aws

Guide du développeur

Amazon Glacier

Version de l'API 2012-06-01

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

Amazon Glacier: Guide du développeur

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

Les marques commerciales et la présentation commerciale d’Amazon ne peuvent pas être utilisées en relation avec un produit ou un service extérieur à Amazon, d’une manière susceptible d’entraîner une confusion chez 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

..... x
Qu'est-ce qu'Amazon Glacier ? 1
Utilisez-vous actuellement Amazon Glacier ? 1
Modèle de données 3
Coffre-fort 3
Archivage 4
Tâche 5
Configuration des notifications 6
Opérations prises en charge 7
Opérations de coffre 7
Opérations d'archivage 7
Tâches 7
Accès à Amazon Glacier 8
Régions et points de terminaison 8
Démarrage 9
Étape 1 : Avant de commencer 10
Configurez un Compte AWS 10
Téléchargez le AWS SDK approprié 12
Étape 2 : Créer un coffre 13
Étape 3 : Charger une archive dans un coffre 15
Chargement d'une archive à l'aide de Java 16
Chargement d'une archive à l'aide de .NET 21
Étape 4 : Télécharger une archive à partir d'un coffre 23
Téléchargement d'une archive à l'aide de Java 24
Téléchargement d'une archive à l'aide de .NET 26
Étape 5 : Supprimer une archive d'un coffre 28
Sections connexes 29
Suppression d'une archive à l'aide de Java 29
Suppression d'une archive à l'aide de .NET 30
Suppression d'une archive à l'aide du AWS CLI 31
Étape 6 : Supprimer un coffre 34
Comment procéder ensuite ? 36
Utilisation des coffres 37
Opérations du coffre-fort dans Amazon Glacier 38

Création et suppression des coffres 38
Extraction de métadonnées de coffre 38
Téléchargement d'un inventaire de coffre 39
Configuration de notifications de coffre 39
Création d'un coffre 40
Création d'un coffre à l'aide de Java 41
Création d'un coffre à l'aide de .NET 44
Création d'un coffre à l'aide de REST 49
Création d'un coffre à l'aide de la console 49
Création d'un coffre-fort à l'aide du AWS CLI 49
Extraction de métadonnées de coffre 51
Extraction des métadonnées de coffre à l'aide de Java 51
Extraction de métadonnées de coffre à l'aide du kit .NET 54
Extraction de métadonnées de coffre à l'aide de REST 57
Récupération des métadonnées du coffre-fort à l'aide du AWS CLI 57
Téléchargement d'un inventaire de coffre 58
A propos de l'inventaire 60
Téléchargement d'un inventaire de coffre à l'aide de Java 61
Téléchargement d'un inventaire de coffre à l'aide du kit .NET 69
Téléchargement d'un inventaire de coffre à l'aide de REST 77
Téléchargement d'un inventaire du coffre-fort à l'aide du AWS CLI 77
Configuration de notifications de coffre 80
Concepts généraux 80
Configuration des notifications de coffre à l'aide de Java 82
Configuration de notifications de coffre à l'aide de .NET 85
Configuration de notifications de coffre à l'aide de l'API REST 88
Configuration des notifications de coffre à l'aide de la console 89
Configuration des notifications de coffre à l'aide de l'interface CLI 91
Suppression d'un coffre 93
Suppression d'un coffre à l'aide de Java 93
Suppression d'un coffre à l'aide de .NET 95
Suppression d'un coffre à l'aide de REST 96
Suppression d'un coffre vide à l'aide de la console 96
Suppression d'un coffre-fort à l'aide du AWS CLI 97
Balisage des coffres 101
Marquage des coffres-forts à l'aide de la console Amazon Glacier 101

Marquer les coffres-forts à l'aide du AWS CLI 103
Marquage des coffres-forts à l'aide de l'API Amazon Glacier 104
Sections connexes 104
Verrouillage du coffre-fort 105
Présentation de la fonction de verrouillage de coffre 105
Verrouillage de coffre à l'aide de l'API 106
Verrouillage d'un coffre à l'aide de l'interface CLI 108
Verrouillage d'un coffre à l'aide de la console 110
Utilisation des archives 112
Opérations d'archivage 113
Chargement d'une archive 113
Recherche d'une archive 113
Téléchargement d'une archive 113
Suppression d'une archive 114
Mise à jour d'une archive 114
Conservation des métadonnées d'archive côté client 114
Chargement d'une archive 115
Options de chargement d'une archive 115
Chargement d'une archive en une seule opération 116
Chargement d'archives volumineuses en plusieurs parties 127
Téléchargement d'une archive 146
Récupération des archives 146
Téléchargement d'une archive à l'aide de Java 151
Téléchargement d'une archive à l'aide de .NET 169
Téléchargement d'une archive volumineuse à l'aide de Python 185
Téléchargement d'une archive à l'aide de REST 193
Téléchargement d'une archive à l'aide du AWS CLI 194
Suppression d'une archive 197
Suppression d'une archive à l'aide de Java 199
Suppression d'une archive à l'aide de .NET 201
Suppression d'une archive à l'aide de REST 204
Suppression d'une archive à l'aide du AWS CLI 204
À l'aide du AWS SDKs 208
AWS Bibliothèques SDK pour Java et .NET 208
Qu'est-ce qu'une API de bas niveau ? 208
Qu'est-ce qu'une API de haut niveau ? 209

Quand utiliser l'API de haut niveau et celle de bas niveau 209
Travailler avec AWS SDKs 210
À l'aide du AWS SDK pour Java 211
Utilisation de l'API de bas niveau 211
Utilisation de l'API de haut niveau 212
Exécution d'exemples Java à l'aide d'Eclipse 213
Définition du point de terminaison 214
À l'aide du AWS SDK pour .NET 215
Utilisation de l'API de bas niveau 215
Utilisation de l'API de haut niveau 216
Exécution d'exemples .NET 217
Définition du point de terminaison 218
Exemples de code 219
Principes de base 220
Bonjour Amazon Glacier 221
Actions 222
Scénarios 289
Archivage d’un fichier, obtention de notifications et lancement d’une tâche 290
Obtenir le contenu d’une archive et supprimer l’archive 296
Sécurité 302
Protection des données 303
Chiffrement des données 303
Gestion des clés 304
Confidentialité du trafic inter-réseaux 304
Gestion de l’identité et des accès 305
Public ciblé 305
Authentification par des identités 306
Gestion de l’accès à l’aide de politiques 307
Comment Amazon Glacier fonctionne avec IAM 309
Exemples de politiques basées sur l’identité 316
Exemples de stratégies basées sur les ressources 323
Résolution des problèmes 326
Référence des autorisations d'API Amazon Glacier 328
Journalisation et surveillance 336
Validation de la conformité 337
Résilience 339

Sécurité de l'infrastructure 340
Points de terminaison d'un VPC 340
Stratégies d'extraction de données 341
Choisir une politique de récupération des données Amazon Glacier 341
Stratégie Free Tier Only 342
Stratégie Max Retrieval Rate 343
Stratégie No Retrieval Limit 343
Utilisation de la console Amazon Glacier pour configurer une politique de récupération des données 344
Utilisation de l'API Amazon Glacier pour configurer une politique de récupération des données 344
Utilisation de l'API REST Amazon Glacier pour configurer une politique de récupération des données 345
Utilisation du AWS SDKs pour configurer une politique de récupération de données 345
Balisage des ressources 346
Principes de base du balisage 346
Restrictions liées aux balises 347
Suivi des coûts à l'aide du balisage 347
Gestion du contrôle d'accès avec le balisage 348
Sections connexes 348
Journalisation des audits avec AWS CloudTrail 349
Informations sur Amazon Glacier dans CloudTrail 349
Comprendre les entrées du fichier journal Amazon Glacier 350
Référence API 354
En-têtes de demande communs 355
En-têtes de réponse communs 358
Signature des requêtes 359
Exemple de calcul de signature 360
Calcul des Signatures pour les opérations de streaming 362
Calcul des totaux de contrôle 364
Exemple de hachage d'arborescence 1 : Chargement d'une archive dans le cadre d'une seule demande 366
Exemple de hachage d'arborescence 2 : chargement d'une archive à l'aide d'un chargement partitionné 367
Calcul du hachage d'arborescence d'un fichier 368
Réception des totaux de contrôle lors du téléchargement de données 378

Réponses d'erreur 381
Exemple 1 : Description d'une demande de tâche avec un ID de tâche qui n'existe pas 384
Exemple 2 : Demande d'établissement d'une liste des tâches avec une valeur non valide pour le paramètre de demande 386
Opérations de coffre 386
Interruption du verrouillage du coffre 387
Ajout de balises à un coffre 390
Créer un coffre-fort 394
Terminer le verrouillage du coffre-fort 397
Supprimer le coffre-fort 400
Suppression de la stratégie d'accès au coffre 403
Supprimer les notifications de coffre 406
Description de coffre 409
Obtention de la stratégie d'accès au coffre 413
Obtenir le verrouillage de coffre 417
Obtenir les notifications de coffre 422
Lancer le verrouillage du coffre-fort 425
Affichage des balises d'un coffre 430
Affichage des coffres 433
Suppression des balises d'un coffre 440
Définir la stratégie d'accès aux coffres 444
Définition d'une configuration de notification de coffre 447
Opérations d'archivage 451
Suppression d'une archive 451
Chargement d'archive 454
Opérations de chargement partitionné 460
Interruption du chargement partitionné 461
Achèvement du chargement partitionné 464
Lancement du chargement partitionné 469
Liste des parties 474
Liste des chargements partitionnés 482
Chargement d'une partie 489
Opérations de tâche 496
Description de la tâche 496
Génération de sortie de tâche 507
Lancement d'une tâche 518

Affichage des tâches ... 529
Types de données utilisés dans les opérations de tâche ... 539
CSVInput ... 540
CSVOutput ... 541
Chiffrement ... 543
GlacierJobDescription ... 544
Octroi ... 547
Bénéficiaire ... 548
InputSerialization ... 549
InventoryRetrievalJobInput ... 550
jobParameters ... 551
OutputLocation ... 554
OutputSerialization ... 554
Emplacement S3 ... 555
SelectParameters ... 557
Opérations de récupération des données ... 558
Obtention de la stratégie d'extraction de données ... 558
Répertorier la capacité provisionnée ... 562
Acheter la capacité allouée ... 566
Définition de stratégie d'extraction de données ... 569
Historique du document ... 575
Mises à jour antérieures ... 576
AWS Glossaire ... 579

Cette page est réservée aux clients existants du service Amazon Glacier utilisant Vaults et l'API REST d'origine datant de 2012.

Si vous recherchez des solutions de stockage d'archives, nous vous recommandons d'utiliser les classes de stockage Amazon Glacier dans Amazon S3, S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval et S3 Glacier Deep Archive. Pour en savoir plus sur ces options de stockage, consultez les classes de stockage Amazon Glacier.

Amazon Glacier (service autonome d'origine basé sur un coffre-fort) n'accepte plus de nouveaux clients. Amazon Glacier est un service autonome doté de son propre service APIs qui stocke les données dans des coffres-forts et qui se distingue des classes de stockage Amazon S3 et Amazon S3 Glacier. Vos données existantes resteront sécurisées et accessibles indéfiniment dans Amazon Glacier. Aucune migration n'est requise. Pour un stockage d'archivage à long terme et à faible coût AWS , nous recommandons les classes de stockage Amazon S3 Glacier, qui offrent une expérience client supérieure avec des compartiments S3, une Région AWS disponibilité totale APIs, des coûts réduits et AWS une intégration des services. Si vous souhaitez des fonctionnalités améliorées, envisagez de migrer vers les classes de stockage Amazon S3 Glacier en suivant nos conseils relatifs aux AWS solutions pour le transfert de données depuis les coffres-forts Amazon Glacier vers les classes de stockage Amazon S3 Glacier.

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.

Qu'est-ce qu'Amazon Glacier ?

Si vous utilisez actuellement le service Amazon Glacier et que vous souhaitez en savoir plus, vous trouverez les informations dont vous avez besoin dans ce guide. Amazon Glacier est un service sécurisé et durable d'archivage de données à faible coût et de sauvegarde à long terme à l'aide de coffres-forts. Pour plus d'informations sur la tarification des services Amazon Glacier, consultez la section Tarification Amazon Glacier.

Rubriques

- Utilisez-vous actuellement Amazon Glacier ?
- Modèle de données Amazon Glacier
- Opérations prises en charge dans Amazon Glacier
- Accès à Amazon Glacier

Utilisez-vous actuellement Amazon Glacier ?

> **Note**
>
> Cette section concerne le service Amazon Glacier. Si vous utilisez actuellement les classes de stockage Amazon S3 Glacier (S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval et S3 Glacier Deep Archive), consultez la section Classes de stockage pour les objets d'archivage dans le guide de l'utilisateur Amazon S3.

Si vous utilisez actuellement le service Amazon Glacier et souhaitez en savoir plus, nous vous recommandons de commencer par lire les sections suivantes :

- Qu'est-ce qu'Amazon Glacier ? — Le reste de cette section décrit le modèle de données sous-jacent, les opérations qu'il prend en charge et celles AWS SDKs que vous pouvez utiliser pour interagir avec le service.
- Démarrage : la section Commencer à utiliser Amazon Glacier vous accompagne tout le long des processus de création d'un coffre, de chargement d'archives, de création de tâches de téléchargement d'archives, d'extraction de la sortie d'une tâche et de suppression d'archives.

> **Important**
>
> Amazon Glacier fournit une console. Cependant, toute opération d'archivage, telle que le chargement, le téléchargement ou la suppression, nécessite que vous utilisiez le AWS Command Line Interface (AWS CLI) ou que vous écriviez du code. Il n'y a aucune prise en charge de la console pour les opérations d'archivage. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS CLI code pour effectuer des demandes, soit en utilisant directement l'API REST, soit en utilisant le AWS SDKs.
>
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la AWS CLI référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

Au-delà de la section consacrée à la mise en route, vous souhaiterez probablement en savoir plus sur les opérations d'Amazon Glacier. Les sections suivantes fournissent des informations détaillées sur l'utilisation d'Amazon Glacier à l'aide de l'API REST et de l'API AWS SDKs pour Java et Microsoft .NET :

- Utilisation du AWS SDKs avec Amazon Glacier

  Cette section fournit une vue d'ensemble des éléments AWS SDKs utilisés dans les différents exemples de code de ce guide. Vous pouvez la consulter lorsque vous lisez les sections suivantes. Il inclut une présentation du haut niveau et du bas niveau APIs qu'ils SDKs offrent, les circonstances dans lesquelles les utiliser, ainsi que les étapes courantes pour exécuter les exemples de code fournis dans ce guide.
- Utilisation des coffres-forts dans Amazon Glacier

  Cette section détaille les différentes opérations de coffre, telles que la création d'un coffre, l'extraction des métadonnées d'un coffre, l'utilisation de tâches d'extraction d'inventaire d'un coffre et la configuration de notifications d'un coffre. Outre l'utilisation de la console Amazon Glacier, vous pouvez l'utiliser AWS SDKs pour diverses opérations de coffre-fort. Cette section décrit l'API et fournit des exemples pratiques à l'aide AWS SDK pour Java des AWS SDK pour .NET.
- Utilisation des archives dans Amazon Glacier

Cette section détaille certaines opérations liées aux archives, comme le chargement d'une archive en une seule demande ou l'utilisation d'une opération de chargement partitionné pour charger des archives volumineuses en plusieurs parties. Elle explique également comment créer des tâches en vue de télécharger des archives de manière asynchrone. Elle fournit des exemples avec AWS SDK pour Java et AWS SDK pour .NET.

- Référence d'API pour Amazon Glacier

  Amazon Glacier est un RESTful service. Cette section décrit les opérations REST, notamment la syntaxe, ainsi que les exemples de demandes et de réponses pour toutes les opérations. Les bibliothèques du AWS SDK intègrent cette API, simplifiant ainsi vos tâches de programmation.

Modèle de données Amazon Glacier

Les principaux composants du modèle de données Amazon Glacier incluent les coffres-forts et les archives. Amazon Glacier est un service Web basé sur REST. En termes REST, les coffres et les archives sont les ressources. En outre, le modèle de données Amazon Glacier inclut des ressources de configuration des tâches et des notifications. Ces ressources viennent compléter les ressources de base.

Rubriques

- Coffre-fort
- Archivage
- Tâche
- Configuration des notifications

Coffre-fort

Dans Amazon Glacier, un coffre est un conteneur destiné au stockage d'archives. Un coffre est comparable à un compartiment Amazon S3. Lorsque vous créez un coffre-fort, vous spécifiez un nom et choisissez l' Région AWS endroit où vous souhaitez le créer.

Chaque ressource de coffre possède une adresse unique. Son format général est :

```
https://region-specific-endpoint/account-id/vaults/vault-name
```

Par exemple, supposez que vous créez un coffre (`examplevault`) dans la région USA Ouest (Oregon) de votre compte avec l'ID 111122223333. Vous pouvez attribuer une adresse à ce coffre en utilisant l'URI suivant :

```
https://glacier.us-west-2.amazonaws.com/111122223333/vaults/examplevault
```

Voici la signification des différents composants de l'URI :

- `glacier.us-west-2.amazonaws.com` identifie la région USA Ouest (Oregon).
- `111122223333`est l' Compte AWS identifiant qui possède le coffre.
- `vaults` fait référence à la collection de coffres dont est propriétaire le Compte AWS.
- `examplevault` identifie un coffre spécifique dans la collection de coffres.

Et Compte AWS peut créer des coffres-forts dans tous les coffres-forts pris en charge Région AWS. Pour la liste des produits pris en charge Régions AWS, voirAccès à Amazon Glacier. Dans une région, un compte doit utiliser des noms de coffre uniques. Et Compte AWS peut créer des coffres-forts portant le même nom dans différentes régions.

Vous pouvez stocker un nombre illimité d'archives dans un coffre. En fonction de vos besoins commerciaux ou de ceux de votre application, vous pouvez stocker ces archives dans un seul coffre ou plusieurs coffres.

Amazon Glacier prend en charge diverses opérations de coffre-fort. Les opérations liées aux coffres portent sur une région spécifique. Par exemple, lorsque vous créez un coffre, la création est effectuée pour une région spécifique. Lorsque vous demandez une liste de coffres-forts, vous la demandez à une personne spécifique Région AWS, et la liste qui en résulte inclut uniquement les coffres-forts créés dans cette région spécifique.

Archivage

Une archive peut être constituée de tout type de données : une photo, une vidéo ou un document. Une archive est similaire à un objet Amazon S3 et constitue l'unité de stockage de base dans Amazon Glacier. Chaque archive est associée à un ID unique et une description en option. Vous ne pouvez spécifier cette description facultative qu'au cours du chargement d'une archive. Amazon Glacier attribue un identifiant unique à l'archive Région AWS dans laquelle elle est stockée.

Chaque archive possède une adresse unique. En général, sont format est le suivant :

```
https://region-specific-endpoint/account-id/vaults/vault-name/archives/archive-id
```

Voici un exemple d'URI qui représente une archive stockée dans le coffre `examplevault` dans la région USA Ouest (Oregon) pour le compte 111122223333 :

```
https://glacier.us-west-2.amazonaws.com/111122223333/vaults/
examplevault/archives/NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
```

Vous pouvez stocker un nombre illimité d'archives dans un coffre.

Tâche

Une tâche Amazon Glacier peut récupérer une archive ou obtenir l'inventaire d'un coffre-fort.

La récupération d'archives et d'inventaires de coffres-forts (listes d'archives) est une opération asynchrone dans Amazon Glacier, au cours de laquelle vous lancez d'abord une tâche, puis vous téléchargez le résultat de la tâche une fois qu'Amazon Glacier l'a terminée.

> **Note**
>
> Amazon Glacier propose une solution d'archivage des données par stockage à froid. Si votre application a besoin d'une solution de stockage qui demande une extraction de données en temps réel, vous pouvez envisager d'utiliser Amazon S3. Pour plus d'informations, consultez Amazon Simple Storage Service (Amazon S3).

Pour lancer une tâche d'inventaire de coffre, vous fournissez un nom de coffre. Les tâches d'extraction d'archive requièrent le nom du coffre et l'ID de l'archive. Vous pouvez également indiquer une description facultative de la tâche pour permettre d'identifier les tâches.

Les tâches d'extraction d'archive et d'inventaire de coffre sont associées à un coffre. Un coffre peut avoir plusieurs tâches en cours à un moment donné. Lorsque vous envoyez une demande d'emploi (initiez une tâche), Amazon Glacier vous renvoie un numéro de tâche pour suivre la tâche. Chaque tâche est identifiée de façon unique par un URI au format suivant :

```
https://region-specific-endpoint/account-id/vaults/vault-name/jobs/job-id
```

Voici l'exemple d'une tâche associée à un coffre `examplevault` dans la région USA Ouest (Oregon) pour le compte 111122223333.

```
https://glacier.us-west-2.amazonaws.com/111122223333/vaults/examplevault/jobs/
HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
```

Pour chaque tâche, Amazon Glacier conserve des informations, telles que le type de tâche, la description, la date de création, la date d'achèvement et le statut de la tâche. Vous pouvez obtenir des informations sur une tâche spécifique ou obtenir la liste de toutes vos tâches associées à un coffre. La liste des tâches renvoyées par Amazon Glacier inclut toutes les tâches en cours et celles récemment terminées.

Configuration des notifications

Comme l'exécution des tâches prend du temps, Amazon Glacier propose un mécanisme de notification qui vous avertit lorsqu'une tâche est terminée. Vous pouvez configurer un coffre de sorte qu'il envoie une notification à une rubrique Amazon Simple Notification Service (Amazon SNS) lorsqu'une tâche se termine. Vous pouvez spécifier une seule rubrique Amazon SNS par coffre dans la configuration de notifications.

Amazon Glacier stocke la configuration des notifications sous forme de document JSON. Le texte suivant est un exemple de configuration de notifications :

```
{
   "Topic": "arn:aws:sns:us-west-2:111122223333:mytopic",
   "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Les configurations de notifications sont associées à des coffres ; il ne peut en exister qu'une seule par coffre. Chaque ressource de configuration de notifications est identifiée de façon unique par un URI au format suivant :

```
https://region-specific-endpoint/account-id/vaults/vault-name/notification-
configuration
```

Amazon Glacier prend en charge les opérations de définition, d'obtention et de suppression d'une configuration de notification. Lorsque vous supprimez une configuration de notifications, aucune notification n'est envoyée à la fin d'une opération d'extraction de données dans le coffre.

Opérations prises en charge dans Amazon Glacier

Pour utiliser les coffres-forts et les archives (voirModèle de données Amazon Glacier), Amazon Glacier prend en charge un ensemble d'opérations. Parmi toutes les opérations prises en charge, seules les opérations suivantes sont asynchrones :

- Extraction d'une archive
- Extraction d'un inventaire de coffre (liste des archives)

Ces opérations nécessitent de commencer par lancer la tâche, puis de télécharger la sortie de la tâche. Les sections suivantes résument les opérations d'Amazon Glacier.

Opérations de coffre

Amazon Glacier propose des opérations permettant de créer et de supprimer des coffres-forts. Vous pouvez obtenir une description de coffre pour un coffre spécifique ou pour tous les coffres d'une Région AWS. La description du coffre-fort fournit des informations telles que la date de création, le nombre d'archives qu'il contient, la taille totale en octets utilisée par toutes les archives du coffre-fort et la date à laquelle Amazon Glacier a généré l'inventaire du coffre-fort. Amazon Glacier propose également des opérations permettant de définir, de récupérer et de supprimer une configuration de notification sur le coffre-fort. Pour de plus amples informations, veuillez consulter Utilisation des coffres-forts dans Amazon Glacier.

Opérations d'archivage

Amazon Glacier propose des opérations vous permettant de télécharger et de supprimer des archives. Vous ne pouvez pas mettre à jour une archive existante ; vous devez supprimer l'archive existante et charger une nouvelle archive. Chaque fois que vous chargez une archive, Amazon Glacier génère un nouvel identifiant d'archive. Pour de plus amples informations, veuillez consulter Utilisation des archives dans Amazon Glacier.

Tâches

Vous pouvez lancer une tâche Amazon Glacier pour effectuer une extraction sur une archive ou obtenir l'inventaire d'un coffre-fort.

Les types de tâches Amazon Glacier sont les suivants :

- `archive-retrieval` : extrait une archive.

Pour de plus amples informations, veuillez consulter Téléchargement d'une archive dans Amazon Glacier.

- `inventory-retrieval` : inventorie un coffre.

  Pour de plus amples informations, veuillez consulter Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

Accès à Amazon Glacier

Amazon Glacier est un service RESTful Web qui utilise HTTP et HTTPS comme protocole de transport et la notation d'JavaScript objet (JSON) comme format de sérialisation des messages. Le code de votre application peut envoyer des requêtes directement à l'API du service Web Amazon Glacier. Lorsque vous utilisez l'API REST directement, vous devez écrire le code nécessaire pour signer et authentifier vos demandes. Pour plus d'informations sur l'API, consultez Référence d'API pour Amazon Glacier.

Vous pouvez également simplifier le développement d'applications en utilisant le module AWS SDKs qui enveloppe les appels d'API REST Amazon Glacier. Vous fournissez vos informations d'identification et ces bibliothèques s'occupent de l'authentification et de la signature des demandes. Pour plus d'informations sur l'utilisation du AWS SDKs, consultez Utilisation du AWS SDKs avec Amazon Glacier.

Amazon Glacier fournit également une console. Cependant, toutes les opérations d'archivage et de travail nécessitent que vous écriviez du code et que vous fassiez des requêtes en utilisant directement l'API REST ou les bibliothèques wrapper du AWS SDK. Pour accéder à la console Amazon Glacier, rendez-vous sur la console Amazon Glacier.

Régions et points de terminaison

Vous créez un coffre dans un endroit spécifique Région AWS. Vous envoyez toujours vos demandes Amazon Glacier à un point de terminaison spécifique à un Région AWS. Pour obtenir la liste des solutions Régions AWS prises en charge par Amazon Glacier, consultez la section Points de terminaison et quotas Amazon Glacier dans le manuel de référence AWS général.

Commencer à utiliser Amazon Glacier

Vous pouvez commencer à utiliser Amazon Glacier (Amazon Glacier) en utilisant des coffres-forts et des archives. Un coffre est un conteneur destiné à stocker les archives ; une archive est un quelconque objet, par exemple une photo, une vidéo ou un document, que vous stockez dans un coffre. Une archive est l'unité de stockage de base dans Amazon Glacier. Cet exercice de démarrage fournit des instructions vous permettant d'explorer les opérations de base d'Amazon Glacier sur les coffres-forts et les archives. Pour plus d'informations sur ces ressources, consultez la section Modèle de données Amazon Glacier.

Dans l'exercice de démarrage, vous allez créer un coffre, charger et télécharger une archive, puis supprimer cette archive et ce coffre. Toutes ces opérations peuvent être effectuées par programme. Toutefois, l'exercice de mise en route utilise la console de gestion Amazon Glacier pour créer et supprimer un coffre-fort. Pour charger et télécharger une archive, cette section de démarrage utilise l'API de haut niveau pour le AWS SDK pour Java et le AWS SDK pour .NET. L'API de haut niveau fournit une expérience de programmation simplifiée lorsque vous travaillez avec Amazon Glacier. Pour plus d'informations sur l'utilisation de l'API de haut niveau avec le AWS SDKs, consultezUtilisation du AWS SDKs avec Amazon Glacier.

> **Important**
>
> Amazon Glacier fournit une console. Cependant, toute opération d'archivage, telle que le chargement, le téléchargement ou la suppression, nécessite que vous utilisiez la AWS Command Line Interface (CLI) ou que vous écriviez du code. Il n'y a aucune prise en charge de la console pour les opérations d'archivage. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS CLI code pour effectuer des demandes, soit en utilisant directement l'API REST, soit en utilisant le AWS SDKs.
>
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la AWS CLI référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

Cet exercice de mise en route fournit des exemples de code en Java et C# qui vous permettent de charger et de télécharger une archive. La dernière section de l'exercice de démarrage décrit les étapes qui vous permettront d'en savoir plus sur l'expérience des développeurs avec Amazon Glacier.

Rubriques

- Étape 1 : Avant de commencer à utiliser Amazon Glacier
- Étape 2 : créer un coffre-fort dans Amazon Glacier
- Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier
- Étape 4 : télécharger une archive depuis un coffre-fort dans Amazon Glacier
- Étape 5 : Supprimer une archive d'un coffre-fort dans Amazon Glacier
- Étape 6 : supprimer un coffre-fort dans Amazon Glacier
- Comment procéder ensuite ?

Étape 1 : Avant de commencer à utiliser Amazon Glacier

Avant de commencer cet exercice, vous devez vous inscrire à un Compte AWS (si ce n'est pas déjà fait), puis télécharger l'un des AWS SDKs. Consultez les sections suivantes pour obtenir des instructions.

Rubriques

- Configuration d'un utilisateur Compte AWS et d'un administrateur
- Téléchargez le AWS SDK approprié

Configuration d'un utilisateur Compte AWS et d'un administrateur

Si ce n'est pas déjà fait, vous devez créer un compte Compte AWS et créer un compte administrateur.

Pour effectuer la configuration, suivez les instructions contenues dans les rubriques suivantes.

Configuration Compte AWS et création d'un utilisateur administrateur

Inscrivez-vous pour AWS

Lorsque vous vous inscrivez à Amazon Web Services (AWS), vous êtes automatiquement Compte AWS inscrit à tous les services AWS, y compris Amazon Glacier. Seuls les services que vous utilisez vous sont facturés. Pour plus d'informations sur les taux d'utilisation d'Amazon Glacier, consultez la page de tarification d'Amazon Glacier.

Si vous en avez déjà un Compte AWS, passez directement àTéléchargez le AWS SDK approprié. Si vous n'en avez pas Compte AWS, utilisez la procédure suivante pour en créer un.

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 saisirez 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 AWSest 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.

Afin de créer un utilisateur administrateur, choisissez l'une des options suivantes :

Choisissez un moyen de gérer votre administrateur	À	En	Vous pouvez également
Dans IAM Identity Center (Recommandé)	Utiliser des informations d'identification à court terme pour accéder à AWS. C'est conforme aux bonnes pratiques en matière de sécurité. Pour plus d'informations sur les bonnes pratiques, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.	Suivant les instructions fournies dans Mise en route dans le Guide de l'utilisateur AWS IAM Identity Center .	Configurez l'accès par programmation en configurant le AWS CLI à utiliser AWS IAM Identity Center dans le guide de l'AWS Command Line Interface utilisateur.
Dans IAM (Non recommandé)	Utiliser les informations d'identification à long terme pour accéder à AWS.	Suivant les instructions fournies dans Création d'un utilisateur IAM pour l'accès d'urgence dans le Guide de l'utilisateur IAM.	Configurer l'accès par programmation en suivant les instructions fournies dans Gestion des clés d'accès pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Téléchargez le AWS SDK approprié

Pour essayer l'exercice de démarrage, vous devez choisir le langage de programmation que vous souhaitez utiliser, puis télécharger le AWS SDK adapté à votre plate-forme de développement.

L'exercice de démarrage fournit des exemples en Java et C#.

Téléchargement du AWS SDK pour Java

Pour tester les exemples du manuel du développeur, vous avez besoin du kit AWS SDK pour Java. Vous disposez des options de téléchargement suivantes :

- Si vous utilisez Eclipse, vous pouvez le télécharger et l'installer AWS Toolkit for Eclipse en utilisant le site de mise à jour http://aws.amazon.com/eclipse/. Pour de plus amples informations, veuillez consulter AWS Toolkit for Eclipse.
- Si vous utilisez n'importe quel autre IDE pour créer votre application, téléchargez le kit AWS SDK pour Java.

Téléchargement du AWS SDK pour .NET

Pour tester les exemples en C# du manuel du développeur, vous avez besoin du kit AWS SDK pour .NET. Vous disposez des options de téléchargement suivantes :

- Si vous utilisez Visual Studio, vous pouvez installer à la fois le AWS SDK pour .NET et le AWS Toolkit for Visual Studio. La boîte à outils fournit l' AWS explorateur pour Visual Studio et des modèles de projet que vous pouvez utiliser pour le développement. Pour le télécharger AWS SDK pour .NET, rendez-vous sur http://aws.amazon.com/sdkfornet. Par défaut, le script d'installation installe à la fois le AWS SDK et le. AWS Toolkit for Visual Studio Pour en savoir plus sur la boîte à outils, consulter le Guide de l'utilisateur AWS Toolkit for Visual Studio.
- Si vous utilisez n'importe quel autre IDE pour créer votre application, vous pouvez utiliser le même lien que celui fourni à l'étape précédente et installer uniquement le kit AWS SDK pour .NET.

Étape 2 : créer un coffre-fort dans Amazon Glacier

Un coffre-fort est un conteneur pour le stockage d'archives. La première étape consiste à créer un coffre-fort dans l'un des modèles pris en charge Régions AWS. Pour obtenir la Régions AWS liste des appareils pris en charge par Amazon Glacier, consultez la section Points de terminaison et quotas Amazon Glacier dans le manuel de référence AWS général.

Vous pouvez créer des coffres-forts par programmation ou à l'aide de la console Amazon Glacier. Cette section utilise la console pour créer un coffre.

Pour créer un coffre

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Dans le panneau de navigation de gauche, choisissez Coffres.
3. Choisissez Créer un coffre.

   La page Créer un coffre s'ouvre.
4. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région. Votre coffre se situera dans la région que vous sélectionnez.
5. Dans Nom du coffre, donnez un nom à votre coffre.

   Les conventions de dénomination des coffres sont les suivantes :

   - Le nom d'un coffre doit être unique au sein d'un Compte AWS et de Région AWS ceux dans lesquels le coffre a été créé.
   - Un nom de coffre doit comporter entre 1 et 255 caractères.
   - Un nom de coffre peut contenir uniquement les caractères suivants : a–z, A–Z, 0–9, _ (trait de soulignement), - (trait d'union) et . (point).
6. Sous Notifications d'événements, pour activer ou désactiver les notifications de coffre signalant l'achèvement d'une tâche, choisissez l'un des paramètres suivants :

   - Désactiver les notifications : les notifications sont désactivées, et aucune notification n'est envoyée à une rubrique Amazon Simple Notification Service (Amazon SNS) lorsqu'une tâche spécifiée se termine.

   - Activer les notifications : les notifications sont activées, et une notification est envoyée à la rubrique Amazon SNS fournie lorsqu'une tâche spécifiée se termine.

     Si vous avez choisi Activer les notifications, consultez Configuration des notifications Vault à l'aide de la console Amazon Glacier.
7. Si le nom Région AWS et le nom du coffre sont corrects, choisissez Create vault.

Votre nouveau coffre-fort est désormais répertorié sur la page Coffres-forts de la console Amazon Glacier.

Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier

Dans cette étape, vous allez charger un exemple d'archive sur le coffre que vous avez créé à l'étape précédente (voir Étape 2 : créer un coffre-fort dans Amazon Glacier). Selon la plateforme de développement que vous utilisez, choisissez l'un des liens figurant à la fin de cette section.

> **Important**
>
> Toute opération liée aux archives, telle que le chargement, le téléchargement ou la suppression, nécessite d'utiliser l' AWS Command Line Interface (CLI) ou d'écrire du code. Il n'y a aucune prise en charge de la console pour les opérations d'archivage. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS CLI code pour effectuer des demandes, soit en utilisant directement l'API REST, soit en utilisant le AWS SDKs.
>
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la section AWS CLI Référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

Une archive est un objet, comme une photo, une vidéo ou un document, que vous stockez dans un coffre. Une archive est l'unité de stockage de base dans Amazon Glacier. Vous pouvez charger une archive dans une seule demande. Pour les archives volumineuses, Amazon Glacier fournit une opération d'API de téléchargement en plusieurs parties qui vous permet de télécharger une archive en plusieurs parties.

Dans cette section de mise en route, vous chargez un exemple d'archive dans une seule demande. Pour cet exercice, vous spécifiez un fichier dont la taille est plus petite. Pour les fichiers plus volumineux, le chargement partitionné est plus approprié. Pour de plus amples informations, veuillez consulter Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Rubriques

- Téléchargez une archive dans un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour Java
- Téléchargez une archive dans un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour .NET

Téléchargez une archive dans un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour Java

L'exemple de code Java suivant utilise l'API de haut niveau du AWS SDK pour Java pour télécharger un exemple d'archive dans le coffre-fort. Dans l'exemple de code, notez les éléments suivants :

- L' exemple crée une instance de la classe `AmazonGlacierClient`.
- L'exemple utilise l'opération d'API `upload` de la classe `ArchiveTransferManager` issue de l'API de haut niveau d' AWS SDK pour Java.
- L'exemple utilise la région USA Ouest (Oregon) (`us-west-2`).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du fichier d'archive que vous souhaitez charger.

> **Note**
>
> Amazon Glacier conserve un inventaire de toutes les archives de vos coffres-forts. Lorsque vous chargez l'archive dans l'exemple suivant, elle ne s'affiche pas dans un coffre dans la console de gestion tant que l'inventaire du coffre n'a pas été mis à jour. Cette mise à jour a lieu habituellement une fois par jour.

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.UploadArchiveRequest;
import software.amazon.awssdk.services.glacier.model.UploadArchiveResponse;
import software.amazon.awssdk.services.glacier.model.GlacierException;
import java.io.File;
import java.nio.file.Path;
```

```
import java.nio.file.Paths;
import java.io.FileInputStream;
import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-started.html
 */
public class UploadArchive {

    static final int ONE_MB = 1024 * 1024;

    public static void main(String[] args) {
        final String usage = """

                Usage:   <strPath> <vaultName>\s

                Where:
                   strPath - The path to the archive to upload (for example, C:\\AWS
\\test.pdf).
                   vaultName - The name of the vault.
                """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String strPath = args[0];
        String vaultName = args[1];
        File myFile = new File(strPath);
        Path path = Paths.get(strPath);
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        String archiveId = uploadContent(glacier, path, vaultName, myFile);
        System.out.println("The ID of the archived item is " + archiveId);
```

```
        glacier.close();
    }

    public static String uploadContent(GlacierClient glacier, Path path, String
vaultName, File myFile) {
        // Get an SHA-256 tree hash value.
        String checkVal = computeSHA256(myFile);
        try {
            UploadArchiveRequest uploadRequest = UploadArchiveRequest.builder()
                    .vaultName(vaultName)
                    .checksum(checkVal)
                    .build();

            UploadArchiveResponse res = glacier.uploadArchive(uploadRequest, path);
            return res.archiveId();

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }

    private static String computeSHA256(File inputFile) {
        try {
            byte[] treeHash = computeSHA256TreeHash(inputFile);
            System.out.printf("SHA-256 tree hash = %s\n", toHex(treeHash));
            return toHex(treeHash);

        } catch (IOException ioe) {
            System.err.format("Exception when reading from file %s: %s", inputFile,
ioe.getMessage());
            System.exit(-1);

        } catch (NoSuchAlgorithmException nsae) {
            System.err.format("Cannot locate MessageDigest algorithm for SHA-256:
%s", nsae.getMessage());
            System.exit(-1);
        }
        return "";
    }

    public static byte[] computeSHA256TreeHash(File inputFile) throws IOException,
            NoSuchAlgorithmException {
```

```
        byte[][] chunkSHA256Hashes = getChunkSHA256Hashes(inputFile);
        return computeSHA256TreeHash(chunkSHA256Hashes);
    }

    /**
     * Computes an SHA256 checksum for each 1 MB chunk of the input file. This
     * includes the checksum for the last chunk, even if it's smaller than 1 MB.
     */
    public static byte[][] getChunkSHA256Hashes(File file) throws IOException,
            NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");
        long numChunks = file.length() / ONE_MB;
        if (file.length() % ONE_MB > 0) {
            numChunks++;
        }

        if (numChunks == 0) {
            return new byte[][] { md.digest() };
        }

        byte[][] chunkSHA256Hashes = new byte[(int) numChunks][];
        FileInputStream fileStream = null;

        try {
            fileStream = new FileInputStream(file);
            byte[] buff = new byte[ONE_MB];

            int bytesRead;
            int idx = 0;

            while ((bytesRead = fileStream.read(buff, 0, ONE_MB)) > 0) {
                md.reset();
                md.update(buff, 0, bytesRead);
                chunkSHA256Hashes[idx++] = md.digest();
            }

            return chunkSHA256Hashes;

        } finally {
            if (fileStream != null) {
                try {
                    fileStream.close();
```

```
                } catch (IOException ioe) {
                    System.err.printf("Exception while closing %s.\n %s",
file.getName(),
                            ioe.getMessage());
                }
            }
        }
    }

    /**
     * Computes the SHA-256 tree hash for the passed array of 1 MB chunk
     * checksums.
     */
    public static byte[] computeSHA256TreeHash(byte[][] chunkSHA256Hashes)
            throws NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");
        byte[][] prevLvlHashes = chunkSHA256Hashes;
        while (prevLvlHashes.length > 1) {
            int len = prevLvlHashes.length / 2;
            if (prevLvlHashes.length % 2 != 0) {
                len++;
            }

            byte[][] currLvlHashes = new byte[len][];
            int j = 0;
            for (int i = 0; i < prevLvlHashes.length; i = i + 2, j++) {

                // If there are at least two elements remaining.
                if (prevLvlHashes.length - i > 1) {

                    // Calculate a digest of the concatenated nodes.
                    md.reset();
                    md.update(prevLvlHashes[i]);
                    md.update(prevLvlHashes[i + 1]);
                    currLvlHashes[j] = md.digest();

                } else { // Take care of the remaining odd chunk
                    currLvlHashes[j] = prevLvlHashes[i];
                }
            }

            prevLvlHashes = currLvlHashes;
        }
```

```
        return prevLvlHashes[0];
    }

    /**
     * Returns the hexadecimal representation of the input byte array
     */
    public static String toHex(byte[] data) {
        StringBuilder sb = new StringBuilder(data.length * 2);
        for (byte datum : data) {
            String hex = Integer.toHexString(datum & 0xFF);

            if (hex.length() == 1) {
                // Append leading zero.
                sb.append("0");
            }
            sb.append(hex);
        }
        return sb.toString().toLowerCase();
    }
}
```

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS SDK for Java 2.x API.

Téléchargez une archive dans un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour .NET

L'exemple de code C# suivant utilise l'API de haut niveau du AWS SDK pour .NET pour télécharger un exemple d'archive dans le coffre-fort. Dans l'exemple de code, notez les éléments suivants :

- L'exemple crée une instance de la `ArchiveTransferManager` classe pour le point de terminaison Amazon Glacier Region spécifié.
- Cet exemple utilise la région USA Ouest (Oregon) (`us-west-2`).
- L'exemple utilise l'opération d'API `Upload` de la classe `ArchiveTransferManager` pour charger votre archive. Pour les petites archives, cette opération télécharge l'archive directement sur Amazon Glacier. Pour les archives plus volumineuses, cette opération utilise l'opération d'API de téléchargement en plusieurs parties dans Amazon Glacier pour diviser le téléchargement en

plusieurs parties afin de mieux corriger les erreurs, si des erreurs se produisent lors de la diffusion des données vers Amazon Glacier.

Pour step-by-step obtenir des instructions sur la façon d'exécuter l'exemple suivant, consultezExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec le nom de votre coffre et le nom du fichier d'archive à charger.

> **Note**
>
> Amazon Glacier conserve un inventaire de toutes les archives de vos coffres-forts. Lorsque vous chargez l'archive dans l'exemple suivant, elle n'apparaît pas dans un coffre dans la console de gestion tant que l'inventaire du coffre n'a pas été mis à jour. Cette mise à jour a lieu habituellement une fois par jour.

Example— Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
    class ArchiveUploadHighLevel_GettingStarted
    {
        static string vaultName = "examplevault";
        static string archiveToUpload = "*** Provide file name (with full path) to
 upload ***";

        public static void Main(string[] args)
        {
            try
            {
                var manager = new
 ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
                // Upload an archive.
                string archiveId = manager.Upload(vaultName, "getting started archive
 test", archiveToUpload).ArchiveId;
                Console.WriteLine("Copy and save the following Archive ID for the next
 step.");
```

```
                Console.WriteLine("Archive ID: {0}", archiveId);
                Console.WriteLine("To continue, press Enter");
                Console.ReadKey();
            }
            catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
            catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
            catch (Exception e) { Console.WriteLine(e.Message); }
            Console.WriteLine("To continue, press Enter");
            Console.ReadKey();
        }
    }
}
```

Étape 4 : télécharger une archive depuis un coffre-fort dans Amazon Glacier

Dans cette étape, vous allez télécharger l'exemple d'archive que vous avez chargée précédemment à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

> **Important**
>
> Amazon Glacier fournit une console. Cependant, toute opération d'archivage, telle que le chargement, le téléchargement ou la suppression, nécessite que vous utilisiez la AWS Command Line Interface (CLI) ou que vous écriviez du code. Il n'y a aucune prise en charge de la console pour les opérations d'archivage. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS CLI code pour effectuer des demandes, soit en utilisant directement l'API REST, soit en utilisant le AWS SDKs.
>
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la section AWS CLI Référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

En général, la récupération de vos données depuis Amazon Glacier s'effectue en deux étapes :

1. Lancez une tâche d'extraction.

2. Une fois la tâche terminée, téléchargez les octets de données.

Pour récupérer une archive depuis Amazon Glacier, vous devez d'abord lancer une tâche. Une fois la tâche terminée, vous téléchargez les données. Pour plus d'informations sur les récupérations d'archives, consultez Récupération des archives Amazon Glacier.

Le temps d'accès de votre demande dépend de l'option d'extraction que vous choisissez : Expedited, Standard ou Bulk. Pour toutes les archives à l'exception des plus volumineuses (plus de 250 Mo), les archives auxquelles vous accédez avec des extractions de type Expedited sont généralement disponibles en 1-5 minutes. Les archives récupérées avec des extractions de type Standard sont généralement disponibles en 3-5 heures. Les extractions de type Bulk sont généralement disponibles dans un délai compris entre 5 et 12 heures. Pour plus d'informations sur les différentes options de récupération, consultez la FAQ Amazon Glacier. Pour plus d'informations sur les frais de récupération des données, consultez la page de tarification d'Amazon Glacier.

Les exemples de code présentés dans les rubriques suivantes lancent la tâche, attendent qu'elle soit terminée, puis téléchargent les données de l'archive.

Rubriques

- Téléchargez une archive depuis un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour Java
- Téléchargez une archive depuis un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour .NET

Téléchargez une archive depuis un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour Java

L'exemple de code Java suivant utilise l'API de haut niveau du AWS SDK pour Java pour télécharger l'archive que vous avez chargée à l'étape précédente. Dans l'exemple de code, notez les éléments suivants :

- L' exemple crée une instance de la classe `AmazonGlacierClient`.
- Le code utilise la région USA Ouest (Oregon) (`us-west-2`) qui correspond à l'emplacement du coffre que vous avez créé précédemment à l'Étape 2 : créer un coffre-fort dans Amazon Glacier.
- L'exemple utilise l'opération d'API `download` de la classe `ArchiveTransferManager` issue de l'API de haut niveau d' AWS SDK pour Java. L'exemple crée une rubrique Amazon Simple Notification Service (Amazon SNS) et une file d'attente Amazon Simple Queue Service (Amazon SQS) qui est abonnée à cette rubrique. Si vous avez créé un utilisateur administrateur

Gestion des identités et des accès AWS (IAM) conformément aux instructionsÉtape 1 : Avant de commencer à utiliser Amazon Glacier, celui-ci dispose des autorisations IAM nécessaires pour créer et utiliser la rubrique Amazon SNS et la file d'attente Amazon SQS.

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID d'archive du fichier que vous avez chargé à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

Example— Téléchargement d'une archive à l'aide du AWS SDK pour Java

```
import java.io.File;
import java.io.IOException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.transfer.ArchiveTransferManager;
import com.amazonaws.services.sns.AmazonSNSClient;
import com.amazonaws.services.sqs.AmazonSQSClient;

public class AmazonGlacierDownloadArchive_GettingStarted {
    public static String vaultName = "examplevault";
    public static String archiveId = "*** provide archive ID ***";
    public static String downloadFilePath  = "*** provide location to download archive
 ***";

    public static AmazonGlacierClient glacierClient;
    public static AmazonSQSClient sqsClient;
    public static AmazonSNSClient snsClient;

    public static void main(String[] args) throws IOException {


     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        glacierClient = new AmazonGlacierClient(credentials);
        sqsClient = new AmazonSQSClient(credentials);
        snsClient = new AmazonSNSClient(credentials);

        glacierClient.setEndpoint("glacier.us-west-2.amazonaws.com");
        sqsClient.setEndpoint("sqs.us-west-2.amazonaws.com");
        snsClient.setEndpoint("sns.us-west-2.amazonaws.com");
```

```
        try {
            ArchiveTransferManager atm = new ArchiveTransferManager(glacierClient,
 sqsClient, snsClient);

            atm.download(vaultName, archiveId, new File(downloadFilePath));

        } catch (Exception e)
        {
            System.err.println(e);
        }
    }
}
```

Téléchargez une archive depuis un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour .NET

L'exemple de code C# suivant utilise l'API de haut niveau du AWS SDK pour .NET pour télécharger l'archive que vous avez téléchargée précédemment. Téléchargez une archive dans un coffre-fort d'Amazon Glacier à l'aide du AWS SDK pour .NET Dans l'exemple de code, notez les éléments suivants :

- L'exemple crée une instance de la `ArchiveTransferManager` classe pour le point de terminaison Amazon Glacier Region spécifié.
- L'exemple de code utilise la région USA Ouest (Oregon) (`us-west-2`) qui correspond à l'emplacement du coffre créé précédemment à l'Étape 2 : créer un coffre-fort dans Amazon Glacier.
- L'exemple utilise l'opération d'API `Download` de la classe `ArchiveTransferManager` pour télécharger votre archive. L'exemple crée une rubrique Amazon Simple Notification Service (Amazon SNS) et une file d'attente Amazon Simple Queue Service (Amazon SQS) qui est abonnée à cette rubrique. Si vous avez créé un utilisateur administrateur Gestion des identités et des accès AWS (IAM) conformément aux instructionsÉtape 1 : Avant de commencer à utiliser Amazon Glacier, celui-ci dispose des autorisations IAM nécessaires pour créer et utiliser la rubrique Amazon SNS et la file d'attente Amazon SQS.
- L'exemple lance ensuite la tâche d'extraction d'archive et interroge la file d'attente pour que l'archive soit disponible. Le téléchargement commence dès que l'archive est disponible. Pour plus d'informations sur les délais d'extraction, consultez Options de récupération des archives.

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID d'archive du fichier que vous avez chargé à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

Example— Téléchargez une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
    class ArchiveDownloadHighLevel_GettingStarted
    {
        static string vaultName = "examplevault";
        static string archiveId = "*** Provide archive ID ***";
        static string downloadFilePath = "*** Provide the file name and path to where
 to store the download ***";

        public static void Main(string[] args)
        {
            try
            {
                var manager = new
 ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);

                var options = new DownloadOptions();
                options.StreamTransferProgress +=
 ArchiveDownloadHighLevel_GettingStarted.progress;
                // Download an archive.
                Console.WriteLine("Intiating the archive retrieval job and then polling
 SQS queue for the archive to be available.");
                Console.WriteLine("Once the archive is available, downloading will
 begin.");
                manager.Download(vaultName, archiveId, downloadFilePath, options);
                Console.WriteLine("To continue, press Enter");
                Console.ReadKey();
            }
            catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
            catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
            catch (Exception e) { Console.WriteLine(e.Message); }
```

```
            Console.WriteLine("To continue, press Enter");
            Console.ReadKey();
        }

        static int currentPercentage = -1;
        static void progress(object sender, StreamTransferProgressArgs args)
        {
            if (args.PercentDone != currentPercentage)
            {
                currentPercentage = args.PercentDone;
                Console.WriteLine("Downloaded {0}%", args.PercentDone);
            }
        }
    }
}
```

Étape 5 : Supprimer une archive d'un coffre-fort dans Amazon Glacier

Dans cette étape, vous allez supprimer l'exemple d'archive que vous avez chargé à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

> **Important**
>
> Vous ne pouvez pas supprimer une archive à l'aide de la console Amazon Glacier. Toute opération d'archivage, telle que le chargement, le téléchargement ou la suppression, nécessite que vous utilisiez la AWS Command Line Interface (CLI) ou que vous écriviez du code. Pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez utiliser le code AWS CLI ou écrire du code pour effectuer des demandes, en utilisant soit directement l'API REST, soit en utilisant le AWS SDKs.
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la section AWS CLI Référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

Supprimez l'archive d'exemple en suivant l'une des méthodes suivantes SDKs ou en suivant AWS CLI :

- Supprimer une archive d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Supprimer une archive d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Supprimer une archive dans Amazon Glacier à l'aide du AWS CLI

Sections connexes

- Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier
- Supprimer une archive dans Amazon Glacier

Supprimer une archive d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

L'exemple de code suivant utilise le AWS SDK pour Java pour supprimer l'archive. Dans le code, notez les éléments suivants :

- L'objet `DeleteArchiveRequest` décrit la demande de suppression, y compris le nom du coffre où se trouve l'archive et l'ID de l'archive.
- L'opération `deleteArchive` d'API envoie la demande à Amazon Glacier pour supprimer l'archive.
- L'exemple utilise la région USA Ouest (Oregon) (`us-west-2`).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID d'archive du fichier que vous avez chargé à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

Example— Suppression d'une archive à l'aide du AWS SDK pour Java

```
import java.io.IOException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.DeleteArchiveRequest;

public class AmazonGlacierDeleteArchive_GettingStarted {
```

```
    public static String vaultName = "examplevault";
    public static String archiveId = "*** provide archive ID***";
    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {


     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-west-2.amazonaws.com/");

        try {

            // Delete the archive.
            client.deleteArchive(new DeleteArchiveRequest()
                .withVaultName(vaultName)
                .withArchiveId(archiveId));

            System.out.println("Deleted archive successfully.");

        } catch (Exception e) {
            System.err.println("Archive not deleted.");
            System.err.println(e);
        }
    }
}
```

Supprimer une archive d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

L'exemple de code C# suivant utilise l'API de haut niveau du AWS SDK pour .NET pour supprimer l'archive que vous avez téléchargée à l'étape précédente. Dans l'exemple de code, notez les éléments suivants :

- L'exemple crée une instance de la `ArchiveTransferManager` classe pour le point de terminaison Amazon Glacier Region spécifié.
- Cet exemple utilise la région USA Ouest (Oregon) (`us-west-2`).
- Cet exemple utilise l'opération d'API `Delete` de la classe `ArchiveTransferManager` fournie par l'API de haut niveau d' AWS SDK pour .NET.

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID d'archive du fichier que vous avez chargé à l'Étape 3 : télécharger une archive dans un coffre-fort d'Amazon Glacier.

Example— Suppression d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDeleteHighLevel_GettingStarted
  {
    static string vaultName = "examplevault";
    static string archiveId = "*** Provide archive ID ***";

    public static void Main(string[] args)
    {
      try
      {
        var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
        manager.DeleteArchive(vaultName, archiveId);
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }
  }
}
```

Supprimer une archive dans Amazon Glacier à l'aide du AWS CLI

Vous pouvez supprimer des archives dans Amazon Glacier en utilisant le AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI

- Exemple : suppression d'une archive à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

Exemple : suppression d'une archive à l'aide du AWS CLI

1. Utilisez la commande `initiate-job` pour démarrer une tâche d'extraction d'inventaire. Pour plus d'informations sur la commande `initiate-job`, consultez Lancement d'une tâche.

   ```
   aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
   job-parameters "{\"Type\": \"inventory-retrieval\"}"
   ```

   Sortie attendue :

```
{
    "location": "/111122223333/vaults/awsexamplevault/jobs/*** jobid ***",
    "jobId": "*** jobid ***"
}
```

2. Utilisez la commande `describe-job` pour vérifier le statut de la tâche d'extraction précédente. Pour plus d'informations sur la commande `describe-job`, consultez Description d'une tâche.

```
aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
job-id *** jobid ***
```

Sortie attendue :

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "*** vault arn ***",
    "Completed": false,
    "JobId": "*** jobid ***",
    "Action": "InventoryRetrieval",
    "CreationDate": "*** job creation date ***",
    "StatusCode": "InProgress"
}
```

3. Attendez que la tâche soit terminée.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification sur le coffre-fort ou si vous avez spécifié une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

   Vous pouvez définir la configuration des notifications pour des événements spécifiques concernant le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier. Amazon Glacier envoie un message à la rubrique Amazon SNS spécifiée chaque fois que l'événement spécifique se produit.

4. Une fois la tâche terminée, utilisez la commande `get-job-output` pour télécharger la tâche d'extraction dans le fichier `output.json`. Pour plus d'informations sur la commande `get-job-output`, consultez Obtention de la sortie d'une tâche.

```
aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
 --job-id *** jobid *** output.json
```

Cette commande produit un fichier avec les champs suivants.

```
{
"VaultARN":"arn:aws:glacier:region:111122223333:vaults/awsexamplevault",
"InventoryDate":"*** job completion date ***",
"ArchiveList":[{
{"ArchiveId":"*** archiveid ***",
"ArchiveDescription":"*** archive description (if set) ***",
"CreationDate":"*** archive creation date ***",
"Size":"*** archive size (in bytes) ***",
"SHA256TreeHash":"*** archive hash ***"
}],
"ArchiveId": 123456789

}
```

5. Utilisez la commande `delete-archive` pour supprimer chaque archive d'un coffre jusqu'à ce qu'il n'en reste aucune.

```
aws glacier delete-archive --vault-name awsexamplevault --account-id 111122223333
 --archive-id="*** archiveid ***"
```

Pour plus d'informations sur la commande `delete-archive`, consultez Suppression d'une archive.

Étape 6 : supprimer un coffre-fort dans Amazon Glacier

Un coffre-fort est un conteneur pour le stockage d'archives. Pour supprimer un coffre-fort Amazon Glacier, vous devez d'abord supprimer toutes les archives existantes dans le coffre-fort depuis le dernier inventaire calculé par Amazon Glacier.

Vous pouvez supprimer un coffre-fort par programmation ou à l'aide de la console Amazon Glacier. Pour plus d'informations sur la suppression d'un coffre par programme, consultez la page Supprimer un coffre-fort dans Amazon Glacier.

> **Important**
>
> Si vous avez chargé une archive sur un coffre ou supprimé une archive d'un coffre au cours des dernières 24 heures, vous devez attendre que le dernier inventaire du coffre soit mis à jour avant qu'il affiche les informations les plus récentes. Amazon Glacier prépare régulièrement un inventaire pour chaque coffre, toutes les 24 heures.

Pour supprimer un coffre vide

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Dans le menu Sélectionnez une région, choisissez Région AWS le coffre que vous souhaitez supprimer.

   Dans cet exercice de démarrage, l'exemple de coffre se trouve dans la région USA Ouest (Oregon).
3. Sélectionnez la case d'option située en regard du coffre vide que vous souhaitez supprimer. Si le coffre n'est pas vide, vous devez supprimer toutes les archives avant de supprimer le coffre. Pour de plus amples informations, veuillez consulter Supprimer une archive dans Amazon Glacier.

   > **Important**
   >
   > La suppression d'un coffre ne peut pas être annulée.
4. Sélectionnez Delete (Supprimer).
5. La boîte de dialogue Supprimer le coffre s'affiche. Sélectionnez Delete (Supprimer).

Pour supprimer un coffre non vide

1. Si vous supprimez un coffre non vide, vous devez d'abord supprimer toutes les archives qu'il contient. Vous pouvez le faire en écrivant du code pour effectuer une demande de suppression d'archive à l'aide de l'API REST, du AWS SDK pour Java, du AWS SDK pour .NET ou du AWS CLI. Pour plus d'informations sur la suppression d'archives, consultez la section Étape 5 : Supprimer une archive d'un coffre-fort dans Amazon Glacier.

2. Après avoir vidé le coffre, suivez les étapes permettant de supprimer un coffre vide dans la procédure précédente.

Comment procéder ensuite ?

Maintenant que vous avez essayé l'exercice de démarrage, vous pouvez explorer les sections suivantes pour en savoir plus sur Amazon Glacier.

- Utilisation des coffres-forts dans Amazon Glacier
- Utilisation des archives dans Amazon Glacier

Utilisation des coffres-forts dans Amazon Glacier

Un coffre-fort est un conteneur pour le stockage d'archives. Lorsque vous créez un coffre, vous spécifiez un nom de coffre et le coffre Région AWS dans lequel vous souhaitez le créer. Pour obtenir la liste des solutions Régions AWS prises en charge par Amazon Glacier, consultez la section Points de terminaison et quotas Amazon Glacier dans le manuel de référence AWS général.

Vous pouvez stocker un nombre illimité d'archives dans un coffre.

> **Important**
>
> Amazon Glacier fournit une console. Cependant, toute opération d'archivage, telle que le chargement, le téléchargement ou la suppression, nécessite que vous utilisiez le AWS Command Line Interface (AWS CLI) ou que vous écriviez du code. Il n'y a aucune prise en charge de la console pour les opérations d'archivage. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS CLI code pour effectuer des demandes, soit en utilisant directement l'API REST, soit en utilisant le AWS SDKs.
>
> Pour installer le AWS CLI, voir AWS Command Line Interface. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la AWS CLI référence relative à Amazon Glacier. Pour des exemples d'utilisation du AWS CLI pour télécharger des archives sur Amazon Glacier, consultez la section Utilisation d'Amazon Glacier avec le AWS Command Line Interface.

Rubriques

- Opérations du coffre-fort dans Amazon Glacier
- Création d'un coffre-fort dans Amazon Glacier
- Récupération des métadonnées du coffre-fort dans Amazon Glacier
- Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier
- Configuration des notifications de coffre-fort dans Amazon Glacier
- Supprimer un coffre-fort dans Amazon Glacier
- Marquer vos coffres-forts Amazon Glacier

- Verrou Amazon Glacier Vault

Opérations du coffre-fort dans Amazon Glacier

Amazon Glacier prend en charge diverses opérations de coffre-fort. Les opérations de coffre ne s'appliquent pas à toutes les Régions AWS. En d'autres termes, lorsque vous créez un coffre, vous le créez dans une Région AWS spécifique. Lorsque vous listez des coffres-forts, Amazon Glacier renvoie la liste des coffres-forts à partir de Région AWS celle que vous avez spécifiée dans la demande.

Création et suppression des coffres

Et Compte AWS peut créer jusqu'à 1 000 coffres-forts par. Région AWS Pour obtenir la liste des solutions Régions AWS prises en charge par Amazon Glacier, consultez la section Points de terminaison et quotas Amazon Glacier dans le manuel de référence AWS général.

Vous ne pouvez supprimer un coffre-fort que s'il ne contient aucune archive lors du dernier inventaire calculé par Amazon Glacier et s'il n'y a eu aucune écriture dans le coffre-fort depuis le dernier inventaire.

> **Note**
>
> Amazon Glacier prépare régulièrement un inventaire pour chaque coffre, toutes les 24 heures. Dans la mesure où l'inventaire peut ne pas refléter les informations les plus récentes, Amazon Glacier s'assure que le coffre est bien vide en vérifiant si des opérations d'écriture ont été effectuées depuis le dernier inventaire du coffre.

Pour plus d'informations, consultez Création d'un coffre-fort dans Amazon Glacier et Supprimer un coffre-fort dans Amazon Glacier.

Extraction de métadonnées de coffre

Vous pouvez extraire les informations d'un coffre telles que sa date de création, le nombre d'archives qu'il contient et la taille totale de toutes ses archives. Amazon Glacier fournit des appels d'API vous permettant de récupérer ces informations pour un coffre-fort spécifique ou pour tous les coffres-forts d'un coffre-fort spécifique Région AWS de votre compte. Pour de plus amples informations, veuillez consulter Récupération des métadonnées du coffre-fort dans Amazon Glacier.

Téléchargement d'un inventaire de coffre

Un inventaire de coffre fait référence à la liste des archives contenues dans un coffre. Pour chaque archive présente dans la liste, l'inventaire fournit des informations telles que l'ID de l'archive, sa date de création et sa taille. Amazon Glacier met à jour l'inventaire du coffre-fort une fois par jour, à compter du jour où la première archive est chargée dans le coffre-fort. Un inventaire de coffre doit exister pour que vous puissiez le télécharger.

Le téléchargement d'un inventaire de coffre est une opération asynchrone. Vous devez commencer par lancer une tâche pour télécharger l'inventaire. Après avoir reçu la demande de travail, Amazon Glacier prépare votre inventaire pour le téléchargement. Une fois la tâche terminée, vous pouvez télécharger les données d'inventaire.

Compte tenu de la nature asynchrone de la tâche, vous pouvez utiliser des notifications Amazon Simple Notification Service (Amazon SNS) pour être notifié lorsque la tâche se termine. Vous pouvez spécifier une rubrique Amazon SNS pour chaque demande de tâche individuelle ou configurer votre coffre de sorte qu'il envoie une notification lorsque des événements de coffre déterminés se produisent.

Amazon Glacier prépare régulièrement un inventaire pour chaque coffre, toutes les 24 heures. Si aucun ajout ou aucune suppression d'archive n'a été effectué dans le coffre depuis le dernier inventaire, la date de l'inventaire n'est pas mise à jour.

Lorsque vous lancez une tâche pour un inventaire de coffre-fort, Amazon Glacier renvoie le dernier inventaire généré, qui est un point-in-time instantané et non des données en temps réel. Vous ne trouverez peut-être pas utile d'extraire un inventaire de coffre pour chaque chargement d'archive. Supposons toutefois que vous mainteniez une base de données côté client contenant les métadonnées associées aux archives que vous chargez sur Amazon Glacier. Dans ce cas, vous trouverez peut-être l'inventaire de coffre très utile pour rapprocher les informations contenues dans votre base de données avec l'inventaire de coffre réel.

Pour plus d'informations sur l'extraction d'un inventaire de coffre, consultez la page Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

Configuration de notifications de coffre

La récupération de tout élément d'Amazon Glacier, tel qu'une archive d'un coffre-fort ou un inventaire de coffre-fort, est un processus en deux étapes. Tout d'abord, vous lancez une tâche. Une fois la

tâche terminée, vous téléchargez la sortie. Pour savoir quand votre tâche est terminée, vous pouvez utiliser les notifications Amazon Glacier. Amazon Glacier envoie des messages de notification à une rubrique Amazon Simple Notification Service (Amazon SNS) que vous fournissez.

Vous pouvez configurer des notifications sur un coffre et identifier les événements de coffre et la rubrique Amazon SNS à notifier lorsque ces événements se produisent. Chaque fois que l'événement du coffre-fort se produit, Amazon Glacier envoie une notification à la rubrique Amazon SNS spécifiée. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier.

