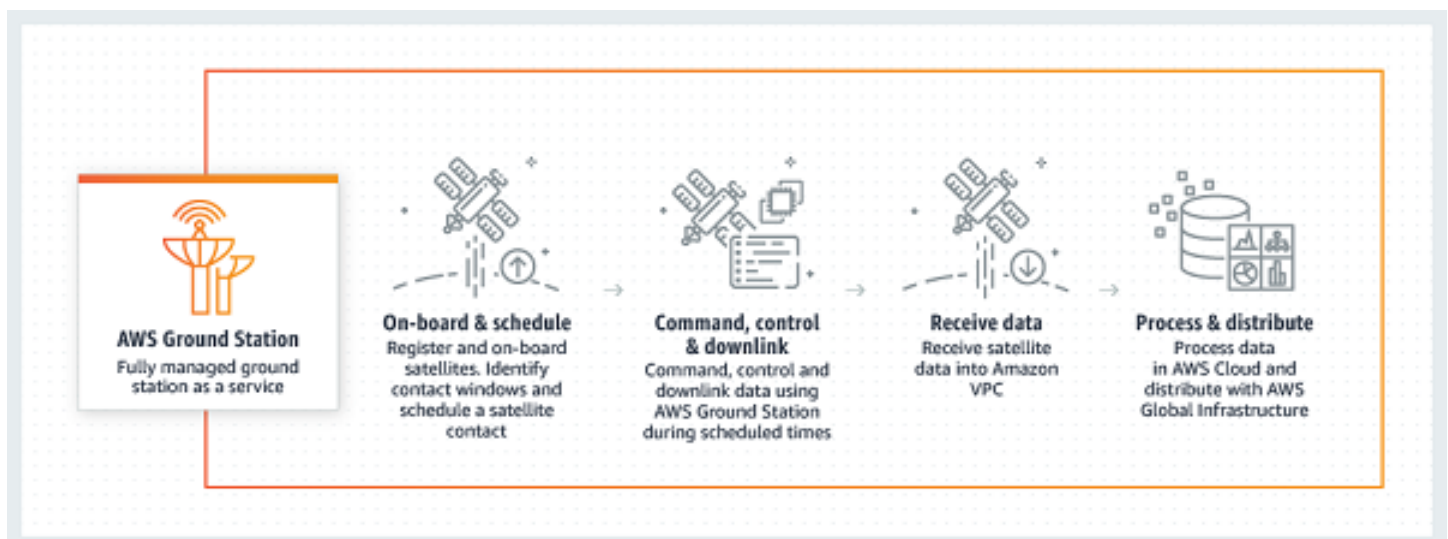
Qu'est-ce que c'est AWS Ground Station ?

AWS Ground Station est un service entièrement géré qui fournit des communications par satellite sécurisées, rapides et prévisibles au sein d'une infrastructure mondiale. Ainsi AWS Ground Station, vous n'avez plus besoin de construire, de gérer ou de faire évoluer votre propre infrastructure de station au sol. AWS Ground Station vous permet de vous concentrer sur l'innovation et d'expérimenter rapidement de nouvelles applications qui ingèrent des données satellites, plutôt que de consacrer des ressources à la construction, à l'exploitation et à la mise à l'échelle de vos propres stations au sol.

Grâce au réseau mondial de fibre optique à faible latence et à large bande passante d'AWS, vous pouvez commencer à traiter vos données satellites quelques secondes après réception par le système d'antenne. Cela vous permet de transformer les données brutes en informations traitées ou en connaissances analysées en quelques secondes.

Pour les organisations ayant des exigences spécifiques, des offres sont AWS Ground Station également proposées [AWS Ground Station Antennes dédiées](#) : des systèmes d'antennes sur mesure qui AWS gèrent en votre nom, fournissant un accès dédié aux antennes fabriquées selon vos spécifications.

Cas d'utilisation courants



AWS Ground Station vous permet de communiquer avec vos satellites de manière bidirectionnelle et prend en charge les cas d'utilisation suivants :

- Données en liaison descendante : [recevez des données de vos satellites, en transmettant des fréquences en bande X et en bande S, transmises à une instance Amazon EC2 en temps réel \(format VITA-49\) ou directement vers un compartiment Amazon S3 de votre compte \(format PCAP\)](#). En outre, pour les satellites qui utilisent un schéma de modulation et de codage pris en charge, vous pouvez choisir entre recevoir des données démodulées et décodées ou des échantillons bruts de fréquence intermédiaire numérique (DigIF) (format VITA-49).
- Données en liaison montante : envoyez des données et des commandes à vos satellites recevant des fréquences en bande S en envoyant des données DigIF (format VITA-49) à transmettre par AWS Ground Station
- Echo Uplink : validez les commandes envoyées à votre vaisseau spatial et effectuez d'autres tâches avancées en recevant le signal transmis sur une antenne physiquement colocalisée.
- Radio définie par logiciel (SDR) /processeur frontal (FEP) : utilisez votre and/or FEP SDR existant, capable de fonctionner sur une instance Amazon EC2, pour traiter vos données en temps réel selon vos formes d'onde existantes et générer send/receive vos produits de données.
- Télémétrie, suivi et commande (TT&C) — Effectuez le TT&C en utilisant une combinaison des cas d'utilisation répertoriés précédemment pour gérer votre flotte de satellites.
- Livraison de données entre régions : gérez plusieurs contacts simultanés à l'aide AWS Ground Station du réseau d'antennes mondial d'une seule région AWS.
- Jumeau numérique — Planification des tests, vérification des configurations et gestion appropriée des erreurs à un coût réduit sans utiliser la capacité de production de l'antenne.

Étapes suivantes

Nous vous recommandons de commencer par lire les sections suivantes :

- Pour en savoir plus sur AWS Ground Station les concepts essentiels, voir [Comment AWS Ground Station fonctionne](#).
- Pour savoir comment configurer votre compte et les ressources à utiliser AWS Ground Station, consultez [Mise en route](#).
- Pour une utilisation par programmation AWS Ground Station, reportez-vous à la référence de [l'AWS Ground Station API](#). La référence d'API décrit AWS Ground Station en détail toutes les opérations d'API. Il fournit également des exemples de demandes, de réponses et d'erreurs pour les protocoles de service Web pris en charge. Vous pouvez utiliser la [AWS CLI](#), ou un [AWS SDK](#), dans le langage de votre choix, pour écrire du code qui interagit avec AWS Ground Station

Comment AWS Ground Station fonctionne

AWS Ground Station utilise des antennes au sol pour faciliter la communication avec votre satellite. Les caractéristiques physiques de ce que les antennes peuvent faire sont abstraites et sont appelées capacités. L'emplacement physique de l'antenne ainsi que ses capacités actuelles peuvent être référencés dans [AWS Ground Station Succursales](#) cette section. Veuillez nous contacter par le biais du [AWS Support Center Consoles](#) si votre cas d'utilisation nécessite des fonctionnalités supplémentaires, des offres de localisation supplémentaires ou des emplacements d'antennes plus précis.

Pour utiliser l'une des AWS Ground Station antennes, vous devez réserver une heure à un endroit précis. Cette réservation est considérée comme un contact. Pour planifier un contact avec succès, AWS Ground Station des données supplémentaires sont nécessaires pour garantir son succès.

- Votre satellite doit être embarqué sur un ou plusieurs sites. Cela garantit que vous êtes autorisé à utiliser les différentes capacités à l'endroit demandé.
- Votre satellite doit avoir une éphéméride valide. Cela garantit que les antennes ont une ligne de visée et peuvent pointer avec précision votre satellite pendant le contact.
- Vous devez avoir un profil de mission valide. Cela vous permet de personnaliser le comportement de ce contact, notamment la manière dont vous recevrez et enverrez des données à votre satellite. Vous pouvez utiliser plusieurs profils de mission pour le même véhicule afin de créer différents contacts adaptés aux différentes postures opérationnelles ou aux différents scénarios que vous rencontrez.

Intégration par satellite

L'intégration d'un satellite AWS Ground Station est un processus en plusieurs étapes impliquant la collecte de données, la validation technique, l'octroi de licences de spectre, ainsi que l'intégration et les tests. La section du guide consacrée à [l'intégration par satellite](#) vous guidera tout au long de ce processus.

Composition du profil de mission

Les informations de fréquence du satellite, [les informations du plan de données](#) et d'autres détails sont encapsulés dans un profil de mission. Le profil de mission est un ensemble de composants de

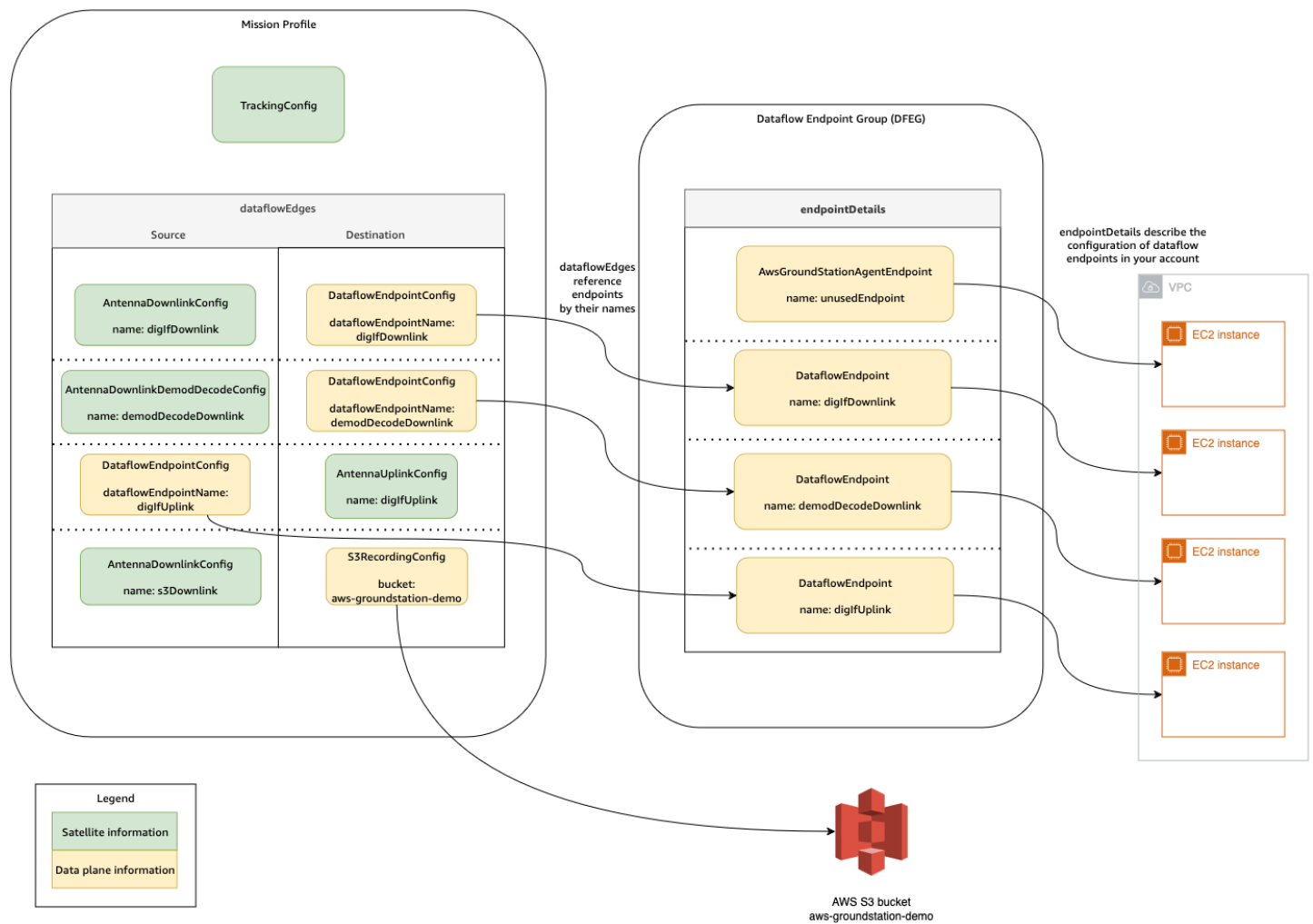
configuration. Cela vous permet de réutiliser les composants de configuration dans différents profils de mission en fonction de votre cas d'utilisation. Étant donné que les profils de mission ne font pas directement référence à des satellites individuels, mais contiennent uniquement des informations sur leurs capacités techniques, les profils de mission peuvent également être réutilisés par plusieurs satellites ayant la même configuration.

Un profil de mission valide comportera une configuration de suivi et un ou plusieurs flux de données. La configuration de suivi indiquera vos préférences en matière de suivi lors d'un contact. Chaque paire de configurations au sein d'un flux de données établit une source et une destination. En fonction de votre satellite et de ses modes de fonctionnement, le nombre exact de flux de données variera dans le profil de mission pour représenter vos voies de communication montantes et descendantes ainsi que les éventuels aspects liés au traitement des données.

- Pour plus d'informations sur la configuration de vos EC2 ressources Amazon VPC, Amazon S3 et Amazon qui seront utilisées lors d'un contact, consultez [Travailler avec des flux de données](#)
- Pour plus de détails sur le comportement de chaque configuration, consultez [Utiliser les AWS Ground Station configurations](#).
- Pour plus de détails sur tous les paramètres attendus, voir [Utiliser les profils AWS Ground Station de mission](#).
- Pour des exemples sur la manière dont différents profils de mission peuvent être créés pour répondre à votre cas d'utilisation, consultez [Exemples de configurations de profil de mission](#).

Le schéma suivant montre un exemple de profil de mission et les ressources supplémentaires nécessaires. Notez que l'exemple montre un point de terminaison de flux de données qui n'est pas nécessaire pour ce profil de mission, nommé UnusedEndpoint, afin de démontrer la flexibilité. L'exemple prend en charge les flux de données suivants :

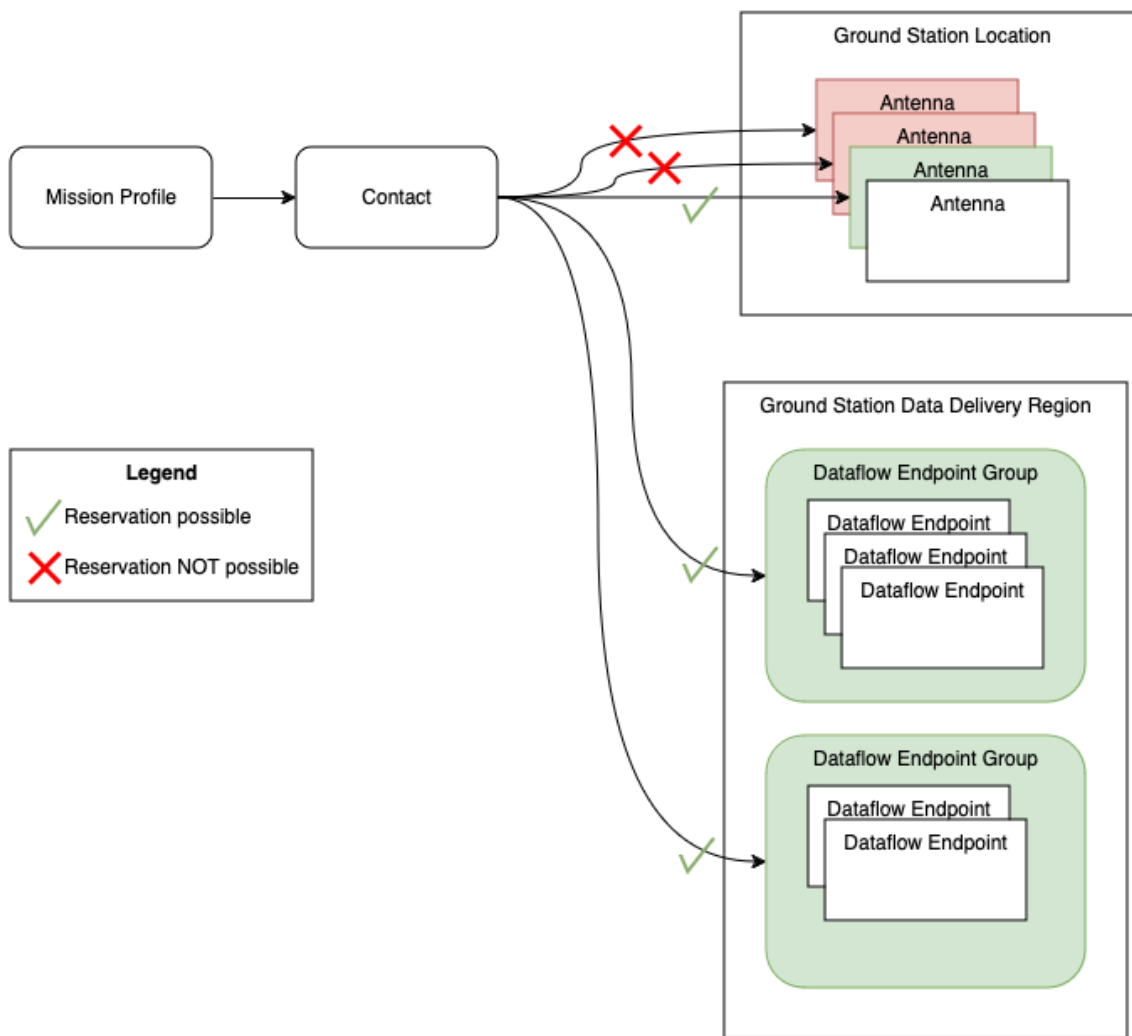
- Liaison descendante synchrone des données numériques à fréquence intermédiaire vers une EC2 instance Amazon que vous gérez. Désigné par le nom digIfDownlink.
- Liaison descendante asynchrone des données numériques à fréquence intermédiaire vers un compartiment Amazon S3. Désigné par le nom aws-groundstation-demodu compartiment.
- Liaison descendante synchrone des données démodulées et décodées vers une instance Amazon EC2 que vous gérez. Désigné par le nom demodDecodeDownlink.
- Liaison montante synchrone entre les données d'une EC2 instance Amazon que vous gérez et une antenne AWS Ground Station gérée. Désigné par le nom digIfUplink.



Planification des contacts

Avec un profil de mission valide, vous pouvez demander un contact avec vos satellites embarqués. La demande de réservation de contact est asynchrone afin de laisser le temps au service d'antenne mondial d'établir un calendrier cohérent dans toutes les AWS régions concernées. Au cours de ce processus, diverses antennes situées à l'emplacement de la station au sol demandé sont évaluées afin de déterminer si elles sont disponibles et capables de traiter le contact. Au cours de ce processus, les points de terminaison de votre flux de données configurés sont également évalués afin de déterminer leur disponibilité. Pendant cette évaluation, le statut du contact sera indiqué dans PLANIFICATION.

Ce processus de planification asynchrone se termine dans les cinq minutes suivant la demande, mais se termine généralement dans la minute qui suit. Veuillez vérifier [Automatisez AWS Ground Station avec des événements](#) la surveillance basée sur les événements lors de la planification.



Les contacts qui peuvent être effectués et qui sont disponibles se traduisent par des contacts PLANIFIÉS. Dans le cas d'un contact planifié, les ressources nécessaires pour effectuer votre contact ont été réservées dans les régions AWS requises, conformément à votre profil de mission. Les contacts qui ne peuvent pas être exécutés ou dont les pièces ne sont pas disponibles se traduiront par des contacts FAILED_TO_SCHEDULE. Voir [Résoudre les problèmes liés aux contacts FAILED_TO_SCHEDULE](#) pour plus de détails sur le débogage.

Exécution du contact

AWS Ground Station orchestrera automatiquement vos ressources gérées par AWS lors de votre réservation de contact. Le cas échéant, vous êtes responsable de l'orchestration des EC2 ressources définies par votre profil de mission en tant que points de terminaison de flux de données. AWS Ground Station fournit [AWS EventBridge Events](#) pour automatiser l'orchestration de vos ressources

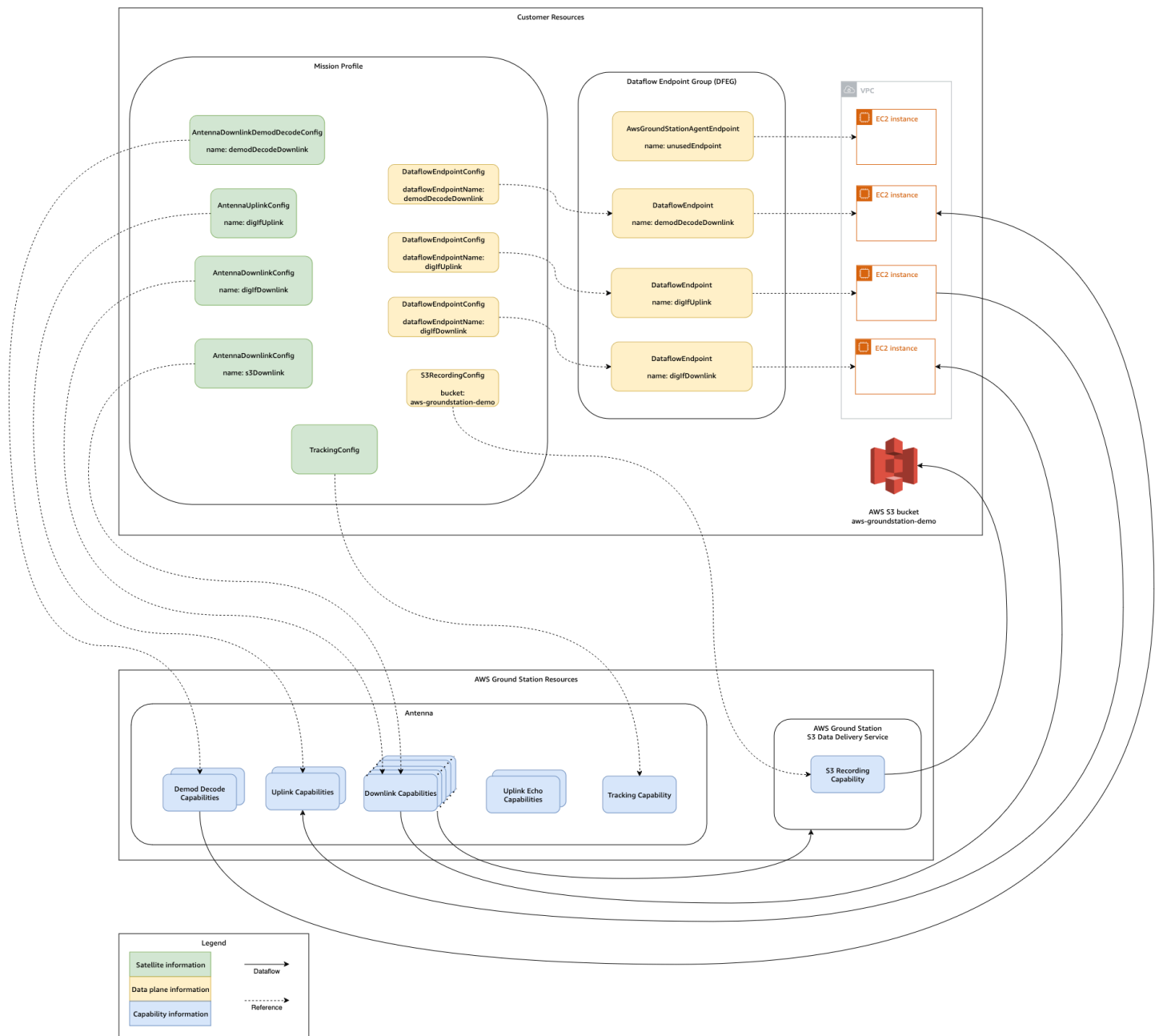
afin de réduire les coûts. Pour plus d'informations, consultez [Automatisez AWS Ground Station avec des événements](#).

Pendant le contact, la télémétrie concernant les performances de votre contact est transmise à AWS. CloudWatch Pour plus d'informations sur la façon de surveiller votre contact pendant l'exécution, veuillez consulter [Comprenez la surveillance avec AWS Ground Station](#).

Le schéma suivant poursuit l'exemple précédent en montrant les mêmes ressources orchestrées lors du contact.

Note

Les capacités de l'antenne n'ont pas toutes été utilisées dans cet exemple. Par exemple, il existe plus d'une douzaine de capacités de liaison descendante disponibles sur chaque antenne qui prennent en charge plusieurs fréquences et polarisations. Pour plus de détails sur le nombre de chaque type de capacité disponible sur les AWS Ground Station antennes, ainsi que sur leurs fréquences et polarisations prises en charge, voir. [AWS Ground Station Fonctionnalités du site](#)



À la fin de votre contact, nous AWS Ground Station évaluerons les performances de votre contact et déterminerons le statut final du contact. Les contacts pour lesquels aucune erreur n'est détectée se traduiront par un statut de contact **TERMINÉ**. Les contacts pour lesquels des erreurs de service ont causé des problèmes de livraison des données pendant le contact se traduiront par un **AWS_FAILED** statut. Les contacts pour lesquels des erreurs du client ou de l'utilisateur ont causé des problèmes de livraison des données pendant le contact se traduiront par un statut d'échec. Les erreurs commises en dehors de l'heure de contact, c'est-à-dire pendant le pré-passage ou après le passage, ne sont pas prises en compte lors de la sélection.

Pour plus d'informations, consultez [Comprendre le cycle de vie des contacts](#).

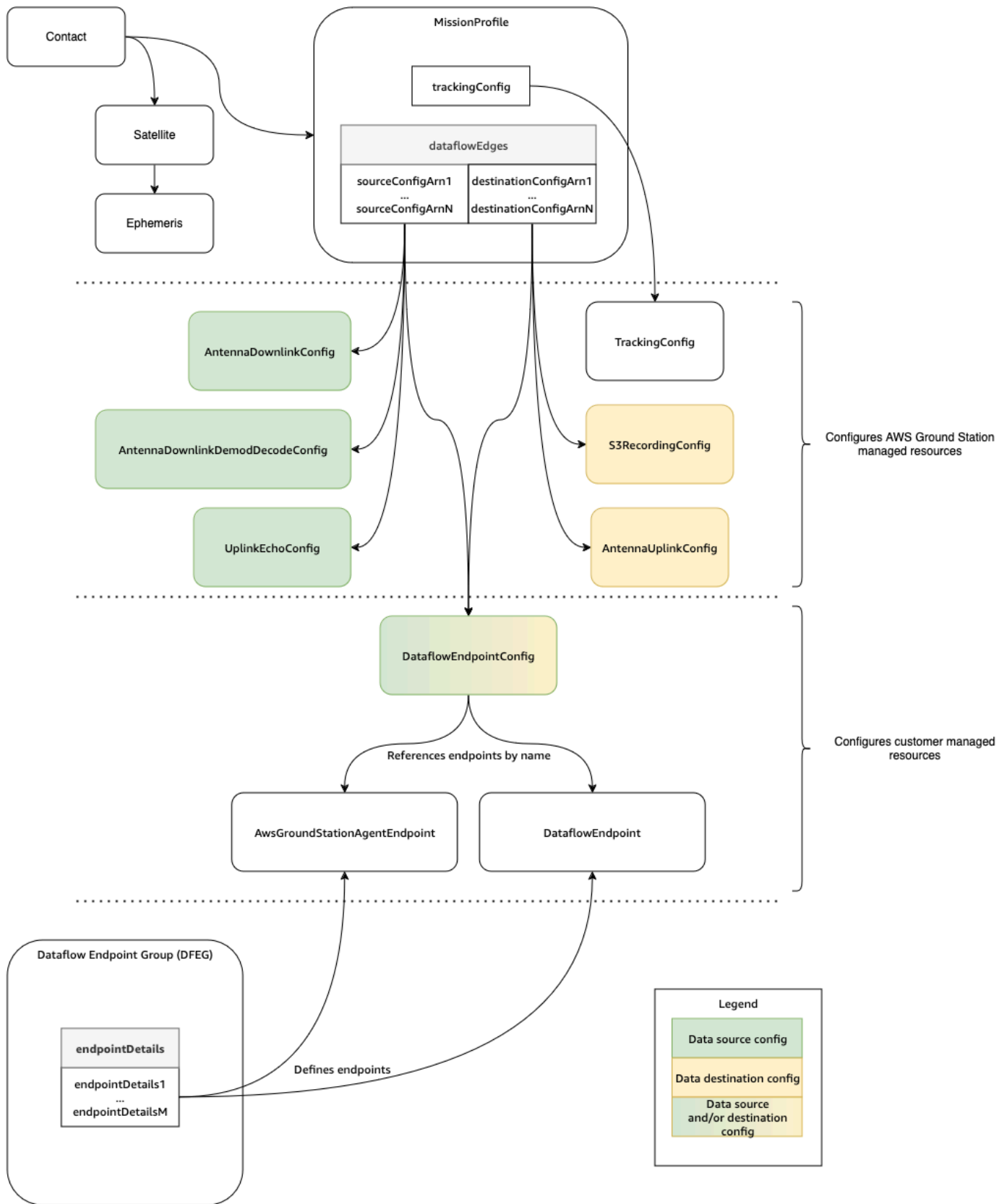
Jumeau numérique

La fonction de jumelage numérique vous AWS Ground Station permet de planifier des contacts en fonction de l'emplacement virtuel des stations au sol. Ces stations terrestres virtuelles sont des répliques exactes des stations terrestres de production, y compris les capacités d'antenne, les masques de site et les coordonnées GPS réelles. La fonction de jumelage numérique vous permet de tester votre flux de travail d'orchestration des contacts pour une fraction du coût par rapport aux stations terrestres de production. Pour plus d'informations, consultez [Utilisez la fonction de jumelage AWS Ground Station numérique](#).

Comprendre les composants AWS Ground Station de base

Cette section fournit des définitions détaillées des principaux composants d'AWS Ground Station.

Le schéma suivant montre les composants principaux AWS Ground Station et la manière dont ils sont liés les uns aux autres. Les flèches indiquent le sens des dépendances entre les composants, chaque composant pointant vers ses dépendances.



Les rubriques suivantes décrivent les AWS Ground Station principaux composants en détail.

Rubriques

- [Utiliser les profils AWS Ground Station de mission](#)
- [Utiliser les AWS Ground Station configurations](#)
- [Utiliser les groupes AWS Ground Station de points de terminaison Dataflow](#)
- [Utiliser l' AWS Ground Station agent](#)

Utiliser les profils AWS Ground Station de mission

Les profils de mission contiennent des configurations et des paramètres concernant la façon dont les contacts sont exécutés. Lorsque vous réservez un contact ou recherchez des contacts disponibles, vous fournissez le profil de mission que vous avez l'intention d'utiliser. Les profils de mission rassemblent toutes vos configurations et définissent comment l'antenne sera configurée et où les données seront acheminées lors de votre contact.

Les profils de mission peuvent être partagés entre des satellites partageant les mêmes caractéristiques radio. Vous pouvez créer des groupes de points de terminaison de flux de données supplémentaires pour limiter le maximum de contacts simultanés que vous souhaitez effectuer pour votre constellation.

Les configurations de suivi sont spécifiées sous forme de champ unique dans le profil de mission. Les configurations de suivi sont utilisées pour spécifier vos préférences en matière de suivi des programmes et de suivi automatique lors de votre contact. Pour de plus amples informations, veuillez consulter [Suivi de Config](#).

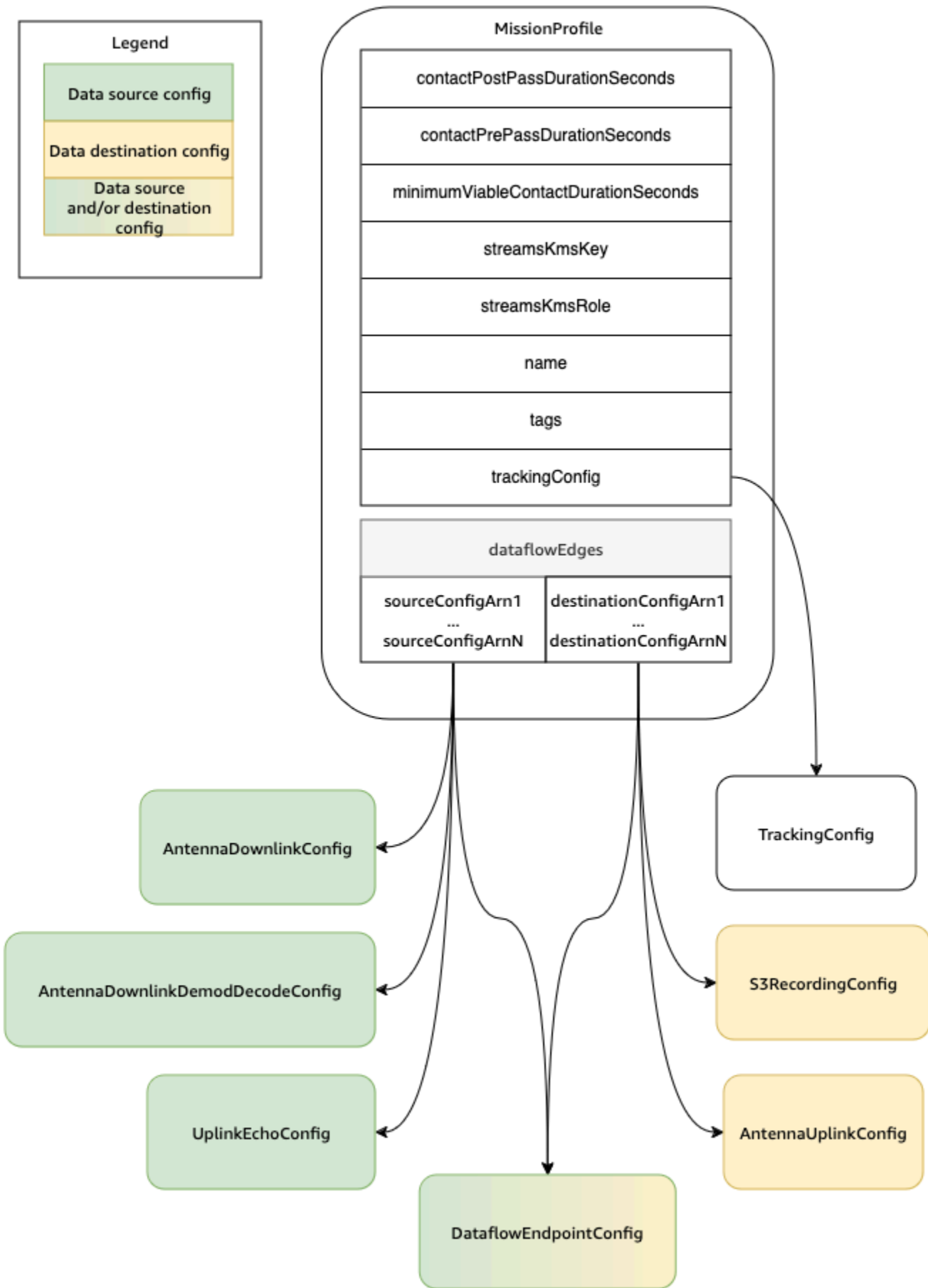
Toutes les autres configurations sont contenues dans le `dataFlowEdges` champ du profil de mission. Ces configurations peuvent être considérées comme des nœuds de flux de données qui représentent chacun une ressource AWS Ground Station gérée capable d'envoyer ou de recevoir des données et la configuration associée. Le `dataFlowEdges` champ définit les nœuds de flux de données source et de destination (configurations) nécessaires. Un seul bord de flux de données est une liste de deux configurations [Amazon Resource Names \(ARNs\)](#) : la première est la configuration source et la seconde est la configuration de destination. En spécifiant une limite de flux de données entre deux configurations, vous indiquez d'où et AWS Ground Station vers où les données doivent circuler lors d'un contact. Pour de plus amples informations, veuillez consulter [Utiliser les AWS Ground Station configurations](#).

Les `contactPrePassDurationSeconds` et `contactPostPassDurationSeconds` permettent de spécifier les heures relatives au contact où vous recevrez une notification d'CloudWatch événement. Pour une chronologie des événements liés à votre contact, veuillez lire [Comprendre le cycle de vie des contacts](#).

Le champ `name` du nom du profil de mission permet de distinguer les profils de mission que vous créez.

Les `streamsKmsRole` et `streamsKmsKey` sont utilisés pour définir le cryptage utilisé AWS Ground Station pour la livraison de vos données avec AWS Ground Station l'Agent. Veuillez consulter [Chiffrement des données pendant le transit pour AWS Ground Station](#).

Le `telemetrySinkConfigArn` champ est facultatif et permet d'activer la AWS Ground Station télémétrie lors des contacts. Lorsque cela est spécifié, AWS Ground Station diffuse des données de télémétrie en temps quasi réel vers votre compte pendant l'exécution de vos contacts. Pour plus d'informations sur la configuration et l'utilisation de la télémétrie, consultez. [Travailler avec la télémétrie](#)



La liste complète des paramètres et des exemples est incluse dans la documentation suivante.

- [AWS::GroundStation::MissionProfile CloudFormation type de ressource](#)

Utiliser les AWS Ground Station configurations

Les configurations sont des ressources AWS Ground Station utilisées pour définir les paramètres de chaque aspect de votre contact. Ajoutez les configs que vous souhaitez à un profil de mission, puis ce profil de mission sera utilisée lors de l'exécution du contact. Vous pouvez définir plusieurs types différents de configs. Les configurations peuvent être regroupées en trois catégories :

- Configurations de suivi
- Configurations de flux de données
- Configurations de télémétrie

A `TrackingConfig` est le seul type de configuration de suivi. Il est utilisé pour configurer le réglage automatique de l'antenne lors d'un contact et est requis dans un profil de mission.

Les configurations qui peuvent être utilisées dans un flux de données de profil de mission peuvent être considérées comme des nœuds de flux de données qui représentent chacun une ressource AWS Ground Station gérée capable d'envoyer ou de recevoir des données. Un profil de mission nécessite au moins une paire de ces configurations, l'une représentant une source de données et l'autre une destination. Ces configurations sont résumées dans le tableau suivant.

Nom de la configuration	Source/destination du flux de données
AntennaDownlinkConfig	Source
AntennaDownlinkDemodDecodeConfig	Source
UplinkEchoConfig	Source
S3 RecordingConfig	Destination
AntennaUplinkConfig	Destination
DataflowEndpointConfig	and/or Destination de la source

A TelemetrySinkConfig est le seul type de configuration de télémétrie. Il est utilisé pour configurer l'endroit où les données de télémétrie seront transmises lors d'un contact, et il est facultatif dans un profil de mission. Lorsqu'il est inclus, il AWS Ground Station diffuse des données télémétriques en temps quasi réel sur votre compte pendant l'exécution de vos contacts.

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur des configurations à l'aide CloudFormation de l' AWS Command Line Interface API ou de l' AWS Ground Station API. Des liens vers la documentation pour des types de configuration spécifiques sont également fournis ci-dessous.

- [AWS::GroundStation::Config CloudFormation type de ressource](#)
- [AWS CLI Référence de configuration](#)
- [Référence de l'API Config](#)

Suivi de Config

Vous pouvez utiliser le suivi des configs dans le profil de mission pour déterminer si autotrack doit être activé pendant vos contacts. Cette config possède un seul paramètre : `autotrack`. Le paramètre `autotrack` peut avoir les valeurs suivantes :

- `REQUIRED` - Autotrack est requis pour vos contacts.
- `PREFERRED` - Autotrack est recommandé pour les contacts, mais les contacts peuvent toujours être exécutés sans autotrack.
- `REMOVED` - Aucun autotrack ne doit être utilisé pour vos contacts.

AWS Ground Station utilisera un suivi programmatique qui pointera en fonction de vos éphémérides lorsque le suivi automatique n'est pas utilisé. Veuillez vous référer [Comprendre comment AWS Ground Station utilise les éphémérides](#) pour plus de détails sur la façon dont les éphémérides sont construites.

Autotrack utilisera le suivi du programme jusqu'à ce que le signal attendu soit détecté. Une fois que cela se produit, le suivi se poursuivra en fonction de la force du signal.

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations de suivi des configurations à l'aide CloudFormation de l' AWS Command Line Interface API ou de l' AWS Ground Station API.

- [AWS::GroundStation::Config TrackingConfig CloudFormation propriété](#)

- [AWS CLI Référence de configuration](#) (voir la `trackingConfig` -> (`structure`) section)
- [TrackingConfig Référence d'API](#)

Config d'antenne de liaison descendante

Vous pouvez utiliser les configurations de liaison descendante de l'antenne pour configurer l'antenne pour la liaison descendante lors de votre contact. Ils consistent en une configuration de spectre qui spécifie la fréquence, la bande passante et la polarisation à utiliser lors de votre contact de liaison descendante.

Cette configuration représente un nœud source dans un flux de données. Il est chargé de numériser les données de radiofréquence. Les données diffusées depuis ce nœud suivront le Data/IP format du signal. Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Si votre cas d'utilisation de la liaison descendante nécessite une démodulation ou un décodage, consultez le [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les configurations de liaison descendante d'antennes à l'aide CloudFormation de l'API AWS Command Line Interface ou de l' AWS Ground Station API.

- [AWS::GroundStation::Config AntennaDownlinkConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la `antennaDownlinkConfig` -> (`structure`) section)
- [AntennaDownlinkConfig Référence d'API](#)

Config de décodage/démodulation des signaux d'antenne de liaison descendante

Les configurations de décodage par démodulation d'antenne sont un type de configuration plus complexe et personnalisable que vous pouvez utiliser pour exécuter des contacts de liaison descendante avec un décodage par démodulation. and/or Si vous souhaitez exécuter ce type de contact, veuillez ouvrir un AWS Support ticket via le [AWS Support Center Console](#). Nous vous aiderons à définir la bonne config et le bon profil de mission pour votre cas d'utilisation.

Cette configuration représente un nœud source dans un flux de données. Il est chargé de numériser les données de radiofréquence et d'effectuer la démodulation et le décodage conformément aux

spécifications. Les données diffusées depuis ce nœud suivront le format Demodulated/Decoded Data/IP. Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Consultez la documentation suivante pour plus d'informations sur la façon d'effectuer des opérations sur les configurations de décodage par démodage de liaison descendante d'antenne à l'aide CloudFormation de l'API ou de l'API AWS Command Line Interface. AWS Ground Station

- [AWS::GroundStation::Config AntennaDownlinkDemodDecodeConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la antennaDownlinkDemodDecodeConfig -> (structure) section)
- [AntennaDownlinkDemodDecodeConfig Référence d'API](#)

Config d'antenne de liaison montante

Vous pouvez utiliser les configurations de liaison montante d'antenne pour configurer l'antenne pour la liaison montante lors de votre contact. Ils consistent en une configuration spectrale avec fréquence, polarisation et puissance isotrope rayonnée effective cible (EIRP). Pour plus d'informations sur la configuration d'un contact pour le bouclage de la liaison montante, consultez. [Config d'écho d'antenne de liaison montante](#)

Cette configuration représente un nœud de destination dans un flux de données. Il convertira le signal de données de radiofréquence numérisé fourni en un signal analogique et l'émettra pour que votre satellite le reçoive. Les données transmises à ce nœud devraient respecter le Data/IP format du signal. Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les configurations de liaison montante d'antenne à l'aide CloudFormation de l'API AWS Command Line Interface ou de l' AWS Ground Station API.

- [AWS::GroundStation::Config AntennaUplinkConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la antennaUplinkConfig -> (structure) section)
- [AntennaUplinkConfig Référence d'API](#)

Config d'écho d'antenne de liaison montante

Les configurations d'écho de liaison montante indiquent comment exécuter un écho de liaison montante. Un écho en liaison montante peut être utilisé pour valider les commandes envoyées à votre vaisseau spatial et effectuer d'autres tâches avancées. Ceci est réalisé en enregistrant le signal réel émis par l' AWS Ground Station antenne (c'est-à-dire la liaison montante). Cela renvoie le signal envoyé par l'antenne à votre point de terminaison de flux de données et doit correspondre au signal transmis. Une configuration d'écho de liaison montante contient l'ARN d'une configuration de liaison montante. L'antenne utilise les paramètres de la configuration de liaison montante indiqués par l'ARN lors de l'exécution d'un écho de liaison montante.

Cette configuration représente un nœud source dans un flux de données. Les données diffusées depuis ce nœud respecteront le Data/IP format du signal. Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les configurations d'écho de liaison montante à l'aide CloudFormation de l'API AWS Command Line Interface ou de l' AWS Ground Station API.

- [AWS::GroundStation::Config UplinkEchoConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la `uplinkEchoConfig` -> (`structure`) section)
- [UplinkEchoConfig Référence d'API](#)

Config de point de terminaison de flux de données

Note

Les configurations des points de terminaison Dataflow sont uniquement utilisées pour la livraison de données à Amazon EC2 et ne sont pas utilisées pour la livraison de données à Amazon S3.

Vous pouvez utiliser les configurations de point de terminaison de flux de données pour spécifier quel point de terminaison de flux de données d'un [groupe de points de terminaison de flux de données](#) à partir duquel ou vers lequel vous souhaitez que les données circulent lors d'un contact. Les deux paramètres d'une configuration de point de terminaison de flux de données spécifient le nom et la région du point de terminaison de flux de données. Lorsque vous réservez un contact, AWS Ground

Station analyse le [profil de mission](#) que vous avez spécifié et tente de trouver un groupe de points de terminaison de flux de données dans la AWS région contenant tous les points de terminaison de flux de données spécifiés par les configurations de point de terminaison de flux de données contenues dans votre profil de mission. Si un groupe de points de terminaison de flux de données approprié est trouvé, le statut du contact deviendra SCHEDULE, sinon il deviendra FAILED_TO_SCHEDULE. Pour plus d'informations sur les statuts possibles d'un contact, consultez [AWS Ground Station statuts des contacts](#).

La `dataflowEndpointName` propriété d'une configuration de point de terminaison de flux de données indique quel point de terminaison de flux de données d'un groupe de points de terminaison de flux de données vers lequel ou depuis lequel les données seront transmises lors d'un contact.

La `dataflowEndpointRegion` propriété indique dans quelle région se trouve le point de terminaison du flux de données. Si une région est spécifiée dans la configuration de votre point de terminaison de flux de données, AWS Ground Station recherche un point de terminaison de flux de données dans la région spécifiée. Si aucune région n'est spécifiée, la région de la station au sol du contact AWS Ground Station sera sélectionnée par défaut. Un contact est considéré comme un contact de livraison de données interrégional si la région du point de terminaison de votre flux de données n'est pas la même que celle de la station au sol du contact. Voir [Travailler avec des flux de données](#) pour plus d'informations sur les flux de données entre régions.

Consultez [Utiliser les groupes AWS Ground Station de points de terminaison Dataflow](#) pour obtenir des conseils sur les avantages que peuvent apporter les différents schémas de dénomination de vos flux de données à votre cas d'utilisation.

Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les configurations des points de terminaison de flux de données à l'aide de l' CloudFormation API ou de l' AWS Command Line Interface API. AWS Ground Station

- [AWS::GroundStation::Config DataflowEndpointConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la `dataflowEndpointConfig` -> (structure) section)
- [DataflowEndpointConfig Référence d'API](#)

Config d'enregistrement Amazon S3

Note

Les configurations d'enregistrement Amazon S3 sont uniquement utilisées pour la livraison de données vers Amazon S3 et ne sont pas utilisées pour la livraison de données vers Amazon EC2.

Cette configuration représente un nœud de destination dans un flux de données. Ce nœud encapsulera les données entrantes provenant du nœud source du flux de données dans des données pcap. Pour des informations plus détaillées sur la façon de créer des flux de données avec cette configuration, voir [Travailler avec des flux de données](#)

Vous pouvez utiliser les configurations d'enregistrement S3 pour spécifier un compartiment Amazon S3 auquel vous souhaitez que les données descendantes soient transmises, ainsi que la convention de dénomination utilisée. Ce qui suit décrit les restrictions et les détails relatifs à ces paramètres :

- Le nom du compartiment Amazon S3 doit commencer par `aws-groundstation`.
- Le rôle IAM doit disposer d'une politique de confiance qui autorise le principal du `aws-groundstation.amazonaws.com` service à assumer ce rôle. Consultez la section [Exemple de politique de confiance](#) ci-dessous pour un exemple. Lors de la création de la configuration, l'identifiant de ressource de configuration n'existe pas, la politique de confiance doit utiliser un astérisque (*) à la place de `your-config-id` et peut être mise à jour après la création avec l'identifiant de ressource de configuration.

Exemple de politique de confiance

Pour plus d'informations sur la façon de mettre à jour la politique de confiance d'un rôle, consultez [la section Gestion des rôles IAM](#) dans le Guide de l'utilisateur IAM.

JSON

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

```

    "Principal": {
      "Service": "groundstation.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "999999999999"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:groundstation:us-
east-1:999999999999:config/s3-recording/your-config-id"
      }
    }
  }
]
}

```

- Le rôle IAM doit disposer d'une politique IAM qui lui permet d'exécuter l'`s3:GetBucketLocation`action sur le compartiment et l'`s3:PutObject`action sur les objets du compartiment. Si le compartiment Amazon S3 possède une politique de compartiment, celle-ci doit également autoriser le rôle IAM à effectuer ces actions. Consultez la section [Exemple de politique de rôle](#) ci-dessous pour un exemple.