Création d'un coffre-fort dans Amazon Glacier

La création d'un coffre ajoute un coffre à la série de coffres de votre compte. An Compte AWS peut créer jusqu'à 1 000 coffres-forts par AWS région. Pour obtenir la liste des AWS régions prises en charge par Amazon Glacier (Amazon Glacier), consultez la section Régions et points de terminaison dans le manuel de référence AWS général.

Lorsque vous créez un coffre, vous devez indiquer un nom de coffre. Les conventions de dénomination des coffres sont les suivantes :

- Les noms peuvent comporter entre 1 et 255 caractères.
- Les caractères autorisés sont a-z, A-Z, 0-9, « _ » (trait de soulignement), « - » (trait d'union) et « . » (point).

Les noms de coffre-fort doivent être uniques au sein d'un compte et de la AWS région dans laquelle le coffre-fort est créé. En d'autres termes, un compte peut créer des coffres-forts portant le même nom dans différentes AWS régions, mais pas dans la même AWS région.

Rubriques

- Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Création d'un coffre-fort dans Amazon Glacier à l'aide de l'API REST
- Création d'un coffre-fort à l'aide de la console Amazon Glacier
- Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

L'API de bas niveau fournit des méthodes pour toutes les opérations de coffre-fort, notamment la création et la suppression de coffres-forts, l'obtention d'une description du coffre-fort et l'obtention d'une liste des coffres-forts créés dans un coffre spécifique. Région AWS Voici les étapes de création d'un coffre à l'aide de AWS SDK pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier le coffre Région AWS dans lequel vous souhaitez créer un coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette Région AWS.

2. Fournissez les informations de demande en créant une instance de la classe `CreateVaultRequest`.

   Amazon Glacier (Amazon Glacier) vous demande de fournir un nom de coffre-fort et votre numéro de compte. Si vous n'indiquez pas d'ID de compte, le système utilise l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.

3. Exécutez la méthode `createVault` en fournissant l'objet de demande comme paramètre.

   La réponse renvoyée par Amazon Glacier est disponible dans l'`CreateVaultResult`objet.

L'extrait de code Java suivant illustre les étapes précédentes. L'extrait crée un coffre dans la région `us-west-2`. `Location`Il imprime l'URI relatif du coffre-fort qui inclut votre identifiant de compte, le Région AWS, et le nom du coffre-fort.

```
AmazonGlacierClient client = new AmazonGlacierClient(credentials);
client.setEndpoint("https://glacier.us-west-2.amazonaws.com");

CreateVaultRequest request = new CreateVaultRequest()
    .withVaultName("*** provide vault name ***");
CreateVaultResult result = client.createVault(request);

System.out.println("Created vault successfully: " + result.getLocation());
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Création de coffre (PUT vault).

Exemple : création d'un coffre-fort à l'aide du AWS SDK pour Java

L'exemple de code Java suivant crée un coffre-fort dans la `us-west-2` région (pour plus d'informations Régions AWS, voirAccès à Amazon Glacier). En outre, l'exemple de code récupère les informations du coffre, répertorie tous les coffres-forts du même coffre Région AWS, puis supprime le coffre créé.

Pour step-by-step obtenir des instructions sur la façon d'exécuter l'exemple suivant, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse.

Example

```
import java.io.IOException;
import java.util.List;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.CreateVaultRequest;
import com.amazonaws.services.glacier.model.CreateVaultResult;
import com.amazonaws.services.glacier.model.DeleteVaultRequest;
import com.amazonaws.services.glacier.model.DescribeVaultOutput;
import com.amazonaws.services.glacier.model.DescribeVaultRequest;
import com.amazonaws.services.glacier.model.DescribeVaultResult;
import com.amazonaws.services.glacier.model.ListVaultsRequest;
import com.amazonaws.services.glacier.model.ListVaultsResult;


public class AmazonGlacierVaultOperations {

    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
```

```
        client.setEndpoint("https://glacier.us-east-1.amazonaws.com/");

        String vaultName = "examplevaultfordelete";

        try {
            createVault(client, vaultName);
            describeVault(client, vaultName);
            listVaults(client);
            deleteVault(client, vaultName);

        } catch (Exception e) {
            System.err.println("Vault operation failed." + e.getMessage());
        }
    }

    private static void createVault(AmazonGlacierClient client, String vaultName) {
        CreateVaultRequest createVaultRequest = new CreateVaultRequest()
            .withVaultName(vaultName);
        CreateVaultResult createVaultResult = client.createVault(createVaultRequest);

        System.out.println("Created vault successfully: " +
createVaultResult.getLocation());
    }

    private static void describeVault(AmazonGlacierClient client, String vaultName) {
        DescribeVaultRequest describeVaultRequest = new DescribeVaultRequest()
            .withVaultName(vaultName);
        DescribeVaultResult describeVaultResult  =
client.describeVault(describeVaultRequest);

        System.out.println("Describing the vault: " + vaultName);
        System.out.print(
                "CreationDate: " + describeVaultResult.getCreationDate() +
                "\nLastInventoryDate: " + describeVaultResult.getLastInventoryDate() +
                "\nNumberOfArchives: " + describeVaultResult.getNumberOfArchives() +
                "\nSizeInBytes: " + describeVaultResult.getSizeInBytes() +
                "\nVaultARN: " + describeVaultResult.getVaultARN() +
                "\nVaultName: " + describeVaultResult.getVaultName());
    }

    private static void listVaults(AmazonGlacierClient client) {
        ListVaultsRequest listVaultsRequest = new ListVaultsRequest();
        ListVaultsResult listVaultsResult = client.listVaults(listVaultsRequest);
```

```
        List<DescribeVaultOutput> vaultList = listVaultsResult.getVaultList();
        System.out.println("\nDescribing all vaults (vault list):");
        for (DescribeVaultOutput vault : vaultList) {
            System.out.println(
                    "\nCreationDate: " + vault.getCreationDate() +
                    "\nLastInventoryDate: " + vault.getLastInventoryDate() +
                    "\nNumberOfArchives: " + vault.getNumberOfArchives() +
                    "\nSizeInBytes: " + vault.getSizeInBytes() +
                    "\nVaultARN: " + vault.getVaultARN() +
                    "\nVaultName: " + vault.getVaultName());
        }
    }

    private static void deleteVault(AmazonGlacierClient client, String vaultName) {
        DeleteVaultRequest request = new DeleteVaultRequest()
            .withVaultName(vaultName);
        client.deleteVault(request);
        System.out.println("Deleted vault: " + vaultName);
    }

}
```

Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode pour créer un coffre-fort.

Rubriques

- Création d'un coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Création d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Création d'un coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET

La classe `ArchiveTransferManager` de l'API de haut niveau propose la méthode `CreateVault` qui vous permet de créer un coffre dans une région AWS .

Exemple : opérations de coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET

L'exemple de code C# suivant crée et supprime un coffre dans la région USA Ouest (Oregon). Pour obtenir la liste des coffres-forts Régions AWS dans lesquels vous pouvez créer des coffres-forts, consultezAccès à Amazon Glacier.

Pour step-by-step obtenir des instructions sur la façon d'exécuter l'exemple suivant, consultezExemples de code en cours d'exécution. Vous devez mettre à jour le code avec le nom du coffre, comme indiqué.

Example

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class VaultCreateDescribeListVaultsDeleteHighLevel
  {
    static string vaultName = "*** Provide vault name ***";

    public static void Main(string[] args)
    {
      try
      {
          var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
          manager.CreateVault(vaultName);
          Console.WriteLine("Vault created. To delete the vault, press Enter");
          Console.ReadKey();
          manager.DeleteVault(vaultName);
          Console.WriteLine("\nVault deleted. To continue, press Enter");
          Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }
  }
}
```

Création d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'API de bas niveau fournit des méthodes pour toutes les opérations de coffre-fort, notamment la création et la suppression de coffres-forts, l'obtention d'une description du coffre-fort et l'obtention d'une liste des coffres-forts créés dans un coffre spécifique. Région AWS Voici les étapes de création d'un coffre à l'aide de AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier le coffre Région AWS dans lequel vous souhaitez créer un coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette Région AWS.
2. Fournissez les informations de demande en créant une instance de la classe `CreateVaultRequest`.

   Amazon Glacier (Amazon Glacier) vous demande de fournir un nom de coffre-fort et votre numéro de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.
3. Exécutez la méthode `CreateVault` en fournissant l'objet de demande comme paramètre.

   La réponse renvoyée par Amazon Glacier est disponible dans l'`CreateVaultResponse`objet.

Exemple : opérations de coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'exemple de code C# suivant illustre les étapes précédentes. Cet exemple crée un coffre dans la région USA Ouest (Oregon). En outre, l'exemple de code récupère les informations du coffre, répertorie tous les coffres-forts du même coffre Région AWS, puis supprime le coffre créé. L'`Location`imprimé est l'URI relatif du coffre-fort qui inclut votre identifiant de compte, le Région AWS, et le nom du coffre-fort.

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Création de coffre (PUT vault).

Pour step-by-step obtenir des instructions sur la façon d'exécuter l'exemple suivant, consultezExemples de code en cours d'exécution. Vous devez mettre à jour le code avec le nom du coffre, comme indiqué.

Example

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class VaultCreateDescribeListVaultsDelete
  {
    static string vaultName = "*** Provide vault name ***";
    static AmazonGlacierClient client;

    public static void Main(string[] args)
    {
       try
      {
         using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Creating a vault.");
          CreateAVault();
          DescribeVault();
          GetVaultsList();
          Console.WriteLine("\nVault created. Now press Enter to delete the vault...");
          Console.ReadKey();
          DeleteVault();
        }
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }

    static void CreateAVault()
    {
      CreateVaultRequest request = new CreateVaultRequest()
```

```
      {
        VaultName = vaultName
      };
      CreateVaultResponse response = client.CreateVault(request);
      Console.WriteLine("Vault created: {0}\n", response.Location);
    }

    static void DescribeVault()
    {
      DescribeVaultRequest describeVaultRequest = new DescribeVaultRequest()
      {
        VaultName = vaultName
      };

      DescribeVaultResponse describeVaultResponse =
client.DescribeVault(describeVaultRequest);
      Console.WriteLine("\nVault description...");
      Console.WriteLine(
        "\nVaultName: " + describeVaultResponse.VaultName +
        "\nVaultARN: " + describeVaultResponse.VaultARN +
        "\nVaultCreationDate: " + describeVaultResponse.CreationDate +
        "\nNumberOfArchives: " + describeVaultResponse.NumberOfArchives +
        "\nSizeInBytes: " + describeVaultResponse.SizeInBytes +
        "\nLastInventoryDate: " + describeVaultResponse.LastInventoryDate
        );
    }

    static void GetVaultsList()
    {
      string lastMarker = null;
      Console.WriteLine("\n List of vaults in your account in the specific
region ...");
      do
      {
        ListVaultsRequest request = new ListVaultsRequest()
        {
          Marker = lastMarker
        };
        ListVaultsResponse response = client.ListVaults(request);

        foreach (DescribeVaultOutput output in response.VaultList)
        {
          Console.WriteLine("Vault Name: {0} \tCreation Date: {1} \t #of archives:
{2}",
```

```
                            output.VaultName, output.CreationDate,
 output.NumberOfArchives);
        }
        lastMarker = response.Marker;
      } while (lastMarker != null);
    }

    static void DeleteVault()
    {
      DeleteVaultRequest request = new DeleteVaultRequest()
      {
        VaultName = vaultName
      };
      DeleteVaultResponse response = client.DeleteVault(request);
    }
  }
}
```

Création d'un coffre-fort dans Amazon Glacier à l'aide de l'API REST

Pour créer un coffre à l'aide de l'API REST, consultez la page Création de coffre (PUT vault).

Création d'un coffre-fort à l'aide de la console Amazon Glacier

Pour créer un coffre-fort à l'aide de la console Amazon Glacier (Amazon Glacier), consultez le didacticiel de mise Étape 2 : créer un coffre-fort dans Amazon Glacier en route.

Création d'un coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Procédez comme suit pour créer un coffre-fort dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : création d'un coffre-fort à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

Exemple : création d'un coffre-fort à l'aide du AWS CLI

1. Utilisez la `create-vault` commande pour créer un coffre-fort nommé *awsexamplevault* sous compte*111122223333*.

   ```
   aws glacier create-vault --vault-name awsexamplevault --account-id 111122223333
   ```

   Sortie attendue :

   ```
   {
       "location": "/111122223333/vaults/awsexamplevault"
   }
   ```

2. Vérifiez la création à l'aide de la commande `describe-vault`.

```
aws glacier describe-vault --vault-name awsexamplevault --account-id 111122223333
```

Récupération des métadonnées du coffre-fort dans Amazon Glacier

Vous pouvez extraire les informations d'un coffre telles que sa date de création, le nombre d'archives qu'il contient et la taille totale de toutes ses archives. Amazon Glacier (Amazon Glacier) fournit des appels d'API vous permettant de récupérer ces informations pour un coffre-fort spécifique ou pour tous les coffres-forts d'une AWS région spécifique dans votre compte.

Si vous récupérez une liste de coffres-forts, Amazon Glacier renvoie la liste triée en fonction des valeurs ASCII des noms de coffres-forts. La liste peut contenir jusqu'à 1 000 coffres. Vous devez toujours vérifier dans la réponse le marqueur à partir duquel poursuivre l'affichage de la liste ; s'il n'y a plus d'éléments, la valeur du champ marker est `null`. Vous pouvez éventuellement limiter le nombre de coffres renvoyés dans la réponse. S'il existe d'autres coffres que ceux renvoyés dans la réponse, le résultat est paginé. Vous devez envoyer des demandes supplémentaires afin d'extraire l'ensemble de coffres suivant.

Rubriques

- Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Extraction de métadonnées de coffre à l'aide de l'API REST
- Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

Rubriques

- Extraire des métadonnées de coffre pour un coffre

- Extraire les métadonnées de coffre pour tous les coffres dans une région
- Exemple : Extraction des métadonnées d'un coffre à l'aide du kit SDK Amazon pour Java

Extraire des métadonnées de coffre pour un coffre

Vous pouvez récupérer les métadonnées d'un coffre-fort spécifique ou de tous les coffres-forts d'une AWS région spécifique. La procédure ci-dessous détaille les étapes à suivre pour extraire les métadonnées d'un coffre spécifique à l'aide de l'API de bas niveau du kit SDK Amazon pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Fournissez les informations de demande en créant une instance de la classe `DescribeVaultRequest`.

   Amazon Glacier (Amazon Glacier) vous demande de fournir un nom de coffre-fort et votre numéro de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.
3. Exécutez la méthode `describeVault` en fournissant l'objet de demande comme paramètre.

   Les informations de métadonnées du coffre renvoyées par Amazon Glacier sont disponibles dans l'`DescribeVaultResult`objet.

L'extrait de code Java suivant illustre les étapes précédentes.

```
DescribeVaultRequest request = new DescribeVaultRequest()
 .withVaultName("*** provide vault name***");

DescribeVaultResult result = client.describeVault(request);

System.out.print(
        "\nCreationDate: " + result.getCreationDate() +
        "\nLastInventoryDate: " + result.getLastInventoryDate() +
        "\nNumberOfArchives: " + result.getNumberOfArchives() +
        "\nSizeInBytes: " + result.getSizeInBytes() +
        "\nVaultARN: " + result.getVaultARN() +
```

```
        "\nVaultName: " + result.getVaultName());
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Description de coffre (GET vault).

Extraire les métadonnées de coffre pour tous les coffres dans une région

Vous pouvez également utiliser `listVaults` cette méthode pour récupérer les métadonnées de tous les coffres-forts d'une AWS région spécifique.

L'extrait de code Java suivant extrait la liste des coffres dans la région `us-west-2`. La demande limite à 5 le nombre de coffres renvoyés dans la réponse. L'extrait de code effectue ensuite une série d'`listVaults`appels pour récupérer la liste complète des coffres-forts dans la AWS région.

```
AmazonGlacierClient client;
client.setEndpoint("https://glacier.us-west-2.amazonaws.com/");

String marker = null;
do {
    ListVaultsRequest request = new ListVaultsRequest()
        .withLimit("5")
        .withMarker(marker);
    ListVaultsResult listVaultsResult = client.listVaults(request);

    List<DescribeVaultOutput> vaultList = listVaultsResult.getVaultList();
    marker = listVaultsResult.getMarker();
    for (DescribeVaultOutput vault : vaultList) {
        System.out.println(
                "\nCreationDate: " + vault.getCreationDate() +
                "\nLastInventoryDate: " + vault.getLastInventoryDate() +
                "\nNumberOfArchives: " + vault.getNumberOfArchives() +
                "\nSizeInBytes: " + vault.getSizeInBytes() +
                "\nVaultARN: " + vault.getVaultARN() +
                "\nVaultName: " + vault.getVaultName());
    }
} while (marker != null);
```

Dans le segment de code précédent, si vous ne spécifiez pas la `Limit` valeur dans la demande, Amazon Glacier renvoie jusqu'à 10 coffres-forts, tels que définis par l'API Amazon Glacier. Si un plus grand nombre de coffres doivent figurer dans la liste, le champ `marker` de la réponse contient l'Amazon Resource Name (ARN) du coffre à partir duquel vous pouvez poursuivre la liste avec une nouvelle demande. Si tel n'est pas le cas, le champ `marker` est null.

Notez que les informations renvoyées pour chaque coffre de la liste sont identiques à celles que vous obtenez en appelant la méthode `describeVault` pour un coffre spécifique.

> **Note**
>
> La méthode `listVaults` appelle l'API REST sous-jacente (consultez Affichage des coffres (GET vaults)).

Exemple : Extraction des métadonnées d'un coffre à l'aide du kit SDK Amazon pour Java

Pour obtenir un exemple de code pratique, consultez la section Exemple : création d'un coffre-fort à l'aide du AWS SDK pour Java. L'exemple de code Java crée un coffre et extrait les métadonnées de celui-ci.

Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

Rubriques

- Extraire des métadonnées de coffre pour un coffre
- Extraire les métadonnées de coffre pour tous les coffres dans une région
- Exemple : extraction des métadonnées du coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Extraire des métadonnées de coffre pour un coffre

Vous pouvez récupérer les métadonnées d'un coffre-fort spécifique ou de tous les coffres-forts d'une AWS région spécifique. Voici les étapes d'extraction de métadonnées de coffre pour un coffre spécifique à l'aide de l'API de bas niveau du kit AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Fournissez les informations de demande en créant une instance de la classe `DescribeVaultRequest`.

   Amazon Glacier (Amazon Glacier) vous demande de fournir un nom de coffre-fort et votre numéro de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.

3. Exécutez la méthode `DescribeVault` en fournissant l'objet de demande comme paramètre.

   Les informations de métadonnées du coffre renvoyées par Amazon Glacier sont disponibles dans l'`DescribeVaultResult`objet.

L'extrait de code C# suivant illustre les étapes précédentes. L'extrait de code extrait les informations de métadonnées d'un coffre existant dans la région USA Ouest (Oregon).

```
AmazonGlacierClient client;
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);

DescribeVaultRequest describeVaultRequest = new DescribeVaultRequest()
{
  VaultName = "*** Provide vault name ***"
};
DescribeVaultResponse describeVaultResponse =
 client.DescribeVault(describeVaultRequest);
Console.WriteLine("\nVault description...");
Console.WriteLine(
   "\nVaultName: " + describeVaultResponse.VaultName +
   "\nVaultARN: " + describeVaultResponse.VaultARN +
   "\nVaultCreationDate: " + describeVaultResponse.CreationDate +
   "\nNumberOfArchives: " + describeVaultResponse.NumberOfArchives +
   "\nSizeInBytes: " + describeVaultResponse.SizeInBytes +
   "\nLastInventoryDate: " + describeVaultResponse.LastInventoryDate
   );
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Description de coffre (GET vault).

Extraire les métadonnées de coffre pour tous les coffres dans une région

Vous pouvez également utiliser `ListVaults` cette méthode pour récupérer les métadonnées de tous les coffres-forts d'une AWS région spécifique.

L'extrait de code C# suivant extrait la liste des coffres de la région USA Ouest (Oregon). La demande limite à 5 le nombre de coffres renvoyés dans la réponse. L'extrait de code effectue ensuite une série d'`ListVaults`appels pour récupérer la liste complète des coffres-forts dans la AWS région.

```
AmazonGlacierClient client;
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);
string lastMarker = null;
Console.WriteLine("\n List of vaults in your account in the specific AWS Region ...");
do
{
  ListVaultsRequest request = new ListVaultsRequest()
  {
    Limit = 5,
    Marker = lastMarker
  };
  ListVaultsResponse response = client.ListVaults(request);

  foreach (DescribeVaultOutput output in response.VaultList)
  {
    Console.WriteLine("Vault Name: {0} \tCreation Date: {1} \t #of archives: {2}",
                      output.VaultName, output.CreationDate, output.NumberOfArchives);
  }
  lastMarker = response.Marker;
} while (lastMarker != null);
```

Dans le segment de code précédent, si vous ne spécifiez pas la `Limit` valeur dans la demande, Amazon Glacier renvoie jusqu'à 10 coffres-forts, tels que définis par l'API Amazon Glacier.

Notez que les informations renvoyées pour chaque coffre de la liste sont identiques à celles que vous obtenez en appelant la méthode `DescribeVault` pour un coffre spécifique.

> **Note**
>
> La méthode `ListVaults` appelle l'API REST sous-jacente (consultez Affichage des coffres (GET vaults)).

Exemple : extraction des métadonnées du coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Pour obtenir un exemple de code pratique, consultez la section Exemple : opérations de coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET. L'exemple de code C# crée un coffre et extrait les métadonnées de coffre.

Extraction de métadonnées de coffre à l'aide de l'API REST

Pour afficher les coffres à l'aide de l'API REST, consultez la section Affichage des coffres (GET vaults). Pour décrire un coffre, consultez la section Description de coffre (GET vault).

Récupération des métadonnées du coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Cet exemple montre comment récupérer les informations et les métadonnées du coffre-fort dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : extraction des métadonnées du coffre-fort à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface

2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

Exemple : extraction des métadonnées du coffre-fort à l'aide du AWS CLI

- Utilisez la `describe-vault` commande pour décrire un coffre-fort nommé *awsexamplevault* sous « compte *111122223333* ».

  ```
  aws glacier describe-vault --vault-name awsexamplevault --account-id 111122223333
  ```

Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier

Une fois que vous avez chargé votre première archive dans votre coffre-fort, Amazon Glacier (Amazon Glacier) crée automatiquement un inventaire du coffre-fort, puis le met à jour environ une fois par jour. Une fois qu'Amazon Glacier a créé le premier inventaire, il faut généralement une demi-journée et jusqu'à un jour avant que ce stock ne soit disponible pour être extrait. Vous pouvez récupérer l'inventaire d'un coffre auprès d'Amazon Glacier en suivant le processus en deux étapes suivant :

1. Lancez une tâche d'extraction d'inventaire à l'aide de l'opération Lancement d'une tâche (POST jobs).

> **Important**
>
> Une stratégie d'extraction de données peut entraîner l'échec de votre demande de lancement de tâche d'extraction et générer une exception `PolicyEnforcedException`. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier. Pour plus d'informations sur l'exception `PolicyEnforcedException`, consultez la section Réponses d'erreur.

2. Une fois la tâche terminée, téléchargez les octets à l'aide de l'opération Génération de sortie de tâche (GET output).

Par exemple, l'extraction d'une archive ou d'un inventaire de stock implique le lancement d'une tâche d'extraction en premier lieu. La demande de tâche est exécutée de façon asynchrone. Lorsque vous lancez une tâche d'extraction, Amazon Glacier crée une tâche et renvoie un ID de tâche dans la réponse. Une fois qu'Amazon Glacier a terminé la tâche, vous pouvez obtenir la sortie correspondante, les octets d'archive ou les données d'inventaire du coffre.

Pour recevoir la sortie de la tâche, la tâche doit être terminée. Pour déterminer l'état de la tâche, vous disposez des options suivantes :

- Attendre la notification de fin de la tâche : vous pouvez spécifier une rubrique Amazon Simple Notification Service (Amazon SNS) sur laquelle Amazon Glacier pourra publier une notification une fois la tâche terminée. Vous pouvez spécifier une rubrique Amazon SNS en employant l'une des méthodes suivantes :
  - Spécifiez une rubrique Amazon SNS par tâche.

    Lorsque vous lancez une tâche, vous avez la possibilité de spécifier une rubrique Amazon SNS.
  - Définissez la configuration des notifications pour le coffre.

    Vous pouvez définir une configuration de notifications pour des événements spécifiques liés au coffre (consultez Configuration des notifications de coffre-fort dans Amazon Glacier). Amazon Glacier envoie un message à la rubrique SNS spécifiée chaque fois que l'événement spécifique se produit.

Si la configuration des notifications est définie dans le coffre et que vous spécifiez également une rubrique Amazon SNS lorsque vous lancez une tâche, Amazon Glacier envoie un message de fin de tâche aux deux rubriques.

Vous pouvez configurer la rubrique SNS de sorte qu'elle vous envoie une notification par e-mail ou qu'elle stocke le message dans une file d'attente Amazon Simple Queue Service (Amazon SQS) que votre application peut interroger. Lorsqu'un message apparaît dans la file d'attente, vous pouvez vérifier si la tâche s'est correctement achevée, puis télécharger la sortie correspondante.

- Demandez des informations sur le poste de manière explicite : Amazon Glacier fournit également une description de l'opération de travail (Description de la tâche (GET JobID)) qui vous permet de demander des informations sur le poste. Vous pouvez envoyer régulièrement cette demande pour obtenir des informations sur la tâche. Toutefois, il est préférable d'utiliser des notifications Amazon SNS.

> **Note**
>
> Les informations que vous obtenez par le biais de la notification SNS sont identiques à celles obtenues en appelant une opération de description de tâche.

Rubriques

- A propos de l'inventaire
- Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Téléchargement d'un inventaire de coffre à l'aide de l'API REST
- Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

A propos de l'inventaire

Amazon Glacier met à jour l'inventaire d'un coffre-fort au moins une fois par jour, à compter du jour où vous chargez une archive dans le coffre-fort pour la première fois. Si aucun ajout ou

aucune suppression d'archive n'a été effectué dans le coffre depuis le dernier inventaire, la date de l'inventaire n'est pas mise à jour. Lorsque vous lancez une tâche pour un inventaire de coffre-fort, Amazon Glacier renvoie le dernier inventaire généré, qui est un point-in-time instantané et non des données en temps réel. Notez qu'une fois qu'Amazon Glacier a créé le premier inventaire pour le coffre, il faut généralement une demi-journée, voire un jour, avant que cet inventaire ne soit disponible pour extraction.

Vous ne trouverez peut-être pas utile d'extraire un inventaire de coffre pour chaque chargement d'archive. Supposons toutefois que vous mainteniez une base de données côté client associant des métadonnées relatives aux archives que vous chargez sur Amazon Glacier. Ensuite, vous trouverez peut-être l'inventaire de coffre très utile pour rapprocher, au besoin, les informations contenues dans votre base de données avec l'inventaire de coffre réel. Vous pouvez limiter le nombre d'éléments de l'inventaire récupérés en filtrant par date de création de l'archive ou en définissant une limite. Pour plus d'informations sur les limites de l'extraction d'inventaire, consultez la page Extraction d'un inventaire par plage.

L'inventaire peut être retourné dans deux formats : valeurs séparées par une virgule (CSV) ou JSON. Vous avez également la possibilité de spécifier le format lorsque vous lancez la tâche d'inventaire. Le format par défaut est JSON. Pour plus d'informations sur les champs de données retournés dans un sortie de tâche d'inventaire, consultez la page Corps de la réponse de l'API d'obtention de la sortie de tâche.

Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

Voici les étapes d'extraction d'un inventaire de coffre à l'aide de l'API de bas niveau du kit AWS SDK pour Java. L'API de haut niveau ne prend pas en charge l'extraction d'un inventaire de coffre.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Lancez une tâche d'extraction d'inventaire en exécutant la méthode `initiateJob`.

   Exécutez `initiateJob` en fournissant les informations de la tâche dans un objet `InitiateJobRequest`.

> **Note**
>
> Notez que si un inventaire n'a pas été exécuté pour le coffre, une erreur est renvoyée. Amazon Glacier (Amazon Glacier) prépare un inventaire pour chaque coffre régulièrement, toutes les 24 heures.

Amazon Glacier renvoie un numéro de tâche en réponse. La réponse est disponible dans une instance de la classe `InitiateJobResult`.

```
InitiateJobRequest initJobRequest = new InitiateJobRequest()
    .withVaultName("*** provide vault name ***")
    .withJobParameters(
            new JobParameters()
                .withType("inventory-retrieval")
                .withSNSTopic("*** provide SNS topic ARN ****")
      );

InitiateJobResult initJobResult = client.initiateJob(initJobRequest);
String jobId = initJobResult.getJobId();
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification sur le coffre-fort ou indiqué une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

   Vous pouvez également interroger Amazon Glacier en appelant la `describeJob` méthode pour déterminer l'état d'achèvement de la tâche. Toutefois, l'approche recommandée consiste à utiliser une rubrique Amazon SNS pour la notification. L'exemple de code présenté dans la section suivante utilise Amazon SNS pour Amazon Glacier afin de publier un message.

4. Téléchargez la sortie de la tâche (données d'inventaire de coffre) en exécutant la méthode `getJobOutput`.

   Vous indiquez votre ID de compte, l'ID de la tâche et le nom du coffre en créant une instance de la classe `GetJobOutputRequest`. Si vous n'indiquez pas d'ID de compte, le système utilise

l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.

La sortie renvoyée par Amazon Glacier est disponible dans l'`GetJobOutputResult`objet.

```
GetJobOutputRequest jobOutputRequest = new GetJobOutputRequest()
        .withVaultName("*** provide vault name ***")
        .withJobId("*** provide job ID ***");
GetJobOutputResult jobOutputResult = client.getJobOutput(jobOutputRequest);
// jobOutputResult.getBody(); provides the output stream.
```

> **Note**
>
> Pour plus d'informations sur l'API REST sous-jacente liée à la tâche, consultez la section Opérations de tâche.

Exemple : Extraction d'un inventaire de coffre à l'aide du kit SDK Amazon pour Java

L'exemple de code Java suivant extrait l'inventaire du coffre spécifié.

L'exemple exécute les tâches suivantes :

- Création d'une rubrique Amazon Simple Notification Service (Amazon SNS).

  Amazon Glacier envoie une notification à cette rubrique une fois le travail terminé.
- Création d'une file d'attente Amazon Simple Queue Service (Amazon SQS).

  L'exemple attache une stratégie à la file d'attente pour permettre à la rubrique Amazon SNS de publier des messages dans la file d'attente.
- Lancement d'une tâche de téléchargement de l'archive spécifiée.

  Dans la demande de travail, le sujet Amazon SNS créé est spécifié afin qu'Amazon Glacier puisse publier une notification sur le sujet une fois le travail terminé.
- Recherche d'un message contenant l'ID de la tâche dans la file d'attente Amazon SQS.

Si un message est présent, l'exemple analyse le JSON et vérifie si la tâche s'est correctement déroulée. Si tel est le cas, il télécharge l'archive.

- Nettoyage par la suppression de la rubrique Amazon SNS et de la file d'attente Amazon SQS qu'il a créées.

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import com.fasterxml.jackson.core.JsonFactory;
import com.fasterxml.jackson.core.JsonParseException;
import com.fasterxml.jackson.core.JsonParser;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;

import com.amazonaws.AmazonClientException;
import com.amazonaws.auth.policy.Policy;
import com.amazonaws.auth.policy.Principal;
import com.amazonaws.auth.policy.Resource;
import com.amazonaws.auth.policy.Statement;
import com.amazonaws.auth.policy.Statement.Effect;
import com.amazonaws.auth.policy.actions.SQSActions;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.GetJobOutputRequest;
import com.amazonaws.services.glacier.model.GetJobOutputResult;
import com.amazonaws.services.glacier.model.InitiateJobRequest;
import com.amazonaws.services.glacier.model.InitiateJobResult;
import com.amazonaws.services.glacier.model.JobParameters;
import com.amazonaws.services.sns.AmazonSNSClient;
import com.amazonaws.services.sns.model.CreateTopicRequest;
import com.amazonaws.services.sns.model.CreateTopicResult;
import com.amazonaws.services.sns.model.DeleteTopicRequest;
import com.amazonaws.services.sns.model.SubscribeRequest;
import com.amazonaws.services.sns.model.SubscribeResult;
import com.amazonaws.services.sns.model.UnsubscribeRequest;
```

```
import com.amazonaws.services.sqs.AmazonSQSClient;
import com.amazonaws.services.sqs.model.CreateQueueRequest;
import com.amazonaws.services.sqs.model.CreateQueueResult;
import com.amazonaws.services.sqs.model.DeleteQueueRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesResult;
import com.amazonaws.services.sqs.model.Message;
import com.amazonaws.services.sqs.model.ReceiveMessageRequest;
import com.amazonaws.services.sqs.model.SetQueueAttributesRequest;


public class AmazonGlacierDownloadInventoryWithSQSPolling {

    public static String vaultName = "*** provide vault name ***";
    public static String snsTopicName = "*** provide topic name ***";
    public static String sqsQueueName = "*** provide queue name ***";
    public static String sqsQueueARN;
    public static String sqsQueueURL;
    public static String snsTopicARN;
    public static String snsSubscriptionARN;
    public static String fileName = "*** provide file name ***";
    public static String region = "*** region ***";
    public static long sleepTime = 600;
    public static AmazonGlacierClient client;
    public static AmazonSQSClient sqsClient;
    public static AmazonSNSClient snsClient;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier." + region + ".amazonaws.com");
        sqsClient = new AmazonSQSClient(credentials);
        sqsClient.setEndpoint("https://sqs." + region + ".amazonaws.com");
        snsClient = new AmazonSNSClient(credentials);
        snsClient.setEndpoint("https://sns." + region + ".amazonaws.com");

        try {
            setupSQS();

            setupSNS();

            String jobId = initiateJobRequest();
```

```
            System.out.println("Jobid = " + jobId);

            Boolean success = waitForJobToComplete(jobId, sqsQueueURL);
            if (!success) { throw new Exception("Job did not complete
successfully."); }

            downloadJobOutput(jobId);

            cleanUp();

        } catch (Exception e) {
            System.err.println("Inventory retrieval failed.");
            System.err.println(e);
        }
    }

    private static void setupSQS() {
        CreateQueueRequest request = new CreateQueueRequest()
            .withQueueName(sqsQueueName);
        CreateQueueResult result = sqsClient.createQueue(request);
        sqsQueueURL = result.getQueueUrl();

        GetQueueAttributesRequest qRequest = new GetQueueAttributesRequest()
            .withQueueUrl(sqsQueueURL)
            .withAttributeNames("QueueArn");

        GetQueueAttributesResult qResult = sqsClient.getQueueAttributes(qRequest);
        sqsQueueARN = qResult.getAttributes().get("QueueArn");

        Policy sqsPolicy =
            new Policy().withStatements(
                    new Statement(Effect.Allow)
                    .withPrincipals(Principal.AllUsers)
                    .withActions(SQSActions.SendMessage)
                    .withResources(new Resource(sqsQueueARN)));
        Map<String, String> queueAttributes = new HashMap<String, String>();
        queueAttributes.put("Policy", sqsPolicy.toJson());
        sqsClient.setQueueAttributes(new SetQueueAttributesRequest(sqsQueueURL,
queueAttributes));

    }
    private static void setupSNS() {
        CreateTopicRequest request = new CreateTopicRequest()
            .withName(snsTopicName);
```

```
        CreateTopicResult result = snsClient.createTopic(request);
        snsTopicARN = result.getTopicArn();

        SubscribeRequest request2 = new SubscribeRequest()
            .withTopicArn(snsTopicARN)
            .withEndpoint(sqsQueueARN)
            .withProtocol("sqs");
        SubscribeResult result2 = snsClient.subscribe(request2);

        snsSubscriptionARN = result2.getSubscriptionArn();
    }
    private static String initiateJobRequest() {

        JobParameters jobParameters = new JobParameters()
            .withType("inventory-retrieval")
            .withSNSTopic(snsTopicARN);

        InitiateJobRequest request = new InitiateJobRequest()
            .withVaultName(vaultName)
            .withJobParameters(jobParameters);

        InitiateJobResult response = client.initiateJob(request);

        return response.getJobId();
    }

    private static Boolean waitForJobToComplete(String jobId, String sqsQueueUrl)
throws InterruptedException, JsonParseException, IOException {

        Boolean messageFound = false;
        Boolean jobSuccessful = false;
        ObjectMapper mapper = new ObjectMapper();
        JsonFactory factory = mapper.getFactory();

        while (!messageFound) {
            List<Message> msgs = sqsClient.receiveMessage(
               new
ReceiveMessageRequest(sqsQueueUrl).withMaxNumberOfMessages(10)).getMessages();

            if (msgs.size() > 0) {
                for (Message m : msgs) {
                    JsonParser jpMessage = factory.createJsonParser(m.getBody());
                    JsonNode jobMessageNode = mapper.readTree(jpMessage);
                    String jobMessage = jobMessageNode.get("Message").textValue();
```

```
                    JsonParser jpDesc = factory.createJsonParser(jobMessage);
                    JsonNode jobDescNode = mapper.readTree(jpDesc);
                    String retrievedJobId = jobDescNode.get("JobId").textValue();
                    String statusCode = jobDescNode.get("StatusCode").textValue();
                    if (retrievedJobId.equals(jobId)) {
                        messageFound = true;
                        if (statusCode.equals("Succeeded")) {
                            jobSuccessful = true;
                        }
                    }
                }

            } else {
              Thread.sleep(sleepTime * 1000);
            }
          }
        return (messageFound && jobSuccessful);
    }

    private static void downloadJobOutput(String jobId) throws IOException {

        GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
            .withVaultName(vaultName)
            .withJobId(jobId);
        GetJobOutputResult getJobOutputResult =
client.getJobOutput(getJobOutputRequest);

        FileWriter fstream = new FileWriter(fileName);
        BufferedWriter out = new BufferedWriter(fstream);
        BufferedReader in = new BufferedReader(new
InputStreamReader(getJobOutputResult.getBody()));
        String inputLine;
        try {
            while ((inputLine = in.readLine()) != null) {
                out.write(inputLine);
            }
        }catch(IOException e) {
            throw new AmazonClientException("Unable to save archive", e);
        }finally{
            try {in.close();}  catch (Exception e) {}
            try {out.close();}  catch (Exception e) {}
        }
        System.out.println("Retrieved inventory to " + fileName);
```

```
    }

    private static void cleanUp() {
        snsClient.unsubscribe(new UnsubscribeRequest(snsSubscriptionARN));
        snsClient.deleteTopic(new DeleteTopicRequest(snsTopicARN));
        sqsClient.deleteQueue(new DeleteQueueRequest(sqsQueueURL));
    }
}
```

Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

Voici les étapes d'extraction d'un inventaire de coffre à l'aide de l'API de bas niveau du kit AWS SDK pour .NET. L'API de haut niveau ne prend pas en charge l'extraction d'un inventaire de coffre.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Lancez une tâche d'extraction d'inventaire en exécutant la méthode `InitiateJob`.

   Fournissez les informations de la tâche dans un objet `InitiateJobRequest`. Amazon Glacier (Amazon Glacier) renvoie un identifiant de tâche en réponse. La réponse est disponible dans une instance de la classe `InitiateJobResponse`.

```
AmazonGlacierClient client;
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);

InitiateJobRequest initJobRequest = new InitiateJobRequest()
{
  VaultName = vaultName,
  JobParameters = new JobParameters()
  {
    Type = "inventory-retrieval",
    SNSTopic = "*** Provide Amazon SNS topic arn ***",
  }
};
InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
string jobId = initJobResponse.JobId;
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification dans le coffre identifiant une rubrique Amazon Simple Notification Service (Amazon SNS) ou si vous avez spécifié une rubrique Amazon SNS lorsque vous avez lancé une tâche, Amazon Glacier envoie un message à cette rubrique une fois la tâche terminée. L'exemple de code présenté dans la section suivante utilise Amazon SNS pour Amazon Glacier afin de publier un message.

   Vous pouvez également interroger Amazon Glacier en appelant la `DescribeJob` méthode pour déterminer l'état d'achèvement de la tâche. Toutefois, l'approche recommandée consiste à utiliser une rubrique Amazon SNS pour la notification.

4. Téléchargez la sortie de la tâche (données d'inventaire de coffre) en exécutant la méthode `GetJobOutput`.

   Vous indiquez votre ID de compte, le nom du coffre et l'ID de la tâche en créant une instance de la classe `GetJobOutputRequest`. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.

   La sortie renvoyée par Amazon Glacier est disponible dans l'`GetJobOutputResponse`objet.

```
GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
{
  JobId = jobId,
  VaultName = vaultName
};

GetJobOutputResponse getJobOutputResponse =
 client.GetJobOutput(getJobOutputRequest);
using (Stream webStream = getJobOutputResponse.Body)
{
   using (Stream fileToSave = File.OpenWrite(fileName))
   {
     CopyStream(webStream, fileToSave);
   }
}
```

> **Note**
>
> Pour plus d'informations sur l'API REST sous-jacente liée à la tâche, consultez la section Opérations de tâche.

Exemple : extraction de l'inventaire d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'exemple de code C# suivant extrait l'inventaire de coffre pour le coffre spécifié.

L'exemple exécute les tâches suivantes :

- Configuration d'une rubrique Amazon SNS.

  Amazon Glacier envoie une notification à cette rubrique une fois le travail terminé.
- Configuration d'une file d'attente Amazon SQS.

  L'exemple attache une politique à la file d'attente pour permettre à la rubrique Amazon SNS de publier des messages.
- Lancement d'une tâche de téléchargement de l'archive spécifiée.

  Dans la demande de travail, l'exemple indique le sujet Amazon SNS afin qu'Amazon Glacier puisse envoyer un message une fois le travail terminé.
- Recherche périodique d'un message dans la file d'attente Amazon SQS.

  Si un message est présent, l'exemple analyse le JSON et vérifie si la tâche s'est correctement déroulée. Si tel est le cas, il télécharge l'archive. L'exemple de code utilise la bibliothèque JSON.NET (consultez le document JSON.NET) pour analyser le JSON.
- Nettoyage par la suppression de la rubrique Amazon SNS et de la file d'attente Amazon SQS qu'il a créées.

Example

```
using System;
using System.Collections.Generic;
using System.IO;
```

```
using System.Threading;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Amazon.SQS;
using Amazon.SQS.Model;
using Newtonsoft.Json;

namespace glacier.amazon.com.docsamples
{
  class VaultInventoryJobLowLevelUsingSNSSQS
  {
    static string topicArn;
    static string queueUrl;
    static string queueArn;
    static string vaultName = "*** Provide vault name ***";
    static string fileName  = "*** Provide file name and path where to store inventory
 ***";
    static AmazonSimpleNotificationServiceClient snsClient;
    static AmazonSQSClient sqsClient;
    const string SQS_POLICY =
        "{" +
        "    \"Version\" : \"2012-10-17\",&TCX5-2025-waiver;" +
        "    \"Statement\" : [" +
        "        {" +
        "            \"Sid\" : \"sns-rule\"," +
        "            \"Effect\" : \"Allow\"," +
        "            \"Principal\" : {\"AWS\" : \"arn:aws:iam::123456789012:root\" }," 
 +
        "            \"Action\"    : \"sqs:SendMessage\"," +
        "            \"Resource\"  : \"{QuernArn}\"," +
        "            \"Condition\" : {" +
        "                \"ArnLike\" : {" +
        "                    \"aws:SourceArn\" : \"{TopicArn}\"" +
        "                }" +
        "            }" +
        "        }" +
        "    ]" +
        "}";

    public static void Main(string[] args)
```

```
    {
      AmazonGlacierClient client;
      try
      {
        using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
            Console.WriteLine("Setup SNS topic and SQS queue.");
            SetupTopicAndQueue();
            Console.WriteLine("To continue, press Enter"); Console.ReadKey();

            Console.WriteLine("Retrieve Inventory List");
            GetVaultInventory(client);
        }
        Console.WriteLine("Operations successful.");
        Console.WriteLine("To continue, press Enter"); Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      finally
      {
       // Delete SNS topic and SQS queue.
       snsClient.DeleteTopic(new DeleteTopicRequest() { TopicArn = topicArn });
       sqsClient.DeleteQueue(new DeleteQueueRequest() { QueueUrl = queueUrl });
      }
    }

    static void SetupTopicAndQueue()
    {
      long ticks = DateTime.Now.Ticks;

      // Setup SNS topic.
      snsClient = new
AmazonSimpleNotificationServiceClient(Amazon.RegionEndpoint.USWest2);
      sqsClient = new AmazonSQSClient(Amazon.RegionEndpoint.USWest2);

      topicArn = snsClient.CreateTopic(new CreateTopicRequest { Name =
"GlacierDownload-" + ticks }).TopicArn;
      Console.Write("topicArn: "); Console.WriteLine(topicArn);

      CreateQueueRequest createQueueRequest =  new CreateQueueRequest();
      createQueueRequest.QueueName = "GlacierDownload-" + ticks;
      CreateQueueResponse createQueueResponse =
sqsClient.CreateQueue(createQueueRequest);
```

```
      queueUrl = createQueueResponse.QueueUrl;
      Console.Write("QueueURL: "); Console.WriteLine(queueUrl);

      GetQueueAttributesRequest getQueueAttributesRequest = new
GetQueueAttributesRequest();
      getQueueAttributesRequest.AttributeNames = new List<string> { "QueueArn" };
      getQueueAttributesRequest.QueueUrl = queueUrl;
      GetQueueAttributesResponse response =
sqsClient.GetQueueAttributes(getQueueAttributesRequest);
      queueArn = response.QueueARN;
      Console.Write("QueueArn: ");Console.WriteLine(queueArn);

      // Setup the Amazon SNS topic to publish to the SQS queue.
      snsClient.Subscribe(new SubscribeRequest()
      {
        Protocol = "sqs",
        Endpoint = queueArn,
        TopicArn = topicArn
      });

      // Add the policy to the queue so SNS can send messages to the queue.
      var policy = SQS_POLICY.Replace("{TopicArn}", topicArn).Replace("{QuernArn}",
queueArn);

      sqsClient.SetQueueAttributes(new SetQueueAttributesRequest()
      {
          QueueUrl = queueUrl,
          Attributes = new Dictionary<string, string>
          {
              { QueueAttributeName.Policy, policy }
          }
      });

   }

   static void GetVaultInventory(AmazonGlacierClient client)
   {
      // Initiate job.
      InitiateJobRequest initJobRequest = new InitiateJobRequest()
      {
        VaultName = vaultName,
        JobParameters = new JobParameters()
        {
          Type = "inventory-retrieval",
```

```
          Description = "This job is to download a vault inventory.",
          SNSTopic = topicArn,
        }
      };

      InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
      string jobId = initJobResponse.JobId;

      // Check queue for a message and if job completed successfully, download
inventory.
      ProcessQueue(jobId, client);
    }

    private static void ProcessQueue(string jobId, AmazonGlacierClient client)
    {
      ReceiveMessageRequest receiveMessageRequest = new ReceiveMessageRequest()
{ QueueUrl = queueUrl, MaxNumberOfMessages = 1 };
      bool jobDone = false;
      while (!jobDone)
      {
        Console.WriteLine("Poll SQS queue");
        ReceiveMessageResponse receiveMessageResponse =
sqsClient.ReceiveMessage(receiveMessageRequest);
        if (receiveMessageResponse.Messages.Count == 0)
        {
          Thread.Sleep(10000 * 60);
          continue;
        }
        Console.WriteLine("Got message");
        Message message = receiveMessageResponse.Messages[0];
        Dictionary<string, string> outerLayer =
JsonConvert.DeserializeObject<Dictionary<string, string>>(message.Body);
        Dictionary<string, object> fields =
JsonConvert.DeserializeObject<Dictionary<string, object>>(outerLayer["Message"]);
        string statusCode = fields["StatusCode"] as string;

        if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_SUCCEEDED,
StringComparison.InvariantCultureIgnoreCase))
        {
          Console.WriteLine("Downloading job output");
          DownloadOutput(jobId, client); // Save job output to the specified file
location.
        }
```

```
        else if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_FAILED,
 StringComparison.InvariantCultureIgnoreCase))
          Console.WriteLine("Job failed... cannot download the inventory.");

        jobDone = true;
        sqsClient.DeleteMessage(new DeleteMessageRequest() { QueueUrl = queueUrl,
 ReceiptHandle = message.ReceiptHandle });
      }
    }

    private static void DownloadOutput(string jobId, AmazonGlacierClient client)
    {
      GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
      {
        JobId = jobId,
        VaultName = vaultName
      };

      GetJobOutputResponse getJobOutputResponse =
 client.GetJobOutput(getJobOutputRequest);
      using (Stream webStream = getJobOutputResponse.Body)
      {
        using (Stream fileToSave = File.OpenWrite(fileName))
        {
          CopyStream(webStream, fileToSave);
        }
      }
    }

    public static void CopyStream(Stream input, Stream output)
    {
      byte[] buffer = new byte[65536];
      int length;
      while ((length = input.Read(buffer, 0, buffer.Length)) > 0)
      {
        output.Write(buffer, 0, length);
      }
    }
  }
}
```

Téléchargement d'un inventaire de coffre à l'aide de l'API REST

Pour télécharger un inventaire de coffre à l'aide de l'API REST

Le téléchargement d'un inventaire de coffre est effectué en deux étapes.

1. Lancez une tâche de type `inventory-retrieval`. Pour de plus amples informations, veuillez consulter Lancement d'une tâche (POST jobs).
2. Une fois la tâche terminée, téléchargez les données d'inventaire. Pour de plus amples informations, veuillez consulter Génération de sortie de tâche (GET output).

Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Procédez comme suit pour télécharger un inventaire de coffres-forts dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : téléchargement d'un inventaire de coffre-fort à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

   ```
   aws help
   ```

- Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

```
aws glacier list-vaults --account-id 123456789012
```

- Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

```
aws configure list
```

Exemple : téléchargement d'un inventaire de coffre-fort à l'aide du AWS CLI

1. Utilisez la commande `initiate-job` pour démarrer une tâche d'extraction d'inventaire.

```
aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
job-parameters='{"Type": "inventory-retrieval"}'
```

Sortie attendue :

```
{
    "location": "/111122223333/vaults/awsexamplevault/jobs/*** jobid ***",
    "jobId": "*** jobid ***"
}
```

2. Utilisez la commande `describe-job` pour vérifier le statut de la tâche d'extraction précédente.

```
aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
job-id *** jobid ***
```

Sortie attendue :

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "*** vault arn ***",
```

```
    "Completed": false,
    "JobId": "*** jobid ***",
    "Action": "InventoryRetrieval",
    "CreationDate": "*** job creation date ***",
    "StatusCode": "InProgress"
}
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. L'ID de tâche n'expire pas pendant au moins 24 heures après qu'Amazon Glacier ait terminé la tâche. Si vous avez défini une configuration de notification sur le coffre-fort ou indiqué une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

   Vous pouvez définir la configuration des notifications pour des événements spécifiques concernant le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier. Amazon Glacier envoie un message à la rubrique SNS spécifiée chaque fois que des événements spécifiques se produisent.

4. Une fois l'opération terminée, utilisez la commande `get-job-output` pour télécharger le travail d'extraction dans le fichier `output.json`.

```
aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
 --job-id *** jobid *** output.json
```

   Cette commande produit un fichier avec les champs suivants.

```
{
"VaultARN":"arn:aws:glacier:region:111122223333:vaults/awsexamplevault",
"InventoryDate":"*** job completion date ***",
"ArchiveList":[
{"ArchiveId":"*** archiveid ***",
"ArchiveDescription":"*** archive description (if set) ***",
"CreationDate":"*** archive creation date ***",
"Size":"*** archive size (in bytes) ***",
"SHA256TreeHash":"*** archive hash ***"
}
{"ArchiveId":
...
```

```
]}
```

Configuration des notifications de coffre-fort dans Amazon Glacier

La récupération de tout élément d'Amazon Glacier, tel qu'une archive d'un coffre-fort ou un inventaire de coffre-fort, est un processus en deux étapes.

1. Lancez une tâche d'extraction.
2. Une fois la tâche terminée, téléchargez la sortie correspondante.

Vous pouvez définir une configuration de notifications sur un coffre de sorte qu'un message soit envoyé à une rubrique Amazon Simple Notification Service (Amazon SNS) dès qu'une tâche se termine.

Rubriques

- Configuration des notifications de coffre-fort dans Amazon Glacier : concepts généraux
- Configuration des notifications de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Configuration des notifications de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Configuration des notifications Vault dans Amazon Glacier à l'aide de l'API REST
- Configuration des notifications Vault à l'aide de la console Amazon Glacier
- Configuration des notifications du coffre-fort à l'aide du AWS Command Line Interface

Configuration des notifications de coffre-fort dans Amazon Glacier : concepts généraux

Une demande de travail de récupération Amazon Glacier est exécutée de manière asynchrone. Vous devez attendre qu'Amazon Glacier ait terminé le travail avant de pouvoir obtenir ses résultats. Vous pouvez régulièrement interroger Amazon Glacier pour déterminer le statut de la tâche, mais cette approche n'est pas optimale. Amazon Glacier prend également en charge les notifications. Lorsqu'une tâche se termine, elle peut publier un message dans une rubrique Amazon Simple Notification Service (Amazon SNS). Pour pouvoir utiliser cette fonctionnalité, vous devez définir une configuration de notifications sur le coffre. Dans la configuration, vous identifiez un ou plusieurs

événements et une rubrique Amazon SNS à laquelle vous souhaitez qu'Amazon Glacier envoie un message lorsque l'événement se produit.

Amazon Glacier définit les événements spécifiquement liés à l'achèvement des tâches (`ArchiveRetrievalCompleted`,`InventoryRetrievalCompleted`) que vous pouvez ajouter à la configuration des notifications du coffre-fort. Lorsqu'une tâche spécifique est terminée, Amazon Glacier publie un message de notification dans la rubrique SNS.

La configuration de notification est un document JSON comme illustré dans l'exemple suivant.

```
{
   "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic",
   "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Vous ne pouvez configurer qu'une seule rubrique Amazon SNS par coffre.

> **Note**
>
> L'ajout d'une configuration de notification à un coffre-fort oblige Amazon Glacier à envoyer une notification chaque fois que l'événement spécifié dans la configuration des notifications se produit. Vous pouvez également spécifier une rubrique Amazon SNS dans chaque demande de lancement de tâche. Si vous ajoutez à la fois la configuration des notifications dans le coffre et que vous spécifiez une rubrique Amazon SNS dans votre demande de travail d'initiation, Amazon Glacier envoie les deux notifications.

Le message d'achèvement de la tâche envoyé par Amazon Glacier inclut des informations telles que le type de tâche (`InventoryRetrieval`,`ArchiveRetrieval`), l'état d'achèvement de la tâche, le nom de la rubrique SNS, le code d'état de la tâche et l'ARN du coffre-fort. Voici un exemple de notification envoyée par Amazon Glacier à une rubrique SNS après la fin d'une `InventoryRetrieval` tâche.

```
{
 "Action": "InventoryRetrieval",
 "ArchiveId": null,
 "ArchiveSizeInBytes": null,
 "Completed": true,
 "CompletionDate": "2012-06-12T22:20:40.790Z",
```

```
 "CreationDate": "2012-06-12T22:20:36.814Z",
 "InventorySizeInBytes":11693,
 "JobDescription": "my retrieval job",
 "JobId":"HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID",
 "SHA256TreeHash":null,
 "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic",
 "StatusCode":"Succeeded",
 "StatusMessage": "Succeeded",
 "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
}
```

Si le champ `Completed` a pour valeur true, vous devez également contrôler la valeur du champ `StatusCode` afin de vérifier si la tâche a abouti ou échoué.

> **Note**
>
> La rubrique Amazon SNS doit autoriser le coffre à publier une notification. Par défaut, seul le propriétaire de la rubrique Amazon SNS peut publier un message dans cette rubrique. Toutefois, si la rubrique Amazon SNS et le coffre appartiennent à des propriétaires différents Comptes AWS, vous devez configurer la rubrique Amazon SNS pour accepter les publications du coffre. Vous pouvez configurer la politique de rubrique Amazon SNS dans la console Amazon SNS.

Pour plus d'informations sur Amazon SNS, consultez Démarrage avec Amazon SNS.

Configuration des notifications de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

Voici les étapes pour configurer des notifications sur un coffre à l'aide de l'API de bas niveau de AWS SDK pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Indiquez les informations sur la configuration de notification en créant une instance de la classe `SetVaultNotificationsRequest`.

Vous devez fournir le nom du coffre, les informations sur la configuration de notification et l'ID de compte. En spécifiant une configuration de notification, vous fournissez l'Amazon Resource Name (ARN) d'une rubrique Amazon SNS existante et un ou plusieurs événements pour lesquels vous souhaitez être notifié. Pour obtenir la liste des événements pris en charge, consultez la section Définition d'une configuration de notification de coffre (PUT notification-configuration).

3. Exécutez la méthode `setVaultNotifications` en fournissant l'objet de demande comme paramètre.

L'extrait de code Java suivant illustre les étapes précédentes. Cet extrait définit une configuration de notification sur un coffre. La configuration demande à Amazon Glacier (Amazon Glacier) d'envoyer une notification à la rubrique Amazon SNS spécifiée lorsque l'`ArchiveRetrievalCompleted`événement ou l'`InventoryRetrievalCompleted`événement se produit.

```
SetVaultNotificationsRequest request = new SetVaultNotificationsRequest()
        .withAccountId("-")
        .withVaultName("*** provide vault name ***")
        .withVaultNotificationConfig(
                new VaultNotificationConfig()
                .withSNSTopic("*** provide SNS topic ARN ***")
                .withEvents("ArchiveRetrievalCompleted", "InventoryRetrievalCompleted")
         );
client.setVaultNotifications(request);
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Opérations de coffre.

Exemple : Configuration de la configuration des notifications sur un coffre-fort à l'aide du AWS SDK pour Java

L'exemple de code Java suivant définit la configuration des notifications d'un coffre, la supprime, puis la restaure. Pour step-by-step obtenir des instructions sur la façon d'exécuter l'exemple suivant, reportez-vous àUtilisation du AWS SDK pour Java avec Amazon Glacier.

Example

```
import java.io.IOException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.DeleteVaultNotificationsRequest;
import com.amazonaws.services.glacier.model.GetVaultNotificationsRequest;
import com.amazonaws.services.glacier.model.GetVaultNotificationsResult;
import com.amazonaws.services.glacier.model.SetVaultNotificationsRequest;
import com.amazonaws.services.glacier.model.VaultNotificationConfig;


public class AmazonGlacierVaultNotifications {

    public static AmazonGlacierClient client;
    public static String vaultName = "*** provide vault name ****";
    public static String snsTopicARN = "*** provide sns topic ARN ***";

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-east-1.amazonaws.com/");

        try {

            System.out.println("Adding notification configuration to the vault.");
            setVaultNotifications();
            getVaultNotifications();
            deleteVaultNotifications();

        } catch (Exception e) {
            System.err.println("Vault operations failed." + e.getMessage());
        }
    }

    private static void setVaultNotifications() {
        VaultNotificationConfig config = new VaultNotificationConfig()
            .withSNSTopic(snsTopicARN)
            .withEvents("ArchiveRetrievalCompleted", "InventoryRetrievalCompleted");

        SetVaultNotificationsRequest request = new SetVaultNotificationsRequest()
```

```
                .withVaultName(vaultName)
                .withVaultNotificationConfig(config);

        client.setVaultNotifications(request);
        System.out.println("Notification configured for vault: " + vaultName);
    }

    private static void getVaultNotifications() {
        VaultNotificationConfig notificationConfig = null;
        GetVaultNotificationsRequest request = new GetVaultNotificationsRequest()
                .withVaultName(vaultName);
        GetVaultNotificationsResult result = client.getVaultNotifications(request);
        notificationConfig = result.getVaultNotificationConfig();

        System.out.println("Notifications configuration for vault: "
                + vaultName);
        System.out.println("Topic: " + notificationConfig.getSNSTopic());
        System.out.println("Events: " + notificationConfig.getEvents());
    }

    private static void deleteVaultNotifications() {
            DeleteVaultNotificationsRequest request = new
 DeleteVaultNotificationsRequest()
                .withVaultName(vaultName);
            client.deleteVaultNotifications(request);
            System.out.println("Notifications configuration deleted for vault: " +
 vaultName);
    }
}
```

Configuration des notifications de coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

Voici les étapes pour configurer des notifications sur un coffre à l'aide de l'API de bas niveau de AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle se trouve le coffre. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Indiquez les informations sur la configuration de notification en créant une instance de la classe `SetVaultNotificationsRequest`.

   Vous devez fournir le nom du coffre, les informations sur la configuration de notification et l'ID de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.

   En spécifiant une configuration de notification, vous fournissez l'Amazon Resource Name (ARN) d'une rubrique Amazon SNS existante et un ou plusieurs événements pour lesquels vous souhaitez être notifié. Pour obtenir la liste des événements pris en charge, consultez la section Définition d'une configuration de notification de coffre (PUT notification-configuration).
3. Exécutez la méthode `SetVaultNotifications` en fournissant l'objet de demande comme paramètre.
4. Après avoir défini une configuration de notification sur un coffre, vous pouvez extraire les informations sur la configuration en appelant la méthode `GetVaultNotifications` et les retirer en appelant la méthode `DeleteVaultNotifications` fournie par le client.

Exemple : Configuration de la configuration des notifications sur un coffre-fort à l'aide du AWS SDK pour .NET

L'exemple de code C# suivant présente les étapes précédentes. L'exemple définit la configuration de notifications pour le coffre (« `examplevault` ») de la région USA Ouest (Oregon), extrait la configuration, puis la supprime. La configuration demande à Amazon Glacier (Amazon Glacier) d'envoyer une notification à la rubrique Amazon SNS spécifiée lorsque l'`ArchiveRetrievalCompleted`événement ou l'`InventoryRetrievalCompleted`événement se produit.

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Opérations de coffre.

Pour step-by-step obtenir des instructions relatives à l'exécution de l'exemple suivant, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté et fournir un nom de coffre existant et une rubrique Amazon SNS.

Example

```
using System;
using System.Collections.Generic;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class VaultNotificationSetGetDelete
  {
    static string vaultName   = "examplevault";
    static string snsTopicARN = "*** Provide Amazon SNS topic ARN ***";

    static IAmazonGlacier client;

    public static void Main(string[] args)
    {
      try
      {
        using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Adding notification configuration to the vault.");
          SetVaultNotificationConfig();
          GetVaultNotificationConfig();
          Console.WriteLine("To delete vault notification configuration, press Enter");
          Console.ReadKey();
          DeleteVaultNotificationConfig();
        }
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }

    static void SetVaultNotificationConfig()
    {

      SetVaultNotificationsRequest request = new SetVaultNotificationsRequest()
      {
        VaultName = vaultName,
```

```
        VaultNotificationConfig = new VaultNotificationConfig()
        {
          Events   = new List<string>() { "ArchiveRetrievalCompleted",
 "InventoryRetrievalCompleted" },
          SNSTopic = snsTopicARN
        }
      };
      SetVaultNotificationsResponse response = client.SetVaultNotifications(request);
    }

    static void GetVaultNotificationConfig()
    {
      GetVaultNotificationsRequest request = new GetVaultNotificationsRequest()
      {
        VaultName = vaultName,
        AccountId = "-"
      };
      GetVaultNotificationsResponse response = client.GetVaultNotifications(request);
      Console.WriteLine("SNS Topic ARN: {0}",
 response.VaultNotificationConfig.SNSTopic);
      foreach (string s in response.VaultNotificationConfig.Events)
        Console.WriteLine("Event : {0}", s);
    }

    static void DeleteVaultNotificationConfig()
    {
      DeleteVaultNotificationsRequest request = new DeleteVaultNotificationsRequest()
      {
        VaultName = vaultName
      };
      DeleteVaultNotificationsResponse response =
 client.DeleteVaultNotifications(request);
    }
  }
}
```

Configuration des notifications Vault dans Amazon Glacier à l'aide de l'API REST

Pour configurer des notifications de coffre à l'aide de l'API REST, consultez la section Définition d'une configuration de notification de coffre (PUT notification-configuration). En outre, vous pouvez également extraire des notifications de coffre (Obtenir les notifications de coffre (GET notification-

configuration)) et supprimer des notifications de coffre (Supprimer les notifications de coffre (DELETE notification-configuration)).

Configuration des notifications Vault à l'aide de la console Amazon Glacier

Cette section explique comment configurer les notifications de coffre-fort à l'aide de la console Amazon Glacier. Lorsque vous configurez des notifications, vous spécifiez les événements d'achèvement de tâche qui envoient une notification à une rubrique Amazon Simple Notification Service (Amazon SNS). En plus de configurer des notifications pour le coffre, vous pouvez aussi spécifier la rubrique dans laquelle les notifications seront publiées au lancement d'une tâche. Si votre coffre est configuré pour envoyer une notification pour un événement spécifique et que vous configurez également des notifications dans la demande de lancement de tâche, deux notifications sont envoyées.

Pour configurer une notification de coffre

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Dans le panneau de navigation de gauche, choisissez Coffres.
3. Dans la liste des coffres, choisissez un coffre.
4. Dans la section Notifications, choisissez Modifier.
5. Sur la page Notifications d'événements, choisissez Activer les notifications.
6. Dans la section Notifications, choisissez l'une des options Amazon Simple Notification Service (Amazon SNS) suivantes, puis suivez les étapes correspondantes :

Options Amazon SNS	Action
Créer une rubrique SNS	1. Choisissez Créer une rubrique SNS. 2. Dans Nom de rubrique, donnez un nom à la nouvelle rubrique. Les noms de rubrique peuvent contenir jusqu'à 256 caractères. Les caractères alphanumériques, les traits d'union (-) et les traits de soulignement (_)

Options Amazon SNS	Action
	sont autorisés. Les noms des sujets doivent être uniques dans le compte et Région AWS. 3. (Facultatif) Si vous souhaitez vous abonner à la rubrique par SMS, saisissez un nom dans Nom d'affichage. Un nom d'affichage peut contenir jusqu'à 100 caractères.
Choisir une rubrique SNS existante	1. Choisissez Choisir une rubrique SNS existante. 2. Sous Spécifier une rubrique SNS, choisissez l'une des options suivantes : • Choisir parmi vos rubriques SNS Une liste déroulante Rubrique SNS s'affiche. Choisissez une rubrique existante dans la liste déroulante. • Saisir l'ARN de rubrique SNS Une zone de texte ARN de rubrique Amazon SNS s'affiche. Saisissez l'Amazon Resource Name (ARN) de votre rubrique SNS Voici comment se présente l'ARN d'une rubrique SNS : `arn:aws:sns: region:account-id :topic-name` Vous pouvez trouver l'ARN d'une rubrique SNS dans la console Amazon SNS.

7. Sous Événements, sélectionnez l'événement (ou les deux) qui va déclencher l'envoi de notifications :

   - Pour n'envoyer une notification qu'à l'achèvement d'une tâche d'extraction d'archive, sélectionnez Tâche d'extraction d'archives effectuée.
   - Pour n'envoyer une notification qu'à l'achèvement d'une tâche d'inventaire de coffre, sélectionnez Tâche d'extraction d'inventaires de coffre effectuée.

Configuration des notifications du coffre-fort à l'aide du AWS Command Line Interface

Cette section explique comment configurer des notifications de coffre à l'aide de l' AWS Command Line Interface. Lorsque vous configurez des notifications, vous spécifiez les événements d'achèvement de tâche qui déclenchent une notification vers une rubrique Amazon Simple Notification Service (Amazon SNS). En plus de configurer des notifications pour le coffre, vous pouvez également spécifier une rubrique vers laquelle publier une notification lorsque vous lancez une tâche. Si votre coffre est configuré pour avertir d'un événement spécifique et que vous spécifiez une notification dans la demande de lancement de tâche, deux notifications sont envoyées.

Pour configurer des notifications de coffre à l'aide de l'interface AWS CLI, procédez comme suit.

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : configurer les notifications du coffre-fort à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l’interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l’utilisateur de l’interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

- Essayez d'utiliser la commande help.

```
aws help
```

- Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

```
aws glacier list-vaults --account-id 123456789012
```

- Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

```
aws configure list
```

Exemple : configurer les notifications du coffre-fort à l'aide du AWS CLI

1. Utilisez la commande `set-vault-notifications` pour configurer les notifications qui seront envoyées lorsque des événements spécifiques se produiront sur un coffre. Par défaut, vous ne recevez pas de notifications.

```
aws glacier set-vault-notifications --vault-name examplevault --account-
id 111122223333 --vault-notification-config file://notificationconfig.json
```

2. La configuration de notification est un document JSON comme illustré dans l'exemple suivant.

```
{
   "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic",
   "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Pour plus d'informations sur l'utilisation des rubriques Amazon SNS pour Amazon Glacier, consultez Configuration des notifications de coffre-fort dans Amazon Glacier : concepts généraux

Pour plus d'informations sur Amazon SNS, consultez Démarrage avec Amazon SNS.

Supprimer un coffre-fort dans Amazon Glacier

Amazon Glacier (Amazon Glacier) supprime un coffre-fort uniquement s'il ne contient aucune archive lors du dernier inventaire calculé et s'il n'y a eu aucune écriture dans le coffre-fort depuis le dernier inventaire. Pour plus d'informations sur la suppression d'archives, consultez la section Supprimer une archive dans Amazon Glacier. Pour plus d'informations sur le téléchargement d'un inventaire de coffre, consultez la section Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

> **Note**
>
> Amazon Glacier prépare régulièrement un inventaire pour chaque coffre, toutes les 24 heures. Dans la mesure où l'inventaire peut ne pas refléter les informations les plus récentes, Amazon Glacier s'assure que le coffre est bien vide en vérifiant si des opérations d'écriture ont été effectuées depuis le dernier inventaire du coffre.

> **Note**
>
> Pour la suppression automatique des archives du coffre-fort, consultez Suppression automatique des archives du coffre-fort dans Amazon S3 Glacier.

Rubriques

- Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java
- Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Suppression d'un coffre-fort dans Amazon Glacier à l'aide de l'API REST
- Supprimer un coffre-fort vide à l'aide de la console Amazon Glacier
- Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour Java

Voici les étapes pour supprimer un coffre à l'aide de l'API de bas niveau du kit AWS SDK pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

Vous devez spécifier AWS la région à partir de laquelle vous souhaitez supprimer un coffre-fort. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Fournissez les informations de demande en créant une instance de la classe `DeleteVaultRequest`.

   Vous devez entrer le nom du coffre et l'ID du compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.

3. Exécutez la méthode `deleteVault` en fournissant l'objet de demande comme paramètre.

   Amazon Glacier (Amazon Glacier) supprime le coffre uniquement s'il est vide. Pour de plus amples informations, veuillez consulter Suppression de coffre (DELETE vault).

L'extrait de code Java suivant illustre les étapes précédentes.

```
try {
    DeleteVaultRequest request = new DeleteVaultRequest()
        .withVaultName("*** provide vault name ***");

    client.deleteVault(request);
    System.out.println("Deleted vault: " + vaultName);
} catch (Exception e) {
    System.err.println(e.getMessage());
}
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Suppression de coffre (DELETE vault).

Exemple : suppression d'un coffre-fort à l'aide du AWS SDK pour Java

Pour obtenir un exemple de code pratique, consultez la section Exemple : création d'un coffre-fort à l'aide du AWS SDK pour Java. L'exemple de code Java illustre les opérations de base sur les coffres, notamment la création et la suppression.

Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS SDK pour .NET

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode pour supprimer un coffre-fort.

Rubriques

- Suppression d'un coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Suppression d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Suppression d'un coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET

La catégorie `ArchiveTransferManager` de l'API de haut niveau comporte la méthode `DeleteVault` qui sert à supprimer un coffre.

Exemple : suppression d'un coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET

Pour obtenir un exemple de code pratique, consultez la section Exemple : opérations de coffre-fort à l'aide de l'API de haut niveau du AWS SDK pour .NET. L'exemple de code C# illustre les opérations de base sur les coffres, notamment la création et la suppression.

Suppression d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Voici les étapes permettant de supprimer un coffre à l'aide du kit AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région à partir de laquelle vous souhaitez supprimer un coffre-fort. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Fournissez les informations de demande en créant une instance de la classe `DeleteVaultRequest`.

   Vous devez entrer le nom du coffre et l'ID du compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.
3. Exécutez la méthode `DeleteVault` en fournissant l'objet de demande comme paramètre.

Amazon Glacier (Amazon Glacier) supprime le coffre uniquement s'il est vide. Pour de plus amples informations, veuillez consulter Suppression de coffre (DELETE vault).

L'extrait de code C# suivant illustre les étapes précédentes. L'extrait extrait les informations de métadonnées d'un coffre existant dans la région par défaut. AWS

```
AmazonGlacier client;
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USEast1);

DeleteVaultRequest request = new DeleteVaultRequest()
{
  VaultName = "*** provide vault name ***"
};

DeleteVaultResponse response = client.DeleteVault(request);
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Suppression de coffre (DELETE vault).

Exemple : suppression d'un coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET

Pour obtenir un exemple de code pratique, consultez la section Exemple : opérations de coffre-fort à l'aide de l'API de bas niveau du AWS SDK pour .NET. L'exemple de code C# illustre les opérations de base sur les coffres, notamment la création et la suppression.

Suppression d'un coffre-fort dans Amazon Glacier à l'aide de l'API REST

Pour supprimer un coffre à l'aide de l'API REST, consultez la section Suppression de coffre (DELETE vault).

Supprimer un coffre-fort vide à l'aide de la console Amazon Glacier

> **Note**
>
> Avant de supprimer un coffre, vous devez supprimer toutes les archives qu'il contient. Vous pouvez le faire en écrivant du code pour effectuer une demande de suppression d'archive

en utilisant l'API REST AWS SDK pour Java, le AWS SDK pour .NET, ou en utilisant le AWS Command Line Interface (AWS CLI). Pour plus d'informations sur la suppression d'archives, consultez la section Étape 5 : Supprimer une archive d'un coffre-fort dans Amazon Glacier.

Une fois le coffre vide, vous pouvez le supprimer en effectuant les étapes suivantes.

Pour supprimer un coffre vide à l'aide de la console Amazon Glacier

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la sur Amazon Glacier Console.
2. Sous Sélectionnez une région, choisissez l' Région AWS endroit où se trouve le coffre.
3. Dans le panneau de navigation de gauche, choisissez Coffres.
4. Dans la liste des coffres, sélectionnez la case d'option en regard du nom du coffre à supprimer, puis choisissez Supprimer en haut de la page.
5. Dans la boîte de dialogue Supprimer le coffre, confirmez que vous souhaitez supprimer le coffre en choisissant Supprimer.

   ⚠ Important

   La suppression d'un coffre ne peut pas être annulée.

6. Pour vérifier que vous avez bien supprimé le coffre, ouvrez la liste des coffres, puis saisissez le nom du coffre que vous avez supprimé. Si le coffre est introuvable, c'est que la suppression a abouti.

Suppression d'un coffre-fort dans Amazon Glacier à l'aide du AWS Command Line Interface

Vous pouvez supprimer des coffres-forts vides ou non vides dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : suppression d'un coffre vide à l'aide du AWS CLI
- Exemple : suppression d'un coffre non vide à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

Exemple : suppression d'un coffre vide à l'aide du AWS CLI

- Utilisez la commande `delete-vault` pour supprimer un coffre qui ne contient pas d'archives.

  - ```
    aws glacier delete-vault --vault-name awsexamplevault --account-id 111122223333
    ```

Exemple : suppression d'un coffre non vide à l'aide du AWS CLI

Amazon Glacier supprime un coffre-fort uniquement s'il ne contient aucune archive lors du dernier inventaire calculé et s'il n'y a eu aucune écriture dans le coffre-fort depuis le dernier inventaire. La
    ```

suppression d'un coffre non vide est un processus en trois étapes : récupérer l'archive dans le rapport IDs d'inventaire d'un coffre, supprimer chaque archive, puis supprimer le coffre.

1. Utilisez la commande `initiate-job` pour démarrer une tâche d'extraction d'inventaire.

```
aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
job-parameters '{"Type": "inventory-retrieval"}'
```

Sortie attendue :

```
{
    "location": "/111122223333/vaults/awsexamplevault/jobs/*** jobid ***",
    "jobId": "*** jobid ***"
}
```

2. Utilisez la commande `describe-job` pour vérifier le statut de la tâche d'extraction précédente.

```
aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
job-id *** jobid ***
```

Sortie attendue :

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "*** vault arn ***",
    "Completed": false,
    "JobId": "*** jobid ***",
    "Action": "InventoryRetrieval",
    "CreationDate": "*** job creation date ***",
    "StatusCode": "InProgress"
}
```

3. Attendez que la tâche se termine.

Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification sur le coffre-fort ou si vous avez spécifié une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

Vous pouvez définir la configuration des notifications pour des événements spécifiques concernant le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier. Amazon Glacier envoie un message à la rubrique SNS spécifiée chaque fois que l'événement spécifique se produit.

4. Une fois l'opération terminée, utilisez la commande `get-job-output` pour télécharger le travail d'extraction dans le fichier `output.json`.

```
aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
 --job-id *** jobid *** output.json
```

Cette commande produit un fichier avec les champs suivants.

```
{
"VaultARN":"arn:aws:glacier:region:111122223333:vaults/awsexamplevault",
"InventoryDate":"*** job completion date ***",
"ArchiveList":[
{"ArchiveId":"*** archiveid ***",
"ArchiveDescription":*** archive description (if set) ***,
"CreationDate":"*** archive creation date ***",
"Size":"*** archive size (in bytes) ***",
"SHA256TreeHash":"*** archive hash ***"
}
{"ArchiveId":
...
]}
```

5. Utilisez la commande `delete-archive` pour supprimer chaque archive d'un coffre jusqu'à ce qu'il n'en reste aucune.

```
aws glacier delete-archive --vault-name awsexamplevault --account-id 111122223333
 --archive-id "*** archiveid ***"
```

> **Note**
>
> Si votre identifiant d'archive commence par un tiret ou un autre caractère spécial, vous devez le placer entre guillemets pour exécuter cette commande.

6. Utilisez la commande `initiate-job` pour démarrer une nouvelle tâche d'extraction d'inventaire.

```
aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
job-parameters '{"Type": "inventory-retrieval"}'
```

7. Une fois l'opération terminée, utilisez la commande `delete-vault` pour supprimer un coffre sans archives.

```
aws glacier delete-vault --vault-name awsexamplevault --account-id 111122223333
```

Marquer vos coffres-forts Amazon Glacier

Vous pouvez attribuer vos propres métadonnées aux coffres-forts d'Amazon Glacier sous forme de balises. Une balise est une paire clé-valeur que vous définissez pour un coffre. Pour plus d'informations de base concernant le balisage, y compris les restrictions sur les balises, consultez la page Balisage des ressources Amazon Glacier.

Les rubriques suivantes décrivent la façon dont vous pouvez ajouter, répertorier et supprimer des balises pour les coffres.

Rubriques

- Marquage des coffres-forts à l'aide de la console Amazon Glacier
- Marquer les coffres-forts à l'aide du AWS CLI
- Marquage des coffres-forts à l'aide de l'API Amazon Glacier
- Sections connexes

Marquage des coffres-forts à l'aide de la console Amazon Glacier

Vous pouvez ajouter, répertorier et supprimer des balises à l'aide de la console Amazon Glacier, comme décrit dans les procédures suivantes.

Pour afficher les balises d'un coffre

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région.
3. Dans le panneau de navigation de gauche, choisissez Coffres.
4. Dans la liste des coffres, choisissez un coffre.
5. Choisissez l'onglet Propriétés du coffre. Accédez à la section Balises pour voir les balises associées au coffre.

Pour ajouter une balise à un coffre

Vous pouvez associer à un coffre jusqu'à 50 balises. Les balises associées à un coffre doivent disposer de clés de balise uniques.

Pour plus d'informations sur les restrictions relatives aux balises, consultez la section Balisage des ressources Amazon Glacier.

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région.
3. Dans le panneau de navigation de gauche, choisissez Coffres.
4. Dans la liste des coffres, choisissez le nom du coffre auquel vous voulez ajouter les balises.
5. Choisissez l'onglet Propriétés du coffre.
6. Dans la section Balises, choisissez Ajouter. La page Ajouter des balises s’affiche.
7. Sur la page Ajouter des balises, spécifiez la clé de balise dans le champ Clé, et indiquez éventuellement une valeur de balise dans le champ Valeur.
8. Sélectionnez Enregistrer les modifications.

Pour modifier une balise

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région.
3. Dans le panneau de navigation de gauche, choisissez Coffres.

4. Dans la liste des coffres, choisissez un nom de coffre.
5. Choisissez l'onglet Propriétés du coffre, puis faites défiler l'écran vers le bas jusqu'à la section Balises.
6. Sous Balises, cochez la case en regard des balises que vous voulez modifier, puis choisissez Modifier. La page Modifier les balises s'affiche.
7. Mettez à jour la clé de balise dans le champ Clé, et mettez éventuellement à jour la valeur de balise dans le champ Valeur.
8. Sélectionnez Enregistrer les modifications.

Pour supprimer une balise d'un coffre

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région.
3. Dans le panneau de navigation de gauche, choisissez Coffres.
4. Dans la liste des coffres, choisissez le nom du coffre dont vous voulez supprimer les balises.
5. Choisissez l'onglet Propriétés du coffre. Faites défiler l'écran vers le bas jusqu'à la section Balises.
6. Sous Balises, cochez la case en regard des balises que vous voulez supprimer, puis choisissez Supprimer.
7. La boîte de dialogue Supprimer des balises s'ouvre. Pour confirmer que vous voulez supprimer les balises sélectionnées, choisissez Supprimer.

Marquer les coffres-forts à l'aide du AWS CLI

Procédez comme suit pour ajouter, répertorier ou supprimer des balises à l'aide du AWS Command Line Interface (AWS CLI).

Chaque balise est constituée d'une clé et d'une valeur. Chaque coffre peut comporter jusqu'à 50 balises.

1. Pour ajouter des balises à un coffre, utilisez la commande `add-tags-to-vault`.

```
aws glacier add-tags-to-vault --vault-name examplevault --account-id 111122223333
 --tags id=1234,date=2020
```

Pour plus d'informations sur cette opération de coffre, consultez Ajout de balises à un coffre.

2. Pour afficher la liste de toutes les balises attachées à un coffre, utilisez la commande `list-tags-for-vault`.

```
aws glacier list-tags-for-vault --vault-name examplevault --account-id 111122223333
```

Pour plus d'informations sur cette opération de coffre, consultez Affichage de la liste des balises d'un coffre.

3. Pour supprimer une ou plusieurs balises de l'ensemble de balises attachées à un coffre, utilisez la commande `remove-tags-from-vault`.

```
aws glacier remove-tags-from-vault --vault-name examplevault --account-
id 111122223333 --tag-keys date
```

Pour plus d'informations sur cette opération de coffre, consultez Suppression de balises d'un coffre.

Marquage des coffres-forts à l'aide de l'API Amazon Glacier

Vous pouvez ajouter, répertorier et supprimer des balises à l'aide de l'API Amazon Glacier. Pour obtenir des exemples, consultez la documentation suivante :

Ajout de balises à un coffre (POST tags add)

Ajoute ou met à jour les balises pour le coffre spécifié.

Affichage des balises d'un coffre (GET tags)

Répertorie les balises pour le coffre spécifié.

Suppression des balises d'un coffre (POST tags remove)

Supprime les balises du coffre spécifié.

Sections connexes

- Balisage des ressources Amazon Glacier

Verrou Amazon Glacier Vault

Les rubriques suivantes décrivent comment verrouiller un coffre-fort dans Amazon Glacier et comment utiliser les politiques de verrouillage du coffre-fort.

Rubriques

- Présentation de la fonction de verrouillage de coffre
- Verrouiller un coffre-fort à l'aide de l'API Amazon Glacier
- Verrouiller un coffre-fort à l'aide du AWS Command Line Interface
- Verrouiller un coffre-fort à l'aide de la console Amazon Glacier

Présentation de la fonction de verrouillage de coffre

Amazon Glacier Vault Lock vous aide à déployer et à appliquer facilement des contrôles de conformité pour les différents coffres-forts Amazon Glacier grâce à une politique Vault Lock. Vous pouvez notamment spécifier des contrôles « à inscription unique et lecture multiple (WORM) » dans une politique de verrouillage de coffre pour empêcher qu'elle soit modifiée par la suite.

> **Important**
>
> Une fois verrouillée, une politique de verrouillage de coffre ne peut plus être modifiée ni supprimée.

Amazon Glacier applique les contrôles définis dans la politique Vault Lock pour vous aider à atteindre vos objectifs de conformité. Par exemple, vous pouvez utiliser des politiques de verrouillage de coffre pour mettre en œuvre la conservation des données. Vous pouvez déployer divers contrôles de conformité dans une politique Vault Lock en utilisant le langage de politique Gestion des identités et des accès AWS (IAM). Pour plus d'informations sur les stratégies de verrouillage de coffre, consultez Stratégies de verrouillage de coffre.

Les politiques de verrouillage de coffre sont à distinguer des stratégies d'accès aux coffres. Ces deux stratégies régissent les contrôles d'accès à votre coffre. Cependant, une politique de verrouillage de coffre peut être verrouillée pour empêcher toute modification ultérieure, ce qui assure une mise en œuvre rigoureuse de vos contrôles de conformité. Vous pouvez utiliser la stratégie de verrouillage de coffre pour déployer des contrôles réglementaires et de conformité, qui imposent généralement des contrôles stricts sur le plan de l'accès aux données.

> **Important**
>
> Nous vous recommandons de créer d'abord un coffre, de définir une politique de verrouillage de coffre, puis de charger vos archives sur le coffre de sorte que la politique s'applique à celles-ci.

En revanche, vous utilisez la stratégies d'accès au coffre pour mettre en œuvre des contrôles d'accès qui ne sont pas liés à la conformité, provisoires ou soumis à des modifications fréquentes. Vous pouvez utiliser à la fois une politique de verrouillage de coffre et une stratégie d'accès au coffre. Par exemple, vous pouvez mettre en place des règles de conservation des données basées sur la durée dans la politique de verrouillage de coffre (refuser les suppressions) et accorder un accès en lecture à des tiers désignés ou à vos partenaires commerciaux (autoriser les lectures) dans votre stratégie d'accès au coffre.

Le verrouillage d'un coffre se déroule en deux étapes :

1. Initiez le verrouillage en attachant une politique de verrouillage de coffre à votre coffre, ce qui a pour effet d'attribuer au verrouillage l'état En cours et de renvoyer un ID de verrouillage. Lorsque la politique est à l'état En cours, vous disposez de 24 heures pour valider votre politique de verrouillage de coffre. Au-delà, l'ID de verrouillage expire. Pour empêcher que le coffre quitte l'état En cours, vous devez finaliser le processus de verrouillage de coffre dans les 24 heures. À défaut, la politique de verrouillage de coffre sera supprimée.
2. Utilisez l'ID de verrouillage pour terminer le verrouillage. Si la politique de verrouillage de coffre ne fonctionne pas comme prévu, vous pouvez arrêter le processus de verrouillage de coffre et recommencer depuis le début. Pour plus d'informations sur l'utilisation de l'API Amazon Glacier pour verrouiller un coffre-fort, consultezVerrouiller un coffre-fort à l'aide de l'API Amazon Glacier.

Verrouiller un coffre-fort à l'aide de l'API Amazon Glacier

Pour verrouiller votre coffre-fort à l'aide de l'API Amazon Glacier, vous devez Lancement de verrouillage de coffre (POST lock-policy) d'abord utiliser une politique de verrouillage du coffre-fort qui spécifie les contrôles que vous souhaitez déployer. L'opération `Initiate Vault Lock` attache la politique au coffre, fait passer le verrouillage de coffre à l'état En cours et renvoie un ID de verrouillage unique. Une fois le verrouillage de coffre à l'état En cours, vous disposez de 24 heures pour finaliser le verrouillage en appelant Achèvement du verrouillage du coffre (POST lockId) avec l'ID de verrouillage renvoyé par l'appel `Initiate Vault Lock`.

> **Important**
>
> - Nous vous recommandons de créer d'abord un coffre, de définir une politique de verrouillage de coffre, puis de charger vos archives sur le coffre de sorte que la politique s'applique à celles-ci.
> - Une fois que la politique de verrouillage de coffre est verrouillée, elle ne peut plus être modifiée ni supprimée.

Si vous ne finalisez pas le processus de verrouillage de coffre dans les 24 heures après le passage à l'état En cours, votre coffre quitte automatiquement l'état En cours et la politique de verrouillage de coffre est supprimée. Vous pouvez appeler à nouveau `Initiate Vault Lock` pour installer une nouvelle politique de verrouillage de coffre et passer l'état à En cours.

L'état En cours vous donne l'occasion de tester une politique de verrouillage de coffre avant de la verrouiller. Une politique de verrouillage de coffre à l'état En cours est pleinement opérationnelle, autant que si le coffre était verrouillé, sauf que vous pouvez encore supprimer la politique en appelant Interruption du verrouillage du coffre (DELETE lock-policy). Pour peaufiner votre politique, vous pouvez répéter la combinaison `Abort Vault Lock`/`Initiate Vault Lock` autant de fois que nécessaire pour valider les modifications que vous apportez à la politique de verrouillage de coffre.

Après avoir validé la politique de verrouillage de coffre, vous pouvez appeler Achèvement du verrouillage du coffre (POST lockId) avec l'ID de verrouillage le plus récent pour finaliser le processus de verrouillage de coffre. Le coffre passe à l'état Verrouillé et la politique de verrouillage de coffre est immuable et ne peut plus être supprimée en appelant `Abort Vault Lock`.

Sections connexes

- Stratégies de verrouillage de coffre
- Interruption du verrouillage du coffre (DELETE lock-policy)
- Achèvement du verrouillage du coffre (POST lockId)
- Obtenir le verrouillage de coffre (GET lock-policy)
- Lancement de verrouillage de coffre (POST lock-policy)

Verrouiller un coffre-fort à l'aide du AWS Command Line Interface

Vous pouvez verrouiller votre coffre-fort à l'aide du AWS Command Line Interface. Une politique de verrouillage de coffre est alors installée sur le coffre spécifié et l'ID de verrouillage renvoyé. Vous devez finaliser le processus de verrouillage de coffre dans les 24 heures. Au-delà, la politique de verrouillage de coffre est supprimée du coffre.

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

1. Utilisez la commande `initiate-vault-lock` pour installer une politique de verrouillage de coffre et définir l'état de verrouillage du coffre sur `InProgress`.

   ```
   aws glacier initiate-vault-lock --vault-name examplevault --account-id 111122223333
    --policy file://lockconfig.json
   ```

2. La configuration du verrouillage est un document JSON comme celui présenté dans l'exemple suivant. Avant d'utiliser cette commande, remplacez le *VAULT_ARN* et *Principal* par les valeurs appropriées à votre cas d'utilisation.

   Pour trouver l'ARN du coffre que vous souhaitez verrouiller, vous pouvez utiliser la commande `list-vaults`.

   ```
   {"Policy":"{\"Version\":\"2012-10-17\",       \"Statement\":[{\"Sid
   \":\"Define-vault-lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":
   \"arn:aws:iam::111122223333:root\"},\"Action\":\"glacier:DeleteArchive
   \",\"Resource\":\"VAULT_ARN\",\"Condition\":{\"NumericLessThanEquals\":
   {\"glacier:ArchiveAgeinDays\":\"365\"}}}]}"}
   ```

3. Après avoir initié le verrouillage du coffre, le `lockId` vous est renvoyé.

   ```
   {
       "lockId": "LOCK_ID"
   }
   ```

Pour finaliser le verrouillage du coffre, vous devez exécuter `complete-vault-lock` dans les 24 heures. À défaut, la politique de verrouillage de coffre sera supprimée du coffre.

```
aws glacier complete-vault-lock --vault-name examplevault --account-id 111122223333 --
lock-id LOCK_ID
```

Sections connexes

- initiate-vault-lock dans la référence de commande de l'AWS CLI
- list-vaults dans le guide de référence des commandes AWS CLI
- complete-vault-lock dans la référence de commande de l'AWS CLI
- Stratégies de verrouillage de coffre
- Interruption du verrouillage du coffre (DELETE lock-policy)
- Achèvement du verrouillage du coffre (POST lockId)
- Obtenir le verrouillage de coffre (GET lock-policy)
- Lancement de verrouillage de coffre (POST lock-policy)

Verrouiller un coffre-fort à l'aide de la console Amazon Glacier

Amazon Glacier Vault Lock vous aide à déployer et à appliquer facilement des contrôles de conformité pour les différents coffres-forts Amazon Glacier grâce à une politique Vault Lock. Pour plus d'informations sur Amazon Glacier Vault Lock, consultez Amazon Glacier Access Control with Vault Lock Policies.

> **Important**
>
> - Nous vous recommandons de créer d'abord un coffre, de définir une politique de verrouillage de coffre, puis de charger vos archives sur le coffre de sorte que la politique s'applique à celles-ci.
> - Une fois que la politique de verrouillage de coffre est verrouillée, elle ne peut plus être modifiée ni supprimée.

Pour lancer une politique de verrouillage du coffre-fort sur votre coffre-fort à l'aide de la console Amazon Glacier

Vous initiez le verrouillage en attachant une politique de verrouillage de coffre à votre coffre, ce qui a pour effet d'attribuer au verrouillage l'état En cours et de renvoyer un ID de verrouillage. Lorsque la politique est à l'état En cours, vous disposez de 24 heures pour valider votre politique de verrouillage de coffre. Au-delà, l'ID de verrouillage expire.

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, sélectionnez-en une dans le sélecteur Région AWS de région.
3. Dans le panneau de navigation de gauche, choisissez Coffres.
4. Sur la page Coffres, choisissez Créer un coffre.
5. Créez un coffre.

> **Important**
>
> Nous vous recommandons de créer d'abord un coffre, de définir une politique de verrouillage de coffre, puis de charger vos archives sur le coffre de sorte que la politique s'applique à celles-ci.

6. Choisissez votre nouveau coffre dans la liste des coffres.
7. Choisissez l'onglet Politiques de coffre.
8. Dans la section Politique de verrouillage de coffre, choisissez Lancer une politique de verrouillage de coffre.
9. Sur la page Lancer une politique de verrouillage de coffre, spécifiez les contrôles de conservation des enregistrements dans votre politique de verrouillage de coffre au format texte dans la zone de texte standard.

> **Note**
>
> Vous pouvez spécifier les contrôles de conservation des enregistrements dans une politique Vault Lock au format texte et lancer le Vault Lock en appelant l'opération `Initiate Vault Lock` API ou via l'interface utilisateur interactive de la console Amazon Glacier. Pour plus d'informations sur le formatage de votre politique Vault Lock, consultez les exemples de politique de verrouillage d'Amazon Glacier Vault.

10. Sélectionnez Enregistrer les modifications.
11. Dans la boîte de dialogue Enregistrer l'ID de verrouillage du coffre, copiez votre ID de verrouillage et enregistrez-le dans un emplacement sécurisé.

> **Important**
>
> Une fois que la politique de verrouillage de coffre est lancée, vous avez 24 heures pour la valider et finaliser le processus de verrouillage. Pour finaliser le processus de verrouillage, vous devez fournir l'ID de verrouillage. Si vous ne le faites pas dans les 24 heures, l'ID de verrouillage expire et votre politique en cours est supprimée.

12. Après avoir enregistré votre ID de verrouillage dans un emplacement sécurisé, choisissez Fermer.
13. Testez votre politique de verrouillage de coffre dans les 24 heures. Si la politique fonctionne comme prévu, choisissez Effectuer une politique de verrouillage de coffre.
14. Dans la boîte de dialogue Terminer le verrouillage du coffre, cochez la case pour prendre acte du caractère irréversible de la finalisation du processus de politique de verrouillage de coffre.
15. Saisissez l'ID de verrouillage qui vous a été fourni dans la zone de texte.
16. Choisissez Terminer le verrouillage du coffre.

Utilisation des archives dans Amazon Glacier

Une archive est un objet, comme une photo, une vidéo ou un document, que vous stockez dans un coffre. Il s'agit d'une unité de stockage de base dans Amazon Glacier (Amazon Glacier). Chaque archive est associée à un ID unique et une description en option. Lorsque vous chargez une archive, Amazon Glacier renvoie une réponse qui inclut un ID d'archive. Cet identifiant d'archive est unique dans la AWS région dans laquelle l'archive est stockée. L'ID suivant est un exemple d'ID d'archive.

```
TJgHcrOSfAkV6hdPqOATYfp_0ZaxL1pIBOc02iZ0gDPMr2ig-
nhwd_PafstsdIf6HSrjHnP-3p6LCJClYytFT_CBhT9CwNxbRaM5MetS3I-
GqwxI3Y8QtgbJbhEQPs0mJ3KExample
```

IDs Les archives ont une longueur de 138 octets. Lorsque vous chargez une archive, vous pouvez indiquer une description facultative. Avec l'ID de l'archive vous pouvez extraire l'archive mais pas sa description.

> **Important**
>
> Amazon Glacier fournit une console de gestion. Vous pouvez utiliser la console pour créer et supprimer des coffres. Cependant, toutes les autres interactions avec Amazon Glacier nécessitent que vous utilisiez la AWS Command Line Interface (CLI) ou que vous écriviez du code. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez utiliser le code AWS CLI ou écrire du code pour effectuer des demandes, en utilisant directement l'API REST ou en utilisant Amazon SDKs. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la section AWS CLI Référence d'Amazon Glacier. Pour installer le AWS CLI, rendez-vous sur AWS Command Line Interface.

Rubriques

- Opérations d'archivage dans Amazon Glacier
- Conservation des métadonnées d'archive côté client
- Téléchargement d'une archive dans Amazon Glacier
- Téléchargement d'une archive dans Amazon Glacier
- Supprimer une archive dans Amazon Glacier

Opérations d'archivage dans Amazon Glacier

Amazon Glacier prend en charge les opérations d'archivage de base suivantes : chargement, téléchargement et suppression. Le téléchargement d'une archive est une opération asynchrone.

Téléchargement d'une archive dans Amazon Glacier

Vous pouvez charger une archive en une seule opération ou en plusieurs parties. L'appel d'API qui vous permet de charger une archive en plusieurs parties s'appelle le chargement partitionné. Pour de plus amples informations, veuillez consulter Téléchargement d'une archive dans Amazon Glacier.

> **Important**
>
> Amazon Glacier fournit une console de gestion. Vous pouvez utiliser la console pour créer et supprimer des coffres. Cependant, toutes les autres interactions avec Amazon Glacier nécessitent que vous utilisiez la AWS Command Line Interface (CLI) ou que vous écriviez du code. Par exemple, pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez utiliser le code AWS CLI ou écrire du code pour effectuer des demandes, en utilisant directement l'API REST ou en utilisant Amazon SDKs. Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la section AWS CLI Référence d'Amazon Glacier. Pour installer le AWS CLI, rendez-vous sur AWS Command Line Interface.

Trouver un identifiant d'archive dans Amazon Glacier

Vous pouvez obtenir l'ID de l'archive en téléchargeant l'inventaire du coffre qui contient l'archive. Pour plus d'informations sur le téléchargement de l'inventaire de coffre, consultez la page Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

Téléchargement d'une archive dans Amazon Glacier

Le téléchargement d'une archive est une opération asynchrone. Vous devez commencer par lancer une tâche de téléchargement d'une archive spécifique. Après avoir reçu la demande de travail, Amazon Glacier prépare votre archive pour le téléchargement. Une fois la tâche terminée, vous pouvez télécharger vos données archivées. En raison de la nature asynchrone de la tâche, vous pouvez demander à Amazon Glacier d'envoyer une notification à une rubrique Amazon

Simple Notification Service (Amazon SNS) une fois la tâche terminée. Vous pouvez spécifier une rubrique SNS pour chaque demande de tâche individuelle ou configurer votre coffre pour envoyer une notification lorsque des événements spécifiques se produisent. Pour en savoir plus sur le téléchargement d'une archive, consultez Téléchargement d'une archive dans Amazon Glacier.

Supprimer une archive dans Amazon Glacier

Amazon Glacier fournit un appel d'API que vous pouvez utiliser pour supprimer une archive à la fois. Pour de plus amples informations, veuillez consulter Supprimer une archive dans Amazon Glacier.

Mettre à jour une archive dans Amazon Glacier

Une fois que vous avez chargé une archive, vous ne pouvez pas mettre à jour son contenu ou sa description. Le seul moyen dont vous disposez pour mettre à jour le contenu de l'archive ou sa description est de supprimer l'archive et de charger une autre archive. Notez que chaque fois que vous chargez une archive, Amazon Glacier vous renvoie un identifiant d'archive unique.

Conservation des métadonnées d'archive côté client

À l'exception de la description facultative de l'archive, Amazon Glacier ne prend en charge aucune métadonnée supplémentaire pour les archives. Lorsque vous chargez une archive, Amazon Glacier attribue un identifiant, une séquence opaque de caractères, à partir de laquelle vous ne pouvez pas déduire le sens de l'archive. Vous pouvez conserver les métadonnées relatives aux archives côté client. Les métadonnées peuvent inclure le nom de l'archive et d'autres informations utiles sur l'archive.

> **Note**
>
> Si vous êtes un client Amazon Simple Storage Service (Amazon S3), vous savez que lorsque vous chargez un objet dans un compartiment, vous pouvez lui attribuer une clé d'objet, par exemple `MyDocument.txt` ou `SomePhoto.jpg`. Dans Amazon Glacier, vous ne pouvez pas attribuer de clé d'objet aux archives que vous chargez.

Si vous gérez des métadonnées d'archives côté client, notez qu'Amazon Glacier gère un inventaire des coffres-forts qui inclut les archives IDs et toutes les descriptions que vous avez fournies lors du chargement de l'archive. Vous pouvez parfois télécharger l'inventaire de coffre pour régler les problèmes de la base de données côté client que vous tenez à jour pour les métadonnées d'archive.

Cependant, Amazon Glacier fait l'inventaire des coffres-forts environ tous les jours. Lorsque vous demandez un inventaire de coffre, Amazon Glacier renvoie le dernier inventaire qu'il a préparé, un instantané à un moment donné.

Téléchargement d'une archive dans Amazon Glacier

Amazon Glacier (Amazon Glacier) fournit une console de gestion que vous pouvez utiliser pour créer et supprimer des coffres-forts. Toutefois, vous ne pouvez pas télécharger d'archives sur Amazon Glacier à l'aide de la console de gestion. Pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez utiliser le code AWS CLI ou écrire du code pour effectuer des demandes, en utilisant directement l'API REST ou en utilisant Amazon SDKs.

Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la AWS CLI section Référence d'Amazon Glacier. Pour installer le AWS CLI, rendez-vous sur AWS Command Line Interface. Les rubriques suivantes relatives au chargement décrivent comment charger des archives sur Amazon Glacier à l'aide du SDK Amazon pour Java, du SDK Amazon pour .NET et de l'API REST.

Rubriques

- Options de téléchargement d'une archive sur Amazon Glacier
- Chargement d'une archive en une seule opération
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)

Options de téléchargement d'une archive sur Amazon Glacier

En fonction de la taille des données que vous chargez, Amazon Glacier propose les options suivantes :

- Charger des archives en une seule opération : vous pouvez charger des archives de 1 octet à 4 Go en une seule opération. Toutefois, nous encourageons les clients d'Amazon Glacier à utiliser le téléchargement partitionné pour charger des archives de plus de 100 Mo. Pour de plus amples informations, veuillez consulter Chargement d'une archive en une seule opération.
- Charger des archives en plusieurs parties : l'API de chargement partitionné vous permet de charger des archives volumineuses, d'une taille maximale d'environ 40 000 Go (10 000 * 4 Go).

  L'appel d'API de chargement partitionné est conçu pour améliorer l'expérience de chargement pour les archives plus volumineuses. Vous pouvez charger des archives en plusieurs parties. Ces

parties peuvent être chargées indépendamment, dans n'importe quel ordre, et en parallèle. En cas d'échec du chargement d'une partie, vous ne devez recharger que cette partie, et non la totalité de l'archive. Vous pouvez utiliser le chargement partitionné pour les archives dont la taille est comprise entre 1 octet et environ 40 000 Go. Pour de plus amples informations, veuillez consulter Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

> **Important**
>
> L'inventaire du coffre Amazon Glacier n'est mis à jour qu'une fois par jour. Lorsque vous chargez une archive, elle est ajoutée à votre coffre, mais elle n'est pas visible (dans la console ou dans votre liste d'inventaire de coffre téléchargé) tant que l'inventaire de coffre n'est pas mis à jour.

Utilisation du AWS Snowball Edge service

AWS Snowball Edge accélère le transfert de grandes quantités de données vers et depuis des appareils appartenant à Amazon, AWS en contournant Internet. Pour plus d'informations, consultez la page de détails de AWS Snowball Edge.

Pour télécharger des données existantes vers Amazon Glacier (Amazon Glacier), vous pouvez envisager d'utiliser l'un des types d' AWS Snowball Edge appareils pour importer des données dans Amazon S3, puis de les déplacer vers la classe de stockage Amazon Glacier à des fins d'archivage selon les règles du cycle de vie. Lorsque vous transférez des objets Amazon S3 vers la classe de stockage Amazon Glacier, Amazon S3 utilise Amazon Glacier en interne pour un stockage durable à moindre coût. Bien que les objets soient stockés dans Amazon Glacier, ils restent des objets Amazon S3 que vous gérez dans Amazon S3, et vous ne pouvez pas y accéder directement via Amazon Glacier.

Pour plus d'informations sur la configuration du cycle de vie d'Amazon S3 et la transition des objets vers la classe de stockage Amazon Glacier, consultez la section Object Lifecycle Management and Transitioning Objects dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Chargement d'une archive en une seule opération

Comme il est décrit dans Téléchargement d'une archive dans Amazon Glacier, vous pouvez charger de petites archives en une seule opération. Toutefois, nous encourageons les clients d'Amazon

Glacier (Amazon Glacier) à utiliser le téléchargement partitionné pour charger des archives de plus de 100 Mo.

Rubriques

- Chargement d'une archive en une seule opération à l'aide du AWS Command Line Interface
- Chargement d'une archive en une seule opération à l'aide du AWS SDK pour Java
- Téléchargement d'une archive en une seule opération à l'aide du AWS SDK pour .NET dans Amazon Glacier
- Chargement d'une archive en une seule opération à l'aide de l'API REST

Chargement d'une archive en une seule opération à l'aide du AWS Command Line Interface

Vous pouvez charger une archive dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : Chargez une archive à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l’interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l’utilisateur de l’interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

   ```
   aws help
   ```

- Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

```
aws glacier list-vaults --account-id 123456789012
```

- Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

```
aws configure list
```

Exemple : Chargez une archive à l'aide du AWS CLI

Pour charger une archive, vous devez avoir créé un coffre. Pour plus d'informations sur la création de coffres, consultez Création d'un coffre-fort dans Amazon Glacier.

1. Utilisez la commande `upload-archive` pour ajouter une archive à un coffre existant. Dans l'exemple ci-dessous, remplacez les valeurs des paramètres `vault name` et `account ID`. Pour le paramètre body, indiquez le chemin du fichier que vous souhaitez charger.

```
aws glacier upload-archive --vault-name awsexamplevault --account-id 123456789012
 --body archive.zip
```

2. Sortie attendue :

```
{
    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
    "checksum": "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/123456789012/vaults/awsexamplevault/archives/
kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
}
```

   Une fois terminée, la commande affiche l'ID de l'archive, la somme de contrôle et l'emplacement dans Amazon Glacier. Pour plus d'informations sur la commande upload-archive, consultez upload-archive dans le guide de référence des commandes AWS CLI .

Chargement d'une archive en une seule opération à l'aide du AWS SDK pour Java

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour Java fournissent une méthode pour télécharger une archive.

Rubriques

- Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java
- Chargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour Java

Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java

La classe `ArchiveTransferManager` de l'API de haut niveau fournit la méthode `upload`, qui permet de charger une archive dans un coffre.

> **Note**
>
> Vous pouvez utiliser la méthode `upload` pour charger des archives petites ou volumineuses. Selon la taille de l'archive que vous chargez, cette méthode détermine si vous devez charger l'archive en une seule opération ou si vous devez utiliser l'API de chargement partitionné pour charger l'archive en plusieurs parties.

Exemple : téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java

L'exemple de code Java suivant charge une archive sur un coffre (`examplevault`) dans la région USA Ouest (Oregon) (`us-west-2`). Pour obtenir la liste des AWS régions et des points de terminaison pris en charge, consultezAccès à Amazon Glacier.

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du coffre sur lequel vous souhaitez effectuer le chargement et le nom du fichier à charger.

Example

```
import java.io.File;
import java.io.IOException;
import java.util.Date;
```

```
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.transfer.ArchiveTransferManager;
import com.amazonaws.services.glacier.transfer.UploadResult;


public class ArchiveUploadHighLevel {
    public static String vaultName = "*** provide vault name ***";
    public static String archiveToUpload = "*** provide name of file to upload ***";

    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {


     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-west-2.amazonaws.com/");

        try {
            ArchiveTransferManager atm = new ArchiveTransferManager(client,
 credentials);

            UploadResult result = atm.upload(vaultName, "my archive " + (new Date()),
 new File(archiveToUpload));
            System.out.println("Archive ID: " + result.getArchiveId());

        } catch (Exception e)
        {
            System.err.println(e);
        }
    }
}
```

Chargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour Java

L'API de bas niveau fournit des méthodes pour toutes les opérations d'archivage. Voici les étapes de chargement d'une archive à l'aide du kit AWS SDK pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

Vous devez spécifier AWS la région dans laquelle vous souhaitez télécharger l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Fournissez les informations de demande en créant une instance de la classe `UploadArchiveRequest`.

   Outre les données que vous souhaitez télécharger, vous devez fournir un total de contrôle (hachage d'arborescence SHA-256) de la charge utile, le nom du coffre, la longueur du contenu des données et votre ID de compte.

   Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.

3. Exécutez la méthode `uploadArchive` en fournissant l'objet de demande comme paramètre.

   En réponse, Amazon Glacier (Amazon Glacier) renvoie l'ID d'archive de l'archive récemment téléchargée.

L'extrait de code Java suivant illustre les étapes précédentes.

```
AmazonGlacierClient client;

UploadArchiveRequest request = new UploadArchiveRequest()
    .withVaultName("*** provide vault name ***")
    .withChecksum(checksum)
    .withBody(new ByteArrayInputStream(body))
    .withContentLength((long)body.length);

UploadArchiveResult uploadArchiveResult = client.uploadArchive(request);

System.out.println("Location (includes ArchiveID): " +
 uploadArchiveResult.getLocation());
```

Exemple : téléchargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour Java

L'exemple de code Java suivant utilise le AWS SDK pour Java pour télécharger une archive dans un coffre (`examplevault`). Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez

mettre à jour le code tel qu'il est présenté avec le nom du coffre sur lequel vous souhaitez effectuer le chargement et le nom du fichier à charger.

```
import java.io.ByteArrayInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.TreeHashGenerator;
import com.amazonaws.services.glacier.model.UploadArchiveRequest;
import com.amazonaws.services.glacier.model.UploadArchiveResult;
public class ArchiveUploadLowLevel {

    public static String vaultName = "*** provide vault name ****";
    public static String archiveFilePath = "*** provide to file upload ****";
    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-east-1.amazonaws.com/");

        try {
            // First open file and read.
            File file = new File(archiveFilePath);
            InputStream is = new FileInputStream(file);
            byte[] body = new byte[(int) file.length()];
            is.read(body);

            // Send request.
            UploadArchiveRequest request = new UploadArchiveRequest()
                .withVaultName(vaultName)
                .withChecksum(TreeHashGenerator.calculateTreeHash(new
 File(archiveFilePath)))
                .withBody(new ByteArrayInputStream(body))
                .withContentLength((long)body.length);

            UploadArchiveResult uploadArchiveResult = client.uploadArchive(request);
```

```
            System.out.println("ArchiveID: " + uploadArchiveResult.getArchiveId());

        } catch (Exception e)
        {
            System.err.println("Archive not uploaded.");
            System.err.println(e);
        }
    }
}
```

Téléchargement d'une archive en une seule opération à l'aide du AWS SDK pour .NET dans Amazon Glacier

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode permettant de télécharger une archive en une seule opération.

Rubriques

- Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Chargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour .NET

Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

La classe `ArchiveTransferManager` de l'API de haut niveau fournit la méthode `Upload` qui permet de charger une archive dans un coffre.

> **Note**
>
> Vous pouvez utiliser la méthode `Upload` pour charger des fichiers petits ou volumineux. Selon la taille du fichier que vous chargez, cette méthode détermine si vous devez effectuer cette opération en une seule fois ou si vous devez utiliser l'API de chargement partitionné pour charger le fichier en plusieurs parties.

Exemple : téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

L'exemple de code C# suivant charge une archive sur un coffre (`examplevault`) dans la région USA Ouest (Oregon).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du fichier que vous souhaitez charger.

Example

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveUploadHighLevel
  {
    static string vaultName = "examplevault";
    static string archiveToUpload = "*** Provide file name (with full path) to upload
 ***";

    public static void Main(string[] args)
    {
       try
      {
         var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
         // Upload an archive.
         string archiveId = manager.Upload(vaultName, "upload archive test",
 archiveToUpload).ArchiveId;
         Console.WriteLine("Archive ID: (Copy and save this ID for use in other
 examples.) : {0}", archiveId);
         Console.WriteLine("To continue, press Enter");
         Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }
  }
}
```

Chargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'API de bas niveau fournit des méthodes pour toutes les opérations d'archivage. Voici les étapes de chargement d'une archive à l'aide du kit AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle vous souhaitez télécharger l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Fournissez les informations de demande en créant une instance de la classe `UploadArchiveRequest`.

   Outre les données que vous souhaitez télécharger, vous devrez fournir un total de contrôle (hachage d'arborescence SHA-256) de la charge utile, le nom du coffre et votre ID de compte.

   Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour .NET avec Amazon Glacier.
3. Exécutez la méthode `UploadArchive` en fournissant l'objet de demande comme paramètre.

   En réponse, Amazon Glacier renvoie l'ID d'archive de l'archive récemment téléchargée.

Exemple : téléchargement d'une archive en une seule opération à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'exemple de code C# suivant présente les étapes précédentes. L'exemple utilise le AWS SDK pour .NET pour télécharger une archive dans un coffre (`examplevault`).

> **Note**
>
> Pour plus d'informations sur l'API REST sous-jacente pour charger une archive en une seule demande, consultez Chargement d'archive (POST archive).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du fichier que vous souhaitez charger.

Example

```
using System;
using System.IO;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveUploadSingleOpLowLevel
  {
    static string vaultName       = "examplevault";
    static string archiveToUpload = "*** Provide file name (with full path) to upload
 ***";

    public static void Main(string[] args)
    {
      AmazonGlacierClient client;
      try
      {
         using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Uploading an archive.");
          string archiveId = UploadAnArchive(client);
          Console.WriteLine("Archive ID: {0}", archiveId);
        }
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }

    static string UploadAnArchive(AmazonGlacierClient client)
    {
      using (FileStream fileStream = new FileStream(archiveToUpload, FileMode.Open,
 FileAccess.Read))
      {
        string treeHash = TreeHashGenerator.CalculateTreeHash(fileStream);
        UploadArchiveRequest request = new UploadArchiveRequest()
        {
          VaultName = vaultName,
```

```
            Body = fileStream,
            Checksum = treeHash
          };
          UploadArchiveResponse response = client.UploadArchive(request);
          string archiveID = response.ArchiveId;
          return archiveID;
        }
      }
    }
}
```

Chargement d'une archive en une seule opération à l'aide de l'API REST

Vous pouvez utiliser l'appel de l'API de chargement d'archive pour charger une archive en une seule opération. Pour de plus amples informations, veuillez consulter Chargement d'archive (POST archive).

Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)

Rubriques

- Processus de chargement partitionné
- En bref
- Téléchargement d'archives volumineuses à l'aide du AWS CLI
- Chargement d'archives volumineuses en plusieurs parties à l'aide du kit SDK Amazon pour Java
- Téléchargement d'archives volumineuses à l'aide du AWS SDK pour .NET
- Chargement d'archives volumineuses en plusieurs parties à l'aide de l'API REST

Processus de chargement partitionné

Comme décrit dansTéléchargement d'une archive dans Amazon Glacier, nous encourageons les clients d'Amazon Glacier (Amazon Glacier) à utiliser le téléchargement partitionné pour charger des archives de plus de 100 mégaoctets (MiB).

1. Lancement du chargement partitionné

   Lorsque vous envoyez une demande pour lancer un téléchargement partitionné, Amazon Glacier renvoie un ID de téléchargement partitionné, qui est un identifiant unique pour votre

téléchargement partitionné. Cet ID sera requis pour toute opération de chargement partitionné suivante. Cet identifiant n'expire pas pendant au moins 24 heures après qu'Amazon Glacier ait terminé la tâche.

Dans votre demande de lancement d'un chargement partitionné, vous devez spécifier la taille des parties en nombre d'octets. Chacune des parties que vous chargez, à l'exception de la dernière partie, doit correspondre à cette taille.

> **Note**
>
> Vous n'avez pas besoin de connaître la taille globale de l'archive lorsque vous utilisez le chargement partitionné. Cela signifie que vous pouvez utiliser des téléchargements partiels dans les cas où vous ne connaissez pas la taille de l'archive lorsque vous commencez à charger l'archive. Vous devez juste définir la taille des parties au moment du lancement d'un chargement partitionné.

Dans la demande de lancement de chargement partitionné, vous pouvez également fournir une description facultative de l'archive.

2. Chargement des parties

   Pour chaque demande de chargement de partie, vous devez inclure l'ID de chargement partitionné que vous avez obtenu à l'étape 1. Dans la demande, vous devez également spécifier la plage de contenu, en octets, en identifiant la position de la partie dans l'archive finale. Amazon Glacier utilise ensuite les informations relatives à la plage de contenus pour assembler l'archive dans le bon ordre. Etant donné que vous indiquez la plage de contenu de chaque partie que vous chargez, il détermine la position de chaque partie dans l'assemblage final de l'archive ; par conséquent, vous pouvez charger les parties dans n'importe quel ordre. Vous pouvez également charger différentes parties en parallèle. Si vous chargez une nouvelle partie en utilisant la même plage de contenu que celle définie pour une partie précédemment chargée, cette dernière est remplacée.

3. Achèvement (ou arrêt) du chargement partitionné

   Après avoir chargé toutes les parties d'une archive, utilisez l'opération d'achèvement. Là encore, vous devez spécifier l'ID de chargement dans votre demande. Amazon Glacier crée une archive en concaténant les parties par ordre croissant en fonction de la plage de contenu que vous avez fournie. La réponse d'Amazon Glacier à une demande de téléchargement partitionné complet

inclut un ID d'archive pour l'archive nouvellement créée. Si vous avez fourni une description d'archive facultative dans la demande Initiate Multipart Upload, Amazon Glacier l'associe à l'archive assemblée. Lorsque votre opération d'achèvement de chargement partitionné aboutit, vous ne pouvez pas faire référence à l'ID de chargement partitionné. Cela signifie que vous ne pouvez pas accéder aux parties associées à l'ID de chargement partitionné.

Si vous arrêtez un chargement partitionné, vous ne pouvez pas charger de parties supplémentaires en utilisant cet ID de chargement partitionné. L'ensemble du stockage consommé par les parties associées au chargement partitionné arrêté est libéré. Si des parties étaient en cours de chargement, ces opérations peuvent encore réussir ou échouer même après un arrêt.

Opérations supplémentaires de chargement partitionné

Amazon Glacier (Amazon Glacier) fournit les appels d'API de téléchargement partitionné supplémentaires suivants.

- Affichage des parties : cette opération vous permet d'afficher la liste des parties d'un chargement partitionné spécifique. Elle renvoie des informations sur les parties que vous chargées dans le cadre d'un chargement partitionné. Pour chaque demande de pièces de liste, Amazon Glacier renvoie des informations portant sur un maximum de 1 000 pièces. S'il existe d'autres parties à afficher pour le chargement partitionné, le résultat est paginé et la réponse renvoyée inclut un marqueur à partir duquel poursuivre l'affichage de la liste. Vous devez envoyer des demandes supplémentaires pour extraire les parties suivantes. Notez que la liste des parties renvoyée n'inclut pas les parties dont le chargement n'est pas terminé.
- Affichage des chargements partitionnés : cette opération vous permet d'obtenir la liste des chargements partitionnés en cours. Un chargement partitionné en cours est un chargement que vous avez lancé, mais que vous n’avez pas encore terminé ou arrêté. Pour chaque demande d'affichage des chargements partitionnés, Amazon Glacier renvoie jusqu'à 1 000 chargements partitionnés. S'il existe d'autres chargements partitionnés à afficher, le résultat est paginé et la réponse renvoyée inclut un marqueur à partir duquel poursuivre l'affichage de la liste. Vous devez envoyer des demandes supplémentaires pour extraire les chargements partitionnés restants.

En bref

Le tableau suivant fournit les principales spécifications du chargement partitionné.

Élément	Spécification
Taille maximale d'archive	10 000 x 4 gibioctets (Gio)
Nombre maximum de parties par chargement	10 000
Taille de partie	De 1 Mio à 4 Gio ; la dernière partie peut être inférieure à 1 Mio. Vous spécifiez la valeur de taille en octets. La taille de la partie doit être un mébioctet (1 024 kibioctets [Kio]) multiplié par une puissance de 2. Par exemple, `1048576` (1 Mio), `2097152` (2 Mio), `4194304` (4 Mio), `8388608` (8 Mio).
Nombre maximum de parties renvoyées pour une demande de liste des parties	1 000
Nombre maximum de chargements partitionnés renvoyés dans le cadre d'une demande d'affichage de chargements partitionnés	1 000

Téléchargement d'archives volumineuses à l'aide du AWS CLI

Vous pouvez charger une archive dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI). Pour améliorer l'expérience de chargement des archives volumineuses, Amazon Glacier fournit plusieurs opérations d'API pour prendre en charge les téléchargements partitionnés. Grâce à ces opérations d'API, vous pouvez charger les archives en plusieurs parties. Ces parties peuvent être chargées indépendamment, dans n'importe quel ordre, et en parallèle. Si le chargement d'une partie échoue, vous devez uniquement recharger cette partie, pas l'archive entière. Vous pouvez utiliser le chargement partitionné pour les archives dont la taille est comprise entre 1 octet et environ 40 000 gibioctets (Gio).

Pour plus d'informations sur les téléchargements partitionnés sur Amazon Glacier, consultez. Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)

Rubriques

- (Prérequis) Configuration du AWS CLI
- (Prérequis) Installation de Python
- (Prérequis) Création d'un coffre-fort Amazon Glacier
- Exemple : téléchargement partiel d'archives volumineuses à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

(Prérequis) Installation de Python

Pour terminer un téléchargement en plusieurs parties, vous devez calculer le hachage SHA256 arborescent de l'archive que vous chargez. Cela est différent du calcul du hachage SHA256 arborescent du fichier que vous souhaitez télécharger. Pour calculer le hachage SHA256 arborescent

de l'archive que vous téléchargez, vous pouvez utiliser Java C# (avec .NET) ou. Python Dans cet exemple, vous allez utiliser Python. Pour savoir comment utiliser Java ou C#, consultez Calcul des totaux de contrôle.

Pour plus d'informations sur l'installation de Python, consultez Install ou update Python dans le Guide du développeur Boto3.

(Prérequis) Création d'un coffre-fort Amazon Glacier

Pour utiliser l'exemple suivant, vous devez avoir créé au moins un coffre-fort Amazon Glacier. Pour plus d'informations sur la création de coffres, consultez Création d'un coffre-fort dans Amazon Glacier.

Exemple : téléchargement partiel d'archives volumineuses à l'aide du AWS CLI

Dans cet exemple, vous allez créer un fichier et le charger en plusieurs parties sur Amazon Glacierà l'aide d'opérations d'API de chargement partitionné.

> **Important**
>
> Avant de commencer cette procédure, assurez-vous d'avoir effectué toutes les étapes prérequises. Pour télécharger une archive, vous devez avoir créé un coffre, le AWS CLI configurer et être prêt à utiliser Java le langage C# ou Python à calculer un hachage SHA256 arborescent.

La procédure suivante utilise les `complete-multipart-upload` AWS CLI commandes `initiate-multipart-uploadupload-multipart-part`, et.

Pour obtenir des informations plus détaillées sur chacun de ces commandes, consultez initiate-multipart-upload, upload-multipart-part et complete-multipart-upload dans le guide de référence des commandes AWS CLI .

1. Utilisez la commande initiate-multipart-upload pour créer une ressource de chargement partitionné. Dans votre demande, spécifiez la taille des parties en nombre d'octets. Chacune des parties que vous chargez, à l'exception de la dernière, doit avoir cette taille. Vous n'avez pas besoin de connaître la taille globale de l'archive lorsque vous lancez un chargement. Cependant, vous aurez besoin de connaître la taille totale, en octets, de chaque partie pour effectuer le chargement à l'étape finale.

Dans la commande ci-dessous, remplacez les valeurs des paramètres `--vault-name` et `--account-ID` par vos propres informations. Cette commande indique que vous allez charger une archive avec des parties d'une taille d'1 mébioctet (Mio) (1 024 x 1024 octets) par fichier. Remplacez la valeur de ce paramètre `--part-size` si nécessaire.

```
aws glacier initiate-multipart-upload --vault-name awsexamplevault --part-
size 1048576 --account-id 123456789012
```

Sortie attendue :

```
{
"location": "/123456789012/vaults/awsexamplevault/multipart-uploads/uploadId",
"uploadId": "uploadId"
}
```

Une fois terminée, la commande affiche l'ID de téléchargement partitionné et l'emplacement de la ressource de téléchargement partitionné dans Amazon Glacier. Cet ID de chargement vous servira dans les prochaines étapes.

2. Pour cet exemple, vous pouvez utiliser les commandes suivantes pour créer un fichier de 4,4 Mio, le scinder en fragments de 1 Mio et charger chacun de ces fragments. Pour charger vos propres fichiers, vous pouvez suivre une procédure similaire en scindant vos données en fragments et en chargeant chaque partie.

   Linux ou macOS

   La commande suivante crée un fichier de 4,4 Mio, nommé `file_to_upload`, sur Linux ou macOS.

   ```
   mkfile -n 9000b file_to_upload
   ```

   Windows

   La commande suivante crée un fichier de 4,4 Mio, nommé `file_to_upload`, sur Windows.

   ```
   fsutil file createnew file_to_upload 4608000
   ```

3. Vous allez ensuite scinder ce fichier en fragments de 1 Mio.

```
split -b 1048576 file_to_upload chunk
```

Vous disposez maintenant des cinq fragments suivants. Les quatre premiers font 1 Mio, et le dernier environ 400 kibioctets (Kio).

```
chunkaa
chunkab
chunkac
chunkad
chunkae
```

4. Utilisez la commande upload-multipart-part pour charger une partie d'une archive. Vous pouvez charger les parties de l'archive dans n'importe quel ordre. Vous pouvez également les charger en parallèle. Vous pouvez charger jusqu'à 10 000 parties pour un chargement partitionné.

   Dans la commande suivante, remplacez les valeurs des paramètres `--vault-name`, `--account-ID` et `--upload-id`. L'ID de chargement doit correspondre à l'ID indiqué comme sortie de la commande `initiate-multipart-upload`. Le paramètre `--range` indique que vous allez charger une partie dont la taille est de 1 Mio (1 024 x 1 024 octets). Cette taille doit correspondre à celle que vous avez spécifiée dans la commande `initiate-multipart-upload`. Ajustez cette valeur de taille si nécessaire. Le paramètre `--body` indique le nom de la partie que vous chargez actuellement.

   ```
   aws glacier upload-multipart-part --body chunkaa --range='bytes 0-1048575/*' --
   vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
   ```

   En cas de succès, la commande génère une sortie contenant le total de contrôle de la partie chargée.

5. Exécutez une nouvelle fois la commande `upload-multipart-part` pour charger les parties restantes de votre chargement partitionné. Mettez à jour les valeurs des paramètres `--range` et `--body` pour chaque commande de sorte qu'elles correspondent à la partie que vous chargez.

   ```
   aws glacier upload-multipart-part --body chunkab --range='bytes 1048576-2097151/*'
    --vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
   ```

   ```
   aws glacier upload-multipart-part --body chunkac --range='bytes 2097152-3145727/*'
    --vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
   ```

```
aws glacier upload-multipart-part --body chunkad --range='bytes 3145728-4194303/*'
 --vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
```

```
aws glacier upload-multipart-part --body chunkae --range='bytes 4194304-4607999/*'
 --vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
```

> **Note**
>
> La valeur du paramètre `--range` de la commande finale est inférieure, car notre chargement fait moins de 1 Mio. En cas de succès, chaque commande génère une sortie contenant le total de contrôle de chaque partie chargée.

6. Ensuite, vous allez assembler l'archive et terminer le chargement. Vous devez inclure la taille totale et le hachage SHA256 arborescent de l'archive.

   Pour calculer le hachage SHA256 arborescent de l'archive, vous pouvez utiliser Java C# ou. Python Dans cet exemple, vous allez utiliser Python. Pour savoir comment utiliser Java ou C#, consultez Calcul des totaux de contrôle.

   Créez le fichier Python `checksum.py` et insérez le code suivant. Si nécessaire, remplacez le nom du fichier d'origine.

   ```
   from botocore.utils import calculate_tree_hash

   checksum = calculate_tree_hash(open('file_to_upload', 'rb'))
   print(checksum)
   ```

7. Exécutez `checksum.py` pour calculer le hachage de SHA256 l'arbre. Le hachage suivant peut ne pas correspondre à votre sortie.

   ```
   $ python3 checksum.py
   $ 3d760edb291bfc9d90d35809243de092aea4c47b308290ad12d084f69988ae0c
   ```

8. Utilisez la commande complete-multipart-upload pour terminer le chargement de l'archive. Remplacez les valeurs des paramètres `--vault-name`, `--account-ID`, `--upload-ID` et `--checksum`. La valeur du paramètre `--archive` indique la taille totale, en octets, de l'archive. Cette valeur doit correspondre à la somme des tailles des différentes parties que vous avez chargées. Remplacez cette valeur si nécessaire.

```
aws glacier complete-multipart-upload --archive-size 4608000 --vault-
name awsexamplevault --account-id 123456789012 --upload-id upload_ID --
checksum checksum
```

Une fois terminée, la commande affiche l'ID, la somme de contrôle et l'emplacement de l'archive dans Amazon Glacier.

Chargement d'archives volumineuses en plusieurs parties à l'aide du kit SDK Amazon pour Java

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour Java fournissent une méthode pour télécharger une archive volumineuse (Téléchargement d'une archive dans Amazon Glaciervoir).

- L'API de haut niveau fournit une méthode que vous pouvez utiliser pour charger des archives de n'importe quelle taille. Selon le fichier que vous chargez, la méthode télécharge une archive en une seule opération ou utilise le support de téléchargement partitionné d'Amazon Glacier (Amazon Glacier) pour charger l'archive en plusieurs parties.
- L'API de bas niveau est mappée étroitement à l'implémentation REST sous-jacente. Par conséquent, elle fournit une méthode pour charger des archives plus petites en une seule opération et un groupe de méthodes qui prennent en charge le chargement partitionné en plusieurs parties pour les archives plus volumineuses. Cette section explique le chargement des archives volumineuses en plusieurs parties à l'aide de l'API de bas niveau.

Pour plus d'informations sur le haut niveau et le bas niveau APIs, consultezUtilisation du AWS SDK pour Java avec Amazon Glacier.

Rubriques

- Téléchargement partiel d'archives volumineuses à l'aide de l'API de haut niveau du AWS SDK pour Java
- Téléchargez des archives volumineuses en plusieurs parties à l'aide de l'API de bas niveau du AWS SDK pour Java

Téléchargement partiel d'archives volumineuses à l'aide de l'API de haut niveau du AWS SDK pour Java

Vous utilisez les mêmes méthodes de l'API de haut niveau pour charger les archives petites ou volumineuses. En fonction de la taille de l'archive, les méthodes d'API de haut niveau décident de charger l'archive en une seule opération ou d'utiliser l'API de téléchargement en plusieurs parties fournie par Amazon Glacier. Pour de plus amples informations, veuillez consulter Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java.

Téléchargez des archives volumineuses en plusieurs parties à l'aide de l'API de bas niveau du AWS SDK pour Java

Pour bénéficier d'un contrôle précis du chargement, vous pouvez utiliser l'API de bas niveau afin de configurer la demande et traiter la réponse. Voici les étapes pour charger des archives volumineuses en plusieurs parties à l'aide du kit AWS SDK pour Java.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle vous souhaitez enregistrer l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Lancez un téléchargement partitionné en appelant la méthode `initiateMultipartUpload`.

   Vous devez fournir le nom du coffre dans lequel vous souhaitez charger l'archive, spécifier la taille des parties de l'archive à charger et indiquer éventuellement une description. Vous fournissez ces informations en créant une instance de la classe `InitiateMultipartUploadRequest`. En réponse, Amazon Glacier renvoie un ID de chargement.
3. Chargez les parties en appelant la méthode `uploadMultipartPart`.