Exemple de politique de rôle

Pour plus d'informations sur la façon de mettre à jour ou d'associer une politique de rôle, consultez [la section Gestion des politiques IAM](#) dans le Guide de l'utilisateur IAM.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation"
      ],
      "Resource": [

```

```

    "arn:aws:s3:::your-bucket-name"
  ],
},
{
  "Effect": "Allow",
  "Action": [
    "s3:PutObject"
  ],
  "Resource": [
    "arn:aws:s3:::your-bucket-name/*"
  ]
}
]
}

```

- Le préfixe sera utilisé pour nommer l'objet de données S3. Vous pouvez spécifier des clés facultatives pour la substitution. Ces valeurs seront remplacées par les informations correspondantes figurant dans vos coordonnées. Par exemple, le préfixe de `{satellite_id}/{year}/{month}/{day}` sera remplacé et donnera une sortie comme `fake_satellite_id/2021/01/10`

Clés facultatives pour la substitution : `{satellite_id} {config-name} {config-id} | {year} | {month} | {day} |`

Consultez la documentation suivante pour plus d'informations sur la façon d'effectuer des opérations sur les configurations d'enregistrement S3 à l'aide CloudFormation de l' AWS Command Line Interface API ou de l' AWS Ground Station API.

- [AWS::GroundStation::Config RecordingConfig CloudFormation Propriété S3](#)
- [AWS CLI Référence de configuration](#) (voir la `s3RecordingConfig` -> (structure) section)
- [Référence de RecordingConfig l'API S3](#)

Config du récepteur de télémétrie

Vous pouvez utiliser les configurations du récepteur de télémétrie pour spécifier où vous souhaitez que les données de télémétrie soient transmises lors des contacts par satellite. Une configuration de récepteur de télémétrie est facultative et est ajoutée à votre profil de mission pour planifier des

contacts compatibles avec la télémétrie. Ce qui suit décrit les restrictions et les détails relatifs à ces paramètres :

- Le rôle IAM doit disposer d'une politique de confiance qui autorise le principal du `groundstation.amazonaws.com` service à assumer ce rôle. Consultez la section [Exemple de politique de confiance](#) ci-dessous pour un exemple.

Exemple de politique de confiance

Pour plus d'informations sur la façon de mettre à jour la politique de confiance d'un rôle, consultez [la section Gestion des rôles IAM](#) dans le Guide de l'utilisateur IAM.

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

- Le rôle IAM doit disposer d'une politique IAM qui lui permet d'exécuter les `kinesis:PutRecords` actions `kinesis:PutRecord` et de `kinesis:DescribeStream` les exécuter sur le flux. Consultez la section [Exemple de politique de rôle](#) ci-dessous pour un exemple.

Exemple de politique de rôle

Pour plus d'informations sur la façon de mettre à jour ou d'associer une politique de rôle, consultez [la section Gestion des politiques IAM](#) dans le Guide de l'utilisateur IAM.

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

```
    "kinesis:DescribeStream",
    "kinesis:PutRecord",
    "kinesis:PutRecords"
  ],
  "Resource": "arn:aws:kinesis:us-east-2:999999999999:stream/your-stream-name"
}
]
```

Lorsque vous incluez une configuration de récepteur de télémétrie dans votre profil de mission, les données de télémétrie AWS Ground Station seront diffusées sur votre compte lors des contacts. Pour plus d'informations sur les types de télémétrie, le format des données et la configuration AWS des ressources nécessaires, consultez [Travailler avec la télémétrie](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les configurations des récepteurs de télémétrie à l'aide CloudFormation de l'API ou de l' AWS Command Line Interface API. AWS Ground Station

- [AWS::GroundStation::Config TelemetrySinkConfig CloudFormation propriété](#)
- [AWS CLI Référence de configuration](#) (voir la telemetrySinkConfig -> (structure) section)
- [TelemetrySinkConfig Référence d'API](#)

Utiliser les groupes AWS Ground Station de points de terminaison Dataflow

Les points de terminaison du flux de données définissent l'emplacement vers lequel vous souhaitez que les données soient diffusées de manière synchrone pendant les contacts. Les points de terminaison de flux de données sont toujours créés dans le cadre d'un groupe de points de terminaison de flux de données. Si vous incluez plusieurs points de terminaison de flux de données dans un groupe, cela signifie que les points de terminaison spécifiés peuvent tous être utilisés conjointement au cours d'un seul contact. Par exemple, si un contact a besoin d'envoyer des données vers trois points de terminaison de flux de données distincts, vous devez disposer de trois points de terminaison dans un seul groupe de points de terminaison du flux de données qui correspondent aux configs de point de terminaison de flux de données de votre profil de mission.

Versions des groupes de points de terminaison Dataflow

AWS Ground Station prend en charge deux versions des groupes de points de terminaison de flux de données :

- [DataflowEndpointGroup- L'implémentation d'origine qui prend en charge les liaisons montantes et descendantes à l'aide d'un point de terminaison de flux de données, et la liaison descendante uniquement pour un point de terminaison agent AWS Ground Station](#)
- DataflowEndpointGroupV2 - Version mise à jour qui prend en charge les flux de données en liaison montante et en liaison descendante pour les points de terminaison de AWS Ground Station l'agent avec une clarté et des fonctionnalités améliorées

Comparaison des groupes de points de terminaison de Dataflow

Fonctionnalité	DataflowEndpointGroup	DataflowEndpointGroupV2
Types de terminaux pris en charge	DataflowEndpoint, AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint, UplinkAwsGroundStationAgentEndpoint
Points de terminaison prenant en charge la liaison montante	DataflowEndpoint	UplinkAwsGroundStationAgentEndpoint
Points de terminaison prenant en charge la liaison descendante	DataflowEndpoint, AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint

DataflowEndpointGroupLa V2 a été créée pour prendre en charge les flux de données en liaison montante et pour clarifier le langage entourant les groupes de points de terminaison des flux de données. Nous recommandons d'utiliser [UplinkAwsGroundStationAgentEndpoint](#)des [DownlinkAwsGroundStationAgentEndpoint](#)terminaux dotés d'une [DataflowEndpointGroupversion V2](#) pour tous les nouveaux cas d'utilisation. DataflowEndpointGroup reste compatible pour des raisons de rétrocompatibilité, mais la DataflowEndpointGroup version 2 fournit des fonctionnalités améliorées et des options de configuration plus claires.

Tip

Les points de terminaison du flux de données sont identifiés par le nom que vous avez choisi lors de l'exécution des contacts. Il n'est pas nécessaire que ces noms soient uniques pour l'ensemble du compte. Cela permet d'exécuter plusieurs contacts entre différents satellites et antennes en même temps en utilisant le même profil de mission. Cela peut être

utile si vous avez une constellation de satellites présentant les mêmes caractéristiques de fonctionnement. Vous pouvez augmenter le nombre de groupes de points de terminaison du flux de données pour qu'il corresponde au nombre maximal de contacts simultanés dont votre constellation de satellites a besoin.

Lorsqu'une ou plusieurs ressources d'un groupe de points de terminaison de flux de données sont en cours d'utilisation pour un contact, l'ensemble du groupe est réservé pendant toute la durée du contact. Vous pouvez exécuter plusieurs contacts simultanément, mais ces contacts doivent être exécutés sur différents groupes de points de terminaison de flux de données.

Important

Les groupes de points de terminaison Dataflow doivent être en HEALTHY état de planifier les contacts qui les utilisent. Pour plus d'informations sur la façon de résoudre les problèmes liés aux groupes de points de terminaison de flux de données qui ne sont pas dans un HEALTHY état, consultez. [Le dépannage DataflowEndpointGroups n'est pas dans un état SAIN](#)

Consultez la documentation suivante pour plus d'informations sur la manière d'effectuer des opérations sur les groupes de points de terminaison de flux de données à l'aide CloudFormation de l'API AWS Command Line Interface ou de l' AWS Ground Station API.

- [AWS::GroundStation::DataflowEndpointGroup CloudFormation type de ressource](#)
- [Référence du groupe Dataflow Endpoint AWS CLI](#)
- [Référence de l'API Dataflow Endpoint Group](#)

Points de terminaison de flux de données

Les membres d'un groupe de points de terminaison de flux de données sont des points de terminaison de flux de données. Les types de points de terminaison pris en charge dépendent de la version du groupe de points de terminaison de flux de données que vous utilisez.

DataflowEndpointGroup points de terminaison

DataflowEndpointGroup [prend en charge les liaisons montantes et descendantes à l'aide d'un point de terminaison de flux de données, et la liaison descendante uniquement pour un point de](#)

[terminaison d'agent.AWS Ground Station](#) Pour les deux types de points de terminaison, vous allez créer les structures de support (par exemple, les adresses IP) avant de créer le groupe de points de terminaison du flux de données. Consultez [Travailler avec des flux de données](#) les recommandations relatives au type de point de terminaison du flux de données à utiliser et à la manière de configurer les structures de support.

Les sections suivantes décrivent les deux types de points de terminaison pris en charge.

Important

Tous les points de terminaison de flux de données au sein d'un même groupe de points de terminaison de flux de données doivent être du même type. Vous ne pouvez pas mélanger les points de [terminaison de AWS Ground Station l'agent avec les points de terminaison Dataflow dans le même](#) groupe. Si votre cas d'utilisation nécessite les deux types de points de terminaison, vous devez créer des groupes de points de terminaison de flux de données distincts pour chaque type.

Pour la DataflowEndpointGroup V2, vous pouvez mixer

[UplinkAwsGroundStationAgentEndpoint](#) et [DownlinkAwsGroundStationAgentEndpoint](#) dans le même groupe.

AWS Ground Station Point final de l'agent

Le point de terminaison de l' AWS Ground Station agent utilise l' AWS Ground Station agent en tant que composant logiciel pour mettre fin aux connexions. Pour créer un point de terminaison d' AWS Ground Station agent, vous devez uniquement renseigner le `AwsGroundStationAgentEndpoint` champ du `EndpointDetails`. Pour plus d'informations sur l' AWS Ground Station agent, consultez le [guide d'utilisation complet de l'AWS Ground Station agent](#).

`AwsGroundStationAgentEndpoint` comprend les éléments suivants :

- `Name`- Le nom du point de terminaison du flux de données. Pour que le contact puisse utiliser ce point de terminaison de flux de données, ce nom doit correspondre au nom utilisé dans la configuration de votre point de terminaison de flux de données.
- `EgressAddress`- L'adresse IP et l'adresse du port utilisés pour faire sortir les données de l'agent.
- `IngressAddress`- L'adresse IP et l'adresse du port utilisés pour saisir les données vers l'agent.

Point de terminaison du flux de données

Le point de terminaison Dataflow utilise une application réseau en tant que composant logiciel pour mettre fin aux connexions. Utilisez Dataflow Endpoint lorsque vous souhaitez établir une liaison montante avec des données de signal numérique, une liaison descendante avec moins de 50 % des données de signal numérique ou une liaison descendante avec MHz des données de signal. Demodulated/Decoded Pour créer un point de terminaison de flux de données, vous devez renseigner les Security Details champs Endpoint et du. EndpointDetails

Endpoint comprend les éléments suivants :

- **Name**- Le nom du point de terminaison du flux de données. Pour que le contact puisse utiliser ce point de terminaison de flux de données, ce nom doit correspondre au nom utilisé dans la configuration de votre point de terminaison de flux de données.
- **Address**- L'adresse IP et l'adresse du port utilisés.

SecurityDetails comprend les éléments suivants :

- **roleArn**- Le nom de ressource Amazon (ARN) d'un rôle qui AWS Ground Station sera chargé de créer des interfaces réseau élastiques (ENIs) dans votre VPC. Ils ENIs servent de points d'entrée et de sortie des données diffusées lors d'un contact.
- **securityGroupIds** - Les groupes de sécurité à attacher aux interfaces réseau Elastic.
- **subnetIds**- Une liste de sous-réseaux dans lesquels AWS Ground Station peuvent être placées des interfaces réseau élastiques pour envoyer des flux à vos instances. Si plusieurs sous-réseaux sont spécifiés, ils doivent être routables les uns vers les autres. Si les sous-réseaux se trouvent dans des zones de disponibilité différentes (AZs), des frais de transfert de données inter-AZ peuvent vous être facturés.

Le rôle IAM transféré `roleArn` doit avoir une politique de confiance qui permet au principal du `groundstation.amazonaws.com` service d'assumer le rôle. Consultez la section [Exemple de politique de confiance](#) ci-dessous pour un exemple. Lors de la création du point de terminaison, l'identifiant de ressource du point de terminaison n'existe pas. La politique de confiance doit donc utiliser un astérisque (*) à la place de *your-endpoint-id*. Il peut être mis à jour après sa création pour utiliser l'identifiant de ressource du point de terminaison afin d'étendre la politique de confiance à ce groupe de points de terminaison de flux de données spécifique.

Le rôle IAM doit disposer d'une politique IAM qui permet AWS Ground Station de configurer le. ENIs Consultez la section [Exemple de politique de rôle](#) ci-dessous pour un exemple.

Exemple de politique de confiance

Pour plus d'informations sur la façon de mettre à jour la politique de confiance d'un rôle, consultez [la section Gestion des rôles IAM](#) dans le Guide de l'utilisateur IAM.

JSON

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

Exemple de politique de rôle

Pour plus d'informations sur la façon de mettre à jour ou d'associer une politique de rôle, consultez [la section Gestion des politiques IAM](#) dans le Guide de l'utilisateur IAM.

JSON

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

DataflowEndpointGroupPoints de terminaison V2

DataflowEndpointGroupLa V2 introduit des types de terminaux spécialisés qui fournissent une configuration plus claire et des fonctionnalités améliorées :

- [UplinkAwsGroundStationAgentEndpoint](#)- Optimisé pour les flux de données en liaison montante
- [DownlinkAwsGroundStationAgentEndpoint](#)- Optimisé pour les flux de données en liaison descendante

Ces points de terminaison spécialisés remplacent les configurations génériques

[AwsGroundStationAgentEndpoint](#) par des configurations spécifiques aux directions qui facilitent la configuration et la gestion de vos flux de données.

Point de terminaison Uplink AWS Ground Station Agent

[UplinkAwsGroundStationAgentEndpoint](#) Il est spécialement conçu pour les flux de données en liaison montante et fournit des options de configuration plus claires. Utilisez ce type de point de terminaison lorsque vous devez fournir des données AWS Ground Station à relier à votre satellite.

[UplinkAwsGroundStationAgentEndpoint](#) comprend les éléments suivants :

- **Name**- Le nom du point de terminaison du flux de données. Pour que le contact puisse utiliser ce point de terminaison de flux de données, ce nom doit correspondre au nom utilisé dans la configuration de votre point de terminaison de flux de données.
- **IngressAddressAndPort**- Adresse IP et adresse de port uniques pour la saisie des données vers l'agent
- **AgentIpAndPortAddress**- Portée de ports pour la communication avec les agents

Point de terminaison de l' AWS Ground Station agent Downlink

[DownlinkAwsGroundStationAgentEndpoint](#) est optimisé pour les flux de données en liaison descendante, y compris les scénarios de liaison descendante à bande étroite, de démodulation/décodage à large bande et d'écho en liaison montante.

`DownlinkAwsGroundStationAgentEndpoint` comprend les éléments suivants :

- **Name**- Le nom du point de terminaison du flux de données. Pour que le contact puisse utiliser ce point de terminaison de flux de données, ce nom doit correspondre au nom utilisé dans la configuration de votre point de terminaison de flux de données.
- **EgressAddressAndPort**- Adresse IP et adresse de port uniques pour la sortie des données par l'agent
- **AgentIpAndPortAddress**- Portée de ports pour la communication avec les agents

Création de groupes de points de terminaison de flux de données

Vous pouvez créer des groupes de points de terminaison de flux de données en utilisant l'une des versions suivantes :

`CreateDataflowEndpointGroup`

[CreateDataflowEndpointGroup](#) À utiliser pour des raisons de rétrocompatibilité ou lorsque vous devez utiliser le générique [AwsGroundStationAgentEndpoint](#) ou [DataflowEndpoint](#) les types.

`CreateDataflowEndpointGroupV2`

Utilisez la [CreateDataflowEndpointGroupversion V2](#) pour les nouvelles implémentations afin de tirer parti des types de points de terminaison spécialisés qui prennent en charge les flux de données en liaison montante et en liaison descendante. Cette API ne prend en charge que [UplinkAwsGroundStationAgentEndpoint](#) et [DownlinkAwsGroundStationAgentEndpoint](#).

Considérations concernant la migration

Si vous utilisez actuellement `DataflowEndpointGroup`, vous pouvez continuer à utiliser votre configuration existante sans modification. AWS Ground Station maintient une rétrocompatibilité totale.

Si vous souhaitez effectuer la migration pour utiliser la nouvelle `DataflowEndpointGroup` version 2 et que vous utilisez actuellement une application [DataflowEndpoint](#) dotée d'un point de terminaison `Dataflow` pour recevoir vos données, vous devrez effectuer la migration pour utiliser l' AWS Ground Station agent à la place. Si vous utilisez déjà un AWS Ground Station agent pour la liaison descendante, vous pouvez également utiliser la même instance d'agent pour la liaison montante. Aucune instance d'agent supplémentaire n'est requise.

Pour migrer vers la `DataflowEndpointGroup` version V2 :

1. En cas de migration depuis `DataflowEndpoint`, configurez l' AWS Ground Station agent en suivant le guide de l'[utilisateur de l'AWS Ground Station agent](#)
2. Identifiez la direction de votre flux de données et créez le type de point de terminaison approprié (ou) [UplinkAwsGroundStationAgentEndpointDownlinkAwsGroundStationAgentEndpoint](#)
3. Créez le [DataflowEndpointGroupV2](#) référençant ces points de terminaison
4. Créez une nouvelle [configuration de point de terminaison de flux de données](#) qui fait référence à la nouvelle `DataflowEndpointGroup` V2 par son nom
5. Créez un nouveau profil de mission qui fait référence à la configuration du point de terminaison du flux de données en tant que périphérie du flux de données
6. Utilisez le nouveau profil de mission pour planifier des contacts
7. Testez votre configuration avant de la déployer en production

Pour plus d'informations sur le flux de travail complet, reportez-vous [Comprendre les composants AWS Ground Station de base](#) aux sections et [Création de configurations](#).

Utiliser l' AWS Ground Station agent

L' AWS Ground Station agent vous permet de recevoir (liaison descendante) des flux de données synchrones à fréquence intermédiaire numérique à large bande (DigIF) lors des contacts avec AWS Ground Station.

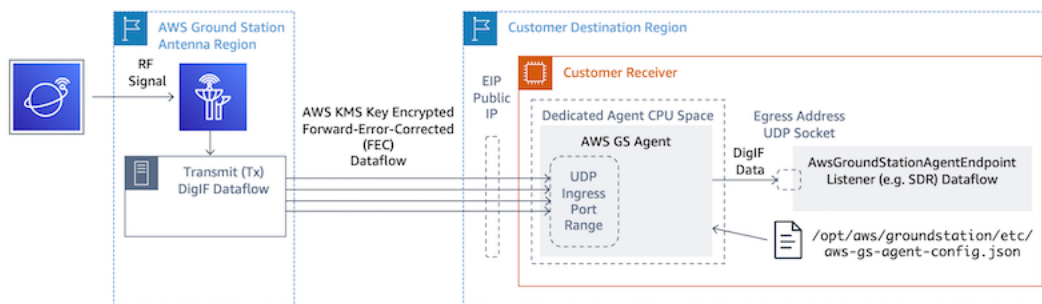
Comment ça marche

Vous pouvez sélectionner deux options pour la livraison des données :

1. Livraison de données à une EC2 instance : livraison de données à une EC2 instance dont vous êtes le propriétaire. Vous gérez l' AWS Ground Station agent. Cette option peut vous convenir le mieux si vous avez besoin d'un traitement des données en temps quasi réel. Consultez la [Travailler avec des flux de données](#) section pour plus d'informations sur la livraison EC2 des données.
2. Livraison des données vers un compartiment S3 - La livraison des données vers votre compartiment AWS S3 est entièrement gérée par AWS Ground Station. Consultez le [Mise en route](#) guide pour plus d'informations sur la livraison de données S3.

Les deux modes de livraison de données nécessitent que vous créiez un ensemble de ressources AWS. L'utilisation de CloudFormation pour créer vos ressources AWS est vivement recommandée afin de garantir la fiabilité, la précision et la facilité de prise en charge. Chaque contact peut uniquement transmettre des données à EC2 S3, mais pas aux deux simultanément.

Le schéma suivant montre un flux de données DigIF d'une région d' AWS Ground Station antenne vers votre EC2 instance avec votre radio définie par logiciel (SDR) ou un écouteur similaire.



Informations supplémentaires

Pour des informations plus détaillées, consultez le [guide de l'utilisateur complet de l'AWS Ground Station agent](#).

Mise en route

Avant de commencer, vous devez vous familiariser avec les concepts de base de AWS Ground Station. Pour de plus amples informations, veuillez consulter [Comment AWS Ground Station fonctionne](#).

Vous trouverez ci-dessous les meilleures pratiques pour Gestion des identités et des accès AWS (IAM) et les autorisations dont vous aurez besoin. Après avoir configuré les rôles appropriés, vous pouvez commencer à suivre les étapes restantes.

Inscrivez-vous pour un Compte AWS

Si vous n'en avez pas Compte AWS, procédez comme suit pour en créer un.

Pour vous inscrire à un Compte AWS

1. Ouvrez l'<https://portal.aws.amazon.com/billing/inscription>.
2. Suivez les instructions en ligne.

Dans le cadre de la procédure d'inscription, vous recevrez un appel téléphonique ou un SMS et vous saisissez un code de vérification en utilisant le clavier numérique du téléphone.

Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWS est créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des Services AWS et des ressources de ce compte. La meilleure pratique de sécurité consiste à attribuer un accès administratif à un utilisateur, et à utiliser uniquement l'utilisateur racine pour effectuer les [tâches nécessitant un accès utilisateur racine](#).

AWS vous envoie un e-mail de confirmation une fois le processus d'inscription terminé. À tout moment, vous pouvez consulter l'activité actuelle de votre compte et gérer votre compte en accédant à <https://aws.amazon.com/> et en choisissant Mon compte.

Création d'un utilisateur doté d'un accès administratif

Une fois que vous vous êtes inscrit à un utilisateur administratif Compte AWS, que vous Utilisez racine d'un compte AWS l'avez sécurisé AWS IAM Identity Center, que vous l'avez activé et que vous en avez créé un, afin de ne pas utiliser l'utilisateur root pour les tâches quotidiennes.

Sécurisez votre Utilisateur racine d'un compte AWS

1. Connectez-vous en [AWS Management Console](#) tant que propriétaire du compte en choisissant Utilisateur root et en saisissant votre adresse Compte AWS e-mail. Sur la page suivante, saisissez votre mot de passe.

Pour obtenir de l'aide pour vous connecter en utilisant l'utilisateur racine, consultez [Connexion en tant qu'utilisateur racine](#) dans le Guide de l'utilisateur Connexion à AWS .

2. Activez l'authentification multifactorielle (MFA) pour votre utilisateur racine.

Pour obtenir des instructions, consultez la section [Activer un périphérique MFA virtuel pour votre utilisateur Compte AWS root \(console\)](#) dans le guide de l'utilisateur IAM.

Création d'un utilisateur doté d'un accès administratif

1. Activez IAM Identity Center.

Pour obtenir des instructions, consultez [Activation d' AWS IAM Identity Center](#) dans le Guide de l'utilisateur AWS IAM Identity Center .

2. Dans IAM Identity Center, octroyez un accès administratif à un utilisateur.

Pour un didacticiel sur l'utilisation du Répertoire IAM Identity Center comme source d'identité, voir [Configurer l'accès utilisateur par défaut Répertoire IAM Identity Center](#) dans le Guide de AWS IAM Identity Center l'utilisateur.

Connexion en tant qu'utilisateur doté d'un accès administratif

- Pour vous connecter avec votre utilisateur IAM Identity Center, utilisez l'URL de connexion qui a été envoyée à votre adresse e-mail lorsque vous avez créé l'utilisateur IAM Identity Center.

Pour obtenir de l'aide pour vous connecter en utilisant un utilisateur d'IAM Identity Center, consultez la section [Connexion au portail AWS d'accès](#) dans le guide de l'Connexion à AWS utilisateur.

Attribution d'un accès à d'autres utilisateurs

1. Dans IAM Identity Center, créez un ensemble d'autorisations qui respecte la bonne pratique consistant à appliquer les autorisations de moindre privilège.

Pour obtenir des instructions, consultez [Création d'un ensemble d'autorisations](#) dans le Guide de l'utilisateur AWS IAM Identity Center .

2. Attribuez des utilisateurs à un groupe, puis attribuez un accès par authentification unique au groupe.

Pour obtenir des instructions, consultez [Ajout de groupes](#) dans le Guide de l'utilisateur AWS IAM Identity Center .

Ajoutez AWS Ground Station des autorisations à votre AWS compte

Pour l'utiliser AWS Ground Station sans avoir besoin d'un utilisateur administratif, vous devez créer une nouvelle politique et l'associer à votre AWS compte.

1. Connectez-vous à la [console IAM AWS Management Console](#) et ouvrez-la.
2. Créez une stratégie. Procédez comme suit :
 - a. Dans le panneau de navigation, choisissez Politiques (Politiques), puis Create Policy (Créer une politique).
 - b. Dans l'onglet JSON modifiez le fichier JSON avec l'une des valeurs suivantes. Utilisez le fichier JSON qui fonctionne le mieux pour votre application.
 - Pour les privilèges administratifs de Ground Station, définissez Action sur groundstation : *

JSON

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

```

    }
  ]
}

```

- Pour des autorisations en lecture seule (Read-only), définissez Action sur `groundstation:get*`, `groundstation:list*` et `groundstation:describe*` comme suit :

JSON

```

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

```

- Pour une sécurité accrue grâce à l'authentification multifactorielle, définissez Action sur `groundstation : *` et Condition/Bool sur `aws ::true` comme suit : `MultiFactorAuthPresent`

JSON

```

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

```

```
}  
  }  
] }  
}
```

3. Dans la console IAM, attachez la politique que vous avez créée à l'utilisateur souhaité.

Pour plus d'informations sur la création d'utilisateurs IAM et l'attachement de stratégies, consultez le [Guide de l'utilisateur IAM](#).

satellite embarqué

L'intégration d'un satellite AWS Ground Station est un processus en plusieurs étapes impliquant la collecte de données, la validation technique, l'octroi de licences de spectre, ainsi que l'intégration et les tests. Des accords de confidentialité (NDAs) sont également requis.

Vue d'ensemble du processus d'intégration des clients

L'intégration des satellites est un processus manuel qui se trouve dans la section [Satellites et ressources](#) de la page de AWS Ground Station console. Ce qui suit décrit le processus global.

1. Consultez la [AWS Ground Station Succursales](#) section pour déterminer si votre satellite répond aux caractéristiques géographiques et aux caractéristiques de radiofréquence.
2. Pour commencer à intégrer votre satellite AWS Ground Station, veuillez envoyer un questionnaire d'intégration par satellite dans la section [Satellites et ressources](#) de la page de la AWS Ground Station console. Veuillez inclure un bref résumé de votre mission et de vos besoins en matière de satellites, y compris le nom de votre organisation, les fréquences requises, la date à laquelle les satellites seront ou ont été lancés, le type d'orbite du satellite et si vous prévoyez de les utiliser [Utilisez la fonction de jumelage AWS Ground Station numérique](#).
3. Une fois votre demande examinée et approuvée, vous AWS Ground Station demanderez une licence réglementaire aux sites spécifiques que vous prévoyez d'utiliser. La durée de cette étape varie en fonction des lieux et des réglementations en vigueur.
4. Une fois cette approbation obtenue, votre satellite sera visible pour que vous puissiez l'utiliser. AWS Ground Station vous enverra une notification vous informant de la réussite de la mise à jour.

(Facultatif) Dénomination des satellites

Après l'intégration, vous souhaitez peut-être ajouter un nom à votre enregistrement satellite pour le reconnaître plus facilement. La AWS Ground Station console peut afficher un nom défini par l'utilisateur pour un satellite ainsi que l'identifiant Norad lorsque vous utilisez la page Contacts. L'affichage du nom du satellite facilite grandement la sélection du bon satellite lors de la planification. Pour ce faire, des [tags](#) peuvent être utilisés.

Le balisage des satellites AWS Ground Station peut être effectué via l'API [tag-resource](#) avec l'interface de ligne de commande AWS ou l'une des interfaces AWS. SDKs Ce guide décrit l'utilisation de la AWS Ground Station CLI pour étiqueter le satellite de diffusion public Aqua (Norad ID 27424).
us-west-2

AWS Ground Station INTERFACE DE LIGNE DE COMMANDE (CLI)

Ils AWS CLI peuvent être utilisés pour interagir avec AWS Ground Station. Avant AWS CLI de pouvoir étiqueter vos satellites, les AWS CLI conditions suivantes doivent être remplies :

- Assurez-vous qu'il AWS CLI est installé. Pour plus d'informations sur l'installation AWS CLI, consultez [Installation de la version 2 de l'interface de ligne de commande AWS](#).
- Assurez-vous qu'il AWS CLI est configuré. Pour plus d'informations sur la configuration AWS CLI, consultez [Configuration de l'interface de ligne de commande AWS version 2](#).
- Enregistrez vos paramètres de configuration utilisés fréquemment et vos informations d'identification dans les fichiers qui sont gérés par l' AWS CLI. Vous avez besoin de ces paramètres et informations d'identification pour réserver et gérer vos AWS Ground Station contacts avec AWS CLI. Pour plus d'informations sur l'enregistrement de votre configuration et de vos paramètres d'identification, consultez la section Paramètres des [fichiers de configuration et d'identification](#).

Une fois AWS CLI configuré et prêt à être utilisé, consultez la page de [référence des commandes de l'AWS Ground Station CLI](#) pour vous familiariser avec les commandes disponibles. Suivez la structure de AWS CLI commande lorsque vous utilisez ce service et préfixez vos commandes avec `groundstation` pour indiquer le service AWS Ground Station que vous souhaitez utiliser. Pour plus d'informations sur la structure de AWS CLI commande, consultez [Structure de commande sur la page de l'AWS CLI](#). Un exemple de structure de commande est fourni ci-dessous.

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

Nommer un satellite

Vous devez d'abord obtenir l'ARN du ou des satellites que vous souhaitez étiqueter. Cela peut être fait via l'API [list-satellites](#) dans l'AWS CLI :

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

L'exécution de la commande CLI ci-dessus renverra un résultat similaire à celui-ci :

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

Trouvez le satellite que vous souhaitez étiqueter et notez le `satelliteArn`. [Une mise en garde importante concernant le balisage est que l'API `tag-resource` nécessite un ARN régional et que l'ARN renvoyé par `list-satellites` est mondial.](#) À l'étape suivante, vous devez augmenter l'ARN avec la région dans laquelle vous souhaitez voir le tag (probablement la région dans laquelle vous planifiez). Pour cet exemple, nous utilisons `us-west-2`. Avec cette modification, l'ARN passera de :

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

par :

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

Pour afficher le nom du satellite dans la console, le satellite doit avoir une étiquette avec "Name" comme clé. De plus, étant donné que nous utilisons le AWS CLI, les guillemets doivent être masqués par une barre oblique inverse. Le tag ressemblera à ceci :

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

Ensuite, vous allez appeler l'API [tag-resource](#) pour étiqueter le satellite. Cela peut être fait avec des AWS CLI méthodes similaires :

```
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\"}'
```

Après cela, vous pourrez voir le nom que vous avez défini pour le satellite dans la AWS Ground Station console.

Changer le nom d'un satellite

Si vous souhaitez modifier le nom d'un satellite, vous pouvez simplement appeler à nouveau [tag-resource](#) avec l'ARN du satellite avec la même "Name" clé, mais avec une valeur différente dans la balise. Cela mettra à jour le tag existant et affichera le nouveau nom dans la console. Voici un exemple d'appel pour cela :

```
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\"}'
```

Supprimer le nom d'un satellite

Le nom défini pour un satellite peut être supprimé à l'aide de l'API [untag-resource](#). Cette API a besoin de l'ARN satellite avec la région dans laquelle se trouve la balise et d'une liste de clés de balise. Pour le nom, la clé du tag est "Name". Voici un exemple d'appel à cette API à l'aide de l'AWS CLI :

```
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
```

Satellites de diffusion publics

Outre l'embarquement de vos propres satellites, vous pouvez demander à embarquer avec des satellites de diffusion publics compatibles qui fournissent une voie de communication descendante accessible au public. Cela vous permet de les utiliser AWS Ground Station pour transférer les données de ces satellites.

Note

Vous ne pourrez pas établir de liaison ascendante vers ces satellites. Vous ne pourrez utiliser que les voies de communication descendantes accessibles au public.

AWS Ground Station prend en charge l'intégration des satellites suivants pour la liaison descendante des données de diffusion directe :

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

Une fois embarqués, ces satellites sont accessibles pour une utilisation immédiate. AWS Ground Station gère un certain nombre de CloudFormation modèles préconfigurés pour faciliter la prise en main du service. Voir [Exemples de configurations de profil de mission](#) des exemples AWS Ground Station d'utilisation.

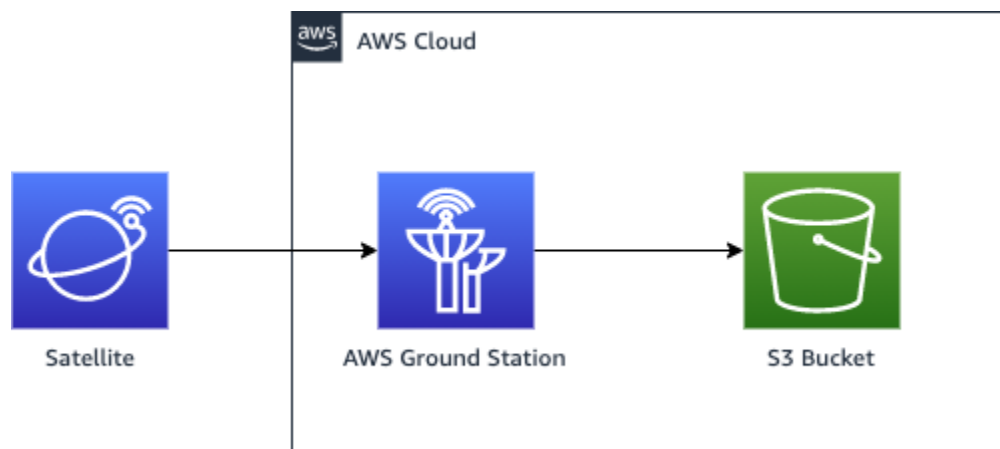
Pour de plus amples informations sur ces satellites et le type de données qu'ils transmettent, veuillez consulter [Aqua](#) et [JPSS-1/NOAA-20 et SNPP](#) et [Terra](#).

Planifiez les voies de communication de votre flux de données

Vous avez le choix entre une communication synchrone ou asynchrone pour chaque voie de communication de votre satellite. En fonction de votre satellite et de votre cas d'utilisation, vous pouvez avoir besoin de l'un ou des deux types. Les voies de communication synchrones permettent des opérations de liaison montante en temps quasi réel ainsi que des opérations de liaison descendante à bande étroite et à large bande. Les voies de communication asynchrones prennent uniquement en charge les opérations de liaison descendante à bande étroite et à large bande.

Livraison de données asynchrone

Avec la livraison de données vers Amazon S3, vos données de contact sont transmises de manière asynchrone à un compartiment Amazon S3 de votre compte. Vos données de contact sont fournies sous forme de fichiers de capture de paquets (pcap) pour permettre de rejouer les données de contact dans une radio définie par logiciel (SDR) ou pour extraire les données de charge utile des fichiers pcap à des fins de traitement. Les fichiers pcap sont envoyés à votre compartiment Amazon S3 toutes les 30 secondes à mesure que les données de contact sont reçues par le matériel de l'antenne pour permettre le traitement des données de contact pendant le contact, si vous le souhaitez. Une fois reçues, vous pouvez traiter les données à l'aide de votre propre logiciel de post-traitement ou utiliser d'autres services AWS tels qu'Amazon SageMaker AI ou Amazon Rekognition. La livraison de données vers Amazon S3 n'est disponible que pour la liaison descendante des données depuis votre satellite ; il n'est pas possible de relier des données vers votre satellite depuis Amazon S3.



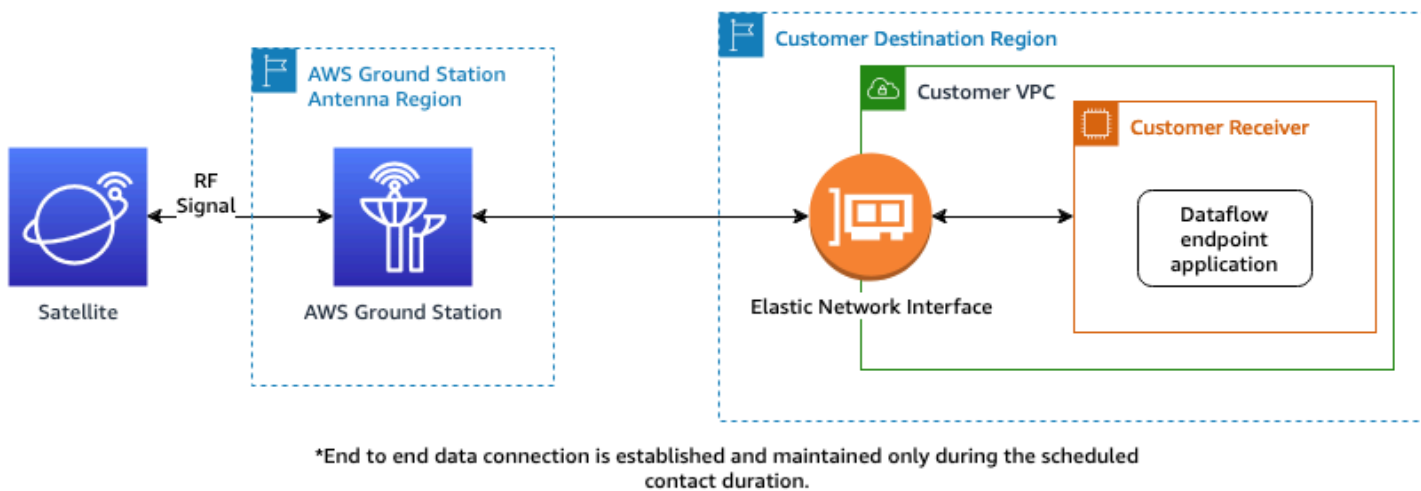
Pour utiliser ce chemin, vous devez créer un compartiment Amazon S3 dans AWS Ground Station auquel envoyer les données. À l'étape suivante, vous devrez également créer une configuration d'enregistrement S3 à l'étape suivante. Consultez le [Config d'enregistrement Amazon S3](#) pour connaître les restrictions relatives à la dénomination des compartiments et pour savoir comment spécifier la convention de dénomination utilisée pour vos fichiers.

Livraison synchrone des données

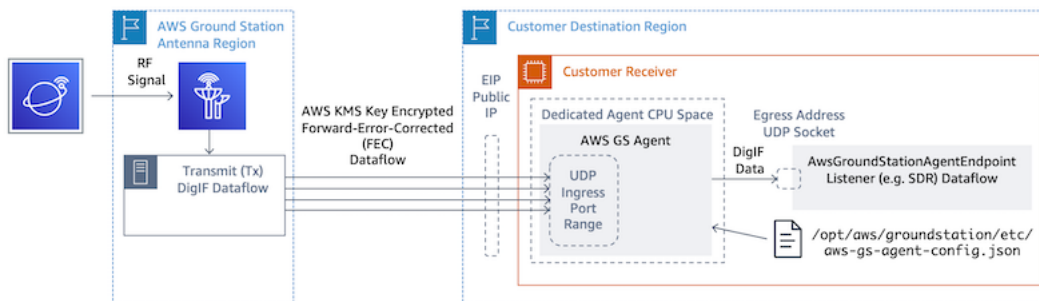
Avec la livraison de données vers Amazon EC2, vos données de contact sont diffusées vers et depuis votre instance Amazon EC2. Vous pouvez traiter vos données en temps réel sur votre instance Amazon EC2 ou les transférer pour un post-traitement.

Pour utiliser un chemin synchrone, vous devez configurer vos instances Amazon EC2 et créer un ou plusieurs groupes de points de terminaison de flux de données. Pour configurer votre instance Amazon EC2, reportez-vous au [Configurer et configurer Amazon EC2](#) Pour créer votre groupe de points de terminaison Dataflow, veuillez vous référer au [Utiliser les groupes AWS Ground Station de points de terminaison Dataflow](#)

Le tableau suivant indique le chemin de communication si vous utilisez la configuration du point de terminaison du flux de données.



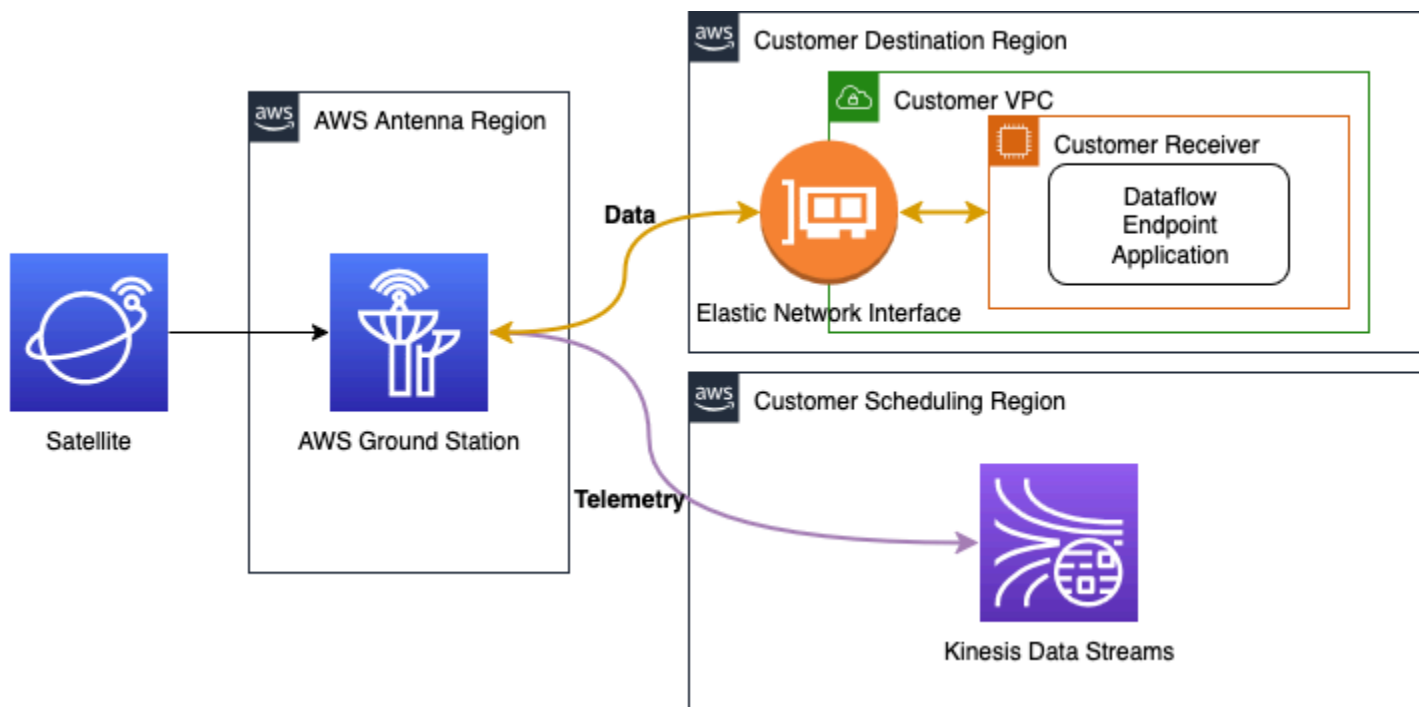
Le tableau suivant indique le chemin de communication si vous utilisez la configuration de l' AWS Ground Station agent.



Planifiez votre télémétrie

AWS Ground Station la télémétrie est une fonctionnalité optionnelle qui diffuse les métriques des AWS Ground Station antennes vers votre AWS compte lors des contacts par satellite. Cela vous permet de surveiller les performances des contacts en temps quasi réel et de créer des solutions de surveillance personnalisées.

Grâce à la AWS Ground Station télémétrie, les mesures des AWS Ground Station antennes sont diffusées directement sur votre compte. Les données de télémétrie commencent à être diffusées au début du contact et se poursuivent pendant toute la durée du contact. Les données de télémétrie sont transmises à votre compte en temps quasi réel au fur et à mesure qu'elles sont échantillonnées à partir du matériel d'antenne. Une fois reçues, vous pouvez traiter les données à l'aide de votre propre logiciel de post-traitement ou utiliser d'autres services AWS tels qu'Amazon Data AWS Lambda Firehose ou.



À l'étape suivante, vous allez créer les configurations nécessaires à votre profil de mission. Si vous souhaitez activer la télémétrie, vous allez créer une configuration de récepteur de télémétrie en plus de votre configuration de suivi et de vos configurations de flux de données. Pour des instructions de configuration détaillées, voir [Configuration de la télémétrie](#).

Pour plus d'informations sur TelemetrySinkConfig, voir [Config du récepteur de télémétrie](#).

Création de configurations

À cette étape, vous avez identifié le satellite, les voies de communication et les ressources IAM, Amazon EC2 et Amazon S3 selon vos besoins. Au cours de cette étape, vous allez créer AWS Ground Station des configurations qui stockent leurs paramètres respectifs.

Configurations de livraison de données

Les premières configurations à créer concernent l'endroit et la manière dont vous souhaitez que les données soient transmises. À l'aide des informations de l'étape précédente, vous allez créer la plupart des types de configuration suivants.

- [Config d'enregistrement Amazon S3](#)- Fournissez des données à votre compartiment Amazon S3.
- [Config de point de terminaison de flux de données](#)- Fournissez des données à votre instance Amazon EC2.

Configuration de télémétrie (facultatif)

Si vous souhaitez recevoir une télémétrie en temps quasi réel lors de vos contacts, vous pouvez créer un `TelemetrySinkConfig`. Cette configuration est facultative et indique où les données de télémétrie AWS Ground Station seront transmises.

- [Config du récepteur de télémétrie](#)- Transmettez des données de télémétrie à votre compte.

Pour des instructions de configuration détaillées, voir [Configuration de la télémétrie](#).

Configurations satellites

Les configurations du satellite indiquent AWS Ground Station comment communiquer avec votre satellite. Vous ferez référence aux informations que vous avez recueillies [satellite embarqué](#).

- [Suivi de Config](#)- Définit les préférences relatives au suivi physique de votre véhicule lors d'un contact. Cela est nécessaire pour la construction du profil de mission.
- [Config d'antenne de liaison descendante](#)- Fournissez des données de radiofréquence numérisées.
- [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#)- Fournissez des données de radiofréquence démodulées et décodées.
- [Config d'antenne de liaison montante](#)- Liez les données vers votre satellite.
- [Config d'écho d'antenne de liaison montante](#)- Diffusez un écho des données de votre signal de liaison montante.

Création d'un profil de mission

À l'aide des configurations créées à l'étape précédente, vous avez identifié comment suivre votre satellite, les moyens possibles de communiquer avec votre satellite et comment activer la télémétrie en temps quasi réel lors de l'exécution du contact. Au cours de cette étape, vous allez créer un ou plusieurs profils de mission. Un profil de mission représente l'agrégation des configurations possibles dans un comportement attendu qui peut ensuite être planifié et exploité.

Pour les derniers paramètres, veuillez vous référer au [type de AWS::GroundStation::MissionProfile CloudFormation ressource](#)

1. Donnez un nom à votre profil de mission. Cela vous permet de comprendre rapidement son utilisation au sein de votre système. Par exemple, vous pouvez avoir un satellite-wideband-narrowband-nominal-operations et un satellite-narrowband-emergency-operations si vous avez un opérateur à bande étroite distinct pour les opérations d'urgence.
2. Définissez votre configuration de suivi.
3. Définissez vos durées de contact minimales viables. Cela vous permet de filtrer les contacts potentiels pour répondre aux besoins de votre mission.
4. Configurez vos `streamsKmsKey` et `streamsKmsRole` ceux utilisés pour chiffrer vos données pendant le transport. Ceci est utilisé pour tous les flux de données de AWS Ground Station l'agent.
5. Définissez vos flux de données. Créez vos flux de données pour qu'ils correspondent aux signaux de votre opérateur à l'aide des configurations que vous avez créées à l'étape précédente.
6. [Facultatif] Définissez la durée en secondes de votre contact avant et après le passage. Ceci est utilisé pour émettre des événements par contact avant et après le contact, respectivement. Pour plus d'informations, consultez [Automatisez AWS Ground Station avec des événements](#).
7. [Facultatif] Configurez votre `telemetrySinkConfigArn` pour activer la télémétrie lors des contacts. Cela vous permet de recevoir des données télémétriques en temps quasi réel directement dans votre compte à des fins de surveillance et d'analyse. Pour plus d'informations, consultez [Travailler avec la télémétrie](#).
8. [Facultatif] Vous pouvez associer des tags à votre profil de mission. Ils peuvent être utilisés pour vous aider à différencier vos profils de mission de manière programmatique.

Vous pouvez faire référence au [Exemples de configurations de profil de mission](#), pour ne voir que quelques-unes des configurations potentielles.

Comprendre les prochaines étapes

Maintenant que vous avez un satellite embarqué et un profil de mission valide, vous êtes prêt à planifier des contacts et à communiquer avec votre satellite avec AWS Ground Station

Vous pouvez planifier un contact de l'une des manières suivantes :

- La [AWS Ground Station console](#).
- La AWS commande [reserve-contact](#) de la CLI.
- Le AWS SDK. [ReserveContact](#) API.

Pour plus d'informations sur la façon dont AWS Ground Station suit la trajectoire de votre satellite et sur la manière dont ces informations sont utilisées, veuillez vous référer à [Comprendre comment AWS Ground Station utilise les éphémérides](#).

AWS Ground Station gère un certain nombre de CloudFormation modèles préconfigurés pour faciliter la prise en main du service. Voir [Exemples de configurations de profil de mission](#) des exemples AWS Ground Station d'utilisation.

Le traitement des données numériques à fréquence intermédiaire ou des données démodulées et décodées qui vous sont fournies AWS Ground Station dépendra de votre cas d'utilisation spécifique. Les articles de blog suivants peuvent vous aider à comprendre certaines des options qui s'offrent à vous :

- [Observation de la Terre automatisée à l'aide de la livraison de données AWS Ground Station Amazon S3](#) (et de son GitHub référentiel associé [awslabs/ aws-groundstation-eos-pipeline](#))
- [Virtualisation du segment terrestre du satellite avec AWS](#)
- [Observation de la Terre à l'aide de AWS Ground Station : un guide pratique](#)
- [Création d'architectures de liaison descendante de données par satellite à haut débit avec DigiF et Amphinicy Blink SDR AWS Ground Station WideBand](#) (et son référentiel associé [aws-samples/ GitHub aws-groundstation-wbdigif-snpp](#))

AWS Ground Station Succursales

AWS Ground Station fournit un réseau mondial de stations au sol à proximité de notre réseau mondial de régions d'infrastructure AWS. Vous pouvez configurer votre utilisation de ces emplacements depuis n'importe quelle région AWS prise en charge. Cela inclut la région AWS dans laquelle les données sont fournies.



Trouver la AWS région pour l'emplacement d'une station au sol

Le réseau AWS Ground Station mondial inclut des stations au sol qui ne sont pas physiquement situées dans la [région AWS](#) à laquelle elles sont connectées. La liste des stations au sol auxquelles vous avez accès peut être récupérée via la [ListGroundStation](#) réponse du SDK AWS. La liste complète des emplacements des stations au sol est présentée ci-dessous, et d'autres seront bientôt disponibles. Reportez-vous au guide d'intégration pour ajouter ou modifier les approbations de site pour vos satellites.

Nom de la station au sol	Emplacement de la station Ground	Nom de la région AWS	Code de région AWS	Remarques
Alaska 1	Alaska, États-Unis	USA Ouest (Oregon)	us-west-2	Pas physiquement situé dans une AWS région
Bahreïn 1	Bahreïn	Moyen-Orient (Bahreïn)	me-south-1	
Le Cap 1	Cape Town, Afrique du Sud	Afrique (Le Cap)	af-south-1	
Dubbo 1	Dubbo, Australie	Asie-Pacifique (Sydney)	ap-southeast-2	Pas physiquement situé dans une AWS région
Hawaï 1	Hawaï, États-Unis	USA Ouest (Oregon)	us-west-2	Pas physiquement situé dans une AWS région
Irlande 1	Irlande	Europe (Irlande)	eu-west-1	
Ohio 1	Ohio, États-Unis	USA Est (Ohio)	us-east-2	
Oregon 1	Oregon, États-Unis	USA Ouest (Oregon)	us-west-2	
Punta Arenas 1	Punta Arenas, Chili	Amérique du Sud (São Paulo)	sa-east-1	Pas physiquement situé dans une AWS région
Séoul 1	Séoul, Corée du Sud	Asie-Pacifique (Séoul)	ap-northeast-2	
Singapour 1	Singapour	Asie-Pacifique (Singapour)	ap-southeast-1	

Nom de la station au sol	Emplacement de la station Ground	Nom de la région AWS	Code de région AWS	Remarques
Stockholm 1	Stockholm, Suède	Europe (Stockholm)	eu-north-1	

AWS Ground Station régions AWS prises en charge

Vous pouvez fournir des données et configurer vos contacts via le SDK AWS ou la AWS Ground Station console depuis les régions AWS prises en charge. Vous pouvez consulter les régions prises en charge et leurs points de terminaison associés dans les points de [AWS Ground Station terminaison et les quotas](#).

Disponibilité des jumeaux numériques

[Utilisez la fonction de jumelage AWS Ground Station numérique](#) est disponible dans toutes les [régions AWS](#) où AWS Ground Station il est disponible. Les stations terrestres à double numérique sont des copies exactes des stations au sol de production avec un préfixe modificateur du nom de la station au sol « Digital Twin ». Par exemple, « Digital Twin Ohio 1 » est une station terrestre double numérique qui est une copie exacte de la station au sol de production « Ohio 1 ».

Antennes dédiées

Outre les stations au sol accessibles au public répertoriées ci-dessus, AWS Ground Station propose des antennes dédiées. Une antenne dédiée est un système d'antenne conçu sur mesure qui AWS gère en votre nom. Une antenne dédiée n'est pas limitée aux emplacements des stations AWS Ground Station au sol existantes et peut être construite avec des capacités supérieures à celles des stations terrestres publiques, comme décrit dans [AWS Ground Station Fonctionnalités du site](#). Les emplacements et les capacités des antennes dédiées ne sont pas divulgués publiquement.

Pour plus d'informations sur les antennes dédiées, consultez [AWS Ground Station Antennes dédiées](#). Pour en savoir plus ou pour commencer à utiliser les antennes dédiées, contactez AWS Support via le [AWS Support Center Console](#).

Antennes de visualisation sur une station au sol

Chaque station au sol possède une ou plusieurs antennes. Vous pouvez visualiser les antennes d'une station au sol à l'aide de l'[ListAntennas](#) API. Cette API renvoie les antennes d'une station au sol spécifiée, y compris le nom de chaque antenne.

Les informations d'antenne sont utiles lorsqu'elles sont combinées à l'[ListGroundStationReservations](#) API pour comprendre la capacité et la disponibilité d'une station au sol. Pour plus d'informations sur la consultation des réservations, consultez [Afficher les réservations de stations au sol](#).

Pour appeler `ListAntennas`, vous devez avoir un satellite embarqué à bord de la station au sol ou disposer des autorisations relatives aux éphémérides d'altitude azimutale pour la station au sol. Pour de plus amples informations, veuillez consulter [Fournir des données sur les éphémérides d'altitude azimutale](#).

Exemple : répertorier les antennes d'une station au sol

L'exemple suivant répertorie toutes les antennes d'une station au sol utilisant le AWS SDK pour Python (Boto3).

```
import boto3

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

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

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

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

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

Afficher les réservations de stations au sol

Vous pouvez consulter les réservations via les antennes d'une station au sol à l'aide de l'[ListGroundStationReservations](#) API. Les réservations représentent des plages horaires sur les antennes, y compris vos contacts programmés. [AWS Ground Station Antennes dédiées](#) les clients voient également les fenêtres de maintenance.

Ces informations vous aident à comprendre la disponibilité des antennes lors de la planification des horaires de contact et fournissent une visibilité sur ce qui se passe sur les antennes d'une station au sol.

Réservations d'annonces

Pour répertorier les réservations, appelez [ListGroundStationReservations](#) avec un identifiant de station au sol et une plage horaire. L'API renvoie les réservations pour toutes les antennes de la station au sol dans le délai spécifié.

Les réservations que vous voyez dépendent de votre niveau d'accès :

- AWS Ground Station Clients publics — Vous ne pouvez voir que vos propres réservations de contacts. Les fenêtres de maintenance et les contacts appartenant à d'autres comptes ne sont pas inclus.
- AWS Ground Station Clients d'antennes dédiées — Vous pouvez consulter toutes les réservations sur vos antennes dédiées, y compris les fenêtres de maintenance et les contacts planifiés par d'autres comptes. Les identifiants de contact ne sont inclus que pour les contacts dont vous êtes le propriétaire. Pour de plus amples informations, veuillez consulter [AWS Ground Station Antennes dédiées](#).

Types de réservation

Chaque réservation possède un type qui indique à quoi sert l'heure de l'antenne :

- **Contact** — Une réservation de contact représente le temps d'antenne réservé aux communications par satellite. Les heures de début et de fin de réservation reflètent la réservation complète de l'antenne, y compris l'heure avant et après le passage, et pas seulement le créneau d'accès par satellite.
- **Maintenance** — Une réservation de maintenance représente une période pendant laquelle l'antenne n'est pas disponible pour cause de maintenance. Les réservations de maintenance incluent un `maintenanceType` qui indique si la maintenance était planifiée ou non planifiée.

Exemple de code

L'exemple suivant répertorie les réservations effectuées dans une station au sol pour les 7 prochains jours à l'aide du AWS SDK pour Python (Boto3), y compris le filtrage par type de réservation.