   Pour chaque partie que vous chargez, vous devez fournir le nom du coffre, la plage d'octets dans l'archive assemblée finale qui sera chargée dans cette partie, le total de contrôle des données de la partie et l'ID du chargement.
4. Terminez le chargement partitionné en appelant la méthode `completeMultipartUpload`.

   Vous devez fournir l'ID de chargement, le total de contrôle de l'archive entière, la taille de l'archive (taille combinée de toutes les parties que avez chargées) et le nom du coffre. Amazon Glacier construit l'archive à partir des parties chargées et renvoie un ID d'archive.

Exemple : téléchargement d'une archive volumineuse en plusieurs parties à l'aide du AWS SDK pour Java

L'exemple de code Java suivant utilise le AWS SDK pour Java pour télécharger une archive dans un coffre (`examplevault`). Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du fichier que vous souhaitez charger.

> **Note**
>
> Cet exemple concerne des tailles de partie comprises entre 1 Mo et 1 Go. Toutefois, Amazon Glacier prend en charge des tailles de partie pouvant aller jusqu'à 4 Go.

Example

```
import java.io.ByteArrayInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.security.NoSuchAlgorithmException;
import java.util.Arrays;
import java.util.Date;
import java.util.LinkedList;
import java.util.List;

import com.amazonaws.AmazonClientException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.TreeHashGenerator;
import com.amazonaws.services.glacier.model.CompleteMultipartUploadRequest;
import com.amazonaws.services.glacier.model.CompleteMultipartUploadResult;
import com.amazonaws.services.glacier.model.InitiateMultipartUploadRequest;
import com.amazonaws.services.glacier.model.InitiateMultipartUploadResult;
import com.amazonaws.services.glacier.model.UploadMultipartPartRequest;
import com.amazonaws.services.glacier.model.UploadMultipartPartResult;
import com.amazonaws.util.BinaryUtils;

public class ArchiveMPU {
```

```
    public static String vaultName = "examplevault";
    // This example works for part sizes up to 1 GB.
    public static String partSize = "1048576"; // 1 MB.
    public static String archiveFilePath = "*** provide archive file path ***";
    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-west-2.amazonaws.com/");

        try {
            System.out.println("Uploading an archive.");
            String uploadId = initiateMultipartUpload();
            String checksum = uploadParts(uploadId);
            String archiveId = CompleteMultiPartUpload(uploadId, checksum);
            System.out.println("Completed an archive. ArchiveId: " + archiveId);

        } catch (Exception e) {
            System.err.println(e);
        }

    }

    private static String initiateMultipartUpload() {
        // Initiate
        InitiateMultipartUploadRequest request = new InitiateMultipartUploadRequest()
            .withVaultName(vaultName)
            .withArchiveDescription("my archive " + (new Date()))
            .withPartSize(partSize);

        InitiateMultipartUploadResult result = client.initiateMultipartUpload(request);

        System.out.println("ArchiveID: " + result.getUploadId());
        return result.getUploadId();
    }

    private static String uploadParts(String uploadId) throws AmazonServiceException,
NoSuchAlgorithmException, AmazonClientException, IOException {

        int filePosition = 0;
        long currentPosition = 0;
```

```
        byte[] buffer = new byte[Integer.valueOf(partSize)];
        List<byte[]> binaryChecksums = new LinkedList<byte[]>();

        File file = new File(archiveFilePath);
        FileInputStream fileToUpload = new FileInputStream(file);
        String contentRange;
        int read = 0;
        while (currentPosition < file.length())
        {
            read = fileToUpload.read(buffer, filePosition, buffer.length);
            if (read == -1) { break; }
            byte[] bytesRead = Arrays.copyOf(buffer, read);

            contentRange = String.format("bytes %s-%s/*", currentPosition,
currentPosition + read - 1);
            String checksum = TreeHashGenerator.calculateTreeHash(new
ByteArrayInputStream(bytesRead));
            byte[] binaryChecksum = BinaryUtils.fromHex(checksum);
            binaryChecksums.add(binaryChecksum);
            System.out.println(contentRange);

            //Upload part.
            UploadMultipartPartRequest partRequest = new UploadMultipartPartRequest()
            .withVaultName(vaultName)
            .withBody(new ByteArrayInputStream(bytesRead))
            .withChecksum(checksum)
            .withRange(contentRange)
            .withUploadId(uploadId);

            UploadMultipartPartResult partResult =
client.uploadMultipartPart(partRequest);
            System.out.println("Part uploaded, checksum: " + partResult.getChecksum());

            currentPosition = currentPosition + read;
        }
        fileToUpload.close();
        String checksum = TreeHashGenerator.calculateTreeHash(binaryChecksums);
        return checksum;
    }

    private static String CompleteMultiPartUpload(String uploadId, String checksum)
throws NoSuchAlgorithmException, IOException {

        File file = new File(archiveFilePath);
```

```
        CompleteMultipartUploadRequest compRequest = new
 CompleteMultipartUploadRequest()
            .withVaultName(vaultName)
            .withUploadId(uploadId)
            .withChecksum(checksum)
            .withArchiveSize(String.valueOf(file.length()));

        CompleteMultipartUploadResult compResult =
 client.completeMultipartUpload(compRequest);
        return compResult.getLocation();
    }
}
```

Téléchargement d'archives volumineuses à l'aide du AWS SDK pour .NET

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode pour télécharger des archives volumineuses en plusieurs parties (Téléchargement d'une archive dans Amazon Glaciervoir).

- L'API de haut niveau fournit une méthode que vous pouvez utiliser pour charger des archives de n'importe quelle taille. Selon le fichier que vous chargez, la méthode télécharge l'archive en une seule opération ou utilise le support de téléchargement en plusieurs parties d'Amazon Glacier (Amazon Glacier) pour télécharger l'archive en plusieurs parties.
- L'API de bas niveau est mappée étroitement à l'implémentation REST sous-jacente. Par conséquent, elle fournit une méthode pour charger des archives plus petites en une seule opération et un groupe de méthodes qui prennent en charge le chargement partitionné en plusieurs parties pour les archives plus volumineuses. Cette section explique le chargement des archives volumineuses en plusieurs parties à l'aide de l'API de bas niveau.

Pour plus d'informations sur le haut niveau et le bas niveau APIs, consultezUtilisation du AWS SDK pour .NET avec Amazon Glacier.

Rubriques

- Téléchargement partiel d'archives volumineuses à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Téléchargement partiel d'archives volumineuses à l'aide de l'API de bas niveau du AWS SDK pour .NET

Téléchargement partiel d'archives volumineuses à l'aide de l'API de haut niveau du AWS SDK pour .NET

Vous utilisez les mêmes méthodes de l'API de haut niveau pour charger les archives petites ou volumineuses. En fonction de la taille de l'archive, les méthodes d'API de haut niveau décident de charger l'archive en une seule opération ou d'utiliser l'API de téléchargement en plusieurs parties fournie par Amazon Glacier. Pour de plus amples informations, veuillez consulter Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET.

Téléchargement partiel d'archives volumineuses à l'aide de l'API de bas niveau du AWS SDK pour .NET

Pour bénéficier d'un contrôle précis du chargement, vous pouvez utiliser l'API de bas niveau afin de configurer la demande et traiter la réponse. Voici les étapes pour charger des archives volumineuses en plusieurs parties à l'aide du kit AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région dans laquelle vous souhaitez enregistrer l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.
2. Lancez un téléchargement partitionné en appelant la méthode `InitiateMultipartUpload`.

   Vous devez fournir le nom du coffre dans lequel vous souhaitez charger l'archive, spécifier la taille des parties de l'archive à charger et indiquer éventuellement une description. Vous fournissez ces informations en créant une instance de la classe `InitiateMultipartUploadRequest`. En réponse, Amazon Glacier renvoie un ID de chargement.
3. Chargez les parties en appelant la méthode `UploadMultipartPart`.

   Pour chaque partie que vous chargez, vous devez fournir le nom du coffre, la plage d'octets dans l'archive assemblée finale qui sera chargée dans cette partie, le total de contrôle des données de la partie et l'ID du chargement.
4. Terminez le chargement partitionné en appelant la méthode `CompleteMultipartUpload`.

   Vous devez fournir l'ID de chargement, le total de contrôle de l'archive entière, la taille de l'archive (taille combinée de toutes les parties que avez chargées) et le nom du coffre. Amazon Glacier construit l'archive à partir des parties chargées et renvoie un ID d'archive.

Exemple : Chargement d'une archive volumineuse en plusieurs parties à l'aide du kit SDK Amazon pour .NET

L'exemple de code C# suivant utilise le AWS SDK pour .NET pour télécharger une archive dans un coffre (`examplevault`). Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec le nom d'un fichier que vous souhaitez charger.

Example

```
using System;
using System.Collections.Generic;
using System.IO;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveUploadMPU
  {
    static string vaultName       = "examplevault";
    static string archiveToUpload = "*** Provide file name (with full path) to upload
 ***";
    static long partSize          = 4194304; // 4 MB.

    public static void Main(string[] args)
    {
      AmazonGlacierClient client;
      List<string> partChecksumList = new List<string>();
      try
      {
         using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Uploading an archive.");
          string uploadId  = InitiateMultipartUpload(client);
          partChecksumList = UploadParts(uploadId, client);
          string archiveId = CompleteMPU(uploadId, client, partChecksumList);
          Console.WriteLine("Archive ID: {0}", archiveId);
        }
        Console.WriteLine("Operations successful. To continue, press Enter");
        Console.ReadKey();
      }
```

```
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }

    static string InitiateMultipartUpload(AmazonGlacierClient client)
    {
      InitiateMultipartUploadRequest initiateMPUrequest = new
InitiateMultipartUploadRequest()
      {

        VaultName = vaultName,
        PartSize = partSize,
        ArchiveDescription = "Test doc uploaded using MPU."
      };

      InitiateMultipartUploadResponse initiateMPUresponse =
client.InitiateMultipartUpload(initiateMPUrequest);

      return initiateMPUresponse.UploadId;
    }

    static List<string> UploadParts(string uploadID, AmazonGlacierClient client)
    {
      List<string> partChecksumList = new List<string>();
      long currentPosition = 0;
      var buffer = new byte[Convert.ToInt32(partSize)];

      long fileLength = new FileInfo(archiveToUpload).Length;
      using (FileStream fileToUpload = new FileStream(archiveToUpload, FileMode.Open,
FileAccess.Read))
      {
        while (fileToUpload.Position < fileLength)
        {
          Stream uploadPartStream = GlacierUtils.CreatePartStream(fileToUpload,
partSize);
          string checksum = TreeHashGenerator.CalculateTreeHash(uploadPartStream);
          partChecksumList.Add(checksum);
          // Upload part.
          UploadMultipartPartRequest uploadMPUrequest = new
UploadMultipartPartRequest()
          {
```

```
            VaultName = vaultName,
            Body = uploadPartStream,
            Checksum = checksum,
            UploadId = uploadID
          };
          uploadMPUrequest.SetRange(currentPosition, currentPosition +
 uploadPartStream.Length - 1);
          client.UploadMultipartPart(uploadMPUrequest);

          currentPosition = currentPosition + uploadPartStream.Length;
        }
      }
      return partChecksumList;
    }

    static string CompleteMPU(string uploadID, AmazonGlacierClient client, List<string>
 partChecksumList)
    {
      long fileLength = new FileInfo(archiveToUpload).Length;
      CompleteMultipartUploadRequest completeMPUrequest = new
 CompleteMultipartUploadRequest()
      {
        UploadId = uploadID,
        ArchiveSize = fileLength.ToString(),
        Checksum = TreeHashGenerator.CalculateTreeHash(partChecksumList),
        VaultName = vaultName
      };

      CompleteMultipartUploadResponse completeMPUresponse =
 client.CompleteMultipartUpload(completeMPUrequest);
      return completeMPUresponse.ArchiveId;
    }
  }
}
```

Chargement d'archives volumineuses en plusieurs parties à l'aide de l'API REST

Comme indiqué dans Chargement d'archives volumineuses en plusieurs parties (chargement partitionné), le chargement partitionné fait référence à un ensemble d'opérations qui vous permettent de charger une archive en plusieurs parties et d'effectuer des opérations connexes. Pour plus d'informations sur ces opérations, consultez les rubriques de référence API suivantes :

- Lancement de chargement partitionné (POST multipart-uploads)
- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Annulation de chargement partitionné (DELETE uploadID)
- Répertorier les parties (GET uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)

Téléchargement d'une archive dans Amazon Glacier

Amazon Glacier fournit une console de gestion que vous pouvez utiliser pour créer et supprimer des coffres-forts. Cependant, vous ne pouvez pas télécharger d'archives depuis Amazon Glacier à l'aide de la console de gestion. Pour télécharger des données, telles que des photos, des vidéos et d'autres documents, vous devez soit utiliser le AWS Command Line Interface (AWS CLI), soit écrire du code pour effectuer des requêtes, en utilisant soit directement l'API REST, soit en utilisant le AWS SDKs.

Pour plus d'informations sur l'utilisation d'Amazon Glacier avec le AWS CLI, consultez la AWS CLI référence relative à Amazon Glacier. Pour installer le AWS CLI, voir AWS Command Line Interface. Les rubriques suivantes décrivent comment télécharger des archives sur Amazon Glacier à l'aide de l'API REST Amazon Glacier AWS SDK pour Java, de AWS SDK pour .NET, et de l'API REST Amazon Glacier.

Rubriques

- Récupération des archives Amazon Glacier
- Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS SDK pour Java
- Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Téléchargement d'une archive volumineuse à l'aide du traitement parallèle avec Python
- Téléchargement d'une archive à l'aide de l'API REST
- Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS CLI

Récupération des archives Amazon Glacier

La récupération d'une archive depuis Amazon Glacier est une opération asynchrone au cours de laquelle vous lancez d'abord une tâche, puis vous téléchargez le résultat une fois la tâche terminée. Pour lancer une tâche de récupération d'archives, vous devez utiliser l'opération d'Lancement d'une tâche (POST jobs)API REST ou l'équivalent dans le AWS CLI, ou le AWS SDKs.

Rubriques

- Options de récupération des archives
- Récupération d'une plage d'archives

La récupération d'une archive depuis Amazon Glacier est un processus en deux étapes. Ce qui suit est un aperçu de ce processus.

Pour récupérer une archive

1. Lancez une tâche d'extraction d'archive.

   a. Obtenez l'ID de l'archive que vous souhaitez extraire. Vous pouvez obtenir l'ID de l'archive à partir d'un inventaire du coffre. Vous pouvez obtenir l'ID de l'archive avec l'API REST AWS CLI, ou AWS SDKs. Pour de plus amples informations, veuillez consulter Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

   b. Lancez une tâche qui demande à Amazon Glacier de préparer une archive complète ou une partie de l'archive pour un téléchargement ultérieur à l'aide de cette Lancement d'une tâche (POST jobs) opération.

   Lorsque vous lancez une tâche, Amazon Glacier renvoie un ID de tâche dans la réponse et exécute la tâche de manière asynchrone. (Vous ne pouvez pas télécharger la sortie de la tâche tant que celle-ci n'est pas terminée, comme indiqué à l'étape 2.)

   > **Important**
   >
   > Pour les extractions de type Standard uniquement, une politique d'extraction de données peut entraîner l'échec de votre demande `Initiate Job` avec une exception `PolicyEnforcedException`. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier. Pour plus d'informations sur l'exception `PolicyEnforcedException`, consultez la section Réponses d'erreur.

   Si nécessaire, vous pouvez restaurer de larges segments des données stockées dans Amazon Glacier. Pour plus d'informations sur la restauration des données à partir des classes de stockage Amazon Glacier, consultez la section Classes de stockage pour les objets d'archivage dans le guide de l'utilisateur d'Amazon Simple Storage Service.

2. Une fois la tâche terminée, téléchargez les octets à l'aide de l'opération Génération de sortie de tâche (GET output).

   Vous pouvez télécharger tous les octets ou spécifier une plage d'octets afin de télécharger une seule partie de la sortie de la tâche. Pour les sorties plus volumineuses, le téléchargement en plusieurs parties s'avère utile en cas d'échec du téléchargement, par exemple en raison d'une défaillance réseau. Si vous générez la sortie de la tâche dans le cadre d'une seule demande et qu'une défaillance réseau se produit, vous devez redémarrer le téléchargement de la sortie depuis le début. En revanche, si vous téléchargez la sortie en plusieurs parties et qu'une défaillance se produit, vous devez uniquement redémarrer le téléchargement de la partie et non celui de la sortie entière.

Amazon Glacier doit terminer une tâche avant que vous puissiez obtenir sa sortie. Après son achèvement, une tâche n'expire pas avant au moins 24 heures, ce qui signifie que vous disposez d'un délai de 24 heures pour télécharger la sortie une fois la tâche terminée. La restauration peut expirer à tout moment 24 heures après la fin de la tâche. Pour déterminer si votre tâche est terminée, vérifiez son état en utilisant l'une des options suivantes :

- Attendez une notification de fin de tâche : vous pouvez spécifier une rubrique Amazon Simple Notification Service (Amazon SNS) sur laquelle Amazon Glacier pourra publier une notification une fois la tâche terminée. Amazon Glacier envoie une notification uniquement une fois la tâche terminée.

  Vous pouvez spécifier une rubrique Amazon SNS pour une tâche au moment où vous la lancez. En plus de spécifier une rubrique Amazon SNS dans votre demande d'emploi, si votre coffre-fort contient des notifications définies pour les événements de récupération d'archives, Amazon Glacier publie également une notification à cette rubrique SNS. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier.
- Demander des informations sur les offres d'emploi de manière explicite : vous pouvez également utiliser l'opération d'`Describe Job`API Amazon Glacier (Description de la tâche (GET JobID)) pour demander régulièrement des informations sur les offres d'emploi. Cependant, nous vous recommandons d'utiliser des notifications Amazon SNS.

> **Note**
>
> Les informations que vous obtenez en utilisant une notification Amazon SNS sont identiques à celles obtenues en appelant l'opération d'API `Describe Job`.

Options de récupération des archives

Lorsque vous lancez une tâche en vue d'extraire une archive, vous pouvez spécifier l'une des options d'exaction suivantes, en fonction de vos exigences en matière de temps d'accès et de coûts. Pour plus d'informations sur les tarifs de récupération, consultez la section Tarification d'Amazon Glacier.

- Expedited : les extractions de type Expedited vous permettent d'accéder rapidement à vos données stockées dans la classe de stockage S3 Glacier Flexible Retrieval ou au niveau Accès aux archives S3 Intelligent-Tiering lorsque des demandes urgentes occasionnelles de restauration d'archives se présentent. Pour toutes les archives à l'exception des plus volumineuses (plus de 250 Mo), les données auxquelles vous accédez à l'aide des extractions Expedited sont généralement disponibles dans un délai compris entre 1 et 5 minutes. La capacité provisionnée garantit que la capacité des récupérations rapides est disponible lorsque vous en avez besoin. Pour de plus amples informations, veuillez consulter Capacité provisionnée.
- Standard : les extractions de type Standard vous permettent d'accéder à vos archives en plusieurs heures. Les extractions Standard prennent généralement entre 3 et 5 heures. Standard constitue l'option par défaut pour les demandes de récupération qui ne spécifient pas l'option de récupération.
- En vrac : les extractions en masse constituent l'option la moins coûteuse d'Amazon Glacier. Vous pouvez l'utiliser pour récupérer de grandes quantités, voire des pétaoctets, de données à moindre coût par jour. Les extractions Bulk prennent généralement entre 5 et 12 heures.

Le tableau suivant récapitule les options de récupération d'archive. Pour plus d'informations sur la tarification, consultez Tarification Amazon Glacier.

Pour effectuer une ou une `Bulk` extraction `ExpeditedStandard`, définissez l'élément de requête dans la `Tier` demande d'opération de l'RestoreObjectAPI REST sur l'option que vous souhaitez, ou sur l'équivalent dans le AWS Command Line Interface (AWS CLI) ou AWS SDKs. Si vous avez acheté une capacité provisionnée, toutes les récupérations rapides de type Expedited sont automatiquement transmises par le biais de votre capacité provisionnée.

Capacité provisionnée

La capacité provisionnée vous permet de disposer d'une capacité d'extraction pour les extractions de type Expedited lorsque vous en avez besoin. Chaque unité de capacité permet d'effectuer au moins trois extractions accélérées toutes les 5 minutes et fournit un débit de récupération allant jusqu'à 150 mégaoctets par seconde (). MBps

Si votre charge de travail nécessite un accès extrêmement fiable et prévisible à un sous-ensemble de vos données en quelques minutes, nous vous recommandons d'acheter de la capacité d'extraction provisionnée. Sans capacité provisionnée, les extractions de type Expedited sont généralement acceptées, sauf dans les rares cas où la demande est particulièrement élevée. Toutefois, si vous avez besoin d'un accès aux récupérations rapides en toutes circonstances, vous devez acheter la capacité de récupération provisionnée.

Achat d'une capacité allouée

Vous pouvez acheter des unités de capacité provisionnées à l'aide de la console Amazon Glacier, de l'opération de l'Acheter la capacité provisionnée (POST provisioned-capacity)API REST AWS SDKs, du ou du AWS CLI. Pour obtenir des informations sur la tarification de la capacité allouée, consultez la section Tarification d'Amazon Glacier.

Une unité de capacité provisionnée a une durée d'un mois à partir de la date et de l'heure d'achat.

Si la date de début est le 31e jour du mois, la date d'expiration correspond au dernier jour du mois suivant. Par exemple, si la date de début est le 31 août, la date d'expiration est le 30 septembre. Si la date de début est le 31 janvier, la date d'expiration est le 28 février.

Pour acheter de la capacité provisionnée à l'aide de la console Amazon Glacier

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Dans le panneau de navigation de gauche, choisissez Paramètres d'extraction de données.
3. Sous Unités de capacité provisionnées (PCUs), choisissez Acheter un PCU. La boîte de dialogue Acheter la PCU s'affiche.
4. Si vous souhaitez acheter de la capacité provisionnée, saisissez **`confirm`** dans la zone Confirmer l'achat.
5. Choisissez Acheter la PCU.

Récupération d'une plage d'archives

Lorsque vous extrayez une archive d'Amazon Glacier, vous pouvez éventuellement spécifier une plage, ou une partie, de l'archive à extraire. Le comportement par défaut consiste à extraire la totalité de l'archive. La spécification d'une plage d'octets peut vous être utile lorsque vous souhaitez effectuer les actions suivantes :

- Gérez vos téléchargements de données : Amazon Glacier autorise le téléchargement des données récupérées pendant 24 heures après la fin de la demande de récupération. Par conséquent, vous souhaiterez peut-être extraire uniquement certaines parties de l'archive afin de pouvoir gérer le programme des téléchargements dans la fenêtre de téléchargement définie.
- Extraire une partie ciblée d'une archive volumineuse : par exemple, supposez que vous avez regroupé un grand nombre de fichiers, que vous les avez chargés sous la forme d'une archive unique et que vous souhaitez à présent extraire une partie des fichiers. Dans ce cas, vous pouvez spécifier une plage de l'archive qui contient les fichiers qui vous intéressent en utilisant une demande d'extraction. Vous pouvez aussi lancer plusieurs demandes d'extraction, chacune d'elles définissant une plage d'un ou de plusieurs fichiers.

Lors du lancement d'une tâche d'extraction à l'aide d'extractions par plage, vous devez fournir une plage alignée en termes de méga-octet. En d'autres termes, la plage d'octets peut commencer à zéro (début de l'archive) ou à n'importe quel intervalle d'1 Mo suivant (1 Mo, 2 Mo, 3 Mo, etc.).

La fin de la section peut correspondre à la fin de votre archive ou être indiquée, là encore, selon tout intervalle de 1 Mo supérieur à celui de départ. Par ailleurs, si vous souhaitez obtenir des valeurs de total de contrôle lorsque vous téléchargez les données (une fois que la tâche d'extraction est terminée), la plage que vous demandez lors du lancement de la tâche doit également être alignée avec le hachage d'arborescence. Vous pouvez utiliser des totaux de contrôle pour vérifier que les données n'ont pas été endommagées pendant la transmission. Pour plus d'informations sur l'alignement en termes de méga-octets et l'alignement avec le hachage d'arborescence, consultez la section Réception des totaux de contrôle lors du téléchargement de données.

Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS SDK pour Java

Le haut niveau et le bas niveau APIs fournis par le SDK Amazon pour Java fournissent une méthode pour télécharger une archive.

Rubriques

- Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java
- Téléchargement d'une archive à l'aide de l'API de bas niveau du AWS SDK pour Java

Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java

La classe `ArchiveTransferManager` de l'API de haut niveau fournit la méthode `download` que vous pouvez utiliser pour télécharger une archive.

> **Important**
>
> La classe `ArchiveTransferManager` crée une rubrique Amazon Simple Notification Service (Amazon SNS) et une file d'attente Amazon Simple Queue Service (Amazon SQS) qui est abonnée à cette rubrique. Elle lance ensuite la tâche d'extraction d'archive et interroge la file d'attente pour que l'archive soit disponible. Une fois que l'archive est disponible, le téléchargement commence. Pour plus d'informations sur les délais d'extraction, consultez Options de récupération des archives.

Exemple : téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour Java

L'exemple de code Java suivant télécharge une archive à partir d'un coffre (`examplevault`) de la région USA Ouest (Oregon) (`us-west-2`).

Pour step-by-step obtenir des instructions relatives à l'exécution de cet exemple, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code avec un ID d'archive existant et le chemin d'accès au fichier local où vous souhaitez enregistrer l'archive téléchargée, comme illustré ci-après.

Example

```
import java.io.File;
import java.io.IOException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.transfer.ArchiveTransferManager;
import com.amazonaws.services.sns.AmazonSNSClient;
import com.amazonaws.services.sqs.AmazonSQSClient;
```

```
public class ArchiveDownloadHighLevel {
    public static String vaultName = "examplevault";
    public static String archiveId = "*** provide archive ID ***";
    public static String downloadFilePath  = "*** provide location to download archive
 ***";

    public static AmazonGlacierClient glacierClient;
    public static AmazonSQSClient sqsClient;
    public static AmazonSNSClient snsClient;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        glacierClient = new AmazonGlacierClient(credentials);

        sqsClient = new AmazonSQSClient(credentials);
        snsClient = new AmazonSNSClient(credentials);
        glacierClient.setEndpoint("glacier.us-west-2.amazonaws.com");
        sqsClient.setEndpoint("sqs.us-west-2.amazonaws.com");
        snsClient.setEndpoint("sns.us-west-2.amazonaws.com");

        try {
            ArchiveTransferManager atm = new ArchiveTransferManager(glacierClient,
 sqsClient, snsClient);

            atm.download(vaultName, archiveId, new File(downloadFilePath));
            System.out.println("Downloaded file to " + downloadFilePath);

        } catch (Exception e)
        {
            System.err.println(e);
        }
    }
}
```

Téléchargement d'une archive à l'aide de l'API de bas niveau du AWS SDK pour Java

Voici les étapes d'extraction d'un inventaire de coffre à l'aide de l'API de bas niveau du kit AWS SDK pour Java :

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

Vous devez spécifier AWS la région à partir de laquelle vous souhaitez télécharger l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Lancez une tâche `archive-retrieval` en exécutant la méthode `initiateJob`.

   Vous fournissez des informations sur la tâche, telles que l'ID d'archive de l'archive que vous souhaitez télécharger et la rubrique Amazon SNS facultative sur laquelle vous souhaitez qu'Amazon Glacier (Amazon Glacier) publie un message de fin de tâche, en créant une instance de la `InitiateJobRequest` classe. Amazon Glacier renvoie un numéro de tâche en réponse. La réponse est disponible dans une instance de la classe `InitiateJobResult`.

```
JobParameters jobParameters = new JobParameters()
    .withArchiveId("*** provide an archive id ***")
    .withDescription("archive retrieval")
    .withRetrievalByteRange("*** provide a retrieval range***") // optional
    .withType("archive-retrieval");

InitiateJobResult initiateJobResult = client.initiateJob(new InitiateJobRequest()
    .withJobParameters(jobParameters)
    .withVaultName(vaultName));

String jobId = initiateJobResult.getJobId();
```

   Vous pouvez éventuellement spécifier une plage d'octets pour demander à Amazon Glacier de préparer uniquement une partie de l'archive. Par exemple, vous pouvez mettre à jour la demande précédente en ajoutant l'instruction suivante pour demander à Amazon Glacier de ne préparer que la partie 1 Mo à 2 Mo de l'archive.

```
int ONE_MEG = 1048576;
String retrievalByteRange = String.format("%s-%s", ONE_MEG, 2*ONE_MEG -1);

JobParameters jobParameters = new JobParameters()
    .withType("archive-retrieval")
    .withArchiveId(archiveId)
    .withRetrievalByteRange(retrievalByteRange)
    .withSNSTopic(snsTopicARN);

InitiateJobResult initiateJobResult = client.initiateJob(new InitiateJobRequest()
    .withJobParameters(jobParameters)
```

```
    .withVaultName(vaultName));

String jobId = initiateJobResult.getJobId();
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification dans le coffre identifiant une rubrique Amazon Simple Notification Service (Amazon SNS) ou si vous avez spécifié une rubrique Amazon SNS lorsque vous avez lancé une tâche, Amazon Glacier envoie un message à cette rubrique une fois la tâche terminée.

   Vous pouvez également interroger Amazon Glacier en appelant la `describeJob` méthode pour déterminer l'état d'achèvement de la tâche. Toutefois, l'approche recommandée consiste à utiliser une rubrique Amazon SNS pour la notification.

4. Téléchargez la sortie de la tâche (données archivées) en exécutant la méthode `getJobOutput`.

   Vous indiquez les informations sur la demande, telles que l'ID de la tâche et le nom du coffre, en créant une instance de la classe `GetJobOutputRequest`. La sortie renvoyée par Amazon Glacier est disponible dans l'`GetJobOutputResult`objet.

```
GetJobOutputRequest jobOutputRequest = new GetJobOutputRequest()
        .withJobId("*** provide a job ID ***")
        .withVaultName("*** provide a vault name ****");
GetJobOutputResult jobOutputResult = client.getJobOutput(jobOutputRequest);

// jobOutputResult.getBody() // Provides the input stream.
```

   L'extrait de code précédent télécharge la totalité de la sortie de la tâche. Vous pouvez éventuellement extraire une partie de la sortie ou télécharger la totalité de la sortie en plusieurs parties en spécifiant la plage d'octets dans votre demande `GetJobOutputRequest`.

```
GetJobOutputRequest jobOutputRequest = new GetJobOutputRequest()
        .withJobId("*** provide a job ID ***")
        .withRange("bytes=0-1048575")   // Download only the first 1 MB of the
 output.
        .withVaultName("*** provide a vault name ****");
```

En réponse à votre `GetJobOutput` appel, Amazon Glacier renvoie la somme de contrôle de la partie des données que vous avez téléchargée, si certaines conditions sont remplies. Pour de plus amples informations, veuillez consulter Réception des totaux de contrôle lors du téléchargement de données.

Pour vérifier que le téléchargement ne comporte aucune erreur, vous pouvez ensuite calculer la somme de contrôle côté client et la comparer à la somme de contrôle envoyée par Amazon Glacier dans la réponse.

Pour une tâche de récupération d'archives avec la plage optionnelle spécifiée, lorsque vous obtenez la description de la tâche, elle inclut la somme de contrôle de la plage que vous récupérez (). SHA256 TreeHash Vous pouvez utiliser cette valeur pour mieux vérifier la précision de la totalité de la plage d'octets que vous téléchargez plus tard. Par exemple, si vous lancez une tâche pour extraire une plage d'une archive alignée sur le hachage d'arborescence, que vous téléchargez la sortie en plusieurs parties et que chacune de vos demandes `GetJobOutput` renvoie un total de contrôle, vous pouvez ensuite calculer le total de contrôle de chaque partie téléchargée côté client, puis calculer le hachage d'arborescence. Vous pouvez le comparer à la somme de contrôle renvoyée par Amazon Glacier en réponse à votre demande de travail de description afin de vérifier que l'ensemble de la plage d'octets que vous avez téléchargée est identique à la plage d'octets stockée dans Amazon Glacier.

Pour obtenir un exemple pratique, consultez Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour Java—Download in Chunks.

Exemple 1 : extraction d'une archive à l'aide de l'API de bas niveau du AWS SDK pour Java

L'exemple de code Java suivant télécharge une archive à partir du coffre spécifié. Une fois la tâche terminée, l'exemple télécharge la totalité de la sortie dans un seul appel `getJobOutput`. Pour obtenir un exemple de téléchargement d'une sortie en plusieurs parties, consultez la section Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour Java—Download in Chunks.

L’exemple exécute les tâches suivantes :

- Création d'une rubrique Amazon Simple Notification Service (Amazon SNS).

  Amazon Glacier envoie une notification à cette rubrique une fois le travail terminé.

- Création d'une file d'attente Amazon Simple Queue Service (Amazon SQS).

  L'exemple attache une stratégie à la file d'attente pour permettre à la rubrique Amazon SNS de publier des messages dans la file d'attente.
- Lancement d'une tâche de téléchargement de l'archive spécifiée.

  Dans la demande de travail, le sujet Amazon SNS créé est spécifié afin qu'Amazon Glacier puisse publier une notification sur le sujet une fois le travail terminé.
- Recherche périodiquement un message contenant l'ID de la tâche dans la file d'attente Amazon SQS.

  Si un message est présent, l'exemple analyse le JSON et vérifie si la tâche s'est correctement déroulée. Si tel est le cas, il télécharge l'archive.
- Nettoyage par la suppression de la rubrique Amazon SNS et de la file d'attente Amazon SQS qu'il a créées.

```
import java.io.BufferedInputStream;
import java.io.BufferedOutputStream;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileOutputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import org.codehaus.jackson.JsonFactory;
import org.codehaus.jackson.JsonNode;
import org.codehaus.jackson.JsonParseException;
import org.codehaus.jackson.JsonParser;
import org.codehaus.jackson.map.ObjectMapper;

import com.amazonaws.AmazonClientException;
import com.amazonaws.auth.policy.Policy;
import com.amazonaws.auth.policy.Principal;
import com.amazonaws.auth.policy.Resource;
```

```
import com.amazonaws.auth.policy.Statement;
import com.amazonaws.auth.policy.Statement.Effect;
import com.amazonaws.auth.policy.actions.SQSActions;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.GetJobOutputRequest;
import com.amazonaws.services.glacier.model.GetJobOutputResult;
import com.amazonaws.services.glacier.model.InitiateJobRequest;
import com.amazonaws.services.glacier.model.InitiateJobResult;
import com.amazonaws.services.glacier.model.JobParameters;
import com.amazonaws.services.sns.AmazonSNSClient;
import com.amazonaws.services.sns.model.CreateTopicRequest;
import com.amazonaws.services.sns.model.CreateTopicResult;
import com.amazonaws.services.sns.model.DeleteTopicRequest;
import com.amazonaws.services.sns.model.SubscribeRequest;
import com.amazonaws.services.sns.model.SubscribeResult;
import com.amazonaws.services.sns.model.UnsubscribeRequest;
import com.amazonaws.services.sqs.AmazonSQSClient;
import com.amazonaws.services.sqs.model.CreateQueueRequest;
import com.amazonaws.services.sqs.model.CreateQueueResult;
import com.amazonaws.services.sqs.model.DeleteQueueRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesResult;
import com.amazonaws.services.sqs.model.Message;
import com.amazonaws.services.sqs.model.ReceiveMessageRequest;
import com.amazonaws.services.sqs.model.SetQueueAttributesRequest;


public class AmazonGlacierDownloadArchiveWithSQSPolling {

    public static String archiveId = "*** provide archive ID ****";
    public static String vaultName = "*** provide vault name ***";
    public static String snsTopicName = "*** provide topic name ***";
    public static String sqsQueueName = "*** provide queue name ***";
    public static String sqsQueueARN;
    public static String sqsQueueURL;
    public static String snsTopicARN;
    public static String snsSubscriptionARN;
    public static String fileName = "*** provide file name ***";
    public static String region = "*** region ***";
    public static long sleepTime = 600;
    public static AmazonGlacierClient client;
    public static AmazonSQSClient sqsClient;
    public static AmazonSNSClient snsClient;
```

```
    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier." + region + ".amazonaws.com");
        sqsClient = new AmazonSQSClient(credentials);
        sqsClient.setEndpoint("https://sqs." + region + ".amazonaws.com");
        snsClient = new AmazonSNSClient(credentials);
        snsClient.setEndpoint("https://sns." + region + ".amazonaws.com");

        try {
            setupSQS();

            setupSNS();

            String jobId = initiateJobRequest();
            System.out.println("Jobid = " + jobId);

            Boolean success = waitForJobToComplete(jobId, sqsQueueURL);
            if (!success) { throw new Exception("Job did not complete
successfully."); }

            downloadJobOutput(jobId);

            cleanUp();

        } catch (Exception e) {
            System.err.println("Archive retrieval failed.");
            System.err.println(e);
        }
    }

    private static void setupSQS() {
        CreateQueueRequest request = new CreateQueueRequest()
            .withQueueName(sqsQueueName);
        CreateQueueResult result = sqsClient.createQueue(request);
        sqsQueueURL = result.getQueueUrl();

        GetQueueAttributesRequest qRequest = new GetQueueAttributesRequest()
            .withQueueUrl(sqsQueueURL)
            .withAttributeNames("QueueArn");
```

```
        GetQueueAttributesResult qResult = sqsClient.getQueueAttributes(qRequest);
        sqsQueueARN = qResult.getAttributes().get("QueueArn");

        Policy sqsPolicy =
            new Policy().withStatements(
                    new Statement(Effect.Allow)
                    .withPrincipals(Principal.AllUsers)
                    .withActions(SQSActions.SendMessage)
                    .withResources(new Resource(sqsQueueARN)));
        Map<String, String> queueAttributes = new HashMap<String, String>();
        queueAttributes.put("Policy", sqsPolicy.toJson());
        sqsClient.setQueueAttributes(new SetQueueAttributesRequest(sqsQueueURL,
queueAttributes));

    }
    private static void setupSNS() {
        CreateTopicRequest request = new CreateTopicRequest()
            .withName(snsTopicName);
        CreateTopicResult result = snsClient.createTopic(request);
        snsTopicARN = result.getTopicArn();

        SubscribeRequest request2 = new SubscribeRequest()
            .withTopicArn(snsTopicARN)
            .withEndpoint(sqsQueueARN)
            .withProtocol("sqs");
        SubscribeResult result2 = snsClient.subscribe(request2);

        snsSubscriptionARN = result2.getSubscriptionArn();
    }
    private static String initiateJobRequest() {

        JobParameters jobParameters = new JobParameters()
            .withType("archive-retrieval")
            .withArchiveId(archiveId)
            .withSNSTopic(snsTopicARN);

        InitiateJobRequest request = new InitiateJobRequest()
            .withVaultName(vaultName)
            .withJobParameters(jobParameters);

        InitiateJobResult response = client.initiateJob(request);

        return response.getJobId();
    }
```

```
   private static Boolean waitForJobToComplete(String jobId, String sqsQueueUrl)
throws InterruptedException, JsonParseException, IOException {

       Boolean messageFound = false;
       Boolean jobSuccessful = false;
       ObjectMapper mapper = new ObjectMapper();
       JsonFactory factory = mapper.getJsonFactory();

       while (!messageFound) {
           List<Message> msgs = sqsClient.receiveMessage(
              new
ReceiveMessageRequest(sqsQueueUrl).withMaxNumberOfMessages(10)).getMessages();

           if (msgs.size() > 0) {
               for (Message m : msgs) {
                   JsonParser jpMessage = factory.createJsonParser(m.getBody());
                   JsonNode jobMessageNode = mapper.readTree(jpMessage);
                   String jobMessage = jobMessageNode.get("Message").getTextValue();

                   JsonParser jpDesc = factory.createJsonParser(jobMessage);
                   JsonNode jobDescNode = mapper.readTree(jpDesc);
                   String retrievedJobId = jobDescNode.get("JobId").getTextValue();
                   String statusCode = jobDescNode.get("StatusCode").getTextValue();
                   if (retrievedJobId.equals(jobId)) {
                       messageFound = true;
                       if (statusCode.equals("Succeeded")) {
                           jobSuccessful = true;
                       }
                   }
               }

           } else {
             Thread.sleep(sleepTime * 1000);
           }
         }
       return (messageFound && jobSuccessful);
   }

   private static void downloadJobOutput(String jobId) throws IOException {

       GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
           .withVaultName(vaultName)
           .withJobId(jobId);
```

```
        GetJobOutputResult getJobOutputResult =
 client.getJobOutput(getJobOutputRequest);

        InputStream input = new BufferedInputStream(getJobOutputResult.getBody());
        OutputStream output = null;
        try {
            output = new BufferedOutputStream(new FileOutputStream(fileName));

            byte[] buffer = new byte[1024 * 1024];

            int bytesRead = 0;
            do {
                bytesRead = input.read(buffer);
                if (bytesRead <= 0) break;
                output.write(buffer, 0, bytesRead);
            } while (bytesRead > 0);
        } catch (IOException e) {
            throw new AmazonClientException("Unable to save archive", e);
        } finally {
            try {input.close();}  catch (Exception e) {}
            try {output.close();} catch (Exception e) {}
        }
        System.out.println("Retrieved archive to " + fileName);
    }

    private static void cleanUp() {
        snsClient.unsubscribe(new UnsubscribeRequest(snsSubscriptionARN));
        snsClient.deleteTopic(new DeleteTopicRequest(snsTopicARN));
        sqsClient.deleteQueue(new DeleteQueueRequest(sqsQueueURL));
    }
}
```

Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour Java —Download in Chunks

L'exemple de code Java suivant extrait une archive d'Amazon Glacier. L'exemple de code télécharge la sortie de la tâche en plusieurs parties en spécifiant une plage d'octets dans un objet `GetJobOutputRequest`.

```
import java.io.BufferedInputStream;
import java.io.ByteArrayInputStream;
import java.io.FileOutputStream;
```

```
import java.io.IOException;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import com.fasterxml.jackson.core.JsonFactory;
import com.fasterxml.jackson.core.JsonParseException;
import com.fasterxml.jackson.core.JsonParser;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;

import com.amazonaws.auth.policy.Policy;
import com.amazonaws.auth.policy.Principal;
import com.amazonaws.auth.policy.Resource;
import com.amazonaws.auth.policy.Statement;
import com.amazonaws.auth.policy.Statement.Effect;
import com.amazonaws.auth.policy.actions.SQSActions;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.TreeHashGenerator;
import com.amazonaws.services.glacier.model.GetJobOutputRequest;
import com.amazonaws.services.glacier.model.GetJobOutputResult;
import com.amazonaws.services.glacier.model.InitiateJobRequest;
import com.amazonaws.services.glacier.model.InitiateJobResult;
import com.amazonaws.services.glacier.model.JobParameters;
import com.amazonaws.services.sns.AmazonSNSClient;
import com.amazonaws.services.sns.model.CreateTopicRequest;
import com.amazonaws.services.sns.model.CreateTopicResult;
import com.amazonaws.services.sns.model.DeleteTopicRequest;
import com.amazonaws.services.sns.model.SubscribeRequest;
import com.amazonaws.services.sns.model.SubscribeResult;
import com.amazonaws.services.sns.model.UnsubscribeRequest;
import com.amazonaws.services.sqs.AmazonSQSClient;
import com.amazonaws.services.sqs.model.CreateQueueRequest;
import com.amazonaws.services.sqs.model.CreateQueueResult;
import com.amazonaws.services.sqs.model.DeleteQueueRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesRequest;
import com.amazonaws.services.sqs.model.GetQueueAttributesResult;
import com.amazonaws.services.sqs.model.Message;
import com.amazonaws.services.sqs.model.ReceiveMessageRequest;
import com.amazonaws.services.sqs.model.SetQueueAttributesRequest;


public class ArchiveDownloadLowLevelWithRange {
```

```
    public static String vaultName = "*** provide vault name ***";
    public static String archiveId = "*** provide archive id ***";
    public static String snsTopicName = "glacier-temp-sns-topic";
    public static String sqsQueueName = "glacier-temp-sqs-queue";
    public static long downloadChunkSize = 4194304; // 4 MB
    public static String sqsQueueARN;
    public static String sqsQueueURL;
    public static String snsTopicARN;
    public static String snsSubscriptionARN;
    public static String fileName = "*** provide file name to save archive to ***";
    public static String region   = "*** region ***";
    public static long sleepTime  = 600;

    public static AmazonGlacierClient client;
    public static AmazonSQSClient sqsClient;
    public static AmazonSNSClient snsClient;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier." + region + ".amazonaws.com");
        sqsClient = new AmazonSQSClient(credentials);
        sqsClient.setEndpoint("https://sqs." + region + ".amazonaws.com");
        snsClient = new AmazonSNSClient(credentials);
        snsClient.setEndpoint("https://sns." + region + ".amazonaws.com");

        try {
            setupSQS();

            setupSNS();

            String jobId = initiateJobRequest();
            System.out.println("Jobid = " + jobId);

            long archiveSizeInBytes = waitForJobToComplete(jobId, sqsQueueURL);
            if (archiveSizeInBytes==-1) { throw new Exception("Job did not complete
successfully."); }

            downloadJobOutput(jobId, archiveSizeInBytes);

            cleanUp();
```

```
        } catch (Exception e) {
            System.err.println("Archive retrieval failed.");
            System.err.println(e);
        }
    }

    private static void setupSQS() {
        CreateQueueRequest request = new CreateQueueRequest()
            .withQueueName(sqsQueueName);
        CreateQueueResult result = sqsClient.createQueue(request);
        sqsQueueURL = result.getQueueUrl();

        GetQueueAttributesRequest qRequest = new GetQueueAttributesRequest()
            .withQueueUrl(sqsQueueURL)
            .withAttributeNames("QueueArn");

        GetQueueAttributesResult qResult = sqsClient.getQueueAttributes(qRequest);
        sqsQueueARN = qResult.getAttributes().get("QueueArn");

        Policy sqsPolicy =
            new Policy().withStatements(
                    new Statement(Effect.Allow)
                    .withPrincipals(Principal.AllUsers)
                    .withActions(SQSActions.SendMessage)
                    .withResources(new Resource(sqsQueueARN)));
        Map<String, String> queueAttributes = new HashMap<String, String>();
        queueAttributes.put("Policy", sqsPolicy.toJson());
        sqsClient.setQueueAttributes(new SetQueueAttributesRequest(sqsQueueURL,
queueAttributes));

    }
    private static void setupSNS() {
        CreateTopicRequest request = new CreateTopicRequest()
            .withName(snsTopicName);
        CreateTopicResult result = snsClient.createTopic(request);
        snsTopicARN = result.getTopicArn();

        SubscribeRequest request2 = new SubscribeRequest()
            .withTopicArn(snsTopicARN)
            .withEndpoint(sqsQueueARN)
            .withProtocol("sqs");
        SubscribeResult result2 = snsClient.subscribe(request2);
```

```
        snsSubscriptionARN = result2.getSubscriptionArn();
    }
    private static String initiateJobRequest() {

        JobParameters jobParameters = new JobParameters()
            .withType("archive-retrieval")
            .withArchiveId(archiveId)
            .withSNSTopic(snsTopicARN);

        InitiateJobRequest request = new InitiateJobRequest()
            .withVaultName(vaultName)
            .withJobParameters(jobParameters);

        InitiateJobResult response = client.initiateJob(request);

        return response.getJobId();
    }

    private static long waitForJobToComplete(String jobId, String sqsQueueUrl) throws
InterruptedException, JsonParseException, IOException {

        Boolean messageFound = false;
        Boolean jobSuccessful = false;
        long archiveSizeInBytes = -1;
        ObjectMapper mapper = new ObjectMapper();
        JsonFactory factory = mapper.getFactory();

        while (!messageFound) {
            List<Message> msgs = sqsClient.receiveMessage(
               new
ReceiveMessageRequest(sqsQueueUrl).withMaxNumberOfMessages(10)).getMessages();

            if (msgs.size() > 0) {
                for (Message m : msgs) {
                    JsonParser jpMessage = factory.createJsonParser(m.getBody());
                    JsonNode jobMessageNode = mapper.readTree(jpMessage);
                    String jobMessage = jobMessageNode.get("Message").textValue();

                    JsonParser jpDesc = factory.createJsonParser(jobMessage);
                    JsonNode jobDescNode = mapper.readTree(jpDesc);
                    String retrievedJobId = jobDescNode.get("JobId").textValue();
                    String statusCode = jobDescNode.get("StatusCode").textValue();
                    archiveSizeInBytes =
jobDescNode.get("ArchiveSizeInBytes").longValue();
```

```
                    if (retrievedJobId.equals(jobId)) {
                        messageFound = true;
                        if (statusCode.equals("Succeeded")) {
                            jobSuccessful = true;
                        }
                    }
                }

            } else {
              Thread.sleep(sleepTime * 1000);
            }
          }
        return (messageFound && jobSuccessful) ? archiveSizeInBytes : -1;
    }

    private static void downloadJobOutput(String jobId, long archiveSizeInBytes) throws
IOException {

        if (archiveSizeInBytes < 0) {
            System.err.println("Nothing to download.");
            return;
        }

        System.out.println("archiveSizeInBytes: " + archiveSizeInBytes);
        FileOutputStream fstream = new FileOutputStream(fileName);
        long startRange = 0;
        long endRange = (downloadChunkSize > archiveSizeInBytes) ? archiveSizeInBytes
-1 : downloadChunkSize - 1;

        do {

            GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
                .withVaultName(vaultName)
                .withRange("bytes=" + startRange + "-" + endRange)
                .withJobId(jobId);
            GetJobOutputResult getJobOutputResult =
client.getJobOutput(getJobOutputRequest);

            BufferedInputStream is = new
BufferedInputStream(getJobOutputResult.getBody());
            byte[] buffer = new byte[(int)(endRange - startRange + 1)];

            System.out.println("Checksum received: " +
getJobOutputResult.getChecksum());
```

```
            System.out.println("Content range " +
 getJobOutputResult.getContentRange());


            int totalRead = 0;
            while (totalRead < buffer.length) {
                int bytesRemaining = buffer.length - totalRead;
                int read = is.read(buffer, totalRead, bytesRemaining);
                if (read > 0) {
                    totalRead = totalRead + read;
                } else {
                    break;
                }

            }
            System.out.println("Calculated checksum: " +
 TreeHashGenerator.calculateTreeHash(new ByteArrayInputStream(buffer)));
            System.out.println("read = " + totalRead);
            fstream.write(buffer);

            startRange = startRange + (long)totalRead;
            endRange = ((endRange + downloadChunkSize) >  archiveSizeInBytes) ?
 archiveSizeInBytes : (endRange + downloadChunkSize);
            is.close();
        } while (endRange <= archiveSizeInBytes  && startRange < archiveSizeInBytes);

        fstream.close();
        System.out.println("Retrieved file to " + fileName);

    }

    private static void cleanUp() {
        snsClient.unsubscribe(new UnsubscribeRequest(snsSubscriptionARN));
        snsClient.deleteTopic(new DeleteTopicRequest(snsTopicARN));
        sqsClient.deleteQueue(new DeleteQueueRequest(sqsQueueURL));
    }
}
```

Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS SDK pour .NET

Le haut niveau et le bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode pour télécharger une archive.

Rubriques

- Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Téléchargement d'une archive à l'aide de l'API de bas niveau du AWS SDK pour .NET

Téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

La classe `ArchiveTransferManager` de l'API de haut niveau fournit la méthode `Download` que vous pouvez utiliser pour télécharger une archive.

> **Important**
>
> La classe `ArchiveTransferManager` crée une rubrique Amazon Simple Notification Service (Amazon SNS) et une file d'attente Amazon Simple Queue Service (Amazon SQS) qui est abonnée à cette rubrique. Elle lance ensuite la tâche d'extraction d'archive et interroge la file d'attente pour que l'archive soit disponible. Une fois que l'archive est disponible, le téléchargement commence. Pour plus d'informations sur les délais d'extraction, consultez Options de récupération des archives

Exemple : téléchargement d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

L'exemple de code C# suivant télécharge une archive à partir d'un coffre (`examplevault`) de la région USA Ouest (Oregon).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExemples de code en cours d'exécution. Vous devez mettre à jour le code avec un ID d'archive existant et le chemin d'accès au fichier local où vous souhaitez enregistrer l'archive téléchargée, comme illustré ci-après.

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
```

```
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDownloadHighLevel
  {
    static string vaultName        = "examplevault";
    static string archiveId        = "*** Provide archive ID ***";
    static string downloadFilePath = "*** Provide the file name and path to where to
 store the download ***";

    public static void Main(string[] args)
    {
      try
      {
        var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);

        var options = new DownloadOptions();
        options.StreamTransferProgress += ArchiveDownloadHighLevel.progress;
        // Download an archive.
        Console.WriteLine("Intiating the archive retrieval job and then polling SQS
 queue for the archive to be available.");
        Console.WriteLine("Once the archive is available, downloading will begin.");
        manager.Download(vaultName, archiveId, downloadFilePath, options);
        Console.WriteLine("To continue, press Enter");
        Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }

    static int currentPercentage = -1;
    static void progress(object sender, StreamTransferProgressArgs args)
    {
      if (args.PercentDone != currentPercentage)
      {
        currentPercentage = args.PercentDone;
        Console.WriteLine("Downloaded {0}%", args.PercentDone);
      }
    }
  }
```

```
}
```

Téléchargement d'une archive à l'aide de l'API de bas niveau du AWS SDK pour .NET

Voici les étapes à suivre pour télécharger une archive Amazon Glacier (Amazon Glacier) à l'aide de l'API de bas niveau du AWS SDK pour .NET.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier AWS la région à partir de laquelle vous souhaitez télécharger l'archive. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Lancez une tâche `archive-retrieval` en exécutant la méthode `InitiateJob`.

   Vous fournissez des informations sur la tâche, telles que l'ID d'archive de l'archive que vous souhaitez télécharger et la rubrique Amazon SNS facultative sur laquelle vous souhaitez qu'Amazon Glacier publie un message de fin de tâche, en créant une instance de la `InitiateJobRequest` classe. Amazon Glacier renvoie un numéro de tâche en réponse. La réponse est disponible dans une instance de la classe `InitiateJobResponse`.

   ```
   AmazonGlacierClient client;
   client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);

   InitiateJobRequest initJobRequest = new InitiateJobRequest()
   {
     VaultName = vaultName,
     JobParameters = new JobParameters()
     {
       Type      = "archive-retrieval",
       ArchiveId = "*** Provide archive id ***",
       SNSTopic  = "*** Provide Amazon SNS topic ARN ***",
     }
   };

   InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
   string jobId = initJobResponse.JobId;
   ```

   Vous pouvez éventuellement spécifier une plage d'octets pour demander à Amazon Glacier de préparer uniquement une partie de l'archive, comme indiqué dans la demande suivante. La demande indique à Amazon Glacier de préparer uniquement la partie de 1 Mo à 2 Mo de l'archive.

```
AmazonGlacierClient client;
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);


InitiateJobRequest initJobRequest = new InitiateJobRequest()
{
  VaultName = vaultName,
  JobParameters = new JobParameters()
  {
    Type      = "archive-retrieval",
    ArchiveId = "*** Provide archive id ***",
    SNSTopic  = "*** Provide Amazon SNS topic ARN ***",
  }
};
// Specify byte range.
int ONE_MEG = 1048576;
initJobRequest.JobParameters.RetrievalByteRange = string.Format("{0}-{1}", ONE_MEG, 2
 * ONE_MEG -1);

InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
string jobId = initJobResponse.JobId;
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification dans le coffre identifiant une rubrique Amazon Simple Notification Service (Amazon SNS) ou si vous avez spécifié une rubrique Amazon SNS lorsque vous avez lancé une tâche, Amazon Glacier envoie un message à cette rubrique une fois la tâche terminée. L'exemple de code présenté dans la section suivante utilise Amazon SNS pour qu'Amazon Glacier publie un message.

   Vous pouvez également interroger Amazon Glacier en appelant la `DescribeJob` méthode pour déterminer l'état d'achèvement de la tâche. Toutefois, l'approche recommandée consiste à utiliser une rubrique Amazon SNS pour la notification.

4. Téléchargez la sortie de la tâche (données archivées) en exécutant la méthode `GetJobOutput`.

   Vous indiquez les informations sur la demande, telles que l'ID de la tâche et le nom du coffre, en créant une instance de la classe `GetJobOutputRequest`. La sortie renvoyée par Amazon Glacier est disponible dans l'`GetJobOutputResponse`objet.

```
GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
{
  JobId = jobId,
  VaultName = vaultName
};

GetJobOutputResponse getJobOutputResponse = client.GetJobOutput(getJobOutputRequest);
using (Stream webStream = getJobOutputResponse.Body)
{
  using (Stream fileToSave = File.OpenWrite(fileName))
  {
     CopyStream(webStream, fileToSave);
  }
}
```

L'extrait de code précédent télécharge la totalité de la sortie de la tâche. Vous pouvez éventuellement extraire une partie de la sortie ou télécharger la totalité de la sortie en plusieurs parties en spécifiant la plage d'octets dans votre demande `GetJobOutputRequest`.

```
GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
{
  JobId = jobId,
  VaultName = vaultName
};
getJobOutputRequest.SetRange(0, 1048575); // Download only the first 1 MB chunk of
 the output.
```

En réponse à votre `GetJobOutput` appel, Amazon Glacier renvoie la somme de contrôle de la partie des données que vous avez téléchargée, si certaines conditions sont remplies. Pour de plus amples informations, veuillez consulter Réception des totaux de contrôle lors du téléchargement de données.

Pour vérifier que le téléchargement ne comporte aucune erreur, vous pouvez ensuite calculer la somme de contrôle côté client et la comparer à la somme de contrôle envoyée par Amazon Glacier dans la réponse.

Pour une tâche de récupération d'archives dont la plage facultative est spécifiée, lorsque vous obtenez la description de la tâche, celle-ci inclut la somme de contrôle de la plage que vous récupérez (SHA256TreeHash). Vous pouvez utiliser cette valeur pour vérifier davantage

l'exactitude de l'ensemble de la plage d'octets que vous téléchargerez ultérieurement. Par exemple, si vous lancez une tâche pour extraire une plage d'une archive alignée sur le hachage d'arborescence, que vous téléchargez la sortie en plusieurs parties et que chacune de vos demandes `GetJobOutput` renvoie un total de contrôle, vous pouvez ensuite calculer le total de contrôle de chaque partie téléchargée côté client, puis calculer le hachage d'arborescence. Vous pouvez le comparer à la somme de contrôle renvoyée par Amazon Glacier en réponse à votre demande de travail de description afin de vérifier que l'ensemble de la plage d'octets que vous avez téléchargée est identique à la plage d'octets stockée dans Amazon Glacier.

Pour obtenir un exemple pratique, consultez Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour .NET—Download in Chunks.

Exemple 1 : extraction d'une archive à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'exemple de code C# suivant télécharge une archive à partir du coffre spécifié. Une fois la tâche terminée, l'exemple télécharge la totalité de la sortie dans un seul appel `GetJobOutput`. Pour obtenir un exemple de téléchargement d'une sortie en plusieurs parties, consultez la section Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour .NET—Download in Chunks.

L'exemple exécute les tâches suivantes :

- Configuration d'une rubrique Amazon Simple Notification Service (Amazon SNS).

  Amazon Glacier envoie une notification à cette rubrique une fois le travail terminé.
- Configuration d'une file d'attente Amazon Simple Queue Service (Amazon SQS).

  L'exemple attache une politique à la file d'attente pour permettre à la rubrique Amazon SNS de publier des messages.
- Lancement d'une tâche de téléchargement de l'archive spécifiée.

  Dans la demande de travail, l'exemple indique le sujet Amazon SNS afin qu'Amazon Glacier puisse envoyer un message une fois le travail terminé.
- Recherche périodique d'un message dans la file d'attente Amazon SQS.

  Si un message est présent, l'exemple analyse le JSON et vérifie si la tâche s'est correctement déroulée. Si tel est le cas, il télécharge l'archive. L'exemple de code utilise la bibliothèque JSON.NET (consultez le document JSON.NET) pour analyser le JSON.

- Nettoyage par la suppression de la rubrique Amazon SNS et de la file d'attente Amazon SQS qu'il a créées.

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Threading;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Amazon.SQS;
using Amazon.SQS.Model;
using Newtonsoft.Json;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDownloadLowLevelUsingSNSSQS
  {
    static string topicArn;
    static string queueUrl;
    static string queueArn;
    static string vaultName = "*** Provide vault name ***";
    static string archiveID = "*** Provide archive ID ***";
    static string fileName  = "*** Provide the file name and path to where to store
 downloaded archive ***";
    static AmazonSimpleNotificationServiceClient snsClient;
    static AmazonSQSClient sqsClient;
    const string SQS_POLICY =
        "{" +
        "    \"Version\" : \"2012-10-17\",&TCX5-2025-waiver;" +
        "    \"Statement\" : [" +
        "        {" +
        "            \"Sid\" : \"sns-rule\"," +
        "            \"Effect\" : \"Allow\"," +
        "            \"Principal\" : {\"Service\" : \"sns.amazonaws.com\" }," +
        "            \"Action\"    : \"sqs:SendMessage\"," +
        "            \"Resource\"  : \"{QueueArn}\"," +
        "            \"Condition\" : {" +
        "                \"ArnLike\" : {" +
        "                    \"aws:SourceArn\" : \"{TopicArn}\"" +
```

```
        "                }" +
        "            }" +
        "        }" +
        "    ]" +
        "}";

    public static void Main(string[] args)
    {
      AmazonGlacierClient client;
      try
      {
        using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Setup SNS topic and SQS queue.");
          SetupTopicAndQueue();
          Console.WriteLine("To continue, press Enter"); Console.ReadKey();
          Console.WriteLine("Retrieving...");
          RetrieveArchive(client);
        }
        Console.WriteLine("Operations successful. To continue, press Enter");
        Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      finally
      {
        // Delete SNS topic and SQS queue.
        snsClient.DeleteTopic(new DeleteTopicRequest() { TopicArn = topicArn });
        sqsClient.DeleteQueue(new DeleteQueueRequest() { QueueUrl = queueUrl });
      }
    }

    static void SetupTopicAndQueue()
    {
      snsClient = new
AmazonSimpleNotificationServiceClient(Amazon.RegionEndpoint.USWest2);
      sqsClient = new AmazonSQSClient(Amazon.RegionEndpoint.USWest2);

      long ticks = DateTime.Now.Ticks;
      topicArn = snsClient.CreateTopic(new CreateTopicRequest { Name =
"GlacierDownload-" + ticks }).TopicArn;
      Console.Write("topicArn: "); Console.WriteLine(topicArn);
```

```
      CreateQueueRequest createQueueRequest = new CreateQueueRequest();
      createQueueRequest.QueueName = "GlacierDownload-" + ticks;
      CreateQueueResponse createQueueResponse =
sqsClient.CreateQueue(createQueueRequest);
      queueUrl = createQueueResponse.QueueUrl;
      Console.Write("QueueURL: "); Console.WriteLine(queueUrl);

      GetQueueAttributesRequest getQueueAttributesRequest = new
GetQueueAttributesRequest();
      getQueueAttributesRequest.AttributeNames = new List<string> { "QueueArn" };
      getQueueAttributesRequest.QueueUrl = queueUrl;
      GetQueueAttributesResponse response =
sqsClient.GetQueueAttributes(getQueueAttributesRequest);
      queueArn = response.QueueARN;
      Console.Write("QueueArn: "); Console.WriteLine(queueArn);

      // Setup the Amazon SNS topic to publish to the SQS queue.
      snsClient.Subscribe(new SubscribeRequest()
      {
        Protocol = "sqs",
        Endpoint = queueArn,
        TopicArn = topicArn
      });

      // Add policy to the queue so SNS can send messages to the queue.
      var policy = SQS_POLICY.Replace("{TopicArn}", topicArn).Replace("{QueueArn}",
queueArn);

      sqsClient.SetQueueAttributes(new SetQueueAttributesRequest()
      {
          QueueUrl = queueUrl,
          Attributes = new Dictionary<string, string>
          {
              { QueueAttributeName.Policy, policy }
          }
      });
   }

   static void RetrieveArchive(AmazonGlacierClient client)
   {
      // Initiate job.
      InitiateJobRequest initJobRequest = new InitiateJobRequest()
      {
        VaultName = vaultName,
```

```
        JobParameters = new JobParameters()
        {
          Type = "archive-retrieval",
          ArchiveId = archiveID,
          Description = "This job is to download archive.",
          SNSTopic = topicArn,
        }
      };
      InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
      string jobId = initJobResponse.JobId;

      // Check queue for a message and if job completed successfully, download archive.
      ProcessQueue(jobId, client);
    }

    private static void ProcessQueue(string jobId, AmazonGlacierClient client)
    {
      ReceiveMessageRequest receiveMessageRequest = new ReceiveMessageRequest()
{ QueueUrl = queueUrl, MaxNumberOfMessages = 1 };
      bool jobDone = false;
      while (!jobDone)
      {
        Console.WriteLine("Poll SQS queue");
        ReceiveMessageResponse receiveMessageResponse =
sqsClient.ReceiveMessage(receiveMessageRequest);
        if (receiveMessageResponse.Messages.Count == 0)
        {
          Thread.Sleep(10000 * 60);
          continue;
        }
        Console.WriteLine("Got message");
        Message message = receiveMessageResponse.Messages[0];
        Dictionary<string, string> outerLayer =
JsonConvert.DeserializeObject<Dictionary<string, string>>(message.Body);
        Dictionary<string, object> fields =
JsonConvert.DeserializeObject<Dictionary<string, object>>(outerLayer["Message"]);
        string statusCode = fields["StatusCode"] as string;

        if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_SUCCEEDED,
StringComparison.InvariantCultureIgnoreCase))
        {
          Console.WriteLine("Downloading job output");
          DownloadOutput(jobId, client); // Save job output to the specified file
location.
```

```
        }
        else if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_FAILED,
 StringComparison.InvariantCultureIgnoreCase))
          Console.WriteLine("Job failed... cannot download the archive.");

        jobDone = true;
        sqsClient.DeleteMessage(new DeleteMessageRequest() { QueueUrl = queueUrl,
 ReceiptHandle = message.ReceiptHandle });
      }
    }

    private static void DownloadOutput(string jobId, AmazonGlacierClient client)
    {
      GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
      {
        JobId = jobId,
        VaultName = vaultName
      };

      GetJobOutputResponse getJobOutputResponse =
 client.GetJobOutput(getJobOutputRequest);
      using (Stream webStream = getJobOutputResponse.Body)
      {
          using (Stream fileToSave = File.OpenWrite(fileName))
          {
              CopyStream(webStream, fileToSave);
          }
      }
    }

    public static void CopyStream(Stream input, Stream output)
    {
      byte[] buffer = new byte[65536];
      int length;
      while ((length = input.Read(buffer, 0, buffer.Length)) > 0)
      {
        output.Write(buffer, 0, length);
      }
    }
  }
}
```

Exemple 2 : extraction d'une archive à l'aide de l'API de bas niveau de la sortie AWS SDK pour .NET —Download in Chunks

L'exemple de code C# suivant extrait une archive d'Amazon Glacier. L'exemple de code télécharge la sortie de la tâche en plusieurs parties en spécifiant la plage d'octets dans un objet `GetJobOutputRequest`.