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

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

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

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

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

paginator = ground_station_client.get_paginator("list_ground_station_reservations")
page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    startTime=start_time,
```

```
        endTime=end_time,
        PaginationConfig={
            "MaxItems": 100,
            "PageSize": 20,
        },
    )

for page in page_iterator:
    for reservation in page["reservationList"]:
        reservation_type = reservation["reservationType"]
        antenna_name = reservation["antennaName"]
        res_start = reservation["startTime"]
        res_end = reservation["endTime"]

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

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

        print()

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

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

for page in page_iterator:
```

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

AWS Ground Station masques de site

Des masques de site sont associés à chaque [emplacement d' AWS Ground Station antenne](#). Ces masques empêchent les antennes situées à cet endroit d'émettre ou de recevoir lorsqu'elles pointent dans certaines directions, généralement près de l'horizon. Les masques peuvent prendre en compte :

- Caractéristiques du terrain géographique entourant l'antenne — Par exemple, cela inclut des éléments tels que des montagnes ou des bâtiments, qui bloqueraient un signal de radiofréquence (RF) ou empêcheraient la transmission.
- Interférences de fréquence radio (RFI) : cela affecte à la fois la capacité de réception (les sources RFI externes ont un impact sur un signal de liaison descendante dans les antennes de la station AWS) et de transmission (le signal RF transmis par les antennes de la station au sol AWS a un impact négatif sur les récepteurs externes).
- Autorisations légales — Les autorisations de site locales pour exploiter AWS Ground Station dans chaque région peuvent inclure des restrictions spécifiques, telles qu'un angle d'élévation minimal pour la transmission.

Ces masques de site peuvent changer au fil du temps. Par exemple, de nouveaux bâtiments peuvent être construits à proximité d'une antenne, les sources RFI peuvent changer ou l'autorisation légale peut être renouvelée avec différentes restrictions. Les masques du site AWS Ground Station sont mis à votre disposition dans le cadre d'un accord de confidentialité (NDA).

Masques spécifiques au client

Outre les masques de site AWS Ground Station présents sur chaque site, vous pouvez avoir des masques supplémentaires en raison des restrictions relatives à votre propre autorisation légale de communiquer avec vos satellites dans une région donnée. Ces masques peuvent être configurés dans AWS Ground Station de case-by-case manière à garantir la conformité lors de l'utilisation

d'AWS Ground Station pour communiquer avec ces satellites. Contactez l'équipe AWS Ground Station pour plus de détails.

Impact des masques de site sur les temps de contact disponibles

Il existe deux types de masques de site : les masques de site de liaison montante (transmission) et les masques de site de liaison descendante (réception).

Lorsque vous listez les heures de contact disponibles à l'aide de l' `ListContacts` opération, AWS Ground Station affiche les durées de visibilité en fonction du moment où votre satellite s'élèvera au-dessus du masque de liaison descendante et se placera en dessous de celui-ci. Les temps de contact disponibles sont basés sur cette fenêtre de visibilité du masque en lien descendant. Cela garantit que vous ne réservez pas de temps lorsque votre satellite se trouve sous le masque de liaison descendante.

Les masques de site Uplink ne sont pas appliqués aux temps de contact disponibles, même si le profil de mission inclut une [configuration de liaison montante d'antenne](#) dans un bord de flux de données. Cela vous permet d'utiliser tout le temps de contact disponible pour la liaison descendante, même si la liaison montante peut ne pas être disponible pendant une partie de cette période en raison du masque du site de liaison montante. Cependant, le signal de liaison montante peut ne pas être transmis pendant une partie ou la totalité du temps réservé à un contact satellite. Vous êtes responsable de la prise en compte du masque de liaison montante fourni lors de la planification des transmissions en liaison montante.


La partie d'un contact qui n'est pas disponible pour la liaison montante varie en fonction de la trajectoire du satellite pendant le contact, par rapport au masque du site de liaison montante à l'emplacement de l'antenne. Dans les régions où les masques de site de liaison montante et de liaison descendante sont similaires, cette durée est généralement courte. Dans d'autres régions, où le masque de liaison montante peut être considérablement plus élevé que le masque de site de liaison descendante, cela peut entraîner l'indisponibilité d'une partie importante, voire de la totalité, de la durée du contact pour la liaison montante. Le temps de contact complet vous est facturé, même si une partie du temps réservé n'est pas disponible pour la liaison montante.

AWS Ground Station Fonctionnalités du site

Pour simplifier votre expérience, AWS Ground Station détermine un ensemble commun de capacités pour un type d'antenne, puis déploie plusieurs antennes sur un emplacement de station au sol. Une partie des étapes d'intégration permet de s'assurer que votre satellite est compatible avec les types d'antennes situés à un endroit précis. Lorsque vous réservez un contact, vous déterminez

indirectement le type d'antenne utilisé. Cela garantit que votre expérience à un emplacement donné de station au sol reste la même au fil du temps, quelles que soient les antennes utilisées. Les performances spécifiques de votre contact peuvent varier en raison d'une grande variété de préoccupations environnementales, telles que les conditions météorologiques sur le site.

À l'heure actuelle, tous les sites prennent en charge les fonctionnalités suivantes :

 Note

Chaque ligne du tableau suivant indique un chemin de communication indépendant, sauf indication contraire. Des lignes dupliquées existent pour refléter nos capacités multicanaux qui permettent d'utiliser simultanément plusieurs voies de communication.

Type de capacité	Gamme de fréquences	Plage de bande passante	Polarization	Nom commun	Remarques
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	RHCP	Liaison descendante à large bande X	Cette fonctionnalité nécessite l'utilisation de l' AWS Ground Station agent .
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	RHCP		Cette fonctionnalité n'est pas prise en charge dans Alaska 1 ou Punta Arenas 1.
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	RHCP		La bande passante globale ne

Type de capacité	Gamme de fréquences	Plage de bande passante	Polarization	Nom commun	Remarques
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	RHCP		doit pas dépasser 400 MHz par polarisation à chaque emplacement.
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	LHCP		Toutes les plages de fréquences utilisées ne doivent pas se chevaucher.
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	LHCP		
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	LHCP		
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	LHCP		
liaison descendante entre antennes	7750 - 8500 MHz	50 à 400 MHz	LHCP		

Type de capacité	Gamme de fréquences	Plage de bande passante	Polarization	Nom commun	Remarques
liaison descendante entre antennes	2200 - 2290 MHz	Jusqu'à 40 MHz	RHCP	Liaison descendante en bande S	Une seule polarisation peut être utilisée à la fois
liaison descendante entre antennes	2200 - 2290 MHz	Jusqu'à 40 MHz	LHCP		
liaison descendante entre antennes	7750 - 8500 MHz	Jusqu'à 40 MHz	RHCP	Liaison descendante à bande étroite en bande X	Une seule polarisation peut être utilisée à la fois
liaison descendante entre antennes	7750 - 8500 MHz	Jusqu'à 40 MHz	LHCP		
antenne-uplink	2025 - 2110 MHz	Jusqu'à 40 MHz	RHCP	Liaison montante en bande S	Une seule polarisation peut être utilisée à la fois
antenne-uplink	2025 - 2110 MHz	Jusqu'à 40 MHz	LHCP		

EIRP 20-50 dBW

Type de capacité	Gamme de fréquences	Plage de bande passante	Polarization	Nom commun	Remarques
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	RHCP	Écho Uplink	Correspond aux restrictions relatives aux liaisons montantes entre antennes
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	LHCP		
antenna-downlink-demod-decode	7750 - 8500 MHz	Jusqu'à 500 MHz	RHCP	Liaison descendante démodulée et décodée en bande X	
antenna-downlink-demod-decode	7750 - 8500 MHz	Jusqu'à 500 MHz	LHCP		
suivi	N/A	N/A	N/A	N/A	Support pour le suivi automatique et le suivi des programmes

* RHCP = polarisation circulaire droitère et LHCP = polarisation circulaire gaucheère. Pour plus d'informations sur la polarisation, voir [Polarisation circulaire](#).

Comprendre comment AWS Ground Station utilise les éphémérides

Une [éphéméride](#), éphémérides au pluriel, est un fichier ou une structure de données fournissant la trajectoire d'objets astronomiques. Historiquement, ce fichier ne faisait référence qu'à des données tabulaires, mais progressivement, il a été dirigé vers une grande variété de fichiers de données indiquant la trajectoire d'un engin spatial.

L'API Ephemeris permet de télécharger des éphémérides personnalisées pour une utilisation avec un satellite AWS Ground Station . [Ces éphémérides remplacent les éphémérides par défaut de Space-Track \(voir :\). Données d'éphémérides par défaut](#) Nous prenons en charge la réception de données éphémérides aux formats Orbit Ephemeris Message (OEM), élément à deux lignes (TLE) et élévation azimutale.

AWS Ground Station utilise les données d'éphémérides pour déterminer à quel moment les contacts deviennent disponibles en fonction des éphémérides fournies et pour commander correctement les antennes du réseau. AWS Ground Station [Par défaut, aucune action n'est requise pour fournir AWS Ground Station des éphémérides si un identifiant NORAD a été attribué à votre satellite.](#)

Le téléchargement d'éphémérides personnalisées peut améliorer la qualité du suivi, gérer les premières opérations lorsqu'aucune éphéméride [Space-Track](#) n'est disponible et prendre en compte les manœuvres. AWS Ground Station

Il prend également AWS Ground Station en charge un format d'élévation azimutale, qui vous permet de spécifier directement les directions de pointage des antennes sans fournir d'informations sur l'orbite du satellite. Cela est utile pour les scénarios où un pointage précis de l'antenne est requis car les informations sur la trajectoire du satellite sont imprécises ou inconnues.

Rubriques

- [Données d'éphémérides par défaut](#)
- [Fournir des données d'éphémérides personnalisées](#)
- [Réservez des contacts avec des éphémérides personnalisées](#)
- [Comprendre quelle éphéméride est utilisée](#)
- [Obtenir l'éphéméride actuelle d'un satellite](#)
- [Revenir aux données d'éphémérides par défaut](#)

Données d'éphémérides par défaut

Par défaut, AWS Ground Station utilise les données accessibles au public provenant de [Space-Track](#), et aucune action n'est requise pour fournir ces AWS Ground Station éphémérides par défaut. [Ces éphémérides sont des ensembles d'éléments à deux lignes \(TLEs\) associés à l'identifiant NORAD de votre satellite](#). Toutes les éphémérides par défaut ont une priorité de. 0 Par conséquent, elles seront toujours remplacées par toutes les éphémérides personnalisées non expirées téléchargées via l'API des éphémérides, qui doivent toujours avoir une priorité égale ou supérieure. 1

Les satellites sans identifiant NORAD doivent télécharger des données d'éphémérides personnalisées sur. AWS Ground Station Par exemple, les satellites qui viennent d'être lancés ou qui sont volontairement omis du catalogue [Space-Track](#) n'auraient aucun identifiant NORAD et auraient besoin d'éphémérides personnalisées téléchargées. Pour plus d'informations sur la fourniture de données d'éphémérides personnalisées, voir : [Fournir des données d'éphémérides personnalisées](#).

Fournir des données d'éphémérides personnalisées

Important

L'API ephemeris est actuellement en version préliminaire

L'accès à l'API Ephemeris n'est fourni qu'en cas de besoin. Si vous souhaitez pouvoir télécharger des données d'éphémérides personnalisées, veuillez ouvrir un AWS Support ticket via le. [AWS Support Center Console](#) Notre équipe travaillera avec vous pour activer cette fonctionnalité en fonction de vos besoins spécifiques.

Présentation de

L'API Ephemeris permet de télécharger des éphémérides personnalisées pour une utilisation avec un satellite AWS Ground Station . [Ces éphémérides remplacent les éphémérides par défaut de Space-Track \(voir :\). \[Données d'éphémérides par défaut\]\(#\)](#) Nous prenons en charge la réception de données éphémérides aux formats Orbit Ephemeris Message (OEM), élément à deux lignes (TLE) et élévation azimutale.

AWS Ground Station traite les éphémérides comme des données d'utilisation [individualisées](#). Si vous utilisez cette fonctionnalité facultative, AWS utilisera vos données d'éphéméride pour fournir une assistance en matière de dépannage.

Le téléchargement d'éphémérides personnalisées peut améliorer la qualité du suivi, gérer les opérations pour lesquelles aucune éphéméride [Space-Track](#) n'est disponible et prendre en compte les manœuvres. AWS Ground Station

Pour résoudre le problème d'une éphéméride non valide, consultez : [Résoudre les problèmes liés aux éphémérides non valides](#)

Exemple : utilisation d'éphémérides fournies par le client avec AWS Ground Station

[Pour des instructions plus détaillées sur l'utilisation des éphémérides fournies par le client avec AWS Ground Station, voir Utilisation des éphémérides fournies par le client avec et son référentiel associé aws-samples/. AWS Ground Station GitHub aws-groundstation-cpe](#)

Fournir des données d'éphémérides TLE

Important

L'API ephemeris est actuellement en version préliminaire

L'accès à l'API Ephemeris n'est fourni qu'en cas de besoin. Si vous souhaitez pouvoir télécharger des données d'éphémérides personnalisées, veuillez ouvrir un AWS Support ticket via le [AWS Support Center Console](#) Notre équipe travaillera avec vous pour activer cette fonctionnalité en fonction de vos besoins spécifiques.

Présentation de

Les ensembles d'éléments à deux lignes (TLE) constituent un format normalisé pour décrire les orbites des satellites. L'API Ephemeris permet de télécharger des éphémérides TLE pour les utiliser avec un satellite AWS Ground Station . [Ces éphémérides remplacent les éphémérides par défaut de Space-Track \(voir :\). Données d'éphémérides par défaut](#)

AWS Ground Station traite les éphémérides comme des données d'utilisation [individualisées](#). Si vous utilisez cette fonctionnalité facultative, AWS utilisera vos données d'éphéméride pour fournir une assistance en matière de dépannage.

Le téléchargement d'éphémérides TLE personnalisées peut améliorer la qualité du suivi, gérer les premières opérations lorsqu'aucune éphéméride [Space-Track](#) n'est disponible et prendre en compte les manœuvres. AWS Ground Station

Note

Lorsque vous fournissez des éphémérides personnalisées avant qu'un numéro de catalogue satellite ne soit attribué à votre satellite, vous pouvez l'utiliser 00000 pour le champ du numéro de catalogue satellite du TLE et 000 pour la partie du numéro de lancement du champ de désignation international du TLE (par exemple 24000A pour un véhicule lancé en 2024).

Pour plus d'informations sur le format de TLEs, consultez la section [Ensemble d'éléments sur deux lignes](#).

Création d'une éphéméride TLE

Une éphéméride TLE peut être créée à l'aide de l'[CreateEphemeris](#) action de l' AWS Ground Station API. Cette action téléchargera une éphéméride en utilisant les données contenues dans le corps de la demande ou provenant d'un compartiment S3 spécifié.

Il est important de noter que le téléchargement d'une éphéméride définit l'éphéméride et lance un flux de travail asynchrone qui validera VALIDATING et générera des contacts potentiels à partir de votre éphéméride. Ce n'est qu'une fois qu'une éphéméride aura passé ce flux de travail et ENABLED sera devenue qu'elle sera utilisée pour les contacts. Vous devez effectuer un sondage [DescribeEphemeris](#) pour connaître le statut des éphémérides ou utiliser des CloudWatch événements pour suivre les changements de statut des éphémérides.

Pour résoudre le problème d'une éphéméride non valide, consultez : [Résoudre les problèmes liés aux éphémérides non valides](#)

Exemple : créer un ensemble d'éphémérides d'un élément à deux lignes (TLE) via une API

La AWS SDKs CLI et peut être utilisée pour télécharger un ensemble d'éphémérides d'un élément à deux lignes (TLE) AWS Ground Station via l'[CreateEphemeris](#) appel. Cette éphéméride sera utilisée à la place des données d'éphéméride par défaut pour un satellite (voir). [Données d'éphémérides par défaut](#) Cet exemple montre comment procéder à l'aide du [AWS SDK pour Python \(Boto3\)](#).

Un ensemble TLE est un objet au format JSON qui enchaîne un ou plusieurs TLEs objets pour construire une trajectoire continue. L' TLEs ensemble TLE doit former un ensemble continu que nous pouvons utiliser pour construire une trajectoire (c'est-à-dire aucun intervalle de temps entre TLEs les deux ensembles TLE). Un exemple d'ensemble TLE est illustré ci-dessous :

```
[
  {
    "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

Les plages de temps TLEs d'un ensemble TLE doivent correspondre exactement pour obtenir une trajectoire continue valide.

Un ensemble TLE peut être téléchargé via le client AWS Ground Station boto3 comme suit :

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

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

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

```

    expirationTime=datetime.now(timezone.utc) + timedelta(days=3),
    priority=2,
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   20318.54719794   .000000075   00000-0
26688-4 0  9997",
                    "tleLine2": "2 25994   98.2007   30.6589 0001234   89.2782   18.9934
14.57114995111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
            ]
        }
    },
)

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

```

Cet appel renverra un EphemerisID qui pourra être utilisé pour référencer l'éphéméride à l'avenir. Par exemple, nous pouvons utiliser l'EphemerisID fourni lors de l'appel ci-dessus pour demander le statut de l'éphéméride :

```

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

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

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

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

```

```

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

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

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

response = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)

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

```

Un exemple de réponse à l'[DescribeEphemeris](#) action est fourni ci-dessous

```

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

Il est recommandé d'interroger l'[DescribeEphemeris](#) itinéraire ou d'utiliser CloudWatch des événements pour suivre l'état des éphémérides téléchargées, car elles doivent passer par un flux de travail de validation asynchrone avant d'être définies et de devenir utilisables pour la planification ENABLED et l'exécution de contacts.

Notez que l'identifiant NORAD dans l'ensemble TLE, TLEs dans les exemples ci-dessus, doit correspondre 25994 à l'identifiant NORAD attribué à votre satellite dans la base de données [Space-Track](#).

Exemple : téléchargement de données d'éphémérides TLE depuis un compartiment S3

Il est également possible de télécharger un fichier d'éphémérides TLE directement depuis un compartiment S3 en pointant sur le compartiment et la clé d'objet. AWS Ground Station récupérera l'objet en votre nom. Les informations sur le cryptage des données au repos AWS Ground Station sont détaillées dans : [Chiffrement des données au repos pour AWS Ground Station](#).

Vous trouverez ci-dessous un exemple de téléchargement d'un fichier d'éphémérides TLE à partir d'un compartiment S3.

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

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

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

# Create sample TLE set data
# Note: For actual satellites, use real TLE data from sources like Space-Track
tle_set_data = [
    {
```

```

        "tleLine1": "1 25994U 99068A 20318.54719794 .00000075 00000-0 26688-4 0
9997",
        "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934
14.57114995111906",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc),
            "endTime": datetime.now(timezone.utc) + timedelta(days=3),
        },
    },
    {
        "tleLine1": "1 25994U 99068A 20321.54719794 .00000075 00000-0 26688-4 0
9998",
        "tleLine2": "2 25994 98.2007 33.6589 0001234 89.2782 18.9934
14.57114995112342",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc) + timedelta(days=3),
            "endTime": datetime.now(timezone.utc) + timedelta(days=7),
        },
    },
]

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

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

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

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

s3_tle_ephemeris = ground_station_client.create_ephemeris(
    name="2022-11-05 S3 TLE Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,
    ephemeris={"tle": {"s3object": {"bucket": bucket_name, "key": object_key}}},
)

```

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

Fournir des données d'éphémérides OEM

Important

L'API ephemeris est actuellement en version préliminaire

L'accès à l'API Ephemeris n'est fourni qu'en cas de besoin. Si vous souhaitez pouvoir télécharger des données d'éphémérides personnalisées, veuillez ouvrir un AWS Support ticket via le [AWS Support Center Console](#). Notre équipe travaillera avec vous pour activer cette fonctionnalité en fonction de vos besoins spécifiques.

Présentation de

Le message OEM (Orbit Ephemeris Message) est un format normalisé pour représenter les données de trajectoire des engins spatiaux. L'API Ephemeris permet de télécharger des éphémérides OEM pour les utiliser avec un satellite AWS Ground Station. [Ces éphémérides remplacent les éphémérides par défaut de Space-Track \(voir :\). Données d'éphémérides par défaut](#)

AWS Ground Station traite les éphémérides comme des données d'utilisation [individualisées](#). Si vous utilisez cette fonctionnalité facultative, AWS nous utiliserons les données de vos éphémérides pour fournir une assistance en matière de dépannage.

Le téléchargement d'éphémérides OEM personnalisées peut améliorer la qualité du suivi, gérer les premières opérations lorsqu'aucune éphéméride [Space-Track](#) n'est disponible et prendre en compte les manœuvres. AWS Ground Station

Note

Lorsque vous fournissez des éphémérides personnalisées avant qu'un numéro de catalogue satellite ne soit attribué à votre satellite, vous pouvez l'utiliser `satelliteId` pour la `OBJECT_ID` partie de l'OEM.

Pour plus d'informations sur le format de OEMs, consultez [Format d'éphéméride OEM](#).

Format d'éphéméride OEM

AWS Ground Station traite les éphémérides fournies par le client OEM conformément à la [norme CCSDS](#) avec quelques restrictions supplémentaires. Les fichiers OEM doivent être au format KVN. Le tableau suivant décrit les différents champs d'un OEM et explique en quoi il AWS Ground Station diffère de la norme CCSDS.

Section	Champ	CCSDS requis	AWS Ground Station requis	Remarques
En-tête	CCSDS_OEM _VERS	Oui	Oui	Valeur requise : 2,0
	COMMENT	Non	Non	
	CLASSIFIC ATION	Non	Non	
	DATE DE CRÉATION	Oui	Oui	
	AUTEUR	Oui	Oui	
	IDENTIFIA NT_MESSAGE	Non	Non	
Métadonnées	META_START	Oui	Oui	
	COMMENT	Non	Non	
	NOM_OBJET	Oui	Oui	
	IDENTIFIA NT_OBJET	Oui	Oui	
	NOM_CENTRE	Oui	Oui	Valeur requise : Terre
	REF_FRAME	Oui	Oui	Valeurs acceptées :

Section	Champ	CCSDS requis	AWS Ground Station requis	Remarques
				EME2000, ITRF2000
	REF_FRAME_EPOCH	Non	Non pris en charge*	Pas nécessaire car les REF_FRAMEs ont une époque implicite
	TIME_SYSTEM	Oui	Oui	Valeur requise : UTC
	HEURE DE DÉBUT	Oui	Oui	
	HEURE_DE_DÉMARRAGE_UTILISABLE	Non	Non	
	TIME_STOP_UTILE	Non	Non	
	STOP_TIME	Oui	Oui	
	INTERPOLATION	Non	Oui	Nécessaire pour AWS Ground Station générer des angles de pointage précis pour les contacts.

Section	Champ	CCSDS requis	AWS Ground Station requis	Remarques
	DEGRÉ_D'INTERPOLATION	Non	Oui	Nécessaire pour AWS Ground Station générer des angles de pointage précis pour les contacts. Le degré spécifié sera utilisé si possible, mais un degré inférieur sera utilisé s'il n'y a pas suffisamment de données dans le segment.
	META_STOP	Oui	Oui	
Données	X	Oui	Oui	Représenté dans km
	Y	Oui	Oui	Représenté dans km
	Z	Oui	Oui	Représenté dans km
	X_POINT	Oui	Oui	Représenté dans km/s
	Y_POINT	Oui	Oui	Représenté dans km/s

Section	Champ	CCSDS requis	AWS Ground Station requis	Remarques
	Z_POINT	Oui	Oui	Représenté dans km/s
	X_DOT	Non	Non	Représenté dans km/s ²
	Y_DDOT	Non	Non	Représenté dans km/s ²
	Z_DDOT	Non	Non	Représenté dans km/s ²
Matrice de covariance	COVARIANC E_START	Non	Non	
	EPOCH	Non	Non	
	COV_REF_F RAME	Non	Non	
	COVARIANC E_STOP	Non	Non	

* Si des lignes non prises en charge par AWS Ground Station sont incluses dans l'OEM fourni, celui-ci échouera à la validation.

Les écarts importants par rapport à la norme CCSDS sont les AWS Ground Station suivants :

- CCSDS_OEM_VER est tenu de l'être 2.0.
- REF_FRAME doit être l'un EME2000 ou l'autre ITRF2000.
- REF_FRAME_EPOCH n'est pas pris en charge par AWS Ground Station.
- CENTER_NAME est tenu de l'être Earth.
- TIME_SYSTEM est tenu de l'être UTC.
- INTERPOLATION et INTERPOLATION_DEGREE sont tous deux requis pour les AWS Ground Station éphémérides fournies par le client.

- AWS Ground Station s'écarte de CCSDS 5.2.4.7 en autorisant les blocs de données OEM ne contenant pas suffisamment d'enregistrements de données d'éphémérides à effectuer une interpolation à la valeur spécifiée. INTERPOLATION_DEGREE Dans ce cas, AWS Ground Station utilisera le degré d'interpolation le plus élevé possible inférieur ou égal à celui spécifié INTERPOLATION_DEGREE.

Exemple d'éphéméride OEM au format KVN

Voici un exemple tronqué d'éphéméride OEM au format KVN pour le satellite de diffusion public JPSS-1.

```

CCSDS_OEM_VERS = 2.0

COMMENT Orbit data are consistent with planetary ephemeris DE-430

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

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

2024-07-22T00:00:00.000000  5.905147360000000e+02  -1.860082793999999e+03
-6.944807075000000e+03  -5.784245796000000e+00  4.347501391999999e+00
-1.657256863000000e+00

2024-07-22T00:01:00.000000  2.425572045154201e+02  -1.595860765983339e+03
-7.030938457373539e+03  -5.810660250794190e+00  4.457103652219009e+00
-1.212889340333023e+00

2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
-7.090262617183503e+03  -5.814973972202444e+00  4.549739160042560e+00
-7.639633689161465e-01

2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
-7.122556683227951e+03  -5.797176562437553e+00  4.625064829516728e+00
-3.121687831090774e-01

```

```
2024-07-22T00:04:00.000000 -8.015427368657785e+02 -7.709137891269565e+02
-7.127699477194810e+03 -5.757338007808417e+00 4.682800822515077e+00
1.407953645161997e-01
2024-07-22T00:05:00.000000 -1.145240083085062e+03 -4.886583601179489e+02
-7.105671911254255e+03 -5.695608435738609e+00 4.722731329786999e+00
5.932259682105052e-01
2024-07-22T00:06:00.000000 -1.484582479061495e+03 -2.045451985605701e+02
-7.056557069672793e+03 -5.612218005854990e+00 4.744705579872771e+00
1.043421397392599e+00
```

Création d'une éphéméride OEM

Une éphéméride OEM peut être créée à l'aide de l'[CreateEphemeris](#) action de l' AWS Ground Station API. Cette action téléchargera une éphéméride en utilisant les données contenues dans le corps de la demande ou provenant d'un compartiment S3 spécifié.

Il est important de noter que le téléchargement d'une éphéméride définit l'éphéméride et lance un flux de travail asynchrone qui validera `VALIDATING` et générera des contacts potentiels à partir de votre éphéméride. Ce n'est qu'une fois qu'une éphéméride aura passé ce flux de travail et `ENABLED` sera devenue qu'elle sera utilisée pour les contacts. Vous devez effectuer un sondage [DescribeEphemeris](#) pour connaître le statut des éphémérides ou utiliser des CloudWatch événements pour suivre les changements de statut des éphémérides.

Pour résoudre le problème d'une éphéméride non valide, consultez : [Résoudre les problèmes liés aux éphémérides non valides](#)

Exemple : téléchargement de données d'éphémérides OEM depuis un compartiment S3

Il est également possible de télécharger un fichier d'éphéméride OEM directement depuis un compartiment S3 en pointant sur le compartiment et la clé d'objet. AWS Ground Station récupérera l'objet en votre nom. Les informations sur le cryptage des données au repos AWS Ground Station sont détaillées dans : [Chiffrement des données au repos pour AWS Ground Station](#).

Vous trouverez ci-dessous un exemple de téléchargement d'un fichier d'éphémérides OEM à partir d'un compartiment S3

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

# Create AWS clients
```

```

s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

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

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

COMMENT Orbit data are consistent with planetary ephemeris DE-430

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

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

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

```

```

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

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

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

print("OEM data uploaded successfully to S3")

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

s3_oem_ephemeris = ground_station_client.create_ephemeris(
    name="2024-07-22 S3 OEM Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,
    ephemeris={"oem": {"s3object": {"bucket": bucket_name, "key": object_key}}},
)

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

```

Vous trouverez ci-dessous un exemple de données renvoyées par l'[DescribeEphemeris](#) action appelée pour les éphémérides OEM téléchargées dans le bloc d'exemple de code précédent.

```

{
  "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"
}
}
```

Fournir des données sur les éphémérides d'altitude azimutale

Important

La fonction d'éphéméride d'altitude azimutale est actuellement en état d'aperçu et nécessite une intégration explicite.

La fonctionnalité des éphémérides d'élévation d'azimut est soumise à un contrôle d'accès strict pour un nombre limité de cas d'utilisation spécialisés prédéterminés. L'accès est nettement plus restrictif que pour les fonctionnalités d'éphémérides standard fournies par le client. Pour plus d'informations sur les cas d'utilisation approuvés et le processus de demande d'accès, veuillez ouvrir un AWS Support ticket via le [AWS Support Center Console](#). Notre équipe vous guidera tout au long du processus d'approbation pour les cas d'utilisation spécialisés.

Présentation de

Les éphémérides d'élévation d'azimut permettent de spécifier directement les directions de pointage des antennes sans fournir d'informations orbitales pour les satellites. Au lieu de télécharger des données éphémérides décrivant l'orbite d'un satellite, vous fournissez des angles d'azimut et d'élévation horodatés qui indiquent à l'antenne exactement où pointer tout au long d'un contact.

AWS Ground Station traite les éphémérides comme des données d'utilisation [individualisées](#). Si vous utilisez cette fonctionnalité facultative, AWS utilisera vos données d'éphéméride pour fournir une assistance en matière de dépannage.

Cette approche est particulièrement utile pour les scénarios suivants :

- Assistance aux premières opérations : pendant la phase de lancement et d'orbite précoce (LEOP) lorsque des données orbitales précises ne sont pas disponibles ou que les paramètres orbitaux changent rapidement.

- Modèles de pointage personnalisés : mise en œuvre de séquences de pointage spécifiques pour les tests d'antennes ou les opérations non standard.

Note

Lorsque vous utilisez des éphémérides d'altitude azimutale, l'ARN du satellite peut être omis de la demande de réservation de contact. Si l'ARN du satellite n'est pas omis, il sera toujours inclus dans les données de contact, mais les éphémérides d'élévation d'azimut seront utilisées pour le pointage de l'antenne plutôt que pour effectuer une résolution prioritaire des éphémérides. L'éphéméride d'élévation azimutale est associée à une station au sol spécifique et définit les directions de pointage de l'antenne pour cet emplacement.

Format de données des éphémérides d'altitude azimutale

Les données d'éphémérides d'altitude azimutale se composent de valeurs d'azimut et d'altitude étiquetées dans le temps et organisées en segments. Chaque segment contient une série d'angles d'azimut et d'élévation qui couvrent une plage de temps spécifique.

Les principaux éléments des données d'éphémérides d'altitude azimutale sont les suivants :

- Station au sol : station au sol spécifique où cette éphéméride d'altitude azimutale sera utilisée.
- Unité d'angle : unité de mesure des angles (DEGREE_ANGLE ou RADIAN).
- Segments : une ou plusieurs collections limitées dans le temps d'angles d'azimut et d'élévation.
- Angles horodatés : valeurs d'azimut et d'altitude individuelles avec horodatage associé.

Chaque segment nécessite :

- Une époque de référence (heure de base pour le segment)
- Une plage horaire valide (heures de début et de fin du segment)
- Au moins 5 paires horodatées azimuth/elevation

Contraintes d'élévation de l'azimut :

- Azimut en degrés : -180° à 360°
- Azimut en radians : $-\pi$ à 2π

- Altitude en degrés : -90° à 90°
- Altitude en radians : $-\pi/2$ à $\pi/2$
- Les valeurs temporelles doivent être classées par ordre croissant dans chaque segment
- Les segments ne doivent pas se chevaucher dans le temps

Pour plus d'informations, consultez la documentation de l'[CreateEphemeris](#) API et le type de [TimeAzEl](#) données.

Création d'éphémérides d'élévation d'azimut

Les éphémérides d'élévation azimutale sont créées à l'aide de la même action d'[CreateEphemeris](#) API, mais avec le type d'éphéméride. `azEl` Les principales différences par rapport aux éphémérides TLE et OEM sont les suivantes :

- Vous devez spécifier un `groundStation` paramètre
- Le `satelliteId` paramètre doit être omis de la demande
- Les paramètres de priorité ne s'appliquent pas (chaque éphéméride d'azimut est spécifique à une station au sol)
- Chaque segment doit contenir au moins 5 azimuth/elevation points pour permettre une interpolation de Lagrange de 4e ordre
- Les limites et exigences supplémentaires sont détaillées dans la documentation de [CreateEphemeris](#) l'API

Il est important de noter que le téléchargement d'une éphéméride définit l'éphéméride et lance un flux de travail asynchrone qui validera `VALIDATING` et générera des contacts potentiels à partir de votre éphéméride. Une éphéméride ne sera utilisée pour les contacts qu'une fois qu'elle aura passé ce flux de travail et que son statut sera passé. `ENABLED` Vous devez effectuer un sondage [DescribeEphemeris](#) pour connaître le statut des éphémérides ou utiliser des CloudWatch événements pour suivre les changements de statut des éphémérides.

Pour résoudre le problème d'une éphéméride non valide, consultez : [Résoudre les problèmes liés aux éphémérides non valides](#)

Exemple : créer des éphémérides d'altitude azimutale via l'API

L'exemple suivant montre comment créer une éphéméride d'altitude azimutale à l'aide du AWS SDK pour Python (Boto3) :

```

import boto3

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

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

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

```

Dans cet exemple :

- Les données d'altitude azimutale sont associées à la station au sol « Ohio 1 »

- Les angles sont spécifiés en degrés
- Le segment couvre une période de 15 minutes
- Les dt valeurs sont décalées en secondes atomiques par rapport à l'époque de référence
- Six azimuth/elevation paires sont fournies (le minimum est de 5)

Exemple : télécharger des données d'altitude azimutale depuis S3

Pour les ensembles de données plus volumineux, vous pouvez télécharger les données d'altitude azimutale à partir d'un compartiment S3 :

```
import boto3
import json

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

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

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

```

        "referenceEpoch": "2024-03-15T10:15:00Z",
        "validTimeRange": {
            "startTime": "2024-03-15T10:15:00Z",
            "endTime": "2024-03-15T10:30:00Z",
        },
        "azElList": [
            {"dt": 0.0, "az": 70.0, "el": 35.0},
            {"dt": 180.0, "az": 75.0, "el": 40.0},
            {"dt": 360.0, "az": 80.0, "el": 45.0},
            {"dt": 540.0, "az": 85.0, "el": 50.0},
            {"dt": 720.0, "az": 90.0, "el": 55.0},
            {"dt": 900.0, "az": 95.0, "el": 50.0},
        ],
    },
],
}

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

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

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

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

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

```

L'objet S3 doit contenir une structure JSON avec les données d'altitude azimutale dans le même format que celui indiqué dans l'exemple de téléchargement direct.

Réservation de contacts avec des éphémérides d'élévation d'azimut

Lorsque vous utilisez une éphéméride d'élévation d'azimut pour réserver un contact, le processus est différent de celui des éphémérides TLE et OEM :

1. Créez l'éphéméride d'élévation azimutale à l'aide de [CreateEphemeris](#)
2. Attendez que l'éphéméride atteigne le statut ENABLED
3. Réservez le contact en utilisant [ReserveContact](#) suivi des dérogations

Exemple de réservation d'un contact avec des éphémérides d'élévation d'azimut :

```
import boto3
from datetime import datetime
import time

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

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

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



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

Note

Le `satelliteArn` paramètre peut être omis lors de la réservation d'un contact avec des éphémérides d'élévation d'azimut. L'antenne suivra les angles d'azimut et d'élévation spécifiés pendant le contact.

Liste des contacts disponibles

Lorsque vous utilisez des éphémérides d'élévation d'azimut, l'API nécessite des [ListContacts](#) paramètres spécifiques :

- Le `satelliteArn` paramètre peut être omis de la demande
- Vous devez fournir un `ephemeris` paramètre avec l'identifiant de l'éphéméride à élévation azimutale pour spécifier les éphémérides à utiliser
- Les fenêtres de contact disponibles indiquent lorsque les angles d'azimut et d'élévation fournis sont au-dessus du [masque de site](#) de la station au sol demandée
- Vous devez toujours fournir `groundStation` et `missionProfileArn`

Exemple de création d'une éphéméride d'altitude azimutale et de liste des contacts disponibles à l'aide de cette éphéméride :

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

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

# Step 1: Create azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Stockholm AzEl Ephemeris",
    ephemeris={
        "azEl": {
            "groundStation": "Stockholm 1",
            "data": {
```

```

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

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

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

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

```

```

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

# Step 3: List available contacts using the azimuth elevation ephemeris
print("Listing available contacts with azimuth elevation ephemeris...")

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

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

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

```

Note

Le ephemeris paramètre avec l'ID d'altitude d'azimut doit être fourni lors de la liste des contacts afin de spécifier les éphémérides d'élévation d'azimut à utiliser pour déterminer les fenêtres de contact. S'il satelliteArn est inclus, il sera associé aux données de contact,

mais les éphémérides d'élévation azimutale seront utilisées pour le pointage de l'antenne plutôt que pour effectuer une résolution prioritaire des éphémérides.

Réservez des contacts avec des éphémérides personnalisées

Présentation de

Lorsque vous utilisez des éphémérides personnalisées (TLE, OEM ou élévation d'azimut), vous pouvez réserver des contacts à l'aide de l'API. [ReserveContact](#) Cette section décrit deux flux de travail courants pour la réservation de contacts et les points importants à prendre en compte pour garantir une planification des contacts réussie.

AWS Ground Station les antennes sont des ressources partagées entre plusieurs clients. Cela signifie que même si une fenêtre de contact apparaît disponible lorsque vous listez des contacts, un autre client peut la réserver avant vous. Il est donc essentiel de vérifier que votre contact arrive dans l'`SCHEDULED` état après la réservation et de mettre en place un suivi approprié des changements d'état du contact.

Important

Pour les éphémérides d'altitude azimutale, le `satelliteArn` paramètre peut être omis dans la `ReserveContact` demande, et vous devez fournir l'identifiant de l'éphéméride. `trackingOverrides` Pour les éphémérides TLE et OEM, vous devez toujours fournir le `satelliteArn`

Flux de travail de réservation de contacts

Il existe deux flux de travail principaux pour réserver des contacts avec des éphémérides personnalisées :

1. **List-then-reserve** flux de travail : Répertoriez d'abord les fenêtres de contact disponibles en utilisant [ListContacts](#), puis sélectionnez et réservez une fenêtre spécifique. Cette approche est utile lorsque vous souhaitez voir toutes les opportunités disponibles avant de faire une sélection.
2. **Processus de réservation directe** : réservez directement un contact pour un créneau horaire spécifique sans avoir d'abord répertorié les contacts disponibles. Cette approche est utile lorsque

vous connaissez déjà l'heure de contact souhaitée ou que vous travaillez avec des horaires prédéterminés.

Les deux flux de travail sont valides et le choix dépend de vos exigences opérationnelles. Les sections suivantes fournissent des exemples de chaque approche.

Flux de travail 1 : Répertoire les contacts disponibles puis réserver

Ce flux de travail recherche d'abord les fenêtres de contact disponibles, puis réserve une fenêtre spécifique. Cela est utile lorsque vous souhaitez voir toutes les opportunités disponibles avant de faire une sélection.

Exemple : liste et réserve avec éphémérides d'altitude azimutale

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

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

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

```

        {"dt": 540.0, "az": 60.0, "el": 25.0},
        {"dt": 720.0, "az": 65.0, "el": 30.0},
        {"dt": 900.0, "az": 70.0, "el": 35.0},
    ],
    }
],
}
},
),
)

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

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

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

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

```

```

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

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

```

Exemple : Répertoire et réserver avec les éphémérides TLE

```

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

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

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

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

```

```

        "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934
14.57114995111906",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc),
            "endTime": datetime.now(timezone.utc) + timedelta(days=7),
        },
    }
]
}
},
)

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

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

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

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

if contacts["contactList"]:
    # Reserve the first available contact

```

```

contact = contacts["contactList"][0]
print(f"Reserving contact from {contact['startTime']} to {contact['endTime']}...")

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

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

```

Workflow 2 : réservation par contact direct

Ce flux de travail réserve directement un contact sans répertorier au préalable les fenêtres disponibles. Cette approche est utile lorsque vous connaissez déjà l'heure de contact souhaitée ou que vous implémentez une planification automatisée.

Exemple : réservation directe avec éphémérides d'élévation d'azimut

```

import boto3
from datetime import datetime, timezone
import time

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

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

# Create azimuth elevation ephemeris for the specific contact time
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact AzEl Ephemeris",
    ephemeris={
        "azEl": {

```

```

    "groundStation": "Ohio 1",
    "data": {
      "azElData": {
        "angleUnit": "DEGREE_ANGLE",
        "azElSegmentList": [
          {
            "referenceEpoch": contact_start.isoformat(),
            "validTimeRange": {
              "startTime": contact_start.isoformat(),
              "endTime": contact_end.isoformat(),
            },
            "azElList": [
              {"dt": 0.0, "az": 45.0, "el": 10.0},
              {"dt": 180.0, "az": 50.0, "el": 15.0},
              {"dt": 360.0, "az": 55.0, "el": 20.0},
              {"dt": 540.0, "az": 60.0, "el": 25.0},
              {"dt": 720.0, "az": 65.0, "el": 30.0},
              {"dt": 900.0, "az": 70.0, "el": 35.0},
            ],
          },
        ],
      },
    },
  },
)

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

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

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

```

```
reservation = ground_station_client.reserve_contact(
    # Note: satelliteArn is omitted for azimuth elevation
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-profile",
    groundStation="Ohio 1",
    startTime=contact_start,
    endTime=contact_end,
    trackingOverrides={"programTrackSettings": {"azEl": {"ephemerisId":
ephemeris_id}}},
)

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

Exemple : réservation directe avec TLE ephemeris

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

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

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

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

# Create TLE ephemeris
print("Creating TLE ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact TLE Ephemeris",
    satelliteId=satellite_id,
    enabled=True,
    expirationTime=contact_end + timedelta(days=1),
    priority=1,
    ephemeris={
        "tle": {
            "tleData": [
                {
```

```

        "tleLine1": "1 25994U 99068A 24080.50000000 .00000075 00000-0
26688-4 0 9999",
        "tleLine2": "2 25994 98.2007 35.6589 0001234 89.2782 18.9934
14.57114995112000",
        "validTimeRange": {
            "startTime": (contact_start - timedelta(hours=1)).isoformat(),
            "endTime": (contact_end + timedelta(hours=1)).isoformat(),
        },
    }
]
}
},
)

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

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

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

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

```

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

Surveillance des changements d'état des contacts

Après avoir réservé un contact, il est important de surveiller son état afin de s'assurer qu'il effectue une transition réussie SCHEDULED et d'être informé de tout problème. AWS Ground Station envoie des événements à Amazon EventBridge pour tous les changements d'état des contacts.

Les états de contact suivent ce cycle de vie :

- SCHEDULING- Le contact est en cours de traitement pour la planification
- SCHEDULED- Le contact a été planifié avec succès et sera exécuté
- FAILED_TO_SCHEDULE- Le contact n'a pas pu être planifié (état du terminal)

Pour plus d'informations sur les états et le cycle de vie des contacts, consultez [Comprendre le cycle de vie des contacts](#).

Mettre en œuvre le suivi de l'état des contacts avec EventBridge

Pour surveiller les changements d'état des contacts en temps réel, vous pouvez configurer une EventBridge règle Amazon qui déclenche une fonction Lambda chaque fois qu'un contact de la Ground Station change d'état. Cette approche est plus efficace et plus évolutive que l'interrogation de l'état du contact.

Étapes d'implémentation

1. Création d'une fonction Lambda pour traiter les événements de changement d'état des contacts
2. Créez une EventBridge règle qui correspond aux événements de changement d'état des contacts de la Ground Station
3. Ajouter la fonction Lambda comme cible pour la règle

Exemple de gestionnaire de fonctions Lambda

Pour un exemple complet de fonction Lambda qui traite les événements de changement d'état des contacts, consultez la `GroundStationCloudWatchEventHandlerLambda` ressource du modèle. `AquaSnppJpssTerraDigIF.yml` CloudFormation Ce modèle est disponible dans le compartiment Amazon S3 d'accueil du AWS Ground Station client. Pour obtenir des instructions sur l'accès à ce modèle, consultez la [Assemblage](#) section de l'exemple de point de terminaison du flux de données.

EventBridge configuration des règles

La EventBridge règle doit utiliser le modèle d'événement suivant pour correspondre à tous les changements d'état des contacts de la Ground Station :

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

Pour filtrer uniquement des états spécifiques (par exemple, les défaillances), vous pouvez ajouter un filtre détaillé :

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

Pour obtenir des instructions détaillées sur la création de EventBridge règles avec des cibles Lambda, consultez la section [Création de règles qui réagissent aux événements](#) dans le guide de EventBridge l'utilisateur Amazon.

Configuration de EventBridge règles pour l'automatisation

Vous pouvez créer des EventBridge règles pour répondre automatiquement aux changements d'état des contacts. Par exemple :

- Envoyer des notifications lorsqu'un contact ne parvient pas à planifier
- Déclenchez des fonctions Lambda pour préparer les ressources lorsqu'un contact entre PREPASS
- Enregistrez les contacts complétés à des fins d'audit

Pour obtenir des informations détaillées sur la configuration EventBridge des règles relatives aux AWS Ground Station événements, consultez [Automatisez AWS Ground Station avec des événements](#).

Bonnes pratiques et considérations

Gestion des conflits de planification

Les AWS Ground Station antennes étant des ressources partagées, une fenêtre de contact qui apparaît disponible `ListContacts` peut être réservée par un autre client avant que vous ne puissiez la réserver. Pour gérer cela :

1. Vérifiez toujours l'état du contact après la réservation
2. Implémenter une logique de nouvelle tentative avec des fenêtres temporelles alternatives
3. Envisagez de réserver des contacts bien à l'avance lorsque cela est possible
4. Utiliser EventBridge les événements pour surveiller les `FAILED_TO_SCHEDULE` états

Calendrier de validation des éphémérides

N'oubliez pas que les éphémérides doivent être en `ENABLED` état avant de pouvoir les utiliser pour réserver des contacts. Le processus de validation prend généralement de quelques secondes à quelques minutes selon le type et la taille des éphémérides. Vérifiez toujours le statut des éphémérides avant de tenter de réserver des contacts.

Considérations relatives au calendrier des contacts

Lorsque vous utilisez des éphémérides personnalisées :

- Assurez-vous que votre éphéméride couvre toute la durée du contact
- [Pour les éphémérides d'élévation d'azimut, vérifiez que les angles maintiennent l'antenne au-dessus du masque de site pendant toute la durée du contact](#)
- Tenez compte des délais d'expiration des éphémérides lors de la planification de futurs contacts

Différences entre les API par type d'éphéméride

L'`ReserveContactAPI` se comporte différemment selon le type d'éphéméride :

Type d'éphéméride	Learn par satellite requis	Suivi des annulations requis
TLE	Oui	Non (facultatif)
OEM	Oui	Non (facultatif)
Élévation de l'azimut	Non (facultatif)	Oui

Comprendre quelle éphéméride est utilisée

Les éphémérides ont une priorité, une date d'expiration et un indicateur activé. Ensemble, ils déterminent quelles éphémérides sont utilisées pour le suivi lors d'un contact.

Éphémérides TLE et OEM

Pour les éphémérides OEM et TLE, une seule éphéméride peut être active pour chaque satellite. L'éphéméride qui sera utilisée est l'éphéméride activée la plus prioritaire dont la date d'expiration se situe dans le futur. Une valeur de priorité plus élevée indique une priorité plus élevée. Les temps de contact disponibles renvoyés par [ListContacts](#) sont basés sur cette éphéméride. Si plusieurs ENABLED éphémérides ont la même priorité, les éphémérides les plus récemment créées ou mises à jour seront utilisées.

Note

AWS Ground Station [dispose d'un quota de service sur le nombre d'éphémérides ENABLED fournies par le client par satellite \(voir : Quotas de service\)](#). Pour télécharger des données d'éphémérides après avoir atteint ce quota, supprimez (en utilisant [DeleteEphemeris](#)) ou désactivez (en utilisant) les éphémérides les moins [UpdateEphemeris](#) prioritaires/les plus anciennes créées par le client.

[Si aucune éphéméride n'a été créée, ou si aucune éphéméride n'a de ENABLED statut, une éphéméride par défaut AWS Ground Station sera utilisée pour le satellite \(depuis Space-Track\), si elle est disponible.](#) Cette éphéméride par défaut a la priorité 0.

Éphémérides d'élévation d'azimut

Les éphémérides d'élévation d'azimut fonctionnent différemment des éphémérides OEM et TLE. Chaque éphéméride d'altitude azimutale est associée à une station au sol spécifique et n'a pas de priorité. Lorsque vous réservez un contact avec des éphémérides d'altitude d'azimut, vous spécifiez explicitement les éphémérides d'altitude d'azimut à utiliser par le biais du paramètre `trackingOverrides`.

Principales différences pour les éphémérides d'élévation d'azimut :

- Pas de système de priorité : vous sélectionnez explicitement les éphémérides pour chaque contact
- Spécifique à une station au sol : chaque éphéméride est associée à une station au sol particulière
- Aucune solution de secours automatique : si l'éphéméride spécifiée n'est pas disponible, le contact échouera

Note

Les éphémérides d'élévation d'azimut ne sont pas en concurrence avec les éphémérides OEM et TLE. Ils sont sélectionnés explicitement lors de la réservation d'un contact et ne sont utilisés que lorsque des dérogations au suivi sont spécifiées.

Effet des nouvelles éphémérides sur les contacts précédemment programmés

Utilisez l'[DescribeContact API](#) pour visualiser les effets des nouvelles éphémérides sur les contacts précédemment planifiés en renvoyant les durées de visibilité actives.

Pour les éphémérides OEM et TLE, les contacts planifiés avant le téléchargement d'une nouvelle éphéméride conserveront l'heure de contact initialement prévue, tandis que le suivi de l'antenne utilisera les éphémérides actives. Si la position de l'engin spatial, basée sur les éphémérides actives, diffère considérablement de celle des éphémérides précédentes, cela peut entraîner une réduction du temps de contact du satellite avec l'antenne en raison du fait que l'engin spatial fonctionne en dehors du masque du site. Par conséquent, nous vous recommandons d'annuler et de reprogrammer vos futurs contacts après avoir chargé une nouvelle éphéméride très différente de l'éphéméride précédente.

Grâce à l'[DescribeContact API](#), vous pouvez déterminer la partie de votre futur contact qui est inutilisable en raison du fonctionnement du vaisseau spatial en dehors du masque du transmit/receive site en comparant votre contact `startTime` prévu `endTime` avec le `visibilityStartTime` et `visibilityEndTime` renvoyé. Si vous choisissez d'annuler et de reprogrammer vos futurs contacts, la plage de temps de contact ne doit pas dépasser la plage de visibilité de plus de 30 secondes. Les contacts annulés peuvent entraîner des frais s'ils sont annulés trop près du moment du contact. Pour plus d'informations sur les contacts annulés, voir : [Ground Station FAQs](#).

Pour les éphémérides d'altitude azimutale, les contacts planifiés utiliseront les éphémérides spécifiques sélectionnées lors de la réservation du contact. Si vous devez mettre à jour les données d'altitude azimutale d'un contact programmé, vous pouvez annuler et reprogrammer le contact avec une nouvelle éphéméride.

Obtenir l'éphéméride actuelle d'un satellite

Les éphémérides actuellement utilisées par AWS Ground Station un satellite spécifique peuvent être récupérées en appelant les actions [GetSatellite](#) ou [ListSatellites](#). Ces deux méthodes renverront des métadonnées pour les éphémérides actuellement utilisées. Ces métadonnées d'éphémérides sont différentes pour les éphémérides personnalisées téléchargées vers AWS Ground Station et pour les éphémérides par défaut.

Note

Les éphémérides d'altitude azimutale ne sont pas associées aux satellites et ne sont donc pas renvoyées par `GetSatellite` ou `ListSatellites`. Pour récupérer des informations sur les éphémérides d'altitude azimutale, utilisez l'[DescribeEphemeris](#) API avec l'identifiant d'éphéméride spécifique ou utilisez-la pour voir toutes les éphémérides disponibles [ListEphemerides](#) pour votre compte.

Les éphémérides par défaut source incluront uniquement les champs `epoch` et `epochC`. C'est l'[époque](#) de l'[ensemble d'éléments à deux lignes](#) extrait de [Space-Track](#), et il est actuellement utilisé pour calculer la trajectoire du satellite.

Une éphéméride personnalisée aura une source valeur de `CUSTOMER_PROVIDED` et inclura un identifiant unique dans le `ephemerisId` champ. Cet identifiant unique peut être utilisé pour rechercher les éphémérides via l'[DescribeEphemeris](#) action. Un `name` champ facultatif sera

renvoyé si un nom a été attribué à l'éphéméride lors du téléchargement AWS Ground Station via l'[CreateEphemeris](#)action.

Il est important de noter que les éphémérides sont mises à jour dynamiquement AWS Ground Station afin que les données renvoyées ne soient qu'un instantané des éphémérides utilisées au moment de l'appel à l'API.

Exemple de [GetSatellite](#)retour pour un satellite utilisant une éphéméride par défaut

```
{
  "satelliteId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "satelliteArn": "arn:aws:groundstation::111122223333:satellite/e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "noradSatelliteID": 25994,
  "groundStations": [
    "Ohio 1",
    "Oregon 1"
  ],
  "currentEphemeris": {
    "source": "SPACE_TRACK",
    "epoch": 1528245583.619
  }
}
```

Exemple [GetSatellite](#)de satellite utilisant une éphéméride personnalisée

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

```
}
```

Liste des éphémérides d'élévation d'azimut

Comme les éphémérides d'altitude azimutale ne sont pas associées aux satellites, vous devez utiliser différentes méthodes pour découvrir et récupérer les informations API les concernant :

1. [ListEphemerides](#) À utiliser pour répertorier toutes les éphémérides de votre compte, y compris les éphémérides d'altitude azimutale. Vous pouvez filtrer par statut et par type d'éphéméride.
2. [DescribeEphemeris](#) À utiliser avec un identifiant d'éphéméride spécifique pour obtenir des informations détaillées sur une éphéméride d'altitude azimutale.
3. [DescribeContact](#) À utiliser avec un identifiant de contact spécifique pour obtenir des informations détaillées sur une éphéméride utilisée pour le contact.

Exemple de [ListEphemerides](#) réponse incluant une éphéméride d'élévation d'azimut :

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

Note

Dans la [ListEphemerides](#) réponse, les éphémérides à élévation d'azimut auront un `groundStation` champ au lieu d'un `satelliteId` champ, ce qui les rendra faciles à identifier.

Revenir aux données d'éphémérides par défaut

Lorsque vous téléchargez des données d'éphémérides personnalisées, elles remplacent les éphémérides AWS Ground Station utilisées par défaut pour ce satellite en particulier. AWS Ground Station n'utilise pas à nouveau les éphémérides par défaut tant qu'aucune éphéméride non expirée fournie par le client n'est actuellement disponible. AWS Ground Station ne répertorie pas non plus les contacts après la date d'expiration des éphémérides actuellement fournies par le client, même si une éphéméride par défaut est disponible après cette date d'expiration.

Note

Les éphémérides d'altitude azimutale n'ont pas de valeurs par défaut et ne remplacent pas les éphémérides satellites. Ils sont sélectionnés de manière explicite lors de la réservation d'un contact à l'aide du `trackingOverrides` paramètre. Si vous ne souhaitez plus utiliser les éphémérides d'altitude azimutale, il vous suffit de réserver des contacts sans spécifier de dérogations de suivi, et le système utilisera les éphémérides satellites actives à la place.

Inversion des éphémérides TLE et OEM

Pour revenir aux éphémérides [Space-Track](#) par défaut pour un satellite, vous devez effectuer l'une des opérations suivantes :

- Supprimez (en utilisant [DeleteEphemeris](#)) ou désactivez (en utilisant [UpdateEphemeris](#)) toutes les éphémérides activées fournies par le client. Vous pouvez répertorier les éphémérides fournies par le client pour un satellite à l'aide de. [ListEphemerides](#)
- Attendez que toutes les éphémérides existantes fournies par le client expirent.

Vous pouvez confirmer que l'éphéméride par défaut est utilisée en appelant [GetSatellite](#) et en vérifiant que la source éphéméride actuelle du satellite est bien celle du satellite. `SPACE_TRACK` Voir [Données d'éphémérides par défaut](#) pour plus d'informations sur les éphémérides par défaut.

Gestion des éphémérides d'élévation d'azimut

Étant donné que les éphémérides d'altitude azimutale sont sélectionnées de manière explicite pour chaque contact et ne sont pas associées aux satellites, il n'existe aucun concept de « retour » à une valeur par défaut. Vous pouvez plutôt gérer les éphémérides d'altitude azimutale comme suit :

- Pour arrêter d'utiliser les éphémérides d'élévation d'azimut : il suffit de réserver de nouveaux contacts sans spécifier ni spécifier un. `trackingOverrides` `satelliteArn` Le contact utilisera plutôt les éphémérides actives du satellite spécifié.
- Pour supprimer les éphémérides d'altitude d'azimut non utilisées : utilisez cette option [DeleteEphemeris](#) pour supprimer les éphémérides d'altitude d'azimut qui ne sont plus nécessaires. Notez que vous ne pouvez pas supprimer une éphéméride actuellement utilisée par un contact programmé.

Pour répertorier toutes les éphémérides d'élévation d'azimut de votre compte, utilisez [ListEphemerides](#). Les éphémérides d'élévation d'azimut peuvent être identifiées par le `ephemerisType` champ ou par la présence d'un champ au lieu d'un `groundStation` champ dans la réponse. `satelliteId`

Travailler avec des flux de données

AWS Ground Station utilise une relation entre nœud et périphérie pour créer des flux de données afin de permettre le traitement en flux de vos données. Chaque nœud est représenté par une configuration qui décrit le traitement attendu. Pour illustrer ce concept, considérez un flux de données allant de `antenna-downlink` à `s3-recording`. Le `antenna-downlink` nœud représente la transformation analogique-numérique du spectre des fréquences radio selon les paramètres définis dans la configuration. Il `s3-recording` représente un nœud de calcul qui recevra les données entrantes et les stockera dans votre compartiment S3. Le flux de données qui en résulte est une livraison asynchrone de données RF numérisées vers un compartiment S3 en fonction de vos spécifications.

Dans le cadre de votre profil de mission, vous pouvez créer de nombreux flux de données pour répondre à vos besoins. Les sections suivantes décrivent comment configurer vos autres ressources AWS à utiliser AWS Ground Station et proposent des recommandations pour créer des flux de données. Pour obtenir des informations détaillées sur le comportement de chaque nœud, notamment s'il est considéré comme un nœud source ou de destination, consultez [Utiliser les AWS Ground Station configurations](#).

Rubriques

- [AWS Ground Station interfaces de plan de données](#)
- [Utiliser la diffusion de données entre régions](#)
- [Installation et configuration d'Amazon S3](#)
- [Configuration et configuration d'Amazon VPC](#)
- [Configurer et configurer Amazon EC2](#)

AWS Ground Station interfaces de plan de données

La structure de données résultante du flux de données que vous avez choisi dépend de la source du flux de données. Les détails de ces formats vous sont fournis lors de l'embarquement de vos satellites. Les formats utilisés pour chaque type de flux de données sont résumés ci-dessous.

- liaison descendante entre antennes
 - (Bande passante less-than-or-equal -40MHz) les données sont fournies sous forme de paquets de données de [signal/format IP VITA-49](#).

- (Bande passante supérieure à 40MHz) les données sont livrées sous forme de paquets de AWS Ground Station classe 2.
- antenna-downlink-demod-decode
 - Les données sont livrées sous forme de Demodulated/Decoded paquets au format Data/IP.
- antenne-uplink
 - Les données doivent être fournies sous forme de paquets de [données de signal/format IP VITA-49](#).
- antenna-uplink-echo
 - Les données sont fournies sous forme de paquets de [données de signal/format IP VITA-49](#).

Utiliser la diffusion de données entre régions

La fonction de transmission de données AWS Ground Station entre régions vous donne la flexibilité d'envoyer vos données depuis une antenne vers n'importe quelle AWS région AWS Ground Station prise en charge. Cela signifie que vous pouvez gérer votre infrastructure dans une seule région AWS et planifier des contacts pour toutes celles dans lesquelles [AWS Ground Station Succursales](#) vous êtes intégré.

Lorsque vous recevrez vos coordonnées dans un compartiment Amazon S3, il AWS Ground Station gèrera tous les aspects de la livraison pour vous.

Pour utiliser la livraison de données entre régions vers une EC2 instance Amazon (à l'aide de l' AWS Ground Station agent ou d'un point de terminaison de flux de données), le point de terminaison du flux de données doit être créé dans votre région AWS actuelle et vous devez spécifier la même région. dataflow-endpoint-config AWS Ground Station gèrera pour vous la diffusion des données entre les régions.

Installation et configuration d'Amazon S3

Vous pouvez utiliser un compartiment Amazon S3 pour recevoir vos signaux de liaison descendante en utilisant AWS Ground Station. Pour créer le s3-recording-config de destination, vous devez être en mesure de spécifier un compartiment Amazon S3 et un rôle IAM qui autorise AWS Ground Station l'écriture de fichiers dans le compartiment.

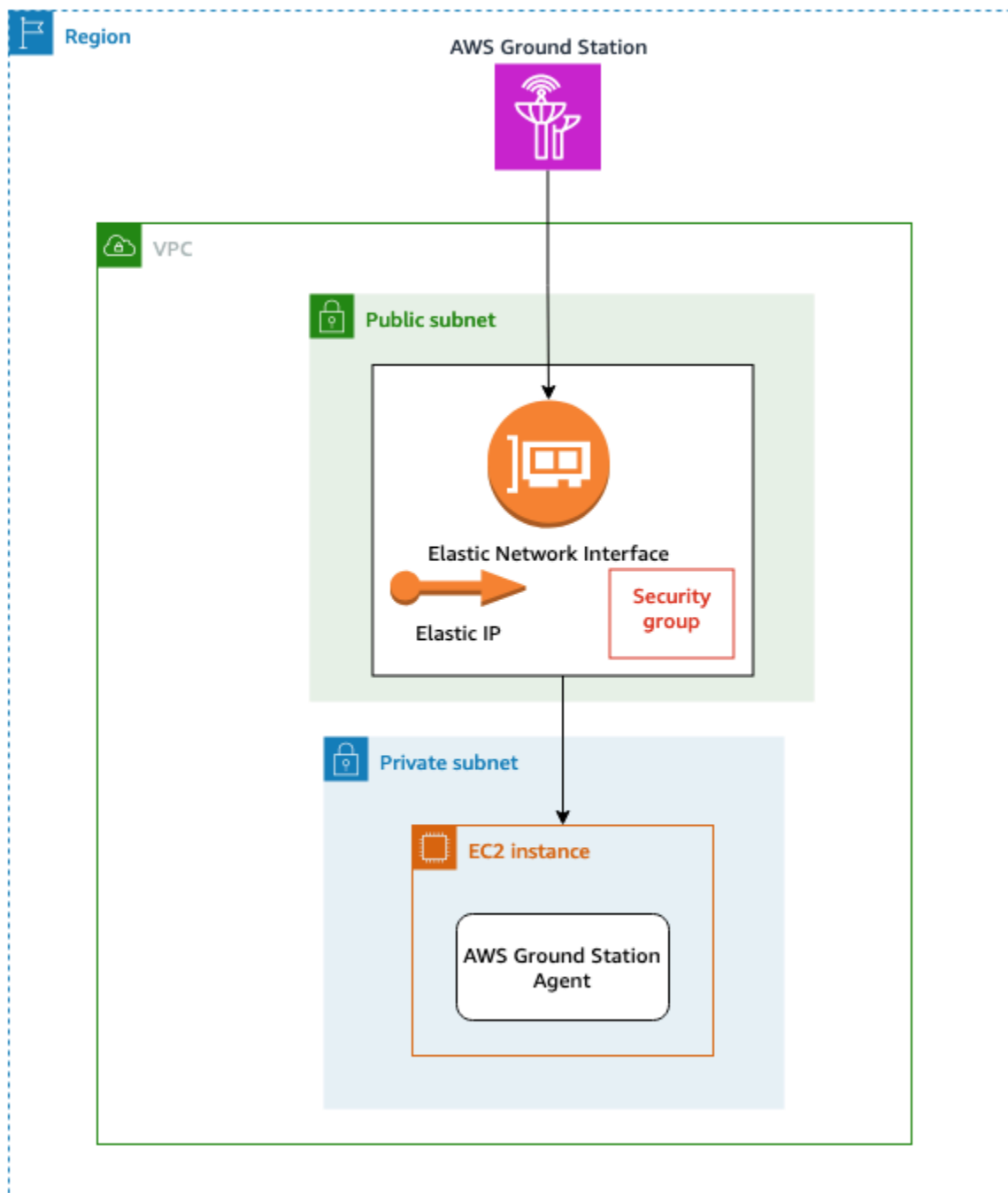
Consultez [Config d'enregistrement Amazon S3](#) les restrictions relatives au compartiment Amazon S3, au rôle IAM ou à la création de AWS Ground Station configuration.

Configuration et configuration d'Amazon VPC

Un guide complet pour configurer un VPC dépasse le cadre de ce guide. Pour une compréhension approfondie, veuillez consulter le guide de [l'utilisateur Amazon VPC](#).

Dans cette section, il est décrit comment votre point de terminaison Amazon EC2 et votre point de terminaison de flux de données peuvent exister au sein d'un VPC. AWS Ground Station ne prend pas en charge plusieurs points de livraison pour un flux de données donné ; il est prévu que chaque flux de données se termine vers un seul récepteur. EC2 Comme nous nous attendons à un seul EC2 récepteur, la configuration n'est pas redondante multi-AZ. Pour des exemples complets qui utiliseront votre VPC, veuillez consulter. [Exemples de configurations de profil de mission](#)

Configuration VPC avec agent AWS Ground Station



Vos données satellites sont fournies à une instance d' AWS Ground Station agent située à proximité de l'antenne. L' AWS Ground Station agent rayera puis cryptera vos données à l'aide de la AWS KMS clé que vous fournissez. Chaque bande est envoyée à votre adresse [Amazon EC2 Elastic IP \(EIP\)](#) depuis l'antenne source sur le backbone du réseau AWS. Les données arrivent à votre EC2 instance via l'[Amazon EC2 Elastic Network Interface \(ENI\)](#) ci-jointe. Une fois sur votre EC2 instance,

L'AWS Ground Station agent installé déchiffre vos données et effectue une correction d'erreur directe (FEC) pour récupérer les données perdues, puis les transmet à l'adresse IP et au port que vous avez spécifiés dans votre configuration.

La liste ci-dessous présente les considérations de configuration uniques à prendre en compte lors de la configuration de votre VPC pour la livraison de l'agent AWS Ground Station.

Groupe de sécurité - Il est recommandé de configurer un groupe de sécurité dédié uniquement au trafic AWS Ground Station. Ce groupe de sécurité doit autoriser le trafic entrant UDP sur la même plage de ports que vous spécifiez dans votre groupe de points de terminaison Dataflow. AWS Ground Station gère une liste de préfixes gérée par AWS afin de limiter vos autorisations aux seules AWS Ground Station adresses IP. Consultez les [listes de préfixes gérées par AWS](#) pour savoir comment remplacer les dans PrefixListIdvos régions de déploiement.

Elastic Network Interface (ENI) : vous devez associer le groupe de sécurité ci-dessus à cette ENI et le placer dans votre sous-réseau public.

Note

Le quota par défaut pour le nombre de groupes de sécurité attachés par ENI est de 5. Il s'agit d'une limite ajustable jusqu'à 16, voir [Amazon VPC Quotas](#).

Le CloudFormation modèle suivant montre comment créer l'infrastructure décrite dans cette section.

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:

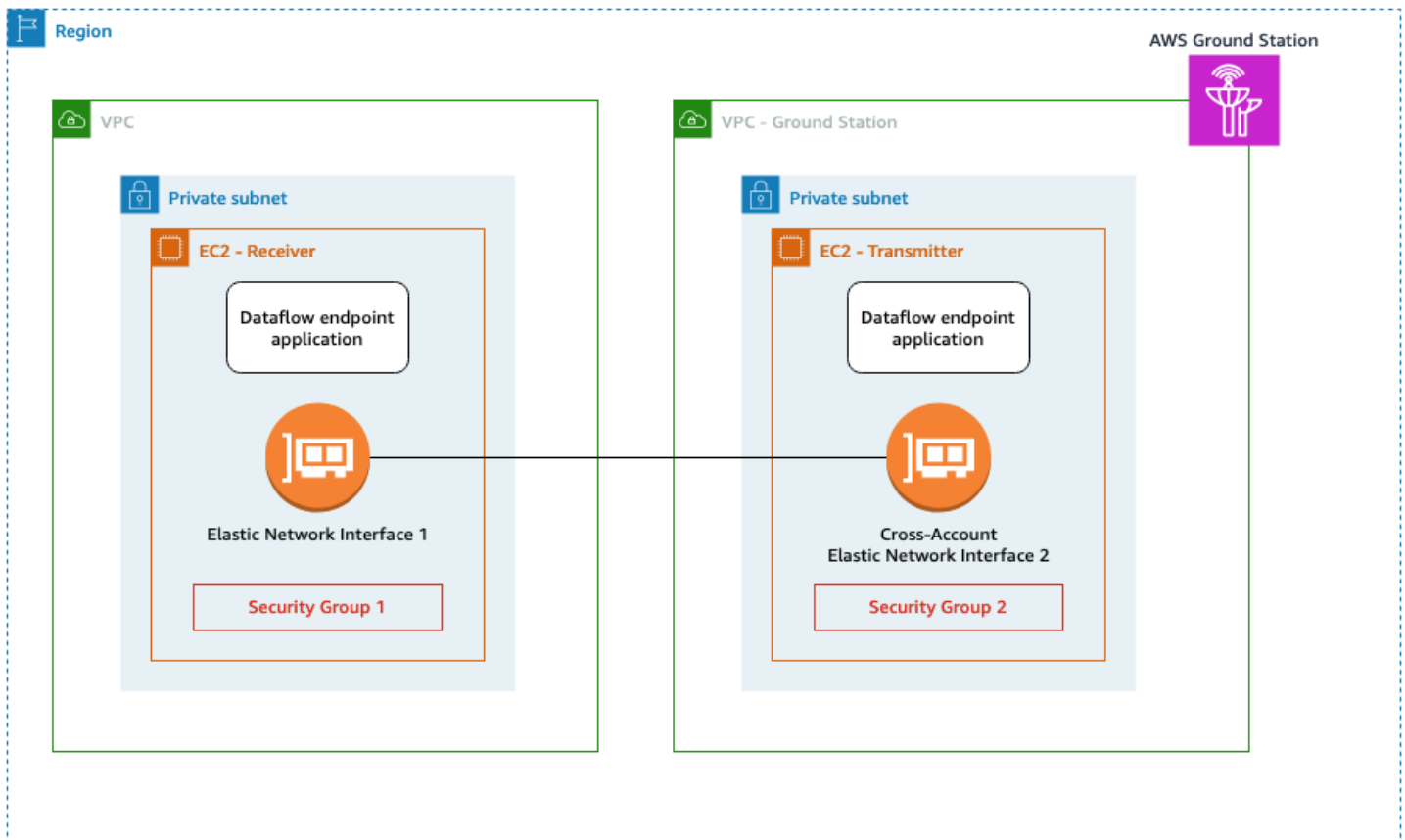
AllocationId:

Fn::GetAtt: [*ReceiveInstanceEIP*, AllocationId]

NetworkInterfaceId:

Ref: *InstanceNetworkInterface*

Configuration VPC avec un point de terminaison de flux de données



Vos données satellites sont fournies à une instance d'application de point de terminaison de flux de données située à proximité de l'antenne. Les données sont ensuite envoyées via [Amazon EC2 Elastic Network Interface \(ENI\)](#) multi-comptes à partir d'un VPC appartenant à AWS Ground Station. Les données arrivent ensuite à votre EC2 instance via l'ENI attachée à votre EC2 instance Amazon. L'application de point de terminaison de flux de données installée le transmettra ensuite à l'adresse IP et au port que vous avez spécifiés dans votre configuration. L'inverse de ce flux se produit pour les connexions en liaison montante.

La liste ci-dessous présente des considérations de configuration uniques lors de la configuration de votre VPC pour la livraison de points de terminaison de flux de données.

Note

Le quota par défaut pour le nombre de groupes de sécurité attachés par ENI est de 5. Il s'agit d'une limite ajustable jusqu'à 16, voir [Amazon VPC Quotas](#).

Rôle IAM : le rôle IAM fait partie du point de terminaison du flux de données et n'apparaît pas dans le diagramme. Rôle IAM utilisé pour créer et associer l'ENI entre comptes à l'instance AWS Ground Station Amazon EC2.

Groupe de sécurité 1 : ce groupe de sécurité est attaché à l'ENI qui sera associé à l' EC2 instance Amazon de votre compte. Il doit autoriser le trafic UDP en provenance du groupe de sécurité 2 sur les ports spécifiés dans votre dataflow-endpoint-group.

Elastic Network Interface (ENI) 1 : vous devez associer le groupe de sécurité 1 à cet ENI et le placer dans un sous-réseau.

Sous-réseau : vous devez vous assurer qu'au moins une adresse IP est disponible par flux de données pour l' EC2 instance Amazon de votre compte. Pour plus de détails sur le dimensionnement des sous-réseaux, voir Blocs CIDR de [sous-réseau](#)

Groupe de sécurité 2 : ce groupe de sécurité est référencé dans le point de terminaison Dataflow. Ce groupe de sécurité sera rattaché à l'ENI qui AWS Ground Station sera utilisé pour placer des données dans votre compte.

Région - Pour plus d'informations sur les régions prises en charge pour les connexions entre régions, voir [Utiliser la diffusion de données entre régions](#).

Le CloudFormation modèle suivant montre comment créer l'infrastructure décrite dans cette section.

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
subnet 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
```

Configurer et configurer Amazon EC2

La configuration correcte de votre EC2 instance Amazon est nécessaire pour que la livraison synchrone de VITA-49 soit fournie via Signal/IP data or VITA-49 Extension data/IP l' AWS Ground Station agent ou un point de terminaison de flux de données. Selon vos besoins spécifiques, vous pouvez exécuter le processeur frontal (FE) ou la radio définie par logiciel (SDR) directement sur la même instance, ou vous devrez peut-être utiliser des EC2 instances supplémentaires. La sélection et l'installation de votre FE ou SDR ne sont pas abordées dans ce guide de l'utilisateur. Pour plus d'informations sur les formats de données spécifiques, consultez [AWS Ground Station interfaces de plan de données](#).

Pour plus d'informations sur nos conditions de service, veuillez consulter les [conditions AWS de service](#).

Logiciel commun fourni

AWS Ground Station fournit des logiciels courants pour faciliter la configuration de votre EC2 instance Amazon.

AWS Ground Station Agent

L' AWS Ground Station agent reçoit les données de liaison descendante à fréquence intermédiaire numérique (DigiF) et sort les données déchiffrées qui permettent ce qui suit :

- Capacité de liaison descendante DigiF de 40 à 400 MHz % MHz de bande passante.
- Livraison de données DigiF à haut débit et à faible instabilité vers n'importe quelle adresse IP publique AWS (IP élastique) du réseau. AWS
- Livraison de données fiable grâce à la correction d'erreur directe (FEC).
- Livraison sécurisée des données à l'aide d'une AWS KMS clé de chiffrement gérée par le client.

Pour plus d'informations, consultez le [Guide de AWS Ground Station l'utilisateur de l'agent](#).

Application de point de terminaison Dataflow

Application réseau utilisée pour envoyer et recevoir des données entre les emplacements des AWS Ground Station antennes et vos EC2 instances Amazon. AWS Ground Station II peut être utilisé pour la liaison montante et la liaison descendante des données.

Radio définie par logiciel (SDR)

Radio définie par logiciel (SDR) qui peut être utilisée pour moduler/démoduler le signal utilisé pour communiquer avec votre satellite.

AWS Ground Station Images de machines Amazon (AMIs)

Pour réduire les temps de construction et de configuration de ces installations, des offres AMIs préconfigurées sont AWS Ground Station également disponibles. L'application réseau AMIs dotée d'un flux de données et d'une radio définie par logiciel (SDR) sont mises à la disposition de votre compte une fois votre intégration terminée. Vous pouvez les trouver dans la EC2 console Amazon en recherchant Groundstation dans le fichier privé [Amazon Machine Images \(AMIs\)](#). Les AMIs with AWS Ground Station Agent sont publics et peuvent être trouvés dans la EC2 console Amazon en recherchant Groundstation dans les [Amazon Machine Images publiques \(AMIs\)](#).

Travailler avec la télémétrie

AWS Ground Station la télémétrie fournit des mesures en temps quasi réel à partir des AWS Ground Station antennes lors de vos contacts avec les satellites. Vous pouvez utiliser les données de télémétrie pour surveiller les performances des contacts, détecter les anomalies et prendre des décisions éclairées concernant vos communications par satellite.

Rubriques

- [Comment fonctionne la télémétrie](#)
- [Types de télémétrie disponibles](#)
- [Disponibilité par région](#)
- [Configuration de la télémétrie](#)
- [Comprendre les données de télémétrie](#)

Comment fonctionne la télémétrie

Pour utiliser la télémétrie, vous configurez un `TelemetrySinkConfig` qui indique où les données de télémétrie AWS Ground Station doivent être transmises. Vous ajoutez ensuite cette configuration à votre profil de mission à l'aide du `telemetrySinkConfigArn` champ. Lors de contacts utilisant un profil de mission compatible avec la télémétrie, les données de télémétrie AWS Ground Station sont transmises à votre compte.

Le processus de transmission de données télémétriques fonctionne comme suit :

1. Vous créez un flux Kinesis Data Streams dans AWS votre compte pour recevoir des données de télémétrie. Le stream doit être créé dans le même compte et dans la même région que ceux à partir desquels vous programmez vos contacts.
2. Vous créez un rôle IAM qui AWS Ground Station autorise l'écriture de données dans votre flux.
3. Vous créez un `TelemetrySinkConfig` qui fait référence à votre flux et à votre rôle IAM.
4. Vous les ajoutez `TelemetrySinkConfig` à votre profil de mission.
5. Vous listez et réservez des contacts à l'aide du nouveau profil de mission compatible avec la télémétrie.
6. Lors des contacts utilisant ce profil de mission, les données de télémétrie sont diffusées en temps quasi réel sur votre AWS Ground Station flux Kinesis Data Streams.

7. Vous consommez et traitez les données de télémétrie de votre flux à l'aide de AWS services ou de vos propres applications.

Types de télémétrie disponibles

AWS Ground Station fournit les types de télémétrie suivants lors des contacts :

Note

AWS Ground Station travaille à l'augmentation du nombre de types de télémétrie pris en charge

Télémétrie de pointage

Fournit des informations sur la direction de pointage de l'antenne lors des contacts avec le satellite. Ce type de télémétrie est toujours envoyé lors d'un contact et inclut les angles d'azimut et d'élévation réels et commandés. Pour de plus amples informations, veuillez consulter [Télémétrie de pointage](#).

Télémétrie de suivi

Fournit des informations sur l'état du suivi de l'antenne et les erreurs de suivi. Ce type de télémétrie est envoyé lorsque le suivi automatique est activé dans votre configuration de suivi. Pour de plus amples informations, veuillez consulter [Télémétrie de suivi](#).

Disponibilité par région

La télémétrie est disponible dans toutes les AWS régions où AWS Ground Station elle est implantée. Pendant l'exécution du contact, la télémétrie sera transmise depuis l' AWS Ground Station antenne à la région depuis laquelle vous avez programmé votre contact, fournissant ainsi une assistance interrégionale.

Pour une liste complète des AWS Ground Station régions et des emplacements des stations au sol, voir [AWS Ground Station Succursales](#).

Configuration de la télémétrie

Suivez ces étapes pour configurer la télémétrie pour vos AWS Ground Station contacts. Une fois cette configuration terminée, les données de télémétrie seront transmises à votre flux Kinesis Data Streams lors des contacts utilisant un profil de mission compatible avec la télémétrie. Pour une compréhension approfondie de Kinesis Data Streams, veuillez consulter le Guide de [l'utilisateur de Kinesis Data Streams](#).

Étape 1 : créer des AWS ressources prerequisites

L'CloudFormation extrait suivant montre comment créer les AWS ressources nécessaires à la diffusion de données télémétriques. Cet extrait crée un flux Kinesis Data Streams et un rôle IAM qui AWS Ground Station autorise l'écriture de données de télémétrie dans le flux.

TelemetryStream:

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

TelemetryRole:

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

```
- kinesis:PutRecords
Resource: !GetAtt TelemetryStream.Arn
```

La liste ci-dessous présente les considérations de configuration uniques à prendre en compte lors de la configuration de la diffusion de télémétrie pour AWS Ground Station

Flux Kinesis Data Streams : le flux utilise le mode capacité à la demande, qui évolue automatiquement en fonction du débit. Cela est recommandé pour la plupart des cas d'utilisation. Le flux est configuré pour conserver les données pendant 24 heures. Par défaut, le flux utilise le chiffrement AWS géré. Pour utiliser le chiffrement géré par le client AWS Key Management Service, ajoutez la `StreamEncryption` propriété et mettez à jour la politique de rôle IAM pour inclure les autorisations. `kms:GenerateDataKey` Pour plus d'informations, consultez la section [Protection des données dans Amazon Kinesis Data Streams](#).

Rôle IAM : le rôle IAM permet au principal du `groundstation.amazonaws.com` service d'assumer le rôle et d'écrire des données de télémétrie dans votre flux Kinesis Data Streams. La politique de rôle accorde des autorisations pour `kinesis:DescribeStream`, `kinesis:PutRecord`, et `kinesis:PutRecords` des actions sur le flux. Consultez [Config du récepteur de télémétrie](#) pour obtenir des conseils sur la mise en place de la politique de confiance et de la politique des rôles.

Configuration supplémentaire : ajoutez `iam:PassRole` des autorisations à l'utilisateur ou au rôle IAM que vous utilisez pour les appels AWS Ground Station d'API. Cela vous permet de transmettre le rôle de télémétrie AWS Ground Station lors de la création d'un `TelemetrySinkConfig`

Exemple `PassRole` de politique

Pour plus d'informations sur la façon de mettre à jour ou d'associer une politique de rôle, consultez [la section Gestion des politiques IAM](#) dans le Guide de l'utilisateur IAM. Pour plus d'informations sur l'`iam:PassRole` autorisation, consultez [Accorder à un utilisateur l'autorisation de transmettre un rôle à un service AWS](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
    },
  ],
}
```

```
"Resource": "arn:aws:iam::999999999999:role/your-telemetry-delivery-role-name"
  }
]
}
```

Étape 2 : Création d'un TelemetrySinkConfig

Créez un TelemetrySinkConfig qui définit la manière dont les données de télémétrie AWS Ground Station seront transmises à votre flux Kinesis Data Streams. Utilisez l'ARN du flux et l'ARN du rôle issus des sorties de la CloudFormation pile à l'étape 1.

Note

Lorsque vous créez un TelemetrySinkConfig, il AWS Ground Station vérifiera l'accès à votre flux Kinesis Data Streams en fournissant un enregistrement de test vide avec une clé `test` de partition de.

Pour plus d'informations sur la création d'un TelemetrySinkConfig, consultez [Config du récepteur de télémétrie](#).

Étape 3 : ajouter la télémétrie à votre profil de mission

Créez un profil de mission. Pour plus d'informations sur la création de profils de mission, consultez [Utiliser les profils AWS Ground Station de mission](#). Ajoutez-le `telemetrySinkConfigArn` à votre profil de mission pour permettre la transmission de données télémétriques lors des contacts. Utilisez l'ARN du TelemetrySinkConfig créé à l'étape 2.

Étape 4 : Planifier un contact

Planifiez un contact à l'aide de votre profil de mission compatible avec la télémétrie. Pendant le contact, AWS Ground Station diffusera les données de télémétrie vers votre flux Kinesis Data Streams.

À quoi s'attendre lors des contacts

- Démarrage de la télémétrie : les données commencent à être diffusées dès que le contact démarre.

- Livraison en temps quasi réel - La télémétrie arrive dans votre flux Kinesis Data Streams en temps quasi réel.
- Durée du contact - Les données sont conservées pendant toute la durée du contact.
- Arrêt automatique : la télémétrie arrête le streaming lorsque le contact prend fin.

Surveillance de la livraison

Vous pouvez surveiller la diffusion de données télémétriques à l'aide de :

- Statistiques relatives aux flux Kinesis Data Streams : archivez les enregistrements CloudWatch entrants. Pour plus d'informations, consultez la section [Surveillance d'Amazon Kinesis Data Streams](#).
- Journaux des applications : vérifiez le traitement des données dans vos applications consommant des données provenant du flux.
- Kinesis Data Viewer : utilisez la console de streaming Kinesis Data Streams pour consulter des exemples d'enregistrements de votre flux.

Étapes suivantes

Une fois la configuration terminée, vous pouvez :

- Découvrez le format des données de télémétrie et les types de télémétrie disponibles. Consultez [Comprendre les données de télémétrie](#).
- Créez des applications pour traiter les données de télémétrie issues de votre flux Kinesis Data Streams. Pour plus d'informations, consultez [Building Consumers for Amazon Kinesis Data Streams](#).
- Créez des tableaux de bord et des alertes à l'aide CloudWatch d'autres AWS services.
- Consultez les conseils de résolution des problèmes si vous rencontrez des problèmes. Consultez [Résoudre les problèmes de télémétrie](#).

Comprendre les données de télémétrie

Les données de télémétrie sont fournies sous forme d'enregistrements JSON codés en Base64 à votre flux Kinesis Data Streams. Chaque enregistrement contient des informations collectées lors de

vosre contact par satellite, notamment des métadonnées sur le contact et les mesures télémétriques échantillonnées.

Vue d'ensemble des formats de données

Chaque enregistrement de télémétrie contient les éléments suivants :

Type et version de télémétrie

Identifie le type spécifique de données de télémétrie et la version de son schéma. Cela vous permet d'analyser les différents types de télémétrie de manière appropriée. Pour plus d'informations sur le versionnement des schémas, consultez [Versionnement et évolution des schémas](#).

ID de portée

Identifiant unique pour l'étendue de la télémétrie. Cela vous permet de corréler les données de télémétrie avec des contacts spécifiques.

Métadonnées

Informations contextuelles sur la télémétrie.

Données

Les mesures de télémétrie échantillonnées spécifiques au type de télémétrie.

Clé de partition

Les enregistrements de télémétrie sont transmis à votre flux Kinesis Data Streams avec une clé de partition au format suivant :

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

Cette clé de partition garantit que toute la télémétrie d'un type donné pour un seul contact est transmise à la même partition au sein de votre flux Kinesis Data Streams, ce qui permet de trier au mieux le flux de télémétrie de ce contact.

Télémétrie de pointage

La télémétrie de pointage fournit des informations sur la direction de pointage de l'antenne lors des contacts avec le satellite. Ce type de télémétrie est toujours envoyé lors d'un contact.

Champs de données

Exemple d'horodatage

Heure à laquelle les données de télémétrie ont été échantillonnées, au format ISO-8601 en UTC avec une précision de la milliseconde.

azimut

Angle d'azimut réel de l'antenne en degrés.

élévation

Angle d'élévation réel de l'antenne en degrés.

Azimut commandé

Angle d'azimut commandé en degrés. Il s'agit de l'angle d'azimut cible que l'antenne tente d'atteindre.

Élévation commandée

Angle d'élévation commandé en degrés. Il s'agit de l'angle d'élévation cible que l'antenne tente d'atteindre.

Note

La position réelle de l'antenne peut différer de la position commandée en raison de limitations physiques ou de retards mécaniques pendant le contact.

Champs de métadonnées

Station au sol

Nom de la station au sol (par exemple, « Ohio 1 »).

Identifiant du satellite

Identifiant de la ressource satellite dans AWS Ground Station.

contactId

Identifiant du contact.

Exemple JSON

```
{
  "telemetryTypeAndVersion": "POINTING#1.0.0",
  "telemetryType": "POINTING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
    "satelliteId": "87654321-4321-4321-4321-210987654321",
    "contactId": "12345678-1234-1234-1234-123456789012"
  },
  "data": {
    "sampleTimestamp": "2025-12-08T12:00:00.123Z",
    "azimuth": 180.5,
    "elevation": 45.2,
    "commandedAzimuth": 180.0,
    "commandedElevation": 45.0
  }
}
```

Télémétrie de suivi

La télémétrie de suivi fournit des informations sur l'état du suivi des antennes et les erreurs de suivi. Ce type de télémétrie est envoyé lorsque le suivi automatique est activé dans votre configuration de suivi et lorsque l'antenne utilise activement le suivi automatique.

Note

Si le `autotrack` paramètre de votre `TrackingConfig` est défini sur `REMOVED`, aucune télémétrie de suivi ne sera délivrée. Pour plus d'informations sur le suivi des configurations, consultez [Suivi de Config](#).

Champs de données

Exemple d'horodatage

Heure à laquelle les données de télémétrie ont été échantillonnées, au format ISO-8601 en UTC avec une précision de la milliseconde.

État du suivi

État de suivi actuel de l'antenne. Les valeurs possibles sont :

- TRACKING— L'antenne a réussi à capter un signal correspondant au profil de la mission et le suit activement dans le ciel. Il s'agit de l'état de fonctionnement nominal lors d'un contact.
- ACQUIRING— L'antenne est en train de localiser le signal et de le verrouiller. Le système utilise actuellement un suivi programmatique, un pointage basé sur les données d'éphémérides.
- MASKED— La position prévue du satellite se trouve derrière un masque de suivi automatique, ce qui signifie que l'antenne ne peut pas utiliser le suivi automatique de manière fiable dans cette direction de pointage spécifique. Cela se produit généralement dans les zones de fortes interférences RF, telles que les basses altitudes.

trackingErrorAzimuth

Erreur de suivi sur l'axe azimutal, mesurée en degrés.

trackingErrorElevation

Erreur de suivi sur l'axe d'élévation, mesurée en degrés.

Note

Les valeurs d'erreur de suivi représentent les ajustements par rapport à la piste de programme basée sur les éphémérides qui AWS Ground Station s'applique pendant le suivi automatique afin de maximiser la puissance du signal.

Champs de métadonnées

La télémétrie de suivi inclut les mêmes champs de métadonnées que la télémétrie de pointage : `groundStation`, et `satelliteId` `contactId`

Exemple JSON

```
{
  "telemetryTypeAndVersion": "TRACKING#1.0.0",
  "telemetryType": "TRACKING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
```

```
"satelliteId": "87654321-4321-4321-4321-210987654321",
"contactId": "12345678-1234-1234-1234-123456789012"
},
"data": {
  "sampleTimestamp": "2025-12-08T12:00:00.123Z",
  "trackingStatus": "TRACKING",
  "trackingErrorAzimuth": 0.2,
  "trackingErrorElevation": 0.1
}
}
```

Lecture de données depuis le flux Kinesis Data Streams

Les données de télémétrie sont transmises à votre flux Kinesis Data Streams et peuvent être consommées selon les modèles de consommation de flux standard. Lorsque vous lisez les données de votre flux, tenez compte des points suivants.

Décodage Base64

Les données du flux Kinesis Data Streams sont codées en Base64. Vous devez décoder les données avant de les analyser au format JSON. Pour plus d'informations, consultez la section [Travailler avec Amazon Kinesis Data Streams](#).

Utilisation du Kinesis Data Viewer

Pour accéder rapidement à vos données de télémétrie, la console de streaming Kinesis Data Streams propose une fonctionnalité de visualisation de données. Lorsque vous utilisez cette fonctionnalité :

- La télémétrie peut être transmise à n'importe quel fragment de votre flux.
- La position de départ par défaut est lue à partir des derniers enregistrements de la partition.
- Vous devrez peut-être ajuster la partition sélectionnée et utiliser la position de départ « À l'horodatage » pour afficher les enregistrements reçus.

Utilisation de la bibliothèque cliente Kinesis

La Kinesis Client Library (KCL) gère de nombreuses complexités associées à la consommation de données issues du flux Kinesis Data Streams, notamment la gestion des partitions, les points de contrôle et l'équilibrage de charge. Nous recommandons d'utiliser KCL pour les applications de télémétrie de production relatives à la consommation.

Pour plus d'informations, voir [Développement de consommateurs à l'aide de la bibliothèque cliente Kinesis](#).

Bonnes pratiques en matière de consommation

- Minimisez le temps de latence : utilisez Enhanced Fan-Out pour lire le flux Kinesis Data Streams avec un débit dédié et une latence inférieure à celle d'un sondage. Pour plus d'informations, consultez la section [Développement de consommateurs de fan-out améliorés](#).
- Stream dédié : utilisez un flux Kinesis Data Streams dédié pour AWS Ground Station votre intégration de télémétrie. Le partage d'un flux avec d'autres applications peut entraîner une saturation du débit d'écriture et des échecs de transmission de données télémétriques.
- Capacité à la demande : déployez votre flux Kinesis Data Streams en mode de provisionnement à la demande pour permettre le dimensionnement automatique des partitions en fonction du débit.
- Surveillez le débit : surveillez la régulation de votre flux à l'aide de métriques. CloudWatch Pour plus d'informations, consultez la section [Surveillance d'Amazon Kinesis Data Streams](#).

Versionnement et évolution des schémas

Les schémas de télémétrie sont versionnés pour prendre en charge l'évolution au fil du temps. Le `telemetryVersion` champ de chaque enregistrement indique la version du schéma.

Gestion des modifications de schéma

- De nouveaux types de télémétrie pourraient être introduits à l'avenir.
- Les types de télémétrie existants peuvent recevoir de nouvelles versions avec des modifications importantes.
- Vos applications doivent tolérer les types et versions de télémétrie inconnus.
- Analysez les `telemetryVersion` champs `telemetryTypeAndVersion` `telemetryType`, et pour déterminer comment traiter chaque enregistrement.

Nous vous recommandons de mettre en œuvre une sérialisation des charges utiles adaptée aux versions, capable de gérer plusieurs versions de schéma avec élégance, afin de permettre à vos applications de continuer à fonctionner lorsque de nouvelles versions sont introduites.

Travailler avec des contacts

Vous pouvez saisir des données satellites, identifier l'emplacement des antennes, communiquer et planifier l'heure des antennes pour certains satellites à l'aide de la AWS Ground Station console ou du AWS SDK dans la langue de votre choix. AWS CLI Vous pouvez consulter, annuler et replanifier les réservations de contact jusqu'à 15 minutes avant le début du contact*. Vous pouvez également mettre à jour un contact pour spécifier une dérogation aux éphémérides (y compris les données de suivi azimut/altitude, OEM ou TLE) ou modifier le satellite cible. Pour de plus amples informations, veuillez consulter [Mettre à jour les contacts et la gestion des versions des contacts](#). Vous pouvez également consulter les détails relatifs à votre plan de tarification à la minute réservé si vous utilisez le modèle de tarification à la minute AWS Ground Station réservé.

AWS Ground Station prend en charge la diffusion de données entre régions. Les configurations de point de terminaison de flux de données qui font partie du profil de mission que vous sélectionnez déterminent dans quelle(s) région(s) les données sont diffusées. Pour plus d'informations sur l'utilisation de la livraison de données entre régions, consultez [Utiliser la diffusion de données entre régions](#).

Pour planifier des contacts, vos ressources doivent être configurées. Si vous n'avez pas configuré vos ressources, consultez [Mise en route](#). Lorsqu'il [ReserveContact](#) est appelé, AWS Ground Station prend un instantané du profil de mission et configure les ressources à utiliser tout au long du cycle de vie du contact. Les modifications apportées à ces ressources à l'aide du [UpdateMissionProfile](#) et ne [UpdateConfig](#) APIs seront pas reflétées dans les contacts réservés avant les mises à jour. Si vous souhaitez que les modifications des ressources soient appliquées à un contact déjà planifié, vous devez d'abord annuler le contact en utilisant [CancelContact](#), puis le replanifier en utilisant [ReserveContact](#).

* Les contacts annulés peuvent entraîner des frais s'ils sont annulés trop près du moment du contact. Pour plus d'informations sur les contacts annulés, voir : [Ground Station FAQs](#).

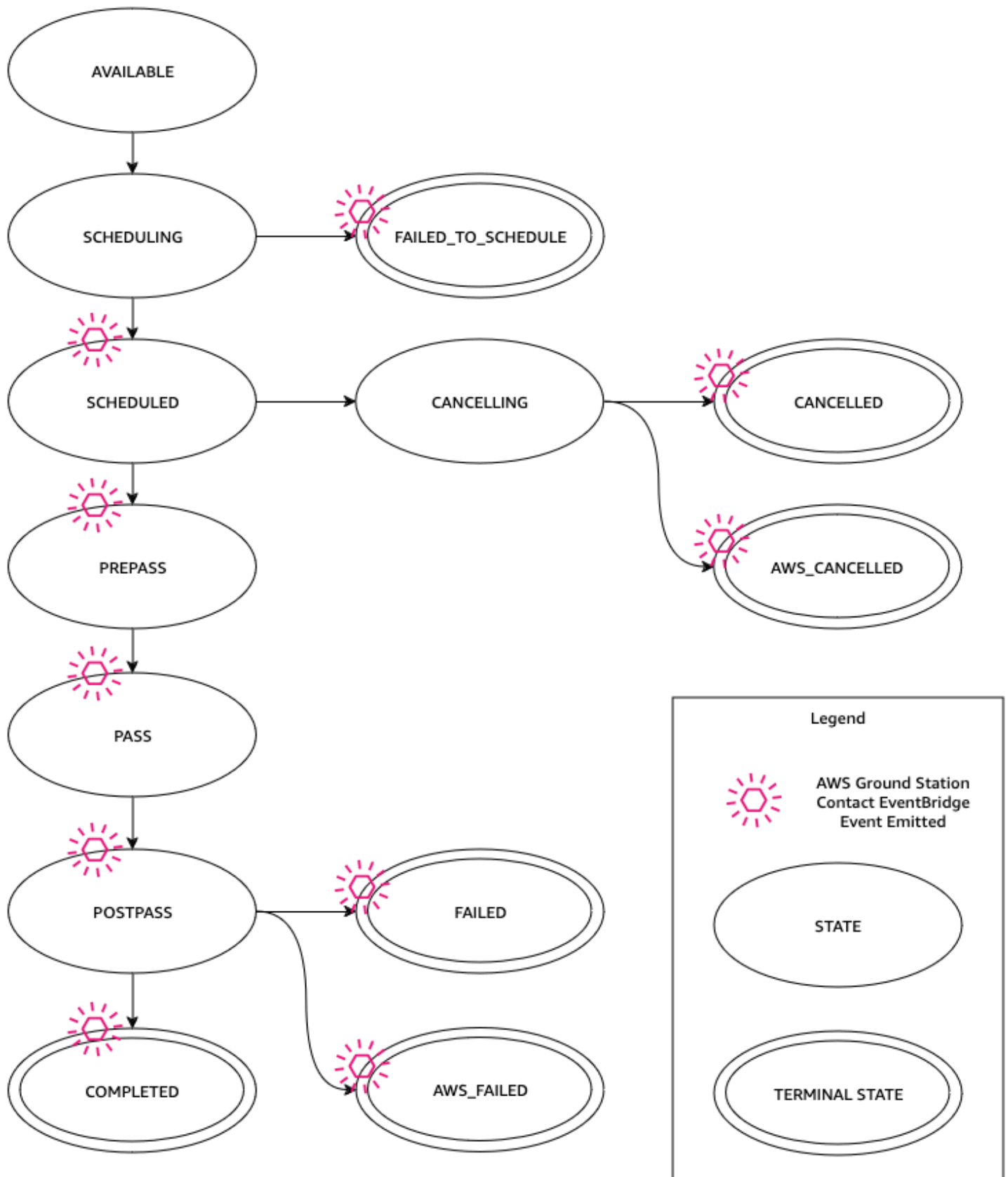
Rubriques

- [Comprendre le cycle de vie des contacts](#)
- [Comprendre la facturation des contacts](#)
- [Mettre à jour les contacts et la gestion des versions des contacts](#)

Comprendre le cycle de vie des contacts

Comprendre le cycle de vie des contacts peut vous aider à automatiser et à résoudre divers problèmes lors de leur utilisation AWS Ground Station. Le schéma suivant montre le cycle de vie des AWS Ground Station contacts ainsi que les événements Event Bridge émis au cours du cycle de vie. Il est important de noter que les états COMPLETED, FAILED, FAILED_TO_SCHEDULE, CANCELLED et AWS_CANCELLED, sont des états terminaux. AWS_FAILED Les contacts ne sortiront pas de l'état terminal. Consultez le [AWS Ground Station statuts des contacts](#) pour plus de détails sur ce que chaque statut indique et s'il est possible d'arrêter ou d'annuler son utilisation.

[CancelContact](#)



AWS Ground Station statuts des contacts

Le statut d'un AWS Ground Station contact donne un aperçu de ce qui arrive à ce contact à un moment donné.

Statuts des contacts

Le tableau suivant décrit les statuts que peut avoir un contact :

Statut	Description	Terminal	Annulable	Arrêtable
DISPONIBLE	Le contact est disponible pour être réservé.	Non	N/A	N/A
PLANIFICATION	Le contact est en cours de planification.	Non	Oui	Non
PLANIFIÉ	Le contact a été planifié avec succès.	Non	Oui	Non
ÉCHEC DU PLANNING	Le contact n'a pas pu être planifié.	Oui	Non	Non
PRÉPASSE	Le contact commence bientôt et les ressources sont en cours de préparation.	Non	Oui	Non
PASSER	Le contact est en cours d'exécution et le satellite est en cours de communication.	Non	Non	Oui
LAISSEZ-PASSER POSTAL	La communication est terminée et les ressources utilisées sont en cours de nettoyage.	Non	Non	Non
TERMINÉ	Le contact s'est terminé sans erreur.	Oui	Non	Non

Statut	Description	Terminal	Annulable	Arrêtable
ÉCHEC	Le contact a échoué en raison d'un problème lié à la configuration de vos ressources.	Oui	Non	Non
AWS_FAILED	Le contact a échoué en raison d'un problème dans le AWS Ground Station service.	Oui	Non	Non
ANNULATION	Le contact est en cours d'annulation.	Non	Non	Non
AWS_CANCELLED	Le contact a été annulé par le AWS Ground Station service. La maintenance de l'antenne ou du site et la dérive des éphémérides sont des exemples de situations où cela peut se produire.	Oui	Non	Non
CANCELLED	Le contact a été annulé par vous.	Oui	Non	Non

Note

Pour plus d'informations sur les implications de facturation des contacts annulés ou interrompus, consultez [Comprendre la facturation des contacts](#).

Conservation des données de contact

AWS Ground Station conserve les données de contact pendant 1 an après la [ReserveContact](#) demande de réservation d'un contact. Après la période d'un an, les données de contact sont supprimées.

Si vous devez conserver les données de contact au-delà d'un an, il est recommandé de les exporter avant l'expiration de la période de conservation. Pour plus d'informations sur la façon d'accéder aux données de contact et de les exporter, consultez :

- [AWS Ground Station API Reference](#)
- [AWS Ground Station Référence des commandes d'interface de ligne de commande](#)

Comprendre la facturation des contacts

Avec AWS Ground Station, vous ne payez que pour le temps d'antenne que vous utilisez. AWS Ground Station les compteurs contactent l'utilisation par minute. Pour chaque contact, le service calcule la durée du contact du début à la fin et l'arrondit à la minute la plus proche. Cette durée mesurée détermine vos frais pour ce contact.

Votre tarif dépend de deux facteurs principaux :

- Bande passante : quantité de bande passante réservée au contact (bande étroite ou large bande)
- Emplacement de la station au sol — Les tarifs varient en fonction de l'emplacement de la station au sol

Définitions de bande passante

AWS Ground Station classe les contacts en deux niveaux de bande passante en fonction de la bande passante instantanée :

- Bande étroite — Tout contact dont la bande passante instantanée est inférieure ou égale à 40 MHz
- Bande large — Tout contact dont la bande passante instantanée est supérieure à 40 MHz

Modes de planification

AWS Ground Station propose deux modes de planification :

- À la demande — Payez pour l'accès à l'antenne sans engagement à long terme
- Réserve — Offre un tarif réduit et une planification améliorée par rapport à la demande, avec un engagement mensuel. Le tarif des minutes réservées est disponible pour les clients qui s'engagent à une utilisation mensuelle pendant une certaine période.

Pour obtenir des informations tarifaires spécifiques à votre compte ou pour en savoir plus sur le mode de planification réservé, contactez votre représentant AWS.

CancelContact

L'utilisation de l' [CancelContact](#) API varie en fonction de l'état du contact lorsque vous l'appellez :

- Avant le début du contact : annule complètement le contact
- Après le début du contact et avant la fin du contact : arrête le contact en cours

Lorsque vous annulez un contact, la facturation dépend de votre mode de planification et de la date à laquelle vous annulez. Pour plus d'informations, contactez votre représentant AWS.

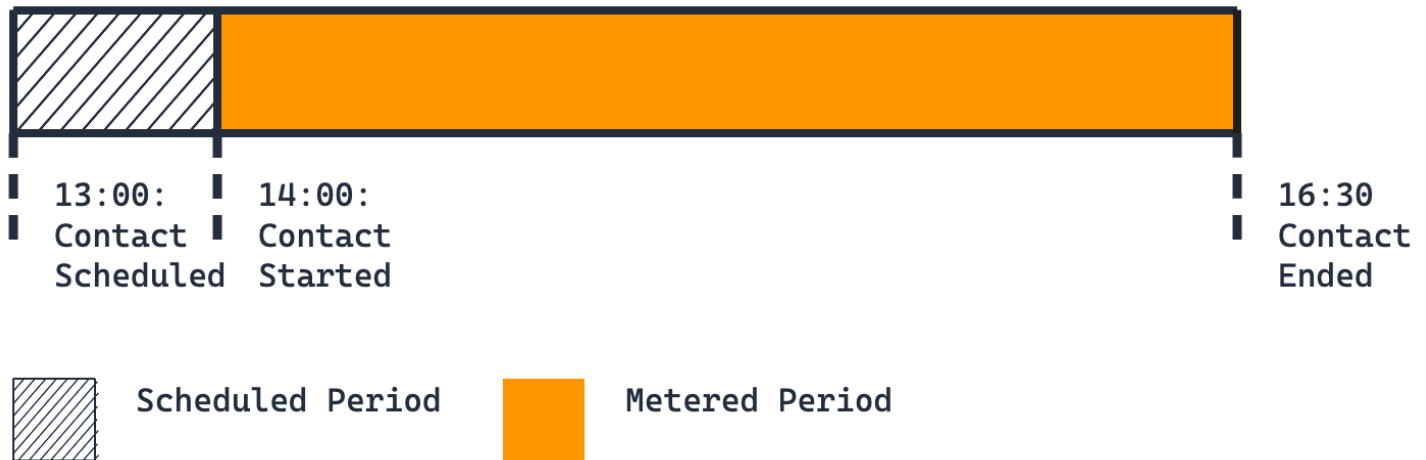
Lorsque vous arrêtez un contact, vous êtes facturé pour la partie du contact exécutée et pour le temps restant non couvert par des contacts dupliqués. Dans ce contexte, un contact dupliqué a été :

- Programmé sur la même station au sol que le contact arrêté d'origine
- Planifié avec le même identifiant de compte AWS que le contact arrêté d'origine
- Réservé après l'émission de la commande pour arrêter le contact d'origine

Les scénarios suivants montrent comment cette mesure fonctionne dans la pratique.

Scénario 1 : contact unique

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30.



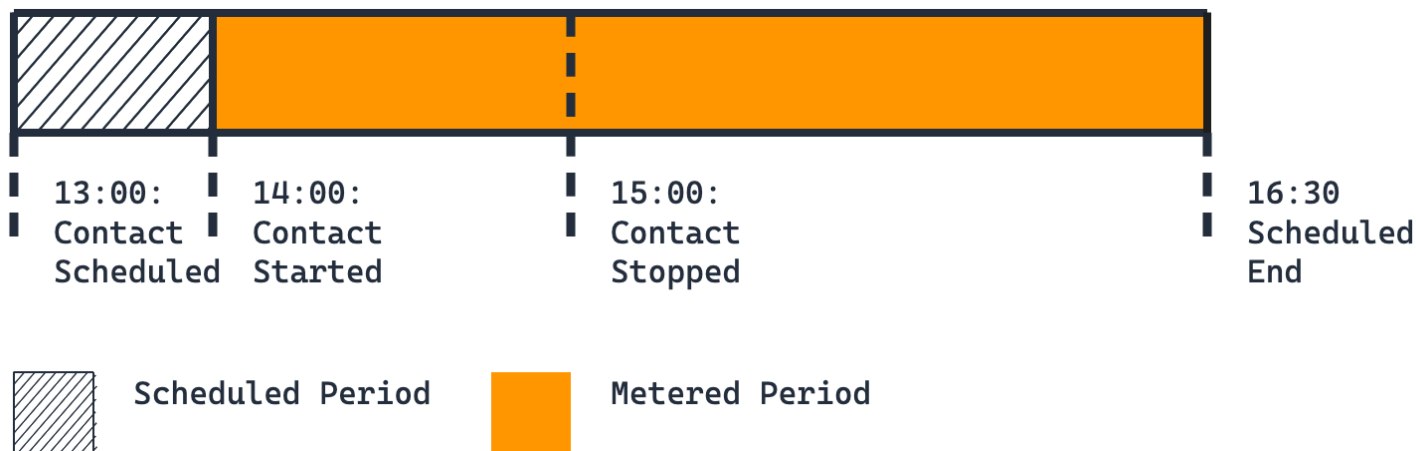
Répartition de la facturation :

- Premier contact : 150 minutes (durée complète)

150 minutes vous sont facturées. Il s'agit du scénario de base dans lequel un contact se termine comme prévu sans interruption ni annulation.

Scénario 2 : contact à arrêt unique

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre contact.