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Threading;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Amazon.SQS;
using Amazon.SQS.Model;
using Newtonsoft.Json;
using System.Collections.Specialized;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDownloadLowLevelUsingSQLSNSOutputUsingRange
  {
    static string topicArn;
    static string queueUrl;
    static string queueArn;
    static string vaultName = "*** Provide vault name ***";
    static string archiveId = "*** Provide archive ID ***";
    static string fileName  = "*** Provide the file name and path to where to store
 downloaded archive ***";
    static AmazonSimpleNotificationServiceClient snsClient;
    static AmazonSQSClient sqsClient;
    const string SQS_POLICY =
        "{" +
        "    \"Version\" : \"2012-10-17\",&TCX5-2025-waiver;" +
        "    \"Statement\" : [" +
        "        {" +
        "            \"Sid\" : \"sns-rule\"," +
        "            \"Effect\" : \"Allow\"," +
```

```
        "            \"Principal\" : {\"AWS\" : \"arn:aws:iam::123456789012:root\" },"
+
        "            \"Action\"  : \"sqs:SendMessage\"," +
        "            \"Resource\"  : \"{QuernArn}\"," +
        "            \"Condition\" : {" +
        "                \"ArnLike\" : {" +
        "                    \"aws:SourceArn\" : \"{TopicArn}\"" +
        "                }" +
        "            }" +
        "        }" +
        "    ]" +
        "}";

    public static void Main(string[] args)
    {
      AmazonGlacierClient client;

      try
      {
          using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
          {
              Console.WriteLine("Setup SNS topic and SQS queue.");
              SetupTopicAndQueue();
              Console.WriteLine("To continue, press Enter"); Console.ReadKey();

              Console.WriteLine("Download archive");
              DownloadAnArchive(archiveId, client);
        }
        Console.WriteLine("Operations successful. To continue, press Enter");
        Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      finally
      {
        // Delete SNS topic and SQS queue.
        snsClient.DeleteTopic(new DeleteTopicRequest() { TopicArn = topicArn });
        sqsClient.DeleteQueue(new DeleteQueueRequest() { QueueUrl = queueUrl });
      }
    }

       static void SetupTopicAndQueue()
    {
```

```
      long ticks = DateTime.Now.Ticks;

      // Setup SNS topic.
      snsClient = new
AmazonSimpleNotificationServiceClient(Amazon.RegionEndpoint.USWest2);
      sqsClient = new AmazonSQSClient(Amazon.RegionEndpoint.USWest2);

      topicArn = snsClient.CreateTopic(new CreateTopicRequest { Name =
"GlacierDownload-" + ticks }).TopicArn;
      Console.Write("topicArn: "); Console.WriteLine(topicArn);

      CreateQueueRequest createQueueRequest = new CreateQueueRequest();
      createQueueRequest.QueueName = "GlacierDownload-" + ticks;
      CreateQueueResponse createQueueResponse =
sqsClient.CreateQueue(createQueueRequest);
      queueUrl = createQueueResponse.QueueUrl;
      Console.Write("QueueURL: "); Console.WriteLine(queueUrl);

      GetQueueAttributesRequest getQueueAttributesRequest = new
GetQueueAttributesRequest();
      getQueueAttributesRequest.AttributeNames = new List<string> { "QueueArn" };
      getQueueAttributesRequest.QueueUrl = queueUrl;
      GetQueueAttributesResponse response =
sqsClient.GetQueueAttributes(getQueueAttributesRequest);
      queueArn = response.QueueARN;
      Console.Write("QueueArn: "); Console.WriteLine(queueArn);

      // Setup the Amazon SNS topic to publish to the SQS queue.
      snsClient.Subscribe(new SubscribeRequest()
      {
        Protocol = "sqs",
        Endpoint = queueArn,
        TopicArn = topicArn
      });

      // Add the policy to the queue so SNS can send messages to the queue.
      var policy = SQS_POLICY.Replace("{TopicArn}", topicArn).Replace("{QuernArn}",
queueArn);

      sqsClient.SetQueueAttributes(new SetQueueAttributesRequest()
      {
          QueueUrl = queueUrl,
          Attributes = new Dictionary<string, string>
          {
```

```
            { QueueAttributeName.Policy, policy }
        }
    });
  }

  static void DownloadAnArchive(string archiveId, AmazonGlacierClient client)
  {
    // Initiate job.
    InitiateJobRequest initJobRequest = new InitiateJobRequest()
    {

      VaultName = vaultName,
      JobParameters = new JobParameters()
      {
        Type = "archive-retrieval",
        ArchiveId = archiveId,
        Description = "This job is to download the archive.",
        SNSTopic = topicArn,
      }
    };
    InitiateJobResponse initJobResponse = client.InitiateJob(initJobRequest);
    string jobId = initJobResponse.JobId;

    // Check queue for a message and if job completed successfully, download archive.
    ProcessQueue(jobId, client);
  }

  private static void ProcessQueue(string jobId, AmazonGlacierClient client)
  {
      var receiveMessageRequest = new ReceiveMessageRequest() { QueueUrl = queueUrl,
MaxNumberOfMessages = 1 };
      bool jobDone = false;
      while (!jobDone)
      {
          Console.WriteLine("Poll SQS queue");
          ReceiveMessageResponse receiveMessageResponse =
sqsClient.ReceiveMessage(receiveMessageRequest);
          if (receiveMessageResponse.Messages.Count == 0)
          {
              Thread.Sleep(10000 * 60);
              continue;
          }
          Console.WriteLine("Got message");
          Message message = receiveMessageResponse.Messages[0];
```

```
            Dictionary<string, string> outerLayer =
JsonConvert.DeserializeObject<Dictionary<string, string>>(message.Body);
            Dictionary<string, object> fields =
JsonConvert.DeserializeObject<Dictionary<string, object>>(outerLayer["Message"]);
            string statusCode = fields["StatusCode"] as string;
            if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_SUCCEEDED,
StringComparison.InvariantCultureIgnoreCase))
            {
                long archiveSize = Convert.ToInt64(fields["ArchiveSizeInBytes"]);
                Console.WriteLine("Downloading job output");
                DownloadOutput(jobId, archiveSize, client); // This where we save job
output to the specified file location.
            }
            else if (string.Equals(statusCode, GlacierUtils.JOB_STATUS_FAILED,
StringComparison.InvariantCultureIgnoreCase))
                Console.WriteLine("Job failed... cannot download the archive.");
            jobDone = true;
            sqsClient.DeleteMessage(new DeleteMessageRequest() { QueueUrl = queueUrl,
ReceiptHandle = message.ReceiptHandle });
        }
    }

    private static void DownloadOutput(string jobId, long archiveSize,
AmazonGlacierClient client)
    {
      long partSize = 4 * (long)Math.Pow(2, 20);  // 4 MB.
      using (Stream fileToSave = new FileStream(fileName, FileMode.Create,
FileAccess.Write))
      {

        long currentPosition = 0;
        do
        {
          GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
          {
            JobId = jobId,
            VaultName = vaultName
          };

          long endPosition = currentPosition + partSize - 1;
          if (endPosition > archiveSize)
            endPosition = archiveSize;

          getJobOutputRequest.SetRange(currentPosition, endPosition);
```

```
          GetJobOutputResponse getJobOutputResponse =
  client.GetJobOutput(getJobOutputRequest);

          using (Stream webStream = getJobOutputResponse.Body)
          {
            CopyStream(webStream, fileToSave);
          }
          currentPosition += partSize;
        } while (currentPosition < archiveSize);
      }
    }

    public static void CopyStream(Stream input, Stream output)
    {
      byte[] buffer = new byte[65536];
      int length;
      while ((length = input.Read(buffer, 0, buffer.Length)) > 0)
      {
        output.Write(buffer, 0, length);
      }
    }
  }
}
```

Téléchargement d'une archive volumineuse à l'aide du traitement parallèle avec Python

Cette rubrique explique comment télécharger une archive volumineuse depuis Amazon S3 Glacier (S3 Glacier) à l'aide du traitement parallèle avec Python. Cette approche vous permet de télécharger de manière fiable des archives de toutes tailles en les divisant en petits morceaux pouvant être traités indépendamment.

Présentation de

Le script Python fourni dans cet exemple exécute les tâches suivantes :

1. Configure les AWS ressources nécessaires (rubrique Amazon SNS et files d'attente Amazon SQS) pour les notifications
2. Lance une tâche de récupération d'archives avec Amazon Glacier
3. Surveille une file d'attente Amazon SQS pour détecter les notifications de fin de tâche
4. Divise la grande archive en morceaux faciles à gérer

5. Télécharge des segments en parallèle à l'aide de plusieurs threads de travail
6. Enregistre chaque morceau sur le disque pour un réassemblage ultérieur

Conditions préalables

Avant de commencer, assurez-vous d'avoir :

- Python 3.6 ou version ultérieure installé
- AWS SDK pour Python (Boto3) installé
- AWS informations d'identification configurées avec les autorisations appropriées pour Amazon Glacier, Amazon SNS et Amazon SQS
- Espace disque suffisant pour stocker les fragments d'archive téléchargés

Exemple : téléchargement d'une archive à l'aide du traitement parallèle avec Python

Le script Python suivant montre comment télécharger une archive volumineuse depuis Amazon Glacier à l'aide du traitement parallèle :

```
import boto3
import time
import json
import jmespath
import re
import concurrent.futures
import os

output_file_path = "output_directory_path"
vault_name = "vault_name"

chunk_size = 1000000000 #1gb - size of chunks for parallel download.
notify_queue_name = 'GlacierJobCompleteNotifyQueue' # SQS queue for Glacier recall
 notification
chunk_download_queue_name='GlacierChunkReadyNotifyQueue' # SQS queue for chunks
sns_topic_name = 'GlacierRecallJobCompleted' # the SNS topic to be notified when
 Glacier archive is restored.
chunk_queue_visibility_timeout = 7200 # 2 hours - this may need to be adjusted.
region = 'us-east-1'
archive_id = "archive_id_to_restore"
```

```
retrieve_archive = True # set to false if you do not want to restore from Glacier -
 useful for restarting or parallel processing of the chunk queue.
workers = 12 # the number of parallel worker threads for downloading chunks.

def setup_queues_and_topic():
    sqs = boto3.client('sqs')
    sns = boto3.client('sns')

    # Create the SNS topic
    topic_response = sns.create_topic(
        Name=sns_topic_name
    )
    topic_arn = topic_response['TopicArn']
    print("Creating the SNS topic " + topic_arn)

    # Create the notification queue
    notify_queue_response = sqs.create_queue(
        QueueName=notify_queue_name,
        Attributes={
            'VisibilityTimeout': '300',  # 5 minutes
            'ReceiveMessageWaitTimeSeconds': '20'  # Enable long polling
        }
    )
    notify_queue_url = notify_queue_response['QueueUrl']
    print("Creating the archive-retrieval notification queue " + notify_queue_url)

    # Create the chunk download queue
    chunk_queue_response = sqs.create_queue(
        QueueName=chunk_download_queue_name,
        Attributes={
            'VisibilityTimeout': str(chunk_queue_visibility_timeout),  # 5 minutes
            'ReceiveMessageWaitTimeSeconds': '0'
        }
    )
    chunk_queue_url = chunk_queue_response['QueueUrl']

    print("Creating the chunk ready notification queue " + chunk_queue_url)


   # Get the ARN for the notification queue
    notify_queue_attributes = sqs.get_queue_attributes(
        QueueUrl=notify_queue_url,
        AttributeNames=['QueueArn']
    )
```

```
    notify_queue_arn = notify_queue_attributes['Attributes']['QueueArn']

    # Set up the SNS topic policy on the notification queue
    queue_policy = {
        "Version": "2012-10-17",
        "Statement": [{
            "Sid": "allow-sns-messages",
            "Effect": "Allow",
            "Principal": {"AWS": "*"},
            "Action": "SQS:SendMessage",
            "Resource": notify_queue_arn,
            "Condition": {
                "ArnEquals": {
                    "aws:SourceArn": topic_arn
                }
            }
        }]
    }

    # Set the queue policy
    sqs.set_queue_attributes(
        QueueUrl=notify_queue_url,
        Attributes={
            'Policy': json.dumps(queue_policy)
        }
    )

    # Subscribe the notification queue to the SNS topic
    sns.subscribe(
        TopicArn=topic_arn,
        Protocol='sqs',
        Endpoint=notify_queue_arn
    )

    return {
        'topic_arn': topic_arn,
        'notify_queue_url': notify_queue_url,
        'chunk_queue_url': chunk_queue_url
    }


def split_and_send_chunks(archive_size, job_id,chunk_queue_url):
    ranges = []
    current = 0
```

```
    chunk_number = 0

    while current < archive_size:
        chunk_number += 1
        next_range = min(current + chunk_size - 1, archive_size - 1)
        ranges.append((current, next_range, chunk_number))
        current = next_range + 1

    # Send messages to SQS queue
    for start, end, chunk_number in ranges:
        body = {"start": start, "end": end, "job_id": job_id, "chunk_number":
 chunk_number}
        body = json.dumps(body)
        print("Sending SQS message for range:" + str(body))
        response = sqs.send_message(
            QueueUrl=chunk_queue_url,
            MessageBody=str(body)
        )

def GetJobOutputChunks(job_id, byterange, chunk_number):
    glacier = boto3.client('glacier')
    response = glacier.get_job_output(
        vaultName=vault_name,
        jobId=job_id,
        range=byterange,

    )

    with open(os.path.join(output_file_path,str(chunk_number)+".chunk"), 'wb') as
 output_file:
        output_file.write(response['body'].read())

    return response

def ReceiveArchiveReadyMessages(notify_queue_url,chunk_queue_url):

    response = sqs.receive_message(
        QueueUrl=notify_queue_url,
        AttributeNames=['All'],
        MaxNumberOfMessages=1,
        WaitTimeSeconds=20,
        MessageAttributeNames=['Message']
    )
    print("Polling archive retrieval job ready queue...")
```

```
    # Checking that there is a Messages key before proceeding. No 'Messages' key likely
 means the queue is empty

    if 'Messages' in response:
        print("Received a message from the archive retrieval job queue")
        jsonresponse = response
        # Loading the string into JSON and checking that ArchiveSizeInBytes key is
 present before continuing.
        jsonresponse=json.loads(jsonresponse['Messages'][0]['Body'])
        jsonresponse=json.loads(jsonresponse['Message'])
        if 'ArchiveSizeInBytes' in jsonresponse:
            receipt_handle = response['Messages'][0]['ReceiptHandle']
            if jsonresponse['ArchiveSizeInBytes']:
                archive_size = jsonresponse['ArchiveSizeInBytes']

                print(f'Received message: {response}')
                if archive_size > chunk_size:
                    split_and_send_chunks(archive_size,
 jsonresponse['JobId'],chunk_queue_url)

                    sqs.delete_message(
                    QueueUrl=notify_queue_url,
                    ReceiptHandle=receipt_handle)

            else:
                print("No ArchiveSizeInBytes value found in message")
                print(response)

    else:
        print('No messages available in the queue at this time.')

    time.sleep(1)

def ReceiveArchiveChunkMessages(chunk_queue_url):
    response = sqs.receive_message(
        QueueUrl=chunk_queue_url,
        AttributeNames=['All'],
        MaxNumberOfMessages=1,
        WaitTimeSeconds=0,
        MessageAttributeNames=['Message']
    )
    print("Polling archive chunk queue...")
    print(response)
```

```
    # Checking that there is a Messages key before proceeding. No 'Messages' key likely
 means the queue is empty
    if 'Messages' in response:
        jsonresponse = response
        # Loading the string into JSON and checking that ArchiveSizeInBytes key is
 present before continuing.
        jsonresponse=json.loads(jsonresponse['Messages'][0]['Body'])
        if 'job_id' in jsonresponse: #checking that there is a job id before continuing
            job_id = jsonresponse['job_id']
            byterange = "bytes="+str(jsonresponse['start']) + '-' +
 str(jsonresponse['end'])
            chunk_number = jsonresponse['chunk_number']
            receipt_handle = response['Messages'][0]['ReceiptHandle']
            if jsonresponse['job_id']:
                print(f'Received message: {response}')
                GetJobOutputChunks(job_id,byterange,chunk_number)
                sqs.delete_message(
                QueueUrl=chunk_queue_url,
                ReceiptHandle=receipt_handle)
    else:
        print('No messages available in the chunk queue at this time.')

def initiate_archive_retrieval(archive_id, topic_arn):
    glacier = boto3.client('glacier')

    job_parameters = {
        "Type": "archive-retrieval",
        "ArchiveId": archive_id,
        "Description": "Archive retrieval job",
        "SNSTopic": topic_arn,
        "Tier": "Bulk"  # You can change this to "Standard" or "Expedited" based on
 your needs
    }

    try:
        response = glacier.initiate_job(
            vaultName=vault_name,
            jobParameters=job_parameters
        )

        print("Archive retrieval job initiated:")
        print(f"Job ID: {response['jobId']}")
        print(f"Job parameters: {job_parameters}")
        print(f"Complete response: {json.dumps(response, indent=2)}")
```

```
        return response['jobId']

    except Exception as e:
        print(f"Error initiating archive retrieval job: {str(e)}")
        raise

def run_async_tasks(chunk_queue_url, workers):
    max_workers = workers  # Set the desired maximum number of concurrent tasks
    with concurrent.futures.ThreadPoolExecutor(max_workers=max_workers) as executor:
        for _ in range(max_workers):
            executor.submit(ReceiveArchiveChunkMessages, chunk_queue_url)

# One time setup of the necessary queues and topics.
queue_and_topic_atts = setup_queues_and_topic()

topic_arn = queue_and_topic_atts['topic_arn']
notify_queue_url = queue_and_topic_atts['notify_queue_url']
chunk_queue_url = queue_and_topic_atts['chunk_queue_url']

if retrieve_archive:
    print("Retrieving the defined archive... The topic arn we will notify when
 recalling the archive is: "+topic_arn)
    job_id = initiate_archive_retrieval(archive_id, topic_arn)
else:
    print("Retrieve archive is false, polling queues and downloading only.")

while True:
   ReceiveArchiveReadyMessages(notify_queue_url,chunk_queue_url)
   run_async_tasks(chunk_queue_url,workers)
```

Utilisation du script

Pour utiliser ce script, procédez comme suit :

1. Remplacez les valeurs d'espace réservé dans le script par vos informations spécifiques :
   - *output_file_path*: répertoire dans lequel les fichiers partiels seront enregistrés
   - *vault_name*: nom de votre coffre-fort S3 Glacier
   - *notify_queue_name*: nom de la file d'attente de notifications de tâches
   - *chunk_download_queue_name*: nom de la file d'attente de téléchargement des tronçons

- *sns_topic_name*: nom de la rubrique SNS
- *region*: AWS région dans laquelle se trouve votre coffre-fort
- *archive_id*: ID de l'archive à récupérer

2. Exécutez le script :

```
python download_large_archive.py
```

3. Une fois tous les fragments téléchargés, vous pouvez les combiner dans un seul fichier à l'aide d'une commande telle que :

```
cat /path/to/chunks/*.chunk > complete_archive.file
```

Importantes considérations

Lorsque vous utilisez ce script, tenez compte des points suivants :

- L'extraction des archives depuis S3 Glacier peut prendre plusieurs heures, selon le niveau de récupération sélectionné.
- Le script s'exécute indéfiniment, interrogeant continuellement les files d'attente. Vous souhaiterez peut-être ajouter une condition de résiliation en fonction de vos besoins spécifiques.
- Assurez-vous de disposer de suffisamment d'espace disque pour stocker tous les fragments de votre archive.
- Si le script est interrompu, vous pouvez le redémarrer `retrieve_archive=False` pour continuer à télécharger des fragments sans lancer une nouvelle tâche de récupération.
- Ajustez les *workers* paramètres *chunk_size* et en fonction de la bande passante de votre réseau et des ressources système.
- AWS Les frais standard s'appliquent pour les extractions Amazon S3, Amazon SNS et l'utilisation d'Amazon SQS.

Téléchargement d'une archive à l'aide de l'API REST

Pour télécharger une archive à l'aide de l'API REST

Le téléchargement d'une archive est un processus en deux étapes.

1. Lancez une tâche de type `archive-retrieval`. Pour de plus amples informations, veuillez consulter Lancement d'une tâche (POST jobs).
2. Une fois la tâche terminée, téléchargez les données de l'archive. Pour de plus amples informations, veuillez consulter Génération de sortie de tâche (GET output).

Téléchargement d'une archive dans Amazon Glacier à l'aide du AWS CLI

Vous pouvez télécharger des archives dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : télécharger une archive à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l'interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l'utilisateur de l'interface AWS Command Line Interface :

   Installation du AWS Command Line Interface

   Configuration du AWS Command Line Interface
2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

```
aws configure list
```

Exemple : télécharger une archive à l'aide du AWS CLI

> **Note**
>
> Pour télécharger vos archives, vous devez connaître leur ID. Suivez les étapes 1 à 4 pour récupérer les ID de vos archives. Si vous connaissez déjà les ID des archives que vous souhaitez télécharger, passez à l'étape 5.

1. Utilisez la commande `initiate-job` pour démarrer une tâche d'extraction d'inventaire. Le rapport d'inventaire répertorie les ID de vos archives.

   ```
   aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
   job-parameters="{\"Type\":\"inventory-retrieval\"}"
   ```

   Sortie attendue :

   ```
   {
       "location": "/111122223333/vaults/awsexamplevault/jobs/*** jobid ***",
       "jobId": "*** jobid ***"
   }
   ```

2. Utilisez la commande `describe-job` pour vérifier le statut de la précédente commande de tâche .

   ```
   aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
   job-id *** jobid ***
   ```

   Sortie attendue :

   ```
   {
       "InventoryRetrievalParameters": {
           "Format": "JSON"
   ```

```
    },
    "VaultARN": "*** vault arn ***",
    "Completed": false,
    "JobId": "*** jobid ***",
    "Action": "InventoryRetrieval",
    "CreationDate": "*** job creation date ***",
    "StatusCode": "InProgress"
}
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification sur le coffre-fort ou si vous avez spécifié une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

   Vous pouvez définir la configuration des notifications pour des événements spécifiques concernant le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier. Amazon Glacier envoie un message à la rubrique SNS spécifiée chaque fois que l'événement spécifique se produit.

4. Une fois l'opération terminée, utilisez la commande `get-job-output` pour télécharger le travail d'extraction dans le fichier `output.json`. Ce fichier contient les ID de vos archives.

   ```
   aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
    --job-id *** jobid *** output.json
   ```

   Cette commande produit un fichier avec les champs suivants.

   ```
   {
   "VaultARN":"arn:aws:glacier:region:111122223333:vaults/awsexamplevault",
   "InventoryDate":"*** job completion date ***",
   "ArchiveList":[
   {"ArchiveId":"*** archiveid ***",
   "ArchiveDescription":*** archive description (if set) ***,
   "CreationDate":"*** archive creation date ***",
   "Size":"*** archive size (in bytes) ***",
   "SHA256TreeHash":"*** archive hash ***"
   }
   {"ArchiveId":
   ```

```
...
]}
```

5. Utilisez la commande `initiate-job` pour démarrer le processus d'extraction de chaque archive d'un coffre. Vous devez attribuer au paramètre de tâche la valeur `archive-retrieval` comme indiqué ci-dessous.

```
aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333
 --job-parameters="{\"Type\":\"archive-retrieval\",\"ArchiveId\":\"*** archiveId
 ***\"}"
```

6. Attendez que la tâche `archive-retrieval` se termine. Utilisez la commande `describe-job` pour vérifier le statut de la commande précédente.

```
aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
job-id *** jobid ***
```

7. Une fois la tâche ci-dessus terminée, utilisez la commande `get-job-output` pour télécharger l'archive.

```
aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
 --job-id *** jobid *** output_file_name
```

Supprimer une archive dans Amazon Glacier

Vous ne pouvez pas supprimer une archive à l'aide de la console de gestion Amazon Glacier (Amazon Glacier). Pour supprimer une archive, vous devez utiliser la AWS Command Line Interface (CLI) ou écrire du code pour effectuer une demande de suppression en utilisant directement l'API REST ou les bibliothèques wrapper AWS SDK pour Java et .NET. Les rubriques suivantes expliquent comment utiliser les bibliothèques wrapper AWS SDK pour Java et .NET, l'API REST et le AWS CLI.

Rubriques

- Suppression d'une archive dans Amazon Glacier à l'aide du AWS SDK pour Java
- Suppression d'une archive dans Amazon Glacier à l'aide du AWS SDK pour .NET
- Suppression d'une archive Amazon Glacier à l'aide de l'API REST
- Suppression d'une archive dans Amazon Glacier à l'aide du AWS Command Line Interface

Vous pouvez supprimer une archive à la fois à partir d'un coffre. Pour supprimer l'archive, vous devez fournir son ID d'archive dans votre demande de suppression. Vous pouvez obtenir l'ID de l'archive en téléchargeant l'inventaire du coffre qui contient l'archive. Pour plus d'informations sur le téléchargement de l'inventaire de coffre, consultez la page Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

Une fois que vous avez supprimé une archive, vous pouvez toujours faire une demande de lancement d'une tâche afin d'extraire l'archive supprimée, mais la tâche d'extraction de l'archive échouera.

Les extractions d'archives en cours pour un ID d'archivage lorsque vous supprimez l'archive peuvent ou non selon les scénarios suivants :

- Si la tâche de récupération des archives prépare activement les données pour le téléchargement lorsqu'Amazon Glacier reçoit la demande de suppression de l'archive, l'opération de récupération des archives risque d'échouer.
- Si la tâche de récupération de l'archive a correctement préparé l'archive pour le téléchargement lorsqu'Amazon Glacier reçoit la demande de suppression de l'archive, vous pourrez télécharger le résultat.

Pour plus d'informations sur l'extraction de l'archive, consultez la section Téléchargement d'une archive dans Amazon Glacier.

Cette opération est idempotente. La suppression d'une archive déjà supprimée n'entraîne pas d'erreur.

Une fois que vous avez supprimé une archive, si vous téléchargez immédiatement l'inventaire du coffre, l'archive supprimée peut être incluse dans la liste, car Amazon Glacier prépare l'inventaire du coffre-fort une fois par jour uniquement.

> **Note**
>
> Pour la suppression automatique des archives du coffre-fort, consultez Suppression automatique des archives du coffre-fort dans Amazon S3 Glacier.

Suppression d'une archive dans Amazon Glacier à l'aide du AWS SDK pour Java

Voici les étapes à suivre pour supprimer une archive à l'aide de l'API de AWS SDK pour Java bas niveau.

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier une AWS région dans laquelle l'archive que vous souhaitez supprimer est stockée. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Fournissez les informations de demande en créant une instance de la classe `DeleteArchiveRequest`.

   Vous devez fournir un ID d'archive, un nom de coffre et votre ID de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDK pour Java avec Amazon Glacier.

3. Exécutez la méthode `deleteArchive` en fournissant l'objet de demande comme paramètre.

L'extrait de code Java suivant illustre les étapes précédentes.

```
AmazonGlacierClient client;

DeleteArchiveRequest request = new DeleteArchiveRequest()
    .withVaultName("*** provide a vault name ***")
    .withArchiveId("*** provide an archive ID ***");

client.deleteArchive(request);
```

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Suppression d'une archive (DELETE archive).

Exemple : suppression d'une archive à l'aide du AWS SDK pour Java

L'exemple de code Java suivant utilise le AWS SDK pour Java pour supprimer une archive. Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, consultezExécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse. Vous devez mettre à jour le code tel qu'il est présenté avec le nom du coffre et l'ID de l'archive à supprimer.

Example

```
import java.io.IOException;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.services.glacier.AmazonGlacierClient;
import com.amazonaws.services.glacier.model.DeleteArchiveRequest;

public class ArchiveDelete {

    public static String vaultName = "*** provide vault name ****";
    public static String archiveId = "*** provide archive ID***";
    public static AmazonGlacierClient client;

    public static void main(String[] args) throws IOException {

     ProfileCredentialsProvider credentials = new ProfileCredentialsProvider();

        client = new AmazonGlacierClient(credentials);
        client.setEndpoint("https://glacier.us-east-1.amazonaws.com/");

        try {

            // Delete the archive.
            client.deleteArchive(new DeleteArchiveRequest()
                .withVaultName(vaultName)
                .withArchiveId(archiveId));

            System.out.println("Deleted archive successfully.");

        } catch (Exception e) {
            System.err.println("Archive not deleted.");
            System.err.println(e);
        }
    }
}
```

Suppression d'une archive dans Amazon Glacier à l'aide du AWS SDK pour .NET

Les niveaux de haut et de bas niveau APIs fournis par le SDK Amazon pour .NET fournissent une méthode pour supprimer une archive.

Rubriques

- Suppression d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET
- Supprimer une archive à l'aide de l'API de bas niveau AWS SDK pour .NET

Suppression d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

La classe `ArchiveTransferManager` de l'API de haut niveau comporte la méthode `DeleteArchive` qui sert à supprimer une archive.

Exemple : suppression d'une archive à l'aide de l'API de haut niveau du AWS SDK pour .NET

L'exemple de code C# suivant utilise l'API de haut niveau de AWS SDK pour .NET pour supprimer une archive. Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID de l'archive à supprimer.

Example

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Transfer;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDeleteHighLevel
  {
    static string vaultName = "examplevault";
    static string archiveId = "*** Provide archive ID ***";

    public static void Main(string[] args)
```

```
    {
      try
      {
        var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
        manager.DeleteArchive(vaultName, archiveId);
        Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
    }
  }
}
```

Supprimer une archive à l'aide de l'API de bas niveau AWS SDK pour .NET

Voici les étapes pour supprimer une archive à l'aide du kit AWS SDK pour .NET :

1. Créez une instance de la classe `AmazonGlacierClient` (le client).

   Vous devez spécifier une AWS région dans laquelle l'archive que vous souhaitez supprimer est stockée. Toutes les opérations que vous effectuez à l'aide de ce client s'appliquent à cette AWS région.

2. Fournissez les informations de demande en créant une instance de la classe `DeleteArchiveRequest`.

   Vous devez fournir un ID d'archive, un nom de coffre et votre ID de compte. Si vous n'indiquez pas d'ID de compte, l'ID de compte associé aux informations d'identification que vous avez indiquées pour signer la demande est pris en compte. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDKs avec Amazon Glacier.

3. Exécutez la méthode `DeleteArchive` en fournissant l'objet de demande comme paramètre.

Exemple : suppression d'une archive à l'aide de l'API de bas niveau du AWS SDK pour .NET

L'exemple de code C# suivant illustre les étapes précédentes. L'exemple utilise l'API de bas niveau du AWS SDK pour .NET pour supprimer une archive.

> **Note**
>
> Pour obtenir des informations sur l'API REST sous-jacente, consultez la section Suppression d'une archive (DELETE archive).

Pour step-by-step obtenir des instructions sur la façon d'exécuter cet exemple, reportez-vous àExemples de code en cours d'exécution. Vous devez mettre à jour le code tel qu'il est présenté avec l'ID de l'archive à supprimer.

Example

```
using System;
using Amazon.Glacier;
using Amazon.Glacier.Model;
using Amazon.Runtime;

namespace glacier.amazon.com.docsamples
{
  class ArchiveDeleteLowLevel
  {
    static string vaultName = "examplevault";
    static string archiveId = "*** Provide archive ID ***";

    public static void Main(string[] args)
    {
      AmazonGlacierClient client;
      try
      {
        using (client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2))
        {
          Console.WriteLine("Deleting the archive");
          DeleteAnArchive(client);
        }
        Console.WriteLine("Operations successful. To continue, press Enter");
        Console.ReadKey();
      }
      catch (AmazonGlacierException e) { Console.WriteLine(e.Message); }
      catch (AmazonServiceException e) { Console.WriteLine(e.Message); }
      catch (Exception e) { Console.WriteLine(e.Message); }
      Console.WriteLine("To continue, press Enter");
      Console.ReadKey();
```

```
    }

    static void DeleteAnArchive(AmazonGlacierClient client)
    {
      DeleteArchiveRequest request = new DeleteArchiveRequest()
      {
        VaultName = vaultName,
        ArchiveId = archiveId
      };
      DeleteArchiveResponse response = client.DeleteArchive(request);
    }
  }
}
```

Suppression d'une archive Amazon Glacier à l'aide de l'API REST

Vous pouvez utiliser l'API de suppression d'archive pour supprimer une archive.

- Pour plus d'informations sur l'API de suppression d'archive, consultez la page Suppression d'une archive (DELETE archive).
- Pour plus d'informations sur l'utilisation de l'API REST, consultez Référence d'API pour Amazon Glacier.

Suppression d'une archive dans Amazon Glacier à l'aide du AWS Command Line Interface

Vous pouvez supprimer des archives dans Amazon Glacier (Amazon Glacier) à l'aide du AWS Command Line Interface (AWS CLI).

Rubriques

- (Prérequis) Configuration du AWS CLI
- Exemple : suppression d'une archive à l'aide du AWS CLI

(Prérequis) Configuration du AWS CLI

1. Téléchargez et configurez l’interface AWS CLI. Pour obtenir des instructions, consultez les rubriques suivantes dans le Guide de l’utilisateur de l’interface AWS Command Line Interface :

Installation du AWS Command Line Interface

Configuration du AWS Command Line Interface

2. Vérifiez votre AWS CLI configuration en saisissant les commandes suivantes à l'invite de commande. Ces commandes ne fournissent pas directement d'informations d'identification, par conséquent ce sont les informations du profil par défaut qui sont utilisées.

   - Essayez d'utiliser la commande help.

     ```
     aws help
     ```

   - Pour obtenir la liste des coffres-forts Amazon Glacier sur le compte configuré, utilisez la `list-vaults` commande. *123456789012*Remplacez-le par votre Compte AWS identifiant.

     ```
     aws glacier list-vaults --account-id 123456789012
     ```

   - Pour voir les données de configuration actuelles pour le AWS CLI, utilisez la `aws configure list` commande.

     ```
     aws configure list
     ```

Exemple : suppression d'une archive à l'aide du AWS CLI

1. Utilisez la commande initiate-job pour démarrer une tâche d'extraction d'inventaire.

   ```
   aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333 --
   job-parameters="{\"Type\":\"inventory-retrieval\"}"
   ```

   Sortie attendue :

   ```
   {
       "location": "/111122223333/vaults/awsexamplevault/jobs/*** jobid ***",
       "jobId": "*** jobid ***"
   }
   ```

2. Utilisez la commande describe-job pour vérifier le statut de la tâche d'extraction précédente.

```
aws glacier describe-job --vault-name awsexamplevault --account-id 111122223333 --
job-id *** jobid ***
```

Sortie attendue :

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "*** vault arn ***",
    "Completed": false,
    "JobId": "*** jobid ***",
    "Action": "InventoryRetrieval",
    "CreationDate": "*** job creation date ***",
    "StatusCode": "InProgress"
}
```

3. Attendez que la tâche se termine.

   Vous devez attendre que la sortie de la tâche soit prête pour pouvoir effectuer le téléchargement. Si vous avez défini une configuration de notification sur le coffre-fort ou si vous avez spécifié une rubrique Amazon Simple Notification Service (Amazon SNS) lorsque vous avez lancé la tâche, Amazon Glacier envoie un message à la rubrique une fois la tâche terminée.

   Vous pouvez définir la configuration des notifications pour des événements spécifiques concernant le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier. Amazon Glacier envoie un message à la rubrique SNS spécifiée chaque fois que l'événement spécifique se produit.

4. Une fois l'opération terminée, utilisez la commande get-job-output pour télécharger le travail d'extraction dans le fichier `output.json`.

```
aws glacier get-job-output --vault-name awsexamplevault --account-id 111122223333
 --job-id *** jobid *** output.json
```

Cette commande produit un fichier avec les champs suivants.

```
{
"VaultARN":"arn:aws:glacier:region:111122223333:vaults/awsexamplevault",
"InventoryDate":"*** job completion date ***",
"ArchiveList":[
{"ArchiveId":"*** archiveid ***",
"ArchiveDescription":*** archive description (if set) ***,
"CreationDate":"*** archive creation date ***",
"Size":"*** archive size (in bytes) ***",
"SHA256TreeHash":"*** archive hash ***"
}
{"ArchiveId":
...
]}
```

5. Utilisez la commande `delete-archive` pour supprimer chaque archive d'un coffre jusqu'à ce qu'il n'en reste aucune.

```
aws glacier delete-archive --vault-name awsexamplevault --account-id 111122223333
 --archive-id *** archiveid ***
```

Utilisation du AWS SDKs avec Amazon Glacier

AWS vous permet SDKs de développer des applications pour Amazon Glacier. Les bibliothèques du SDK encapsulent l'API Amazon Glacier sous-jacente, simplifiant ainsi vos tâches de programmation. Par exemple, pour chaque demande envoyée à Amazon Glacier, vous devez inclure une signature afin d'authentifier vos demandes. Lorsque vous utilisez les bibliothèques du SDK, vous devez uniquement fournir vos informations de AWS sécurité dans votre code. Les bibliothèques calculent la signature nécessaire et l'incluent dans la demande envoyée à Amazon Glacier. Ils AWS SDKs fournissent des bibliothèques qui correspondent à l'API REST sous-jacente et fournissent des objets que vous pouvez utiliser pour créer facilement des demandes et traiter des réponses.

Rubriques

- AWS Bibliothèques SDK pour Java et .NET
- Utilisation d'Amazon Glacier avec un AWS SDK
- Utilisation du AWS SDK pour Java avec Amazon Glacier
- Utilisation du AWS SDK pour .NET avec Amazon Glacier

Le AWS Command Line Interface (AWS CLI) est un outil unifié qui permet de gérer votre Services AWS, y compris Amazon Glacier. Pour plus d'informations sur le téléchargement du AWS CLI, voir AWS Command Line Interface. Pour obtenir la liste des commandes de la CLI Amazon Glacier, consultez la référence des AWS CLI commandes.

AWS Bibliothèques SDK pour Java et .NET

AWS SDKs Pour Java et .NET, vous trouverez des bibliothèques wrapper de haut niveau et de bas niveau.

Vous trouverez des exemples d'utilisation d'Amazon Glacier en consultant le guide du développeur AWS SDK pour Java et l' AWS SDK pour .NET ensemble de ce guide.

Qu'est-ce qu'une API de bas niveau ?

Les bibliothèques wrapper de bas niveau mappent étroitement l'API REST sous-jacente (Référence d'API pour Amazon Glacier) prise en charge par Amazon Glacier. Pour chaque opération REST Amazon Glacier, l'API de bas niveau fournit une méthode correspondante, un objet de demande pour

que vous fournissiez les informations de demande et un objet de réponse pour que vous puissiez traiter la réponse d'Amazon Glacier. Les bibliothèques d'encapsulation de bas niveau constituent l'implémentation la plus complète des opérations sous-jacentes d'Amazon Glacier.

Pour plus d'informations sur ces bibliothèques SDK, consultez les pages Utilisation du AWS SDK pour Java avec Amazon Glacier et Utilisation du AWS SDK pour .NET avec Amazon Glacier.

Qu'est-ce qu'une API de haut niveau ?

Pour simplifier davantage le développement d'applications, ces bibliothèques offrent une abstraction de niveau supérieur pour certaines opérations. Par exemple :

- Chargement d'une archive : pour charger une archive à l'aide de l'API de bas niveau, outre le nom du fichier et le nom du coffre dans lequel vous souhaitez enregistrer l'archive, vous devez fournir un total de contrôle (hachage d'arborescence SHA-256) de la charge utile. Toutefois, l'API de haut niveau calcule le total de contrôle pour vous.
- Téléchargement d'une archive ou d'un inventaire de coffre : pour télécharger une archive en utilisant l'API de bas niveau, vous devez d'abord lancer une tâche, attendre qu'elle se termine, puis obtenir la sortie correspondante. Vous devez écrire du code supplémentaire pour configurer une rubrique Amazon Simple Notification Service (Amazon SNS) afin qu'Amazon Glacier vous avertisse lorsque la tâche est terminée. Vous avez également besoin d'un mécanisme d'interrogation pour vérifier si un message de fin de tâche a été publié dans la rubrique. L'API de haut niveau fournit une méthode permettant de télécharger une archive qui s'occupe de toutes ces étapes. Vous spécifiez uniquement un ID d'archivage et un chemin d'accès du dossier dans lequel vous souhaitez enregistrer les données téléchargées.

Pour plus d'informations sur ces bibliothèques SDK, consultez les pages Utilisation du AWS SDK pour Java avec Amazon Glacier et Utilisation du AWS SDK pour .NET avec Amazon Glacier.

Quand utiliser l'API de haut niveau et celle de bas niveau

En général, si l'API de haut niveau fournit des méthodes dont vous avez besoin pour effectuer une opération, vous devez l'utiliser en raison de sa simplicité. Toutefois, si l'API de haut niveau n'offre pas la fonctionnalité recherchée, vous pouvez utiliser l'API de bas niveau. En outre, l'API de bas niveau permet un contrôle précis de l'opération comme logique de nouvelle tentative en cas de défaillance. Par exemple, lors du chargement d'une archive, l'API de haut niveau utilise la taille de fichier pour déterminer s'il faut charger l'archive en une seule opération ou utiliser l'API de

chargement partitionné. L'API dispose également d'une logique de nouvelle tentative intégrée en cas de défaillance d'un chargement. Toutefois, votre application peut avoir besoin d'un contrôle plus précis sur ces décisions, auquel cas vous pouvez utiliser l'API de bas niveau.

Utilisation d'Amazon Glacier avec un AWS SDK

AWS des kits de développement logiciel (SDKs) sont disponibles pour de nombreux langages de programmation courants. Chaque kit SDK fournit une API, des exemples de code et de la documentation qui facilitent la création d'applications par les développeurs dans leur langage préféré.

Documentation SDK	Exemples de code
AWS SDK pour C++	AWS SDK pour C++ exemples de code
AWS CLI	AWS CLI exemples de code
AWS SDK pour Go	AWS SDK pour Go exemples de code
AWS SDK pour Java	AWS SDK pour Java exemples de code
AWS SDK pour JavaScript	AWS SDK pour JavaScript exemples de code
AWS SDK pour Kotlin	AWS SDK pour Kotlin exemples de code
AWS SDK pour .NET	AWS SDK pour .NET exemples de code
AWS SDK pour PHP	AWS SDK pour PHP exemples de code
Outils AWS pour PowerShell	Outils AWS pour PowerShell exemples de code
AWS SDK pour Python (Boto3)	AWS SDK pour Python (Boto3) exemples de code
AWS SDK pour Ruby	AWS SDK pour Ruby exemples de code
AWS SDK pour Rust	AWS SDK pour Rust exemples de code
AWS SDK pour SAP ABAP	AWS SDK pour SAP ABAP exemples de code
AWS SDK pour Swift	AWS SDK pour Swift exemples de code

Pour des exemples spécifiques à Amazon Glacier, consultezExemples de code pour Amazon Glacier à l'aide d'Amazon Glacier AWS SDKs.

> **Exemple de disponibilité**
>
> Vous n'avez pas trouvé ce dont vous avez besoin ? Demandez un exemple de code en utilisant le lien Provide feedback (Fournir un commentaire) en bas de cette page.

Utilisation du AWS SDK pour Java avec Amazon Glacier

AWS SDK pour Java Il fournit à la fois un haut niveau et un bas niveau APIs pour Amazon Glacier (Amazon Glacier), comme décrit dansUtilisation du AWS SDKs avec Amazon Glacier. Pour plus d'informations sur le téléchargement du AWS SDK pour Java, consultez Amazon SDK for Java.

> **Note**
>
> AWS SDK pour Java fournit des clients sécurisés pour accéder à Amazon Glacier. En tant que bonne pratique, vos applications doivent créer un seul client et le réutiliser entre les threads.

Rubriques

- Utilisation de l'API de bas niveau
- Utilisation de l'API de haut niveau
- Exécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse
- Définition du point de terminaison

Utilisation de l'API de bas niveau

La `AmazonGlacierClient` classe de bas niveau fournit toutes les méthodes qui correspondent aux opérations REST sous-jacentes d'Amazon Glacier (Référence d'API pour Amazon Glacier). Lorsque vous appelez l'une de ces méthodes, vous devez créer un objet de demande correspondant et fournir un objet de réponse dans lequel la méthode peut renvoyer la réponse d'Amazon Glacier à l'opération.

Par exemple, la classe `AmazonGlacierClient` fournit la méthode `createVault` pour créer un coffre. Cette méthode est mappée à l'opération REST de création de coffre sous-jacente (consultez la

section Création de coffre (PUT vault)). Pour utiliser cette méthode, vous devez créer des instances de l'`CreateVaultResult`objet qui reçoit la réponse d'Amazon Glacier, comme indiqué dans l'extrait de code Java suivant :

```
AmazonGlacierClient client = new AmazonGlacierClient(credentials);
client.setEndpoint("https://glacier.us-west-2.amazonaws.com/");

CreateVaultRequest request = new CreateVaultRequest()
    .withAccountId("-")
    .withVaultName(vaultName);
CreateVaultResult result = client.createVault(createVaultRequest);
```

Tous les exemples de bas niveau de ce manuel utilisent ce modèle.

> **Note**
>
> Le segment de code précédent spécifie la valeur `AccountID` lors de la création de la demande. Toutefois, lorsque vous utilisez le AWS SDK pour Java, le `AccountId` contenu de la requête est facultatif. Par conséquent, tous les exemples de bas niveau de ce guide ne définissent pas cette valeur. `AccountId`C'est l' Compte AWS identifiant. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier l' Compte AWS ID ou éventuellement un « - », auquel cas Amazon Glacier utilise l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas les traits d'union. Lors de l'utilisation AWS SDK pour Java, si vous ne fournissez pas l'identifiant du compte, la bibliothèque définit l'identifiant du compte sur « - ».

Utilisation de l'API de haut niveau

Pour simplifier davantage le développement de votre application, la classe AWS SDK pour Java fournit la `ArchiveTransferManager` classe qui implémente une abstraction de niveau supérieur pour certaines des méthodes de l'API de bas niveau. Il fournit des méthodes utiles, telles que les méthodes `upload` et `download`, pour les opérations d'archivage.

Par exemple, l'extrait de code Java suivant utilise la méthode de haut niveau `upload` pour charger une archive.

```
String vaultName = "examplevault";
String archiveToUpload = "c:/folder/exampleArchive.zip";

ArchiveTransferManager atm = new ArchiveTransferManager(client, credentials);
String archiveId = atm.upload(vaultName, "Tax 2012 documents", new
 File(archiveToUpload)).getArchiveId();
```

Notez que toutes les opérations que vous effectuez s'appliquent à la AWS région que vous avez spécifiée lors de la création de l'`ArchiveTransferManager`objet. Si vous ne spécifiez aucune AWS région, elle est AWS SDK pour Java définie `us-east-1` comme AWS région par défaut.

Tous les exemples de haut niveau de ce manuel utilisent ce modèle.

> **Note**
>
> La classe `ArchiveTransferManager` de haut niveau peut être construite avec une instance `AmazonGlacierClient` ou `AWSCredentials`.

Exécution d'exemples Java pour Amazon Glacier à l'aide d'Eclipse

Pour démarrer avec les exemples de code Java, la méthode la plus simple consiste à installer le kit AWS Toolkit for Eclipse le plus récent. Pour plus d'informations sur l'installation ou la mise à jour de la boîte à outils la plus récente, accédez à la page http://aws.amazon.com/eclipse. Les tâches suivantes vous guident à travers la création et le test des exemples de code Java fournis dans cette section.

Procédure générale de création d'exemples de code Java

1	Créez un profil d'identification par défaut pour vos AWS informations d'identification, comme décrit dans la AWS SDK pour Java rubrique Fournir des AWS informations d'identification dans le SDK Amazon pour Java.
2	Créez un nouveau projet AWS Java dans Eclipse. Ce projet est préconfiguré avec le kit AWS SDK pour Java.
3	Copiez le code dans la section que vous êtes en train de consulter et collez-le dans votre projet.

4	Mettez à jour le code en fournissant les données requises. Par exemple, si vous chargez un fichier, indiquez le chemin du fichier et le nom du compartiment.
5	Exécutez le code. Vérifiez si l'objet a été créé à l'aide de AWS Management Console. Pour plus d'informations à ce sujet AWS Management Console, rendez-vous sur http://aws.amazon.com/console/.

Définition du point de terminaison

Par défaut, il AWS SDK pour Java utilise le point de terminaison`https://glacier.us-east-1.amazonaws.com`. Vous pouvez définir le point de terminaison de manière explicite, comme illustré dans les extraits de code Java suivants.

L'extrait de code suivant montre comment définir le point de terminaison sur la région USA Ouest (Oregon) (`us-west-2`) dans l'API de bas niveau.

Example

```
client = new AmazonGlacierClient(credentials);
client.setEndpoint("glacier.us-west-2.amazonaws.com");
```

L'extrait de code suivant montre comment définir le point de terminaison sur la région USA Ouest (Oregon) dans l'API de haut niveau.

```
glacierClient = new AmazonGlacierClient(credentials);
sqsClient = new AmazonSQSClient(credentials);
snsClient = new AmazonSNSClient(credentials);

glacierClient.setEndpoint("glacier.us-west-2.amazonaws.com");
sqsClient.setEndpoint("sqs.us-west-2.amazonaws.com");
snsClient.setEndpoint("sns.us-west-2.amazonaws.com");

ArchiveTransferManager atm = new ArchiveTransferManager(glacierClient, sqsClient,
 snsClient);
```

Pour obtenir la liste des AWS régions et des points de terminaison pris en charge, consultezAccès à Amazon Glacier.

Utilisation du AWS SDK pour .NET avec Amazon Glacier

L' AWS SDK pour .NET API est disponible dans`AWSSDK.dll`. Pour plus d'informations sur le téléchargement du AWS SDK pour .NET, consultez la section Bibliothèques d'exemples de codes. Comme décrit dansUtilisation du AWS SDKs avec Amazon Glacier, il AWS SDK pour .NET fournit à la fois le haut niveau et le bas niveau APIs.

> **Note**
>
> L'API de bas niveau et l'API de haut niveau fournissent des clients sécurisés pour accéder à Amazon Glacier. En tant que bonne pratique, vos applications doivent créer un seul client et le réutiliser entre les threads.

Rubriques

- Utilisation de l'API de bas niveau
- Utilisation de l'API de haut niveau
- Exemples de code en cours d'exécution
- Définition du point de terminaison

Utilisation de l'API de bas niveau

La `AmazonGlacierClient` classe de bas niveau fournit toutes les méthodes qui correspondent aux opérations REST sous-jacentes d'Amazon Glacier (Amazon Glacier) (Référence d'API pour Amazon Glacier). Lorsque vous appelez l'une de ces méthodes, vous devez créer un objet de demande correspondant et fournir un objet de réponse dans lequel la méthode peut renvoyer une réponse Amazon Glacier à l'opération.

Par exemple, la classe `AmazonGlacierClient` fournit la méthode `CreateVault` pour créer un coffre. Cette méthode est mappée à l'opération REST de création de coffre sous-jacente (consultez la section Création de coffre (PUT vault)). Pour utiliser cette méthode, vous devez créer des instances des `CreateVaultResponse` classes `CreateVaultRequest` et afin de fournir les informations de demande et de recevoir une réponse d'Amazon Glacier, comme indiqué dans l'extrait de code C# suivant :

```
AmazonGlacierClient client;
```

```
client = new AmazonGlacierClient(Amazon.RegionEndpoint.USEast1);

CreateVaultRequest request = new CreateVaultRequest()
{
  AccountId = "-",
  VaultName = "*** Provide vault name ***"
};

CreateVaultResponse response = client.CreateVault(request);
```

Tous les exemples de bas niveau de ce manuel utilisent ce modèle.

> **Note**
>
> Le segment de code précédent spécifie la valeur `AccountId` lors de la création de la demande. Toutefois, lorsque vous utilisez le AWS SDK pour .NET, le `AccountId` contenu de la requête est facultatif. Par conséquent, tous les exemples de bas niveau de ce guide ne définissent pas cette valeur. `AccountId`C'est l' Compte AWS identifiant. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier l' Compte AWS ID ou éventuellement un « - », auquel cas Amazon Glacier utilise l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas les traits d'union. Lors de l'utilisation AWS SDK pour .NET, si vous ne fournissez pas l'identifiant du compte, la bibliothèque définit l'identifiant du compte sur « - ».

Utilisation de l'API de haut niveau

Pour simplifier davantage le développement de votre application, la classe AWS SDK pour .NET fournit la `ArchiveTransferManager` classe qui implémente une abstraction de niveau supérieur pour certaines des méthodes de l'API de bas niveau. Il fournit des méthodes utiles, telles que les méthodes `Upload` et `Download`, pour les opérations d'archivage.

Par exemple, l'extrait de code C suivant utilise la méthode de haut niveau `Upload` pour charger une archive.

```
string vaultName = "examplevault";
string archiveToUpload = "c:\folder\exampleArchive.zip";
```

```
var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USEast1);
string archiveId = manager.Upload(vaultName, "archive description",
 archiveToUpload).ArchiveId;
```

Notez que toutes les opérations que vous effectuez s'appliquent à la AWS région que vous avez spécifiée lors de la création de l'`ArchiveTransferManager`objet. Tous les exemples de haut niveau de ce manuel utilisent ce modèle.

> **Note**
>
> La classe `ArchiveTransferManager` de haut niveau a toujours besoin du client `AmazonGlacierClient` de bas niveau, que vous pouvez passer explicitement ou le `ArchiveTransferManager` crée le client.

Exemples de code en cours d'exécution

Pour démarrer avec les exemples de code .NET, la méthode la plus simple consiste à installer le kit AWS SDK pour .NET. Pour plus d'informations, accédez à la documentation Kit SDK Amazon pour .NET.

La procédure suivante décrit les étapes qui vous permettront de tester les exemples de code proposés dans ce manuel.

Processus général de création des exemples de code .NET (à l'aide de Visual Studio)

1	Créez un profil d'identification pour vos AWS informations d'identification, comme décrit dans la rubrique AWS Configuration des informations d'identification du SDK Amazon pour .NET.
2	Créez un projet Visual Studio en utilisant le modèle Projet vide AWS .
3	Dans le fichier de projet `Program.cs` , remplacez le code par celui figurant dans la section que vous consultez actuellement.
4	Exécutez le code. Vérifiez si l'objet a été créé à l'aide de AWS Management Console. Pour plus d'informations à ce sujet AWS Management Console, rendez-vous sur http://aws.amazon.com/console/.

Définition du point de terminaison

Par défaut, le AWS SDK pour .NET point de terminaison est défini sur la région de l'ouest des États-Unis (Oregon) (`https://glacier.us-west-2.amazonaws.com`). Vous pouvez définir le point de terminaison sur d'autres AWS régions, comme indiqué dans les extraits de code C# suivants.

L'extrait de code suivant montre comment définir le point de terminaison sur la région USA Ouest (Oregon) (`us-west-2`) dans l'API de bas niveau.

Example

```
AmazonGlacierClient client = new AmazonGlacierClient(Amazon.RegionEndpoint.USWest2);
```

L'extrait de code suivant montre comment définir le point de terminaison sur la région USA Ouest (Oregon) dans l'API de haut niveau.

```
var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
```

Pour obtenir la liste actuelle des AWS régions et points de terminaison pris en charge, consultezAccès à Amazon Glacier.

Exemples de code pour Amazon Glacier à l'aide d'Amazon Glacier AWS SDKs

Les exemples de code suivants montrent comment utiliser Amazon Glacier avec un kit de développement AWS logiciel (SDK).

Les actions sont des extraits de code de programmes plus larges et doivent être exécutées dans leur contexte. Alors que les actions vous indiquent comment appeler des fonctions de service individuelles, vous pouvez les voir en contexte dans leurs scénarios associés.

Les scénarios sont des exemples de code qui vous montrent comment accomplir des tâches spécifiques en appelant plusieurs fonctions au sein d'un même service ou combinés à d'autres Services AWS.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit de développement logiciel (SDK).

Exemples de code

- Exemples de base pour l'utilisation d'Amazon Glacier AWS SDKs
  - Bonjour Amazon Glacier
  - Actions pour Amazon Glacier utilisant AWS SDKs
    - Utilisation AddTagsToVault avec un AWS SDK ou une CLI
    - Utilisation CreateVault avec un AWS SDK ou une CLI
    - Utilisation DeleteArchive avec un AWS SDK ou une CLI
    - Utilisation DeleteVault avec un AWS SDK ou une CLI
    - Utilisation DeleteVaultNotifications avec un AWS SDK ou une CLI
    - Utilisation DescribeJob avec un AWS SDK ou une CLI
    - Utilisation DescribeVault avec un AWS SDK ou une CLI
    - Utilisation GetJobOutput avec un AWS SDK ou une CLI
    - Utilisation GetVaultNotifications avec un AWS SDK ou une CLI
    - Utilisation InitiateJob avec un AWS SDK ou une CLI
    - Utilisation ListJobs avec un AWS SDK ou une CLI

    - Utilisation ListTagsForVault avec un AWS SDK ou une CLI
    - Utilisation ListVaults avec un AWS SDK ou une CLI
    - Utilisation SetVaultNotifications avec un AWS SDK ou une CLI
    - Utilisation UploadArchive avec un AWS SDK ou une CLI
    - Utilisation UploadMultipartPart avec un AWS SDK ou une CLI
- Scénarios d'utilisation d'Amazon Glacier AWS SDKs
    - Archivez un fichier sur Amazon Glacier, recevez des notifications et lancez une tâche à l'aide d'un AWS SDK
    - Obtenez le contenu de l'archive Amazon Glacier et supprimez l'archive à l'aide d'un AWS SDK

Exemples de base pour l'utilisation d'Amazon Glacier AWS SDKs

Les exemples de code suivants montrent comment utiliser les bases d'Amazon Glacier avec AWS SDKs.

Exemples

- Bonjour Amazon Glacier
- Actions pour Amazon Glacier utilisant AWS SDKs
    - Utilisation AddTagsToVault avec un AWS SDK ou une CLI
    - Utilisation CreateVault avec un AWS SDK ou une CLI
    - Utilisation DeleteArchive avec un AWS SDK ou une CLI
    - Utilisation DeleteVault avec un AWS SDK ou une CLI
    - Utilisation DeleteVaultNotifications avec un AWS SDK ou une CLI
    - Utilisation DescribeJob avec un AWS SDK ou une CLI
    - Utilisation DescribeVault avec un AWS SDK ou une CLI
    - Utilisation GetJobOutput avec un AWS SDK ou une CLI
    - Utilisation GetVaultNotifications avec un AWS SDK ou une CLI
    - Utilisation InitiateJob avec un AWS SDK ou une CLI
    - Utilisation ListJobs avec un AWS SDK ou une CLI
    - Utilisation ListTagsForVault avec un AWS SDK ou une CLI
    - Utilisation ListVaults avec un AWS SDK ou une CLI
    - Utilisation SetVaultNotifications avec un AWS SDK ou une CLI

- Utilisation UploadArchive avec un AWS SDK ou une CLI
- Utilisation UploadMultipartPart avec un AWS SDK ou une CLI

Bonjour Amazon Glacier

L'exemple de code suivant montre comment commencer à utiliser Amazon Glacier.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
using Amazon.Glacier;
using Amazon.Glacier.Model;

namespace GlacierActions;

public static class HelloGlacier
{
    static async Task Main()
    {
        var glacierService = new AmazonGlacierClient();

        Console.WriteLine("Hello Amazon Glacier!");
        Console.WriteLine("Let's list your Glacier vaults:");

        // You can use await and any of the async methods to get a response.
        // Let's get the vaults using a paginator.
        var glacierVaultPaginator = glacierService.Paginators.ListVaults(
            new ListVaultsRequest { AccountId = "-" });

        await foreach (var vault in glacierVaultPaginator.VaultList)
        {
            Console.WriteLine($"{vault.CreationDate}:{vault.VaultName}, ARN:
{vault.VaultARN}");
```

```
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous ListVaultsà la section Référence des AWS SDK pour .NET API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Actions pour Amazon Glacier utilisant AWS SDKs

Les exemples de code suivants montrent comment effectuer des actions Amazon Glacier individuelles avec AWS SDKs. Chaque exemple inclut un lien vers GitHub, où vous pouvez trouver des instructions pour configurer et exécuter le code.

Ces extraits appellent l'API Amazon Glacier et sont des extraits de code de programmes plus volumineux qui doivent être exécutés en contexte. Vous pouvez voir les actions dans leur contexte dans Scénarios d'utilisation d'Amazon Glacier AWS SDKs.

Les exemples suivants incluent uniquement les actions les plus couramment utilisées. Pour une liste complète, consultez le manuel Amazon Glacier API Reference.

Exemples

- Utilisation AddTagsToVault avec un AWS SDK ou une CLI
- Utilisation CreateVault avec un AWS SDK ou une CLI
- Utilisation DeleteArchive avec un AWS SDK ou une CLI
- Utilisation DeleteVault avec un AWS SDK ou une CLI
- Utilisation DeleteVaultNotifications avec un AWS SDK ou une CLI
- Utilisation DescribeJob avec un AWS SDK ou une CLI
- Utilisation DescribeVault avec un AWS SDK ou une CLI
- Utilisation GetJobOutput avec un AWS SDK ou une CLI
- Utilisation GetVaultNotifications avec un AWS SDK ou une CLI

- Utilisation InitiateJob avec un AWS SDK ou une CLI
- Utilisation ListJobs avec un AWS SDK ou une CLI
- Utilisation ListTagsForVault avec un AWS SDK ou une CLI
- Utilisation ListVaults avec un AWS SDK ou une CLI
- Utilisation SetVaultNotifications avec un AWS SDK ou une CLI
- Utilisation UploadArchive avec un AWS SDK ou une CLI
- Utilisation UploadMultipartPart avec un AWS SDK ou une CLI

Utilisation **`AddTagsToVault`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `AddTagsToVault`.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Add tags to the items in an Amazon S3 Glacier vault.
   /// </summary>
   /// <param name="vaultName">The name of the vault to add tags to.</param>
   /// <param name="key">The name of the object to tag.</param>
   /// <param name="value">The tag value to add.</param>
   /// <returns>A Boolean value indicating the success of the action.</returns>
   public async Task<bool> AddTagsToVaultAsync(string vaultName, string key,
string value)
   {
       var request = new AddTagsToVaultRequest
       {
           Tags = new Dictionary<string, string>
               {
                   { key, value },
               },
```

```
            AccountId = "-",
            VaultName = vaultName,
        };

        var response = await _glacierService.AddTagsToVaultAsync(request);
        return response.HttpStatusCode == HttpStatusCode.NoContent;
    }
```

- Pour plus de détails sur l'API, reportez-vous AddTagsToVaultà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante ajoute deux balises à un coffre-fort nommé `my-vault` :

```
aws glacier add-tags-to-vault --account-id - --vault-name my-vault --
tags id=1234,date=july2015
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous AddTagsToVaultà la section Référence des AWS CLI commandes.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`CreateVault`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `CreateVault`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Archivage d'un fichier, obtention de notifications et lancement d'une tâche

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
/// <summary>
/// Create an Amazon S3 Glacier vault.
/// </summary>
/// <param name="vaultName">The name of the vault to create.</param>
/// <returns>A Boolean value indicating the success of the action.</returns>
public async Task<bool> CreateVaultAsync(string vaultName)
{
    var request = new CreateVaultRequest
    {
        // Setting the AccountId to "-" means that
        // the account associated with the current
        // account will be used.
        AccountId = "-",
        VaultName = vaultName,
    };

    var response = await _glacierService.CreateVaultAsync(request);

    Console.WriteLine($"Created {vaultName} at: {response.Location}");

    return response.HttpStatusCode == HttpStatusCode.Created;
}
```

- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante crée un coffre-fort nommé `my-vault` :

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.CreateVaultRequest;
import software.amazon.awssdk.services.glacier.model.CreateVaultResponse;
import software.amazon.awssdk.services.glacier.model.GlacierException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class CreateVault {
    public static void main(String[] args) {
```

```
        final String usage = """

                Usage:    <vaultName>

                Where:
                   vaultName - The name of the vault to create.

                """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String vaultName = args[0];
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        createGlacierVault(glacier, vaultName);
        glacier.close();
    }

    public static void createGlacierVault(GlacierClient glacier, String
 vaultName) {
        try {
            CreateVaultRequest vaultRequest = CreateVaultRequest.builder()
                    .vaultName(vaultName)
                    .build();

            CreateVaultResponse createVaultResult =
 glacier.createVault(vaultRequest);
            System.out.println("The URI of the new vault is " +
 createVaultResult.location());

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des AWS SDK for Java 2.x API.

JavaScript

SDK pour JavaScript (v3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Créez le client.

```
const { GlacierClient } = require("@aws-sdk/client-glacier");
// Set the AWS Region.
const REGION = "REGION";
//Set the Redshift Service Object
const glacierClient = new GlacierClient({ region: REGION });
export { glacierClient };
```

Créez le coffre.

```
// Load the SDK for JavaScript
import { CreateVaultCommand } from "@aws-sdk/client-glacier";
import { glacierClient } from "./libs/glacierClient.js";

// Set the parameters
const vaultname = "VAULT_NAME"; // VAULT_NAME
const params = { vaultName: vaultname };

const run = async () => {
  try {
    const data = await glacierClient.send(new CreateVaultCommand(params));
    console.log("Success, vault created!");
    return data; // For unit tests.
  } catch (err) {
    console.log("Error");
  }
```

```
};
run();
```

- Pour plus d'informations, consultez le Guide du développeur AWS SDK pour JavaScript.
- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des AWS SDK pour JavaScript API.

SDK pour JavaScript (v2)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
// Load the SDK for JavaScript
var AWS = require("aws-sdk");
// Set the region
AWS.config.update({ region: "REGION" });

// Create a new service object
var glacier = new AWS.Glacier({ apiVersion: "2012-06-01" });
// Call Glacier to create the vault
glacier.createVault({ vaultName: "YOUR_VAULT_NAME" }, function (err) {
  if (!err) {
    console.log("Created vault!");
  }
});
```

- Pour plus d'informations, consultez le Guide du développeur AWS SDK pour JavaScript.
- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des AWS SDK pour JavaScript API.