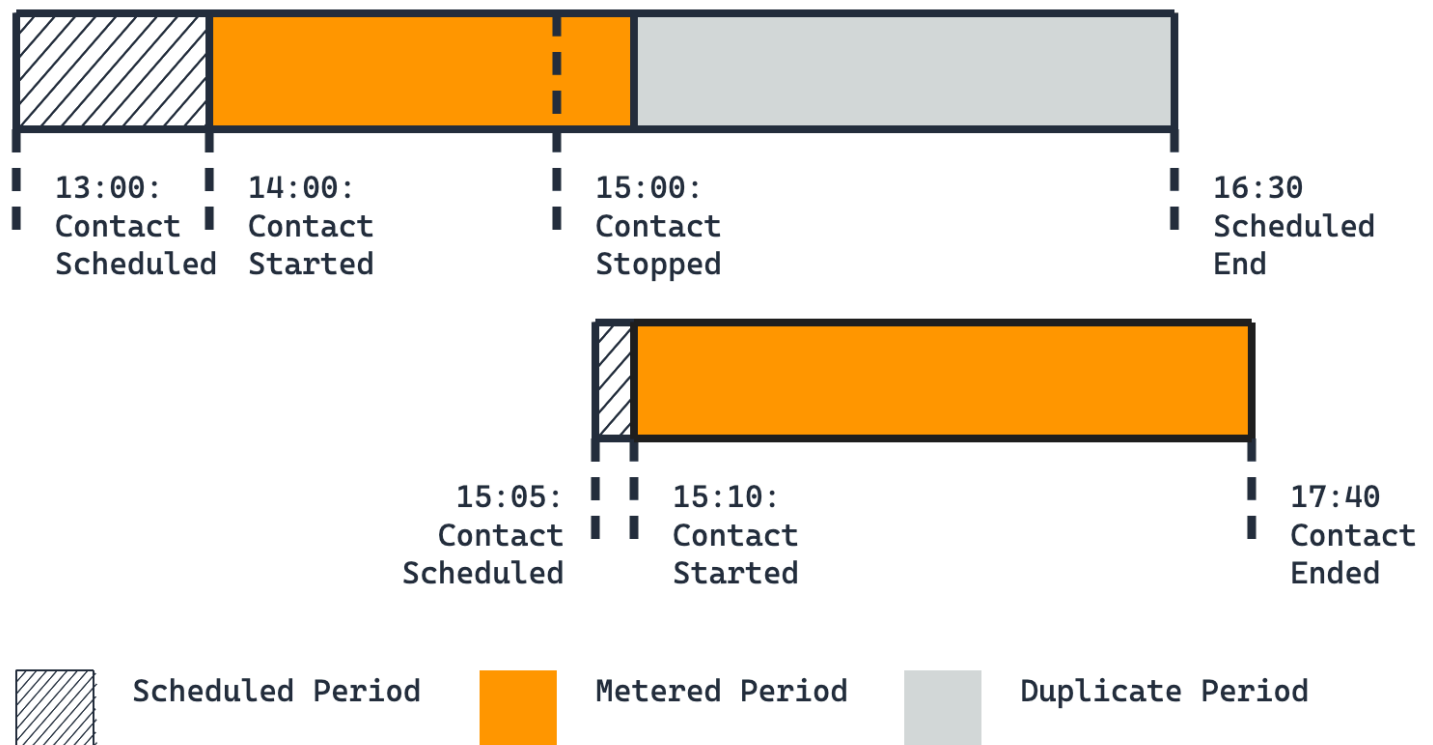
Répartition de la facturation :

- Premier contact : 150 minutes (durée originale complète)

Les 150 minutes vous sont facturées car vous avez arrêté le contact, mais vous n'avez prévu aucun double contact pour couvrir le temps restant (15 h 00-16 h 30). Lorsque vous arrêtez un contact sans planifier de doublons, vous restez responsable de l'intégralité de la durée initialement prévue.

Scénario 3 : double exemplaire

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Après avoir appelé `CancelContact`, vous programmez un autre contact sur la même Ground Station à partir de 15 h 10 pendant 150 minutes.



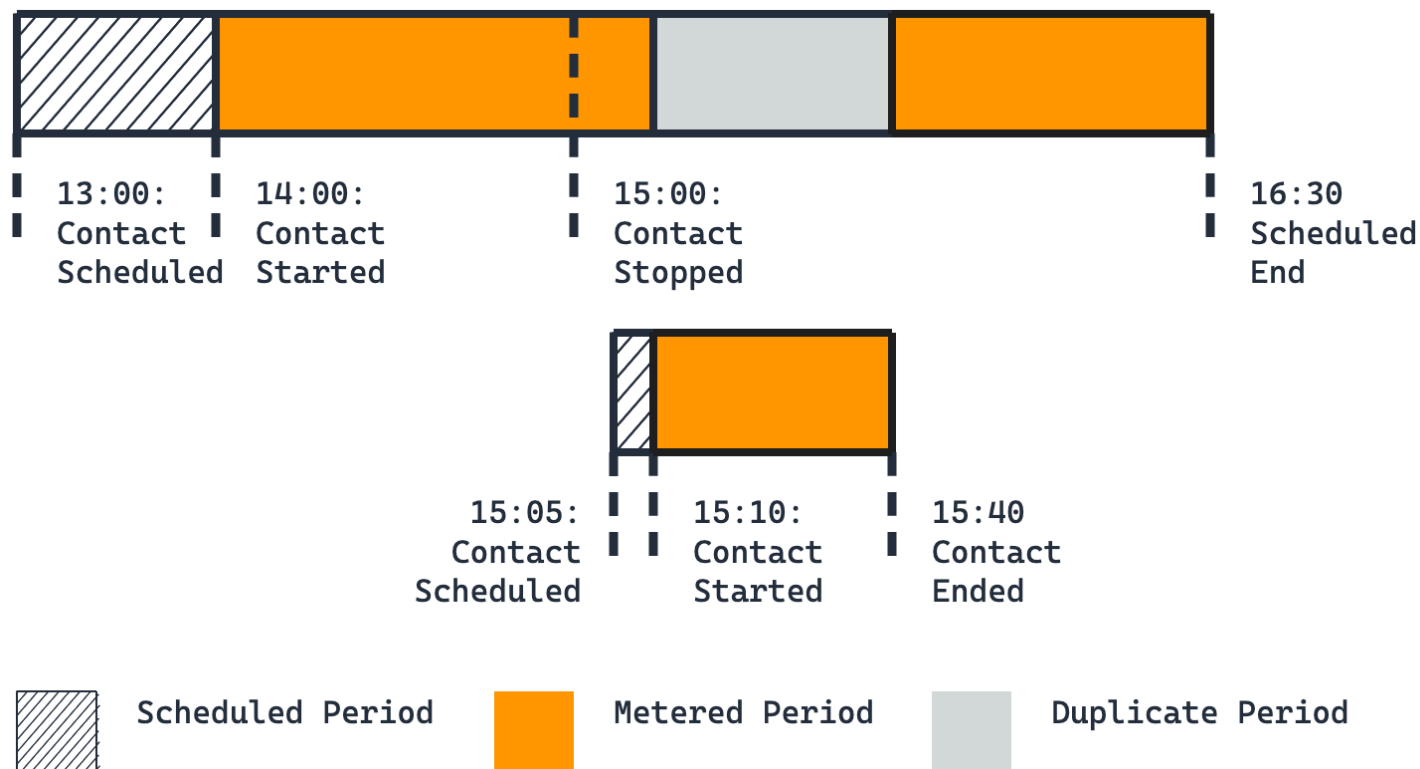
Répartition de la facturation :

- Premier contact : 70 minutes (60 minutes exécutées plus 10 minutes d'arrêt avant le début du deuxième contact)
- Deuxième contact : 150 minutes (durée complète)

Le deuxième contact est un doublon car vous l'avez planifié après avoir arrêté le premier contact. Le duplicata couvre le temps restant entre 15 h 10 et 16 h 30. Vous n'êtes donc facturé que pour le temps pendant lequel le premier contact a réellement été établi, plus les 10 minutes entre l'arrêt et le redémarrage.

Scénario 4 : double court

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Après avoir appelé `CancelContact`, vous programmez un contact de 30 minutes sur la même Ground Station à partir de 15h10.



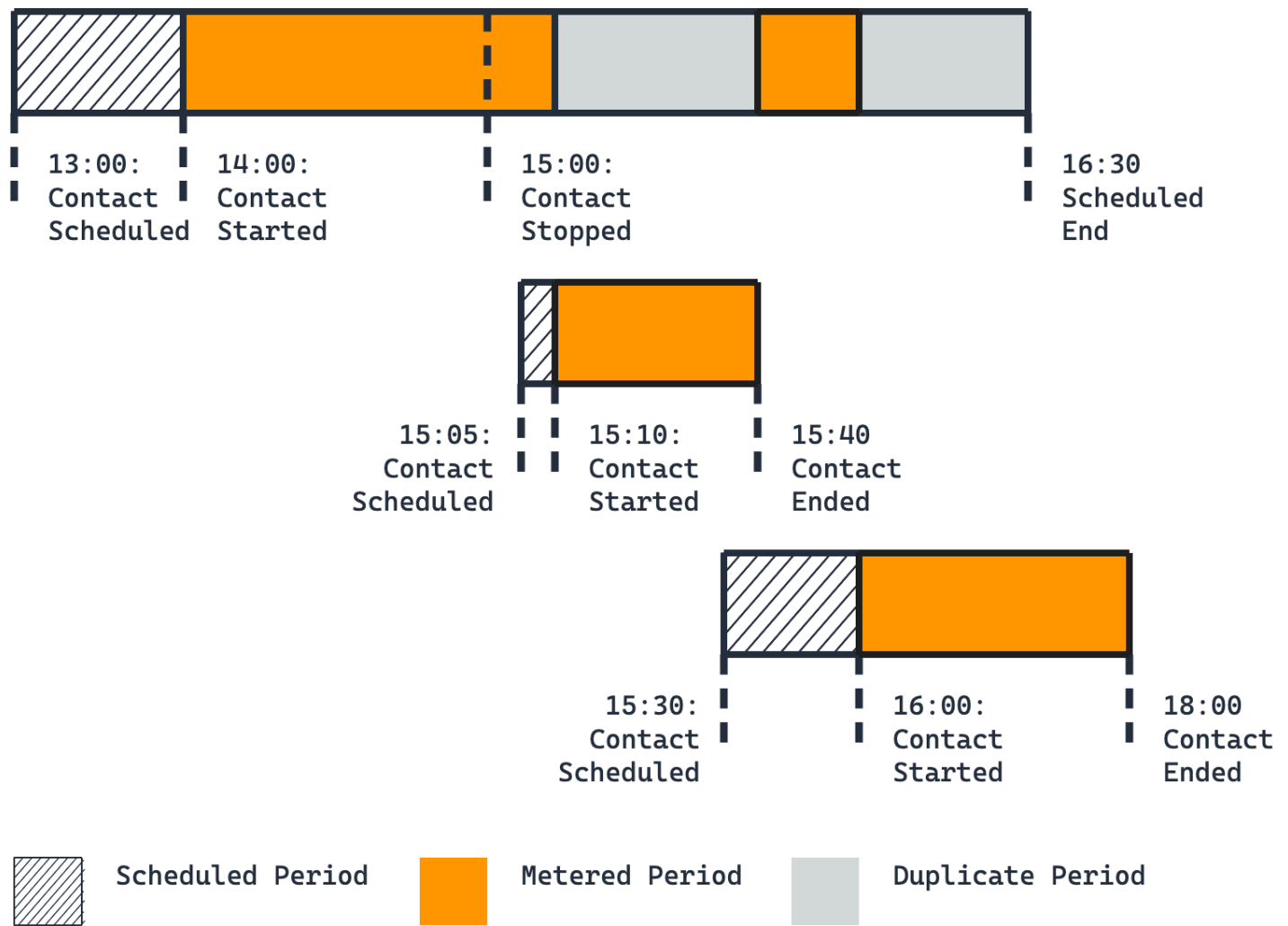
Répartition de la facturation :

- Premier contact : 120 minutes (60 minutes d'exécution + 10 minutes d'arrêt avant le début du deuxième contact, + 50 minutes de temps restant non couvert par le duplicata)
- Deuxième contact : 30 minutes (durée complète)

Le double contact ne couvre que 30 minutes (15:10-15:40) des 90 minutes restantes après l'arrêt du premier contact. Vous êtes facturé à la fois pour les 10 minutes avant le début du doublon et pour les 50 minutes de temps passé à découvert après la fin du doublon (15 h 40 à 16 h 30).

Scénario 5 : plusieurs doublons

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Après avoir appelé `CancelContact`, vous programmez un contact de 30 minutes sur la même Ground Station à partir de 15h10. Plus tard, à 15 h 30, vous programmez un autre contact à partir de 16 h pendant 120 minutes.



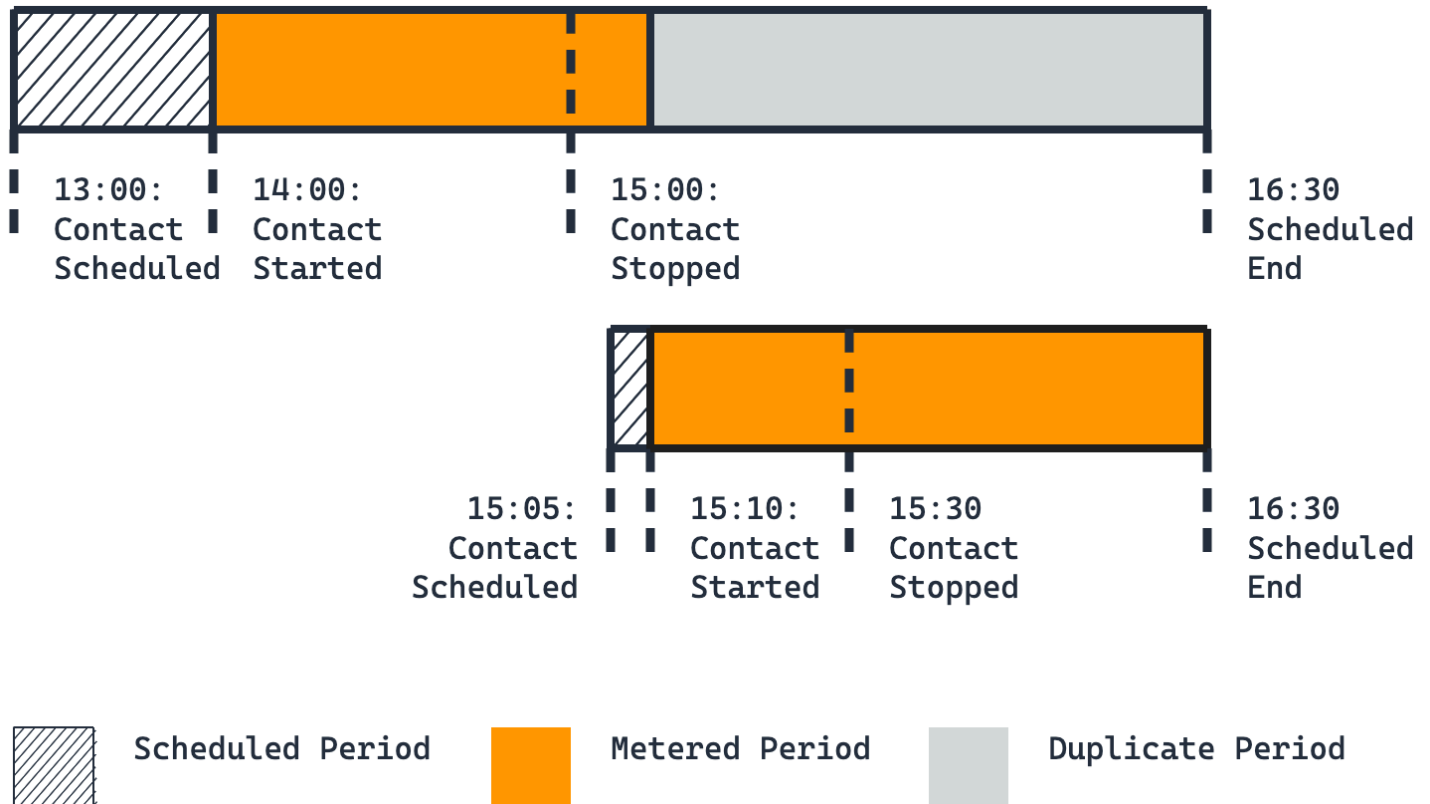
Répartition de la facturation :

- Premier contact : 90 minutes (60 minutes exécutées + 10 minutes d'arrêt avant le début du deuxième contact + 20 minutes d'arrêt entre le deuxième et le troisième contact)
- Deuxième contact : 30 minutes (durée complète)
- Troisième contact : 120 minutes (durée complète)

Le deuxième et le troisième contact sont considérés comme des doublons car vous les avez programmés après avoir arrêté le premier contact. Cependant, les écarts entre les contacts vous sont toujours facturés : 10 minutes entre le premier arrêt (15h00) et le deuxième départ (15h10), et 20 minutes entre le deuxième bout (15h40) et le troisième départ (16h00).

Scénario 6 : arrêts multiples

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Après avoir appelé `CancelContact`, vous planifiez un contact de 80 minutes sur Ground Station Anytown 1, qui commence à 15 h 10 et se termine à 16 h 30. À 15 h 30, vous appelez à nouveau l' `CancelContact` API pour mettre fin à votre double contact.



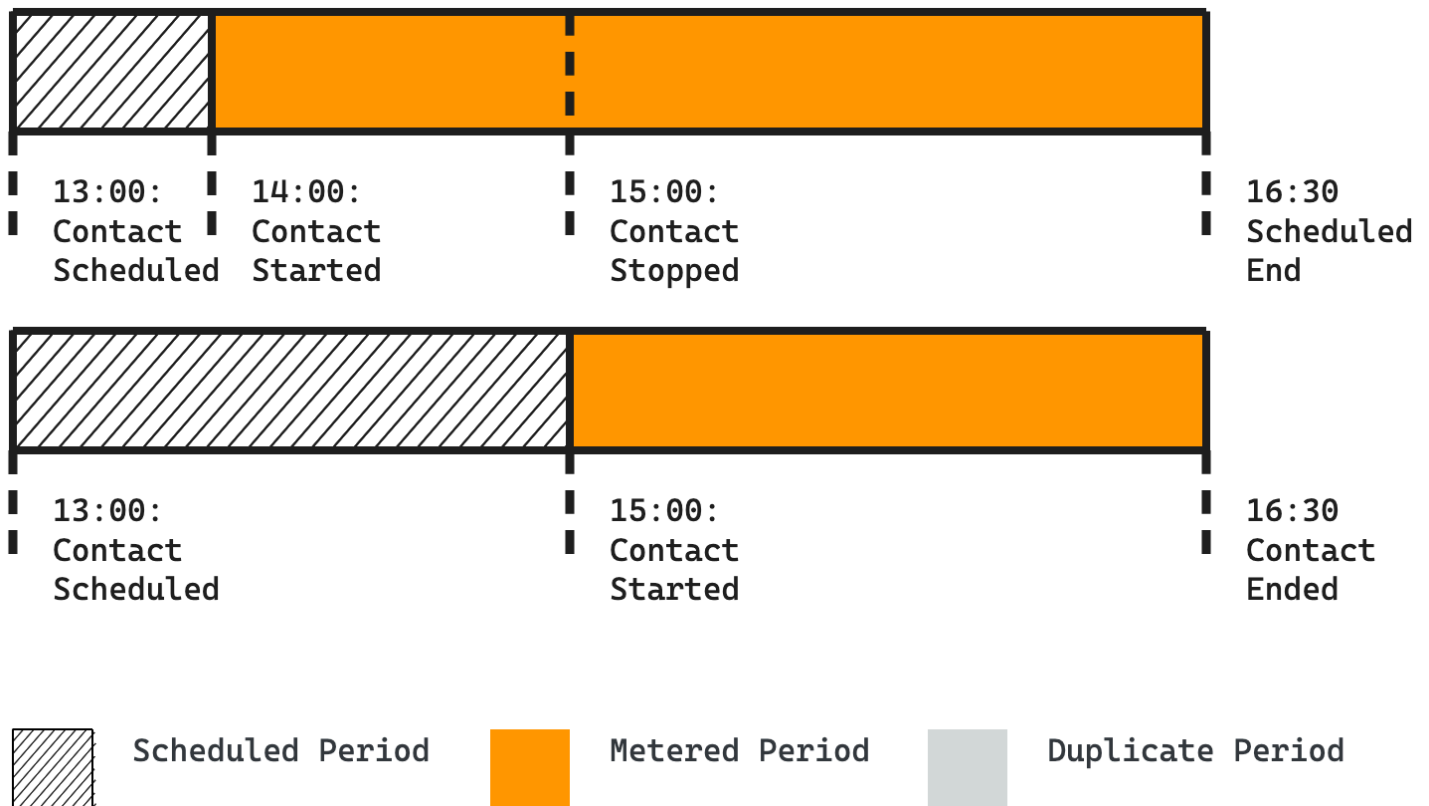
Répartition de la facturation :

- Premier contact : 70 minutes (60 minutes exécutées plus 10 minutes d'arrêt avant le début du deuxième contact)
- Deuxième contact : 80 minutes (durée originale complète)

Le deuxième contact est facturé pour sa durée complète de 80 minutes car vous l'avez arrêté à 15 h 30, laissant 60 minutes du temps initialement prévu (15 h 30 à 16 h 30) vides. À moins que vous ne programmiez un autre double contact pour couvrir le temps restant, vous êtes responsable de toute la durée de tout contact interrompu.

Scénario 7 : station au sol à antennes multiples sans doublon

À 13h00, vous programmez deux contacts sur Ground Station Anytown 1. Le premier est un contact de 150 minutes commençant à 14h00 et se terminant à 16h30. Le second est un contact de 90 minutes commençant à 15 h 00 et se terminant à 16 h 30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Ground Station Anytown 1 est une station au sol à antennes multiples, qui permet aux deux contacts de fonctionner simultanément.



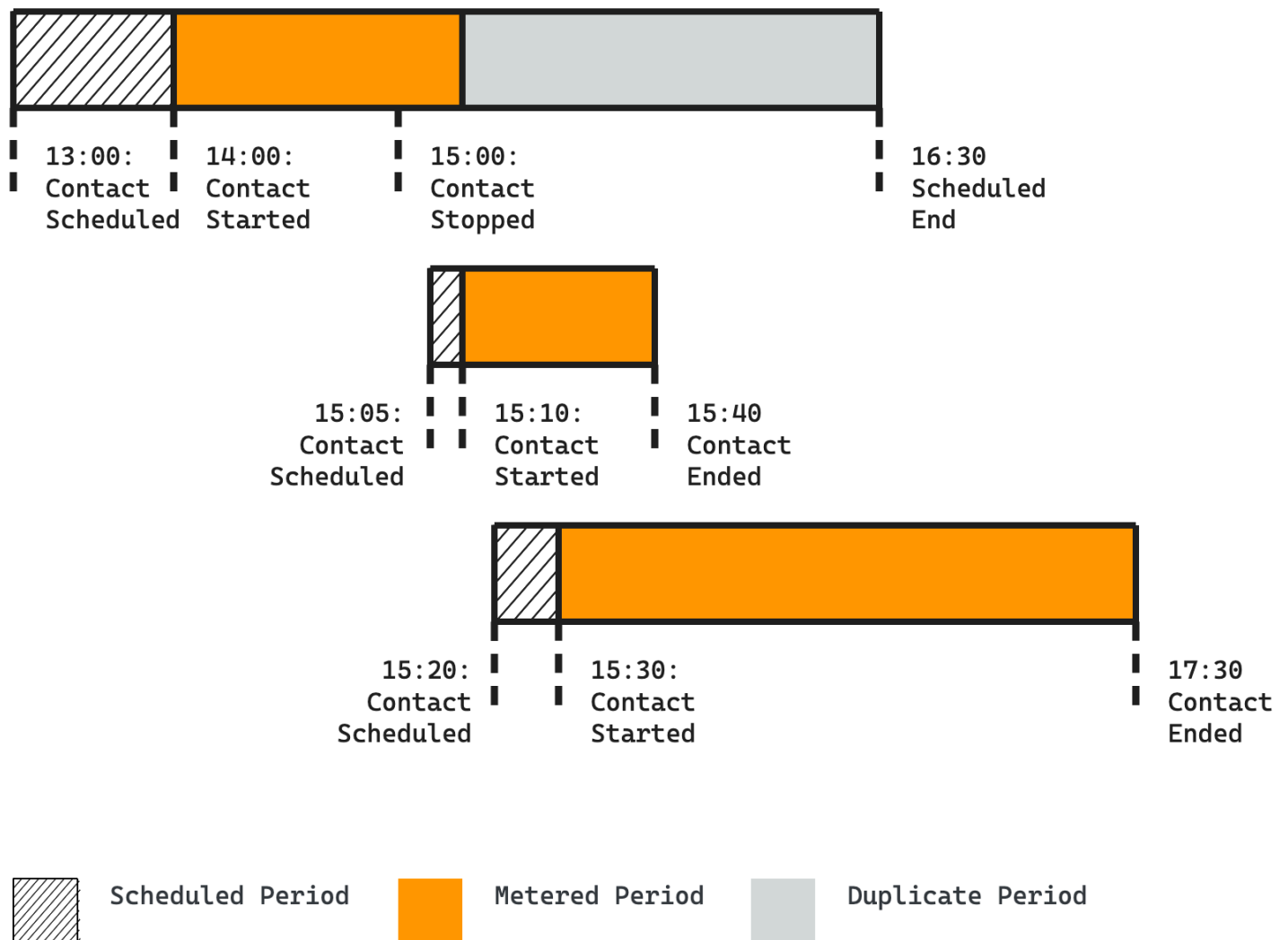
Répartition de la facturation :

- Premier contact : 150 minutes (durée originale complète)
- Deuxième contact : 90 minutes (durée complète)

Bien que le second contact chevauche la partie arrêtée du premier contact, il n'est pas considéré comme un doublon. Le deuxième contact ne répond pas au premier critère pour les doublons : il était prévu à 13h00, avant que vous n'arrêtiez le premier contact à 15h00. Comme il ne s'agit pas d'un doublon, vous êtes facturé pour la durée initiale complète du premier contact, quelle que soit la date à laquelle vous l'avez arrêté.

Scénario 8 : station au sol à antennes multiples avec contacts dupliqués

Vous planifiez un contact de 150 minutes sur Ground Station Anytown 1, qui débutera à 14h00 et se terminera à 16h30. À 15h00, vous appelez l' `CancelContact` API pour arrêter votre premier contact. Après avoir appelé `CancelContact`, vous planifiez un contact de 30 minutes sur Ground Station Anytown 1 à partir de 15h10 et se terminant à 15h40. Plus tard, vous planifiez un autre contact de 90 minutes sur Ground Station Anytown 1 à partir de 15h30 et se terminant à 17h00. Ground Station Anytown 1 est une station au sol à antennes multiples, qui permet aux deux contacts dupliqués de fonctionner simultanément avec des durées de chevauchement.



Répartition de la facturation :

- Premier contact : 70 minutes (60 minutes exécutées plus 10 minutes d'arrêt avant le début du deuxième contact)

- Deuxième contact : 30 minutes (durée complète)
- Troisième contact : 90 minutes (durée complète)

Le deuxième et le troisième contact sont considérés comme des doublons car vous les avez programmés après avoir arrêté le premier contact. L'écart de 10 minutes entre l'arrêt du premier contact (15h00) et le début du deuxième contact (15h10) représente le temps d'arrêt qui vous est facturé par rapport au contact initial.

Mettre à jour les contacts et la gestion des versions des contacts

AWS Ground Station prend en charge la mise à jour des contacts SCHEDULED dotés d'un [statut de PASS contact PREPASS](#), ou. Vous pouvez utiliser l'[UpdateContact](#)API pour spécifier une dérogation aux éphémérides pour un contact (y compris les données de suivi azimut/élévation, OEM ou TLE) sans l'annuler ni le replanifier. Cela est utile pour les opérations de satellites géosynchrones (GEO) où vous devez rediriger une antenne vers un autre satellite lors d'un contact, ou pour le lancement et les premières opérations (LEOPs) où des ajustements de pointage sont nécessaires.

Chaque fois que vous réservez ou mettez à jour un contact, une nouvelle version du contact est AWS Ground Station créée. Les versions de contact fournissent un historique des modifications apportées à un contact et vous permettent de suivre l'état de chaque mise à jour.

Comment fonctionne la gestion des versions des contacts

Lorsque vous appelez [ReserveContact](#), AWS Ground Station crée la première version du contact (version 1) et la renvoie `versionId` dans la réponse. Chaque appel suivant [UpdateContact](#) crée une nouvelle version avec un numéro de version incrémenté.

L'[DescribeContact](#)API renvoie la [version du ACTIVE contact](#) actuel, y compris les informations de version dans le `version` champ de la réponse. L'[ListContacts](#)API inclut également des informations de version pour chaque contact.

Pour consulter une version spécifique d'un contact, utilisez l'[DescribeContactVersion](#)API. Pour répertorier toutes les versions d'un contact, utilisez l'[ListContactVersions](#)API.

Mettre à jour un contact

Vous pouvez appeler [UpdateContact](#) lorsqu'un contact est dans l'état SCHEDULED, PREPASS ou PASS. L'API accepte les paramètres suivants :

- `contactId` — Identifiant du contact à mettre à jour.
- `ClientToken` — Un jeton d'idempotence qui garantit que la demande n'est traitée qu'une seule fois. Si vous réessayez une demande avec le même jeton client, AWS Ground Station renvoie la réponse d'origine sans effectuer à nouveau la mise à jour. Beaucoup génèrent AWS SDKs automatiquement un jeton client pour vous s'il n'en fournit pas un.
- `TrackingOverrides` : nouvelle configuration de suivi pour le contact. Cela inclut les paramètres de piste du programme (azimut/élévation, TLE ou éphémérides OEM).
- `SatelliteArn` — L'ARN du satellite pour le contact. Lorsque vous modifiez le satellite cible ainsi que les paramètres de suivi du programme, indiquez l'ARN du nouveau satellite. Seuls les clients approuvés pour les angles de pointage azimut/altitude peuvent définir cette valeur sur null. Tous les autres clients doivent inclure l'ARN satellite du contact.

Important

L'`UpdateContactAPI` applique tous les paramètres de la demande. Tout paramètre omis ou défini explicitement sur null est traité comme une demande visant à effacer cette valeur, et non à la laisser inchangée. Par exemple, si vous fournissez `trackingOverrides` mais omettez `satelliteArn`, l'ARN du satellite est effacé. Assurez-vous d'inclure toutes les valeurs souhaitées dans chaque demande de mise à jour.

Vous pouvez changer de satellite cible lors d'un contact en fournissant un nouveau `satelliteArn` ainsi que le satellite correspondant `trackingOverrides`. Le nouveau satellite doit être visible depuis la station au sol pendant toute la durée du contact, car l'heure de début et de fin du contact ne change pas avec cette API. Le nouveau satellite doit également être embarqué à bord de la station au sol et disposer de la licence requise par le profil de mission. Le profil de mission du contact ne pouvant pas être modifié, le changement de satellite n'est applicable que lorsque les deux satellites utilisent le même profil de mission.

Important

L'`UpdateContactAPI` ne permet pas de modifier l'heure de début, l'heure de fin ou le profil de mission d'un contact. Pour modifier ces valeurs, annulez le contact et réservez-en un nouveau. L'`UpdateContactAPI` est conçue pour redéfinir la configuration de pointage de

l'antenne, par exemple pour passer d'un satellite à un autre ou mettre à jour les données d'éphémérides.

Important

L'UpdateContactAPI ne prend pas en charge les contacts dont le profil de mission utilise des [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#) configurations. Pour modifier la configuration de ces contacts, annulez-les et réservez-en un nouveau.

L'UpdateContactAPI renvoie le `contactId` et le `nouveautionId`. La mise à jour est traitée de manière asynchrone. Permet [DescribeContactVersion](#) de vérifier l'état de la mise à jour. Certains AWS SDKs AWS Command Line Interface proposent un `ContactUpdated` serveur qui interroge jusqu'à ce que la version atteigne le statut `ACTIVE` ou `FAILED_TO_UPDATE`.

Note

Une seule mise à jour peut être en cours à la fois. Si la dernière version du contact est en état `UPDATING`, l'API renvoie un `ConflictException`. Attendez que la mise à jour en cours atteigne le statut `ACTIVE` ou `FAILED_TO_UPDATE` avant de soumettre une autre mise à jour.

Statuts des versions de contact

Chaque version de contact possède l'un des statuts suivants :

Statut	Description
MISE À JOUR	La version est appliquée au contact. La mise à jour a été soumise et est en cours de traitement par AWS Ground Station.
ACTIF	La version est la configuration actuellement active pour le contact. La station au sol utilise les paramètres de cette version.

Statut	Description
REPLACÉE	La version était auparavant active mais a été remplacée par une version plus récente.
ÉCHEC DE LA MISE À JOUR	La mise à jour n'a pas pu être appliquée. Le contact revient à la version précédemment active. Vérifiez les <code>failureMessage</code> champs <code>failureCodes</code> et pour plus de détails.

Exemples de code

Les exemples suivants montrent comment utiliser le versionnement des contacts APIs avec le AWS SDK pour Python (Boto3).

Exemple : mise à jour d'un contact

L'exemple suivant met à jour un contact avec de nouvelles dérogations de suivi et attend que la mise à jour soit terminée à l'aide du serveur `ContactUpdated` Boto3.

```
import boto3
import uuid

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

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

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

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

update_response = ground_station_client.update_contact(
    contactId=contact_id,
```

```

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

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

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

from botocore.exceptions import WaiterError

waiter = ground_station_client.get_waiter("contact_updated")

try:
    waiter.wait(
        contactId=contact_id,
        versionId=version_id,
        WaiterConfig={
            "Delay": 5,
            "MaxAttempts": 180,
        },
    )
    print(f"Contact updated successfully. Version {version_id} is now active.")
except WaiterError as e:
    # WaiterError is raised when the version reaches FAILED_TO_UPDATE
    # or when MaxAttempts is exceeded. Retrieve the current version to inspect.
    version_response = ground_station_client.describe_contact_version(
        contactId=contact_id,
        versionId=version_id,
    )
    version_status = version_response["version"]["status"]
    if version_status == "FAILED_TO_UPDATE":

```

```

    failure_codes = version_response["version"].get("failureCodes", [])
    failure_message = version_response["version"].get("failureMessage", "")
    print(f"Update failed. Codes: {failure_codes}, Message: {failure_message}")
else:
    print(f"Waiter timed out. Current version status: {version_status}. Error:
    {e}")

```

Exemple : décrire une version de contact

L'exemple suivant permet de récupérer les détails d'une version de contact spécifique, notamment son statut, sa configuration et toute information relative à une défaillance.

```

import boto3

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

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

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

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

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

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

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

# Display contact details for this version

```

```

print(f"\nContact ID: {response['contactId']}")
print(f"Contact Status: {response['contactStatus']}")
print(f"Ground Station: {response['groundStation']}")
print(f"Start Time: {response['startTime']}")
print(f"End Time: {response['endTime']}")

if response.get("satelliteArn"):
    print(f"Satellite ARN: {response['satelliteArn']}")

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

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

```

Exemple : Répertoire les versions des contacts

L'exemple suivant répertorie toutes les versions d'un contact pour afficher l'historique complet des modifications, en utilisant la pagination pour gérer de grands ensembles de résultats.

```

import boto3

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

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

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

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

```

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

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

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

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

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

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

Considérations

- Les contacts créés avant l'introduction de la fonctionnalité de gestion des versions des contacts ne disposent pas d'informations de version. Le fait [DescribeContactVersionListContactVersions](#)d'appeler ou de contacter ces contacts renvoie un `ResourceNotFoundException`.
- Lorsque vous mettez à jour les dérogations de suivi d'un contact en cours, il y a une brève période de transition pendant laquelle l'antenne s'adapte à la nouvelle configuration de pointage. Pendant ce temps, la réception ou la transmission du signal peuvent être interrompues.
- Les versions de contact ne peuvent pas être supprimées. Permet [ListContactVersions](#)d'afficher l'historique complet des modifications apportées à un contact.
- L'`UpdateContactAPI` ne peut être appelée qu'à partir de la région de planification du contact.

Utilisez la fonction de jumelage AWS Ground Station numérique

La fonction de jumelage numérique pour vous AWS Ground Station fournit un environnement dans lequel vous pouvez tester et intégrer votre logiciel de gestion de mission satellite et de commande et de contrôle. La fonction de jumelage numérique vous permet de tester la planification, la vérification des configurations et la gestion appropriée des erreurs sans utiliser la capacité de l'antenne de production. Le test de votre AWS Ground Station intégration à l'aide de la fonction de jumelage numérique vous permet d'avoir une confiance accrue dans la capacité de votre système à gérer les opérations de vos satellites de manière fluide. Il vous permet également de réaliser des tests AWS Ground Station APIs sans utiliser la capacité de production ni nécessiter de licence de spectre.

Pour commencer [satellite embarqué](#), suivez en demandant à être intégré à la fonction de jumeau numérique. Une fois que votre satellite est intégré à la fonction de jumelage numérique, vous pouvez planifier des contacts avec des stations terrestres à double signal numérique. La liste des stations au sol auxquelles vous avez accès peut être récupérée via la [ListGroundStations](#) réponse du SDK AWS. Les stations terrestres à jumelles numériques sont des copies exactes des stations au sol répertoriées [AWS Ground Station Succursales](#) avec un préfixe modificateur du nom de la station au sol « Digital Twin ». Cela inclut leurs capacités d'antenne et leurs métadonnées, y compris, mais sans s'y limiter, le masque du site et les coordonnées GPS réelles. À l'heure actuelle, la fonction de jumelage numérique ne prend pas en charge la transmission de données telle que décrite dans [Travailler avec des flux de données](#).

Une fois intégrée, la fonctionnalité de jumelage numérique émet les mêmes EventBridge événements Amazon et les mêmes réponses API que le service de production, comme décrit dans. [Automatisez AWS Ground Station avec des événements](#) Ces événements vous permettront d'affiner vos configurations et vos groupes de points de terminaison de flux de données.

AWS Ground Station Antennes dédiées

AWS Ground Station Les antennes dédiées sont des systèmes d'antennes conçus sur mesure qui AWS gère en votre nom. Contrairement aux AWS Ground Station antennes publiques où vous partagez le temps d'antenne avec d'autres clients, une antenne dédiée vous donne un accès dédié à une antenne construite selon vos spécifications. Vous pouvez avoir une ou plusieurs antennes dédiées. Une antenne dédiée est connectée au AWS réseau mondial, et vous interagissez avec celui-ci en utilisant les mêmes AWS Ground Station APIs flux de travail que ceux que vous utilisez avec les antennes publiques, en plus d'une meilleure visibilité sur l'utilisation des antennes.

Qu'est-ce qu'une antenne dédiée

Une antenne dédiée est un système d'antenne physique conçu sur mesure pour votre organisation et entièrement géré par AWS. Vous pouvez avoir une ou plusieurs antennes dédiées. Une antenne dédiée se distingue des AWS Ground Station antennes publiques des manières suivantes :

- Construit sur mesure — Une antenne dédiée est construite selon vos spécifications. Les capacités d'une antenne dédiée ne se limitent pas aux capacités des antennes publiques décrites dans [AWS Ground Station Fonctionnalités du site](#).
- Emplacement flexible — Les antennes dédiées ne sont pas limitées aux emplacements AWS Ground Station d'antennes existants. Une antenne dédiée peut être connectée à n'importe quelle AWS région existante, y compris les régions où elle n' AWS Ground Station est pas disponible actuellement. Vous pouvez travailler avec vous AWS pour déterminer l'emplacement et la région qui répondent à vos besoins.
- Accès dédié — Vous disposez d'un accès dédié à l'antenne plutôt que de partager le temps d'antenne avec d'autres AWS Ground Station clients sur une base par contact.

Toutes les AWS Ground Station antennes, y compris les antennes dédiées, sont entièrement gérées par AWS. Cela inclut la maintenance et la connectivité au AWS réseau mondial. Vous interagissez avec une antenne dédiée en utilisant les mêmes flux AWS Ground Station APIs de travail que ceux que vous utilisez avec les antennes publiques. Vous planifiez les contacts, configurez les profils de mission et transmettez les données de la même manière.

Une antenne dédiée peut être partagée par plusieurs AWS comptes. Le client titulaire du contrat d'antenne dédiée choisit les comptes à intégrer. Chaque compte intégré peut programmer des

contacts sur l'antenne de manière indépendante et dispose d'une visibilité sur les réservations de tous les comptes partageant l'antenne.

Pour en savoir plus sur les antennes dédiées ou pour commencer, contactez AWS Support via le [AWS Support Center Console](#).

Visibilité améliorée des réservations

Lorsque vous utilisez l'[ListGroundStationReservations](#) API sur votre antenne dédiée, des informations supplémentaires qui ne sont pas disponibles sur les antennes publiques s'affichent. Le tableau suivant résume les différences comportementales de l'[ListGroundStationReservations](#) API pour les antennes dédiées par rapport aux AWS Ground Station antennes publiques.

Comportement	Antenne dédiée	Antenne publique
Vos propres contacts	Visible avec tous les détails, y compris <code>contactId</code>	Visible avec tous les détails, y compris <code>contactId</code>
Contacts des autres comptes	Visible uniquement avec des plages horaires, sans <code>contactId</code>	Non visible
Fenêtres de maintenance	Visible avec <code>PLANNED</code> ou <code>UNPLANNED</code> <code>maintenanceType</code>	Non visible

Les fenêtres de maintenance représentent les périodes pendant lesquelles l'antenne n'est pas disponible pour les communications par satellite. Le `maintenanceType` champ indique si la maintenance était `PLANNED` ou `UNPLANNED`. Lorsqu'une maintenance non planifiée est planifiée, les contacts qui se chevauchent avec la fenêtre de maintenance peuvent être annulés par AWS Ground Station.

Lorsque vous consultez les contacts d'autres AWS comptes qui partagent votre antenne dédiée, la réservation inclut le créneau horaire et les informations sur l'antenne, mais elles ne `contactId` sont pas incluses.

⚠ Important

La visibilité améliorée des réservations s'applique uniquement à votre antenne dédiée. Lorsque vous utilisez des AWS Ground Station antennes publiques, vous bénéficiez de la même visibilité que n'importe quel autre client. Les fenêtres de maintenance ou les réservations provenant d'autres comptes ne s'affichent pas sur les antennes publiques.

Pour plus d'informations sur les réservations d'annonces, consultez [Afficher les réservations de stations au sol](#).

Ressources connexes

- [Afficher les réservations de stations au sol](#)
- [AWS Ground Station Succursales](#)
- [AWS Ground Station Fonctionnalités du site](#)
- [ListGroundStationReservations](#) dans la Référence d'API AWS Ground Station

Comprenez la surveillance avec AWS Ground Station

La surveillance est un enjeu important pour assurer la fiabilité, la disponibilité et les performances de AWS Ground Station. AWS fournit les outils de surveillance suivants pour surveiller AWS Ground Station, signaler tout problème et prendre des mesures automatiques le cas échéant.

- Amazon EventBridge Events fournit un flux en temps quasi réel d'événements système décrivant les modifications apportées aux AWS ressources. EventBridge Les événements permettent une informatique automatisée axée sur les événements, car vous pouvez rédiger des règles qui surveillent certains événements et déclenchent des actions automatisées dans d'autres AWS services lorsque ces événements se produisent. Pour plus d'informations sur les EventBridge événements, consultez le [guide de l'utilisateur Amazon EventBridge Events](#).
- AWS CloudTrail capture les appels d'API et les événements associés effectués par ou pour le compte de votre AWS compte et envoie les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes appelés AWS, l'adresse IP source à partir de laquelle les appels ont été effectués et la date des appels. Pour plus d'informations AWS CloudTrail, consultez le [guide de AWS CloudTrail l'utilisateur](#).
- Amazon CloudWatch Metrics capture les statistiques de vos contacts planifiés lors de l'utilisation AWS Ground Station. CloudWatch Les métriques vous permettent d'analyser les données en fonction de votre canal, de votre polarisation et de l'identifiant du satellite afin d'identifier l'intensité du signal et les erreurs chez vos contacts. Pour plus d'informations, consultez la section [Utilisation CloudWatch des métriques Amazon](#).
- [AWS Notifications des utilisateurs](#) peut être utilisé pour configurer des canaux de diffusion afin d'être informé AWS Ground Station des événements. Vous recevez une notification lorsqu'un événement correspond à une règle que vous avez spécifiée. Vous pouvez recevoir des notifications d'événements via plusieurs canaux, notamment par e-mail, [Amazon Q Developer dans les applications de chat](#), les notifications par chat ou les notifications [AWS Console Mobile Application](#) push. Vous pouvez également consulter les notifications dans le [centre de notifications de](#) la AWS console. Notifications des utilisateurs prendre en charge l'agrégation, qui peut réduire le nombre de notifications que vous recevez lors d'événements spécifiques.

Utilisez les rubriques suivantes pour surveiller AWS Ground Station.

Rubriques

- [Automatisez AWS Ground Station avec des événements](#)

- [Enregistrez les appels AWS Ground Station d'API avec AWS CloudTrail](#)
- [Afficher les statistiques avec Amazon CloudWatch](#)

Automatisez AWS Ground Station avec des événements

Note

Le terme « événement » est utilisé partout dans le présent document. CloudWatch Les événements et EventBridge les deux sont le même service sous-jacent et la même API. Les règles permettant de faire correspondre les événements entrants et de les acheminer vers des cibles à des fins de traitement peuvent être établies à l'aide de l'un ou l'autre service.

Les événements vous permettent d'automatiser vos AWS services et de répondre automatiquement aux événements du système tels que les problèmes de disponibilité des applications ou les modifications des ressources. Les événements liés AWS aux services sont diffusés en temps quasi réel. Vous pouvez écrire des règles simples pour indiquer quels événements vous intéressent et les actions automatisées à effectuer quand un événement correspond à une règle. Certaines des actions qui peuvent être déclenchées automatiquement sont les suivantes :

- Invoquer une fonction AWS Lambda
- Appel de la fonctionnalité Exécuter la commande d'Amazon EC2
- Relais de l'événement à Amazon Kinesis Data Streams
- Activation d'une machine à AWS Step Functions états
- Notification d'une rubrique Amazon SNS ou d'une file d'attente Amazon SQS

Voici quelques exemples d'utilisation d'événements avec AWS Ground Station :

- Invocation d'une fonction Lambda pour automatiser le démarrage et l'arrêt des instances Amazon EC2 en fonction de l'état de l'événement.
- Publication sur une rubrique Amazon SNS chaque fois qu'un contact change d'état. Ces rubriques peuvent être configurées pour envoyer des notifications par e-mail au début ou à la fin des contacts.

Pour plus d'informations, consultez le [guide de l'utilisateur Amazon EventBridge Events](#).

AWS Ground Station Types d'événements

Note

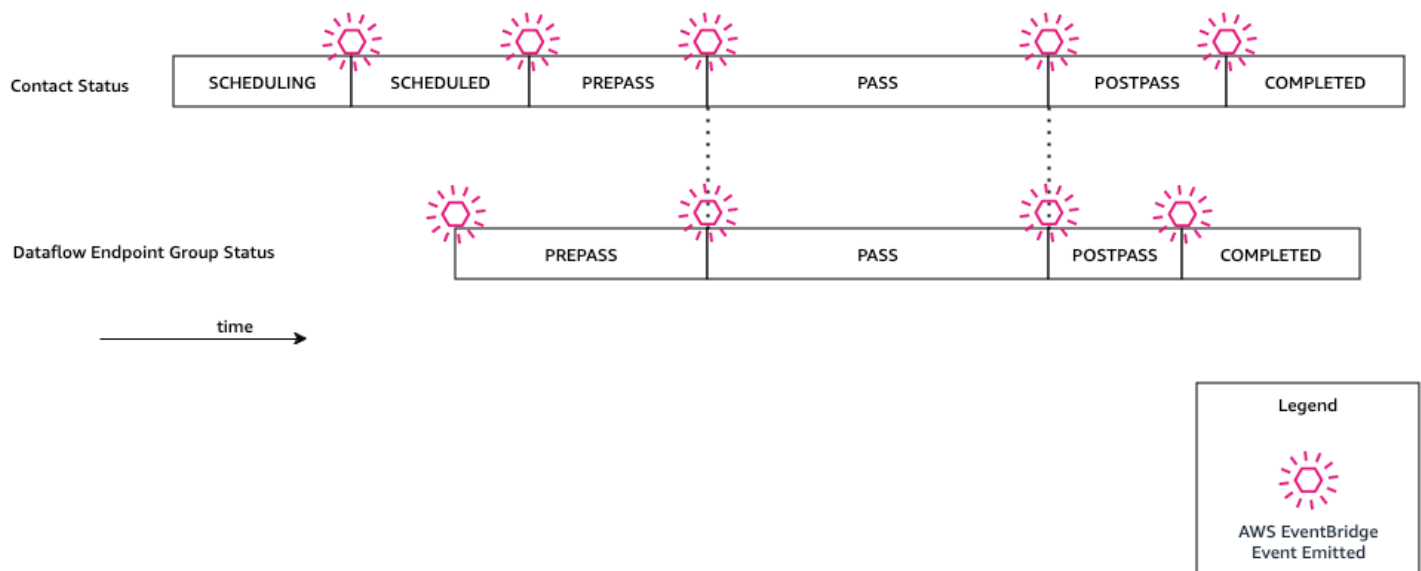
La valeur « source » de tous les événements générés par AWS Ground Station est « aws.groundstation ».

AWS Ground Station émet des événements liés aux changements d'état pour vous permettre de personnaliser votre automatisation. Actuellement, AWS Ground Station prend en charge les événements de changement d'état des contacts, les événements de modification du groupe de points de terminaison du flux de données et les événements de changement d'état des éphémérides. Les sections suivantes fournissent des informations détaillées sur chaque type.

Chronologie des événements de contact

AWS Ground Station émet des événements lorsque votre contact change d'état. Pour plus d'informations sur la nature de ces changements d'État et sur la signification des États eux-mêmes, voir [Comprendre le cycle de vie des contacts](#). Tous les groupes de points de terminaison de flux de données utilisés dans votre contact sont associés à un ensemble indépendant d'événements qui sont également émis. Au cours de cette même période, nous émettons également des événements pour votre groupe de points de terminaison de flux de données. Vous pouvez configurer l'heure précise des événements avant et après le passage lorsque vous configurez votre profil de mission et votre groupe de points de terminaison de flux de données.

Le schéma suivant montre les statuts et les événements émis pour un contact nominal et son groupe de points de terminaison de flux de données associé.



Modification de l'état d'un contact Ground Station

Si vous souhaitez effectuer une action spécifique lorsqu'un prochain contact change d'état, vous pouvez définir une règle pour automatiser cette action. Ceci est utile lorsque vous souhaitez recevoir des notifications sur les changements d'état de votre contact. Si vous souhaitez modifier le moment où vous recevez ces événements, vous pouvez modifier le profil de votre mission [contactPrePassDurationSeconds](#) et [contactPostPassDurationSeconds](#). Les événements sont envoyés à la région à partir de laquelle le contact a été planifié.

Un exemple d'événement est fourni ci-dessous.

```
{
  "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"
  }
}

```

Les valeurs possibles pour `contactStatus` sont définies dans [the section called “AWS Ground Station statuts des contacts”](#).

Modification de l'état du groupe de points de terminaison du flux de données Ground Station

Si vous souhaitez effectuer une action lorsque votre groupe de points de terminaison de flux de données est utilisé pour recevoir des données, vous pouvez configurer une règle pour automatiser cette action. Cela vous permettra d'effectuer différentes actions en réponse aux états changeants du groupe de points de terminaison de flux de données. Si vous souhaitez modifier le moment où vous recevez ces événements, utilisez un groupe de points de terminaison de flux de données avec un et différent [contactPrePassDurationSeconds](#). [contactPostPassDurationSeconds](#) Cet événement sera envoyé à la région du groupe de points de terminaison de flux de données.

Vous trouverez un exemple ci-dessous.

```

{
  "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"
}
}
```

Les états possibles pour `dataflowEndpointGroupState` sont : PREPASS, PASS, POSTPASS et COMPLETED.

Événements Ephemeris

Changement d'état des éphémérides de la station au sol

Si vous souhaitez effectuer une action lorsqu'une éphéméride change d'état, vous pouvez définir une règle pour automatiser cette action. Cela vous permet d'effectuer différentes actions en réponse au changement d'état d'une éphéméride. Par exemple, vous pouvez effectuer une action lorsque la validation d'une éphéméride est terminée, et c'est maintenant le cas. `ENABLED` La notification de cet événement sera envoyée à la région où les éphémérides ont été téléchargées.

Vous trouverez un exemple ci-dessous.

```
{
  "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-
bccccca005000"
  ],
  "detail": {
    "ephemerisStatus": "ENABLED",
    "ephemerisId": "111111-cccc-bbbb-a555-bccccca005000",
  }
}
```

```
    "satelliteId": "10313191-c9d9-4ecb-a5f2-bc55cab050ec"  
  }  
}
```

Les états possibles pour les `ephemerisStatus` incluent
ENABLED,VALIDATING,INVALID,ERROR,DISABLED, EXPIRED

Enregistrez les appels AWS Ground Station d'API avec AWS CloudTrail

AWS Ground Station est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans AWS Ground Station. CloudTrail capture tous les appels d'API AWS Ground Station sous forme d'événements. Les appels capturés incluent des appels provenant de la AWS Ground Station console et des appels de code vers les opérations de l' AWS Ground Station API. Si vous créez un suivi, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris les événements pour AWS Ground Station. Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents dans la CloudTrail console dans Historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite AWS Ground Station, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le [guide de AWS CloudTrail l'utilisateur](#).

AWS Ground Station Informations dans CloudTrail

CloudTrail est activé sur votre AWS compte lorsque vous le créez. Lorsqu'une activité se produit dans AWS Ground Station, cette activité est enregistrée dans un CloudTrail événement avec d'autres événements de AWS service dans l'historique des événements. Vous pouvez consulter, rechercher et télécharger les événements récents dans votre AWS compte. Pour plus d'informations, consultez la section [Affichage des événements à l'aide de l'historique des CloudTrail événements](#).

Pour un enregistrement continu des événements de votre AWS compte, y compris des événements pour AWS Ground Station, créez un parcours. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un journal d'activité dans la console, il s'applique à toutes les régions AWS. Le journal enregistre les événements de toutes les

régions de la AWS partition et transmet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour plus d'informations, consultez les ressources suivantes :

- [Vue d'ensemble de la création d'un journal d'activité](#)
- [CloudTrail Services et intégrations pris en charge](#)
- [Configuration des notifications Amazon SNS pour CloudTrail](#)
- [Réception de fichiers CloudTrail journaux de plusieurs régions](#) et [réception de fichiers CloudTrail journaux de plusieurs comptes](#)

Toutes les AWS Ground Station actions sont enregistrées CloudTrail et documentées dans la [référence de l'AWS Ground Station API](#). Par exemple, les appels au `ReserveContact`, `CancelContact` et les `ListConfigs` actions génèrent des entrées dans les fichiers CloudTrail journaux.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été faite avec les informations d'identification de l'utilisateur root ou Gestion des identités et des accès AWS (IAM).
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la demande a été faite par un autre AWS service.

Pour plus d'informations, consultez la section [Élément userIdentity CloudTrail](#).

Comprendre les entrées du fichier AWS Ground Station journal

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment Amazon S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

L'exemple suivant montre une entrée de CloudTrail journal illustrant l'ReserveContactation.