PowerShell

Outils pour PowerShell V4

Exemple 1 : crée un nouveau coffre-fort pour le compte de l'utilisateur. Comme aucune valeur n'a été fournie au AccountId paramètre -, les applets de commande utilisent la valeur par défaut « - » pour indiquer le compte courant.

```
New-GLCVault -VaultName myvault
```

Sortie :

```
/01234567812/vaults/myvault
```

- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des Outils AWS pour PowerShell applets de commande (V4).

Outils pour PowerShell V5

Exemple 1 : crée un nouveau coffre-fort pour le compte de l'utilisateur. Comme aucune valeur n'a été fournie au AccountId paramètre -, les applets de commande utilisent la valeur par défaut « - » pour indiquer le compte courant.

```
New-GLCVault -VaultName myvault
```

Sortie :

```
/01234567812/vaults/myvault
```

- Pour plus de détails sur l'API, reportez-vous CreateVaultà la section Référence des Outils AWS pour PowerShell applets de commande (V5).

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    def create_vault(self, vault_name):
        """
        Creates a vault.

        :param vault_name: The name to give the vault.
        :return: The newly created vault.
        """
        try:
            vault = self.glacier_resource.create_vault(vaultName=vault_name)
            logger.info("Created vault %s.", vault_name)
        except ClientError:
            logger.exception("Couldn't create vault %s.", vault_name)
            raise
        else:
            return vault
```

- Pour plus de détails sur l'API, consultez CreateVaultle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`DeleteArchive`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DeleteArchive`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Obtenir le contenu d'une archive et supprimer l'archive

CLI

AWS CLI

Pour supprimer une archive d'un coffre-fort

L'exemple `delete-archive` suivant supprime l'archive spécifiée de `example_vault`.

```
aws glacier delete-archive \
    --account-id 111122223333  \
    --vault-name example_vault \
    --archive-id Sc0u9ZP8yaWkmh-XGlIvAVprtLhaLCGnNwNl5I5x9HqPIkX5mjc0DrId3Ln-Gi_k2HzmlIDZUz117KSdVMdMXLuFWi9PJUitxWO73edQ43eTlMWkH0pd9zVSAuV_XXZBVhKhyGhJ7w
```

Cette commande ne produit aucune sortie.

- Pour plus de détails sur l'API, reportez-vous DeleteArchiveà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.DeleteArchiveRequest;
import software.amazon.awssdk.services.glacier.model.GlacierException;

/**
```

```
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DeleteArchive {
    public static void main(String[] args) {
        final String usage = """

                Usage:    <vaultName> <accountId> <archiveId>

                Where:
                   vaultName - The name of the vault that contains the archive to
 delete.
                   accountId - The account ID value.
                   archiveId - The archive ID value.
                """;

        if (args.length != 3) {
            System.out.println(usage);
            System.exit(1);
        }

        String vaultName = args[0];
        String accountId = args[1];
        String archiveId = args[2];
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        deleteGlacierArchive(glacier, vaultName, accountId, archiveId);
        glacier.close();
    }

    public static void deleteGlacierArchive(GlacierClient glacier, String
 vaultName, String accountId,
            String archiveId) {
        try {
            DeleteArchiveRequest delArcRequest = DeleteArchiveRequest.builder()
                    .vaultName(vaultName)
                    .accountId(accountId)
```

```
                    .archiveId(archiveId)
                    .build();

            glacier.deleteArchive(delArcRequest);
            System.out.println("The archive was deleted.");

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous DeleteArchiveà la section Référence des AWS SDK for Java 2.x API.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def delete_archive(archive):
        """
        Deletes an archive from a vault.
```

```
        :param archive: The archive to delete.
        """
        try:
            archive.delete()
            logger.info(
                "Deleted archive %s from vault %s.", archive.id,
archive.vault_name
            )
        except ClientError:
            logger.exception("Couldn't delete archive %s.", archive.id)
            raise
```

- Pour plus de détails sur l'API, consultez DeleteArchivele AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`DeleteVault`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DeleteVault`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Obtenir le contenu d'une archive et supprimer l'archive

CLI

AWS CLI

La commande suivante supprime un coffre-fort nommé `my-vault` :

```
aws glacier delete-vault --vault-name my-vault --account-id -
```

Cette commande ne génère pas de sortie. Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous DeleteVaultà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.DeleteVaultRequest;
import software.amazon.awssdk.services.glacier.model.GlacierException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DeleteVault {
    public static void main(String[] args) {

        final String usage = """

                Usage:    <vaultName>

                Where:
                   vaultName - The name of the vault to delete.\s
                """;
```

```
        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String vaultName = args[0];
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        deleteGlacierVault(glacier, vaultName);
        glacier.close();
    }

    public static void deleteGlacierVault(GlacierClient glacier, String
 vaultName) {
        try {
            DeleteVaultRequest delVaultRequest = DeleteVaultRequest.builder()
                    .vaultName(vaultName)
                    .build();

            glacier.deleteVault(delVaultRequest);
            System.out.println("The vault was deleted!");

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous DeleteVaultà la section Référence des AWS SDK for Java 2.x API.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def delete_vault(vault):
        """
        Deletes a vault.

        :param vault: The vault to delete.
        """
        try:
            vault.delete()
            logger.info("Deleted vault %s.", vault.name)
        except ClientError:
            logger.exception("Couldn't delete vault %s.", vault.name)
            raise
```

- Pour plus de détails sur l'API, consultez DeleteVaultle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`DeleteVaultNotifications`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DeleteVaultNotifications`.

CLI

AWS CLI

Pour supprimer les notifications SNS d'un coffre-fort

L'exemple `delete-vault-notifications` suivant supprime les notifications envoyées par Amazon Simple Notification Service (Amazon SNS) pour le coffre-fort spécifié.

```
aws glacier delete-vault-notifications \
    --account-id 111122223333 \
    --vault-name example_vault
```

Cette commande ne produit aucune sortie.

- Pour plus de détails sur l'API, reportez-vous DeleteVaultNotificationsà la section Référence des AWS CLI commandes.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
```

```
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def stop_notifications(notification):
        """
        Stops notifications to the configured Amazon SNS topic.

        :param notification: The notification configuration to remove.
        """
        try:
            notification.delete()
            logger.info("Notifications stopped.")
        except ClientError:
            logger.exception("Couldn't stop notifications.")
            raise
```

- Pour plus de détails sur l'API, consultez DeleteVaultNotificationsle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`DescribeJob`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeJob`.

CLI

AWS CLI

La commande suivante récupère des informations sur une tâche d'extraction d'inventaire sur un coffre-fort nommé `my-vault` :

```
aws glacier describe-job --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW
```

Sortie :

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
    "Completed": false,
    "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
    "Action": "InventoryRetrieval",
    "CreationDate": "2015-07-17T20:23:41.616Z",
    "StatusCode": "InProgress"
}
```

L'ID de tâche se trouve dans la sortie de `aws glacier initiate-job` et `aws glacier list-jobs`. Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous DescribeJobà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell V4

Exemple 1 : renvoi des détails de la tâche spécifiée. Lorsque la tâche est terminée avec succès, l'applet de commande GCJob Read-Output peut être utilisée pour récupérer le contenu de la tâche (une archive ou une liste d'inventaire) dans le système de fichiers local.

```
Get-GLCJob -VaultName myvault -JobId "op1x...JSbthM"
```

Sortie :

```
Action                       : ArchiveRetrieval
```

```
ArchiveId                     : o9O9j...X-TpIhQJw
ArchiveSHA256TreeHash         : 79f3ea754c02f58...dc57bf4395b
ArchiveSizeInBytes            : 38034480
Completed                     : False
CompletionDate                : 1/1/0001 12:00:00 AM
CreationDate                  : 12/13/2018 11:00:14 AM
InventoryRetrievalParameters :
InventorySizeInBytes          : 0
JobDescription                :
JobId                         : op1x...JSbthM
JobOutputPath                 :
OutputLocation                :
RetrievalByteRange            : 0-38034479
SelectParameters              :
SHA256TreeHash                : 79f3ea754c02f58...dc57bf4395b
SNSTopic                      :
StatusCode                    : InProgress
StatusMessage                 :
Tier                          : Standard
VaultARN                      : arn:aws:glacier:us-west-2:012345678912:vaults/test
```

- Pour plus de détails sur l'API, reportez-vous DescribeJobà la section Référence des Outils AWS pour PowerShell applets de commande (V4).

Outils pour PowerShell V5

Exemple 1 : renvoi des détails de la tâche spécifiée. Lorsque la tâche est terminée avec succès, l'applet de commande GCJob Read-Output peut être utilisée pour récupérer le contenu de la tâche (une archive ou une liste d'inventaire) dans le système de fichiers local.

```
Get-GLCJob -VaultName myvault -JobId "op1x...JSbthM"
```

Sortie :

```
Action                        : ArchiveRetrieval
ArchiveId                     : o9O9j...X-TpIhQJw
ArchiveSHA256TreeHash         : 79f3ea754c02f58...dc57bf4395b
ArchiveSizeInBytes            : 38034480
Completed                     : False
CompletionDate                : 1/1/0001 12:00:00 AM
CreationDate                  : 12/13/2018 11:00:14 AM
InventoryRetrievalParameters :
InventorySizeInBytes          : 0
```

```
JobDescription              :
JobId                       : op1x...JSbthM
JobOutputPath               :
OutputLocation              :
RetrievalByteRange          : 0-38034479
SelectParameters            :
SHA256TreeHash              : 79f3ea754c02f58...dc57bf4395b
SNSTopic                    :
StatusCode                  : InProgress
StatusMessage               :
Tier                        : Standard
VaultARN                    : arn:aws:glacier:us-west-2:012345678912:vaults/test
```

- Pour plus de détails sur l'API, reportez-vous DescribeJobà la section Référence des Outils AWS pour PowerShell applets de commande (V5).

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def get_job_status(job):
        """
        Gets the status of a job.
```

```
        :param job: The job to query.
        :return: The current status of the job.
        """
        try:
            job.load()
            logger.info(
                "Job %s is performing action %s and has status %s.",
                job.id,
                job.action,
                job.status_code,
            )
        except ClientError:
            logger.exception("Couldn't get status for job %s.", job.id)
            raise
        else:
            return job.status_code
```

- Pour plus de détails sur l'API, consultez DescribeJoble AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`DescribeVault`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeVault`.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
```

```
    /// Describe an Amazon S3 Glacier vault.
    /// </summary>
    /// <param name="vaultName">The name of the vault to describe.</param>
    /// <returns>The Amazon Resource Name (ARN) of the vault.</returns>
    public async Task<string> DescribeVaultAsync(string vaultName)
    {
        var request = new DescribeVaultRequest
        {
            AccountId = "-",
            VaultName = vaultName,
        };

        var response = await _glacierService.DescribeVaultAsync(request);

        // Display the information about the vault.
        Console.WriteLine($"{response.VaultName}\tARN: {response.VaultARN}");
        Console.WriteLine($"Created on: {response.CreationDate}\tNumber
of Archives: {response.NumberOfArchives}\tSize (in bytes):
{response.SizeInBytes}");
        if (response.LastInventoryDate != DateTime.MinValue)
        {
            Console.WriteLine($"Last inventory: {response.LastInventoryDate}");
        }

        return response.VaultARN;
    }
```

- Pour plus de détails sur l'API, reportez-vous DescribeVaultà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante récupère les données relatives à un coffre-fort nommé `my-vault` :

```
aws glacier describe-vault --vault-name my-vault --account-id -
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous DescribeVaultà la section Référence des AWS CLI commandes.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`GetJobOutput`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `GetJobOutput`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Obtenir le contenu d'une archive et supprimer l'archive

CLI

AWS CLI

La commande suivante enregistre la sortie d'une tâche d'inventaire du coffre-fort dans un fichier du répertoire actuel nommé `output.json` :

```
aws glacier get-job-output --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW output.json
```

Le `job-id` est disponible dans la sortie de `aws glacier list-jobs`. Notez que le nom du fichier de sortie est un argument positionnel qui n'est pas préfixé par un nom d'option. Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

Sortie :

```
{
    "status": 200,
    "acceptRanges": "bytes",
    "contentType": "application/json"
}
```

`output.json`:

```
{"VaultARN":"arn:aws:glacier:us-west-2:0123456789012:vaults/
my-vault","InventoryDate":"2015-04-07T00:26:18Z","ArchiveList":
[{"ArchiveId":"kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw","ArchiveDescription":"multipart
 upload
 test","CreationDate":"2015-04-06T22:24:34Z","Size":3145728,"SHA256TreeHash":"9628195fcdb
```

- Pour plus de détails sur l'API, reportez-vous GetJobOutputà la section Référence des AWS CLI commandes.

PowerShell

Outils pour PowerShell V4

Exemple 1 : télécharge le contenu de l’archive dont l’extraction était planifiée dans le cadre de la tâche spécifiée et stocke le contenu dans un fichier sur disque. Le téléchargement valide le checksum pour vous, s’il est disponible. Si vous le souhaitez, la réponse complète, y compris le checksum, peut être renvoyée en spécifiant **`-Select '*'`**.

```
Read-GLCJobOutput -VaultName myvault -JobId "HSWjArc...Zq2XLiW" -FilePath "c:
\temp\blue.bin"
```

- Pour plus de détails sur l'API, reportez-vous GetJobOutputà la section Référence des Outils AWS pour PowerShell applets de commande (V4).

Outils pour PowerShell V5

Exemple 1 : télécharge le contenu de l’archive dont l’extraction était planifiée dans le cadre de la tâche spécifiée et stocke le contenu dans un fichier sur disque. Le téléchargement valide le checksum pour vous, s’il est disponible. Si vous le souhaitez, la réponse complète, y compris le checksum, peut être renvoyée en spécifiant **`-Select '*'`**.

```
Read-GLCJobOutput -VaultName myvault -JobId "HSWjArc...Zq2XLiW" -FilePath "c:
\temp\blue.bin"
```

- Pour plus de détails sur l'API, reportez-vous GetJobOutputà la section Référence des Outils AWS pour PowerShell applets de commande (V5).

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def get_job_output(job):
        """
        Gets the output of a job, such as a vault inventory or the contents of an
        archive.

        :param job: The job to get output from.
        :return: The job output, in bytes.
        """
        try:
            response = job.get_output()
            out_bytes = response["body"].read()
            logger.info("Read %s bytes from job %s.", len(out_bytes), job.id)
            if "archiveDescription" in response:
                logger.info(
                    "These bytes are described as '%s'",
 response["archiveDescription"]
                )
        except ClientError:
            logger.exception("Couldn't get output for job %s.", job.id)
            raise
        else:
```

```
            return out_bytes
```

- Pour plus de détails sur l'API, consultez GetJobOutputle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`GetVaultNotifications`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `GetVaultNotifications`.

CLI

AWS CLI

La commande suivante obtient une description de la configuration des notifications pour un coffre-fort nommé `my-vault` :

```
aws glacier get-vault-notifications --account-id - --vault-name my-vault
```

Sortie :

```
{
    "vaultNotificationConfig": {
        "Events": [
            "InventoryRetrievalCompleted",
            "ArchiveRetrievalCompleted"
        ],
        "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault"
    }
}
```

Si aucune notification n’a été configurée pour le coffre-fort, une erreur est renvoyée. Amazon Glacier nécessite un argument d’ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d’union pour spécifier le compte en cours d’utilisation.

- Pour plus de détails sur l'API, reportez-vous GetVaultNotificationsà la section Référence des AWS CLI commandes.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def get_notification(vault):
        """
        Gets the currently notification configuration for a vault.

        :param vault: The vault to query.
        :return: The notification configuration for the specified vault.
        """
        try:
            notification = vault.Notification()
            logger.info(
                "Vault %s notifies %s on %s events.",
                vault.name,
                notification.sns_topic,
                notification.events,
            )
        except ClientError:
```

```
            logger.exception("Couldn't get notification data for %s.",
vault.name)
            raise
        else:
            return notification
```

- Pour plus de détails sur l'API, consultez GetVaultNotificationsle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`InitiateJob`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `InitiateJob`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Archivage d'un fichier, obtention de notifications et lancement d'une tâche

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Extrayez une archive d'un coffre-fort. Cet exemple utilise la ArchiveTransferManager classe. Pour plus de détails sur l'API, voir ArchiveTransferManager.

```
    /// <summary>
    /// Download an archive from an Amazon S3 Glacier vault using the Archive
```

```
    /// Transfer Manager.
    /// </summary>
    /// <param name="vaultName">The name of the vault containing the object.</
param>
    /// <param name="archiveId">The Id of the archive to download.</param>
    /// <param name="localFilePath">The local directory where the file will
    /// be stored after download.</param>
    /// <returns>Async Task.</returns>
    public async Task<bool> DownloadArchiveWithArchiveManagerAsync(string
 vaultName, string archiveId, string localFilePath)
    {
        try
        {
            var manager = new ArchiveTransferManager(_glacierService);

            var options = new DownloadOptions
            {
                StreamTransferProgress = Progress!,
            };

            // Download an archive.
            Console.WriteLine("Initiating the archive retrieval job and then
 polling SQS queue for the archive to be available.");
            Console.WriteLine("When the archive is available, downloading will
 begin.");
            await manager.DownloadAsync(vaultName, archiveId, localFilePath,
 options);

            return true;
        }
        catch (AmazonGlacierException ex)
        {
            Console.WriteLine(ex.Message);
            return false;
        }
    }

    /// <summary>
    /// Event handler to track the progress of the Archive Transfer Manager.
    /// </summary>
    /// <param name="sender">The object that raised the event.</param>
    /// <param name="args">The argument values from the object that raised the
    /// event.</param>
    static void Progress(object sender, StreamTransferProgressArgs args)
```

```
    {
        if (args.PercentDone != _currentPercentage)
        {
            _currentPercentage = args.PercentDone;
            Console.WriteLine($"Downloaded {_currentPercentage}%");
        }
    }
```

- Pour plus de détails sur l'API, reportez-vous InitiateJobà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante lance une tâche visant à obtenir un inventaire du coffre-fort `my-vault` :

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
 '{"Type": "inventory-retrieval"}'
```

Sortie :

```
{
    "location": "/0123456789012/vaults/my-vault/jobs/
zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
    "jobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW"
}
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

La commande suivante lance une tâche visant à extraire une archive depuis le coffre-fort `my-vault` :

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-
parameters file://job-archive-retrieval.json
```

`job-archive-retrieval.json` est un fichier JSON situé dans le dossier local, qui indique le type de tâche, l'ID d'archive et certains paramètres facultatifs :

```
{
  "Type": "archive-retrieval",
  "ArchiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
  "Description": "Retrieve archive on 2015-07-17",
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-topic"
}
```

IDs Les archives sont disponibles dans la sortie de `aws glacier upload-archive` et`aws glacier get-job-output`.

Sortie :

```
{
    "location": "/011685312445/vaults/mwunderl/jobs/l7IL5-
EkXyEY9Ws95fClzIbk2O5uLYaFdAYOi-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav",
    "jobId": "l7IL5-EkXy2O5uLYaFdAYOiEY9Ws95fClzIbk-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav"
}
```

Pour plus d'informations sur le format des paramètres de tâche, consultez Lancement d'une tâche dans la Référence des API d'Amazon Glacier.

- Pour plus de détails sur l'API, reportez-vous InitiateJobà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Extrayez un inventaire de coffre-fort.

```
import software.amazon.awssdk.core.ResponseBytes;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.JobParameters;
import software.amazon.awssdk.services.glacier.model.InitiateJobResponse;
import software.amazon.awssdk.services.glacier.model.GlacierException;
import software.amazon.awssdk.services.glacier.model.InitiateJobRequest;
import software.amazon.awssdk.services.glacier.model.DescribeJobRequest;
import software.amazon.awssdk.services.glacier.model.DescribeJobResponse;
import software.amazon.awssdk.services.glacier.model.GetJobOutputRequest;
import software.amazon.awssdk.services.glacier.model.GetJobOutputResponse;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class ArchiveDownload {
    public static void main(String[] args) {

        final String usage = """

                Usage:    <vaultName> <accountId> <path>

                Where:
                   vaultName - The name of the vault.
                   accountId - The account ID value.
                   path - The path where the file is written to.
                """;

        if (args.length != 3) {
            System.out.println(usage);
            System.exit(1);
        }
```

```
        String vaultName = args[0];
        String accountId = args[1];
        String path = args[2];
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        String jobNum = createJob(glacier, vaultName, accountId);
        checkJob(glacier, jobNum, vaultName, accountId, path);
        glacier.close();
    }

    public static String createJob(GlacierClient glacier, String vaultName,
String accountId) {
        try {
            JobParameters job = JobParameters.builder()
                    .type("inventory-retrieval")
                    .build();

            InitiateJobRequest initJob = InitiateJobRequest.builder()
                    .jobParameters(job)
                    .accountId(accountId)
                    .vaultName(vaultName)
                    .build();

            InitiateJobResponse response = glacier.initiateJob(initJob);
            System.out.println("The job ID is: " + response.jobId());
            System.out.println("The relative URI path of the job is: " +
response.location());
            return response.jobId();

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);

        }
        return "";
    }

    // Poll S3 Glacier = Polling a Job may take 4-6 hours according to the
    // Documentation.
    public static void checkJob(GlacierClient glacier, String jobId, String name,
String account, String path) {
```

```
        try {
            boolean finished = false;
            String jobStatus;
            int yy = 0;

            while (!finished) {
                DescribeJobRequest jobRequest = DescribeJobRequest.builder()
                        .jobId(jobId)
                        .accountId(account)
                        .vaultName(name)
                        .build();

                DescribeJobResponse response = glacier.describeJob(jobRequest);
                jobStatus = response.statusCodeAsString();

                if (jobStatus.compareTo("Succeeded") == 0)
                    finished = true;
                else {
                    System.out.println(yy + " status is: " + jobStatus);
                    Thread.sleep(1000);
                }
                yy++;
            }

            System.out.println("Job has Succeeded");
            GetJobOutputRequest jobOutputRequest = GetJobOutputRequest.builder()
                    .jobId(jobId)
                    .vaultName(name)
                    .accountId(account)
                    .build();

            ResponseBytes<GetJobOutputResponse> objectBytes =
glacier.getJobOutputAsBytes(jobOutputRequest);
            // Write the data to a local file.
            byte[] data = objectBytes.asByteArray();
            File myFile = new File(path);
            OutputStream os = new FileOutputStream(myFile);
            os.write(data);
            System.out.println("Successfully obtained bytes from a Glacier
vault");
            os.close();

        } catch (GlacierException | InterruptedException | IOException e) {
            System.out.println(e.getMessage());
```

```
            System.exit(1);

        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous InitiateJobà la section Référence des AWS SDK for Java 2.x API.

PowerShell

Outils pour PowerShell V4

Exemple 1 : démarre une tâche pour extraire une archive dans le coffre-fort spécifié appartenant à l'utilisateur. L'état de la tâche peut être vérifié à l'aide de l'GLCJob applet de commande Get-. Lorsque le travail est terminé avec succès, l'applet de commande GCJob Read-Output peut être utilisée pour récupérer le contenu de l'archive dans le système de fichiers local.

```
Start-GLCJob -VaultName myvault -JobType "archive-retrieval" -JobDescription
 "archive retrieval" -ArchiveId "o9O9j...TX-TpIhQJw"
```

Sortie :

```
JobId            JobOutputPath Location
-----            ------------- --------
op1x...JSbthM                  /012345678912/vaults/test/jobs/
op1xe...I4HqCHkSJSbthM
```

- Pour plus de détails sur l'API, reportez-vous InitiateJobà la section Référence des Outils AWS pour PowerShell applets de commande (V4).

Outils pour PowerShell V5

Exemple 1 : démarre une tâche pour extraire une archive dans le coffre-fort spécifié appartenant à l'utilisateur. L'état de la tâche peut être vérifié à l'aide de l'GLCJob applet de commande Get-. Lorsque le travail est terminé avec succès, l'applet de commande GCJob Read-Output peut être utilisée pour récupérer le contenu de l'archive dans le système de fichiers local.

```
Start-GLCJob -VaultName myvault -JobType "archive-retrieval" -JobDescription
 "archive retrieval" -ArchiveId "o9O9j...TX-TpIhQJw"
```

Sortie :

```
JobId            JobOutputPath Location
-----            ------------- --------
op1x...JSbthM                  /012345678912/vaults/test/jobs/
op1xe...I4HqCHkSJSbthM
```

- Pour plus de détails sur l'API, reportez-vous InitiateJobà la section Référence des Outils AWS pour PowerShell applets de commande (V5).

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Extrayez un inventaire de coffre-fort.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def initiate_inventory_retrieval(vault):
        """
        Initiates an inventory retrieval job. The inventory describes the
 contents
```

```
        of the vault. Standard retrievals typically complete within 3–5 hours.
        When the job completes, you can get the inventory by calling
 get_output().

        :param vault: The vault to inventory.
        :return: The inventory retrieval job.
        """
        try:
            job = vault.initiate_inventory_retrieval()
            logger.info("Started %s job with ID %s.", job.action, job.id)
        except ClientError:
            logger.exception("Couldn't start job on vault %s.", vault.name)
            raise
        else:
            return job
```

Extrayez d'une archive d'un coffre-fort.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def initiate_archive_retrieval(archive):
        """
        Initiates an archive retrieval job. Standard retrievals typically
 complete
        within 3–5 hours. When the job completes, you can get the archive
 contents
        by calling get_output().

        :param archive: The archive to retrieve.
        :return: The archive retrieval job.
        """
        try:
```

```
            job = archive.initiate_archive_retrieval()
            logger.info("Started %s job with ID %s.", job.action, job.id)
        except ClientError:
            logger.exception("Couldn't start job on archive %s.", archive.id)
            raise
        else:
            return job
```

- Pour plus de détails sur l'API, consultez InitiateJoble AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`ListJobs`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `ListJobs`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans les exemples de code suivants :

- Archivage d'un fichier, obtention de notifications et lancement d'une tâche
- Obtenir le contenu d'une archive et supprimer l'archive

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
```

```
    /// List Amazon S3 Glacier jobs.
    /// </summary>
    /// <param name="vaultName">The name of the vault to list jobs for.</param>
    /// <returns>A list of Amazon S3 Glacier jobs.</returns>
    public async Task<List<GlacierJobDescription>> ListJobsAsync(string
 vaultName)
    {
        var request = new ListJobsRequest
        {
            // Using a hyphen "-" for the Account Id will
            // cause the SDK to use the Account Id associated
            // with the current account.
            AccountId = "-",
            VaultName = vaultName,
        };

        var response = await _glacierService.ListJobsAsync(request);

        return response.JobList;
    }
```

- Pour plus de détails sur l'API, reportez-vous ListJobsà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante répertorie les tâches en cours et récemment terminées pour un coffre-fort nommé `my-vault` :

```
aws glacier list-jobs --account-id - --vault-name my-vault
```

Sortie :

```
{
    "JobList": [
        {
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-
vault",
```

```
            "RetrievalByteRange": "0-3145727",
            "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
            "Completed": false,
            "SHA256TreeHash":
 "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
            "JobId": "l7IL5-EkXyEY9Ws95fClzIbk2O5uLYaFdAYOi-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav",
            "ArchiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--
zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
            "JobDescription": "Retrieve archive on 2015-07-17",
            "ArchiveSizeInBytes": 3145728,
            "Action": "ArchiveRetrieval",
            "ArchiveSHA256TreeHash":
 "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
            "CreationDate": "2015-07-17T21:16:13.840Z",
            "StatusCode": "InProgress"
        },
        {
            "InventoryRetrievalParameters": {
                "Format": "JSON"
            },
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-
vault",
            "Completed": false,
            "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
            "Action": "InventoryRetrieval",
            "CreationDate": "2015-07-17T20:23:41.616Z",
            "StatusCode": ""InProgress""
        }
    ]
}
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous ListJobsà la section Référence des AWS CLI commandes.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def list_jobs(vault, job_type):
        """
        Lists jobs by type for the specified vault.

        :param vault: The vault to query.
        :param job_type: The type of job to list.
        :return: The list of jobs of the requested type.
        """
        job_list = []
        try:
            if job_type == "all":
                jobs = vault.jobs.all()
            elif job_type == "in_progress":
                jobs = vault.jobs_in_progress.all()
            elif job_type == "completed":
                jobs = vault.completed_jobs.all()
            elif job_type == "succeeded":
                jobs = vault.succeeded_jobs.all()
            elif job_type == "failed":
                jobs = vault.failed_jobs.all()
            else:
```

```
                jobs = []
                logger.warning("%s isn't a type of job I can get.", job_type)
            for job in jobs:
                job_list.append(job)
                logger.info("Got %s %s job %s.", job_type, job.action, job.id)
        except ClientError:
            logger.exception("Couldn't get %s jobs from %s.", job_type,
vault.name)
            raise
        else:
            return job_list
```

- Pour plus de détails sur l'API, consultez ListJobsle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`ListTagsForVault`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `ListTagsForVault`.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
    /// List tags for an Amazon S3 Glacier vault.
    /// </summary>
    /// <param name="vaultName">The name of the vault to list tags for.</param>
    /// <returns>A dictionary listing the tags attached to each object in the
    /// vault and its tags.</returns>
```

```
    public async Task<Dictionary<string, string>> ListTagsForVaultAsync(string
 vaultName)
    {
        var request = new ListTagsForVaultRequest
        {
            // Using a hyphen "-" for the Account Id will
            // cause the SDK to use the Account Id associated
            // with the default user.
            AccountId = "-",
            VaultName = vaultName,
        };

        var response = await _glacierService.ListTagsForVaultAsync(request);

        return response.Tags;
    }
```

- Pour plus de détails sur l'API, reportez-vous ListTagsForVaultà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante répertorie les balises appliquées à un coffre-fort nommé `my-vault` :

```
aws glacier list-tags-for-vault --account-id - --vault-name my-vault
```

Sortie :

```
{
    "Tags": {
        "date": "july2015",
        "id": "1234"
    }
}
```

Amazon Glacier nécessite un argument d’ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d’union pour spécifier le compte en cours d’utilisation.

- Pour plus de détails sur l'API, reportez-vous ListTagsForVaultà la section Référence des AWS CLI commandes.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`ListVaults`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `ListVaults`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Archivage d'un fichier, obtention de notifications et lancement d'une tâche

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
/// <summary>
/// List the Amazon S3 Glacier vaults associated with the current account.
/// </summary>
/// <returns>A list containing information about each vault.</returns>
public async Task<List<DescribeVaultOutput>> ListVaultsAsync()
{
    var glacierVaultPaginator = _glacierService.Paginators.ListVaults(
        new ListVaultsRequest { AccountId = "-" });
    var vaultList = new List<DescribeVaultOutput>();

    await foreach (var vault in glacierVaultPaginator.VaultList)
    {
```

```
            vaultList.Add(vault);
        }

        return vaultList;
    }
```

- Pour plus de détails sur l'API, reportez-vous ListVaultsà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante répertorie les coffres-forts dans le compte et la région par défaut :

```
aws glacier list-vaults --account-id -
```

Sortie :

```
{
    "VaultList": [
        {
            "SizeInBytes": 3178496,
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-
vault",
            "LastInventoryDate": "2015-04-07T00:26:19.028Z",
            "VaultName": "my-vault",
            "NumberOfArchives": 1,
            "CreationDate": "2015-04-06T21:23:45.708Z"
        }
    ]
}
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous ListVaultsà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.model.ListVaultsRequest;
import software.amazon.awssdk.services.glacier.model.ListVaultsResponse;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.DescribeVaultOutput;
import software.amazon.awssdk.services.glacier.model.GlacierException;
import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class ListVaults {
    public static void main(String[] args) {
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        listAllVault(glacier);
        glacier.close();
    }

    public static void listAllVault(GlacierClient glacier) {
        boolean listComplete = false;
        String newMarker = null;
        int totalVaults = 0;
        System.out.println("Your Amazon Glacier vaults:");
```

```
        try {
            while (!listComplete) {
                ListVaultsResponse response = null;
                if (newMarker != null) {
                    ListVaultsRequest request = ListVaultsRequest.builder()
                            .marker(newMarker)
                            .build();

                    response = glacier.listVaults(request);
                } else {
                    ListVaultsRequest request = ListVaultsRequest.builder()
                            .build();
                    response = glacier.listVaults(request);
                }

                List<DescribeVaultOutput> vaultList = response.vaultList();
                for (DescribeVaultOutput v : vaultList) {
                    totalVaults += 1;
                    System.out.println("* " + v.vaultName());
                }

                // Check for further results.
                newMarker = response.marker();
                if (newMarker == null) {
                    listComplete = true;
                }
            }

            if (totalVaults == 0) {
                System.out.println("No vaults found.");
            }

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, reportez-vous ListVaultsà la section Référence des AWS SDK for Java 2.x API.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    def list_vaults(self):
        """
        Lists vaults for the current account.
        """
        try:
            for vault in self.glacier_resource.vaults.all():
                logger.info("Got vault %s.", vault.name)
        except ClientError:
            logger.exception("Couldn't list vaults.")
            raise
```

- Pour plus de détails sur l'API, consultez ListVaultsle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`SetVaultNotifications`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `SetVaultNotifications`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Archivage d'un fichier, obtention de notifications et lancement d'une tâche

CLI

AWS CLI

La commande suivante configure les notifications SNS pour un coffre-fort nommé `my-vault` :

```
aws glacier set-vault-notifications --account-id - --vault-name my-vault --vault-notification-config file://notificationconfig.json
```

`notificationconfig.json` est un fichier JSON dans le dossier actuel qui indique une rubrique SNS et les événements à publier :

```
{
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
  "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

- Pour plus de détails sur l'API, reportez-vous SetVaultNotificationsà la section Référence des AWS CLI commandes.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    def set_notifications(self, vault, sns_topic_arn):
        """
        Sets an Amazon Simple Notification Service (Amazon SNS) topic as a target
        for notifications. Amazon S3 Glacier publishes messages to this topic for
        the configured list of events.

        :param vault: The vault to set up to publish notifications.
        :param sns_topic_arn: The Amazon Resource Name (ARN) of the topic that
                              receives notifications.
        :return: Data about the new notification configuration.
        """
        try:
            notification = self.glacier_resource.Notification("-", vault.name)
            notification.set(
                vaultNotificationConfig={
                    "SNSTopic": sns_topic_arn,
                    "Events": [
                        "ArchiveRetrievalCompleted",
                        "InventoryRetrievalCompleted",
                    ],
                }
            )
```

```
            logger.info(
                "Notifications will be sent to %s for events %s from %s.",
                notification.sns_topic,
                notification.events,
                notification.vault_name,
            )
        except ClientError:
            logger.exception(
                "Couldn't set notifications to %s on %s.", sns_topic_arn,
vault.name
            )
            raise
        else:
            return notification
```

- Pour plus de détails sur l'API, consultez SetVaultNotificationsle AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`UploadArchive`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `UploadArchive`.

Les exemples d’actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l’exemple de code suivant :

- Archivage d’un fichier, obtention de notifications et lancement d’une tâche

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Upload an object to an Amazon S3 Glacier vault.
   /// </summary>
   /// <param name="vaultName">The name of the Amazon S3 Glacier vault to upload
   /// the archive to.</param>
   /// <param name="archiveFilePath">The file path of the archive to upload to
the vault.</param>
   /// <returns>A Boolean value indicating the success of the action.</returns>
   public async Task<string> UploadArchiveWithArchiveManager(string vaultName,
string archiveFilePath)
   {
       try
       {
           var manager = new ArchiveTransferManager(_glacierService);

           // Upload an archive.
           var response = await manager.UploadAsync(vaultName, "upload archive
test", archiveFilePath);
           return response.ArchiveId;
       }
       catch (AmazonGlacierException ex)
       {
           Console.WriteLine(ex.Message);
           return string.Empty;
       }
   }
```

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS SDK pour .NET API.

CLI

AWS CLI

La commande suivante charge une archive du dossier actuel nommé `archive.zip` vers un coffre-fort nommé `my-vault` :

```
aws glacier upload-archive --account-id - --vault-name my-vault --
body archive.zip
```

Sortie :

```
{
    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--
zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
    "checksum":
 "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/0123456789012/vaults/my-vault/archives/
kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
}
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

Pour récupérer une archive chargée, lancez une tâche d'extraction à l'aide de la commande aws glacier initiate-job.

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.glacier.GlacierClient;
import software.amazon.awssdk.services.glacier.model.UploadArchiveRequest;
import software.amazon.awssdk.services.glacier.model.UploadArchiveResponse;
import software.amazon.awssdk.services.glacier.model.GlacierException;
import java.io.File;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.io.FileInputStream;
import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class UploadArchive {

    static final int ONE_MB = 1024 * 1024;

    public static void main(String[] args) {
        final String usage = """

                Usage:   <strPath> <vaultName>\s

                Where:
                   strPath - The path to the archive to upload (for example, C:\
\AWS\\test.pdf).
                   vaultName - The name of the vault.
                """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String strPath = args[0];
```

```
        String vaultName = args[1];
        File myFile = new File(strPath);
        Path path = Paths.get(strPath);
        GlacierClient glacier = GlacierClient.builder()
                .region(Region.US_EAST_1)
                .build();

        String archiveId = uploadContent(glacier, path, vaultName, myFile);
        System.out.println("The ID of the archived item is " + archiveId);
        glacier.close();
    }

    public static String uploadContent(GlacierClient glacier, Path path, String
vaultName, File myFile) {
        // Get an SHA-256 tree hash value.
        String checkVal = computeSHA256(myFile);
        try {
            UploadArchiveRequest uploadRequest = UploadArchiveRequest.builder()
                    .vaultName(vaultName)
                    .checksum(checkVal)
                    .build();

            UploadArchiveResponse res = glacier.uploadArchive(uploadRequest,
path);
            return res.archiveId();

        } catch (GlacierException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }

    private static String computeSHA256(File inputFile) {
        try {
            byte[] treeHash = computeSHA256TreeHash(inputFile);
            System.out.printf("SHA-256 tree hash = %s\n", toHex(treeHash));
            return toHex(treeHash);

        } catch (IOException ioe) {
            System.err.format("Exception when reading from file %s: %s",
inputFile, ioe.getMessage());
            System.exit(-1);
```

```
        } catch (NoSuchAlgorithmException nsae) {
            System.err.format("Cannot locate MessageDigest algorithm for SHA-256:
%s", nsae.getMessage());
            System.exit(-1);
        }
        return "";
    }

    public static byte[] computeSHA256TreeHash(File inputFile) throws
IOException,
            NoSuchAlgorithmException {

        byte[][] chunkSHA256Hashes = getChunkSHA256Hashes(inputFile);
        return computeSHA256TreeHash(chunkSHA256Hashes);
    }

    /**
     * Computes an SHA256 checksum for each 1 MB chunk of the input file. This
     * includes the checksum for the last chunk, even if it's smaller than 1 MB.
     */
    public static byte[][] getChunkSHA256Hashes(File file) throws IOException,
            NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");
        long numChunks = file.length() / ONE_MB;
        if (file.length() % ONE_MB > 0) {
            numChunks++;
        }

        if (numChunks == 0) {
            return new byte[][] { md.digest() };
        }

        byte[][] chunkSHA256Hashes = new byte[(int) numChunks][];
        FileInputStream fileStream = null;

        try {
            fileStream = new FileInputStream(file);
            byte[] buff = new byte[ONE_MB];

            int bytesRead;
            int idx = 0;

            while ((bytesRead = fileStream.read(buff, 0, ONE_MB)) > 0) {
```

```
                md.reset();
                md.update(buff, 0, bytesRead);
                chunkSHA256Hashes[idx++] = md.digest();
            }

            return chunkSHA256Hashes;

        } finally {
            if (fileStream != null) {
                try {
                    fileStream.close();
                } catch (IOException ioe) {
                    System.err.printf("Exception while closing %s.\n %s",
file.getName(),
                            ioe.getMessage());
                }
            }
        }
    }

    /**
     * Computes the SHA-256 tree hash for the passed array of 1 MB chunk
     * checksums.
     */
    public static byte[] computeSHA256TreeHash(byte[][] chunkSHA256Hashes)
            throws NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");
        byte[][] prevLvlHashes = chunkSHA256Hashes;
        while (prevLvlHashes.length > 1) {
            int len = prevLvlHashes.length / 2;
            if (prevLvlHashes.length % 2 != 0) {
                len++;
            }

            byte[][] currLvlHashes = new byte[len][];
            int j = 0;
            for (int i = 0; i < prevLvlHashes.length; i = i + 2, j++) {

                // If there are at least two elements remaining.
                if (prevLvlHashes.length - i > 1) {

                    // Calculate a digest of the concatenated nodes.
                    md.reset();
```

```
                    md.update(prevLvlHashes[i]);
                    md.update(prevLvlHashes[i + 1]);
                    currLvlHashes[j] = md.digest();

                } else { // Take care of the remaining odd chunk
                    currLvlHashes[j] = prevLvlHashes[i];
                }
            }

            prevLvlHashes = currLvlHashes;
        }

        return prevLvlHashes[0];
    }

    /**
     * Returns the hexadecimal representation of the input byte array
     */
    public static String toHex(byte[] data) {
        StringBuilder sb = new StringBuilder(data.length * 2);
        for (byte datum : data) {
            String hex = Integer.toHexString(datum & 0xFF);

            if (hex.length() == 1) {
                // Append leading zero.
                sb.append("0");
            }
            sb.append(hex);
        }
        return sb.toString().toLowerCase();
    }
}
```

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS SDK for Java 2.x API.

JavaScript

SDK pour JavaScript (v3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Créez le client.

```
const { GlacierClient } = require("@aws-sdk/client-glacier");
// Set the AWS Region.
const REGION = "REGION";
//Set the Redshift Service Object
const glacierClient = new GlacierClient({ region: REGION });
export { glacierClient };
```

Chargez l'archive.

```
// Load the SDK for JavaScript
import { UploadArchiveCommand } from "@aws-sdk/client-glacier";
import { glacierClient } from "./libs/glacierClient.js";

// Set the parameters
const vaultname = "VAULT_NAME"; // VAULT_NAME

// Create a new service object and buffer
const buffer = new Buffer.alloc(2.5 * 1024 * 1024); // 2.5MB buffer
const params = { vaultName: vaultname, body: buffer };

const run = async () => {
  try {
    const data = await glacierClient.send(new UploadArchiveCommand(params));
    console.log("Archive ID", data.archiveId);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error uploading archive!", err);
  }
};
```

```
run();
```

- Pour plus d'informations, consultez le Guide du développeur AWS SDK pour JavaScript.
- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS SDK pour JavaScript API.

SDK pour JavaScript (v2)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
// Load the SDK for JavaScript
var AWS = require("aws-sdk");
// Set the region
AWS.config.update({ region: "REGION" });

// Create a new service object and buffer
var glacier = new AWS.Glacier({ apiVersion: "2012-06-01" });
buffer = Buffer.alloc(2.5 * 1024 * 1024); // 2.5MB buffer

var params = { vaultName: "YOUR_VAULT_NAME", body: buffer };
// Call Glacier to upload the archive.
glacier.uploadArchive(params, function (err, data) {
  if (err) {
    console.log("Error uploading archive!", err);
  } else {
    console.log("Archive ID", data.archiveId);
  }
});
```

- Pour plus d'informations, consultez le Guide du développeur AWS SDK pour JavaScript.
- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des AWS SDK pour JavaScript API.

PowerShell

Outils pour PowerShell V4

Exemple 1 : télécharge un seul fichier dans le coffre-fort spécifié, en renvoyant l'ID de l'archive et la somme de contrôle calculée.

```
Write-GLCArchive -VaultName myvault -FilePath c:\temp\blue.bin
```

Sortie :

```
FilePath                  ArchiveId             Checksum
--------                  ---------             --------
C:\temp\blue.bin          o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
```

Exemple 2 : charge le contenu d'une hiérarchie de dossiers dans le coffre-fort spécifié dans le compte de l'utilisateur. Pour chaque fichier chargé, l'applet de commande émet le nom du fichier, l'ID d'archive correspondant et la somme de contrôle calculée de l'archive.

```
Write-GLCArchive -VaultName myvault -FolderPath . -Recurse
```

Sortie :

```
FilePath                  ArchiveId             Checksum
--------                  ---------             --------
C:\temp\blue.bin          o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
C:\temp\green.bin         qXAfOdSG...czo729UHXrw d50a1...9184b9
C:\temp\lum.bin           39aNifP3...q9nb8nZkFIg 28886...5c3e27
C:\temp\red.bin           vp7E6rU_...Ejk_HhjAxKA e05f7...4e34f5
C:\temp\Folder1\file1.txt _eRINlip...5Sxy7dD2BaA d0d2a...c8a3ba
C:\temp\Folder2\file2.iso -Ix3jlmu...iXiDh-XfOPA 7469e...3e86f1
```

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des Outils AWS pour PowerShell applets de commande (V4).

Outils pour PowerShell V5

Exemple 1 : télécharge un seul fichier dans le coffre-fort spécifié, en renvoyant l'ID de l'archive et la somme de contrôle calculée.

```
Write-GLCArchive -VaultName myvault -FilePath c:\temp\blue.bin
```

Sortie :

```
FilePath                    ArchiveId             Checksum
--------                    ---------             --------
C:\temp\blue.bin            o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
```

Exemple 2 : charge le contenu d'une hiérarchie de dossiers dans le coffre-fort spécifié dans le compte de l'utilisateur. Pour chaque fichier chargé, l'applet de commande émet le nom du fichier, l'ID d'archive correspondant et la somme de contrôle calculée de l'archive.

```
Write-GLCArchive -VaultName myvault -FolderPath . -Recurse
```

Sortie :

```
FilePath                    ArchiveId             Checksum
--------                    ---------             --------
C:\temp\blue.bin            o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
C:\temp\green.bin           qXAfOdSG...czo729UHXrw d50a1...9184b9
C:\temp\lum.bin             39aNifP3...q9nb8nZkFIg 28886...5c3e27
C:\temp\red.bin             vp7E6rU_...Ejk_HhjAxKA e05f7...4e34f5
C:\temp\Folder1\file1.txt   _eRINlip...5Sxy7dD2BaA d0d2a...c8a3ba
C:\temp\Folder2\file2.iso   -Ix3jlmu...iXiDh-XfOPA 7469e...3e86f1
```

- Pour plus de détails sur l'API, reportez-vous UploadArchiveà la section Référence des Outils AWS pour PowerShell applets de commande (V5).

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
```

```
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def upload_archive(vault, archive_description, archive_file):
        """
        Uploads an archive to a vault.

        :param vault: The vault where the archive is put.
        :param archive_description: A description of the archive.
        :param archive_file: The archive file to put in the vault.
        :return: The uploaded archive.
        """
        try:
            archive = vault.upload_archive(
                archiveDescription=archive_description, body=archive_file
            )
            logger.info(
                "Uploaded %s with ID %s to vault %s.",
                archive_description,
                archive.id,
                vault.name,
            )
        except ClientError:
            logger.exception(
                "Couldn't upload %s to %s.", archive_description, vault.name
            )
            raise
        else:
            return archive
```

- Pour plus de détails sur l'API, consultez UploadArchivele AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Utilisation **`UploadMultipartPart`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `UploadMultipartPart`.

CLI

AWS CLI

La commande suivante télécharge le premier Mio (1 024 x 1 024 octets) d'une archive :

```
aws glacier upload-multipart-part --body part1 --range 'bytes
 0-1048575/*' --account-id - --vault-name my-vault --upload-
id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ
```

Amazon Glacier nécessite un argument d'ID de compte pour effectuer des opérations, mais vous pouvez utiliser un trait d'union pour spécifier le compte en cours d'utilisation.

Le paramètre corps indique le chemin d'un fichier partiel sur le système de fichiers local. Le paramètre plage prend une plage de contenu HTTP indiquant les octets que la partie occupe dans l'archive terminée. L'ID de chargement est renvoyé par la commande `aws glacier initiate-multipart-upload` et peut également être obtenu avec `aws glacier list-multipart-uploads`.

Pour plus d'informations sur les téléchargements partitionnés vers Amazon Glacier à l'aide de l'interface AWS de ligne de commande, consultez la section Utilisation d'Amazon Glacier dans le guide de l'utilisateur de la AWS CLI.

- Pour plus de détails sur l'API, reportez-vous UploadMultipartPartà la section Référence des AWS CLI commandes.

JavaScript

SDK pour JavaScript (v2)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Créez un chargement partitionné de blocs d'un mégaoctet d'un objet Buffer.

```
// Create a new service object and some supporting variables
var glacier = new AWS.Glacier({ apiVersion: "2012-06-01" }),
  vaultName = "YOUR_VAULT_NAME",
  buffer = new Buffer(2.5 * 1024 * 1024), // 2.5MB buffer
  partSize = 1024 * 1024, // 1MB chunks,
  numPartsLeft = Math.ceil(buffer.length / partSize),
  startTime = new Date(),
  params = { vaultName: vaultName, partSize: partSize.toString() };

// Compute the complete SHA-256 tree hash so we can pass it
// to completeMultipartUpload request at the end
var treeHash = glacier.computeChecksums(buffer).treeHash;

// Initiate the multipart upload
console.log("Initiating upload to", vaultName);
// Call Glacier to initiate the upload.
glacier.initiateMultipartUpload(params, function (mpErr, multipart) {
  if (mpErr) {
    console.log("Error!", mpErr.stack);
    return;
  }
  console.log("Got upload ID", multipart.uploadId);

  // Grab each partSize chunk and upload it as a part
  for (var i = 0; i < buffer.length; i += partSize) {
    var end = Math.min(i + partSize, buffer.length),
      partParams = {
        vaultName: vaultName,
        uploadId: multipart.uploadId,
        range: "bytes " + i + "-" + (end - 1) + "/*",
        body: buffer.slice(i, end),
      };

    // Send a single part
    console.log("Uploading part", i, "=", partParams.range);
    glacier.uploadMultipartPart(partParams, function (multiErr, mData) {
      if (multiErr) return;
      console.log("Completed part", this.request.params.range);
      if (--numPartsLeft > 0) return; // complete only when all parts uploaded

      var doneParams = {
        vaultName: vaultName,
```

```
        uploadId: multipart.uploadId,
        archiveSize: buffer.length.toString(),
        checksum: treeHash, // the computed tree hash
      };

      console.log("Completing upload...");
      glacier.completeMultipartUpload(doneParams, function (err, data) {
        if (err) {
          console.log("An error occurred while uploading the archive");
          console.log(err);
        } else {
          var delta = (new Date() - startTime) / 1000;
          console.log("Completed upload in", delta, "seconds");
          console.log("Archive ID:", data.archiveId);
          console.log("Checksum:  ", data.checksum);
        }
      });
    });
  }
});
```

- Pour plus d'informations, consultez le Guide du développeur AWS SDK pour JavaScript.
- Pour plus de détails sur l'API, reportez-vous UploadMultipartPartà la section Référence des AWS SDK pour JavaScript API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Scénarios d'utilisation d'Amazon Glacier AWS SDKs

Les exemples de code suivants vous montrent comment implémenter des scénarios courants dans Amazon Glacier avec AWS SDKs. Ces scénarios vous montrent comment accomplir des tâches spécifiques en appelant plusieurs fonctions au sein d'Amazon Glacier ou en les combinant à d'autres Services AWS. Chaque exemple inclut un lien vers le code source complet, où vous trouverez des instructions sur la configuration et l'exécution du code.

Les scénarios ciblent un niveau d'expérience intermédiaire pour vous aider à comprendre les actions de service dans leur contexte.

Exemples

- Archivez un fichier sur Amazon Glacier, recevez des notifications et lancez une tâche à l'aide d'un AWS SDK
- Obtenez le contenu de l'archive Amazon Glacier et supprimez l'archive à l'aide d'un AWS SDK

Archivez un fichier sur Amazon Glacier, recevez des notifications et lancez une tâche à l'aide d'un AWS SDK

L'exemple de code suivant illustre comment :

- Créez un coffre-fort Amazon Glacier.
- Configurez le coffre pour qu'il publie les notifications dans une rubrique Amazon SNS.
- Chargez un fichier d'archive sur le coffre.
- Lancez une tâche d'extraction d'archive.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Créez une classe qui englobe les opérations Amazon Glacier.

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

logger = logging.getLogger(__name__)


class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""
```

```
    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    def create_vault(self, vault_name):
        """
        Creates a vault.

        :param vault_name: The name to give the vault.
        :return: The newly created vault.
        """
        try:
            vault = self.glacier_resource.create_vault(vaultName=vault_name)
            logger.info("Created vault %s.", vault_name)
        except ClientError:
            logger.exception("Couldn't create vault %s.", vault_name)
            raise
        else:
            return vault


    def list_vaults(self):
        """
        Lists vaults for the current account.
        """
        try:
            for vault in self.glacier_resource.vaults.all():
                logger.info("Got vault %s.", vault.name)
        except ClientError:
            logger.exception("Couldn't list vaults.")
            raise


    @staticmethod
    def upload_archive(vault, archive_description, archive_file):
        """
        Uploads an archive to a vault.

        :param vault: The vault where the archive is put.
        :param archive_description: A description of the archive.
```

```
        :param archive_file: The archive file to put in the vault.
        :return: The uploaded archive.
        """
        try:
            archive = vault.upload_archive(
                archiveDescription=archive_description, body=archive_file
            )
            logger.info(
                "Uploaded %s with ID %s to vault %s.",
                archive_description,
                archive.id,
                vault.name,
            )
        except ClientError:
            logger.exception(
                "Couldn't upload %s to %s.", archive_description, vault.name
            )
            raise
        else:
            return archive


    @staticmethod
    def initiate_archive_retrieval(archive):
        """
        Initiates an archive retrieval job. Standard retrievals typically
complete
        within 3–5 hours. When the job completes, you can get the archive
contents
        by calling get_output().

        :param archive: The archive to retrieve.
        :return: The archive retrieval job.
        """
        try:
            job = archive.initiate_archive_retrieval()
            logger.info("Started %s job with ID %s.", job.action, job.id)
        except ClientError:
            logger.exception("Couldn't start job on archive %s.", archive.id)
            raise
        else:
            return job
```

```
    @staticmethod
    def list_jobs(vault, job_type):
        """
        Lists jobs by type for the specified vault.

        :param vault: The vault to query.
        :param job_type: The type of job to list.
        :return: The list of jobs of the requested type.
        """
        job_list = []
        try:
            if job_type == "all":
                jobs = vault.jobs.all()
            elif job_type == "in_progress":
                jobs = vault.jobs_in_progress.all()
            elif job_type == "completed":
                jobs = vault.completed_jobs.all()
            elif job_type == "succeeded":
                jobs = vault.succeeded_jobs.all()
            elif job_type == "failed":
                jobs = vault.failed_jobs.all()
            else:
                jobs = []
                logger.warning("%s isn't a type of job I can get.", job_type)
            for job in jobs:
                job_list.append(job)
                logger.info("Got %s %s job %s.", job_type, job.action, job.id)
        except ClientError:
            logger.exception("Couldn't get %s jobs from %s.", job_type,
vault.name)
            raise
        else:
            return job_list


    def set_notifications(self, vault, sns_topic_arn):
        """
        Sets an Amazon Simple Notification Service (Amazon SNS) topic as a target
        for notifications. Amazon S3 Glacier publishes messages to this topic for
        the configured list of events.

        :param vault: The vault to set up to publish notifications.
        :param sns_topic_arn: The Amazon Resource Name (ARN) of the topic that
                              receives notifications.
```

```
        :return: Data about the new notification configuration.
        """
        try:
            notification = self.glacier_resource.Notification("-", vault.name)
            notification.set(
                vaultNotificationConfig={
                    "SNSTopic": sns_topic_arn,
                    "Events": [
                        "ArchiveRetrievalCompleted",
                        "InventoryRetrievalCompleted",
                    ],
                }
            )
            logger.info(
                "Notifications will be sent to %s for events %s from %s.",
                notification.sns_topic,
                notification.events,
                notification.vault_name,
            )
        except ClientError:
            logger.exception(
                "Couldn't set notifications to %s on %s.", sns_topic_arn,
 vault.name
            )
            raise
        else:
            return notification
```

Appelez des fonctions sur la classe encapsuleur pour créer un coffre et charger un fichier, puis configurez le coffre pour qu'il publie des notifications et lancez une tâche afin d'extraire l'archive.

```
def upload_demo(glacier, vault_name, topic_arn):
    """
    Shows how to:
    * Create a vault.
    * Configure the vault to publish notifications to an Amazon SNS topic.
    * Upload an archive.
    * Start a job to retrieve the archive.

    :param glacier: A Boto3 Amazon S3 Glacier resource.
```

```
    :param vault_name: The name of the vault to create.
    :param topic_arn: The ARN of an Amazon SNS topic that receives notification
of
                      Amazon S3 Glacier events.
    """
    print(f"\nCreating vault {vault_name}.")
    vault = glacier.create_vault(vault_name)
    print("\nList of vaults in your account:")
    glacier.list_vaults()
    print(f"\nUploading glacier_basics.py to {vault.name}.")
    with open("glacier_basics.py", "rb") as upload_file:
        archive = glacier.upload_archive(vault, "glacier_basics.py", upload_file)
    print(
        "\nStarting an archive retrieval request to get the file back from the "
        "vault."
    )
    glacier.initiate_archive_retrieval(archive)
    print("\nListing in progress jobs:")
    glacier.list_jobs(vault, "in_progress")
    print(
        "\nBecause Amazon S3 Glacier is intended for infrequent retrieval, an "
        "archive request with Standard retrieval typically completes within 3-5 "
        "hours."
    )
    if topic_arn:
        notification = glacier.set_notifications(vault, topic_arn)
        print(
            f"\nVault {vault.name} is configured to notify the "
            f"{notification.sns_topic} topic when {notification.events} "
            f"events occur. You can subscribe to this topic to receive "
            f"a message when the archive retrieval completes.\n"
        )
    else:
        print(
            f"\nVault {vault.name} is not configured to notify an Amazon SNS
topic "
            f"when the archive retrieval completes so wait a few hours."
        )
    print("\nRetrieve your job output by running this script with the --retrieve
flag.")
```

- Pour plus de détails sur l'API, consultez les rubriques suivantes dans la Référence des API du kit AWS SDK for Python (Boto3).
  - CreateVault
  - InitiateJob
  - ListJobs
  - ListVaults
  - SetVaultNotifications
  - UploadArchive

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Obtenez le contenu de l'archive Amazon Glacier et supprimez l'archive à l'aide d'un AWS SDK

L'exemple de code suivant illustre comment :

- Répertoriez les tâches pour un coffre-fort Amazon Glacier et obtenez le statut des tâches.
- Obtenir la sortie d'une tâche d'extraction d'archive terminée.
- Supprimer une archive.
- Supprimer un coffre.

Python

Kit SDK for Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Créez une classe qui englobe les opérations Amazon Glacier.

```
import argparse
```

```
import logging
import os
import boto3
from botocore.exceptions import ClientError

logger = logging.getLogger(__name__)


class GlacierWrapper:
    """Encapsulates Amazon S3 Glacier API operations."""

    def __init__(self, glacier_resource):
        """
        :param glacier_resource: A Boto3 Amazon S3 Glacier resource.
        """
        self.glacier_resource = glacier_resource


    @staticmethod
    def list_jobs(vault, job_type):
        """
        Lists jobs by type for the specified vault.

        :param vault: The vault to query.
        :param job_type: The type of job to list.
        :return: The list of jobs of the requested type.
        """
        job_list = []
        try:
            if job_type == "all":
                jobs = vault.jobs.all()
            elif job_type == "in_progress":
                jobs = vault.jobs_in_progress.all()
            elif job_type == "completed":
                jobs = vault.completed_jobs.all()
            elif job_type == "succeeded":
                jobs = vault.succeeded_jobs.all()
            elif job_type == "failed":
                jobs = vault.failed_jobs.all()
            else:
                jobs = []
                logger.warning("%s isn't a type of job I can get.", job_type)
            for job in jobs:
                job_list.append(job)
```

```
                logger.info("Got %s %s job %s.", job_type, job.action, job.id)
        except ClientError:
            logger.exception("Couldn't get %s jobs from %s.", job_type,
vault.name)
            raise
        else:
            return job_list


    @staticmethod
    def get_job_output(job):
        """
        Gets the output of a job, such as a vault inventory or the contents of an
        archive.

        :param job: The job to get output from.
        :return: The job output, in bytes.
        """
        try:
            response = job.get_output()
            out_bytes = response["body"].read()
            logger.info("Read %s bytes from job %s.", len(out_bytes), job.id)
            if "archiveDescription" in response:
                logger.info(
                    "These bytes are described as '%s'",
response["archiveDescription"]
                )
        except ClientError:
            logger.exception("Couldn't get output for job %s.", job.id)
            raise
        else:
            return out_bytes


    @staticmethod
    def delete_archive(archive):
        """
        Deletes an archive from a vault.

        :param archive: The archive to delete.
        """
        try:
            archive.delete()
            logger.info(
```

```
                "Deleted archive %s from vault %s.", archive.id,
 archive.vault_name
            )
        except ClientError:
            logger.exception("Couldn't delete archive %s.", archive.id)
            raise


    @staticmethod
    def delete_vault(vault):
        """
        Deletes a vault.

        :param vault: The vault to delete.
        """
        try:
            vault.delete()
            logger.info("Deleted vault %s.", vault.name)
        except ClientError:
            logger.exception("Couldn't delete vault %s.", vault.name)
            raise
```

Appelez des fonctions sur la classe encapsuleur pour obtenir le contenu de l'archive d'une tâche terminée, puis supprimez l'archive.

```
def retrieve_demo(glacier, vault_name):
    """
    Shows how to:
    * List jobs for a vault and get job status.
    * Get the output of a completed archive retrieval job.
    * Delete an archive.
    * Delete a vault.

    :param glacier: A Boto3 Amazon S3 Glacier resource.
    :param vault_name: The name of the vault to query for jobs.
    """
    vault = glacier.glacier_resource.Vault("-", vault_name)
    try:
        vault.load()
    except ClientError as err:
        if err.response["Error"]["Code"] == "ResourceNotFoundException":
```

```
            print(
                f"\nVault {vault_name} doesn't exist. You must first run this
script "
                f"with the --upload flag to create the vault."
            )
            return
        else:
            raise

    print(f"\nGetting completed jobs for {vault.name}.")
    jobs = glacier.list_jobs(vault, "completed")
    if not jobs:
        print("\nNo completed jobs found. Give it some time and try again
later.")
        return

    retrieval_job = None
    for job in jobs:
        if job.action == "ArchiveRetrieval" and job.status_code == "Succeeded":
            retrieval_job = job
            break
    if retrieval_job is None:
        print(
            "\nNo ArchiveRetrieval jobs found. Give it some time and try again "
            "later."
        )
        return

    print(f"\nGetting output from job {retrieval_job.id}.")
    archive_bytes = glacier.get_job_output(retrieval_job)
    archive_str = archive_bytes.decode("utf-8")
    print("\nGot archive data. Printing the first 10 lines.")
    print(os.linesep.join(archive_str.split(os.linesep)[:10]))

    print(f"\nDeleting the archive from {vault.name}.")
    archive = glacier.glacier_resource.Archive(
        "-", vault.name, retrieval_job.archive_id
    )
    glacier.delete_archive(archive)

    print(f"\nDeleting {vault.name}.")
    glacier.delete_vault(vault)
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans le Guide de référence des API du kit SDK AWS pour Python (Boto3).
  - DeleteArchive
  - DeleteVault
  - GetJobOutput
  - ListJobs

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation d'Amazon Glacier avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit SDK.

Sécurité dans Amazon Glacier

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cette notion par les termes sécurité du cloud et sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS Cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. L'efficacité de notre sécurité est régulièrement testée et vérifiée par des auditeurs tiers dans le cadre des programmes de conformitéAWS. Pour en savoir plus sur les programmes de conformité qui s'appliquent à Amazon Glacier (Amazon Glacier), consultez la section AWS Services concernés par programme de conformité.
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris la sensibilité de vos données, les exigences de votre organisation, et la législation et la réglementation applicables.

Cette documentation vous aidera à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation d'Amazon Glacier. Les rubriques suivantes expliquent comment configurer Amazon Glacier pour atteindre vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui peuvent vous aider à surveiller et à sécuriser vos ressources Amazon Glacier.

Rubriques

- Protection des données dans Amazon Glacier
- Identity and Access Management pour Amazon Glacier
- Journalisation et surveillance dans Amazon Glacier
- Validation de conformité pour Amazon Glacier
- Résilience dans Amazon Glacier
- Sécurité de l'infrastructure dans Amazon Glacier

Protection des données dans Amazon Glacier

Amazon Glacier (Amazon Glacier) fournit un stockage cloud hautement durable pour l'archivage des données et la sauvegarde à long terme. Amazon Glacier est conçu pour offrir une durabilité de 99,999999999 % et fournit des fonctionnalités complètes de sécurité et de conformité qui peuvent vous aider à répondre aux exigences réglementaires strictes. Amazon Glacier stocke les données de manière redondante dans plusieurs zones de AWS disponibilité (AZ) et sur plusieurs appareils au sein de chaque zone. Pour augmenter la durabilité, Amazon Glacier stocke vos données de manière synchrone sur plusieurs sites AZs avant de confirmer la réussite du chargement.

Pour plus d'informations sur l'infrastructure cloud AWS mondiale, consultez la section Infrastructure mondiale.

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de n'accorder aux utilisateurs, groupes ou rôles individuels que les autorisations nécessaires à l'accomplissement de leurs tâches.

Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-2 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d’informations sur les points de terminaison FIPS (Federal Information Processing Standard) disponibles, consultez Federal Information Processing Standard (FIPS) 140-2 (Normes de traitement de l’information fédérale).

Rubriques

- Chiffrement des données
- Gestion des clés
- Confidentialité du trafic inter-réseaux

Chiffrement des données

La protection des données fait référence à la protection des données en transit (lorsqu'elles sont acheminées vers et depuis Amazon Glacier) et au repos (lorsqu'elles sont stockées dans AWS des centres de données). Vous pouvez protéger les données en transit qui sont chargées directement sur Amazon Glacier à l'aide du protocole SSL (Secure Sockets Layer) ou du chiffrement côté client.

Vous pouvez également accéder à Amazon Glacier via Amazon S3. Amazon S3 prend en charge la configuration du cycle de vie sur un compartiment Amazon S3, ce qui vous permet de transférer des objets vers la classe de stockage Amazon Glacier à des fins d'archivage. Les données en transit

entre Amazon S3 et Amazon Glacier via des politiques de cycle de vie sont cryptées à l'aide du protocole SSL.