Exemple : 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"  
}
```

Afficher les statistiques avec Amazon CloudWatch

Lors d'un contact, capture et envoie AWS Ground Station automatiquement les données à des CloudWatch fins d'analyse. Vos données peuvent être consultées dans la CloudWatch console Amazon. Pour plus d'informations sur l'accès et CloudWatch les métriques, consultez [Using Amazon CloudWatch Metrics](#).

La fonction de AWS Ground Station télémétrie peut également être utilisée pour recevoir des métriques en temps quasi réel lors des contacts. CloudWatch les statistiques ne sont pas disponibles en temps quasi réel et peuvent entraîner des retards de livraison. CloudWatch agrège également les métriques sur une période d'une seconde, ce qui peut réduire la granularité des données. La fonction de télémétrie fournit les statistiques individuelles et les transmet en temps quasi réel directement sur votre AWS compte. Pour de plus amples informations, veuillez consulter [Travailler avec la télémétrie](#).

Important

AWS Ground Station émet CloudWatch des métriques vers la AWS région associée à l'emplacement de la station au sol du contact, et non vers la AWS région à partir de laquelle le contact a été planifié. Pour consulter les statistiques d'un contact, vous devez accéder CloudWatch à la région de la station au sol. Pour plus d'informations sur AWS la région associée à l'emplacement de chaque station au sol, consultez [Trouver la AWS région pour l'emplacement d'une station au sol](#). Pour recevoir des données de télémétrie dans la région à partir de laquelle vous planifiez vos contacts, vous pouvez utiliser la fonction de AWS Ground Station télémétrie. Pour plus d'informations, consultez [Travailler avec la télémétrie](#).

AWS Ground Station Métriques et dimensions

Quelles métriques sont disponibles ?

Les statistiques suivantes sont disponibles auprès de AWS Ground Station.

Note

Les métriques spécifiques émises dépendent des AWS Ground Station capacités utilisées. Selon votre configuration, seul un sous-ensemble des métriques ci-dessous peut être émis.

Métrique	Dimensions métriques	Description
AzimuthAngle	Satelliteld	Angle d'azimut de l'antenne. Le nord réel est à 0 degré et l'est à 90 degrés. Unités : degrés
BitErrorRate	Canal, Polarisation, Satelliteld	Le taux d'erreur sur les bits dans un nombre donné de transmissions binaires. Les erreurs de bits sont causées par le bruit, la distorsion ou les interférences Unités : erreurs de bits par unité de temps
BlockErrorRate	Canal, Polarisation, Satelliteld	Taux d'erreur de blocs dans un nombre donné de blocs reçus. Les erreurs de blocs sont

Métrique	Dimensions métriques	Description
		<p>causées par des interférences.</p> <p>Unités : blocs erronés/Nombre total de blocs</p>
CarrierFrequencyRecovery_Cn0	Catégorie, Config, Satellitelid	<p>Rapport entre le support et la densité du bruit par unité de bande passante.</p> <p>Unités : décibel-Hertz (dB-Hz)</p>
CarrierFrequencyRecovery_Locked	Catégorie, Config, Satellitelid	<p>Réglé sur 1 lorsque la boucle de récupération de fréquence porteuse du démodulateur est verrouillée et sur 0 lorsqu'elle est déverrouillée.</p> <p>Unités : sans unité</p>

Métrique	Dimensions métriques	Description
CarrierFrequencyRecovery_OffsetFrequency_Hz	Catégorie, Config, SatelliteId	<p>Le décalage entre le centre du signal estimé et la fréquence centrale idéale. Cela est dû au décalage Doppler et au décalage de l'oscillateur local entre le vaisseau spatial et le système d'antenne.</p> <p>Unités : hertz (Hz)</p>
ElevationAngle	SatelliteId	<p>Angle d'élévation de l'antenne . L'horizon est de 0 degré et le zénith de 90 degrés.</p> <p>Unités : degrés</p>
Es/N0	Canal, Polarisation, SatelliteId	<p>Rapport entre l'énergie par symbole et la densité spectrale de puissance du bruit.</p> <p>Unités : décibels (dB)</p>

Métrique	Dimensions métriques	Description
ReceivedPower	Polarisation, SatelliteId	<p>La puissance du signal mesurée dans le démodulateur/décodeur.</p> <p>Unités : décibels par rapport aux milliwatts (dBm)</p>
SymbolTimingRecovery_ErrorVectorMagnitude	Catégorie, Config, SatelliteId	<p>L'amplitude du vecteur d'erreur entre les symboles reçus et les points de constellation idéaux.</p> <p>Unités : pourcentage</p>
SymbolTimingRecovery_Locked	Catégorie, Config, SatelliteId	<p>Réglé sur 1 lorsque le symbole du démodulateur (chronométrage, boucle de restauration) est verrouillé et sur 0 lorsqu'il est déverrouillé</p> <p>Unités : sans unité</p>

Métrique	Dimensions métriques	Description
SymbolTimingRecovery_OffsetSymbolRate	Catégorie, Config, SatelliteId	<p>Le décalage entre le débit de symboles estimé et le taux de symboles de signal idéal. Cela est dû au décalage Doppler et au décalage de l'oscillateur local entre le vaisseau spatial et le système d'antenne.</p> <p>Unités : symboles/seconde</p>

Quelles sont les dimensions utilisées AWS Ground Station ?

Vous pouvez filtrer AWS Ground Station les données à l'aide des dimensions suivantes.

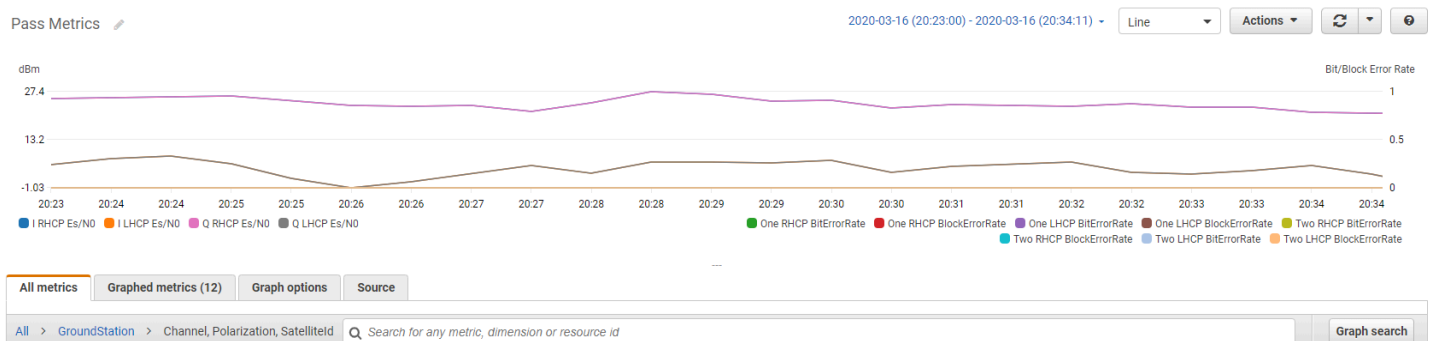
Dimension	Description
Category	Démodulation ou décodage.
Channel	Les canaux de chaque contact incluent Un, Deux, I (en phase) et Q (quadrature).
Config	Une démo d'antenne en liaison descendante décode la configuration arn.

Dimension	Description
Polarization	La polarisation pour chaque contact comprend la LHCP (polarisation circulaire tournant à gauche) ou la RHCP (polarisation circulaire tournant à droite).
SatelliteId	L'ID satellite contient l'ARN du satellite pour vos contacts.

Affichage des métriques

Lors de l'affichage des métriques dans un graphique, il est important de noter que la fenêtre d'agrégation détermine la façon dont vos métriques seront affichées. Chaque métrique d'un contact peut être affichée sous forme de données par seconde pendant 3 heures après la réception des données. Vos données seront agrégées par CloudWatch Metrics sous forme de données par minute après la fin de cette période de 3 heures. Si vous devez consulter vos statistiques sur une mesure de données par seconde, il est recommandé de consulter vos données dans les 3 heures suivant leur réception ou de les conserver en dehors des CloudWatch métriques. Pour plus d'informations sur la CloudWatch rétention, consultez [Amazon CloudWatch Concepts - Concepts - Conservation métrique](#).

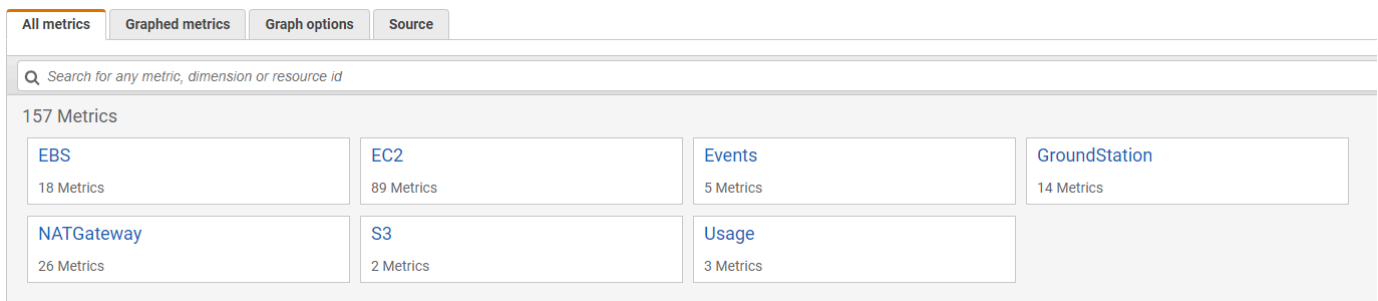
De plus, toutes les données capturées au cours des 60 premières secondes ne contiendront pas suffisamment d'informations pour produire des métriques significatives et ne seront probablement pas affichées. Pour afficher des métriques significatives, il est recommandé d'afficher vos données après 60 secondes.



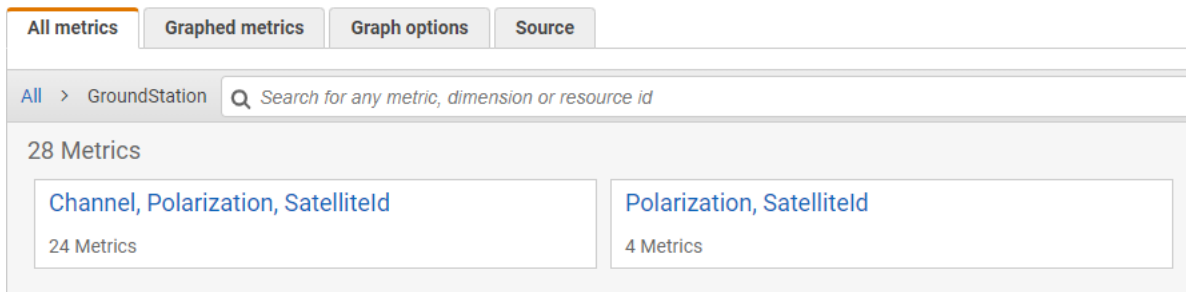
Pour plus d'informations sur la représentation graphique des AWS Ground Station métriques CloudWatch, consultez la section Représentation [graphique des métriques](#).

Pour afficher des métriques à l'aide de la console

1. Déterminez la AWS région associée à l'emplacement de votre station au sol. AWS Ground Station émet CloudWatch des métriques dans la région associée à l'emplacement de la station au sol de votre contact. Pour obtenir la liste des emplacements des stations au sol et AWS des régions associées, voir [Trouver la AWS région pour l'emplacement d'une station au sol](#).
2. Ouvrez la [CloudWatch console](#).
3. Dans le panneau de navigation, sélectionnez Métriques.
4. Sélectionnez l'espace de noms GroundStation.



5. Sélectionnez les dimensions métriques souhaitées (par exemple, canal, polarisation, Satelliteld).



6. L'onglet All metrics (Toutes les métriques) affiche toutes les métriques pour cette dimension dans l'espace de noms. Vous pouvez effectuer les opérations suivantes :
 - a. Pour trier le tableau, utilisez l'en-tête de colonne.
 - b. Pour représenter graphiquement une métrique, cochez la case associée à la métrique. Pour sélectionner toutes les mesures, cochez la case dans la ligne d'en-tête du tableau.
 - c. Pour filtrer par ressource, sélectionnez l'ID de ressource, puis Add to search.
 - d. Pour filtrer par métrique, choisissez le nom de la métrique, puis Add to search (Ajouter à la recherche).

Pour consulter les métriques à l'aide de AWS CLI

AWS Ground Station émet CloudWatch des métriques dans la région associée à l'emplacement de la station au sol de votre contact. Pour la liste des emplacements des stations au sol, leurs AWS régions associées, [Trouver la AWS région pour l'emplacement d'une station au sol](#). *ground-station-region-code* Remplacez-le par le code de AWS région correspondant à l'emplacement de votre station au sol (par exemple, us-west-2 pour Oregon 1, Hawaï 1 ou Alaska 1). Toutes les AWS CLI commandes suivantes de cette procédure doivent utiliser la même région.

1. Assurez-vous qu'il AWS CLI est installé. Pour plus d'informations sur l'installation AWS CLI, consultez [Installation de la version 2 de l'interface de ligne de commande AWS](#).
2. Identifiez la AWS région associée à l'emplacement de votre station au sol.
3. Utilisez la [get-metric-data](#) méthode de la CloudWatch CLI pour générer un fichier qui peut être modifié pour spécifier les métriques qui vous intéressent, puis être utilisé pour interroger ces métriques.

Pour ce faire, exécutez ce qui suit :`aws cloudwatch get-metric-data --region ground-station-region-code --generate-cli-skeleton`. Cela générera un résultat similaire à :

```
{
  "MetricDataQueries": [
    {
      "Id": "",
      "MetricStat": {
        "Metric": {
          "Namespace": "",
          "MetricName": "",
          "Dimensions": [
            {
              "Name": "",
              "Value": ""
            }
          ]
        },
        "Period": 0,
        "Stat": "",
        "Unit": "Seconds"
      }
    }
  ],
}
```

```

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

```

4. Répertoriez CloudWatch les métriques disponibles en exécutant `aws cloudwatch list-metrics --region ground-station-region-code`.

Si vous l'avez récemment utilisée AWS Ground Station, la méthode doit renvoyer une sortie contenant des entrées telles que :

```

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

```

Note

Si cela fait plus de 2 semaines que vous les avez utilisées pour la dernière fois AWS Ground Station, vous devrez inspecter manuellement le [tableau des métriques disponibles](#) pour trouver les noms et les dimensions des métriques dans l'espace de noms des AWS/GroundStation métriques. Pour plus d'informations sur CloudWatch les limitations, voir : [Afficher les métriques disponibles](#)

5. Modifiez le fichier JSON que vous avez créé à l'étape 2 pour qu'il corresponde aux valeurs requises à l'étape 3, par exemple `SatelliteId`, et `Polarization` à partir de vos métriques. Veillez également à mettre à jour les `StartTime` `EndTime` valeurs et pour qu'elles correspondent à celles de votre contact. Par exemple :

```
{
  "MetricDataQueries": [
    {
      "Id": "receivedPowerExample",
      "MetricStat": {
        "Metric": {
          "Namespace": "AWS/GroundStation",
          "MetricName": "ReceivedPower",
          "Dimensions": [
            {
              "Name": "SatelliteId",
              "Value":
                "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-bbbb-cccc-dddd-
                eeeeeeeeeee"
            },
            {
              "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 publie des métriques toutes les 1 à 60 secondes, selon la métrique. Les métriques ne seront pas renvoyées si le `Period` champ a une valeur inférieure à la période de publication de la métrique.

6. Exécutez `aws cloudwatch get-metric-data` avec le fichier de configuration créé lors des étapes précédentes. Vous trouverez un exemple ci-dessous.

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

Les métriques seront fournies avec des horodatages de votre contact. Un exemple de sortie de métrique AWS Ground Station est fourni ci-dessous.

```
{
  "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": []  
}
```

Sécurité dans AWS Ground Station

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficierez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité. AWS fournit des outils et des fonctionnalités spécifiques à la sécurité pour vous aider à atteindre vos objectifs de sécurité. Ces outils et fonctionnalités incluent la sécurité du réseau, la gestion de la configuration, le contrôle d'accès et la sécurité des données.

Lors de l'utilisation AWS Ground Station, nous vous recommandons de suivre les meilleures pratiques du secteur et de mettre en œuvre end-to-end le chiffrement. AWS vous permet APIs d'intégrer le chiffrement et la protection des données. Pour plus d'informations sur AWS la sécurité, consultez le livre blanc [Introduction à la sécurité AWS](#).

Consultez les rubriques suivantes pour apprendre à sécuriser vos ressources .

Rubriques

- [Identity and Access Management pour AWS Ground Station](#)
- [AWS politiques gérées pour AWS Ground Station](#)
- [Utiliser des rôles liés à un service pour Ground Station](#)
- [Chiffrement des données au repos pour AWS Ground Station](#)
- [Chiffrement des données pendant le transit pour AWS Ground Station](#)

Identity and Access Management pour AWS Ground Station

Gestion des identités et des accès AWS (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser AWS Ground Station les ressources. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- [Public ciblé](#)
- [Authentification par des identités](#)
- [Gestion de l'accès à l'aide de politiques](#)
- [Comment AWS Ground Station fonctionne avec IAM](#)

- [Exemples de politiques basées sur l'identité pour AWS Ground Station](#)
- [Résolution des problèmes AWS Ground Station d'identité et d'accès](#)

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir [Résolution des problèmes AWS Ground Station d'identité et d'accès](#))
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir [Comment AWS Ground Station fonctionne avec IAM](#))
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir [Exemples de politiques basées sur l'identité pour AWS Ground Station](#))

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS à l'aide de vos informations d'identification. Vous devez être authentifié en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en tant qu'identité fédérée à l'aide d'informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), d'une authentification unique ou d'informations d'identification. Google/Facebook Pour plus d'informations sur la connexion, consultez [Connexion à votre Compte AWS](#) dans le Guide de l'utilisateur Connexion à AWS .

Pour l'accès par programmation, AWS fournit un SDK et une CLI pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez [Signature AWS Version 4 pour les demandes d'API](#) dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une seule identité de connexion appelée utilisateur Compte AWS root qui dispose d'un accès complet à toutes Services AWS les ressources. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches

qui requièrent des informations d'identification de l'utilisateur racine, consultez [Tâches qui requièrent les informations d'identification de l'utilisateur racine](#) dans le Guide de l'utilisateur IAM.

Identité fédérée

Il est recommandé d'obliger les utilisateurs humains à utiliser la fédération avec un fournisseur d'identité pour accéder à Services AWS l'aide d'informations d'identification temporaires.

Une identité fédérée est un utilisateur provenant de l'annuaire de votre entreprise, de votre fournisseur d'identité Web ou Directory Service qui y accède à Services AWS l'aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées assument des rôles qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Pour plus d'informations, consultez [Qu'est-ce que IAM Identity Center ?](#) dans le Guide de l'utilisateur AWS IAM Identity Center .

Utilisateurs et groupes IAM

Un [utilisateur IAM](#) est une identité qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, voir [Exiger des utilisateurs humains qu'ils utilisent la fédération avec un fournisseur d'identité pour accéder à AWS l'aide d'informations d'identification temporaires](#) dans le guide de l'utilisateur IAM.

[Les groupes IAM](#) spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez [Cas d'utilisation pour les utilisateurs IAM](#) dans le Guide de l'utilisateur IAM.

Rôles IAM

Un [rôle IAM](#) est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Vous pouvez assumer un rôle en [passant d'un rôle utilisateur à un rôle IAM \(console\)](#) ou en appelant une opération AWS CLI ou AWS API. Pour plus d'informations, consultez [Méthodes pour endosser un rôle](#) dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

Gestion de l'accès à l'aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique définit les autorisations lorsqu'elles sont associées à une identité ou à une ressource. AWS évalue ces politiques lorsqu'un directeur fait une demande. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations les documents de politique JSON, consultez [Vue d'ensemble des politiques JSON](#) dans le Guide de l'utilisateur IAM.

À l'aide de politiques, les administrateurs précisent qui a accès à quoi en définissant quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l'opération.

Politiques basées sur l'identité

Les stratégies basées sur l'identité sont des documents de stratégie d'autorisations JSON que vous attachez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez [Définition d'autorisations IAM personnalisées avec des politiques gérées par le client](#) dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être des politiques intégrées (intégrées directement dans une seule identité) ou des politiques gérées (politiques autonomes associées à plusieurs identités). Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez [Choix entre les politiques gérées et les politiques en ligne](#) dans le Guide de l'utilisateur IAM.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Vous devez [spécifier un principal](#) dans une politique basée sur les ressources.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques AWS gérées par IAM dans une stratégie basée sur les ressources.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir les autorisations maximales accordées par les types de politiques les plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez [Limites d'autorisations pour des entités IAM](#) dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCPs) — Spécifiez les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez [Politiques de contrôle de service](#) dans le Guide de l'utilisateur AWS Organizations .
- Politiques de contrôle des ressources (RCPs) : définissez le maximum d'autorisations disponibles pour les ressources de vos comptes. Pour plus d'informations, voir [Politiques de contrôle des ressources \(RCPs\)](#) dans le guide de AWS Organizations l'utilisateur.
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez [Politiques de session](#) dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS déterminer s'il faut autoriser une demande lorsque plusieurs types de politiques sont impliqués, consultez la section [Logique d'évaluation des politiques](#) dans le guide de l'utilisateur IAM.

Comment AWS Ground Station fonctionne avec IAM

Avant d'utiliser IAM pour gérer l'accès à AWS Ground Station, découvrez les fonctionnalités IAM disponibles. AWS Ground Station

Fonctionnalités IAM que vous pouvez utiliser avec AWS Ground Station

Fonctionnalité IAM	AWS Ground Station soutien
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique (spécifiques au service)	Oui
ACLs	Non
ABAC (étiquettes dans les politiques)	Oui
Informations d'identification temporaires	Oui
Autorisations de principal	Oui
Rôles du service	Non
Rôles liés à un service	Oui

Pour obtenir une vue d'ensemble de la façon dont AWS Ground Station les autres AWS services fonctionnent avec la plupart des fonctionnalités IAM, consultez la section [AWS Services compatibles avec IAM](#) dans le Guide de l'utilisateur IAM.

Politiques basées sur l'identité pour AWS Ground Station

Prend en charge les politiques basées sur l'identité : oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez [Définition d'autorisations IAM personnalisées avec des politiques gérées par le client](#) dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez [Références des éléments de politique JSON IAM](#) dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour AWS Ground Station

Pour consulter des exemples de politiques AWS Ground Station basées sur l'identité, consultez. [Exemples de politiques basées sur l'identité pour AWS Ground Station](#)

Politiques basées sur les ressources au sein de AWS Ground Station

Prend en charge les politiques basées sur les ressources : non

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez [spécifier un principal](#) dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. Pour plus d'informations, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

Actions politiques pour AWS Ground Station

Prend en charge les actions de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour consulter la liste des AWS Ground Station actions, voir [Actions définies par AWS Ground Station](#) dans la référence d'autorisation de service.

Les actions de politique en AWS Ground Station cours utilisent le préfixe suivant avant l'action :

```
groundstation
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Pour consulter des exemples de politiques AWS Ground Station basées sur l'identité, consultez [Exemples de politiques basées sur l'identité pour AWS Ground Station](#)

Ressources politiques pour AWS Ground Station

Prend en charge les ressources de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Il est recommandé de définir une ressource à l'aide de son [Amazon Resource Name \(ARN\)](#). Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

Pour consulter la liste des types de AWS Ground Station ressources et leurs caractéristiques ARNs, voir [Ressources définies par AWS Ground Station](#) dans la référence d'autorisation de service. Pour savoir grâce à quelles actions vous pouvez spécifier l'ARN de chaque ressource, consultez [Actions définies par AWS Ground Station](#).

Pour consulter des exemples de politiques AWS Ground Station basées sur l'identité, consultez.

[Exemples de politiques basées sur l'identité pour AWS Ground Station](#)

Clés de conditions de politique pour AWS Ground Station

Prend en charge les clés de condition de politique spécifiques au service : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` indique à quel moment les instructions s'exécutent en fonction de critères définis. Vous pouvez créer des expressions conditionnelles qui utilisent des [opérateurs de condition](#), tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande. Pour voir toutes les clés de condition AWS globales, voir les clés de [contexte de condition AWS globales](#) dans le guide de l'utilisateur IAM.

Pour consulter la liste des clés de AWS Ground Station condition, voir [Clés de condition pour AWS Ground Station](#) la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez la section [Actions définies par AWS Ground Station](#).

Pour consulter des exemples de politiques AWS Ground Station basées sur l'identité, consultez.

[Exemples de politiques basées sur l'identité pour AWS Ground Station](#)

ACLs in AWS Ground Station

Supports ACLs : Non

Les listes de contrôle d'accès (ACLs) contrôlent les principaux (membres du compte, utilisateurs ou rôles) autorisés à accéder à une ressource. ACLs sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

ABAC avec AWS Ground Station

Prise en charge d'ABAC (balises dans les politiques) : Oui

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs appelés balises. Vous pouvez associer des balises aux entités et aux AWS

ressources IAM, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à la balise de la ressource.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'[élément de condition](#) d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez [Définition d'autorisations avec l'autorisation ABAC](#) dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez [Utilisation du contrôle d'accès par attributs \(ABAC\)](#) dans le Guide de l'utilisateur IAM.

Utilisation d'informations d'identification temporaires avec AWS Ground Station

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès à court terme aux AWS ressources et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez [Informations d'identification de sécurité temporaires dans IAM](#) et [Services AWS compatibles avec IAM](#) dans le Guide de l'utilisateur IAM.

Autorisations principales interservices pour AWS Ground Station

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions d'accès direct (FAS) utilisent les autorisations du principal appelant et Service AWS, combinées Service AWS à la demande d'envoi de demandes aux services en aval. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez la section [Sessions de transmission d'accès](#).

Rôles de service pour AWS Ground Station

Prend en charge les rôles de service : Non

Un rôle de service est un [rôle IAM](#) qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer un rôle de service à partir d'IAM. Pour plus d'informations, consultez [Création d'un rôle pour la délégation d'autorisations à un Service AWS](#) dans le Guide de l'utilisateur IAM.

Warning

La modification des autorisations associées à un rôle de service peut perturber AWS Ground Station les fonctionnalités. Modifiez les rôles de service uniquement lorsque AWS Ground Station vous recevez des instructions à cet effet.

Rôles liés à un service pour AWS Ground Station

Prend en charge les rôles liés à un service : oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus d'informations sur la création ou la gestion des rôles liés à un service, consultez [Services AWS qui fonctionnent avec IAM](#). Recherchez un service dans le tableau qui inclut un Yes dans la colonne Rôle lié à un service. Choisissez le lien Oui pour consulter la documentation du rôle lié à ce service.

Exemples de politiques basées sur l'identité pour AWS Ground Station

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou modifier les ressources AWS Ground Station . Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez [Création de politiques IAM \(console\)](#) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par AWS Ground Station, y compris le format de ARNs pour chacun des types de ressources, voir [Actions, ressources et clés de condition AWS Ground Station](#) dans la référence d'autorisation de service.

Rubriques

- [Bonnes pratiques en matière de politiques](#)
- [Utilisation de la AWS Ground Station console](#)
- [Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations](#)

Bonnes pratiques en matière de politiques

Les politiques basées sur l'identité déterminent si quelqu'un peut créer, accéder ou supprimer AWS Ground Station des ressources dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez [politiques gérées par AWS](#) ou [politiques gérées par AWS pour les activités professionnelles](#) dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez [politiques et autorisations dans IAM](#) dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que CloudFormation. Pour plus d'informations, consultez [Conditions pour éléments de politique JSON IAM](#) dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles.

Pour plus d'informations, consultez [Validation de politiques avec IAM Access Analyzer](#) dans le Guide de l'utilisateur IAM.

- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez [Sécurisation de l'accès aux API avec MFA](#) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez [Bonnes pratiques de sécurité dans IAM](#) dans le Guide de l'utilisateur IAM.

Utilisation de la AWS Ground Station console

Pour accéder à la AWS Ground Station console, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails AWS Ground Station des ressources de votre Compte AWS. Si vous créez une politique basée sur l'identité qui est plus restrictive que l'ensemble minimum d'autorisations requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette politique.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour garantir que les utilisateurs et les rôles peuvent toujours utiliser la AWS Ground Station console, associez également la politique AWS Ground Station *ConsoleAccess* ou la politique *ReadOnly* AWS gérée aux entités. Pour plus d'informations, consultez [Ajout d'autorisations à un utilisateur](#) dans le Guide de l'utilisateur IAM.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

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

```

    "Sid": "ViewOwnUserInfo",
    "Effect": "Allow",
    "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
    ],
    "Resource": ["arn:aws:iam::*:user/${aws:username}"]
},
{
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

Résolution des problèmes AWS Ground Station d'identité et d'accès

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec AWS Ground Station IAM.

Rubriques

- [Je ne suis pas autorisé à effectuer une action dans AWS Ground Station](#)
- [Je ne suis pas autorisé à effectuer iam : PassRole](#)
- [Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes AWS Ground Station ressources](#)

Je ne suis pas autorisé à effectuer une action dans AWS Ground Station

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource `my-example-widget` fictive, mais ne dispose pas des autorisations `groundstation:GetWidget` fictives.

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

Dans ce cas, la politique qui s'applique à l'utilisateur `mateojackson` doit être mise à jour pour autoriser l'accès à la ressource `my-example-widget` à l'aide de l'action `groundstation:GetWidget`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez une erreur selon laquelle vous n'êtes pas autorisé à exécuter `iam:PassRole` l'action, vos stratégies doivent être mises à jour afin de vous permettre de transmettre un rôle à AWS Ground Station.

Certains services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, vous devez disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour exécuter une action dans AWS Ground Station. Toutefois, l'action nécessite que le service ait des autorisations accordées par une fonction de service. Mary n'est pas autorisée à transmettre le rôle au service.

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

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes AWS Ground Station ressources

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACLs), vous pouvez utiliser ces politiques pour autoriser les utilisateurs à accéder à vos ressources.

Pour plus d'informations, consultez les éléments suivants :

- Pour savoir si ces fonctionnalités sont prises AWS Ground Station en charge, consultez [Comment AWS Ground Station fonctionne avec IAM](#).
- Pour savoir comment fournir l'accès à vos ressources sur celles Comptes AWS que vous possédez, consultez la section [Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS que vous possédez](#) dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section [Fournir un accès à des ressources Comptes AWS détenues par des tiers](#) dans le guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez [Fournir un accès à des utilisateurs authentifiés en externe \(fédération d'identité\)](#) dans le Guide de l'utilisateur IAM.
- Pour en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez [Accès intercompte aux ressources dans IAM](#) dans le Guide de l'utilisateur IAM.

AWS politiques gérées pour AWS Ground Station

Une politique AWS gérée est une politique autonome créée et administrée par AWS. AWS les politiques gérées sont conçues pour fournir des autorisations pour de nombreux cas d'utilisation courants afin que vous puissiez commencer à attribuer des autorisations aux utilisateurs, aux groupes et aux rôles.

N'oubliez pas que les politiques AWS gérées peuvent ne pas accorder d'autorisations de moindre privilège pour vos cas d'utilisation spécifiques, car elles sont accessibles à tous les AWS clients. Nous vous recommandons de réduire encore les autorisations en définissant des [politiques gérées par le client](#) qui sont propres à vos cas d'utilisation.

Vous ne pouvez pas modifier les autorisations définies dans les politiques AWS gérées. Si les autorisations définies dans une politique AWS gérée sont mises à jour, la mise à jour affecte toutes les identités principales (utilisateurs, groupes et rôles) auxquelles la politique est attachée. AWS est le plus susceptible de mettre à jour une politique AWS gérée lorsqu'une nouvelle politique Service AWS est lancée ou lorsque de nouvelles opérations d'API sont disponibles pour les services existants.

Pour plus d'informations, consultez [Politiques gérées par AWS](#) dans le Guide de l'utilisateur IAM.

AWS politique gérée : AWSGround StationAgentInstancePolicy

Vous pouvez associer la politique `AWSGroundStationAgentInstancePolicy` à vos identités IAM.

Cette politique accorde à AWS Ground Station l'agent des autorisations d'accès à votre instance Amazon EC2, ce qui permet à l'instance d'envoyer et de recevoir des données lors des contacts avec Ground Station. Toutes les autorisations de cette politique proviennent du service Ground Station.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `groundstation`— Permet aux instances de point de terminaison de flux de données d'appeler l'agent Ground Station. APIs

Pour consulter la dernière version du document de politique JSON, consultez [AWSGroundStationAgentInstancePolicy](#) le guide de référence des politiques gérées par AWS.

AWS politique gérée : AWSService RoleForGroundStationDataflowEndpointGroupPolicy

Vous ne pouvez pas vous associer AWSService RoleForGroundStationDataflowEndpointGroupPolicy à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet d' AWS Ground Station effectuer des actions en votre nom. Pour plus d'informations, consultez la section [Utilisation de rôles liés à un service](#).

Cette politique accorde des autorisations EC2 qui permettent de AWS Ground Station trouver des IPv4 adresses publiques.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `ec2:DescribeAddresses`— Permet AWS Ground Station de répertorier toutes les informations IPs associées EIPs en votre nom.
- `ec2:DescribeNetworkInterfaces`— Permet AWS Ground Station d'obtenir des informations sur les interfaces réseau associées aux instances EC2 en votre nom.

Pour consulter la dernière version du document de politique JSON, consultez [AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy](#) le guide de référence des politiques gérées par AWS.

AWS Ground Station mises à jour des politiques AWS gérées

Consultez les détails des mises à jour des politiques AWS gérées AWS Ground Station depuis que ce service a commencé à suivre ces modifications. Pour recevoir des alertes automatiques concernant les modifications apportées à cette page, abonnez-vous au flux RSS sur la page Historique du AWS Ground Station document.

Modifier	Description	Date
AWSGroundStationAgentInstancePolicy – Mise à jour d'une stratégie existante	AWS Ground Station a ajouté de nouvelles autorisations pour permettre aux agents de récupérer la réponse aux tâches URLs pour améliorer les opérations de la Ground Station.	13 novembre 2025
AWSGroundStationAgentInstancePolicy : nouvelle politique	AWS Ground Station a ajouté une nouvelle politique pour fournir à l'instance de point de terminaison du flux de données les autorisations nécessaires pour utiliser l'agent AWS Ground Station.	12 avril 2023
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy : nouvelle politique	AWS Ground Station a ajouté une nouvelle politique qui accorde des autorisations EC2 pour permettre de AWS Ground Station trouver des adresses IPv4 publiques associées aux EIP et des interfaces réseau associées aux instances EC2.	2 novembre 2022
AWS Ground Station a commencé à suivre les modifications	AWS Ground Station a commencé à suivre les modifications apportées aux politiques AWS gérées.	01 mars 2021

Utiliser des rôles liés à un service pour Ground Station

AWS Ground Station utilise des Gestion des identités et des accès AWS rôles liés à un [service](#) (IAM). Un rôle lié à un service est un type unique de rôle IAM directement lié à Ground Station. Les rôles liés aux services sont prédéfinis par Ground Station et incluent toutes les autorisations dont le service a besoin pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service facilite la configuration de Ground Station, car vous n'avez pas à ajouter manuellement les autorisations nécessaires. Ground Station définit les autorisations associées à ses rôles liés aux services et, sauf indication contraire, seule Ground Station peut assumer ses rôles. Les autorisations définies comprennent la politique de confiance et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés à un service, consultez la section [AWS Services qui fonctionnent avec IAM](#) et recherchez les services dont la valeur est Oui dans la colonne Rôles liés à un service. Sélectionnez un Oui ayant un lien pour consulter la documentation du rôle lié à un service, pour ce service.

Autorisations de rôle liées au service pour Ground Station

Ground Station utilise le rôle lié à un service nommé :

`AWSServiceRoleForGroundStationDataflowEndpointGroupAWS` Ground Station utilise ce rôle lié à un service pour appeler EC2 afin de rechercher des adresses publiques. IPv4

Le rôle `AWSService RoleForGroundStationDataflowEndpointGroup` lié à un service fait confiance aux services suivants pour assumer le rôle :

- `groundstation.amazonaws.com`

La politique d'autorisation des rôles nommée `AWSService`

`RoleForGroundStationDataflowEndpointGroupPolicy` permet à Ground Station d'effectuer les actions suivantes sur les ressources spécifiées :

- Action : `ec2:DescribeAddresses` sur `all AWS resources (*)`

Cette action permet à Ground Station de répertorier tous les IPs éléments associés à EIPs.

- Action : `ec2:DescribeNetworkInterfaces` sur `all AWS resources (*)`

Cette action permet à Ground Station d'obtenir des informations sur les interfaces réseau associées aux instances EC2

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez [Autorisations de rôles liés à un service](#) dans le Guide de l'utilisateur IAM.

Création d'un rôle lié à un service pour Ground Station

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez un rôle `DataflowEndpointGroup` dans AWS CLI ou dans l' AWS API, Ground Station crée le rôle lié au service pour vous.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez un `DataflowEndpointGroup`, Ground Station crée à nouveau le rôle lié au service pour vous.

Vous pouvez également utiliser la console IAM pour créer un rôle lié à un service avec le cas d'utilisation de Data Delivery to Amazon EC2. Dans l'API AWS CLI ou dans l' AWS API, créez un rôle lié à un service avec le nom du `groundstation.amazonaws.com` service. Pour plus d'informations, consultez [Création d'un rôle lié à un service](#) dans le Guide de l'utilisateur IAM. Si vous supprimez ce rôle lié à un service, vous pouvez utiliser ce même processus pour créer le rôle à nouveau.

Modification d'un rôle lié à un service pour Ground Station

Ground Station ne vous permet pas de modifier le rôle `AWSServiceRoleForGroundStationDataflowEndpointGroup` lié au service. Après avoir créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez [Modification d'un rôle lié à un service](#) dans le Guide de l'utilisateur IAM.

Supprimer un rôle lié à un service pour Ground Station

Si vous n'avez plus besoin d'utiliser une fonctionnalité ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité inutilisée qui n'est pas surveillée ou gérée activement.

Vous ne pouvez supprimer un rôle lié à un service qu'après l'avoir d'abord supprimé `DataflowEndpointGroups` en utilisant le rôle lié à un service. Cela vous empêche de révoquer par inadvertance les autorisations accordées à votre `DataflowEndpointGroups`. Si un rôle lié à un service est utilisé avec plusieurs rôles `DataflowEndpointGroups`, vous devez supprimer tous ceux `DataflowEndpointGroups` qui utilisent le rôle lié à un service avant de pouvoir le supprimer.

Note

Si le service Ground Station utilise le rôle lorsque vous essayez de supprimer les ressources, la suppression risque d'échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer les ressources de Ground Station utilisées par `AWSServiceRoleForGroundStationDataflowEndpointGroup`

- Supprimez `DataflowEndpointGroups` via l'AWS CLI ou l'API AWS.

Pour supprimer manuellement le rôle lié au service à l'aide d'IAM

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié au `AWSServiceRoleForGroundStationDataflowEndpointGroup` service. Pour plus d'informations, consultez la section [Suppression d'un rôle lié à un service](#) dans le Guide de l'utilisateur IAM.

Régions prises en charge pour les rôles liés au service Ground Station

Ground Station prend en charge l'utilisation de rôles liés au service dans toutes les régions où le service est disponible. Pour plus d'informations, consultez le [tableau des régions](#).

Résolution des problèmes

`NOT_AUTHORIZED_TO_CREATE_SLR`- Cela indique que le rôle de votre compte utilisé pour appeler l' `CreateDataflowEndpointGroup` API n'est pas `iam:CreateServiceLinkedRole` autorisé. Un administrateur `iam:CreateServiceLinkedRole` autorisé doit créer manuellement le rôle lié au service pour votre compte.

Chiffrement des données au repos pour AWS Ground Station

AWS Ground Station fournit un chiffrement par défaut pour protéger vos données sensibles au repos à l'aide de clés de chiffrement AWS détenues par vos soins.

- **AWS clés détenues** : AWS Ground Station utilise ces clés par défaut pour chiffrer automatiquement les données personnelles et les éphémérides directement identifiables. Vous ne pouvez pas consulter, gérer ou utiliser les clés AWS détenues, ni auditer leur utilisation. Cependant, il n'est pas nécessaire de prendre des mesures ou de modifier les programmes pour protéger les clés qui chiffrer les données. Pour plus d'informations, consultez la section [AWS-owned keys](#) dans le [guide du AWS Key Management Service développeur](#).

Le chiffrement des données au repos par défaut permet de réduire les frais opérationnels et la complexité liés à la protection des données sensibles. Dans le même temps, il permet de créer des applications sécurisées qui répondent à une stricte conformité en matière de chiffrement, ainsi qu'aux exigences réglementaires.

AWS Ground Station applique le chiffrement à toutes les données sensibles au repos. Toutefois, pour certaines AWS Ground Station ressources, telles que les éphémérides, vous pouvez choisir d'utiliser une clé gérée par le client à la place des clés gérées par défaut. AWS

- **Clés gérées par le client** : AWS Ground Station prend en charge l'utilisation d'une clé symétrique gérée par le client que vous créez, détenez et gérez à la place du chiffrement AWS détenu existant. Étant donné que vous avez le contrôle total de cette couche de chiffrement, vous pouvez effectuer les tâches suivantes :
 - Établissement et gestion des stratégies de clé
 - Établissement et gestion des politiques IAM et des octrois
 - Activation et désactivation des stratégies de clé
 - Rotation des matériaux de chiffrement de clé
 - Ajout de balises
 - Création d'alias de clé
 - Planification des clés pour la suppression

Pour plus d'informations, consultez la section [clé gérée par le client](#) dans le [guide du AWS Key Management Service développeur](#).

Le tableau suivant récapitule les ressources pour lesquelles l'utilisation de clés gérées par le client est prise en charge par AWS Ground Station.

Type de données	AWS chiffrement par clé détenue	Chiffrement par clé gérée par le client (facultatif)
Données éphémérides utilisées pour calculer la trajectoire d'un satellite	Activé	Activé
Éphémérides d'élévation d'azimut utilisées pour commander les antennes	Activé	Activé

Note

AWS Ground Station active automatiquement le chiffrement au repos Clés détenues par AWS pour protéger gratuitement les données personnelles identifiables. Toutefois, AWS KMS des frais s'appliquent pour l'utilisation d'une clé gérée par le client. Pour plus d'informations sur les tarifs, consultez les [AWS Key Management Service tarifs](#).

Pour plus d'informations AWS KMS, consultez le [guide du AWS Key Management Service développeur](#).

Pour obtenir des informations spécifiques à chaque type de ressource, voir :

- [Chiffrement au repos pour les données d'éphémérides TLE et OEM](#)
- [Chiffrement au repos pour les éphémérides d'élévation d'azimut](#)

Création d'une clé gérée par le client

Vous pouvez créer une clé symétrique gérée par le client en utilisant le AWS Management Console, ou le AWS KMS APIs.

Pour créer une clé symétrique gérée par le client

Suivez les étapes de création d'une clé symétrique gérée par le client dans le [Guide du AWS Key Management Service développeur](#).

Aperçu des politiques clés

Les stratégies de clés contrôlent l'accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une stratégie de clé, qui contient des instructions qui déterminent les personnes pouvant utiliser la clé et comment elles peuvent l'utiliser. Lorsque vous créez votre clé gérée par le client, vous pouvez spécifier une stratégie de clé. Pour plus d'informations, consultez [la section Gestion de l'accès aux clés gérées par le client](#) dans le Guide du AWS Key Management Service développeur.

Pour utiliser votre clé gérée par le client avec AWS Ground Station des ressources, vous devez configurer la politique des clés afin d'accorder les autorisations appropriées au AWS Ground Station service. Les autorisations spécifiques et la configuration des politiques dépendent du type de ressource que vous cryptez :

- Pour les données sur les éphémérides TLE et OEM, reportez-vous à la section [Chiffrement au repos pour les données d'éphémérides TLE et OEM](#) pour obtenir des exigences politiques clés spécifiques et des exemples.
- Pour les données sur les éphémérides d'altitude azimutale, voir les exigences politiques clés spécifiques et [Chiffrement au repos pour les éphémérides d'élévation d'azimut](#) des exemples.

Note

La configuration des politiques clés diffère selon les types d'éphémérides. Les données d'éphémérides TLE et OEM utilisent des autorisations d'accès aux clés, tandis que les éphémérides d'altitude azimutale utilisent des autorisations politiques directes relatives aux clés. Assurez-vous de configurer votre politique de clés en fonction du type de ressource spécifique que vous cryptez.

Pour plus d'informations sur la [spécification des autorisations dans une politique et la résolution des problèmes d'accès par clé](#), consultez le guide du AWS Key Management Service développeur.

Spécification d'une clé gérée par le client pour AWS Ground Station

Vous pouvez spécifier une clé gérée par le client pour chiffrer les ressources suivantes :

- Éphémérides (TLE, OEM et élévation d'azimut)

Lorsque vous créez une ressource, vous pouvez spécifier la clé de données en fournissant un `kmsKeyArn`

- `kmsKeyArn`- Un [identifiant de clé](#) pour une clé gérée par AWS KMS le client

AWS Ground Station contexte de chiffrement

Un [contexte de chiffrement](#) est un ensemble facultatif de paires clé-valeur contenant des informations contextuelles supplémentaires sur les données. AWS KMS utilise le contexte de chiffrement comme données authentifiées supplémentaires pour prendre en charge le chiffrement authentifié. Lorsque vous incluez un contexte de chiffrement dans une demande de chiffrement de données, AWS KMS lie le contexte de chiffrement aux données chiffrées. Pour déchiffrer les données, vous devez inclure le même contexte de chiffrement dans la demande.

AWS Ground Station utilise un contexte de chiffrement différent en fonction de la ressource cryptée et spécifie un contexte de chiffrement spécifique pour chaque attribution de clé créée.

Pour plus de détails sur le contexte de chiffrement spécifique aux ressources, voir :

- [Chiffrement au repos pour les données d'éphémérides TLE et OEM](#)
- [Chiffrement au repos pour les éphémérides d'élévation d'azimut](#)

Chiffrement au repos pour les données d'éphémérides TLE et OEM

Principales exigences politiques relatives aux éphémérides TLE et OEM

Pour utiliser une clé gérée par le client avec des données éphémérides, votre politique en matière de clés doit accorder les autorisations suivantes au AWS Ground Station service :

- [kms:CreateGrant](#)- Crée une autorisation d'accès sur une clé gérée par le client. Accorde AWS Ground Station l'accès pour effectuer [des opérations de subvention](#) sur la clé gérée par le client pour lire et stocker des données cryptées.
- [kms:DescribeKey](#)- Fournit les informations relatives à la clé gérée par le client AWS Ground Station pour lui permettre de valider la clé avant de tenter d'utiliser la clé fournie.

Pour plus d'informations sur [l'utilisation des subventions](#), consultez le guide du AWS Key Management Service développeur.

Autorisations des utilisateurs IAM pour créer des éphémérides à l'aide de clés gérées par le client

Lorsqu'une clé gérée par le client est AWS Ground Station utilisée dans le cadre d'opérations cryptographiques, elle agit pour le compte de l'utilisateur qui crée la ressource éphéméride.

Pour créer une ressource éphéméride à l'aide d'une clé gérée par le client, un utilisateur doit être autorisé à effectuer les opérations suivantes sur la clé gérée par le client :

- [kms:CreateGrant](#)- Permet à l'utilisateur de créer des autorisations sur la clé gérée par le client au nom de AWS Ground Station.
- [kms:DescribeKey](#)- Permet à l'utilisateur de consulter les informations clés gérées par le client pour valider la clé.

Vous pouvez spécifier les autorisations requises dans une politique de clé ou dans une IAM politique si la politique de clé le permet. Ces autorisations garantissent que les utilisateurs peuvent autoriser l'utilisation AWS Ground Station de la clé gérée par le client pour les opérations de chiffrement en leur nom.

Comment AWS Ground Station utilise les subventions AWS KMS pour les éphémérides

AWS Ground Station nécessite une [autorisation de clé](#) pour utiliser votre clé gérée par le client.

Lorsque vous téléchargez une éphéméride cryptée à l'aide d'une clé gérée par le client, vous AWS Ground Station créez une attribution de clé en votre nom en envoyant une [CreateGrant](#) demande à AWS KMS. Les subventions AWS KMS sont utilisées pour donner AWS Ground Station accès à une AWS KMS clé de votre compte.

Cela permet AWS Ground Station d'effectuer les opérations suivantes :

- Appelez [GenerateDataKey](#) pour générer une clé de données chiffrée et la stocker, car la clé de données n'est pas immédiatement utilisée pour chiffrer.
- Appelez [Decrypt](#) pour utiliser la clé de données cryptée stockée afin d'accéder aux données cryptées.
- Appelez [Encrypt](#) pour utiliser la clé de données pour chiffrer les données.
- Configurer un principal sortant pour permettre au service de [RetireGrant](#).

Vous pouvez révoquer l'accès à la subvention à tout moment. Si vous le faites, vous AWS Ground Station ne pourrez accéder à aucune des données chiffrées par la clé gérée par le client, ce qui affectera les opérations qui dépendent de ces données. Par exemple, si vous supprimez une attribution de clé pour une éphéméride actuellement utilisée pour un contact, vous ne AWS Ground Station pourrez pas utiliser les données d'éphéméride fournies pour pointer l'antenne pendant le contact. Cela entraînera la fin du contact dans un état d'échec.

Contexte de chiffrement des éphémérides

Les autorisations clés pour le chiffrement des ressources d'éphémérides sont liées à un ARN satellite spécifique.

```
"encryptionContext": {
  "aws:groundstation:arn":
    "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
  "aws:s3:arn":
    "arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
}
```

Note

Les subventions clés sont réutilisées pour la même paire clé-satellite.

Utilisation du contexte de chiffrement à des fins de surveillance

Lorsque vous utilisez une clé symétrique gérée par le client pour chiffrer vos éphémérides, vous pouvez également utiliser le contexte de chiffrement dans les enregistrements d'audit et les journaux pour identifier la manière dont la clé gérée par le client est utilisée. Le contexte de chiffrement apparaît également dans [les journaux générés par AWS CloudTrail ou Amazon CloudWatch Logs](#).

Utilisation du contexte de chiffrement pour contrôler l'accès à votre clé gérée par le client

Vous pouvez utiliser le contexte de chiffrement dans les stratégies de clé et les politiques IAM comme conditions pour contrôler l'accès à votre clé symétrique gérée par le client. Vous pouvez également utiliser des contraintes de contexte de chiffrement dans un octroi.

AWS Ground Station utilise une contrainte de contexte de chiffrement dans les autorisations afin de contrôler l'accès à la clé gérée par le client dans votre compte ou votre région. La contrainte d'octroi exige que les opérations autorisées par l'octroi utilisent le contexte de chiffrement spécifié.

Vous trouverez ci-dessous des exemples d'instructions de stratégie de clé permettant d'accorder l'accès à une clé gérée par le client dans un contexte de chiffrement spécifique. La condition énoncée dans cette instruction de stratégie exige que les octrois comportent une contrainte de contexte de chiffrement qui spécifie le contexte de chiffrement.

L'exemple suivant montre une politique clé pour les données d'éphémérides liées à un satellite :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Create Grant on key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:CreateGrant",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:EncryptionContext:aws:groundstation:arn":
            "arn:aws:groundstation::123456789012:satellite/satellite-id"
        }
      }
    }
  ]
}
```

Surveillance de vos clés de chiffrement pour détecter les éphémérides

Lorsque vous utilisez une clé gérée par le AWS Key Management Service client avec vos ressources éphémérides, vous pouvez utiliser les [CloudWatch journaux AWS CloudTrailAmazon](#) pour suivre les demandes AWS Ground Station envoyées à. AWS KMS Les exemples suivants sont CloudTrail des événements pour [CreateGrantGenerateDataKey](#), [déchiffrer](#) et surveiller les AWS KMS opérations appelées [DescribeKey](#) pour accéder AWS Ground Station aux données chiffrées par votre clé gérée par le client.

CreateGrant

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources éphémérides, envoyez une AWS Ground Station [CreateGrant](#) demande en votre nom pour accéder à la AWS KMS clé de votre compte. AWS L'autorisation AWS Ground Station créée est spécifique à la ressource associée à la clé gérée par le AWS KMS client. En outre, AWS Ground Station utilise l'[RetireGrant](#) opération pour supprimer une subvention lorsque vous supprimez une ressource.

L'exemple d'événement suivant enregistre l'[CreateGrant](#) opération d'une éphéméride :

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2022-02-22T22:22:22Z",
        "mfaAuthenticated": "false"
      }
    }
  },
}
```

```

    "invokedBy": "AWS Internal"
  },
  "eventTime": "2022-02-22T22:22:22Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "CreateGrant",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "AWS Internal",
  "userAgent": "ExampleDesktop/1.0 (V1; OS)",
  "requestParameters": {
    "operations": [
      "GenerateDataKeyWithoutPlaintext",
      "Decrypt",
      "Encrypt"
    ],
    "constraints": {
      "encryptionContextSubset": {
        "aws:groundstation:arn":
"arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
      }
    },
    "granteePrincipal": "groundstation.us-west-2.amazonaws.com",
    "retiringPrincipal": "groundstation.us-west-2.amazonaws.com",
    "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  },
  "responseElements": {
    "grantId":
"0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
  },
  "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "readOnly": false,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"

```

```
}

```

DescribeKey

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources éphémères, vous AWS Ground Station envoyez une [DescribeKey](#) demande en votre nom pour valider que la clé demandée existe dans votre compte.

L'exemple d'événement suivant enregistre l'opération [DescribeKey](#) :

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/User/Role",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Role",
        "accountId": "111122223333",
        "userName": "User"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2022-02-22T22:22:22Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "invokedBy": "AWS Internal"
},
"eventTime": "2022-02-22T22:22:22Z",
"eventSource": "kms.amazonaws.com",
"eventName": "DescribeKey",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "AWS Internal",
"requestParameters": {
  "keyId": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
}
```

```

},
"responseElements": null,
"requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}

```

GenerateDataKey

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources éphémères, AWS Ground Station envoie une [GenerateDataKey](#) demande à afin de générer une clé de données avec laquelle chiffrer vos données.

L'exemple d'événement suivant enregistre l'[GenerateDataKey](#) opération d'une éphémère :

```

{
  "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

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources d'éphémérides, utilisez l'opération de AWS Ground Station déchiffrement pour [déchiffrer](#) les éphémérides fournies si elles sont déjà chiffrées avec la même clé gérée par le client. Par exemple, si une éphéméride est téléchargée depuis un compartiment S3 et qu'elle est chiffrée dans ce compartiment avec une clé donnée.