Les données au repos stockées dans Amazon Glacier sont automatiquement chiffrées côté serveur à l'aide de la norme de chiffrement avancée 256 bits (AES-256) avec des clés gérées par. AWS Si vous préférez gérer vos propres clés, vous pouvez également utiliser le chiffrement côté client avant de stocker des données dans Amazon Glacier. Pour plus d'informations sur la configuration du chiffrement par défaut pour Amazon S3, consultez Chiffrement par défaut Amazon S3 dans le Guide de l'utilisateur Amazon Simple Storage Service.

Gestion des clés

Le chiffrement côté serveur permet de chiffrer les données au repos, c'est-à-dire qu'Amazon Glacier chiffre vos données au fur et à mesure qu'il les écrit dans ses centres de données et les déchiffre pour vous lorsque vous y accédez. Tant que vous authentifiez votre demande et que vous avez des autorisations d’accès, il n’y a aucune différence dans la manière dont vous accédez aux données chiffrées ou déchiffrées.

Les données au repos stockées dans Amazon Glacier sont automatiquement chiffrées côté serveur à l'aide du protocole AES-256, à l'aide de clés gérées par. AWS Comme mesure de protection supplémentaire, AWS chiffre la clé elle-même à l'aide d'une clé racine que nous renouvelons régulièrement.

Confidentialité du trafic inter-réseaux

L'accès à Amazon Glacier via le réseau se fait par AWS publication APIs. Les clients doivent prendre en charge le protocole TLS (Transport Layer Security) 1.2. Nous recommandons TLS 1.3 ou version ultérieure. Les clients doivent également prendre en charge les suites de chiffrement PFS (Perfect Forward Secrecy) comme Ephemeral Diffie-Hellman (DHE) ou Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes. De plus, vous devez signer les demandes à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète, toutes deux associées à un principal IAM. Vous pouvez également utiliser le AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires qui serviront à signer les demandes.

Points de terminaison d’un VPC

Un point de terminaison de cloud privé virtuel (VPC) vous permet de connecter de manière privée votre VPC aux services pris en charge AWS et aux services de point de terminaison VPC alimentés

par le biais d'une passerelle Internet, d' AWS PrivateLink un périphérique NAT, d'une connexion VPN ou d'une connexion. AWS Direct Connect Bien qu'Amazon Glacier ne prenne pas directement en charge les points de terminaison VPC, vous pouvez tirer parti des points de terminaison VPC Amazon Simple Storage Service (Amazon S3) si vous accédez à Amazon Glacier en tant que niveau de stockage intégré à Amazon S3.

Pour plus d'informations sur la configuration du cycle de vie d'Amazon S3 et la transition des objets vers la classe de stockage Amazon Glacier, consultez la section Object Lifecycle Management and Transitioning Objects dans le guide de l'utilisateur d'Amazon Simple Storage Service. Pour plus d'informations sur les points de terminaison d'un VPC, consultez Points de terminaison d'un VPC dans le Guide de l'utilisateur Amazon VPC.

Identity and Access Management pour Amazon Glacier

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 les ressources Amazon Glacier. 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 Amazon Glacier fonctionne avec IAM
- Exemples de politiques basées sur l'identité pour Amazon Glacier
- Exemples de politiques basées sur les ressources pour Amazon Glacier
- Résolution des problèmes d'identité et d'accès à Amazon Glacier
- Référence des autorisations d'API

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 d'identité et d'accès à Amazon Glacier)
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir Comment Amazon Glacier fonctionne avec IAM)
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir Exemples de politiques basées sur l'identité pour Amazon Glacier)

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS 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 votre annuaire d'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 d'utilisateur à un rôle IAM (console) ou en appelant une opération d' AWS API AWS CLI ou d'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 Amazon Glacier fonctionne avec IAM

Avant d'utiliser IAM pour gérer l'accès à Amazon Glacier, découvrez quelles fonctionnalités IAM peuvent être utilisées avec Amazon Glacier.

Fonctionnalités IAM que vous pouvez utiliser avec Amazon Glacier

Fonctionnalité IAM	Assistance Amazon Glacier
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Oui
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique (spécifiques au service)	Oui
ACLs	Non

Fonctionnalité IAM	Assistance Amazon Glacier
ABAC (étiquettes dans les politiques)	Non
Informations d'identification temporaires	Oui
Autorisations de principals	Non
Rôles du service	Non
Rôles liés à un service	Non

Pour obtenir une vue d'ensemble de la façon dont Amazon Glacier et les autres AWS services fonctionnent avec la plupart des fonctionnalités IAM, consultez les AWS services compatibles avec IAM dans le guide de l'utilisateur IAM.

Politiques basées sur l'identité pour Amazon Glacier

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 Amazon Glacier

Pour consulter des exemples de politiques basées sur l'identité d'Amazon Glacier, consultez. Exemples de politiques basées sur l'identité pour Amazon Glacier

Politiques basées sur les ressources au sein d'Amazon Glacier

Prend en charge les politiques basées sur les ressources : oui

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.

Le service Amazon Glacier ne prend en charge qu'un seul type de politique basée sur les ressources, appelée politique de coffre-fort, qui est attachée à un coffre-fort. Cette politique désigne les principaux autorisés à effectuer des actions sur le coffre.

Les politiques d'Amazon Glacier Vault gèrent les autorisations de la manière suivante :

- Les autorisations utilisateur d'un compte sont gérées avec une seule politique de coffre, non pas avec plusieurs politiques utilisateur distinctes.
- Les autorisations entre comptes sont gérées en tant que solution de remplacement à l'utilisation de rôles IAM.

Exemples de politiques basées sur les ressources dans Amazon Glacier

Pour consulter des exemples de politiques basées sur les ressources d'Amazon Glacier, consultez. Exemples de politiques basées sur les ressources pour Amazon Glacier

Actions politiques pour Amazon Glacier

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 actions Amazon Glacier, consultez la section Actions définies par Amazon Glacier dans le Service Authorization Reference.

Les actions politiques dans Amazon Glacier utilisent le préfixe suivant avant l'action :

```
glacier
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

```
"Action": [
            "glacier:CreateVault",
            "glacier:DescribeVault",
            "glacier:ListVaults"
         ]
```

Vous pouvez aussi spécifier plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante :

```
"Action": "glacier:GetVault*"
```

Pour consulter des exemples de politiques basées sur l'identité d'Amazon Glacier, consultez. Exemples de politiques basées sur l'identité pour Amazon Glacier

Ressources relatives aux politiques pour Amazon Glacier

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 ressources Amazon Glacier et de leurs caractéristiques ARNs, consultez la section Ressources définies par Amazon Glacier dans le Service Authorization Reference. Pour savoir quelles actions vous pouvez spécifier l'ARN de chaque ressource, consultez la section Actions définies par Amazon Glacier.

Dans Amazon Glacier, la ressource principale est un coffre-fort. Amazon Glacier prend en charge les politiques uniquement au niveau du coffre. En d'autres termes, dans une politique IAM, la `Resource` valeur que vous spécifiez peut être un coffre-fort spécifique ou un ensemble de coffres-forts dans une région spécifique AWS . Amazon Glacier ne prend pas en charge les autorisations au niveau de l'archive.

Pour toutes les actions Amazon Glacier, `Resource` indique le coffre-fort sur lequel vous souhaitez accorder les autorisations. Ces ressources sont associées à des noms de ressources Amazon (ARNs) uniques, comme indiqué dans le tableau suivant, et vous pouvez utiliser un caractère générique (*) dans l'ARN pour faire correspondre les noms de coffres-forts commençant par le même préfixe.

Amazon Glacier fournit un ensemble d'opérations permettant d'utiliser les ressources Amazon Glacier. Pour plus d'informations sur les opérations disponibles, consultez la page Référence d'API pour Amazon Glacier.

Certaines actions de l'API Amazon Glacier prennent en charge plusieurs ressources. Par exemple, sachant que `glacier:AddTagsToVault` accède à examplevault1 et à examplevault2, un principal doit disposer d'autorisations d'accès à ces deux ressources. Pour spécifier plusieurs ressources dans une seule instruction, séparez-les ARNs par des virgules.

```
"Resource": [
""arn:aws:glacier:us-west-2:123456789012:vaults/examplevault1",",
""arn:aws:glacier:us-west-2:123456789012:vaults/examplevault2","
]
```

Clés de conditions de politique pour Amazon Glacier

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 condition Amazon Glacier, consultez la section Clés de condition pour Amazon Glacier dans la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez Actions définies par Amazon Glacier.

Pour voir des exemples d'utilisation des clés de condition propres à Glacier, consultez Stratégies de verrouillage de coffre.

ACLs dans Amazon Glacier

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 Amazon Glacier

Prise en charge d'ABAC (balises dans les politiques) : non

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 Amazon Glacier

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 Amazon Glacier

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 Amazon Glacier

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 les fonctionnalités d'Amazon Glacier. Modifiez les rôles de service uniquement lorsque Amazon Glacier fournit des instructions à cet effet.

Rôles liés à un service pour Amazon Glacier

Prend en charge les rôles liés à un service : non

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 Amazon Glacier

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier les ressources Amazon Glacier. 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 Amazon Glacier, y compris le ARNs format de chaque type de ressource, consultez la section Actions, ressources et clés de condition pour Amazon Glacier dans la référence d'autorisation de service.

Voici un exemple de politique qui accorde des autorisations pour trois actions liées au coffre-fort Amazon Glacier (`glacier:CreateVault`, `glacier:DescribeVault` et`glacier:ListVaults`) sur une ressource, en utilisant le nom de ressource Amazon (ARN) qui identifie tous les coffres-

forts de la région. `us-west-2` AWS ARNs identifier les AWS ressources de manière unique. Pour plus d'informations sur l' ARNs utilisation avec Amazon Glacier, consultezRessources relatives aux politiques pour Amazon Glacier.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
         "glacier:CreateVault",
         "glacier:DescribeVault",
         "glacier:ListVaults"
         ],
         "Resource": "arn:aws:glacier:us-west-2:123456789012:vaults/*"
      }
   ]
}
```

La stratégie accorde des autorisations permettant de créer, de répertorier et d'obtenir des descriptions des coffres dans la région `us-west-2`. Le caractère générique (*) à la fin de l'ARN signifie que cette instruction peut correspondre à tous les noms de coffre.

> **Important**
>
> Lorsque vous accordez des autorisations de création d'un coffre à l'aide de l'opération `glacier:CreateVault`, vous devez spécifier un caractère générique (*) dans la mesure où vous ignorez le nom du coffre tant que vous ne l'avez pas créé.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console Amazon Glacier
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations
- Exemples de politiques gérées par le client

Bonnes pratiques en matière de politiques

Les politiques basées sur l'identité déterminent si quelqu'un peut créer, accéder ou supprimer les ressources Amazon Glacier de 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 console Amazon Glacier

Pour accéder à la console Amazon Glacier, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et de consulter les informations relatives aux ressources Amazon Glacier présentes dans 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.

La console Amazon Glacier fournit un environnement intégré qui vous permet de créer et de gérer les coffres-forts Amazon Glacier. Les identités IAM que vous créez doivent au minimum être autorisées à accéder à la console Amazon Glacier pour que l'`glacier:ListVaults`action puisse consulter la console Amazon Glacier, comme indiqué dans l'exemple suivant.

JSON

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

```
        }
```

AWS répond à de nombreux cas d'utilisation courants en fournissant des politiques IAM autonomes créées et administrées par. AWS Les politiques gérées octroient les autorisations requises dans les cas d'utilisation courants et vous évitent d'avoir à réfléchir aux autorisations qui sont requises. Pour plus d'informations, consultez Politiques gérées par AWS dans le Guide de l'utilisateur IAM.

Les politiques AWS gérées suivantes, que vous pouvez associer aux utilisateurs de votre compte, sont spécifiques à Amazon Glacier :

- AmazonGlacierReadOnlyAccess— Accorde un accès en lecture seule à Amazon Glacier via le AWS Management Console.
- AmazonGlacierFullAccess— Accorde un accès complet à Amazon Glacier via le AWS Management Console.

Vous pouvez également créer vos propres politiques IAM personnalisées pour autoriser les actions et les ressources de l'API Amazon Glacier. Vous pouvez associer ces politiques personnalisées aux rôles IAM personnalisés que vous créez pour vos coffres-forts Amazon Glacier.

Les deux politiques AWS gérées par Amazon Glacier décrites dans la section suivante accordent des autorisations pour`glacier:ListVaults`.

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": "*"
        }
    ]
}
```

Exemples de politiques gérées par le client

Dans cette section, vous trouverez des exemples de politiques utilisateur qui accordent des autorisations pour diverses actions Amazon Glacier. Ces politiques fonctionnent lorsque vous utilisez l'API REST Amazon Glacier, Amazon SDKs, ou AWS CLI, le cas échéant, la console de gestion Amazon Glacier.

> **Note**
>
> Tous les exemples utilisent la région de l'Ouest des États-Unis (Oregon) (`us-west-2`) et contiennent un récit fictif. IDs

Exemples

- Exemple 1 : Permettre à un utilisateur de télécharger les archives d'un coffre

- Exemple 2 : Permettre à un utilisateur de créer un coffre et de configurer des notifications
- Exemple 3 : Permettre à un utilisateur de charger des Archives dans un coffre spécifique
- Exemple 4 : Octroyer des autorisations complètes à un utilisateur pour un coffre spécifique

Exemple 1 : Permettre à un utilisateur de télécharger les archives d'un coffre

Pour télécharger une archive, vous devez d'abord lancer une tâche d'extraction de l'archive. Une fois la tâche d'extraction terminée, vous pouvez télécharger les données. L'exemple de stratégie suivant octroie des autorisations pour l'action `glacier:InitiateJob` de lancement d'une tâche (ce qui permet à l'utilisateur d'extraire du coffre une archive ou un inventaire de coffre) et des autorisations pour l'action `glacier:GetJobOutput` pour télécharger les données extraites. La stratégie accorde également des autorisations permettant d'effectuer l'action `glacier:DescribeJob` afin que l'utilisateur puisse obtenir l'état de la tâche. Pour de plus amples informations, veuillez consulter Lancement d'une tâche (POST jobs).

La stratégie accorde ces autorisations sur un coffre nommé `examplevault`. Vous pouvez obtenir l'ARN du coffre-fort depuis la console Amazon Glacier ou par programmation en appelant les actions Description de coffre (GET vault) ou les actions de l'Affichage des coffres (GET vaults)API.

Exemple 2 : Permettre à un utilisateur de créer un coffre et de configurer des notifications

L'exemple de politique suivant octroie les autorisations nécessaires pour créer un coffre dans la région us-west-2, comme indiqué dans l'élément `Resource`, et pour configurer des notifications. Pour plus d'informations sur l'utilisation des notifications, consultez la page Configuration des notifications de coffre-fort dans Amazon Glacier. La politique accorde également l'autorisation de répertorier les coffres-forts de la AWS région et d'obtenir une description spécifique du coffre-fort.

> **Important**
>
> Lorsque vous accordez des autorisations de création d'un coffre à l'aide de l'opération `glacier:CreateVault`, vous devez spécifier un caractère générique (*) `Resource` dans la mesure où vous ignorez le nom du coffre tant que vous ne l'avez pas créé.

Exemple 3 : Permettre à un utilisateur de charger des Archives dans un coffre spécifique

L'exemple de politique suivant octroie les autorisations nécessaires pour charger les archives sur un coffre spécifique de la région us-west-2. Ces autorisations permettent à un utilisateur de télécharger

une archive en une seule fois à l'aide de l'opération d'API Chargement d'archive (POST archive) ou en plusieurs parties à l'aide de l'opération d'API Lancement de chargement partitionné (POST multipart-uploads).

Exemple 4 : Octroyer des autorisations complètes à un utilisateur pour un coffre spécifique

L'exemple de politique suivant accorde des autorisations pour toutes les actions Amazon Glacier sur un coffre-fort nommé`examplevault`.

Exemples de politiques basées sur les ressources pour Amazon Glacier

Un coffre-fort Amazon Glacier peut être associé à une politique d'accès au coffre-fort et à une politique de verrouillage du coffre-fort. Une politique d'accès au coffre-fort Amazon Glacier est une politique basée sur les ressources que vous pouvez utiliser pour gérer les autorisations d'accès à votre coffre-fort. Une stratégie de verrouillage de coffre est une stratégie d'accès au coffre qui peut être verrouillée. Une fois que vous avez verrouillé une stratégie de verrouillage de coffre, celle-ci ne peut plus être modifiée. Vous pouvez utiliser une stratégie de verrouillage de coffre afin d'appliquer les contrôles de conformité.

Rubriques

- Politiques d'accès au coffre-fort
- Stratégies de verrouillage de coffre

Politiques d'accès au coffre-fort

Une politique d'accès au coffre-fort Amazon Glacier est une politique basée sur les ressources que vous pouvez utiliser pour gérer les autorisations d'accès à votre coffre-fort.

Vous pouvez créer une stratégie d'accès aux coffres pour chaque coffre afin de gérer les autorisations. Vous pouvez à tout moment modifier les autorisations contenues dans d'une stratégie d'accès au coffre. Amazon Glacier applique également à chaque coffre-fort une politique de verrouillage qui ne peut pas être modifiée une fois que vous l'avez verrouillé. Pour plus d'informations sur l'utilisation des stratégies de verrouillage de coffre, consultez la page Stratégies de verrouillage de coffre.

Exemples

- Exemple 1 : accorder des autorisations entre comptes pour des actions Amazon Glacier spécifiques
- Exemple 2 : Accorder des autorisations entre comptes pour les opérations de suppression MFA

Exemple 1 : accorder des autorisations entre comptes pour des actions Amazon Glacier spécifiques

L'exemple de politique suivant accorde des autorisations entre comptes à deux Comptes AWS pour un ensemble d'opérations Amazon Glacier sur un coffre-fort nommé`examplevault`.

> **Note**
>
> Le compte qui possède le coffre est facturé pour tous les coûts liés à ce dernier. Toutes les demandes, tous les transferts de données et les coûts d'extraction effectués par des comptes externes autorisés sont facturés au compte qui possède le coffre.

Exemple 2 : Accorder des autorisations entre comptes pour les opérations de suppression MFA

Vous pouvez utiliser l'authentification multifactorielle (MFA) pour protéger vos ressources Amazon Glacier. Pour fournir un niveau supplémentaire de sécurité, MFA requiert des utilisateurs qu'ils prouvent la possession matérielle d'un appareil MFA en fournissant un code MFA valide. Pour plus d'informations sur la configuration de l'accès MFA, consultez Configuration de l'accès aux API protégé par MFA dans le Guide de l'utilisateur IAM.

L'exemple de politique accorde à une personne Compte AWS possédant des informations d'identification temporaires l'autorisation de supprimer des archives d'un coffre-fort nommé examplevault, à condition que la demande soit authentifiée par un périphérique MFA. La stratégie utilise la clé de condition `aws:MultiFactorAuthPresent` pour spécifier cette exigence supplémentaire. Pour plus d'informations, consultez Clés disponibles pour les conditions dans le Guide de l'utilisateur IAM.

Stratégies de verrouillage de coffre

Un coffre-fort Amazon Glacier (Amazon Glacier) peut être associé à une politique d'accès au coffre-fort basée sur les ressources et à une politique de verrouillage du coffre-fort. Une stratégie de verrouillage de coffre est une stratégie d'accès au coffre que vous pouvez verrouiller. L'utilisation d'une politique de verrouillage de coffre peut vous aider à faire respecter les exigences

réglementaires et de conformité. Amazon Glacier fournit un ensemble d'opérations d'API vous permettant de gérer les politiques Vault Lock, voirVerrouiller un coffre-fort à l'aide de l'API Amazon Glacier.

Pour illustrer le concept de stratégie de verrouillage de coffre, supposons que vous soyez tenu de conserver les archives pendant un an avant de pouvoir les supprimer. Pour mettre en place de cette exigence, vous pouvez créer une stratégie de verrouillage de coffre qui interdit aux utilisateurs de supprimer une archive avant un an. Vous pouvez tester cette stratégie avant de la verrouiller. En effet, une fois la stratégie verrouillée, elle ne peut plus être modifiée. Pour plus d'informations sur le processus de verrouillage, consultez la page Stratégies de verrouillage de coffre. Pour gérer d'autres autorisations utilisateur qui peuvent être modifiées, vous pouvez utiliser la stratégie d'accès au coffre (voir la page Politiques d'accès au coffre-fort).

Vous pouvez utiliser l'API Amazon Glacier SDKs AWS CLI, Amazon ou la console Amazon Glacier pour créer et gérer les politiques Vault Lock. Pour obtenir la liste des actions Amazon Glacier autorisées dans le cadre des politiques basées sur les ressources du coffre-fort, consultez. Référence des autorisations d'API

Exemples

- Exemple 1 : Refuser des autorisations de suppression pour les archives datant d'il y a moins de 365 jours
- Exemple 2 : Refuser des autorisations de suppression reposant sur une balise

Exemple 1 : Refuser des autorisations de suppression pour les archives datant d'il y a moins de 365 jours

Supposons qu'une exigence réglementaire vous oblige à conserver les archives pendant 1 an avant de pouvoir les supprimer. Vous pouvez appliquer cette exigence en mettant en œuvre la stratégie suivante de verrouillage de coffre. Cette stratégie refuse l'action `glacier:DeleteArchive` sur le coffre examplevault si l'archive que vous tentez de supprimer a moins d'un an. La politique utilise la clé de condition spécifique à Amazon Glacier `ArchiveAgeInDays` pour appliquer l'exigence de conservation d'un an.

Exemple 2 : Refuser des autorisations de suppression reposant sur une balise

Supposons que vous ayez une règle de conservation basée sur le temps et selon laquelle une archive peut être supprimée si elle a moins d'un an. Parallèlement, supposons que vous ayez besoin d'associer une suspension légale sur vos archives afin d'éviter toute suppression ou modification

pendant une durée indéfinie lors d'une enquête juridique. Dans ce cas, la suspension légale est prioritaire pendant la durée de la règle de conservation basée sur le temps spécifiée dans la stratégie de verrouillage de coffre.

Pour mettre ces deux règles en place, l'exemple de stratégie suivant comporte deux instructions :

- La première instruction refuse les autorisations de suppression pour tout le monde et verrouille le coffre. Ce verrouillage est effectué à l'aide de la balise `LegalHold`.
- La deuxième instruction octroie des autorisations de suppression lorsque l'archive a moins de 365 jours. Mais même si les archives ont moins de 365 jours, personne ne peut les supprimer lorsque la condition de la première instruction est remplie.

Résolution des problèmes d'identité et d'accès à Amazon Glacier

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec Amazon Glacier et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans Amazon Glacier
- Je ne suis pas autorisé à effectuer iam : PassRole
- Je souhaite autoriser des personnes extérieures à moi Compte AWS à accéder à mes ressources Amazon Glacier

Je ne suis pas autorisé à effectuer une action dans Amazon Glacier

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 `glacier:`*`GetWidget`* fictives.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 glacier: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 `glacier:`*`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 un message d'erreur indiquant que vous n'êtes pas autorisé à effectuer l'`iam:PassRole`action, vos politiques doivent être mises à jour pour vous permettre de transmettre un rôle à Amazon Glacier.

Certains vous 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 effectuer une action dans Amazon Glacier. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle 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 autoriser des personnes extérieures à moi Compte AWS à accéder à mes ressources Amazon Glacier

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 Amazon Glacier prend en charge ces fonctionnalités, consultezComment Amazon Glacier 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.

Référence des autorisations d'API

Lorsque vous configurez Comment Amazon Glacier fonctionne avec IAM et écrivez une politique d'autorisations que vous pouvez attacher à une identité IAM (politiques basées sur l'identité) ou à une ressource (politiques basées sur les ressources), vous pouvez utiliser le tableau suivant comme référence. Le chaque opération de l'API Amazon Glacier, les actions correspondantes pour lesquelles vous pouvez accorder des autorisations pour effectuer l'action et la AWS ressource pour laquelle vous pouvez accorder les autorisations.

Vous spécifiez les actions dans l'élément `Action` de la stratégie ainsi que la valeur des ressources dans l'élément `Resource` de la stratégie. De plus, vous pouvez utiliser l'élément `Condition` du langage de politique IAM pour indiquer quand une politique doit prendre effet.

Pour indiquer une action, utilisez le préfixe `glacier:` suivi du nom de l'opération d'API (par exemple, `glacier:CreateVault`). Pour la plupart des actions Amazon Glacier, `Resource` c'est le coffre-fort sur lequel vous souhaitez accorder les autorisations. Vous spécifiez un coffre en tant que valeur `Resource` en utilisant l'ARN du coffre. Pour exprimer des conditions, vous utilisez des clés de condition prédéfinies. Pour de plus amples informations, veuillez consulter Politiques basées sur les ressources au sein d'Amazon Glacier.

Le tableau suivant répertorie les actions qui peuvent être utilisées avec les stratégies basées sur une identité et celles qui sont basées sur une ressource.

> **Note**
>
> Certaines actions peuvent uniquement être utilisées avec les stratégies basées sur une identité. Ces actions sont marquées d'un astérisque (*) après le nom de l'opération d'API dans la première colonne.

API Amazon Glacier et autorisations requises pour les actions

Annulation de chargement partitionné (DELETE uploadID)

Autorisations nécessaires (actions d'API) : `glacier:AbortMultipartUpload`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier :

Interruption du verrouillage du coffre (DELETE lock-policy)

Autorisations nécessaires (actions d'API) : `glacier:AbortVaultLock`

Ressources:

Clés de condition d'Amazon Glacier :

Ajout de balises à un coffre (POST tags add)

Autorisations requises (actions d'API) : `glacier:AddTagsToVault`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Achèvement du chargement partitionné (POST uploadID)

Autorisations requises (actions d'API) : `glacier:CompleteMultipartUpload`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Achèvement du verrouillage du coffre (POST lockId)

Autorisations requises (actions d'API) : `glacier:CompleteVaultLock`

Ressources:

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Création de coffre (PUT vault) *

Autorisations requises (actions d'API) : `glacier:CreateVault`

Ressources:

Clés de condition d'Amazon Glacier :

Suppression d'une archive (DELETE archive)

Autorisations requises (actions d'API) : `glacier:DeleteArchive`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition Amazon Glacier :`glacier:ArchiveAgeInDays`, `glacier:ResourceTag/`*`TagKey`*

Suppression de coffre (DELETE vault)

Autorisations requises (actions d'API) : `glacier:DeleteVault`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Suppression de la stratégie d'accès au coffre (DELETE access-policy)

Autorisations requises (actions d'API) : `glacier:DeleteVaultAccessPolicy`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Supprimer les notifications de coffre (DELETE notification-configuration)

Autorisations requises (actions d'API) : `glacier:DeleteVaultNotifications`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Description de la tâche (GET JobID)

Autorisations requises (actions d'API) : `glacier:DescribeJob`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier :

Description de coffre (GET vault)

Autorisations requises (actions d'API) : `glacier:DescribeVault`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier :

Obtention de la stratégie d'extraction de données (GET policy) *

Autorisations requises (actions d'API) : `glacier:GetDataRetrievalPolicy`

Ressources: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:policies/retrieval-limit-policy`

Clés de condition d'Amazon Glacier :

Génération de sortie de tâche (GET output)

Autorisations requises (actions d'API) : `glacier:GetJobOutput`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Obtention de la stratégie d'accès au coffre (GET access-policy)

Autorisations requises (actions d'API) : `glacier:GetVaultAccessPolicy`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Obtenir le verrouillage de coffre (GET lock-policy)

Autorisations requises (actions d'API) : `glacier:GetVaultLock`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Obtenir les notifications de coffre (GET notification-configuration)

Autorisations requises (actions d'API) : `glacier:GetVaultNotifications`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Lancement d'une tâche (POST jobs)

Autorisations requises (actions d'API) : `glacier:InitiateJob`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition Amazon Glacier :`glacier:ArchiveAgeInDays`, `glacier:ResourceTag/`*`TagKey`*

Lancement de chargement partitionné (POST multipart-uploads)

Autorisations requises (actions d'API) : `glacier:InitiateMultipartUpload`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Lancement de verrouillage de coffre (POST lock-policy)

Autorisations requises (actions d'API) : `glacier:InitiateVaultLock`

Ressources:

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/`*`TagKey`*

Affichage des tâches (GET jobs)

Autorisations requises (actions d'API) : `glacier:ListJobs`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier :

Afficher la liste des chargements partitionnés (GET multipart-uploads)

Autorisations requises (actions d'API) : `glacier:ListMultipartUploads`

Ressources : `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Clés de condition d'Amazon Glacier :

Répertorier les parties (GET uploadID)

Autorisations requises (actions d'API) : `glacier:ListParts`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Affichage des balises d'un coffre (GET tags)

Autorisations requises (actions d'API) : `glacier:ListTagsForVault`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier :

Affichage des coffres (GET vaults)

Autorisations requises (actions d'API) : `glacier:ListVaults`

Ressources:

Clés de condition d'Amazon Glacier :

Suppression des balises d'un coffre (POST tags remove)

Autorisations requises (actions d'API) : `glacier:RemoveTagsFromVault`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/TagKey`

Définition de stratégie d'extraction de données (PUT policy) *

Autorisations requises (actions d'API) : `glacier:SetDataRetrievalPolicy`

Ressources:`arn:aws:glacier:region:account-id:policies/retrieval-limit-policy`

Clés de condition d'Amazon Glacier :

Définir la stratégie d'accès au coffre (PUT access-policy)

Autorisations requises (actions d'API) : `glacier:SetVaultAccessPolicy`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/TagKey`

Définition d'une configuration de notification de coffre (PUT notification-configuration)

Autorisations requises (actions d'API) : `glacier:SetVaultNotifications`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/TagKey`

Chargement d'archive (POST archive)

Autorisations requises (actions d'API) : `glacier:UploadArchive`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/TagKey`

Partie chargement (PUT uploadID)

Autorisations requises (actions d'API) : `glacier:UploadMultipartPart`

Ressources : `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Clés de condition d'Amazon Glacier : `glacier:ResourceTag/TagKey`

Journalisation et surveillance dans Amazon Glacier

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances d'Amazon Glacier (Amazon Glacier) et de vos AWS solutions. Vous devez collecter des données de surveillance provenant de toutes les parties de votre AWS solution afin de pouvoir identifier et corriger plus facilement la source d'une panne, le cas échéant. AWS fournit les outils suivants pour surveiller vos ressources Amazon Glacier et répondre aux incidents potentiels :

CloudWatch Alarmes Amazon

Lorsque vous utilisez Amazon Glacier via Amazon S3, vous pouvez utiliser les CloudWatch alarmes Amazon pour surveiller une seule métrique sur une période que vous spécifiez. Si la métrique dépasse un seuil donné, une notification est envoyée à une rubrique ou AWS Auto Scaling à une politique Amazon SNS. CloudWatch les alarmes n'appellent pas d'actions car elles se trouvent dans un état particulier. L'état doit avoir changé et avoir été conservé pendant un nombre de périodes spécifié. Pour plus d'informations, consultez la section Surveillance des métriques avec Amazon CloudWatch.

AWS CloudTrail Journaux

CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans Amazon Glacier. CloudTrail capture tous les appels d'API pour Amazon Glacier sous forme d'événements, y compris les appels depuis la console Amazon Glacier et les appels de code vers Amazon Glacier APIs. Pour de plus amples informations, veuillez consulter Journalisation des appels d'API Amazon Glacier avec AWS CloudTrail.

AWS Trusted Advisor

Trusted Advisor s'appuie sur les meilleures pratiques apprises en servant des centaines de milliers de AWS clients. Trusted Advisor inspecte votre AWS environnement, puis émet des recommandations lorsque des opportunités se présentent pour économiser de l'argent, améliorer la disponibilité et les performances du système ou contribuer à combler les failles de sécurité. Tous les AWS clients ont accès à cinq Trusted Advisor chèques. Les clients disposant d'un plan de support Business ou Enterprise peuvent consulter tous les Trusted Advisor chèques.

Pour plus d'informations, consultez AWS Trusted Advisor dans le Guide de l'utilisateur Support .

Validation de conformité pour Amazon Glacier

La sécurité et la conformité d'Amazon Glacier (Amazon Glacier) sont évaluées par des auditeurs tiers dans le cadre de plusieurs programmes de AWS conformité, notamment les suivants :

- System and Organization Controls (SOC)
- Norme de sécurité des données de l'industrie des cartes de paiement (PCI DSS)
- Federal Risk and Authorization Management Program (FedRAMP)
- Health Insurance Portability and Accountability Act (HIPAA)

AWS fournit une liste fréquemment mise à jour des AWS services concernés par des programmes de conformité spécifiques sur la page AWS Services in Scope by Compliance Program.

Les rapports d'audit tiers peuvent être téléchargés à l'aide de AWS Artifact. Pour plus d'informations, consultez la section Téléchargement de rapports dans AWS Artifact dans le guide de l'AWS Artifact utilisateur.

Pour plus d'informations sur les programmes de AWS conformité, consultez AWS la section Programmes de conformité.

Lorsque vous utilisez Amazon Glacier, votre responsabilité en matière de conformité dépend de la sensibilité de vos données, des objectifs de conformité de votre entreprise et des lois et réglementations applicables. Si votre utilisation d'Amazon Glacier est soumise à la conformité à des normes telles que HIPAA, PCI ou FedRAMP, fournit des ressources pour vous aider à : AWS

- Verrou Amazon Glacier Vaultvous permet de déployer et d'appliquer facilement des contrôles de conformité pour les différents coffres-forts Amazon Glacier grâce à une politique de verrouillage des coffres-forts. Vous pouvez définir des contrôles, tels que la technique WORM (Write Once Read Many), dans une stratégie de verrouillage de coffre et empêcher que la stratégie fasse l'objet de modifications ultérieures. Une fois que la stratégie est verrouillée, elle ne peut plus être modifiée. Les politiques de verrouillage des coffres-forts peuvent vous aider à vous conformer aux cadres réglementaires tels que SEC17a -4 et HIPAA.
- Les guides de démarrage rapide sur la sécurité et la conformité abordent les considérations architecturales et les étapes à suivre pour déployer des environnements de référence axés sur la sécurité et la conformité. AWS
- Architecting for HIPAA Security and Compliance décrit la manière dont les entreprises les aident AWS à répondre aux exigences de la HIPAA.

- L'outil AWS Well-Architected AWS (WA Tool) est un service dans le cloud qui fournit un processus cohérent vous permettant de revoir et de mesurer votre architecture AWS à l'aide des meilleures pratiques. L'outil AWS WA fournit des recommandations pour rendre vos charges de travail plus fiables, sécurisées, efficaces et rentables.
- AWS Les ressources de conformité fournissent plusieurs classeurs et guides différents qui peuvent s'appliquer à votre secteur d'activité et à votre région.
- Le service AWS Config peut vous aider à évaluer comment les configurations de vos ressources se conforment aux pratiques internes, aux normes et aux directives industrielles.
- AWS Security Hub vous fournit une vue complète de l'état de votre sécurité interne AWS et vous aide à vérifier votre conformité aux normes et aux meilleures pratiques du secteur de la sécurité.

Résilience dans Amazon Glacier

L'infrastructure AWS mondiale est construite autour des régions et des zones de disponibilité. AWS offrent plusieurs zones de disponibilité physiquement séparées et isolées, qui sont reliées par un réseau à faible latence, à haut débit et hautement redondant. Ces zones de disponibilité vous offrent un moyen efficace de concevoir et d'exploiter des applications et des bases de données. Elles sont plus hautement disponibles, tolérantes aux pannes et évolutives que les infrastructures traditionnelles à un ou plusieurs centres de données. Amazon Glacier stocke les données de manière redondante sur plusieurs appareils répartis sur au moins trois zones de disponibilité. Pour augmenter la durabilité, Amazon Glacier stocke vos données de manière synchrone sur plusieurs sites AZs avant de confirmer la réussite du chargement.

Pour plus d'informations sur AWS les régions et les zones de disponibilité, consultez la section Infrastructure AWS mondiale.

Sécurité de l'infrastructure dans Amazon Glacier

En tant que service géré, Amazon Glacier (Amazon Glacier) est protégé par les procédures de sécurité du réseau AWS mondial décrites dans Amazon Web Services : présentation des processus de sécurité.

L'accès à Amazon Glacier via le réseau se fait par AWS publication APIs. Les clients doivent prendre en charge le protocole TLS (Transport Layer Security) 1.2. Nous recommandons TLS 1.3 ou version ultérieure. Les clients doivent également prendre en charge les suites de chiffrement PFS (Perfect Forward Secrecy) comme Ephemeral Diffie-Hellman (DHE) ou Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes. Par ailleurs, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser le AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires pour signer les demandes.

Points de terminaison d’un VPC

Un point de terminaison de cloud privé virtuel (VPC) vous permet de connecter de manière privée votre VPC aux services pris en charge AWS et aux services de point de terminaison VPC alimentés par le biais d'une passerelle Internet, d' AWS PrivateLink un périphérique NAT, d'une connexion VPN ou d'une connexion. AWS Direct Connect Bien qu'Amazon Glacier ne prenne pas directement en charge les points de terminaison VPC, vous pouvez tirer parti des points de terminaison VPC Amazon S3 si vous accédez à Amazon Glacier en tant que niveau de stockage intégré à Amazon S3.

Pour plus d'informations sur la configuration du cycle de vie d'Amazon S3 et la transition des objets vers la classe de stockage Amazon Glacier, consultez la section Object Lifecycle Management and Transitioning Objects dans le guide de l'utilisateur d'Amazon Simple Storage Service. Pour plus d'informations sur les points de terminaison d'un VPC, consultez Points de terminaison d'un VPC dans le Guide de l'utilisateur Amazon VPC.

Politiques de récupération des données Amazon Glacier

Grâce aux politiques de récupération des données d'Amazon Glacier, vous pouvez facilement définir des quotas de récupération de données et gérer les activités de récupération de données au sein de chacun d'entre eux. Compte AWS Région AWS Pour plus d'informations sur les frais de récupération de données Amazon Glacier, consultez la tarification d'Amazon Glacier.

> **Important**
>
> Une politique de récupération de données s'applique uniquement aux extractions standard et gère les demandes de récupération adressées directement à Amazon Glacier.
> Pour plus d'informations sur les classes de stockage Amazon Glacier, consultez les sections Classes de stockage pour l'archivage d'objets et Transitioning objects dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Rubriques

- Choisir une politique de récupération des données Amazon Glacier
- Utilisation de la console Amazon Glacier pour configurer une politique de récupération des données
- Utilisation de l'API Amazon Glacier pour configurer une politique de récupération des données

Choisir une politique de récupération des données Amazon Glacier

Vous pouvez choisir entre trois types de politiques de récupération des données Amazon Glacier : aucune limite de récupération, niveau gratuit uniquement et taux de récupération maximal.

La politique d'extraction de données Aucune limite d'extraction est celle utilisée par défaut pour les extractions. Si vous utilisez cette politique, aucun quota d'extraction n'est défini, et toutes les demandes d'extraction de données valides sont acceptées.

En appliquant une politique de niveau gratuit uniquement, vous pouvez maintenir vos extractions dans les limites de votre allocation quotidienne de niveau AWS gratuit sans encourir de frais de récupération de données. Si vous souhaitez récupérer plus de données que ne le prévoit votre allocation de niveau AWS gratuit, vous pouvez utiliser une politique de taux de récupération maximum pour définir un quota de taux de bytes-per-hour récupération. La politique du taux de

récupération maximal garantit que le taux de récupération maximal pour toutes les tâches de récupération sur votre compte Région AWS ne dépasse pas le bytes-per-hour quota que vous avez défini.

Avec les politiques Offre gratuite uniquement et Taux d'extraction maximal, les demandes d'extraction de données qui dépassent les quotas d'extraction que vous avez spécifiés ne sont pas acceptées. Si vous utilisez une politique de niveau gratuit uniquement, Amazon Glacier rejette de manière synchrone les demandes de récupération qui dépassent votre allocation de niveau AWS gratuit. Si vous utilisez une politique de taux de récupération maximal, Amazon Glacier rejette les demandes de récupération qui font en sorte que le taux de récupération maximal des tâches en cours dépasse le bytes-per-hour quota défini par la politique. Ces stratégies contribuent à simplifier la gestion des coûts d'extraction.

Voici quelques faits utiles concernant les stratégies d'extraction de données :

- Les paramètres de politique de récupération des données ne modifient pas le délai de 3 à 5 heures nécessaire pour récupérer les données d'Amazon Glacier à l'aide des extractions standard.
- La définition d'une nouvelle stratégie d'extraction de données n'affecte pas les tâches d'extraction précédemment acceptées qui sont déjà en cours.
- Si une demande de tâche d'extraction est rejetée en raison d'une politique d'extraction de données, la tâche ou la demande ne vous est pas facturée.
- Vous pouvez définir une politique de récupération de données pour chacun Région AWS, qui régira toutes les activités de récupération de données dans le Région AWS cadre de votre compte. Une politique de récupération de données est spécifique à un particulier, Région AWS car les coûts de récupération des données varient d'un pays à l'autre. Régions AWS Pour plus d'informations, consultez les tarifs d'Amazon Glacier.

Stratégie Free Tier Only

Vous pouvez définir une politique de récupération des données sur le niveau gratuit uniquement pour garantir que vos extractions restent toujours dans les limites de votre allocation de niveau AWS gratuit, afin de ne pas avoir à encourir de frais de récupération de données. Si une demande d'extraction est rejetée, vous recevez un message d'erreur indiquant que la demande a été refusée par la politique d'extraction de données actuelle.

Vous pouvez définir la politique d'extraction de données Offre gratuite uniquement par région. Une fois la politique définie, vous ne pouvez pas extraire dans une même journée plus de données que

vous le permet le quota d'extraction quotidien de l'Offre gratuite d' AWS , qui est calculé au prorata pour la Région AWS en question. Vous n'avez pas non plus à supporter de frais d'extraction de données.

Vous pouvez également passer à une politique Offre gratuite uniquement après avoir enregistré des frais d'extraction de données sur un mois. Dans ce cas, la politique Offre gratuite uniquement s'applique aux nouvelles demandes d'extraction, mais n'a aucune incidence sur les demandes passées. Vous serez facturé des frais cumulés précédemment.

Stratégie Max Retrieval Rate

Vous pouvez définir votre politique de récupération des données sur le taux d'extraction maximal afin de contrôler le taux de récupération maximal en spécifiant un quota de récupération de données ayant un maximum. bytes-per-hour Lorsque vous définissez la politique de récupération des données sur Max Retrieval Rate, une nouvelle demande de récupération est rejetée si le taux de récupération maximal des tâches en cours dépasse le bytes-per-hour quota spécifié par la politique. Si une demande de tâche d'extraction est rejetée, vous recevez un message d'erreur indiquant que la demande a été refusée par la politique d'extraction de données actuelle.

Le fait de définir votre politique de récupération des données en fonction de la politique de taux de récupération maximum peut avoir une incidence sur la quantité de votre allocation de niveau AWS gratuit que vous pouvez utiliser par jour. Par exemple, supposons que vous définissiez le taux de récupération maximum sur 1 Mo par heure. Ce taux est inférieur au taux directeur du niveau AWS gratuit. Pour vous assurer de faire bon usage de l'allocation quotidienne de niveau AWS gratuit, vous pouvez d'abord définir votre politique sur le niveau gratuit uniquement, puis passer à la politique du taux de récupération maximum ultérieurement si nécessaire. Pour plus d'informations sur le calcul de votre allocation de récupération, rendez-vous sur Amazon Glacier FAQs.

Stratégie No Retrieval Limit

Si la politique d'extraction de données que vous choisissez est Aucune limite d'extraction, toutes les demandes d'extraction de données valides sont acceptées, et vos coûts d'extraction de données varient en fonction de votre utilisation.

Utilisation de la console Amazon Glacier pour configurer une politique de récupération des données

Pour créer une politique de récupération des données à l'aide de la console Amazon Glacier

1. Connectez-vous à la console Amazon Glacier AWS Management Console et ouvrez-la https://console.aws.amazon.com/glacier/chez vous.
2. Sous Sélectionnez une région, choisissez-en une dans le Région AWS menu déroulant. Vous pouvez configurer une politique de récupération des données pour chacun Région AWS d'entre eux.
3. Dans le panneau de navigation de gauche, choisissez Paramètres d'extraction de données.
4. Choisissez Modifier. La page Modifier les politiques d'extraction de données s'affiche.
5. Sous Politiques d'extraction de données, choisissez une politique.

   Vous avez le choix entre trois politiques d'extraction de données : Aucune limite d'extraction, Offre gratuite uniquement ou Spécifiez un taux d'extraction maximal.

   - Si vous choisissez Aucune limite d'extraction, toutes les demandes d'extraction de données valides sont acceptées.
   - Si vous choisissez le niveau gratuit uniquement, les demandes de récupération de données supérieures au niveau AWS gratuit ne sont pas acceptées.
   - Si vous choisissez Spécifiez un taux d'extraction maximal, les demandes d'extraction de données sont rejetées si elles amènent le taux d'extraction maximal des tâches en cours à dépasser le taux d'extraction maximal que vous spécifiez. Vous devez spécifier une valeur en gigaoctets (Go) par heure dans le champ Go/Heure sous Taux d'extraction maximal. Lorsque vous saisissez une valeur dans Go/Heure, la console calcule automatiquement une estimation de coût.
6. Sélectionnez Enregistrer les modifications.

Utilisation de l'API Amazon Glacier pour configurer une politique de récupération des données

Vous pouvez consulter et définir une politique de récupération des données en utilisant l'API REST Amazon Glacier ou en utilisant le AWS SDKs.

Utilisation de l'API REST Amazon Glacier pour configurer une politique de récupération des données

Vous pouvez consulter et définir une politique de récupération des données à l'aide de l'API REST Amazon Glacier. Vous pouvez afficher une stratégie d'extraction de données existante en utilisant l'opération Obtention de la stratégie d'extraction de données (GET policy). Pour définir une politique d'extraction de données, utilisez l'opération Définition de stratégie d'extraction de données (PUT policy).

Lorsque vous utilisez l'opération de politique PUT, sélectionnez le type de politique d'extraction de données en définissant la valeur du champ JSON `Strategy` sur `BytesPerHour`, `FreeTier` ou `None`. Définir `BytesPerHour` revient à choisir l'option Spécifiez un taux d'extraction maximal dans la console, `FreeTier` à choisir l'option Offre gratuite uniquement et `None` à choisir l'option Aucune limite d'extraction.

Lorsque vous utilisez l'opération Lancement d'une tâche (POST jobs) pour lancer une tâche d'extraction de données qui dépasse le taux d'extraction maximal défini dans votre politique d'extraction de données, l'opération `Initiate Job` s'arrête et génère une exception.

Utilisation du AWS SDKs pour configurer une politique de récupération de données

AWS vous permet SDKs de développer des applications pour Amazon Glacier. Elles SDKs fournissent des bibliothèques qui correspondent à l'API REST sous-jacente et fournissent des objets qui vous permettent de créer facilement des demandes et de traiter des réponses. Pour de plus amples informations, veuillez consulter Utilisation du AWS SDKs avec Amazon Glacier.

Balisage des ressources Amazon Glacier

Une étiquette est une étiquette que vous attribuez à une AWS ressource. Chaque balise est constituée d'une clé et d'une valeur que vous définissez toutes deux. Vous pouvez attribuer les balises que vous définissez aux ressources du coffre-fort Amazon Glacier (Amazon Glacier). L'utilisation de balises est un moyen simple mais puissant de gérer les AWS ressources et d'organiser les données, y compris les données de facturation.

Rubriques

- Principes de base du balisage
- Restrictions liées aux balises
- Suivi des coûts à l'aide du balisage
- Gestion du contrôle d'accès avec le balisage
- Sections connexes

Principes de base du balisage

Vous utilisez la console Amazon Glacier AWS Command Line Interface (AWS CLI) ou l'API Amazon Glacier pour effectuer les tâches suivantes :

- Ajout de balises à un coffre
- Etablissement d'une liste de balises pour un coffre
- Suppression des balises d'un coffre

Pour plus d'informations sur la façon d'ajouter, de répertorier et de supprimer des balises, consultez la page Marquer vos coffres-forts Amazon Glacier.

Les balises vous permettent de classer vos coffres par catégorie. Par exemple, vous pouvez classer les coffres par objectif, propriétaire ou environnement. Dans la mesure où vous avez défini la clé et la valeur de chaque balise, vous pouvez créer un ensemble personnalisé de catégories répondant à vos besoins spécifiques. Par exemple, vous pouvez définir un ensemble de balises vous permettant de suivre les coffres par propriétaire et par objectif pour le coffre. Voici quelques exemples de balises :

- Propriétaire : nom

- Objectif : archives vidéo
- Environnement : production

Restrictions liées aux balises

Les restrictions de balise de base sont les suivantes :

- Le nombre maximum de balises pour une ressource (coffre) est de 50.
- Les clés et valeurs de balise sont sensibles à la casse.

Les restrictions clés des balises sont les suivantes :

- Au sein d'un ensemble de balises pour un coffre, chaque clé de balise doit être unique. Si vous ajoutez une balise avec une clé qui est déjà en cours d'utilisation, la nouvelle balise remplacera la paire clé-valeur existante.
- Les clés de balise ne peuvent pas commencer par `aws:` car ce préfixe est réservé à l'usage de AWS. AWS peut créer des balises commençant par ce préfixe en votre nom, mais vous ne pouvez ni les modifier ni les supprimer.
- Les clés de balise doivent comporter de 1 à 128 caractères Unicode.
- Les clés de balise doivent comporter les caractères suivants : lettres Unicode, chiffres, espaces et les caractères spéciaux suivants : `_ . / = + - @`.

Les restrictions des valeurs de balise sont les suivantes :

- Les valeurs de balise doivent comporter de 0 à 255 caractères Unicode.
- Les valeurs de balise peuvent être vides. Autrement, elles doivent se composer des caractères suivants : lettres Unicode, chiffres, espaces et les caractères spéciaux suivants :`_ . / = + - @`.

Suivi des coûts à l'aide du balisage

Vous pouvez utiliser des balises pour classer et suivre vos AWS coûts. Lorsque vous appliquez des balises à des AWS ressources, y compris des coffres-forts, votre rapport de répartition des AWS coûts inclut l'utilisation et les coûts agrégés par balises. Vous pouvez appliquer des balises associées à des catégories métier (telles que les centres de coûts, les noms d'applications et les propriétaires) pour organiser les coûts relatifs à divers services. Pour plus d'informations, consultez Utilisation des

identifications de répartition des coûts pour les rapports de facturation personnalisés dans le Guide de l'utilisateur AWS Billing .

Gestion du contrôle d'accès avec le balisage

Vous pouvez utiliser des balises comme condition dans une instruction de stratégie d'accès. Par exemple, vous pouvez configurer une balise de suspension pour motifs juridiques et l'inclure comme condition d'une stratégie de conservation des données établissant que « la suppression des archives par qui que ce soit est refusée si la valeur de la balise de suspension juridique est définie sur `True`. » Vous pouvez déployer la stratégie de conservation des données et définir la balise de suspension juridique sur `False` dans des conditions normales. Si vos données doivent être suspendues pour faciliter l'enquête, vous pouvez facilement activer la suspension juridique en définissant la valeur de la balise sur `True` et en supprimant la suspension de la même façon par la suite. Pour plus d'informations, consultez Contrôle de l'accès à l'aide de balises dans le Guide de l'utilisateur IAM.

Sections connexes

- Marquer vos coffres-forts Amazon Glacier

Journalisation des appels d'API Amazon Glacier avec AWS CloudTrail

Amazon Glacier (Amazon Glacier) est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans Amazon Glacier. CloudTrail capture tous les appels d'API pour Amazon Glacier sous forme d'événements, y compris les appels depuis la console Amazon Glacier et les appels de code vers Amazon Glacier APIs. Si vous créez un suivi, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris des événements pour Amazon Glacier. Si vous ne configurez pas de journal d'activité, vous pouvez toujours afficher les événements les plus récents dans la console CloudTrail dans Historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande envoyée à Amazon Glacier, l'adresse IP à partir de laquelle la demande a été faite, l'auteur de la demande, la date à laquelle elle a été faite, ainsi que des informations supplémentaires.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

Informations sur Amazon Glacier dans CloudTrail

CloudTrail est activé sur votre compte Compte AWS lorsque vous créez le compte. Lorsqu'une activité a lieu dans Amazon Glacier, elle 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 Compte AWS. 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 survenus dans votre environnement Compte AWS, y compris des événements relatifs à Amazon Glacier, créez un parcours. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un parcours dans la console, celui-ci s'applique à toutes les AWS régions. Le journal enregistre les événements de toutes les AWS 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 en savoir plus, consultez :

- Présentation 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 actions d'Amazon Glacier sont enregistrées CloudTrail et documentées dans leRéférence d'API pour Amazon Glacier. Par exemple, les appels aux Création de coffre (PUT vault)Suppression de coffre (DELETE vault), et Affichage des coffres (GET vaults) les 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 effectuée avec une identité d'utilisateur root ou d'autres informations d'identification.
- 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 journal Amazon Glacier

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 les Description de coffre (GET vault) actions Création de coffre (PUT vault) Suppression de coffre (DELETE vault)Affichage des coffres (GET vaults),, et.

```
{
    "Records": [
        {
```

```
            "awsRegion": "us-east-1",
            "eventID": "52f8c821-002e-4549-857f-8193a15246fa",
            "eventName": "CreateVault",
            "eventSource": "glacier.amazonaws.com",
            "eventTime": "2014-12-10T19:05:15Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "999999999999",
            "requestID": "HJiLgvfXCY88QJAC6rRoexS9ThvI21Q1Nqukfly02hcUPPo",
            "requestParameters": {
                "accountId": "-",
                "vaultName": "myVaultName"
            },
            "responseElements": {
                "location": "/999999999999/vaults/myVaultName"
            },
            "sourceIPAddress": "127.0.0.1",
            "userAgent": "aws-sdk-java/1.9.6 Mac_OS_X/10.9.5 Java_HotSpot(TM)_64-
Bit_Server_VM/25.25-b02/1.8.0_25",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "999999999999",
                "arn": "arn:aws:iam::999999999999:user/myUserName",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "myUserName"
            }
        },
        {
            "awsRegion": "us-east-1",
            "eventID": "cdd33060-4758-416a-b7b9-dafd3afcec90",
            "eventName": "DeleteVault",
            "eventSource": "glacier.amazonaws.com",
            "eventTime": "2014-12-10T19:05:15Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "999999999999",
            "requestID": "GGdw-VfhVfLCFwAM6iVUvMQ6-fMwSqSO9FmRd0eRSa_Fc7c",
            "requestParameters": {
                "accountId": "-",
                "vaultName": "myVaultName"
            },
            "responseElements": null,
            "sourceIPAddress": "127.0.0.1",
```

```
            "userAgent": "aws-sdk-java/1.9.6 Mac_OS_X/10.9.5 Java_HotSpot(TM)_64-
Bit_Server_VM/25.25-b02/1.8.0_25",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "999999999999",
                "arn": "arn:aws:iam::999999999999:user/myUserName",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "myUserName"
            }
        },
        {
            "awsRegion": "us-east-1",
            "eventID": "355750b4-e8b0-46be-9676-e786b1442470",
            "eventName": "ListVaults",
            "eventSource": "glacier.amazonaws.com",
            "eventTime": "2014-12-10T19:05:15Z",
            "eventType": "AwsApiCall",
            "eventVersion": "1.02",
            "recipientAccountId": "999999999999",
            "requestID": "yPTs22ghTsWprFivb-2u30FAaDALIZP17t4jM_xL9QJQyVA",
            "requestParameters": {
                "accountId": "-"
            },
            "responseElements": null,
            "sourceIPAddress": "127.0.0.1",
            "userAgent": "aws-sdk-java/1.9.6 Mac_OS_X/10.9.5 Java_HotSpot(TM)_64-
Bit_Server_VM/25.25-b02/1.8.0_25",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "999999999999",
                "arn": "arn:aws:iam::999999999999:user/myUserName",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "myUserName"
            }
        },
        {
            "awsRegion": "us-east-1",
            "eventID": "569e830e-b075-4444-a826-aa8b0acad6c7",
            "eventName": "DescribeVault",
            "eventSource": "glacier.amazonaws.com",
            "eventTime": "2014-12-10T19:05:15Z",
            "eventType": "AwsApiCall",
```

```
            "eventVersion": "1.02",
            "recipientAccountId": "999999999999",
            "requestID": "QRt1ZdFLGn0TCm784HmKafBmcB2lVaV81UU3fsOR3PtoIiM",
            "requestParameters": {
                "accountId": "-",
                "vaultName": "myVaultName"
            },
            "responseElements": null,
            "sourceIPAddress": "127.0.0.1",
            "userAgent": "aws-sdk-java/1.9.6 Mac_OS_X/10.9.5 Java_HotSpot(TM)_64-
Bit_Server_VM/25.25-b02/1.8.0_25",
            "userIdentity": {
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
                "accountId": "999999999999",
                "arn": "arn:aws:iam::999999999999:user/myUserName",
                "principalId": "A1B2C3D4E5F6G7EXAMPLE",
                "type": "IAMUser",
                "userName": "myUserName"
            }
        }
    ]
}
```

Référence d'API pour Amazon Glacier

Amazon Glacier prend en charge un ensemble d'opérations, en particulier un ensemble d'appels d' RESTfulAPI, qui vous permettent d'interagir avec le service.

Vous pouvez utiliser n'importe quelle bibliothèque de programmation capable d'envoyer des requêtes HTTP pour envoyer vos requêtes REST à Amazon Glacier. Lorsque vous envoyez une demande REST, Amazon Glacier exige que vous authentifiiez chaque demande en la signant. En outre, lorsque vous chargez une archive, vous devez également calculer le total de contrôle de la charge utile et l'inclure dans votre demande. Pour de plus amples informations, veuillez consulter Signature des requêtes.

En cas d'erreur, vous devez savoir quelle réponse d'erreur Amazon Glacier envoie afin de pouvoir la traiter. En plus de documenter les opérations REST, cette section fournit toutes les informations vous permettant d'effectuer des appels d'API REST directement.

Vous pouvez soit utiliser directement les appels d'API REST, soit utiliser Amazon SDKs qui fournit des bibliothèques d'encapsulation. Ces bibliothèques signent chaque demande que vous envoyez et calculent le total de contrôle de la charge utile indiquée dans votre demande. Par conséquent, l'utilisation d'Amazon SDKs simplifie votre tâche de codage. Ce guide du développeur fournit des exemples pratiques d'opérations de base sur Amazon Glacier à l'aide de AWS SDK pour Java et .NET. Pour plus d'informations, voir,Utilisation du AWS SDKs avec Amazon Glacier.

Rubriques

- En-têtes de demande communs
- En-têtes de réponse communs
- Signature des requêtes
- Calcul des totaux de contrôle
- Réponses d’erreur
- Opérations de coffre
- Opérations d'archivage
- Opérations de chargement partitionné
- Opérations de tâche
- Types de données utilisés dans les opérations de tâche
- Opérations de récupération des données

En-têtes de demande communs

Les demandes REST Amazon Glacier (Amazon Glacier) incluent des en-têtes contenant des informations de base sur la demande. Le tableau suivant décrit les en-têtes qui peuvent être utilisés par toutes les requêtes REST Amazon Glacier.

Nom de l'en-tête	Description	Obligatoire
`Authorization`	En-tête nécessaire à la signature des demandes. Amazon Glacier nécessite la version 4 de Signature. Pour de plus amples informations, veuillez consulter Signature des requêtes. Type : String	Oui
`Content-Length`	Longueur du corps de la demande (sans les en-têtes). Type : String Condition : nécessaire uniquement pour l'API Chargement d'archive (POST archive).	Conditionnel
`Date`	Date qui peut être utilisée pour créer la signature contenue dans l'en-tête `Authorization` . Si l'en-tête `Date` doit être utilisé pour la signature, il doit être spécifié au format de base ISO 8601. Dans ce cas, l'en-tête `x-amz-date` n'est pas nécessaire. Notez que lorsque `x-amz-date` est présent, il remplace toujours la valeur de l'en-tête `Date`. Si l'en-tête Date n'est pas utilisé pour la signature, il peut se présenter sous l'un des formats de date complets spécifiés par RFC 2616, section 3.3. Par exemple, la date/heure suivante `Wed, 10 Feb 2017 12:00:00 GMT` est un date/time en-tête valide à utiliser avec Amazon Glacier.	Conditionnel

Nom de l'en-tête	Description	Obligatoire
	Si vous utilisez l'en-tête `Date` pour la signature, il doit être au format de base ISO 8601`YYYYMMDD'T'HHMMSS'Z'` . Type : String Condition : si `Date`est spécifié, mais n'est pas au format de base ISO 8601 base, vous devez également inclure l'en-tête `x-amz-date` . Si `Date` est spécifié au format de base ISO 8601, cela suffit pour signer les demandes et l'en-tête `x-amz-date` n'est pas nécessaire. Pour plus d'informations, consultez la page Gestion des dates dans Signature Version 4 dans le Glossaire Amazon Web Services.	
`Host`	Cet en-tête spécifie le point de terminaison de service auquel vous envoyez vos demandes. La valeur doit être de la forme `glacier.`*`region`*`.amazonaws.com` « », *`region`* remplacée par une désignation de AWS région telle que`us-west-2` . Type : String	Oui

Nom de l'en-tête	Description	Obligatoire
`x-amz-content-sha256`	SHA256 Somme de contrôle calculée d'une charge utile complète téléchargée avec ouChargement d'archive (POST archive). Partie chargement (PUT uploadID) Cet en-tête n'est pas identique à l'en-tête `x-amz-sha256-tree-hash` bien que les valeurs soient identiques pour quelques petites charges utiles. Lorsque `x-amz-content-sha256` est obligatoire, aussi bien `x-amz-content-sha256` que `x-amz-sha256-tree-hash` doit être spécifié. Type : String Condition : obligatoire pour l'API de streaming, Chargement d'archive (POST archive) et Partie chargement (PUT uploadID).	Conditionnel
`x-amz-date`	Date utilisée pour créer la signature dans l'en-tête Authorization. Le format doit correspondre à la norme ISO 8601 de base au format `YYYYMMDD'T'HHMMSS'Z'` . Par exemple, ce qui suit date/time `20170210T120000Z` est valide `x-amz-date` pour une utilisation avec Amazon Glacier. Type : String Condition : `x-amz-date` est un en-tête facultatif pour toutes les demandes. Il peut être utilisé pour signer les demandes. Si l'en-tête `Date` est spécifié au format de base ISO 8601, `x-amz-date` n'est pas nécessaire. Lorsque `x-amz-date` est présent, il remplace toujours la valeur de l'en-tête `Date`. Pour plus d'informations, consultez la page Gestion des dates dans Signature Version 4 dans le Glossaire Amazon Web Services.	Conditionnel

Nom de l'en-tête	Description	Obligatoire
`x-amz-glacier-version`	Version de l'API Amazon Glacier à utiliser. La version actuelle est `2012-06-01` . Type : String	Oui
`x-amz-sha256-tree-hash`	Somme de contrôle calculée SHA256 par hachage arborescent pour une archive (Chargement d'archive (POST archive)) ou une partie d'archive () téléchargée. Partie chargement (PUT uploadID) Pour plus d'informations sur le calcul de ce total de contrôle, consultez la section Calcul des totaux de contrôle. Type : String Par défaut : aucun Condition : Obligatoire pour Chargement d'archive (POST archive) et Partie chargement (PUT uploadID).	Conditionnel

En-têtes de réponse communs

Le tableau suivant décrit les en-têtes de réponse communs à la plupart des réponses d'API.

Nom	Description
`Content-Length`	Longueur en octets du corps de la réponse. Type : String
`Date`	La date et l'heure de réponse d'Amazon Glacier (Amazon Glacier), par exemple,`Wed, 10 Feb 2017 12:00:00 GMT`. Le format de la date doit être l'un des formats de date complète spécifiés par RFC 2616, section 3.3. Notez que la valeur `Date` renvoyée peut être légèrement décalée par rapport aux autres dates. Par exemple, la date renvoyée par une demande

Nom	Description
	Chargement d'archive (POST archive) peut ne pas correspondre à celle qui est indiquée pour l'archive dans une liste d'inventaire pour le coffre. Type : String
`x-amzn-RequestId`	Une valeur créée par Amazon Glacier qui identifie de manière unique votre demande. Si vous rencontrez un problème avec Amazon Glacier, vous AWS pouvez utiliser cette valeur pour le résoudre. Il est recommandé d'enregistrer ces valeurs. Type : String
`x-amz-sha256-tree-hash`	Somme de contrôle SHA256 arborescente de l'archive ou du corps de l'inventaire. Pour plus d'informations sur le calcul de ce total de contrôle, consultez la section Calcul des totaux de contrôle. Type : String

Signature des requêtes

Amazon Glacier exige que vous authentifiiez chaque demande que vous envoyez en signant la demande. Pour signer une demande, vous calculez une signature numérique à l'aide d'une fonction de hachage cryptographique. Un hachage cryptographique est une fonction qui renvoie une valeur de hachage unique basée sur l'entrée. L'entrée de la fonction de hachage contient le texte de la demande et votre clé d'accès secrète. La fonction de hachage renvoie une valeur de hachage que vous incluez dans la demande comme votre signature. La signature fait partie de l'en-tête `Authorization` de votre demande.

Après réception de votre demande, Amazon Glacier recalcule la signature en utilisant la même fonction de hachage et les mêmes entrées que celles que vous avez utilisées pour signer la demande. Si la signature obtenue correspond à celle de la demande, Amazon Glacier traite la demande. Sinon, la demande est rejetée.

Amazon Glacier prend en charge l'authentification à l'aide de AWS Signature Version 4. Le processus de calcul d'une signature peut être divisé en trois tâches :

- Tâche 1 : créer une demande canonique

  Réorganiser votre demande HTTP dans un format canonique. L'utilisation d'un formulaire canonique est nécessaire car Amazon Glacier utilise le même formulaire canonique lorsqu'il recalcule une signature pour la comparer à celle que vous avez envoyée.

- Tâche 2 : créer une chaîne de connexion

  Créez une chaîne que vous utiliserez comme une des valeurs d'entrée pour votre fonction de hachage cryptographique. La chaîne, appelée la chaîne de connexion, est une concaténation du nom de l'algorithme de hachage, de la date de la demande, d'une chaîne d'informations d'identification et de la demande convertie sous forme canonique de la tâche précédente. La chaîne d'étendue des informations d'identification elle-même est une concaténation d'informations de date, de AWS région et de service.

- Tâche 3 : créer une signature

  Créez une signature pour votre demande à l'aide d'une fonction de hachage cryptographique qui accepte deux chaînes d'entrée : votre chaîne de connexion et une clé dérivée. La clé dérivée est calculée en commençant par votre clé d'accès secrète et en utilisant la chaîne de portée des informations d'identification pour créer une série de codes d'authentification des messages basés sur le hachage ()HMACs. Notez que la fonction de hachage utilisée dans cette étape de signature n'est pas l'algorithme de hachage arborescent utilisé dans Amazon Glacier pour charger les données. APIs

Rubriques

- Exemple de calcul de signature
- Calcul des Signatures pour les opérations de streaming

Exemple de calcul de signature

L'exemple suivant vous guide à travers les détails de la création d'une signature pour Création de coffre (PUT vault). L'exemple peut être utilisé comme référence pour vérifier votre méthode de calcul de signature. Pour plus d'informations, consultez la section Signature des demandes AWS d'API dans le guide de l'utilisateur IAM.

Dans cet exemple il est supposé que :

- L'horodatage de la demande est `Fri, 25 May 2012 00:24:53 GMT`.

- Le point de terminaison est la région USA Est (Virginie du Nord), `us-east-1`.

La syntaxe générale de la demande (y compris le corps JSON) est :

```
PUT /-/vaults/examplevault HTTP/1.1
Host: glacier.us-east-1.amazonaws.com
Date: Fri, 25 May 2012 00:24:53 GMT
Authorization: SignatureToBeCalculated
x-amz-glacier-version: 2012-06-01
```

La forme canonique de la demande calculée pour Tâche 1 : Créer une demande canonique est :

```
PUT
/-/vaults/examplevault

host:glacier.us-east-1.amazonaws.com
x-amz-date:20120525T002453Z
x-amz-glacier-version:2012-06-01

host;x-amz-date;x-amz-glacier-version
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

La dernière ligne de la demande canonique est le hachage du corps de la demande. Notez également la troisième ligne vide dans la demande canonique. Elle provient du fait qu'il n'y a aucun paramètre de requête pour cette API.