L'exemple d'événement suivant enregistre l'opération de [déchiffrement](#) d'une éphéméride :

```

{
  "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"
}

```

Chiffrement au repos pour les éphémérides d'élévation d'azimut

Principales exigences politiques relatives aux éphémérides d'altitude azimutale

Pour utiliser une clé gérée par le client avec des données d'éphémérides d'altitude azimutale, votre politique en matière de clés doit accorder les autorisations suivantes au service. AWS Ground Station Contrairement aux données d'éphémérides TLE et OEM qui utilisent des autorisations, les

éphémérides à élévation azimutale utilisent des autorisations politiques directes pour les opérations de chiffrement. Il s'agit d'une méthode plus simple pour gérer les autorisations et utiliser vos clés.

- [kms:GenerateDataKey](#)- Génère des clés de données pour chiffrer vos données d'éphémérides d'altitude azimutale.
- [kms:Decrypt](#)- Déchiffre les clés de données cryptées lorsque vous accédez à vos données d'éphémérides d'altitude azimutale.

Exemple de politique clé accordant l' AWS Ground Station accès à une clé gérée par le client

Note

Dans le cas des éphémérides d'altitude azimutale, vous devez configurer ces autorisations directement dans la politique clé. Le directeur du AWS Ground Station service régional (par exemple `groundstation.region.amazonaws.com`) doit obtenir ces autorisations dans vos principales déclarations de politique. Si ces déclarations ne sont pas ajoutées à la politique clé, vous ne pouvez pas stocker ou accéder à vos éphémérides d'élévation d'azimut personnalisées.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      }
    }
  ]
}
```

```
    },
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": "*"
}
]
```

Autorisations des utilisateurs IAM pour créer des éphémérides d'altitude azimutale à l'aide de clés gérées par le client

Lorsqu'une clé gérée par le client est AWS Ground Station utilisée dans le cadre d'opérations cryptographiques, elle agit pour le compte de l'utilisateur qui crée la ressource d'éphémérides d'altitude azimutale.

Pour créer une ressource d'éphémérides d'altitude azimutale à l'aide d'une clé gérée par le client, un utilisateur doit être autorisé à effectuer les opérations suivantes sur la clé gérée par le client :

- [kms:GenerateDataKey](#)- Permet à l'utilisateur de générer des clés de données pour chiffrer les données d'éphémérides d'altitude azimutale.
- [kms:Decrypt](#)- Permet à l'utilisateur de déchiffrer les clés de données lorsqu'il accède aux données des éphémérides d'altitude azimutale.
- [kms:DescribeKey](#)- Permet à l'utilisateur de consulter les informations relatives aux clés gérées par le client afin de valider la clé.

Vous pouvez spécifier les autorisations requises dans une politique de clé ou dans une IAM politique si la politique de clé le permet. Ces autorisations garantissent que les utilisateurs peuvent autoriser l'utilisation AWS Ground Station de la clé gérée par le client pour les opérations de chiffrement en leur nom.

Comment AWS Ground Station utilise les politiques clés pour les éphémérides d'altitude azimutale

Lorsque vous fournissez des données d'éphémérides d'altitude azimutale à l'aide d'une clé gérée par le client, utilisez AWS Ground Station des politiques clés pour accéder à votre clé de chiffrement. Les autorisations sont accordées directement par le AWS Ground Station biais de déclarations politiques

clés plutôt que par le biais de subventions, comme c'est le cas pour les données d'éphémérides TLE ou OEM.

Si vous supprimez AWS Ground Station l'accès à la clé gérée par le client, vous AWS Ground Station ne pourrez accéder à aucune des données chiffrées par cette clé, ce qui affectera les opérations qui dépendent de ces données. Par exemple, si vous supprimez des autorisations politiques clés pour les éphémérides d'azimut actuellement utilisées pour un contact, vous ne AWS Ground Station pourrez pas utiliser les données d'altitude d'azimut fournies pour commander l'antenne pendant le contact. Cela entraînera la fin du contact dans un état d'échec.

Contexte de chiffrement des éphémérides d'élévation d'azimut

[Lorsqu'il AWS Ground Station utilise votre AWS KMS clé pour chiffrer des données d'éphémérides d'altitude azimutale, le service spécifie un contexte de chiffrement.](#) Le contexte de chiffrement est constitué de données authentifiées supplémentaires (AAD) AWS KMS utilisées pour garantir l'intégrité des données. Autrement dit, quand un contexte de chiffrement est spécifié pour une opération de chiffrement, le service doit spécifier le même contexte de chiffrement pour l'opération de déchiffrement. Dans le cas contraire, le déchiffrement échoue. Le contexte de chiffrement est également écrit dans vos CloudTrail journaux pour vous aider à comprendre pourquoi une AWS KMS clé donnée a été utilisée. Vos CloudTrail journaux peuvent contenir de nombreuses entrées décrivant l'utilisation d'une AWS KMS clé, mais le contexte de chiffrement de chaque entrée de journal peut vous aider à déterminer la raison de cette utilisation particulière.

AWS Ground Station spécifie le contexte de chiffrement suivant lorsqu'il effectue des opérations cryptographiques avec votre clé gérée par le client sur une éphéméride d'altitude azimutale :

```
{
  "encryptionContext": {
    "aws:groundstation:ground-station-id": "Ohio 1",
    "aws:groundstation:arn": "arn:aws:groundstation:us-
east-2:111122223333:ephemeris/00a770b0-082d-45a4-80ed-SAMPLE",
    "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/00a770b0-082d-45a4-80ed-
SAMPLE/raw"
  }
}
```

Le contexte de chiffrement contient :

`aws:groundstation:ground-station-id`

Le nom de la station au sol associée à l'éphéméride de l'azimut.

aws:ground station : arn

L'ARN de la ressource éphéméride.

aws:s3:arn

L'ARN des éphémérides stockées dans Amazon S3.

Contrôle de l'accès à votre clé gérée par le client à l'aide du contexte de chiffrement

Vous pouvez utiliser les instructions de condition IAM pour contrôler l' AWS Ground Station accès à votre clé gérée par le client. L'ajout d'une déclaration de condition sur les kms :Decrypt actions kms :GenerateDataKey et limite les stations au sol pour lesquelles a AWS KMS peut être utilisée.

Vous trouverez ci-dessous des exemples de déclarations de politique clés visant à accorder AWS Ground Station l'accès à votre clé gérée par le client dans une région spécifique pour une station au sol spécifique. La condition de cette déclaration de politique exige que tous cryptent et déchiffrent l'accès à la clé qui spécifie un contexte de chiffrement correspondant à la condition de la politique de clé.

Exemple de politique de clé accordant l' AWS Ground Station accès à une clé gérée par le client pour une station au sol spécifique

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
```

```

        "Service": "groundstation.us-east-1.amazonaws.com"
    },
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:EncryptionContext:aws:groundstation:ground-station-id":
                "specific-ground-station-name"
        }
    }
}

```

Exemple de politique clé accordant l' AWS Ground Station accès à une clé gérée par le client pour plusieurs stations au sol

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": [

```

```

        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": "*",
    "Condition": {
        "StringLike": {
            "kms:EncryptionContext:aws:groundstation:ground-station-id":
[
                "specific-ground-station-name-1",
                "specific-ground-station-name-2"
            ]
        }
    }
}
]
}
}

```

Surveillance de vos clés de chiffrement pour détecter les éphémérides d'élévation azimutale

Lorsque vous utilisez une clé gérée par le AWS KMS client avec vos ressources d'éphémérides d'altitude azimutale, vous pouvez utiliser [CloudTrail](#) ou enregistrer pour suivre [CloudWatch les demandes](#) envoyées à. AWS Ground Station AWS KMS Les exemples suivants sont des CloudTrail événements pour [GenerateDataKey](#) et [Decrypt destinés](#) à surveiller les AWS KMS opérations appelées AWS Ground Station pour accéder aux données chiffrées par votre clé gérée par le client.

GenerateDataKey

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources d'éphémérides d'altitude azimutale, vous envoyez une AWS Ground Station [GenerateDataKey](#) demande AWS KMS à afin de générer une clé de données avec laquelle chiffrer vos données.

L'exemple d'événement suivant enregistre l'[GenerateDataKey](#) opération pour les éphémérides d'altitude azimutale :

```

{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",

```

```

    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "attributes": {
        "creationDate": "2025-08-25T14:45:48Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "AWS Internal"
  },
  "eventTime": "2025-08-25T14:52:02Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "AWS Internal",
  "userAgent": "AWS Internal",
  "requestParameters": {
    "keySpec": "AES_256",
    "encryptionContext": {
      "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
      "aws:groundstation:ground-station-id": "Ohio 1",
      "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
    },
    "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  },
  "responseElements": null,
  "requestID": "ef6f9a8f-8ef6-46a1-bdcb-123456SAMPLE",
  "eventID": "952842d4-1389-3232-b885-123456SAMPLE",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",

```

```

      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "sharedEventID": "8424f6b6-2280-4d1d-b9fd-0348b1546cba",
  "eventCategory": "Management"
}

```

Decrypt

Lorsque vous utilisez une clé gérée par le AWS KMS client pour chiffrer vos ressources d'éphémérides d'altitude d'azimut, AWS Ground Station utilise l'opération de déchiffrement pour [déchiffrer](#) les données d'éphémérides d'altitude d'azimut fournies si elles sont déjà chiffrées avec la même clé gérée par le client.

L'exemple d'événement suivant enregistre l'opération de [déchiffrement pour les éphémérides d'altitude azimutale](#) :

```

{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      }
    }
  },
  "attributes": {
    "creationDate": "2025-08-25T14:45:48Z",
    "mfaAuthenticated": "false"
  }
},

```

```
"invokedBy": "AWS Internal",
"eventTime": "2025-08-25T14:54:01Z",
"eventSource": "kms.amazonaws.com",
"eventName": "Decrypt",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "AWS Internal",
"requestParameters": {
  "encryptionContext": {
    "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
    "aws:groundstation:ground-station-id": "Ohio 1",
    "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
  },
  "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
},
"responseElements": null,
"requestID": "a2f46066-49fb-461a-93cb-123456SAMPLE",
"eventID": "e997b426-e3ad-31c7-a308-123456SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"sharedEventID": "477b568e-7f56-4f04-905c-623ff146f30d",
"eventCategory": "Management"
}
```

Chiffrement des données pendant le transit pour AWS Ground Station

AWS Ground Station fournit un chiffrement par défaut pour protéger vos données sensibles pendant le transport. Les données peuvent être diffusées entre les emplacements des AWS Ground Station

antennes et vos instances Amazon EC2 de deux manières, en fonction de la configuration du profil de mission.

- AWS Ground Station Agent
- Point de terminaison du flux de données

Chaque méthode de diffusion de données gère différemment le chiffrement des données en transit. Les sections suivantes décrivent chaque méthode.

AWS Ground Station Streams d'agents

AWS Ground Station L'agent chiffre ses flux à l'aide de AWS KMS clés gérées par le client. L' AWS Ground Station agent exécuté sur votre instance Amazon EC2 déchiffre automatiquement le flux pour fournir des données déchiffrées.

La AWS KMS clé utilisée pour chiffrer un flux est spécifiée lors de la création d'un `MissionProfile` dans le `streamsKmsKey` paramètre. Toutes les autorisations accordant l' AWS Ground Station accès aux clés sont gérées par le biais de la politique relative aux AWS KMS clés attachée à `streamsKmsKey`.

Flux de points de terminaison de flux de données

Les flux de points de terminaison de flux de données sont chiffrés à l'aide du protocole DTLS ([Datagram Transport Layer Security](#)). Cela se fait à l'aide de certificats auto-signés et ne nécessite aucune configuration supplémentaire.

Exemples de configurations de profil de mission

Les exemples fournis montrent comment utiliser un satellite de diffusion public et créer un profil de mission qui le soutient. Les modèles qui en résultent sont fournis pour vous aider à prendre contact avec un satellite de diffusion publique et pour vous aider à prendre des décisions concernant vos satellites.

Rubriques

- [JPSS-1 - Satellite de diffusion public \(PBS\) - Évaluation](#)
- [Satellite de diffusion publique utilisant la livraison de données Amazon S3](#)
- [Satellite de diffusion public utilisant un point de terminaison de flux de données \(bande étroite\)](#)
- [Satellite de diffusion public utilisant un point de terminaison de flux de données \(démodulé et décodé\)](#)
- [Satellite de diffusion publique utilisant AWS Ground Station l'agent \(large bande\)](#)

JPSS-1 - Satellite de diffusion public (PBS) - Évaluation

Cette section d'exemple correspond au [Vue d'ensemble du processus d'intégration des clients](#). Il fournit une brève analyse de compatibilité avec AWS Ground Station les exemples spécifiques suivants et prépare le terrain pour les exemples spécifiques qui suivent.

Comme indiqué dans la [Satellites de diffusion publics](#) section, vous pouvez utiliser certains satellites, ou les voies de communication d'un satellite, accessibles au public. Dans cette section, nous décrivons [JPSS-1](#) en termes. AWS Ground Station À titre de référence, nous utilisons le [document de contrôle de l'interface radiofréquence \(ICD\) \(ICD\) des données à haut débit \(HRD\) du Joint Polar Satellite System 1 \(JPSS-1\) pour les stations de diffusion directe \(DBS\)](#). Il convient également de noter que le JPSS-1 est associé à l'ID 43013 du NORAD.

Le satellite JPSS-1 propose une voie de communication en liaison montante et trois voies de communication en liaison descendante directe, comme le montre la figure 1-1 de l'ICD. Parmi ces quatre voies de communication, seule la voie de communication descendante de données à haut débit (HRD) est disponible pour la consommation publique. Sur cette base, vous verrez que ce chemin sera également associé à des données beaucoup plus spécifiques. Les quatre voies sont les suivantes :

- Chemin de commande (liaison montante) à une fréquence MHz centrale de 2067,27 avec un débit de données de 2 à 128 kbit/s. Ce chemin n'est pas accessible au public.
- Chemin de télémétrie (liaison descendante) à une fréquence MHz centrale de 2247,5 avec un débit de données de 1 à 524 kbps. Ce chemin n'est pas accessible au public.
- Chemin SMD (liaison descendante) à une fréquence GHz centrale de 26,7034 avec un débit de 150 à 300 Mbps. Ce chemin n'est pas accessible au public.
- La RF pour le trajet HRD (liaison descendante) à une fréquence MHz centrale de 7812 avec un débit de données de 15 Mbps. Il a une MHz bande passante de 30, et c'est le cas right-hand-circular-polarized. Lorsque vous embarquez à bord du JPSS-1 AWS Ground Station, c'est le chemin de communication auquel vous avez accès. Ce chemin de communication contient des données scientifiques sur les instruments, des données d'ingénierie des instruments, des données de télémétrie des instruments et des données d'entretien en temps réel des engins spatiaux.

Lorsque nous comparons les chemins de données potentiels, nous constatons que les chemins de commande (liaison montante), de télémétrie (liaison descendante) et HRD (liaison descendante) répondent aux capacités de fréquence, de bande passante et d'utilisation simultanée multicanaux de AWS Ground Station. Le chemin SMD n'est pas compatible car la fréquence centrale est hors de portée des récepteurs existants. Pour plus d'informations sur les fonctionnalités prises en charge, consultez [AWS Ground Station Fonctionnalités du site](#).

Note

Comme le chemin SMD n'est pas compatible avec AWS Ground Station celui-ci, il ne sera pas représenté dans les exemples de configurations.

Note

Comme les chemins de commande (liaison montante) et de télémétrie (liaison descendante) ne sont pas définis dans l'ICD et ne sont pas disponibles pour un usage public, les valeurs fournies lors de leur utilisation sont fictives.

Satellite de diffusion publique utilisant la livraison de données Amazon S3

Cet exemple s'appuie sur l'analyse effectuée dans la [JPSS-1 - Satellite de diffusion public \(PBS\) - Évaluation](#) section du guide de l'utilisateur.

Pour cet exemple, vous devez supposer un scénario : vous souhaitez capturer le chemin de communication HRD sous forme de fréquence intermédiaire numérique et le stocker pour un futur traitement par lots. Cela permet d'économiser les échantillons bruts en quadrature de phase (I/Q) de radiofréquence (RF) une fois qu'ils ont été numérisés. Une fois que les données se trouvent dans votre compartiment Amazon S3, vous pouvez les démoduler et les décoder à l'aide du logiciel de votre choix. Consultez le [MathWorks didacticiel](#) pour un exemple détaillé de traitement. Après avoir utilisé cet exemple, vous pouvez envisager d'ajouter des composants de tarification au EC2 comptant d'Amazon pour traiter les données et réduire vos coûts de traitement globaux.

Voies de communication

Cette section représente [Planifiez les voies de communication de votre flux de données](#) la mise en route.

Tous les extraits de modèle suivants appartiennent à la section Ressources du CloudFormation modèle.

Resources :

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

Note

Pour plus d'informations sur le contenu d'un CloudFormation modèle, consultez les [sections relatives aux modèles](#).

Compte tenu de notre scénario consistant à fournir un chemin de communication unique à Amazon S3, vous savez que vous n'aurez qu'un seul chemin de livraison asynchrone. Selon la [Livraison de données asynchrone](#) section, vous devez définir un compartiment Amazon S3.

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

En outre, vous devrez créer les rôles et les politiques appropriés afin d' AWS Ground Station autoriser l'utilisation du bucket.

```
# The IAM role that AWS Ground Station will assume to have permission find and write
# data to your S3 bucket.
GroundStationS3DataDeliveryRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Action:
            - 'sts:AssumeRole'
          Effect: Allow
          Principal:
            Service:
              - groundstation.amazonaws.com
          Condition:
            StringEquals:
              "aws:SourceAccount": !Ref AWS::AccountId
            ArnLike:
              "aws:SourceArn": !Sub "arn:aws:groundstation:${AWS::Region}:
${AWS::AccountId}:config/s3-recording/*"

# The S3 bucket policy that defines what actions AWS Ground Station can perform on
your S3 bucket.
GroundStationS3DataDeliveryBucketPolicy:
  Type: AWS::IAM::Policy
  Properties:
    PolicyDocument:
```

```

Statement:
  - Action:
    - 's3:GetBucketLocation'
    Effect: Allow
    Resource:
      - !GetAtt GroundStationS3DataDeliveryBucket.Arn
  - Action:
    - 's3:PutObject'
    Effect: Allow
    Resource:
      - !Join [ "/", [ !GetAtt GroundStationS3DataDeliveryBucket.Arn, "*" ] ]
PolicyName: GroundStationS3DataDeliveryPolicy
Roles:
  - !Ref GroundStationS3DataDeliveryRole

```

AWS Ground Station configurations

Cette section représente [Création de configurations](#) la mise en route.

Vous aurez besoin d'une configuration de suivi pour définir vos préférences en matière d'utilisation du suivi automatique. La sélection de PREFERRED comme autotrack peut améliorer la qualité du signal, mais elle n'est pas obligatoire pour atteindre la qualité du signal en raison de la qualité suffisante des éphémérides JPSS-1.

```

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

```

En fonction du chemin de communication, vous devrez définir une configuration antenne-liaison descendante pour représenter la partie satellite, ainsi qu'un enregistrement S3 pour faire référence au compartiment Amazon S3 que vous venez de créer.

```

# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to

```

```
# downlink data from your satellite.
JpssDownlinkDigIfAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Downlink DigIF Antenna Config"
    ConfigData:
      AntennaDownlinkConfig:
        SpectrumConfig:
          Bandwidth:
            Units: "MHz"
            Value: 30
          CenterFrequency:
            Units: "MHz"
            Value: 7812
          Polarization: "RIGHT_HAND"

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

AWS Ground Station profil de mission

Cette section représente [Création d'un profil de mission](#) la mise en route.

Maintenant que vous disposez des configurations associées, vous pouvez les utiliser pour créer le flux de données. Vous utiliserez les valeurs par défaut pour les autres paramètres.

```
# 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
```

Assemblage

Grâce aux ressources ci-dessus, vous avez désormais la possibilité de planifier des contacts JPSS-1 pour la livraison de données asynchrones à partir de n'importe lequel de vos contacts intégrés. AWS Ground Station [AWS Ground Station Succursales](#)

Ce qui suit est un CloudFormation modèle complet qui inclut toutes les ressources décrites dans cette section combinées dans un modèle unique qui peut être directement utilisé dans CloudFormation.

Le CloudFormation modèle nommé `AquaSnppJpss-1TerraDigIfS3DataDelivery.yml` contient un compartiment Amazon S3 et les AWS Ground Station ressources nécessaires pour planifier les contacts et recevoir les données de diffusion directe du signal VITA-49/IP.

Si Aqua, SNPP, JPSS-1/NOAA-20 et Terra ne sont pas intégrés à votre compte, consultez. [satellite embarqué](#)

Note

Vous pouvez accéder au modèle en accédant au compartiment Amazon S3 du client qui intègre le compartiment Amazon S3 à l'aide AWS d'informations d'identification valides. Les liens ci-dessous utilisent un compartiment Amazon S3 régional. Modifiez le code de `us-west-2` région pour représenter la région correspondante dans laquelle vous souhaitez créer la CloudFormation pile.

En outre, les instructions suivantes utilisent le langage YAML. Cependant, les modèles sont disponibles au format YAML et JSON. Pour utiliser le format JSON, remplacez l'extension de `.yml` fichier par `.json` lors du téléchargement du modèle.

Pour télécharger le modèle à l'aide de AWS CLI, utilisez la commande suivante :

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

Vous pouvez consulter le modèle et le télécharger dans la console en entrant l'URL suivante dans votre navigateur :

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

Vous pouvez spécifier le modèle directement en CloudFormation utilisant le lien suivant :

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

Satellite de diffusion public utilisant un point de terminaison de flux de données (bande étroite)

Cet exemple s'appuie sur l'analyse effectuée dans la [JPSS-1 - Satellite de diffusion public \(PBS\) - Évaluation](#) section du guide de l'utilisateur.

Pour compléter cet exemple, vous devez supposer un scénario : vous souhaitez capturer le chemin de communication HRD sous forme de fréquence intermédiaire numérique (DigIF) et le traiter tel qu'il est reçu par une application de point de terminaison de flux de données sur une instance EC2 Amazon à l'aide d'un SDR.

Voies de communication

Cette section représente [Planifiez les voies de communication de votre flux de données](#) la mise en route. Dans cet exemple, vous allez créer deux sections dans votre CloudFormation modèle : les sections Paramètres et Ressources.

Note

Pour plus d'informations sur le contenu d'un CloudFormation modèle, consultez les [sections relatives aux modèles](#).

Pour la section Paramètres, vous allez ajouter les paramètres suivants. Vous spécifierez leurs valeurs lors de la création de la pile via la CloudFormation console.

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

Vous devez créer une paire de clés et fournir le nom du EC2 EC2Key paramètre Amazon. Consultez la section [Créer une paire de clés pour votre EC2 instance Amazon](#). En outre, vous devrez fournir le bon ID d'AMI spécifique à la région lors de la création de la CloudFormation pile. Consultez [AWS Ground Station Images de machines Amazon \(AMIs\)](#).

Les autres extraits de modèle se trouvent dans la section Ressources du CloudFormation modèle.

Resources:

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

Dans notre scénario consistant à fournir un chemin de communication unique à une EC2 instance, vous disposerez d'un seul chemin de diffusion synchrone. Selon [Livraison synchrone des données](#) cette section, vous devez configurer une EC2 instance Amazon avec une application de point de terminaison de flux de données, et créer un ou plusieurs groupes de points de terminaison de flux de données.

```

# 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 subnet 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

En outre, vous devrez également créer les politiques et les rôles appropriés pour permettre de AWS Ground Station créer une Elastic Network Interface (ENI) dans votre compte.

```
# 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 configurations

Cette section représente [Création de configurations](#) la mise en route.

Vous aurez besoin d'une configuration de suivi pour définir vos préférences en matière d'utilisation du suivi automatique. La sélection de PREFERRED comme autotrack peut améliorer la qualité du signal, mais elle n'est pas obligatoire pour atteindre la qualité du signal en raison de la qualité suffisante des éphémérides JPSS-1.

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

Sur la base du chemin de communication, vous devez définir une configuration antenne-liaison descendante pour représenter la partie satellite, ainsi qu'une configuration de point de terminaison

de flux de données pour faire référence au groupe de points de terminaison de flux de données qui définit les détails du point de terminaison.

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

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

AWS Ground Station profil de mission

Cette section représente [Création d'un profil de mission](#) la mise en route.

Maintenant que vous disposez des configurations associées, vous pouvez les utiliser pour créer le flux de données. Vous utiliserez les valeurs par défaut pour les autres paramètres.

```
# 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
```

Assemblage

Grâce aux ressources ci-dessus, vous avez désormais la possibilité de planifier des contacts JPSS-1 pour une livraison synchrone des données depuis n'importe lequel de vos contacts intégrés. AWS Ground Station [AWS Ground Station Succursales](#)

Ce qui suit est un CloudFormation modèle complet qui inclut toutes les ressources décrites dans cette section combinées dans un modèle unique qui peut être directement utilisé dans CloudFormation.

Le CloudFormation modèle nommé `AquaSnppJpssTerraDigIF.yml` est conçu pour vous permettre de commencer rapidement à recevoir des données de fréquence intermédiaire numérisées (DigIF) pour les satellites Aqua, SNPP, JPSS-1/NOAA-20 et Terra. Il contient une EC2 instance Amazon et les CloudFormation ressources nécessaires pour recevoir les données brutes de diffusion directe de DigiF.

Si Aqua, SNPP, JPSS-1/NOAA-20 et Terra ne sont pas intégrés à votre compte, consultez. [satellite embarqué](#)

Note

Vous pouvez accéder au modèle en accédant au compartiment Amazon S3 du client qui intègre le compartiment Amazon S3 à l'aide AWS d'informations d'identification valides. Les liens ci-dessous utilisent un compartiment Amazon S3 régional. Modifiez le code de us -

west-2 région pour représenter la région correspondante dans laquelle vous souhaitez créer la CloudFormation pile.

En outre, les instructions suivantes utilisent le langage YAML. Cependant, les modèles sont disponibles au format YAML et JSON. Pour utiliser le format JSON, remplacez l'extension de .yaml fichier par .json lors du téléchargement du modèle.

Pour télécharger le modèle à l'aide de AWS CLI, utilisez la commande suivante :

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

Vous pouvez consulter le modèle et le télécharger dans la console en entrant l'URL suivante dans votre navigateur :

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

Vous pouvez définir le modèle directement en CloudFormation utilisant le lien suivant :

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

Quelles ressources supplémentaires le modèle définit-il ?

Le AquaSnppJpssTerraDigIF modèle inclut les ressources supplémentaires suivantes :

- (Facultatif) Déclencheurs d'CloudWatch événements : AWS Lambda fonction déclenchée à l'aide d' CloudWatch événements envoyés AWS Ground Station avant et après un contact. La AWS Lambda fonction démarrera et arrêtera éventuellement votre instance de réception.
- (Facultatif) EC2 Vérification des contacts - Possibilité d'utiliser Lambda pour configurer un système de vérification de vos EC2 instances Amazon pour les contacts avec notification SNS. Il est important de noter que cela peut entraîner des frais en fonction de votre utilisation actuelle.
- Ground Station Amazon Machine Image Retrieval Lambda - La possibilité de sélectionner le logiciel installé sur votre instance et l'AMI de votre choix. Les options du logiciel incluent DDX 2.6.2 Only et DDX 2.6.2 with qRadio 3.6.0. Ces options continueront de s'étendre à mesure que des mises à jour logicielles et des fonctionnalités supplémentaires seront publiées.

- Profils de mission supplémentaires - Profils de mission pour d'autres satellites de diffusion publique (Aqua, SNPP et Terra).
- Configurations supplémentaires de liaison descendante d'antenne - Configurations de liaison descendante d'antenne pour des satellites de diffusion publics supplémentaires (Aqua, SNPP et Terra).

Les valeurs et paramètres pour les satellites dans ce modèle sont déjà renseignés. Ces paramètres vous permettent d'utiliser facilement et AWS Ground Station immédiatement ces satellites. Vous n'avez pas besoin de configurer vos propres valeurs pour pouvoir les utiliser AWS Ground Station lors de l'utilisation de ce modèle. Toutefois, vous pouvez personnaliser les valeurs pour que le modèle fonctionne selon votre cas d'utilisation.

Où puis-je recevoir mes données ?

Le groupe de points de terminaison de flux de données est configuré pour utiliser l'interface réseau d'instance de récepteur créée dans le cadre du modèle. L'instance de réception utilise une application de point de terminaison de flux de données pour recevoir le flux de données depuis le port défini par AWS Ground Station le point de terminaison du flux de données. Une fois reçues, les données sont disponibles pour la consommation via le port UDP 50000 sur l'adaptateur de bouclage de l'instance du récepteur. Pour plus d'informations sur la configuration d'un groupe de points de terminaison de flux de données, consultez. [AWS::GroundStation::DataflowEndpointGroup](#)

Satellite de diffusion public utilisant un point de terminaison de flux de données (démodulé et décodé)

Cet exemple s'appuie sur l'analyse effectuée dans la [JPSS-1 - Satellite de diffusion public \(PBS\) - Évaluation](#) section du guide de l'utilisateur.

Pour compléter cet exemple, vous devez supposer un scénario : vous souhaitez capturer le chemin de communication HRD sous forme de données de diffusion directe démodulées et décodées à l'aide d'un point de terminaison de flux de données. Cet exemple constitue un bon point de départ si vous envisagez de traiter les données à l'aide du logiciel NASA Direct Readout Labs (RT-STPS et IPOPP).

Voies de communication

Cette section représente [Planifiez les voies de communication de votre flux de données](#) la mise en route. Dans cet exemple, vous allez créer deux sections dans votre CloudFormation modèle : les sections Paramètres et Ressources.

Note

Pour plus d'informations sur le contenu d'un CloudFormation modèle, consultez les [sections relatives aux modèles](#).

Pour la section Paramètres, vous allez ajouter les paramètres suivants. Vous spécifierez leurs valeurs lors de la création de la pile via la CloudFormation console.

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

Vous devez créer une paire de clés et fournir le nom du EC2 EC2Key paramètre Amazon. Consultez la section [Créer une paire de clés pour votre EC2 instance Amazon](#). En outre, vous devrez fournir le bon ID d'AMI spécifique à la région lors de la création de la CloudFormation pile. Consultez [AWS Ground Station Images de machines Amazon \(AMIs\)](#).

Les autres extraits de modèle se trouvent dans la section Ressources du CloudFormation modèle.

Resources:

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

Dans notre scénario consistant à fournir un chemin de communication unique à une EC2 instance, vous disposerez d'un seul chemin de diffusion synchrone. Selon [Livraison synchrone des données](#) cette section, vous devez configurer une EC2 instance Amazon avec une application de point de terminaison de flux de données, et créer un ou plusieurs groupes de points de terminaison de flux de données.

```
# 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

```

Vous aurez également besoin des politiques, des rôles et des profils appropriés AWS Ground Station pour créer une Elastic Network Interface (ENI) dans votre compte.

```

# 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 configurations

Cette section représente le guide [Création de configurations](#) de l'utilisateur.

Vous aurez besoin d'une configuration de suivi pour définir vos préférences en matière d'utilisation du suivi automatique. La sélection de PREFERRED comme autotrack peut améliorer la qualité du signal, mais elle n'est pas obligatoire pour atteindre la qualité du signal en raison de la qualité suffisante des éphémérides JPSS-1.

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

```
TrackingConfig:
  Autotrack: "PREFERRED"
```

En fonction du chemin de communication, vous devez définir une antenna-downlink-demod-decodeconfiguration pour représenter la partie satellite, ainsi qu'une configuration de point de terminaison de flux de données pour faire référence au groupe de points de terminaison de flux de données qui définit les détails du point de terminaison.

Note

Pour plus de détails sur la façon de définir les valeurs pour `DemodulationConfig`, `etDecodeConfig`, veuillez consulter [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#).

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

```

        "units":"MHz"
    },
    "range":{
        "value":250,
        "units":"kHz"
    }
},
"symbolTimingRecovery":{
    "symbolRate":{
        "value":15,
        "units":"Msps"
    },
    "range":{
        "value":0.75,
        "units":"ksps"
    },
    "matchedFilter":{
        "type":"ROOT_RAISED_COSINE",
        "rolloffFactor":0.5
    }
}
}
}'
DecodeConfig:
  UnvalidatedJSON: '{
    "edges":[
      {
        "from":"I-Ingress",
        "to":"IQ-Recombiner"
      },
      {
        "from":"Q-Ingress",
        "to":"IQ-Recombiner"
      },
      {
        "from":"IQ-Recombiner",
        "to":"CcsdsViterbiDecoder"
      },
      {
        "from":"CcsdsViterbiDecoder",
        "to":"NrzmDecoder"
      },
      {
        "from":"NrzmDecoder",

```

```

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

```

```

# The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
# from your satellite.
DownlinkDemodDecodeEndpointConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "Aqua SNPP JPSS Downlink Demod Decode Endpoint Config"

```

```

ConfigData:
  DataflowEndpointConfig:
    DataflowEndpointName: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ]
    DataflowEndpointRegion: !Ref AWS::Region

```

AWS Ground Station profil de mission

Cette section représente le guide [Création d'un profil de mission](#) de l'utilisateur.

Maintenant que vous disposez des configurations associées, vous pouvez les utiliser pour créer le flux de données. Vous utiliserez les valeurs par défaut pour les autres paramètres.

```

# 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

```

Assemblage

Grâce aux ressources ci-dessus, vous avez désormais la possibilité de planifier des contacts JPSS-1 pour une livraison synchrone des données depuis n'importe quel appareil intégré. AWS Ground Station [AWS Ground Station Succursales](#)

Ce qui suit est un CloudFormation modèle complet qui inclut toutes les ressources décrites dans cette section combinées dans un modèle unique qui peut être directement utilisé dans CloudFormation.

Le CloudFormation modèle nommé `AquaSnppJpss.yml` est conçu pour vous permettre d'accéder rapidement pour commencer à recevoir des données pour les satellites Aqua, SNPP et JPSS-1/NOAA-20. Il contient une EC2 instance Amazon et les AWS Ground Station ressources nécessaires pour planifier les contacts et recevoir des données de diffusion directe démodulées et décodées.

Si Aqua, SNPP, JPSS-1/NOAA-20 et Terra ne sont pas intégrés à votre compte, consultez. [satellite embarqué](#)

Note

Vous pouvez accéder au modèle en accédant au compartiment Amazon S3 du client qui intègre le compartiment Amazon S3 à l'aide AWS d'informations d'identification valides. Les liens ci-dessous utilisent un compartiment Amazon S3 régional. Modifiez le code de `us-west-2` région pour représenter la région correspondante dans laquelle vous souhaitez créer la CloudFormation pile.

En outre, les instructions suivantes utilisent le langage YAML. Cependant, les modèles sont disponibles au format YAML et JSON. Pour utiliser le format JSON, remplacez l'extension de `.yml` fichier par `.json` lors du téléchargement du modèle.

Pour télécharger le modèle à l'aide de AWS CLI, utilisez la commande suivante :

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

Vous pouvez consulter le modèle et le télécharger dans la console en entrant l'URL suivante dans votre navigateur :

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

Vous pouvez définir le modèle directement en CloudFormation utilisant le lien suivant :

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

Quelles ressources supplémentaires le modèle définit-il ?

Le AquaSnppJpss modèle inclut les ressources supplémentaires suivantes :

- (Facultatif) Déclencheurs d'CloudWatch événements : AWS Lambda fonction déclenchée à l'aide d' CloudWatch événements envoyés AWS Ground Station avant et après un contact. La AWS Lambda fonction démarrera et arrêtera éventuellement votre instance de réception.
- (Facultatif) EC2 Vérification des contacts - Possibilité d'utiliser Lambda pour configurer un système de vérification de vos EC2 instances Amazon pour les contacts avec notification SNS. Il est important de noter que cela peut entraîner des frais en fonction de votre utilisation actuelle.
- Ground Station Amazon Machine Image Retrieval Lambda - La possibilité de sélectionner le logiciel installé sur votre instance et l'AMI de votre choix. Les options logicielles incluent DDX 2.6.2 Only et DDX 2.6.2 with qRadio 3.6.0. Si vous souhaitez utiliser Wideband DigiF Data Delivery et l' AWS Ground Station agent, consultez. [Satellite de diffusion publique utilisant AWS Ground Station l'agent \(large bande\)](#) Ces options continueront de s'étendre à mesure que des mises à jour logicielles et des fonctionnalités supplémentaires seront publiées.
- Profils de mission supplémentaires - Profils de mission pour d'autres satellites de diffusion publique (Aqua, SNPP et Terra).
- Configurations supplémentaires de liaison descendante d'antenne - Configurations de liaison descendante d'antenne pour des satellites de diffusion publics supplémentaires (Aqua, SNPP et Terra).

Les valeurs et paramètres pour les satellites dans ce modèle sont déjà renseignés. Ces paramètres vous permettent d'utiliser facilement et AWS Ground Station immédiatement ces satellites. Vous n'avez pas besoin de configurer vos propres valeurs pour pouvoir les utiliser AWS Ground Station lorsque vous utilisez ce modèle. Toutefois, vous pouvez personnaliser les valeurs pour que le modèle fonctionne selon votre cas d'utilisation.

Où puis-je recevoir mes données ?

Le groupe de points de terminaison de flux de données est configuré pour utiliser l'interface réseau d'instance de récepteur créée dans le cadre du modèle. L'instance de réception utilise une application de point de terminaison de flux de données pour recevoir le flux de données depuis le port défini par AWS Ground Station le point de terminaison du flux de données. Une fois reçues, les données sont disponibles pour la consommation via le port UDP 50000 sur l'adaptateur de bouclage de l'instance du récepteur. Pour plus d'informations sur la configuration d'un groupe de points de terminaison de flux de données, consultez. [AWS::GroundStation::DataflowEndpointGroup](#)

Satellite de diffusion publique utilisant AWS Ground Station l'agent (large bande)

Cet exemple s'appuie sur l'analyse effectuée dans la [JPSS-1 - Satellite de diffusion public \(PBS\) - Évaluation](#) section du guide de l'utilisateur.

Pour compléter cet exemple, vous devez supposer un scénario : vous souhaitez capturer le chemin de communication HRD sous forme de fréquence intermédiaire numérique à large bande (DigIF) et le traiter tel qu'il est reçu par l'agent AWS Ground Station sur une instance Amazon EC2 à l'aide d'un SDR.

Note

Le signal du chemin de communication JPSS HRD actuel a une bande passante de 30 MHz, mais vous allez configurer la configuration antenne-liaison descendante pour le traiter comme un signal avec une MHz bande passante de 100 % afin qu'il puisse circuler sur le chemin correct pour être reçu par l'agent dans cet exemple. AWS Ground Station

Voies de communication

Cette section représente [Planifiez les voies de communication de votre flux de données](#) la mise en route. Pour cet exemple, vous aurez besoin d'une section supplémentaire dans votre CloudFormation modèle qui n'a pas été utilisée dans les autres exemples, la section Mappings.

Note

Pour plus d'informations sur le contenu d'un CloudFormation modèle, consultez les [sections relatives aux modèles](#).

Vous allez commencer par configurer une section Mappings dans votre CloudFormation modèle pour les listes de AWS Ground Station préfixes par région. Cela permet aux listes de préfixes d'être facilement référencées par le groupe de sécurité des EC2 instances Amazon. Pour plus d'informations sur l'utilisation d'une liste de préfixes, consultez [Configuration VPC avec agent AWS Ground Station](#).

Mappings:**PrefixListId:**

```
us-east-2:
  groundstation: pl-087f83ba4f34e3bea
us-west-2:
  groundstation: pl-0cc36273da754ebdc
us-east-1:
  groundstation: pl-0e5696d987d033653
eu-central-1:
  groundstation: pl-03743f81267c0a85e
sa-east-1:
  groundstation: pl-098248765e9effc20
ap-northeast-2:
  groundstation: pl-059b3e0b02af70e4d
ap-southeast-1:
  groundstation: pl-0d9b804fe014a6a99
ap-southeast-2:
  groundstation: pl-08d24302b8c4d2b73
me-south-1:
  groundstation: pl-02781422c4c792145
eu-west-1:
  groundstation: pl-03fa6b266557b0d4f
eu-north-1:
  groundstation: pl-033e44023025215c0
af-south-1:
  groundstation: pl-0382d923a9d555425
```

Pour la section Paramètres, vous allez ajouter les paramètres suivants. Vous spécifierez leurs valeurs lors de la création de la pile via la CloudFormation console.

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

Vous devez créer une paire de clés et fournir le nom du EC2 EC2Key paramètre Amazon. Consultez la section [Créer une paire de clés pour votre EC2 instance Amazon](#).

En outre, vous devrez fournir le bon ID d'AMI spécifique à la région lors de la création de la CloudFormation pile. Consultez [AWS Ground Station Images de machines Amazon \(AMIs\)](#).

Les autres extraits de modèle se trouvent dans la section Ressources du CloudFormation modèle.

Resources:

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

Compte tenu de notre scénario consistant à fournir un chemin de communication unique à une EC2 instance Amazon, vous savez que vous n'aurez qu'un seul chemin de livraison synchrone. Selon la [Livraison synchrone des données](#) section, vous devez installer et configurer une EC2 instance Amazon avec l' AWS Ground Station Agent, et créer un ou plusieurs groupes de points de terminaison de flux de données. Vous allez commencer par configurer le VPC Amazon pour l' AWS Ground Station agent.

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

Pour plus d'informations sur les configurations VPC prises en charge par l' AWS Ground Station agent, consultez la section [Exigences relatives à l'AWS Ground Station agent - diagrammes VPC](#).

Vous allez ensuite configurer l' EC2 instance Amazon de Receiver.

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="42000-55000" >> /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

```

Vous aurez également besoin des politiques, des rôles et des profils appropriés AWS Ground Station pour créer l'Elastic network interface (ENI) dans votre compte.

```
# 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 configurations

Cette section représente [Création de configurations](#) la mise en route.

Vous aurez besoin d'une configuration de suivi pour définir vos préférences en matière d'utilisation du suivi automatique. La sélection de PREFERRED comme autotrack peut améliorer la qualité du signal, mais elle n'est pas obligatoire pour atteindre la qualité du signal en raison de la qualité suffisante des éphémérides JPSS-1.

```
TrackingConfig:
```

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

Sur la base du chemin de communication, vous devez définir une configuration antenne-liaison descendante pour représenter la partie satellite, ainsi qu'une configuration de point de terminaison de flux de données pour faire référence au groupe de points de terminaison de flux de données qui définit les détails du point de terminaison.

```
# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
# downlink data from your satellite.
SnppJpssDownlinkDigIfAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "SNPP JPSS Downlink WBDigIF Antenna Config"
    ConfigData:
      AntennaDownlinkConfig:
        SpectrumConfig:
          Bandwidth:
            Units: "MHz"
            Value: 100
          CenterFrequency:
            Units: "MHz"
            Value: 7812
          Polarization: "RIGHT_HAND"

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

AWS Ground Station profil de mission

Cette section représente [Création d'un profil de mission](#) la mise en route.

Maintenant que vous disposez des configurations associées, vous pouvez les utiliser pour créer le flux de données. Vous utiliserez les valeurs par défaut pour les autres paramètres.

```
# 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
```

Assemblage

Grâce aux ressources ci-dessus, vous avez désormais la possibilité de planifier des contacts JPSS-1 pour une livraison synchrone des données depuis n'importe lequel de vos contacts intégrés. AWS Ground Station [AWS Ground Station Succursales](#)

Ce qui suit est un CloudFormation modèle complet qui inclut toutes les ressources décrites dans cette section combinées dans un modèle unique qui peut être directement utilisé dans CloudFormation.

Le CloudFormation modèle nommé

`DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml` est conçu pour vous permettre de commencer rapidement à recevoir des données de fréquence intermédiaire numérisées (DigIF)

pour les satellites Aqua, SNPP, JPSS-1/NOAA-20 et Terra. Il contient une EC2 instance Amazon et les CloudFormation ressources nécessaires pour recevoir les données brutes de diffusion directe de DigiF à l'aide AWS Ground Station de l'agent.

Si Aqua, SNPP, JPSS-1/NOAA-20 et Terra ne sont pas intégrés à votre compte, consultez. [satellite embarqué](#)

Note

Vous pouvez accéder au modèle en accédant au compartiment Amazon S3 du client qui intègre le compartiment Amazon S3 à l'aide AWS d'informations d'identification valides. Les liens ci-dessous utilisent un compartiment Amazon S3 régional. Modifiez le code de us-west-2 région pour représenter la région correspondante dans laquelle vous souhaitez créer la CloudFormation pile.

En outre, les instructions suivantes utilisent le langage YAML. Cependant, les modèles sont disponibles au format YAML et JSON. Pour utiliser le format JSON, remplacez l'extension de .yml fichier par .json lors du téléchargement du modèle.

Pour télécharger le modèle à l'aide de AWS CLI, utilisez la commande suivante :

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

Vous pouvez consulter le modèle et le télécharger dans la console en entrant l'URL suivante dans votre navigateur :

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

Vous pouvez spécifier le modèle directement en CloudFormation utilisant le lien suivant :

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

Quelles ressources supplémentaires le modèle définit-il ?

Le DirectBroadcastSatelliteWbDigIfEc2DataDelivery modèle inclut les ressources supplémentaires suivantes :

- Interface réseau élastique de l'instance de réception - (Conditionnel) Une interface réseau élastique est créée dans le sous-réseau spécifié par `PublicSubnetId` s'il est fourni. Cela est nécessaire si l'instance du récepteur se trouve dans un sous-réseau privé. L'interface Elastic network sera associée à l'EIP et attachée à l'instance de réception.
- IP élastique de l'instance de réception : adresse IP élastique à laquelle se connectera AWS Ground Station. Cela s'attache à l'instance du récepteur ou à l'interface Elastic Network.
- L'une des associations d'adresses IP élastiques suivantes :
 - Association entre l'instance de récepteur et l'adresse IP élastique : association de l'adresse IP élastique à votre instance de récepteur, si elle n'est pas spécifiée. Cela nécessite de référencer un sous-réseau public.
 - Association entre l'interface réseau élastique de l'instance de récepteur et l'interface IP élastique : association de l'adresse IP élastique à l'interface réseau élastique de l'instance de réception, si elle est spécifiée.
- (Facultatif) Déclencheurs d'événements CloudWatch : AWS Lambda fonction déclenchée à l'aide d'événements CloudWatch envoyés à AWS Ground Station avant et après un contact. La fonction Lambda démarrera et arrêtera éventuellement votre instance de réception.
- (Facultatif) Amazon EC2 Verification pour les contacts : possibilité d'utiliser Lambda pour configurer un système de vérification de vos EC2 instances Amazon pour les contacts avec notification SNS. Il est important de noter que cela peut entraîner des frais en fonction de votre utilisation actuelle.
- Profils de mission supplémentaires - Profils de mission pour d'autres satellites de diffusion publique (Aqua, SNPP et Terra).
- Configurations supplémentaires de liaison descendante d'antenne - Configurations de liaison descendante d'antenne pour des satellites de diffusion publics supplémentaires (Aqua, SNPP et Terra).

Les valeurs et paramètres pour les satellites dans ce modèle sont déjà renseignés. Ces paramètres vous permettent d'utiliser facilement et immédiatement ces satellites avec AWS Ground Station. Vous n'avez pas besoin de configurer vos propres valeurs pour pouvoir les utiliser avec AWS Ground Station lors de l'utilisation de ce modèle. Toutefois, vous pouvez personnaliser les valeurs pour que le modèle fonctionne selon votre cas d'utilisation.

Où puis-je recevoir mes données ?

Le groupe de points de terminaison de flux de données est configuré pour utiliser l'interface réseau d'instance de récepteur créée dans le cadre du modèle. L'instance de réception utilise l'AWS Ground

Station agent pour recevoir le flux de données depuis le port défini par AWS Ground Station le point de terminaison du flux de données. Pour plus d'informations sur la configuration d'un groupe de points de terminaison de flux de données, consultez. [AWS::GroundStation::DataflowEndpointGroup](#) Pour plus d'informations sur l' AWS Ground Station agent, voir [Qu'est-ce que l' AWS Ground Station agent ?](#)

Résolution des problèmes

La documentation suivante peut vous aider à résoudre les problèmes susceptibles de survenir lors de l'utilisation AWS Ground Station.

Rubriques

- [Résoudre les problèmes liés aux contacts qui fournissent des données à Amazon EC2](#)
- [Résoudre les problèmes liés aux contacts défaillants](#)
- [Résoudre les problèmes liés aux mises à jour des contacts](#)
- [Résoudre les problèmes liés aux contacts FAILED_TO_SCHEDULE](#)
- [Le dépannage DataflowEndpointGroups n'est pas dans un état SAIN](#)
- [Résoudre les problèmes liés aux éphémérides non valides](#)
- [Résoudre les problèmes liés aux contacts qui n'ont reçu aucune donnée](#)
- [Résoudre les problèmes de télémétrie](#)

Résoudre les problèmes liés aux contacts qui fournissent des données à Amazon EC2

Si vous ne parvenez pas à terminer un AWS Ground Station contact, vous devez vérifier que votre instance Amazon EC2 est en cours d'exécution, vérifier que votre application de point de terminaison de flux de données est en cours d'exécution et vérifier que le flux de votre application de point de terminaison de flux de données est correctement configuré.

Note

DataDefender (DDX) est un exemple d'application de point de terminaison de flux de données actuellement prise en charge par AWS Ground Station

Prérequis

Les procédures suivantes supposent qu'une instance Amazon EC2 est déjà configurée. Pour configurer une instance Amazon EC2 dans AWS Ground Station, consultez [Getting Started](#).

Étape 1 : vérifier que votre instance EC2 est en cours d'exécution

La procédure suivante montre comment trouver votre instance Amazon EC2 dans la console et comment la démarrer si elle n'est pas en cours d'exécution.

1. Localisez l'instance Amazon EC2 qui a été utilisée pour le contact que vous êtes en train de dépanner. Procédez comme suit :
 - a. Dans votre CloudFormation tableau de bord, sélectionnez la pile qui contient votre instance Amazon EC2.
 - b. Choisissez l'onglet Ressources et localisez votre instance Amazon EC2 dans la colonne Logical ID. Vérifiez que l'instance est créée dans la colonne Statut.
 - c. Dans la colonne Physical ID, choisissez le lien de votre instance Amazon EC2. Vous serez redirigé vers la console de gestion Amazon EC2.
2. Dans la console de gestion Amazon EC2, assurez-vous que l'état de votre instance Amazon EC2 est en cours d'exécution.
3. Si votre instance est en cours d'exécution, passez à l'étape suivante. Si votre instance n'est pas en cours d'exécution, démarrez-la en procédant comme suit :
 - Une fois votre instance Amazon EC2 sélectionnée, choisissez Actions > État de l'instance > Démarrer.

Étape 2 : Déterminer le type d'application de flux de données utilisé

Si vous utilisez l'AWS Ground Station agent pour la livraison de données, veuillez vous rediriger vers la section [AWS Ground Station Agent de résolution des problèmes](#). Sinon, si vous utilisez l'application DataDefender (DDX), continuez sur [the section called “Étape 3 : vérifier que l'application de flux de données est en cours d'exécution”](#).

Étape 3 : vérifier que l'application de flux de données est en cours d'exécution

Pour vérifier le statut de DataDefender , vous devez vous connecter à votre instance dans Amazon EC2. Pour plus de détails sur la connexion à votre instance, consultez la section [Se connecter à votre instance Linux](#).

La procédure suivante fournit des étapes de dépannage utilisant des commandes dans un client SSH.

1. Ouvrez un terminal ou une invite de commande et connectez-vous à votre instance Amazon EC2 à l'aide de SSH. Transférez le port 80 de l'hôte distant afin d'afficher l'interface utilisateur DataDefender Web. Les commandes suivantes montrent comment utiliser SSH pour se connecter à une instance Amazon EC2 via un bastion avec la redirection de port activée.

Note

Vous devez remplacer <SSH KEY><BASTION HOST>, et par <HOST>votre clé SSH spécifique, le nom d'hôte du bastion et le nom d'hôte de l'instance Amazon EC2.

Pour 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>
```

Pour 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. Vérifiez que DataDefender (également appelé DDX) est en cours d'exécution en greppant (en vérifiant) la présence d'un processus nommé ddx en cours d'exécution dans la sortie. La commande de vérification de l'existence d'un processus en cours d'exécution est fournie ci-dessous, avec un exemple de sortie réussie.

```
[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
```

Si elle DataDefender est en cours d'exécution, passez à la section [the section called “Étape 4 : Vérifiez que le flux d'applications de votre flux de données est configuré”](#) Sinon, passez à l'étape suivante.

3. Commencez DataDefender à utiliser la commande ci-dessous.

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

S'il DataDefender est en cours d'exécution après avoir utilisé la commande, passez à la section [the section called “Étape 4 : Vérifiez que le flux d'applications de votre flux de données est configuré”](#) Sinon, passez à l'étape suivante.

4. Inspectez les fichiers suivants à l'aide des commandes ci-dessous pour voir s'il y a eu des erreurs lors de l'installation et de la configuration DataDefender.

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

Note

Un problème courant découvert lors de l'inspection de ces fichiers est que le VPC Amazon sur lequel s'exécute votre instance Amazon EC2 n'a pas accès à Amazon S3 pour télécharger les fichiers d'installation. Si vous découvrez dans vos journaux que c'est le problème, vérifiez les paramètres du VPC Amazon et du groupe de sécurité de votre instance EC2 pour vous assurer qu'ils ne bloquent pas l'accès à Amazon S3.

S'il DataDefender est en cours d'exécution après avoir vérifié les paramètres de votre Amazon VPC, passez à [the section called “Étape 4 : Vérifiez que le flux d'applications de votre flux de données est configuré”](#) Si le problème persiste, [contactez AWS Support](#) et envoyez vos fichiers journaux avec une description de votre problème.

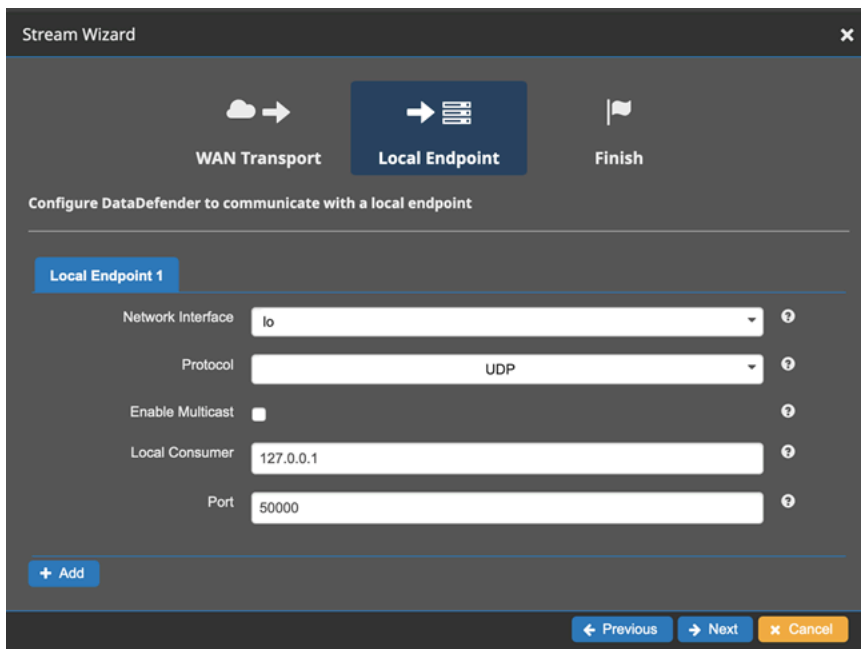
Étape 4 : Vérifiez que le flux d'applications de votre flux de données est configuré

1. Dans un navigateur Web, accédez à votre interface utilisateur DataDefender Web en saisissant l'adresse suivante dans la barre d'adresse : localhost:8080. Ensuite, appuyez sur Entrée.

2. Sur le DataDefender tableau de bord, choisissez Accéder aux détails.
3. Sélectionnez votre flux dans la liste des flux, puis choisissez Edit Stream (Modifier le flux).
4. Dans la boîte de dialogue Stream Wizard (Assistant de flux), procédez comme suit :
 - a. Dans le panneau WAN Transport (Transport WAN) assurez-vous que WAN to LAN (WAN vers LAN) est sélectionné comme direction de flux.
 - b. Dans la zone Port assurez-vous que le port WAN que vous avez choisi pour votre groupe de points de terminaison de flux de données est présent. Par défaut, ce port est 55888. Ensuite, choisissez Suivant.

The screenshot shows the 'Stream Wizard' dialog box. At the top, there are three tabs: 'WAN Transport' (selected), 'Local Endpoint', and 'Finish'. Below the tabs, the text reads 'Configure DataDefender to communicate across the WAN'. There are three input fields: 'Stream Name' with the value 'DownlinkDigIF', 'Stream Direction' with a dropdown menu set to 'WAN to LAN', and 'WAN Transport 1' section containing 'Network Interface' set to 'eth1', 'Enable Multicast' as an unchecked checkbox, and 'Port' set to '55888'. At the bottom, there is a '+ Add' button, a 'Next' button, and a 'Cancel' button.

- c. Dans le panneau Local Endpoint (Point de terminaison local), vérifiez qu'un port valide est présent dans la zone Port. Par défaut, ce port est 50000. Il s'agit du port sur lequel vous recevrez vos données après DataDefender les avoir reçues du AWS Ground Station service. Ensuite, choisissez Suivant.



- d. Choisissez Finish (Terminer) dans le menu restant si vous avez modifié des valeurs. Sinon, vous pouvez procéder à l'annulation à partir du menu Stream Wizard (Assistant de flux).

Vous avez maintenant vérifié que votre instance Amazon EC2 fonctionne et DataDefender est configurée correctement pour recevoir des données de. AWS Ground Station Passez au [the section called "Étape 5 : Assurez-vous que vous disposez d'un nombre suffisant d'adresses IP disponibles dans le sous-réseau de votre ou de vos instances de réception"](#).

Étape 5 : Assurez-vous que vous disposez d'un nombre suffisant d'adresses IP disponibles dans le sous-réseau de votre ou de vos instances de réception

La procédure suivante montre comment trouver le nombre d'adresses IP disponibles dans une instance de récepteur Amazon EC2 dans la console.

1. Pour chaque instance de récepteur Amazon EC2 utilisée pour le contact que vous dépannez. Procédez comme suit :
 - a. Dans votre CloudFormation tableau de bord, sélectionnez la pile qui contient votre instance Amazon EC2.
 - b. Choisissez l'onglet Ressources et localisez votre instance Amazon EC2 dans la colonne Logical ID. Vérifiez que l'instance est créée dans la colonne Statut.

- c. Dans la colonne Physical ID, choisissez le lien de votre instance Amazon EC2. Vous serez redirigé vers la console de gestion Amazon EC2.
2. Dans la console de gestion Amazon EC2, recherchez et cliquez sur le lien de l'ID de sous-réseau dans le résumé de l'instance de votre instance de réception Amazon EC2. Vous serez redirigé vers la console de gestion Amazon VPC correspondante.
3. Sélectionnez le sous-réseau correspondant dans la console de gestion Amazon VPC et vérifiez les détails de votre sous-réseau pour connaître les adresses disponibles. IPv4 Si ce nombre n'est pas au moins égal à celui des points de terminaison de flux de données qui utilisent cette instance de récepteur Amazon EC2, procédez comme suit :
 - a. Mettez à jour le sous-réseau correspondant à votre CloudFormation modèle CidrBlock pour qu'il soit correctement dimensionné. Pour plus de détails sur le dimensionnement des sous-réseaux, voir Blocs CIDR de [sous-réseau](#).
 - b. Redéployez votre stack avec votre CloudFormation modèle mis à jour.

Si vous continuez à rencontrer des problèmes, [contactez le support AWS](#).

Résoudre les problèmes liés aux contacts défectueux

Un contact aura le statut de contact terminal FAILED lorsqu'il AWS Ground Station détecte un problème avec la configuration de vos ressources. Les cas d'utilisation courants susceptibles d'entraîner l'échec des contacts sont présentés ci-dessous, ainsi que les étapes à suivre pour résoudre le problème.

Note

Ce guide est spécifiquement destiné à l'état d'échec du contact et n'est pas destiné à d'autres états de défaillance, tels que `AWS_FAILED`, `AWS_CANCELLED`, ou `FAILED_TO_SCHEDULE`. Pour plus d'informations sur les statuts des contacts, voir [the section called "AWS Ground Station statuts des contacts"](#)

Cas d'utilisation ayant échoué pour un point de terminaison Dataflow

Voici la liste des cas d'utilisation courants qui peuvent entraîner un statut d'échec du contact pour les flux de données basés sur des points de terminaison de flux de données :

- Le point de terminaison de flux de données ne se connecte jamais - La connexion entre l' AWS Ground Station antenne et votre groupe de points de terminaison de flux de données pour un ou plusieurs flux de données n'a jamais été établie.
- Connexion tardive du point de terminaison du flux de données : la connexion entre l' AWS Ground Station antenne et votre groupe de points de terminaison de flux de données pour un ou plusieurs flux de données a été établie après l'heure de début du contact.
- Le sous-réseau du point de terminaison Dataflow n'a plus d'adresses IP disponibles. La solution AWS Ground Station de livraison de données n'est pas en mesure de créer un ENI sur votre réseau privé car aucune adresse IP n'est disponible dans le sous-réseau de l'instance de réception.
- Le sous-réseau du point de terminaison Dataflow n'est pas valide. La solution AWS Ground Station de livraison de données n'est pas en mesure de créer un ENI sur votre réseau privé en raison de l'impossibilité d'accéder au sous-réseau fourni spécifié dans le groupe de points de terminaison Dataflow.

Pour tout cas de défaillance d'un point de terminaison de flux de données, il est recommandé d'examiner les points suivants :

- Vérifiez que l'instance Amazon EC2 du destinataire a bien été démarrée avant l'heure de début du contact.
- Vérifiez que le logiciel de point de terminaison du flux de données était opérationnel pendant le contact.
- Assurez-vous de disposer d'au moins une adresse IP disponible par point de terminaison de flux de données par sous-réseau d'instance de récepteur.
- Assurez-vous que les sous-réseaux associés à votre groupe de points de terminaison de flux de données, via les flux de données configurés dans [Configuration et configuration d'Amazon VPC](#), restent actifs et accessibles. AWS Ground Station

Consultez la section sur les étapes [Résoudre les problèmes liés aux contacts qui fournissent des données à Amazon EC2](#) de résolution des problèmes plus spécifiques.

AWS Ground Station Cas d'utilisation de l'agent ayant échoué

Voici la liste des cas d'utilisation courants qui peuvent entraîner un statut d'échec du contact pour les flux de données basés sur des agents :

- AWS Ground Station État jamais signalé à l'agent : l'agent chargé d'orchestrer la livraison des données sur votre groupe de points de terminaison de flux de données pour un ou plusieurs flux de données n'a jamais correctement signalé l'état à AWS Ground Station. Cette mise à jour du statut devrait avoir lieu quelques secondes après la fin du contact.
- AWS Ground Station Agent démarré en retard : l'agent chargé d'orchestrer la livraison des données sur votre groupe de points de terminaison de flux de données pour un ou plusieurs flux de données a démarré en retard, après l'heure de début du contact.

Pour tout cas de défaillance du flux de données de l' AWS Ground Station agent, il est recommandé de prendre en compte les points suivants :

- Vérifiez que l'instance Amazon EC2 du destinataire a bien été démarrée avant l'heure de début du contact.
- Vérifiez que l'application Agent était opérationnelle au début et pendant le contact.
- Vérifiez que l'application Agent et l'instance Amazon EC2 n'ont pas été arrêtées dans les 15 secondes suivant la fin du contact. Cela donne à l'agent suffisamment de temps pour signaler l'état à AWS Ground Station.

Consultez la section sur les étapes [Résoudre les problèmes liés aux contacts qui fournissent des données à Amazon EC2](#) de résolution des problèmes plus spécifiques.

Résoudre les problèmes liés aux mises à jour des contacts

Lorsque vous appelez l'[UpdateContact](#) API, AWS Ground Station effectue une validation synchrone de la demande. Si la validation est réussie, la mise à jour est traitée de manière asynchrone pour propager les modifications apportées à la région de l'antenne. Les erreurs de validation synchrones sont renvoyées directement dans la réponse HTTP. Les défaillances asynchrones sont signalées par le biais `failureCodes` des `failureMessage` champs de la version du contact, que vous pouvez consulter en appelant [DescribeContactVersion](#) la version qui n'a pas pu être mise à jour.

Pour plus d'informations sur la gestion des versions des contacts, consultez [Mettre à jour les contacts et la gestion des versions des contacts](#).

Erreurs de validation synchrones

Les erreurs suivantes sont renvoyées directement dans la réponse HTTP lorsque la [UpdateContact](#) demande échoue à la validation.

ResourceNotFoundException: Contact introuvable

Cause commune

Le compte spécifié `contactId` n'existe pas ou appartient à un autre AWS compte.

Résolution

1. Vérifiez que `contactId` c'est correct.
2. Vérifiez que vous utilisez les informations d'identification du AWS compte auquel appartient le contact.
3. [ListContacts](#) Utilisez-le pour trouver le bon `contactId`.

ConflictException: Impossible de mettre à jour le contact

Cause commune

Le contact est dans un état qui n'autorise pas les mises à jour. L'[UpdateContact](#) API ne peut être appelée que lorsque le contact est dans l'`PASS` état `SCHEDULEDPREPASS`, ou. Cette erreur se produit également si une autre mise à jour est déjà en cours (la dernière version du contact est dans l'`UPDATING` état actuel).

Résolution

1. Appelez [DescribeContact](#) pour vérifier l'état actuel du contact.
2. Si le contact est dans un état terminal (par exemple, `CANCELLED`), `COMPLETED`, `FAILED`, il ne peut pas être mis à jour. Un contact ne peut être mis à jour que lorsqu'il est dans l'`PASS` état `SCHEDULEDPREPASS`, ou. Pour une liste complète des états du terminal, voir [AWS Ground Station statuts des contacts](#).
3. Si une autre mise à jour est en cours, attendez que la mise à jour actuelle atteigne `ACTIVE` ou `FAILED_TO_UPDATE` son statut avant de soumettre une autre mise à jour. Vous pouvez interroger l'[DescribeContactVersion](#) API ou utiliser les utilitaires de `ContactUpdated` commodité pour serveurs fournis par certains AWS SDKs et le AWS Command Line Interface.

InvalidParameterException: paramètres de demande non valides

Cause commune

La demande contient des paramètres non valides. Les causes courantes incluent :

- Manquant ou vide `clientToken`.
- Plusieurs types de `ProgramTrackSettings` (azimut/élévation, OEM et TLE) sont inclus dans une seule demande. Un seul type est autorisé par demande.
- Réglage `satelliteArn` sur zéro sans approbation pour les [éphémérides d'altitude azimutale à la station au sol](#) du contact.
- `AzElProgramTrackSettingsManquant` lorsque `satelliteArn` est nul.
- Fournir un `ephemerisId` qui n'est pas associé à ce qui est spécifié `satelliteArn`.
- Le satellite ne dispose pas d'une fenêtre de visibilité valide depuis la station au sol pour la plage de temps de contact.
- Le satellite n'est pas embarqué à bord de la station au sol ou ne possède pas la licence requise par le profil de mission.
- Le profil de mission inclut des [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#) configurations, qui ne sont pas prises en charge pour les mises à jour des contacts.

Résolution

1. Consultez le message d'erreur contenu dans la réponse pour savoir quel paramètre n'est pas valide.
2. Assurez-vous de fournir exactement un type `ProgramTrackSettings` de demande.
3. Si vous utilisez des angles de azimuth/elevation pointage sans `satelliteArn`, vérifiez que votre compte est approuvé pour cette fonctionnalité à la station au sol. Pour de plus amples informations, veuillez consulter [Fournir des données sur les éphémérides d'altitude azimutale](#).
4. Vérifiez que l'éphéméride à laquelle vous faites référence est associée au bon satellite et couvre la plage de temps de contact.

`ResourceLimitExceededException`: limite de version maximale atteinte

Cause commune

Le contact a atteint le nombre maximum de versions (128). Chaque appel [UpdateContact](#) crée une nouvelle version, et un contact ne peut pas dépasser cette limite.

Résolution

1. Cette limite ne peut pas être augmentée. Si vous devez apporter d'autres modifications, annulez le contact et réservez-en un nouveau.

Codes de défaillance asynchrones

Les codes d'erreur suivants apparaissent dans le `failureCodes` champ d'une version de contact avec un `FAILED_TO_UPDATE` statut. [DescribeContactVersion](#) À utiliser pour récupérer ces informations. Le `failureMessage` champ fournit un contexte supplémentaire concernant l'échec.

Code de défaillance	Cause courante	Résolution
<code>INTERNAL_ERROR</code>	Une erreur interne inattendue s'est produite lors du traitement de la mise à jour.	Réessayez la mise à jour. Si le problème persiste, contactez AWS Support .
<code>INVALID_SATELLITE_ARN</code>	L'ARN du satellite fourni dans la demande de mise à jour n'est pas valide ou n'existe pas.	Vérifiez l'ARN du satellite et confirmez que le satellite est enregistré dans votre compte.
<code>INVALID_UPDATE_CONTACT_REQUEST</code>	La demande de mise à jour contient des paramètres non valides qui n'ont pas été interceptés lors de la validation synchrone.	Consultez le <code>failureMessage</code> pour plus de détails et corrigez les paramètres de la demande.
<code>EPHEMERIS_NOT_FOUND</code>	Les éphémérides référencées dans les dérogations de suivi n'existent pas.	Vérifiez <code>ephemerisId</code> et confirmez que l'éphéméride n'a pas été supprimée.
<code>EPHEMERIS_TIME_RANGE_INVALID</code>	L'éphéméride ne couvre pas la plage de temps du contact.	Téléchargez une nouvelle éphéméride qui couvre l'ensemble de la plage de temps de contact. Si la plage de temps des éphémérides ne peut pas être prolongée, annulez le contact et réservez-en un nouveau pendant la durée des éphémérides. Pour de plus amples informations, veuillez consulter Fournir des données d'éphémérides personnalisées .

Code de défaillance	Cause courante	Résolution
EPHEMERIS _NOT_ENABLED	L'éphéméride référencée n'est pas en état. ENABLED	Vérifiez le statut des éphémérides et activez-le avant de réessayer la mise à jour.
SATELLITE _DOES_NOT _MATCH_EPHEMERIS	L'éphéméride n'est pas associée au satellite spécifié dans la demande de mise à jour.	Assurez-vous qu'il ephemeris Id appartient au satellite spécifié dans satelliteArn .
NOT_ONBOARDED_TO_AZEL_EPHEMERIS	Votre compte n'est pas autorisé à utiliser les données d'éphémérides d'altitude azimutale à la station au sol du contact. L'éphéméride d'élévation d'azimut est une fonctionnalité restreinte disponible pour un nombre limité de cas d'utilisation spécialisés.	Si des éphémérides d'élévation d'azimut sont requises pour votre cas d'utilisation, ouvrez un AWS Support ticket via le pour demander l'accès. AWS Support Center Console Vous pouvez également envisager d'utiliser des données d'éphémérides TLE ou des données d'éphémérides OEM si elles correspondent à votre cas d'utilisation.
AZEL_EPHEMERIS_NOT_FOUND	L' éphéméride d'élévation d'azimut référencée dans la demande n'existe pas.	Vérifiez ephemerisId et confirmez que l'éphéméride d'altitude azimutale n'a pas été supprimée.
AZEL_EPHEMERIS_WRONG_GROUND_STATION	L' éphéméride d'élévation azimutale a été créée pour une station au sol différente de celle utilisée par le contact.	Téléchargez une nouvelle éphéméride d'altitude azimutale pour la bonne station au sol, ou utilisez une éphéméride existante correspondant à la station au sol du contact.

Code de défaillance	Cause courante	Résolution
AZEL_EPHE MERIS_INV ALID_STATUS	L' éphéméride d'élévation d'azimut n'est pas dans un état valide pour être utilisée.	Vérifiez l'état des éphémérides. Il doit être en ENABLED état. Si la validation de l'éphéméride a échoué, téléchargez une version corrigée.
AZEL_EPHE MERIS_TIM E_RANGE_INVALID	L' éphéméride d'élévation de l'azimut ne couvre pas la plage de temps du contact.	Téléchargez une nouvelle éphéméride d'altitude azimutale qui couvre l'ensemble de la plage de temps de contact. Si la plage de temps des éphémérides ne peut pas être prolongée, annulez le contact et réservez-en un nouveau pendant la durée des éphémérides.

Vérifier l'état d'une mise à jour

Après l'appel `UpdateContact`, la nouvelle version du contact démarre dans l'`UPDATING` état. Pendant ce temps, [DescribeContact](#) continue de renvoyer la version précédemment active du contact. La nouvelle version n'apparaît pas `DescribeContact` tant qu'elle n'a pas été propagée à l'antenne et n'a pas atteint le `ACTIVE` statut. Pour vérifier l'état d'une version spécifique, utilisez [DescribeContactVersion](#).

Pour déterminer si une mise à jour a réussi ou échoué, procédez comme suit :

1. Appelez [DescribeContactVersion](#) avec le `contactId` et `versionId` renvoyé par la `UpdateContact` réponse.
2. Vérifiez le `version.status` champ. Le statut indique `ACTIVE` que la mise à jour a été correctement appliquée. L'état indique `FAILED_TO_UPDATE` que la mise à jour a échoué.
3. Si le statut est le cas `FAILED_TO_UPDATE`, vérifiez les `version.failureMessage` champs `version.failureCodes` et pour plus de détails sur ce qui s'est mal passé.

i Tip

Certains AWS SDKs et le AWS Command Line Interface soutiennent un `ContactUpdated` serveur qui interroge automatiquement `DescribeContactVersion` jusqu'à ce que la version atteigne son `ACTIVE FAILED_TO_UPDATE` statut. Par exemple, AWS Command Line Interface fournit une commande [aws groundstation wait contact-updated](#). Utilisez le serveur au lieu d'implémenter votre propre logique de sondage.

Résoudre les problèmes liés aux contacts FAILED_TO_SCHEDULE

Un contact prend fin dans l'état `FAILED_TO_SCHEDULE` lorsqu'il AWS Ground Station détecte un problème lié à la configuration de vos ressources ou au sein du système interne. Un contact qui se termine par un état `FAILED_TO_SCHEDULE` fournira éventuellement un contexte supplémentaire. `errorMessage` Pour plus d'informations sur la description des contacts, consultez l'[DescribeContactAPI](#).

Les cas d'utilisation courants susceptibles de provoquer des contacts `FAILED_TO_SCHEDULE` sont présentés ci-dessous, ainsi que les étapes permettant de résoudre les problèmes.

i Note

Ce guide est spécifiquement destiné à l'état du contact `FAILED_TO_SCHEDULE` et n'est pas destiné à d'autres états de défaillance, tels que, ou `FAILED`.
`AWS_FAILED``AWS_CANCELLED` Pour plus d'informations sur les statuts des contacts, voir [the section called "AWS Ground Station statuts des contacts"](#)

Les paramètres spécifiés dans votre Antenna Downlink Demod Decode Config ne sont pas pris en charge

Le [profil de mission](#) utilisé pour planifier ce contact avait une [antenna-downlink-demod-decode configuration](#) non valide.

AntennaDownlinkDemodDecode Configuration existante

- Si vos antenna-downlink-demod-decode configurations ont récemment été modifiées, revenez à une version qui fonctionnait auparavant avant de tenter de planifier.
- S'il s'agit d'une modification intentionnelle d'une configuration existante ou d'une configuration existante qui n'est plus correctement planifiée, suivez l'étape suivante pour intégrer une nouvelle AntennaDownlinkDemodDecode configuration.

AntennaDownlinkDemodDecode Configuration nouvellement créée

Contactez-nous AWS Ground Station directement pour intégrer votre nouvelle configuration. Créez un dossier avec [AWS Support](#), y compris contactId celui qui s'est terminé par l'état FAILED_TO_SCHEDULE

Étapes générales de résolution des problèmes

Si les étapes de dépannage précédentes n'ont pas permis de résoudre votre problème :

- Réessayez de planifier le contact ou planifiez un autre contact en utilisant le même profil de mission. Pour plus d'informations sur la façon de réserver un contact, consultez [ReserveContact](#).
- [Si vous continuez à recevoir le statut FAILED_TO_SCHEDULE pour ce profil de mission, contactez AWS Support](#)

Le dépannage DataflowEndpointGroups n'est pas dans un état SAIN

Vous trouverez ci-dessous les raisons pour lesquelles vos groupes de points de terminaison de flux de données ne sont peut-être pas en bon HEALTHY état, ainsi que les mesures correctives appropriées à prendre.

- NO_REGISTERED_AGENT- Démarrez votre instance EC2, qui enregistrera l'agent. Notez que vous devez disposer d'un fichier de configuration de contrôleur valide pour que cet appel réussisse. Reportez-vous au [Utiliser l' AWS Ground Station agent](#) pour plus de détails sur la configuration de ce fichier.
- INVALID_IP_OWNERSHIP- Utilisez l' DeleteDataflowEndpointGroup API pour supprimer le groupe de points de terminaison Dataflow, puis utilisez l' CreateDataflowEndpointGroup API pour recréer le groupe de points de terminaison Dataflow à l'aide des adresses IP et des ports associés à l'instance EC2.

- UNVERIFIED_IP_OWNERSHIP- L'adresse IP n'a pas encore été validée. La validation a lieu périodiquement, ce problème devrait donc se résoudre de lui-même.
- NOT_AUTHORIZED_TO_CREATE_SLR- Le compte n'est pas autorisé à créer le rôle lié au service nécessaire. Consultez les étapes de résolution des problèmes dans [Utiliser des rôles liés à un service pour Ground Station](#)

Résoudre les problèmes liés aux éphémérides non valides

Lorsque vous chargez des données d'éphémérides vers AWS Ground Station, elles sont soumises à un flux de travail de validation asynchrone. Si la validation échoue, le statut de l'éphéméride passe à INVALID. Le message d'erreur contenu dans la [DescribeEphemeris](#) réponse fournit des informations détaillées pour vous aider à identifier et à résoudre le problème.

Comprendre les erreurs de validation des éphémérides

Lorsqu'une éphéméride échoue à la validation, la réponse de l'[DescribeEphemeris](#) API inclut deux champs pour aider à diagnostiquer le problème :

errorCode

Code lisible par machine identifiant l'erreur de validation spécifique. Cela peut être utilisé pour la gestion des erreurs de programmation.

errorMessage

Une description lisible par l'homme de l'erreur de validation avec des détails spécifiques sur ce qui s'est mal passé et des conseils pour y remédier.

Exemple de [DescribeEphemeris](#) réponse pour une éphéméride non valide :

```
{
  "ephemerisId": "abc12345-6789-def0-1234-567890abcdef",
  "name": "My Invalid Ephemeris",
  "status": "INVALID",
  "creationTime": 1620254718.765,
  "invalidReason": "METADATA_INVALID",
  "errorCode": "OBJECT_NAME_MISSING",
  "errorMessage": "Metadata field missing: OBJECT_NAME",
  "suppliedData": {
    "tle": {
```

```
    "ephemerisData": "[...]"  
  }  
}
```

Erreurs de validation courantes pour les éphémérides TLE

Les erreurs de validation les plus courantes rencontrées lors du téléchargement d'éphémérides TLE sont les suivantes :

Numéro de catalogue satellite non concordant

Erreur : « Le numéro de catalogue satellite présent dans l'éphéméride ne correspond pas au numéro de catalogue satellite du satellite associé »

Solution : Vérifiez que le numéro de ID/satellite catalogue du NORAD figurant sur vos lignes TLE correspond au numéro de catalogue satellite de votre satellite. 00000À utiliser pour les satellites sans numéro de catalogue attribué.

Mouvement moyen non valide

Erreur : « Le mouvement moyen des éphémérides fournies est trop différent de celui des éphémérides de référence les plus récentes »

Solution : Vérifiez que vos données TLE sont correctes et représentent une orbite valide. Ground Station utilise les éphémérides Space-Track comme référence lors de la validation.

Erreurs de validation courantes pour les éphémérides OEM

Les erreurs de validation les plus courantes rencontrées lors du téléchargement d'éphémérides OEM sont les suivantes :

Cadre de référence non valide

Erreur : « Le REF_FRAME n'est pas pris en charge »

Solution : mettez à jour votre fichier OEM pour utiliser l'un des cadres de référence pris en charge : EME2000 ou ITRF2000.

Champs obligatoires manquants

Erreur : « Champ de métadonnées manquant : INTERPOLATION »

Solution : ajoutez les champs INTERPOLATION et INTERPOLATION_DEGREE à votre section de métadonnées OEM. Ils sont nécessaires pour AWS Ground Station générer des angles de pointage précis de l'antenne.

Système horaire non pris en charge

Erreur : « Le TIME_SYSTEM n'est pas pris en charge »

Solution : Assurez-vous que votre fichier OEM utilise l'UTC comme système horaire.

Version OEM non prise en charge

Erreur : « Le CCSDS_OEM_VERS n'est pas pris en charge »

Solution : Assurez-vous que votre fichier OEM utilise la version 2.0 de CCSDS OEM.

Erreurs de validation courantes pour les éphémérides d'élévation d'azimut

Les erreurs de validation les plus courantes rencontrées lors du téléchargement d'éphémérides d'élévation d'azimut sont les suivantes :

azimuth/elevation Données manquantes

Erreur : « Aucun TimeAzEl champ n'était présent dans au moins un AzElSegment »

Solution : Assurez-vous que chaque segment de vos données d'altitude azimutale contient au moins une paire horodatée. azimuth/elevation

Plage d'angles d'azimut non valide (degrés)

Erreur : « AzEl az doit être supérieur ou égal à -180 et inférieur ou égal à 360 degrés »

Solution : Vérifiez que les angles d'azimut se situent dans les limites de [-180, 360] degrés.

Plage d'angles d'élévation non valide (degrés)

Erreur : « AzEl el doit être supérieur ou égal à -90 et inférieur ou égal à 90 degrés »

Solution : Vérifiez que les angles d'élévation sont compris entre [-90, 90] degrés.

Plage d'angles d'azimut non valide (radians)

Erreur : « AzEl az doit être supérieur ou égal à -pi et inférieur ou égal à 2 pi radians »

Solution : Vérifiez que les angles d'azimut sont compris entre $[-\pi, 2\pi]$ radians.

Plage d'angles d'élévation non valide (radians)

Erreur : « AzEl el doit être supérieur ou égal à $-\pi/2$ et inférieur ou égal à $\pi/2$ radians »

Solution : Vérifiez que les angles d'élévation sont compris entre $[-\pi/2, \pi/2]$ radians.

Valeurs temporelles non monotones

Erreur : « Les TimeAzEl articles contenus dans un AzElSegment doivent être en ordre dans le temps »

Solution : assurez-vous que les valeurs temporelles de chaque segment augmentent strictement.

Segments hors ordre

Erreur : « AzElSegments doit être temporairement en ordre »

Solution : Assurez-vous que les segments sont classés par ordre chronologique.

Segments qui se chevauchent

Erreur : « La plage de temps d'au moins un segment se chevauche avec d'autres plages temporelles de segment »

Solution : Assurez-vous que chaque segment possède une plage de temps unique qui ne se chevauche pas. Le endTime d'un segment ne doit pas dépasser celui startTime du segment suivant.

Étapes de résolution des problèmes

Si la validation de votre éphéméride échoue, procédez comme suit pour résoudre le problème :

1. Appelez [DescribeEphemeris](#) avec votre identifiant d'éphéméride pour récupérer le errorCode et errorMessage
2. Consultez le message d'erreur pour obtenir des informations spécifiques sur le contrôle de validation qui a échoué.
3. Corrigez les problèmes identifiés dans vos données d'éphémérides.
4. Téléchargez une nouvelle éphéméride avec les données corrigées à l'aide de [CreateEphemeris](#)
5. Surveillez le nouvel état des éphémérides jusqu'à ce qu'il atteigne ENABLED cet état.

6. Supprimez les éphémérides non valides en indiquant [DeleteEphemeris](#) si elles ne sont plus nécessaires.

Référence complète du code d'erreur

Les sections suivantes fournissent un mappage complet de toutes les `errorCode` valeurs qui peuvent être renvoyées en cas d'échec de la validation des éphémérides, organisées par catégorie de haut niveau `invalidReason`.

Motif non valide : **METADATA_INVALID**

Ces erreurs se produisent lorsque les champs de métadonnées obligatoires sont manquants, mal formatés ou contiennent des valeurs non prises en charge dans les données d'éphémérides.

Code d'erreur	Message d'erreur
IDENTIFIANT_SATCAT_ID NON CONCORDANT	Le numéro de catalogue satellite présent dans les éphémérides TLE ne correspond pas au numéro de catalogue satellite du satellite associé
OEM_VERSION_NON PRISE EN CHARGE	Les éphémérides présentes <code>CCSDS_OEM_VERS</code> dans l'OEM ne sont pas prises en charge. Valeurs prises en charge : [2.0]
ORIGINATEUR_MANQUANT	Le champ <code>ORIGINATOR</code> d'en-tête est absent des éphémérides OEM
DATE_DE_CRÉATION_M ANQUANTE	Le champ <code>CREATION_DATE</code> d'en-tête est absent des éphémérides OEM
NOM_OBJET_MANQUANT	Le champ de <code>OBJECT_NAME</code> métadonnées est absent des éphémérides OEM
IDENTIFIANT_OBJET MANQUANT	Le champ de <code>OBJECT_ID</code> métadonnées est absent des éphémérides OEM
REF_FRAME_NON PRIS EN CHARGE	Les éphémérides présentes <code>REF_FRAME</code> dans l'OEM ne sont pas prises en charge. Valeurs prises en charge : [EME2000, ITRF2000]

Code d'erreur	Message d'erreur
REF_FRAME_EPOCH_NON PRIS EN CHARGE	Le champ de REF_FRAME_EPOCH métadonnées dans les éphémérides OEM n'est pas pris en charge. Veuillez supprimer ce champ de l'éphéméride
TIME_SYSTEM_UNSUPPORTED	Les éphémérides présentes TIME_SYSTEM dans l'OEM ne sont pas prises en charge. Valeurs prises en charge : [UTC]
CENTER_BODY_UNSUPPORTED	Les éphémérides présentes CENTER_BODY dans l'OEM ne sont pas prises en charge. Valeurs prises en charge : [Earth]
INTERPOLATION_MANQUANTE	Le champ de INTERPOLATION métadonnées est absent des éphémérides OEM
DEGRÉ_INTERPOLATION_INCORRECT	Le degré d'interpolation dans les éphémérides OEM doit être supérieur à 0 pour la méthode d'interpolation.
AZ_EL_SEGMENT_LIST_MISSING	Le azElSegmentList champ est manquant
INSUFFICIENT_TIME_AZ_EL	Aucun TimeAzEl champ n'était présent dans au moins un azElSegmentList

Motif non valide : **TIME_RANGE_INVALID**

Ces erreurs se produisent lorsque l'éphéméride contient des plages temporelles non valides, notamment des problèmes liés aux start/end heures, à l'ordre des segments, à des segments qui se chevauchent ou à des incohérences temporelles.

Code d'erreur	Message d'erreur
HEURE DE DÉBUT DANS LE FUTUR	L'heure de début des éphémérides est dans le futur, mais doit être dans le passé
END_TIME_IN_PAST	La fin des éphémérides appartient au passé, mais elle doit l'être dans le futur

Code d'erreur	Message d'erreur
EXPIRATION_TIME_TROP_TÔT	Le délai d'expiration indiqué est antérieur à l'heure de fin de l'éphéméride
START_TIME_METADATA_TOO_EARLY	La valeur <code>START_TIME</code> des métadonnées est antérieure à la première heure présente dans les données d'éphémérides OEM
STOP_TIME_METADATA_TOO_LATE	La valeur <code>STOP_TIME</code> des métadonnées est postérieure à la dernière date de présence dans les données d'éphémérides OEM
AZ_EL_SEGMENT_END_TIME_BEFORE_START_TIME	Le endTime d'au moins un segment de données se situe avant le segment startTime
AZ_EL_SEGMENT_TIME_S_OVERLAP	La plage de temps d'au moins un segment chevauche d'autres plages temporelles de segment
AZ_EL_SEGMENTS_OUT_OF_ORDER	Les segments ne sont pas ordonnés dans le temps
TIME_AZ_EL_ITEMS_OUT_OF_ORDER	Les TimeAzEl articles contenus dans un AzElSegment doivent être temporellement en ordre
AZ_EL_SEGMENT_REFERENCE_EPOCH_INVALID	L'époque de référence d'un segment n'est pas valide ou n'est pas correctement formatée
AZ_EL_SEGMENT_START_TIME_INVALID	L'heure de début comprise dans la plage horaire valide d'un segment ne commence pas après le premier segment
AZ_EL_SEGMENT_END_TIME_INVALID	L'heure de fin comprise dans la plage horaire valide d'un segment ne se termine pas après le dernier segment
AZ_EL_SEGMENT_VALID_TIME_RANGE_INVALID	La plage horaire valide pour un segment n'est pas valide

Code d'erreur	Message d'erreur
AZ_EL_SEGMENT_END_TIME_TOO_LATE	L'heure de fin d'un segment dépasse la durée maximale autorisée par rapport à l'époque de référence
AZ_EL_TOTAL_DURATION_EXCEEDED	La durée totale sur tous les segments dépasse la durée maximale autorisée de l'angle de pointage

Motif non valide : **TRAJECTORY_INVALID**

Ces erreurs se produisent lorsque l'éphéméride contient des données de trajectoire non valides, notamment des problèmes liés aux paramètres orbitaux, aux plages d'angles ou aux unités.

Code d'erreur	Message d'erreur
MOTION_MOYEN_INCORRECT	Le mouvement moyen des éphémérides TLE fournies est trop différent des éphémérides de référence les plus récentes. Remarque : Ground Station utilise les éphémérides Space-Track comme référence lors de la validation
TIME_AZ_EL_AZ_RADIAN_RANGE_INVALID	AzEl <u>az</u> doit être supérieur ou égal à $-\pi$ et inférieur ou égal à 2π radians
TIME_AZ_EL_EL_RADIAN_RANGE_INVALID	AzEl <u>el</u> doit être supérieur ou égal à $-\pi/2$ et inférieur ou égal à $\pi/2$ radians
TIME_AZ_EL_AZ_DEGREE_RANGE_INVALID	AzEl <u>az</u> doit être supérieur ou égal à -180 et inférieur ou égal à 360 degrés
TIME_AZ_EL_EL_DEGREE_RANGE_INVALID	AzEl <u>el</u> doit être supérieur ou égal à -90 degrés et inférieur ou égal à 90 degrés
TIME_AZ_EL_ANGLE_UNITS_INVALID	Unités AzEl angulaires non valides

Motif non valide : **KMS_KEY_INVALID**

Ces erreurs se produisent en cas de problème avec la clé AWS Key Management Service (KMS) utilisée pour chiffrer les données éphémères.

Code d'erreur	Message d'erreur
AUTORISATIONS_KMS_INSUFFISANTES	Ground Station ne dispose pas des autorisations suffisantes pour accéder à la clé KMS de cet éphémère

Motif non valide : **VALIDATION_ERROR**

Ces erreurs se produisent en cas de problèmes généraux de validation des données d'éphémères qui n'entrent pas dans les autres catégories spécifiques.

Code d'erreur	Message d'erreur
INTERNAL_ERROR	Une erreur interne s'est produite lors de la validation des éphémères
FORMAT_FICHER_INVALIDE	Le format de fichier éphémère n'est pas valide ou est endommagé. Vérifiez que le fichier est conforme au format attendu pour le type d'éphémère

Résoudre les problèmes liés aux contacts qui n'ont reçu aucune donnée

Il est possible qu'un contact semble avoir réussi, mais qu'il n'ait toujours pas reçu de données. Cela peut signifier que vous recevez des fichiers PCAP vides ou que vous ne recevez aucun fichier PCAP si vous utilisez la livraison de données S3. Cela peut se produire pour plusieurs raisons. Ce qui suit décrit certaines des causes et explique comment y remédier.

Configuration de liaison descendante incorrecte

Chaque contact recevant des données d'un satellite sera associé à un [Config d'antenne de liaison descendante](#) ou [Config de décodage/démodulation des signaux d'antenne de liaison descendante](#).

Si la configuration spécifiée ne correspond pas au signal transmis par un satellite, il ne AWS Ground Station sera pas en mesure de recevoir le signal transmis. Il en résultera qu'aucune donnée ne sera reçue par AWS Ground Station.

Pour résoudre ce problème, vérifiez que les configurations que vous utilisez correspondent au signal transmis par votre satellite. Par exemple, vérifiez que vous avez défini les bons paramètres de fréquence centrale, de bande passante, de polarisation et, si nécessaire, de démodulation et de décodage.

Manœuvre du satellite

Il arrive qu'un satellite effectue une manœuvre qui désactive temporairement certains de ses systèmes de communication. La manœuvre peut également modifier de manière significative la position du satellite dans le ciel. AWS Ground Station ne sera pas en mesure de recevoir un signal d'un satellite qui n'émet aucun signal, ou si l'éphéméride utilisée fait pointer l' AWS Ground Station antenne vers un endroit du ciel où le satellite n'est pas présent.

Si vous essayez de communiquer avec un satellite de diffusion public exploité par la NOAA, vous trouverez peut-être un message décrivant une panne ou une manœuvre sur la page des messages d'[alerte satellite](#) de la NOAA. Le message peut inclure une chronologie indiquant à quel moment la transmission de données devrait reprendre, ou cette chronologie peut être publiée dans un message suivant.

Si vous communiquez avec vos propres satellites, il est de votre responsabilité de comprendre le fonctionnement de vos satellites et l'impact que cela peut avoir sur la communication avec AWS Ground Station. Si vous effectuez une manœuvre qui aura un impact sur la trajectoire du satellite, cela peut inclure la fourniture de données d'éphémérides personnalisées mises à jour. Pour plus d'informations sur la fourniture de données d'éphémérides personnalisées, consultez [Comprendre comment AWS Ground Station utilise les éphémérides](#)

AWS Ground Station panne

Si cela AWS Ground Station entraîne l'échec ou l'annulation d'un contact, le statut du contact AWS Ground Station sera défini sur `AWS_FAILED`, ou `AWS_CANCELLED`. Pour plus d'informations sur le cycle de vie des contacts, consultez [Comprendre le cycle de vie des contacts](#). Dans certains cas, il AWS Ground Station peut y avoir une défaillance qui empêche l'envoi des données sur votre compte, mais qui n'entraîne pas le `AWS_CANCELLED` statut « `AWS_FAILED` » du contact. Dans ce cas, vous AWS Ground Station devez publier un événement spécifique au compte sur votre tableau de

bord AWS Health. Pour plus d'informations sur le tableau AWS de bord Health, consultez le [AWS Health User Guide](#).

Résoudre les problèmes de télémétrie

Utilisez les informations suivantes pour résoudre les problèmes courants liés à la télémétrie.

Problèmes de configuration courants

Erreurs d'autorisation IAM

Symptômes

Lorsque vous appelez `CreateConfig` pour créer un `TelemetrySinkConfig`, vous recevez un message d'erreur :

```
Unable to write to Kinesis Data Streams stream. Ensure that Ground Station has kinesis:PutRecord permissions for the given stream
```

Les causes

- Le rôle IAM spécifié dans le `TelemetrySinkConfig` ne dispose pas des autorisations requises pour écrire dans le flux Kinesis Data Streams.
- La politique de confiance relative au rôle IAM ne permet pas AWS Ground Station d'assumer ce rôle.
- L'ARN du flux Kinesis Data Streams dans `TelemetrySinkConfig` le est incorrect ou le flux n'existe pas.

Des solutions

1. Vérifiez que le rôle IAM existe et qu'il dispose des autorisations appropriées. Vérifiez [Étape 2 : Création d'un TelemetrySinkConfig](#) et assurez-vous que toutes les étapes ont été suivies.
2. Vérifiez qu'il AWS Ground Station peut assumer votre rôle IAM :

```
aws iam get-role --role-name GroundStationTelemetryRole
```

Vérifiez que la politique de confiance inclut `groundstation.amazonaws.com` un principe de service fiable.

3. Vérifiez que le rôle IAM dispose des autorisations Kinesis requises :

```
aws iam list-attached-role-policies --role-name GroundStationTelemetryRole
```

Assurez-vous que la politique inclut `kinesis:DescribeStreamkinesis:PutRecord`, et `kinesis:PutRecords` les autorisations pour votre stream.

4. Vérifiez que le flux Kinesis Data Streams existe et que l'ARN est correct :

```
aws kinesis describe-stream \  
  --stream-name your-stream-name \  
  --region us-east-2
```

5. Si vous utilisez un chiffrement géré par le client, vérifiez que le rôle IAM est `kms:GenerateDataKey` autorisé à accéder à votre clé. AWS KMS

PassRole erreurs d'autorisation

Symptômes

Lorsque vous appelez `CreateConfig`, vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à transmettre le rôle IAM.

Solution

Assurez-vous que votre utilisateur ou rôle IAM est `iam:PassRole` autorisé à utiliser le rôle IAM de télémétrie. Ajoutez la politique suivante à votre utilisateur ou à votre rôle :

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

Problèmes de configuration du flux Kinesis Data Streams

Symptômes

La transmission de la télémétrie échoue ou est intermittente.

Les causes

- La capacité du flux Kinesis Data Streams est insuffisante pour le débit de télémétrie.
- Le flux est utilisé par d'autres applications, ce qui entraîne une limitation des écritures.

Des solutions

1. Vérifiez l'état du stream :

```
aws kinesis describe-stream \  
  --stream-name your-stream-name \  
  --region us-east-2
```

2. Surveillez la limitation des écritures à l'aide CloudWatch de métriques :

```
aws cloudwatch get-metric-statistics \  
  --namespace AWS/Kinesis \  
  --metric-name WriteProvisionedThroughputExceeded \  
  --dimensions Name=StreamName,Value=your-stream-name \  
  --start-time 2025-12-08T00:00:00Z \  
  --end-time 2025-12-08T23:59:59Z \  
  --period 60 \  
  --statistics Sum \  
  --region us-east-2
```

3. Si un étranglement est détecté, pensez à :

- Passage en mode capacité à la demande pour une mise à l'échelle automatique.
- Utilisation d'un flux dédié pour la AWS Ground Station télémétrie.
- Si vous utilisez le mode provisionné, augmentez le nombre de partitions.

Problèmes de transmission de données télémétriques

Aucune donnée de télémétrie n'apparaît

Symptômes

Après avoir planifié un contact avec un profil de mission compatible avec la télémétrie, aucune donnée de télémétrie n'apparaît dans votre flux Kinesis Data Streams.

Causes possibles et solutions

La télémétrie n'est pas activée dans le profil de mission

Vérifiez que le profil de mission utilisé pour le contact inclut `telemetrySinkConfigArn` :

```
aws groundstation get-mission-profile \  
  --mission-profile-id 12345678-1234-1234-1234-123456789012 \  
  --region us-east-2
```

Vérifiez la sortie du `telemetrySinkConfigArn` champ. S'il n'est pas présent, la télémétrie n'est pas activée dans le profil de mission.

Problème d'autorisations liées aux rôles IAM

Consultez les étapes de résolution des problèmes liés aux autorisations IAM dans [Erreurs d'autorisation IAM](#)

Le flux Kinesis Data Streams n'existe pas ou se trouve dans la mauvaise région

Vérifiez que le flux existe dans la bonne région :

```
aws kinesis describe-stream \  
  --stream-name your-stream-name \  
  --region us-east-2
```

Le contact n'a pas encore commencé

La transmission de données télémétriques commence à l'heure de début du contact. Vérifiez que le contact a démarré en vérifiant son statut :

```
aws groundstation describe-contact \  
  --contact-id 12345678-1234-1234-1234-123456789012 \  
  --region us-east-2
```

```
--region us-east-2
```

Données de télémétrie intermittentes

Symptômes

Les données de télémétrie sont fournies de manière incohérente en cas de lacunes ou d'enregistrements manquants.

Causes possibles

- Problèmes de capacité ou de limitation des flux Kinesis Data Streams. Consultez [Problèmes de configuration du flux Kinesis Data Streams](#).
- Problèmes de connectivité réseau entre AWS Ground Station et votre flux Kinesis Data Streams.

Des solutions

- Surveillez les métriques des flux Kinesis Data Streams pour CloudWatch détecter toute limitation ou erreur.
- Assurez-vous que votre flux utilise le mode capacité à la demande ou dispose d'une capacité provisionnée suffisante.
- Utilisez un flux dédié pour la AWS Ground Station télémétrie afin d'éviter les conflits avec d'autres applications.

Problèmes liés au format des données

Erreurs d'analyse JSON

Symptômes

Votre application rencontre des erreurs lors de l'analyse des enregistrements de télémétrie au format JSON.

Des solutions

- Vérifiez le décodage Base64 : les données du flux Kinesis Data Streams sont codées en Base64. Assurez-vous de décoder les données avant de les analyser au format JSON. Pour de plus amples informations, veuillez consulter [Lecture de données depuis le flux Kinesis Data Streams](#).

- Vérifiez les enregistrements vides : AWS Ground Station peut envoyer des enregistrements de validation vides lors de la création d'un TelemetrySinkConfig. Votre application doit gérer les enregistrements vides ou mal formés avec élégance.
- Implémentez une analyse basée sur les versions : analysez d'abord les `telemetryVersion` champs `telemetryTypeAndVersion` et `telemetryType`, et afin de déterminer le schéma approprié pour chaque enregistrement.

Types ou versions de télémétrie inconnus

Symptômes

Votre application rencontre des types ou des versions de télémétrie qu'elle ne reconnaît pas.

Solution

Ce comportement est normal car de nouveaux types de télémétrie et de nouvelles versions de schéma peuvent être introduits au fil du temps. Votre candidature doit :

- Enregistrez les types et versions inconnus à des fins de surveillance.
- Poursuivre le traitement des types et des versions connus.
- Implémentez une gestion gracieuse des schémas inconnus.

Pour plus d'informations sur le versionnement des schémas, consultez [Versionnement et évolution des schémas](#).

Obtenir de l'aide

Si vous continuez à rencontrer des problèmes après avoir suivi les étapes de résolution des problèmes, contactez le AWS Support.

Informations à fournir

Lorsque vous contactez le support, fournissez les informations suivantes :

- Contacter IDs qui rencontre des problèmes
- ID de profil de mission utilisé
- TelemetrySinkConfig ARN
- Kinesis Data Streams (ARN)

- ARN du rôle IAM et politiques associées
- Messages d'erreur provenant de CloudWatch Logs ou de votre application
- Horodatage lorsque des problèmes sont survenus
- Mesures de résolution des problèmes déjà prises

Pour obtenir une AWS Ground Station assistance générale, consultez le [guide de AWS Ground Station l'utilisateur](#).

Quotas et limites

Vous pouvez consulter les régions prises en charge, leurs points de terminaison associés, ainsi que les quotas sur les points de [AWS Ground Station terminaison et les quotas](#).

Vous pouvez utiliser la [console Service Quotas](#), l'[API AWS](#) et l'[interface de ligne de commande AWS](#) afin de demander des augmentations de quotas, si nécessaire.

Modalités du service

Pour connaître les conditions de AWS Ground Station service, reportez-vous aux [conditions de service AWS](#).

Historique du document pour le guide de AWS Ground Station l'utilisateur

Le tableau suivant décrit les modifications importantes apportées à chaque version du guide de l'AWS Ground Station utilisateur.

Modification	Description	Date
Nouvelle fonctionnalité	Ajout de documentation pour les antennes AWS Ground Station dédiées. Pour plus d'informations, consultez la section Antennes AWS Ground Station dédiées .	14 avril 2026
Nouvelle fonctionnalité	Ajout de documentation pour l' UpdateContact API et la gestion des versions des contacts. Pour plus d'informations, voir Mettre à jour les contacts et gérer les versions des contacts .	14 avril 2026
Nouvelle fonctionnalité	Ajout de documentation pour le ListAntennas et ListGroundStationReservations APIs. Pour plus d'informations, consultez les sections AWS Ground Station Emplacements et Afficher les réservations de stations au sol .	14 avril 2026
Mise à jour de documentation	Des fonctionnalités supplémentaires ont été ajoutées à l' CancelContact API et incluent des informations sur	10 décembre 2025

ces fonctionnalités et leurs implications en matière de mesure. Pour plus d'informations, voir [Comprendre la mesure des contacts](#).

[Mise à jour de documentation](#)

Il a été précisé que les CloudWatch métriques sont émises dans la région associée à la station au sol du contact. Corrige les liens brisés.

2 décembre 2025

[Politique AWS gérée mise à jour](#)

AWS Ground Station a mis à jour la politique gérée `AWSGroundStationAgentInstancePolicy` afin d'inclure des autorisations supplémentaires pour récupérer les réponses aux URLs tâches. Pour plus d'informations, voir les [AWS Ground Station mises à jour des politiques AWS gérées](#).

13 novembre 2025

[Nouvelle fonctionnalité](#)

Le guide de l'utilisateur a été mis à jour pour inclure les éphémérides d'élévation d'azimut. Pour plus d'informations, voir [Fournir des données d'éphémérides d'altitude azimutale](#)

22 octobre 2025

Mise à jour de documentation	La livraison de données entre régions ne nécessite plus de configuration ou d'approbations spéciales. Pour plus d'informations, voir Utiliser la livraison de données entre régions .	11 septembre 2025
Mise à jour de documentation	Clarification ajoutée sur l'utilisation des contacts des ressources configurées.	4 avril 2025
Nouvelle fonctionnalité	Le guide de l'utilisateur a été mis à jour pour inclure le jumeau AWS Ground Station numérique.	6 août 2024
Mise à jour de documentation	De nombreuses sections du guide de l'utilisateur ont été mises à jour, notamment de nouveaux diagrammes, des exemples, etc.	18 juillet 2024
Mise à jour de documentation	Ajout d'un flux RSS au guide de l'utilisateur.	18 juillet 2024
Mise à jour de documentation	Divisez le guide de l'utilisateur de l'AWS Ground Station agent en un guide de l'utilisateur distinct.	18 juillet 2024
Nouvelle fonctionnalité	Les contacts peuvent désormais être programmés jusqu'à 30 secondes en dehors des plages horaires de visibilité. Les temps de visibilité sont inclus dans DescribeContact les réponses.	26 mars 2024

Mise à jour de documentation	Organisation améliorée et ajout de la section « Sélection des instances EC2 et planification du processeur ».	6 mars 2024
Mise à jour de documentation	Ajout de nouvelles bonnes pratiques au guide de l'utilisateur de l' AWS Ground Station agent pour exécuter des services et des processus parallèlement à l' AWS Ground Station agent.	23 février 2024
Mise à jour de documentation	Ajout de la page des notes de version de l'agent.	21 février 2024
Mise à jour du modèle	Ajout de la prise en charge d'un sous-réseau public distinct dans le DataDelivery modèle DirectBroadcastSatelliteWbDigiF Ec 2.	14 février 2024
Mise à jour de documentation	Ajout d'une référence à AWS Notifications des utilisateurs dans la documentation de surveillance.	6 août 2023
Mise à jour de documentation	Ajout d'instructions pour étiqueter les satellites avec un nom à afficher dans la AWS Ground Station console.	26 juillet 2023
Nouvelle fonctionnalité	Ajout du guide de l'utilisateur de l' AWS Ground Station agent pour la sortie de Wideband DigiF Data Delivery.	12 avril 2023

Nouvelle politique AWS gérée	AWS Ground Station a ajouté une nouvelle politique nommée AWSGroundStationAgentInstancePolicy.	12 avril 2023
Nouvelle fonctionnalité	Mise à jour du guide de l'utilisateur pour la sortie de CPE Preview.	9 novembre 2022
Nouvelle politique AWS gérée	AWS Ground Station a ajouté le AWSService RoleForGroundStationDataflowEndpointGroup service-linked-role (SLR) qui inclut une nouvelle politique nommée AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy.	2 novembre 2022
Nouvelle fonctionnalité	Mise à jour du guide de l'utilisateur pour inclure l'intégration avec AWS CLI.	17 avril 2020
Nouvelle fonctionnalité	Le guide de l'utilisateur a été mis à jour pour inclure l'intégration avec CloudWatch Metrics.	24 février 2020
Nouveau modèle	Satellites de diffusion publics (AquaSnppJpss modèle) ajouté au guide de AWS Ground Station l'utilisateur.	19 février 2020
Nouvelle fonctionnalité	Mise à jour du guide de l'utilisateur pour inclure la transmission des données entre régions.	5 février 2020

Mise à jour de documentation	Exemples et descriptions mis à jour pour la surveillance AWS Ground Station avec CloudWatch Events.	4 février 2020
Mise à jour de documentation	Les emplacements des modèles ont été mis à jour et les sections Démarrez et Dépannage ont été révisées.	19 décembre 2019
Nouvelle section de résolution des problèmes	Section de résolution des problèmes ajoutée au guide de AWS Ground Station l'utilisateur.	7 novembre 2019
Nouveau sujet de mise en route	Mise à jour de la rubrique Getting Started, qui inclut les CloudFormation modèles les plus récents.	1 juillet 2019
Version Kindle	Version Kindle publiée du guide de l'AWS Ground Station utilisateur.	20 juin 2019
Nouveau guide et service	Il s'agit de la version initiale AWS Ground Station et du guide de AWS Ground Station l'utilisateur.	23 mai 2019

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le [AWS glossaire](#) dans la Glossaire AWS référence.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.