La chaîne de connexion pour la tâche 2 : créer une chaîne de connexion est :

```
AWS4-HMAC-SHA256
20120525T002453Z
20120525/us-east-1/glacier/aws4_request
5f1da1a2d0feb614dd03d71e87928b8e449ac87614479332aced3a701f916743
```

La première ligne de la chaîne de connexion est l'algorithme, la deuxième ligne est l'horodatage, la troisième ligne est les informations d'identification, et la dernière ligne est un hachage de la demande canonique issu de la tâche 1 : créer une demande canonique. Le nom du service à utiliser dans les informations d'identification est `glacier`.

Pour la tâche 3 : créer une signature, la clé dérivée peut être représentée sous la forme :

```
derived key = HMAC(HMAC(HMAC(HMAC("AWS4" + YourSecretAccessKey,"20120525"),"us-
east-1"),"glacier"),"aws4_request")
```

Si la clé d'accès secrète `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY` est utilisée, la signature calculée est :

```
3ce5b2f2fffac9262b4da9256f8d086b4aaf42eba5f111c21681a65a127b7c2a
```

L'étape finale consiste à construire l'en-tête `Authorization`. Pour la clé d'accès de la démonstration `AKIAIOSFODNN7EXAMPLE`, l'en-tête (avec les sauts de ligne ajoutés pour faciliter la lecture) est :

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20120525/us-east-1/
glacier/aws4_request,
SignedHeaders=host;x-amz-date;x-amz-glacier-version,
Signature=3ce5b2f2fffac9262b4da9256f8d086b4aaf42eba5f111c21681a65a127b7c2a
```

Calcul des Signatures pour les opérations de streaming

Chargement d'archive (POST archive) et Partie chargement (PUT uploadID) sont des opérations de streaming qui exigent d'inclure un en-tête supplémentaire `x-amz-content-sha256` lors de la signature et de l'envoi de votre demande. Les étapes de signature pour les opérations de streaming sont identiques à celles des autres opérations, mais comprennent aussi l'ajout de l'en-tête de streaming.

Le calcul de l'en-tête de diffusion `x-amz-content-sha256` est basé sur le SHA256 hachage de l'ensemble du contenu (charge utile) à télécharger. Notez que ce calcul est différent de l' SHA256 arbre hash (Calcul des totaux de contrôle). Hormis les cas triviaux, la valeur de hachage SHA 256 des données de charge utile sera différente du hachage SHA256 arborescent des données de charge utile.

Si les données de charge utile sont spécifiées sous forme de tableau d'octets, vous pouvez utiliser l'extrait de code Java suivant pour calculer le hachage. SHA256

```
public static byte[] computePayloadSHA256Hash2(byte[] payload) throws
 NoSuchAlgorithmException, IOException {
```

```
    BufferedInputStream bis =
       new BufferedInputStream(new ByteArrayInputStream(payload));
    MessageDigest messageDigest = MessageDigest.getInstance("SHA-256");
    byte[] buffer = new byte[4096];
    int bytesRead = -1;
    while ( (bytesRead = bis.read(buffer, 0, buffer.length)) != -1 ) {
        messageDigest.update(buffer, 0, bytesRead);
    }
    return messageDigest.digest();
}
```

De même, en C#, vous pouvez calculer le SHA256 hachage des données de charge utile comme indiqué dans l'extrait de code suivant.

```
public static byte[] CalculateSHA256Hash(byte[] payload)
{
    SHA256 sha256 = System.Security.Cryptography.SHA256.Create();
    byte[] hash = sha256.ComputeHash(payload);

    return hash;
}
```

Exemple de calcul de signature pour l'API de streaming

L'exemple suivant explique en détail comment créer une signature pour l'un des deux Chargement d'archive (POST archive) types de diffusion en continu APIs dans Amazon Glacier. Dans cet exemple il est supposé que :

- L'horodatage de la demande est `Mon, 07 May 2012 00:00:00 GMT`.
- Le point de terminaison est la région USA Est (Virginie du Nord), us-east-1.
- La charge utile du contenu est une chaîne « Bienvenue sur Amazon Glacier ».

La syntaxe de demande générale (y compris le corps JSON) est illustrée dans l'exemple ci-dessous. Notez que l'en-tête `x-amz-content-sha256` est inclus. Dans cet exemple simplifié, le paramètre `x-amz-sha256-tree-hash` et le paramètre `x-amz-content-sha256` ont la même valeur. Cependant, ce n'est pas le cas pour les téléchargements d'archives supérieures à 1 Mo.

```
POST /-/vaults/examplevault HTTP/1.1
Host: glacier.us-east-1.amazonaws.com
```

```
Date: Mon, 07 May 2012 00:00:00 GMT
x-amz-archive-description: my archive
x-amz-sha256-tree-hash: SHA256 tree hash
x-amz-content-sha256: SHA256 payload hash
Authorization: SignatureToBeCalculated
x-amz-glacier-version: 2012-06-01
```

La forme canonique de la demande calculée pour la tâche 1 : créer une demande canonique est illustrée ci-dessous : Notez que l'en-tête de streaming `x-amz-content-sha256` est inclus avec sa valeur. Cela signifie que vous devez d'abord lire la charge utile et calculer le SHA256 hachage, puis calculer la signature.

```
POST
/-/vaults/examplevault

host:glacier.us-east-1.amazonaws.com
x-amz-content-sha256:726e392cb4d09924dbad1cc0ba3b00c3643d03d14cb4b823e2f041cff612a628
x-amz-date:20120507T000000Z
x-amz-glacier-version:2012-06-01

host;x-amz-content-sha256;x-amz-date;x-amz-glacier-version
726e392cb4d09924dbad1cc0ba3b00c3643d03d14cb4b823e2f041cff612a628
```

Le reste du calcul de la signature suit les étapes décrites dans Exemple de calcul de signature. L'en-tête `Authorization` qui utilise la clé d'accès secrète `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY` et la clé d'accès `AKIAIOSFODNN7EXAMPLE` est illustré ci-dessous (avec des sauts de ligne ajoutés pour faciliter la lecture) :

```
Authorization=AWS4-HMAC-SHA256
Credential=AKIAIOSFODNN7EXAMPLE/20120507/us-east-1/glacier/aws4_request,
SignedHeaders=host;x-amz-content-sha256;x-amz-date;x-amz-glacier-version,
Signature=b092397439375d59119072764a1e9a144677c43d9906fd98a5742c57a2855de6
```

Calcul des totaux de contrôle

Lors du chargement d'une archive, vous devez inclure les deux en-têtes `x-amz-sha256-tree-hash` et `x-amz-content-sha256`. L'en-tête `x-amz-sha256-tree-hash` est un total de contrôle

de la charge utile dans le corps de votre demande. Cette rubrique explique comment calculer l'en-tête `x-amz-sha256-tree-hash`. L'en-tête `x-amz-content-sha256` est un hachage de la totalité de la charge utile et est obligatoire pour l'autorisation. Pour de plus amples informations, veuillez consulter Exemple de calcul de signature pour l'API de streaming.

La charge utile de votre demande peut être :

- Une archive entière – Lorsque vous chargez une archive dans une même demande à l'aide de l'API de chargement d'archive, vous envoyez l'archive entière dans le corps de la demande. Dans ce cas, vous devez inclure le total de contrôle de l'intégralité de l'archive.
- Une partie d'archive – Lorsque vous chargez une archive en plusieurs parties à l'aide de l'API de chargement partitionné, vous envoyez seulement une partie de l'archive dans le corps de la demande. Dans ce cas, vous incluez le total de contrôle de la partie de l'archive. Et, une fois que vous avez chargé toutes les parties, vous envoyez une demande d'achèvement de chargement partitionné qui doit inclure le total de contrôle de la totalité de l'archive.

Le total de contrôle de la charge utile est un hachage d'arborescence SHA-256. Il est appelé hachage d'arborescence car dans le processus de calcul du total de contrôle, vous calculez une arborescence des valeurs de hachage SHA-256. La valeur de hachage située à la racine est le total de contrôle de la totalité de l'archive.

> **Note**
>
> Cette section décrit une procédure permettant de calculer le hachage d'arborescence SHA-256. Vous pouvez toutefois utiliser n'importe quelle autre procédure à condition que celle-ci génère le même résultat.

Pour calculer le hachage d'arborescence SHA-256, procédez comme suit :

1. Pour chaque bloc de 1 Mo des données de charge utile, calculez le hachage SHA-256. Le dernier bloc de données peut être inférieur à 1 Mo. Par exemple, si vous chargez une archive de 3,2 Mo, vous calculez les valeurs de hachage SHA-256 pour chacun des trois premiers blocs de 1 Mo de données, puis vous calculez le hachage SHA-256 pour le dernier bloc de 0,2 Mo de données. Ces valeurs de hachage constituent les nœuds terminaux de l'arborescence.

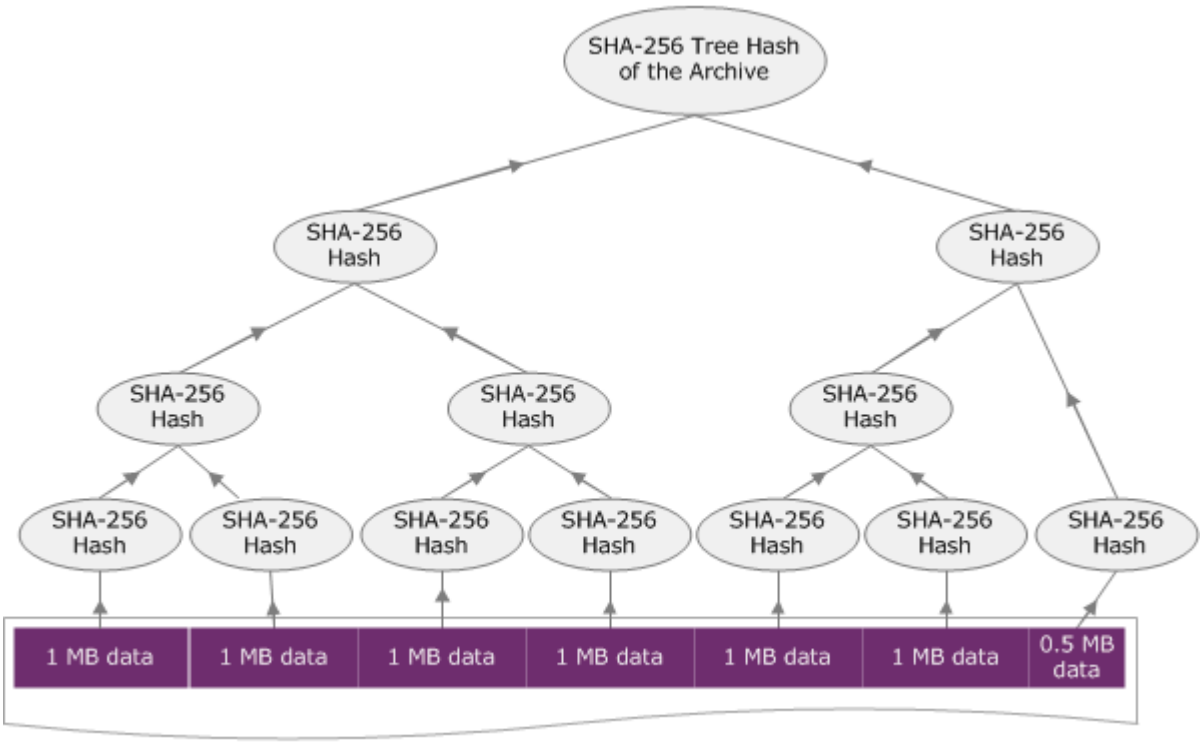
2. Créez le niveau suivant de l'arborescence.
   a. Concaténez deux valeurs de hachage de nœud enfant consécutives et calculez le hachage SHA-256 des valeurs de hachage concaténées. Cette concaténation et la génération du hachage SHA-256 produisent un nœud parent pour les deux nœuds enfants.
   b. S'il ne reste qu'un nœud enfant, faites remonter cette valeur de hachage au niveau supérieur dans l'arborescence.
3. Répétez l'étape 2 jusqu'à ce que l'arborescence obtenue possède une racine. La racine de l'arborescence fournit un hachage de la totalité de l'archive et une racine de la sous-arborescence appropriée fournit le hachage pour la partie dans le cadre d'un chargement partitionné.

Rubriques

- Exemple de hachage d'arborescence 1 : Chargement d'une archive dans le cadre d'une seule demande
- Exemple de hachage d'arborescence 2 : chargement d'une archive à l'aide d'un chargement partitionné
- Calcul du hachage d'arborescence d'un fichier
- Réception des totaux de contrôle lors du téléchargement de données

Exemple de hachage d'arborescence 1 : Chargement d'une archive dans le cadre d'une seule demande

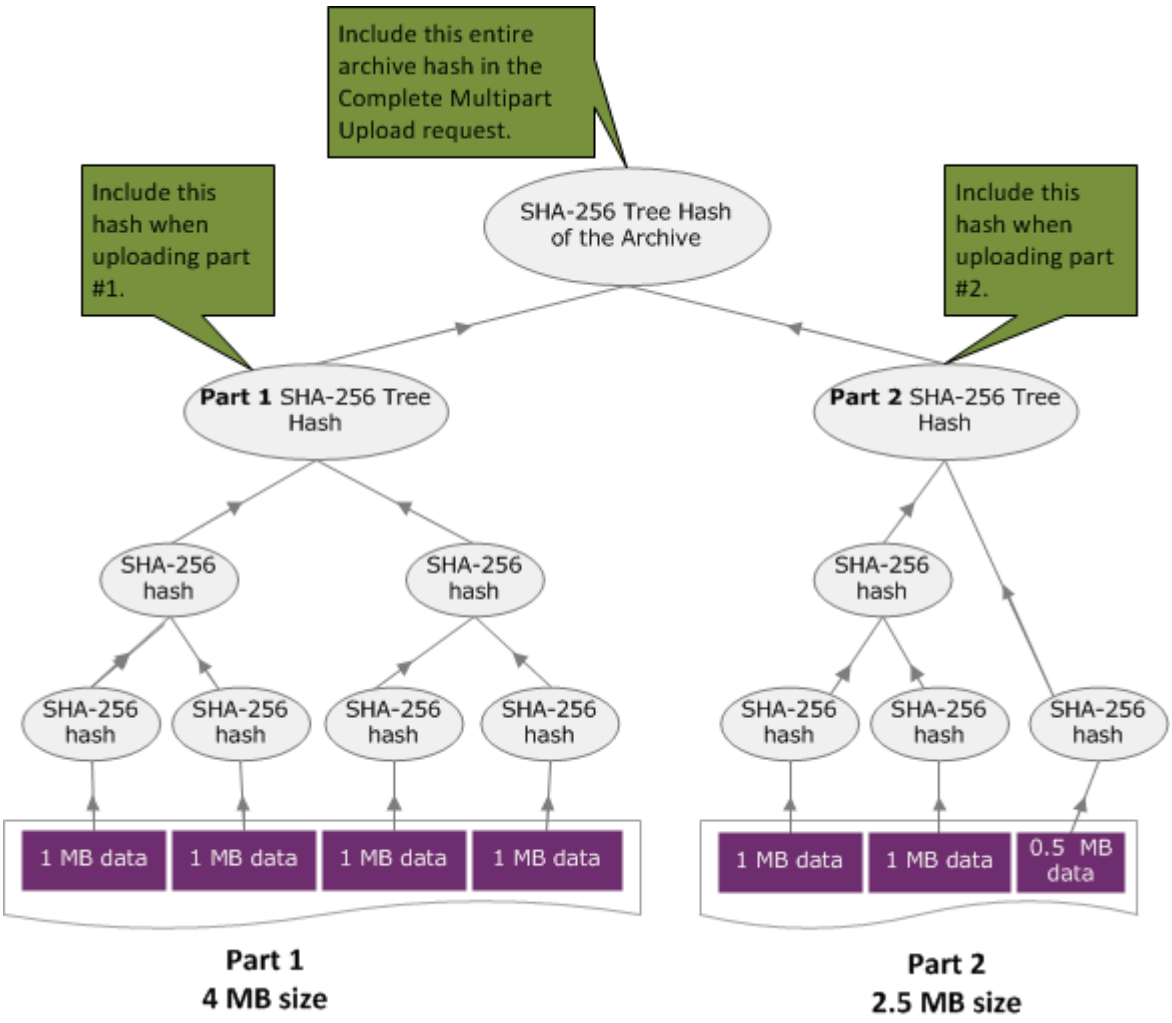
Lorsque vous chargez une archive dans le cadre d'une seule demande à l'aide de l'API de chargement d'archive (consultez la section Chargement d'archive (POST archive)), la charge utile de la demande comprend la totalité de l'archive. En conséquence, vous devez inclure le hachage d'arborescence de la totalité de l'archive dans l'en-tête de demande `x-amz-sha256-tree-hash`. Supposons que vous souhaitiez charger une archive de 6,5 Mo. Le schéma suivant illustre le processus de création du hachage SHA-256 de l'archive. Vous lisez l'archive et calculez le hachage SHA-256 pour chaque bloc de 1 Mo. Vous calculez également le hachage du dernier bloc de 0,5 Mo de données, puis vous créez l'arborescence comme indiqué dans la procédure précédente.



6.5 MB Archive

Exemple de hachage d'arborescence 2 : chargement d'une archive à l'aide d'un chargement partitionné

Le processus de calcul du hachage d'arborescence lors du chargement d'une archive à l'aide du chargement partitionné est identique à celui utilisé lors du chargement de l'archive dans le cadre d'une seule demande. La seule différence réside dans le fait que pour un chargement partitionné, vous ne chargez qu'une partie de l'archive dans chaque demande (à l'aide de l'API Partie chargement (PUT uploadID)), et par conséquent, vous ne fournissez que le total de contrôle de la partie dans l'en-tête de demande `x-amz-sha256-tree-hash`. Cependant, une fois que vous avez chargé toutes les parties, vous devez envoyer la demande d'achèvement de chargement partitionné (consultez la section Achèvement du chargement partitionné (POST uploadID)) avec un hachage d'arborescence de la totalité de l'archive dans l'en-tête de demande `x-amz-sha256-tree-hash`.



Calcul du hachage d'arborescence d'un fichier

Les algorithmes présentés ici ont été sélectionnés à des fins de démonstration. Vous pouvez optimiser le code en fonction des besoins de votre scénario d'implémentation. Si vous utilisez un SDK Amazon pour programmer sur Amazon Glacier (Amazon Glacier), le calcul du hachage de l'arbre est effectué pour vous et il vous suffit de fournir la référence du fichier.

Example 1 : Exemple Java

L'exemple suivant montre comment calculer le hachage SHA256 arborescent d'un fichier à l'aide de Java. Vous pouvez exécuter cet exemple en fournissant un emplacement de fichier comme argument ou vous pouvez utiliser la méthode `TreeHashExample.computeSHA256TreeHash` directement à partir de votre code.

```
import java.io.File;
import java.io.FileInputStream;
```

```
import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

public class TreeHashExample {

static final int ONE_MB = 1024 * 1024;

    /**
     * Compute the Hex representation of the SHA-256 tree hash for the specified
     * File
     *
     * @param args
     *            args[0]: a file to compute a SHA-256 tree hash for
     */
    public static void main(String[] args) {

        if (args.length < 1) {
            System.err.println("Missing required filename argument");
            System.exit(-1);
        }

        File inputFile = new File(args[0]);
        try {

            byte[] treeHash = computeSHA256TreeHash(inputFile);
            System.out.printf("SHA-256 Tree Hash = %s\n", toHex(treeHash));

        } catch (IOException ioe) {
            System.err.format("Exception when reading from file %s: %s", inputFile,
                    ioe.getMessage());
            System.exit(-1);

        } catch (NoSuchAlgorithmException nsae) {
            System.err.format("Cannot locate MessageDigest algorithm for SHA-256: %s",
                    nsae.getMessage());
            System.exit(-1);
        }
    }

    /**
     * Computes the SHA-256 tree hash for the given file
     *
     * @param inputFile
```

```
     *            a File to compute the SHA-256 tree hash for
     * @return a byte[] containing the SHA-256 tree hash
     * @throws IOException
     *             Thrown if there's an issue reading the input file
     * @throws NoSuchAlgorithmException
     */
    public static byte[] computeSHA256TreeHash(File inputFile) throws IOException,
            NoSuchAlgorithmException {

        byte[][] chunkSHA256Hashes = getChunkSHA256Hashes(inputFile);
        return computeSHA256TreeHash(chunkSHA256Hashes);
    }

    /**
     * Computes a SHA256 checksum for each 1 MB chunk of the input file. This
     * includes the checksum for the last chunk even if it is smaller than 1 MB.
     *
     * @param file
     *            A file to compute checksums on
     * @return a byte[][] containing the checksums of each 1 MB chunk
     * @throws IOException
     *             Thrown if there's an IOException when reading the file
     * @throws NoSuchAlgorithmException
     *             Thrown if SHA-256 MessageDigest can't be found
     */
    public static byte[][] getChunkSHA256Hashes(File file) throws IOException,
            NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");

        long numChunks = file.length() / ONE_MB;
        if (file.length() % ONE_MB > 0) {
            numChunks++;
        }

        if (numChunks == 0) {
            return new byte[][] { md.digest() };
        }

        byte[][] chunkSHA256Hashes = new byte[(int) numChunks][];
        FileInputStream fileStream = null;

        try {
            fileStream = new FileInputStream(file);
```

```
            byte[] buff = new byte[ONE_MB];

            int bytesRead;
            int idx = 0;
            int offset = 0;

            while ((bytesRead = fileStream.read(buff, offset, ONE_MB)) > 0) {
                md.reset();
                md.update(buff, 0, bytesRead);
                chunkSHA256Hashes[idx++] = md.digest();
                offset += bytesRead;
            }

            return chunkSHA256Hashes;

        } finally {
            if (fileStream != null) {
                try {
                    fileStream.close();
                } catch (IOException ioe) {
                    System.err.printf("Exception while closing %s.\n %s",
file.getName(),
                            ioe.getMessage());
                }
            }
        }
    }

    /**
     * Computes the SHA-256 tree hash for the passed array of 1 MB chunk
     * checksums.
     *
     * This method uses a pair of arrays to iteratively compute the tree hash
     * level by level. Each iteration takes two adjacent elements from the
     * previous level source array, computes the SHA-256 hash on their
     * concatenated value and places the result in the next level's destination
     * array. At the end of an iteration, the destination array becomes the
     * source array for the next level.
     *
     * @param chunkSHA256Hashes
     *            An array of SHA-256 checksums
     * @return A byte[] containing the SHA-256 tree hash for the input chunks
     * @throws NoSuchAlgorithmException
     *             Thrown if SHA-256 MessageDigest can't be found
```

```
     */
    public static byte[] computeSHA256TreeHash(byte[][] chunkSHA256Hashes)
            throws NoSuchAlgorithmException {

        MessageDigest md = MessageDigest.getInstance("SHA-256");

        byte[][] prevLvlHashes = chunkSHA256Hashes;

        while (prevLvlHashes.length > 1) {

            int len = prevLvlHashes.length / 2;
            if (prevLvlHashes.length % 2 != 0) {
                len++;
            }

            byte[][] currLvlHashes = new byte[len][];

            int j = 0;
            for (int i = 0; i < prevLvlHashes.length; i = i + 2, j++) {

                // If there are at least two elements remaining
                if (prevLvlHashes.length - i > 1) {

                    // Calculate a digest of the concatenated nodes
                    md.reset();
                    md.update(prevLvlHashes[i]);
                    md.update(prevLvlHashes[i + 1]);
                    currLvlHashes[j] = md.digest();

                } else { // Take care of remaining odd chunk
                    currLvlHashes[j] = prevLvlHashes[i];
                }
            }

            prevLvlHashes = currLvlHashes;
        }

        return prevLvlHashes[0];
    }

    /**
     * Returns the hexadecimal representation of the input byte array
     *
     * @param data
```

```
     *            a byte[] to convert to Hex characters
     * @return A String containing Hex characters
     */
    public static String toHex(byte[] data) {
        StringBuilder sb = new StringBuilder(data.length * 2);

        for (int i = 0; i < data.length; i++) {
            String hex = Integer.toHexString(data[i] & 0xFF);

            if (hex.length() == 1) {
                // Append leading zero.
                sb.append("0");
            }
            sb.append(hex);
        }
        return sb.toString().toLowerCase();
    }
}
```

Example 2 : Exemple C# .NET

L'exemple suivant montre comment calculer le hachage SHA256 arborescent d'un fichier. Vous pouvez exécuter cet exemple en fournissant un emplacement de fichier comme argument.

```
using System;
using System.IO;

using System.Security.Cryptography;

namespace ExampleTreeHash
{
    class Program
    {
        static int ONE_MB = 1024 * 1024;

        /**
        * Compute the Hex representation of the SHA-256 tree hash for the
        * specified file
        *
        * @param args
        *            args[0]: a file to compute a SHA-256 tree hash for
        */
        public static void Main(string[] args)
```

```
        {
            if (args.Length < 1)
            {
                Console.WriteLine("Missing required filename argument");
                Environment.Exit(-1);
            }
            FileStream inputFile = File.Open(args[0], FileMode.Open, FileAccess.Read);
            try
            {
                byte[] treeHash = ComputeSHA256TreeHash(inputFile);
                Console.WriteLine("SHA-256 Tree Hash = {0}",
BitConverter.ToString(treeHash).Replace("-", "").ToLower());
                Console.ReadLine();
                Environment.Exit(-1);
            }
            catch (IOException ioe)
            {
                Console.WriteLine("Exception when reading from file {0}: {1}",
                    inputFile, ioe.Message);
                Console.ReadLine();
                Environment.Exit(-1);
            }
            catch (Exception e)
            {
                Console.WriteLine("Cannot locate MessageDigest algorithm for SHA-256:
{0}",
                    e.Message);
                Console.WriteLine(e.GetType());
                Console.ReadLine();
                Environment.Exit(-1);
            }
            Console.ReadLine();
        }


        /**
         * Computes the SHA-256 tree hash for the given file
         *
         * @param inputFile
         *            A file to compute the SHA-256 tree hash for
         * @return a byte[] containing the SHA-256 tree hash
         */
        public static byte[] ComputeSHA256TreeHash(FileStream inputFile)
        {
```

```
            byte[][] chunkSHA256Hashes = GetChunkSHA256Hashes(inputFile);
            return ComputeSHA256TreeHash(chunkSHA256Hashes);
        }


        /**
         * Computes a SHA256 checksum for each 1 MB chunk of the input file. This
         * includes the checksum for the last chunk even if it is smaller than 1 MB.
         *
         * @param file
         *            A file to compute checksums on
         * @return a byte[][] containing the checksums of each 1MB chunk
         */
        public static byte[][] GetChunkSHA256Hashes(FileStream file)
        {
            long numChunks = file.Length / ONE_MB;
            if (file.Length % ONE_MB > 0)
            {
                numChunks++;
            }

            if (numChunks == 0)
            {
                return new byte[][] { CalculateSHA256Hash(null, 0) };
            }
            byte[][] chunkSHA256Hashes = new byte[(int)numChunks][];

            try
            {
                byte[] buff = new byte[ONE_MB];

                int bytesRead;
                int idx = 0;

                while ((bytesRead = file.Read(buff, 0, ONE_MB)) > 0)
                {
                    chunkSHA256Hashes[idx++] = CalculateSHA256Hash(buff, bytesRead);
                }
                return chunkSHA256Hashes;
            }
            finally
            {
                if (file != null)
                {
```

```
                try
                {
                    file.Close();
                }
                catch (IOException ioe)
                {
                    throw ioe;
                }
            }
        }

    }

    /**
     * Computes the SHA-256 tree hash for the passed array of 1MB chunk
     * checksums.
     *
     * This method uses a pair of arrays to iteratively compute the tree hash
     * level by level. Each iteration takes two adjacent elements from the
     * previous level source array, computes the SHA-256 hash on their
     * concatenated value and places the result in the next level's destination
     * array. At the end of an iteration, the destination array becomes the
     * source array for the next level.
     *
     * @param chunkSHA256Hashes
     *            An array of SHA-256 checksums
     * @return A byte[] containing the SHA-256 tree hash for the input chunks
     */
    public static byte[] ComputeSHA256TreeHash(byte[][] chunkSHA256Hashes)
    {
        byte[][] prevLvlHashes = chunkSHA256Hashes;
        while (prevLvlHashes.GetLength(0) > 1)
        {

            int len = prevLvlHashes.GetLength(0) / 2;
            if (prevLvlHashes.GetLength(0) % 2 != 0)
            {
                len++;
            }

            byte[][] currLvlHashes = new byte[len][];

            int j = 0;
            for (int i = 0; i < prevLvlHashes.GetLength(0); i = i + 2, j++)
```

```
                {

                    // If there are at least two elements remaining
                    if (prevLvlHashes.GetLength(0) - i > 1)
                    {

                        // Calculate a digest of the concatenated nodes
                        byte[] firstPart = prevLvlHashes[i];
                        byte[] secondPart = prevLvlHashes[i + 1];
                        byte[] concatenation = new byte[firstPart.Length +
 secondPart.Length];
                        System.Buffer.BlockCopy(firstPart, 0, concatenation, 0,
 firstPart.Length);
                        System.Buffer.BlockCopy(secondPart, 0, concatenation,
 firstPart.Length, secondPart.Length);

                        currLvlHashes[j] = CalculateSHA256Hash(concatenation,
 concatenation.Length);

                    }
                    else
                    { // Take care of remaining odd chunk
                        currLvlHashes[j] = prevLvlHashes[i];
                    }
                }

                prevLvlHashes = currLvlHashes;
            }

            return prevLvlHashes[0];
        }

        public static byte[] CalculateSHA256Hash(byte[] inputBytes, int count)
        {
            SHA256 sha256 = System.Security.Cryptography.SHA256.Create();
            byte[] hash = sha256.ComputeHash(inputBytes, 0, count);
            return hash;
        }
    }
}
```

Réception des totaux de contrôle lors du téléchargement de données

Lorsque vous extrayez une archive à l'aide de l'API de lancement de tâche (consultez la section Lancement d'une tâche (POST jobs)), vous pouvez éventuellement spécifier une plage à extraire de l'archive. De même, lorsque vous téléchargez vos données à l'aide de l'API de génération de sortie de tâche (consultez la section Génération de sortie de tâche (GET output)), vous pouvez éventuellement spécifier une plage de données à télécharger. Vous devez comprendre deux caractéristiques essentielles de ces plages lorsque vous procédez à l'extraction et au téléchargement des données de votre archive. La plage à extraire doit être alignée en termes de méga-octets avec l'archive. La plage à extraire et la plage à télécharger doivent toutes les deux être alignées avec le hachage d'arborescence pour vous permettre de recevoir les valeurs de total de contrôle lorsque vous téléchargez vos données. Les définitions de ces deux types d'alignement de plage sont les suivantes :

- Aligné en mégaoctets : une plage [StartByte, EndBytes] est alignée en mégaoctets (1024* 1024) lorsqu'elle StartBytesest divisible par 1 Mo et EndBytesplus 1 est divisible par 1 Mo ou est égale à la fin de l'archive spécifiée (taille d'octet d'archive moins 1). Une plage utilisée dans l'API de lancement de tâche, si elle est spécifiée, doit être alignée en termes de méga-octets.
- Aligné par arbre - Une plage [StartBytes, EndBytes] est un hachage arborescent aligné par rapport à une archive si et seulement si la racine du hachage arborescent construit sur cette plage est équivalente à un nœud dans le hachage arborescent de l'ensemble de l'archive. La plage à extraire et la plage à télécharger doivent toutes les deux être alignées avec le hachage d'arborescence pour vous permettre de recevoir les valeurs de total de contrôle des données que vous téléchargez. Pour obtenir un exemple de plages et de leur relation avec le hachage d'arborescence d'une archive, consultez la section Exemple de hachage d'arborescence : Extraction d'une plage d'une archive qui est alignée avec le hachage d'arborescence.

  Notez qu'une plage alignée avec le hachage d'arborescence est également alignée en termes de méga-octets. En revanche, une plage alignée en termes de méga-octets n'est pas nécessairement alignée avec le hachage d'arborescence.

Les cas suivants décrivent à quel moment vous recevez une valeur de total de contrôle lorsque vous téléchargez vos données archivées :

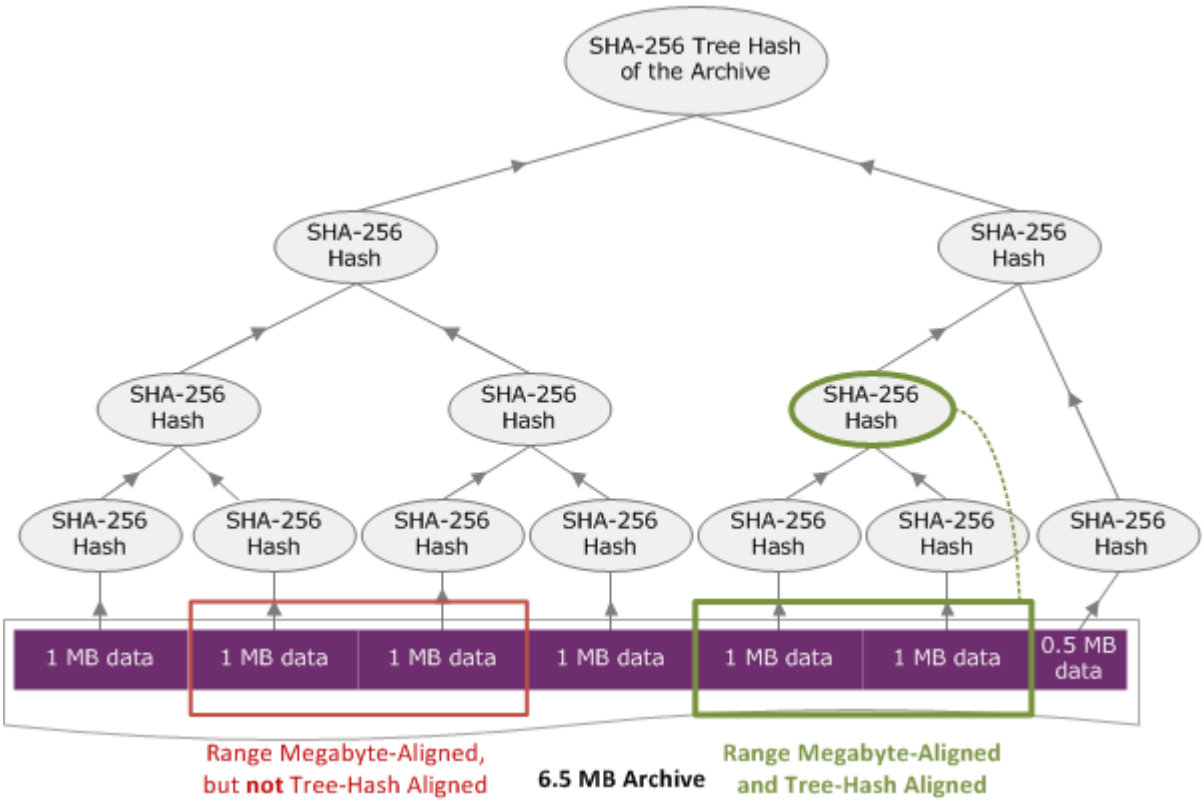
- Si vous ne spécifiez pas de plage à extraire dans la demande de lancement de tâche et que vous téléchargez la totalité de l'archive dans la demande d'affichage de tâche.

- Si vous ne spécifiez pas de plage à extraire dans la demande de lancement de tâche et que vous indiquez une plage à télécharger dans la demande d'affichage de tâche qui est alignée avec le hachage d'arborescence.
- Si vous spécifiez une plage à extraire dans la demande de lancement de tâche qui est alignée avec le hachage d'arborescence et que vous téléchargez la totalité de la plage dans la demande d'affichage de tâche.
- Si vous spécifiez une plage à extraire dans la demande de lancement de tâche qui est alignée avec le hachage d'arborescence et que vous indiquez une plage à télécharger dans la demande d'affichage de tâche qui est alignée avec le hachage d'arborescence.

Si vous spécifiez une plage à extraire dans la demande de lancement de tâche qui n'est pas alignée avec le hachage d'arborescence, vous pouvez tout de même obtenir vos données archivées, mais aucune valeur de total de contrôle n'est renvoyée lorsque vous téléchargez des données dans le cadre de la demande d'affichage de tâche.

Exemple de hachage d'arborescence : Extraction d'une plage d'une archive qui est alignée avec le hachage d'arborescence

Imaginons que votre coffre contient une archive de 6,5 Mo et que vous souhaitez extraire 2 Mo de l'archive. La façon dont vous spécifiez la plage de 2 Mo dans la demande de lancement de tâche détermine si vous recevez des valeurs de total de contrôle de données lorsque vous téléchargez vos données. Le schéma suivant illustre deux plages de 2 Mo que vous pouvez télécharger pour l'archive de 6,5 Mo. Les deux plages sont alignées en termes de méga-octets, mais une seule est alignée avec le hachage d'arborescence.



Spécification de plage alignée avec le hachage d'arborescence

Cette section fournit la spécification exacte d'une plage alignée avec le hachage d'arborescence. Les plages alignées avec le hachage d'arborescence sont essentielles lorsque vous téléchargez une partie d'une archive et que vous spécifiez la plage de données à extraire et la plage de données extraites à télécharger. Si ces deux plages sont alignées avec le hachage d'arborescence, vous recevez des valeurs de total de contrôle lorsque vous téléchargez les données.

Une plage [A, B] est alignée avec le hachage d'arborescence par rapport à une archive si et seulement si lorsqu'un nouveau hachage d'arborescence est créé sur [A, B], la racine du hachage d'arborescence de cette plage équivaut à un nœud du hachage de la totalité de l'archive. Pour obtenir une illustration, consultez le schéma dans la section Exemple de hachage d'arborescence : Extraction d'une plage d'une archive qui est alignée avec le hachage d'arborescence. Cette section fournit la spécification de l'alignement avec le hachage d'arborescence.

Prenez par exemple [P, Q) comme requête de plage pour une archive de N méga-octets (Mo) et P et Q comme multiples d'un Mo. Notez que la plage inclusive réelle est [P Mo, Q MB - 1 octet], mais pour simplifier, nous la présentons sous la forme [P, Q). Ces considérations étant présentées,

- Si P est un nombre impair, il n'y a qu'une seule plage possible à être alignée sur le hachage d'arborescence, à savoir [P, P + 1 Mo).
- Si P est un nombre pair et que k est le nombre maximal, où P peut être écrit sous la forme 2k * X, il y a au maximum k plages alignées sur le hachage d'arborescence qui commencent par P. X est un entier supérieur à 0. Les plages alignées avec le hachage d'arborescence sont classées dans les catégories suivantes :
  - Pour chaque i, où (0 <= i <= k) et où $P + 2^i < N$, [P, $Q + 2^i$) est une plage alignée avec le hachage d'arborescence.
  - P = 0 est le cas spécial où A = 2[lgN]*0

Réponses d’erreur

En cas d'erreur, l'API retourne l'une des exceptions suivantes :

Code	Description	HTTP Status Code	Type
`AccessDeniedException`	Renvoyé en cas de tentative d'accès à une ressource non autorisée par une politique Gestion des identités et des accès AWS (IAM) ou si un Compte AWS identifiant incorrect a été utilisé dans l'URI de la demande. Pour de plus amples informations, veuillez consulter Identity and Access Management pour Amazon Glacier.	`403 Forbidden`	Client
`BadRequest`	Retourné si la demande ne peut pas être traitée.	`400 Bad Request`	Client
`ExpiredTokenException`	Retourné si le jeton de sécurité utilisé dans la demande a expiré.	`403 Forbidden`	Client

Code	Description	HTTP Status Code	Type
`InsufficientCapacityException`	Renvoyé si la capacité est insuffisante pour traiter la demande rapide. Cette erreur s'applique uniquement aux récupérations rapides et non aux récupérations standard ou en bloc.	`503 Service Unavailable`	Serveur
`InvalidParameterValueException`	Retourné si un paramètre de la demande n'a pas été spécifié correctement.	`400 Bad Request`	Client
`InvalidSignatureException`	Retourné si la signature de la demande n'est pas valide.	`403 Forbidden`	Client
`LimitExceededException`	Renvoyé si la demande dépasse l'une des limites suivantes : une limite de coffre, une limite de balise ou une limite de capacité allouée.	`400 Bad Request`	Client
`MissingAuthenticationTokenException`	Retourné si les données d'authentification ne sont pas trouvées pour la demande.	`400 Bad Request`	Client
`MissingParameterValueException`	Retourné si un en-tête ou un paramètre obligatoire n'apparaît pas dans la demande.	`400 Bad Request`	Client

Code	Description	HTTP Status Code	Type
`PolicyEnforcedException`	Retourné si une tâche d'extraction va dépasser la limite actuelle définie pour le taux d'extraction de la stratégie de données. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier.	`400 Bad Request`	Client
`ResourceNotFoundException`	Retourné si la ressource spécifiée, par exemple un coffre, un ID de chargement ou un ID de tâche n'existe pas.	`404 Not Found`	Client
`RequestTimeoutException`	Renvoyé en cas de téléchargement d'une archive et si le délai d'attente d'Amazon Glacier (Amazon Glacier) expire pendant la réception du téléchargement.	`408 Request Timeout`	Client
`SerializationException`	Retourné si le corps de la demande n'est pas valide. Si une charge utile JSON est incluse, vérifiez qu'elle est bien formée.	`400 Bad Request`	Client
`ServiceUnavailableException`	Retourné si le service ne peut pas terminer la demande.	`500 Internal Server Error`	Serveur
`ThrottlingException`	Renvoyé si vous devez réduire le taux de demandes adressées à Amazon Glacier.	`400 Bad Request`	Client

Code	Description	HTTP Status Code	Type
`UnrecognizedClientException`	Retourné si l'ID de clé d'accès ou le jeton de sécurité n'est pas valide.	`400 Bad Request`	Client

Plusieurs Amazon Glacier APIs renvoient la même exception, mais avec des messages d'exception différents pour vous aider à résoudre l'erreur spécifique rencontrée.

Amazon Glacier renvoie les informations d'erreur dans le corps de la réponse. Les exemples suivants illustrent quelques réponses envoyées en cas d'erreur.

Exemple 1 : Description d'une demande de tâche avec un ID de tâche qui n'existe pas

Supposons que vous envoyiez une demande Description de la tâche (GET JobID) correspondant à une tâche qui n'existe pas. Cela signifie que vous spécifiez un ID de tâche qui n'existe pas.

```
GET /-/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVEXAMPLEbadJobID HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

En réponse, Amazon Glacier renvoie la réponse d'erreur suivante.

```
HTTP/1.1 404 Not Found
x-amzn-RequestId: AAABaZ9N92Iiyv4N7sru3ABEpSQkuFtmH3NP6aAC51ixfjg
Content-Type: application/json
Content-Length: 185
Date: Wed, 10 Feb 2017 12:00:00 GMT
{
  "code": "ResourceNotFoundException",
  "message": "The job ID was not found: HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVEXAMPLEbadJobID",
```

```
  "type": "Client"
  }
```

Où :

Code

L'une des exceptions générales.

Type : chaîne

Message

Description générique de la condition d'erreur propre à l'API qui retourne l'erreur.

Type : chaîne

Type

Source de l'erreur. Ce champ peut comporter l'une des valeurs suivantes : `Client`, `Server` ou `Unknown`.

Type : chaîne.

Notez le point suivant dans la réponse précédente :

- Pour la réponse à l'erreur, Amazon Glacier renvoie les valeurs du code d'état de `4xx` et`5xx`. Dans cet exemple, le code de statut est `404 Not Found`.
- La valeur d'en-tête `Content-Type` `application/json` indique JSON dans le corps
- Le JSON dans le corps fournit les informations relatives à l'erreur.

Dans la demande précédente, au lieu d'un ID de tâche erroné, supposons que vous ayez spécifié un coffre qui n'existe pas. La réponse retourne un message différent.

```
HTTP/1.1 404 Not Found
x-amzn-RequestId: AAABBeC9Zw0rp_5D0L8VfB3FA_WlTupqTKAUehMcPhdgni0
Content-Type: application/json
Content-Length: 154
Date: Wed, 10 Feb 2017 12:00:00 GMT
{
  "code": "ResourceNotFoundException",
```

```
  "message": "Vault not found for ARN: arn:aws:glacier:us-west-2:012345678901:vaults/
examplevault",
  "type": "Client"
}
```

Exemple 2 : Demande d'établissement d'une liste des tâches avec une valeur non valide pour le paramètre de demande

Dans cet exemple, vous envoyez une demande Affichage des tâches (GET jobs) pour extraire les tâches de coffre avec un `statuscode` spécifique et vous fournissez une valeur `statuscode` `finished` incorrecte au lieu des valeurs acceptables `InProgress`, `Succeeded` ou`Failed`.

```
GET /-/vaults/examplevault/jobs?statuscode=finished HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Amazon Glacier renvoie le `InvalidParameterValueException` avec le message approprié.

```
HTTP/1.1 400 Bad Request
x-amzn-RequestId: AAABaZ9N92Iiyv4N7sru3ABEpSQkuFtmH3NP6aAC51ixfjg
Content-Type: application/json
Content-Length: 141
Date: Wed, 10 Feb 2017 12:00:00 GMT
{
  "code": "InvalidParameterValueException",
  "message": "The job status code is not valid: finished",
  "type: "Client"
}
```

Opérations de coffre

Les opérations de coffre-fort disponibles dans Amazon Glacier sont les suivantes.

Rubriques

- Interruption du verrouillage du coffre (DELETE lock-policy)

- Ajout de balises à un coffre (POST tags add)
- Création de coffre (PUT vault)
- Achèvement du verrouillage du coffre (POST lockId)
- Suppression de coffre (DELETE vault)
- Suppression de la stratégie d'accès au coffre (DELETE access-policy)
- Supprimer les notifications de coffre (DELETE notification-configuration)
- Description de coffre (GET vault)
- Obtention de la stratégie d'accès au coffre (GET access-policy)
- Obtenir le verrouillage de coffre (GET lock-policy)
- Obtenir les notifications de coffre (GET notification-configuration)
- Lancement de verrouillage de coffre (POST lock-policy)
- Affichage des balises d'un coffre (GET tags)
- Affichage des coffres (GET vaults)
- Suppression des balises d'un coffre (POST tags remove)
- Définir la stratégie d'accès au coffre (PUT access-policy)
- Définition d'une configuration de notification de coffre (PUT notification-configuration)

Interruption du verrouillage du coffre (DELETE lock-policy)

Description

Cette opération arrête le processus de verrouillage de coffre si l'état du verrouillage de coffre n'est pas `Locked`. Si l'état du verrouillage du coffre est `Locked` lorsque cette opération est demandée, l'opération renvoie une erreur `AccessDeniedException`. Si le processus de verrouillage de coffre est arrêté, la politique de verrouillage de coffre est supprimée du coffre spécifié.

Le verrouillage du coffre est passé à l'état `InProgress` en appelant Lancement de verrouillage de coffre (POST lock-policy). Le verrouillage du coffre est passé à l'état `Locked` en appelant Achèvement du verrouillage du coffre (POST lockId). Vous pouvez obtenir l'état d'un verrouillage de coffre en appelant Obtenir le verrouillage de coffre (GET lock-policy). Pour plus d'informations sur le processus de verrouillage de coffre, consultez la section Verrou Amazon Glacier Vault. Pour plus d'informations sur les stratégies de verrouillage de coffre, consultez la page Stratégies de verrouillage de coffre.

Cette opération est idempotente. Vous pouvez appeler avec succès cette opération plusieurs fois si le verrouillage du coffre a l'état `InProgress` ou si aucune stratégie n'est associée au coffre.

Requêtes

Pour supprimer la stratégie de verrouillage du coffre, envoyez une demande HTTP `DELETE` à l'URI de la sous-ressource `lock-policy` du coffre.

Syntaxe

```
DELETE /AccountId/vaults/vaultName/lock-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Si la politique est correctement supprimée, Amazon Glacier renvoie une `HTTP 204 No Content` réponse.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant montre comment arrêter le processus de verrouillage de coffre.

Exemple de requête

Dans le cadre de cet exemple, une demande `DELETE` est envoyée à la sous-ressource `lock-policy` du coffre nommé **`examplevault`**.

```
DELETE /-/vaults/examplevault/lock-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

```
x-amz-glacier-version: 2012-06-01
```

Exemple de réponse

Si la politique est correctement supprimée, Amazon Glacier renvoie une `HTTP 204 No Content` réponse, comme illustré dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Achèvement du verrouillage du coffre (POST lockId)
- Obtenir le verrouillage de coffre (GET lock-policy)
- Lancement de verrouillage de coffre (POST lock-policy)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Ajout de balises à un coffre (POST tags add)

Cette opération ajoute les balises spécifiées à un coffre. Chaque balise est constituée d'une clé et d'une valeur. Chaque coffre peut comporter jusqu'à 50 balises. Si votre demande entraîne le dépassement de la limite de balise pour le coffre, l'erreur `LimitExceededException` est générée par l'opération.

Si une balise existe déjà sur le coffre sous une clé spécifiée, la valeur de clé existante est remplacée. Pour en savoir plus sur les identifications, consultez Balisage des ressources Amazon Glacier.

Syntaxe de la requête

Pour ajouter des balises à un coffre, envoyez une demande HTTP POST à l'URI des balises, comme illustré dans l'exemple de syntaxe suivant.

```
POST /AccountId/vaults/vaultName/tags?operation=add HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01

{
   "Tags":
      {
         "string": "string",
         "string": "string"
      }
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Nom	Description	Obligatoire
`operation=add`	Paramètre de chaîne de requête unique `operation` doté de la valeur `add` pour le distinguer de l'opération Suppression des balises d'un coffre (POST tags remove).	Oui

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande contient les champs JSON suivants.

Balises

Balises à ajouter au coffre. Chaque balise est constituée d'une clé et d'une valeur. La valeur peut être une chaîne vide.

Type : mappage chaîne/chaîne

Contraintes de longueur : longueur minimale fixée à 1. Longueur maximale fixée à 10.

Obligatoire : oui

Réponses

Si la demande d'opération aboutit, le service renvoie une réponse HTTP `204 No Content`.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP POST avec les balises à ajouter au coffre.

```
POST /-/vaults/examplevault/tags?operation=add HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: length
x-amz-glacier-version: 2012-06-01

{
  "Tags":
    {
       "examplekey1": "examplevalue1",
       "examplekey2": "examplevalue2"
    }
}
```

Exemple de réponse

Si la demande est acceptée, Amazon Glacier renvoie unHTTP 204 No Content, comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Affichage des balises d'un coffre (GET tags)

- Suppression des balises d'un coffre (POST tags remove)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Création de coffre (PUT vault)

Description

Cette opération crée un nouveau coffre avec le nom spécifié. Le nom du coffre doit être unique au sein d'une AWS région pour un Compte AWS. Vous pouvez créer jusqu'à 1 000 coffres par compte. Pour plus d'informations sur la création d'autres coffres-forts, rendez-vous sur la page détaillée du produit Amazon Glacier.

Vous devez utiliser les instructions suivantes pour nommer un coffre.

- Les noms peuvent comporter entre 1 et 255 caractères.
- Les caractères autorisés sont a-z, A-Z, 0-9, « _ » (trait de soulignement), « - » (trait d'union) et « . » (point).

Cette opération est idempotente, vous pouvez envoyer la même demande plusieurs fois et elle n'a aucun autre effet après la première création par Amazon Glacier (Amazon Glacier) du coffre spécifié.

Requêtes

Syntaxe

Pour créer un coffre, envoyez une demande HTTP PUT à l'URI du coffre à créer.

```
PUT /AccountId/vaults/VaultName HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Content-Length: Length
```

```
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande pour cette opération doit être vide (0 octet).

Réponses

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`Location`	Chemin d'accès par URI relatif du coffre qui a été créé. Type : String

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP PUT pour créer un coffre nommé `examplevault`.

```
PUT /-/vaults/examplevault HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Content-Length: 0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Amazon Glacier crée le coffre et renvoie le chemin URI relatif du coffre dans l'`Location`en-tête. L'ID de compte est toujours affiché dans l'en-tête `Location`, que la demande spécifie un ID de compte ou un trait d'union ('-').

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Location: /111122223333/vaults/examplevault
```

Sections connexes

- Affichage des coffres (GET vaults)
- Suppression de coffre (DELETE vault)
- Identity and Access Management pour Amazon Glacier

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Achèvement du verrouillage du coffre (POST lockId)

Description

Cette opération termine le processus de verrouillage du coffre en passant le verrouillage du coffre de l'état `InProgress` à l'état `Locked`, ce qui provoque l'immuabilité de la stratégie de verrouillage du coffre. Le verrouillage du coffre est passé à l'état `InProgress` en appelant Lancement de verrouillage de coffre (POST lock-policy). Vous pouvez obtenir l'état du verrouillage du coffre en appelant Obtenir le verrouillage de coffre (GET lock-policy). Pour plus d'informations sur le processus de verrouillage de coffre, consultez la section Verrou Amazon Glacier Vault.

Cette opération est idempotente. Cette demande aboutit toujours si l'état du verrouillage du coffre est `Locked` et que l'ID de verrouillage fourni correspond à l'ID de verrouillage utilisé initialement pour verrouiller le coffre.

Si un ID de verrouillage non valide est passé dans la demande lorsque le verrouillage du coffre a l'état `Locked`, l'opération renvoie une erreur `AccessDeniedException`. Si un ID de verrouillage non valide est passé dans la demande lorsque l'état du verrouillage du coffre est `InProgress`, l'opération génère une erreur `InvalidParameter`.

Requêtes

Pour terminer le processus de verrouillage du coffre, envoyez une demande HTTP `POST` à l'URI de la sous-ressource `lock-policy` du coffre avec un ID de verrouillage valide.

Syntaxe

```
POST /AccountId/vaults/vaultName/lock-policy/lockId HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « `-` » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

La valeur `lockId` est l'ID de verrouillage obtenu d'une demande Lancement de verrouillage de coffre (POST lock-policy).

Paramètres de demande

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Si la demande d'opération aboutit, le service renvoie une réponse HTTP `204 No Content`.

Syntaxe

```
HTTP/1.1 204 No Content
```

```
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d'erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP POST avec l'ID de verrouillage pour terminer le processus de verrouillage du coffre.

```
POST /-/vaults/examplevault/lock-policy/AE863rKkWZU53SLW5be4DUcW HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: length
x-amz-glacier-version: 2012-06-01
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier (Amazon Glacier) renvoie une `HTTP 204 No Content` réponse, comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
```

```
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Interruption du verrouillage du coffre (DELETE lock-policy)
- Obtenir le verrouillage de coffre (GET lock-policy)
- Lancement de verrouillage de coffre (POST lock-policy)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Suppression de coffre (DELETE vault)

Description

Cette opération supprime un coffre. Amazon Glacier (Amazon Glacier) supprimera un coffre-fort uniquement s'il ne contient aucune archive selon le dernier inventaire et s'il n'y a eu aucune écriture dans le coffre-fort depuis le dernier inventaire. Si l'une de ces conditions n'est pas remplie, la suppression du coffre échoue (c'est-à-dire que le coffre n'est pas supprimé) et Amazon Glacier renvoie un message d'erreur.

Vous pouvez utiliser l'Description de coffre (GET vault)opération qui fournit des informations sur le coffre-fort, notamment le nombre d'archives qu'il contient ; toutefois, les informations sont basées sur le dernier inventaire du coffre-fort généré par Amazon Glacier.

Cette opération est idempotente.

> **Note**
>
> Lorsque vous supprimez un coffre, la stratégie d'accès au coffre qui lui est attachée est également supprimée. Pour plus d'informations sur les stratégies d'accès au coffre, consultez la section Politiques d'accès au coffre-fort.

Requêtes

Pour supprimer un coffre, envoyez une demande `DELETE` à l'URI de ressource de coffre.

Syntaxe

```
DELETE /AccountId/vaults/VaultName HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant supprime un coffre nommé `examplevault`. L'exemple de demande est une demande `DELETE` envoyée à l'URI de la ressource (le coffre) à supprimer.

```
DELETE /-/vaults/examplevault HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Création de coffre (PUT vault)
- Affichage des coffres (GET vaults)
- Lancement d'une tâche (POST jobs)
- Identity and Access Management pour Amazon Glacier

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Suppression de la stratégie d'accès au coffre (DELETE access-policy)

Description

Cette opération supprime la stratégie d'accès associée au coffre spécifié. L'opération est finalement cohérente, c'est-à-dire qu'Amazon Glacier (Amazon Glacier) peut mettre un certain temps à supprimer complètement la politique d'accès, et vous pouvez toujours constater l'effet de cette politique pendant une courte période après avoir envoyé la demande de suppression.

Cette opération est idempotente. Vous pouvez appeler une suppression plusieurs fois même si aucune stratégie n'est associée au coffre. Pour plus d'informations sur les stratégies d'accès au coffre, consultez la section Politiques d'accès au coffre-fort.

Requêtes

Pour supprimer la stratégie d'accès au coffre actuelle, envoyez une demande HTTP `DELETE` à l'URI de la sous-ressource `access-policy` du coffre.

Syntaxe

```
DELETE /AccountId/vaults/vaultName/access-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

En réponse, Amazon Glacier renvoie `204 No Content` si la stratégie est supprimée avec succès.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
```

```
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant montre comment supprimer une stratégie d'accès au coffre.

Exemple de requête

Dans le cadre de cet exemple, une demande `DELETE` est envoyée à la sous-ressource `access-policy` du coffre nommé **`examplevault`**.

```
DELETE /-/vaults/examplevault/access-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
x-amz-glacier-version: 2012-06-01
```

Exemple de réponse

En réponse, si la politique est correctement supprimée, Amazon Glacier renvoie un`204 No Content`, comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
```

```
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Obtention de la stratégie d'accès au coffre (GET access-policy)
- Définir la stratégie d'accès au coffre (PUT access-policy)

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Supprimer les notifications de coffre (DELETE notification-configuration)

Description

Cette opération supprime la configuration de notification définie pour un coffre Définition d'une configuration de notification de coffre (PUT notification-configuration). L'opération est finalement cohérente, c'est-à-dire qu'Amazon Glacier (Amazon Glacier) peut mettre un certain temps à désactiver complètement les notifications, et il se peut que vous continuiez à recevoir certaines notifications pendant une courte période après avoir envoyé la demande de suppression.

Requêtes

Pour supprimer la configuration d'une notification de coffre, envoyez une demande `DELETE`à la sous-ressource `notification-configuration` du coffre.

Syntaxe

```
DELETE /AccountId/vaults/VaultName/notification-configuration HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant illustre la suppression de la configuration de la notification pour un coffre.

Exemple de requête

Dans le cadre de cet exemple, une demande `DELETE` est envoyée à la sous-ressource `notification-configuration` du coffre nommé `examplevault`.

```
DELETE /111122223333/vaults/examplevault/notification-configuration HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Obtenir les notifications de coffre (GET notification-configuration)
- Définition d'une configuration de notification de coffre (PUT notification-configuration)
- Identity and Access Management pour Amazon Glacier

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Description de coffre (GET vault)

Description

Cette opération renvoie des informations sur un coffre, y compris son Amazon Resource Name (ARN), sa date de création, le nombre d'archives qu'il contient et la taille totale de toutes ces archives. Le nombre d'archives et leur taille totale sont ceux du dernier inventaire de coffres-forts généré par Amazon Glacier (Amazon Glacier) (voirUtilisation des coffres-forts dans Amazon Glacier). Amazon Glacier génère des inventaires de coffres-forts environ tous les jours. Cela signifie que si vous ajoutez ou retirez une archive dans un coffre et que vous envoyez immédiatement une demande de description de coffre, la réponse peut ne pas refléter les modifications.

Requêtes

Pour obtenir des informations sur un coffre, envoyez une demande `GET` à l'URI de la ressource de coffre spécifique.

Syntaxe

```
GET /AccountId/vaults/VaultName HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length

{
  "CreationDate" : String,
  "LastInventoryDate" : String,
  "NumberOfArchives" : Number,
  "SizeInBytes" : Number,
  "VaultARN" : String,
  "VaultName" : String
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

CreationDate

Date UTC à laquelle le coffre a été créé.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

LastInventoryDate

Date UTC à laquelle Amazon Glacier a terminé le dernier inventaire du coffre. Pour plus d'informations sur le lancement d'un inventaire pour un coffre, consultez la section Lancement d'une tâche (POST jobs).

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

NumberOfArchives

Nombre d'archives présentes dans le coffre en fonction du dernier inventaire du coffre. Ce champ renvoie une valeur null si un inventaire n'a pas encore été exécuté sur le coffre, par exemple, si celui-ci vient d'être créé.

Type : nombre

SizeInBytes

Taille totale, exprimée en octets, des archives présentes dans le coffre, y compris l'éventuelle surcharge de chaque archive, à la date du dernier inventaire. Ce champ renvoie une valeur `null` si un inventaire n'a pas encore été exécuté sur le coffre, par exemple, si celui-ci vient d'être créé.

Type : nombre

VaultARN

Amazon Resource Name (ARN) du coffre.

Type : chaîne

VaultName

Nom de coffre qui a été spécifié au moment de la création. Le nom de coffre est également inclus dans l'ARN du coffre.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant montre comment obtenir des informations concernant le coffre nommé `examplevault`.

```
GET /-/vaults/examplevault HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Content-Type: application/json
Content-Length: 260

{
  "CreationDate" : "2012-02-20T17:01:45.198Z",
  "LastInventoryDate" : "2012-03-20T17:03:43.221Z",
  "NumberOfArchives" : 192,
  "SizeInBytes" : 78088912,
  "VaultARN" : "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault",
  "VaultName" : "examplevault"
}
```

Sections connexes

- Création de coffre (PUT vault)
- Affichage des coffres (GET vaults)
- Suppression de coffre (DELETE vault)
- Lancement d'une tâche (POST jobs)
- Identity and Access Management pour Amazon Glacier

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Obtention de la stratégie d'accès au coffre (GET access-policy)

Description

Cette opération extrait la sous-ressource `access-policy` définie au niveau du coffre (pour plus d'informations sur la définition de cette sous-ressource, consultez Définir la stratégie d'accès au coffre (PUT access-policy)). Si aucune stratégie d'accès n'est définie sur le coffre, l'opération renvoie une erreur `404 Not found`. Pour plus d'informations sur les stratégies d'accès au coffre, consultez la section Politiques d'accès au coffre-fort.

Requêtes

Pour renvoyer la stratégie d'accès au coffre actuelle, envoyez une demande HTTP `GET` à l'URI de la sous-ressource `access-policy` du coffre.

Syntaxe

```
GET /AccountId/vaults/vaultName/access-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

En réponse, Amazon Glacier (Amazon Glacier) renvoie la politique d'accès au coffre-fort au format JSON dans le corps de la réponse.

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length

{
  "Policy": "string"
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

Stratégie

Stratégie d'accès aux coffres sous forme de chaîne JSON, qui utilise « \ » comme caractère d'échappement.

Type : Chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d'erreur.

Exemples

L'exemple suivant montre comment obtenir une stratégie d'accès au coffre.

Exemple de requête

Dans cet exemple, une demande `GET` est envoyée à l'URI de la sous-ressource `access-policy` d'un coffre.

```
GET /-/vaults/examplevault/access-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier renvoie la politique d'accès au coffre-fort sous forme de chaîne JSON dans le corps de la réponse. La chaîne JSON renvoyée utilise « \ » comme caractère d'échappement, comme illustré dans les exemples Définir la stratégie d'accès au coffre (PUT access-policy). Toutefois, l'exemple suivant montre la chaîne JSON retournée sans caractères d'échappement pour faciliter la lecture.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
```

```
Content-Length: length

{
  "Policy": "
    {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Sid": "allow-time-based-deletes",
          "Principal": {
            "AWS": "999999999999"
          },
          "Effect": "Allow",
          "Action": "glacier:Delete*",
          "Resource": [
            "arn:aws:glacier:us-west-2:999999999999:vaults/examplevault"
          ],
          "Condition": {
            "DateGreaterThan": {
              "aws:CurrentTime": "2018-12-31T00:00:00Z"
            }
          }
        }
      ]
    }
  "
}
```

Sections connexes

- Suppression de la stratégie d'accès au coffre (DELETE access-policy)
- Définir la stratégie d'accès au coffre (PUT access-policy)

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Obtenir le verrouillage de coffre (GET lock-policy)

Description

Cette opération extrait les attributs suivants à partir de la sous-ressource `lock-policy` définie sur le coffre spécifié :

- La stratégie de verrouillage de coffre définie sur le coffre.
- L'état du verrouillage du coffre, qui est soit `InProgess` soit `Locked`.
- Moment où l'ID de verrouillage arrive à expiration. L'ID de verrouillage est utilisé pour terminer le processus de verrouillage du coffre.
- Le moment où le verrouillage du coffre a été lancé et s'est vu attribuer l'état `InProgress`.

Le verrouillage du coffre est passé à l'état `InProgress` en appelant Lancement de verrouillage de coffre (POST lock-policy). Le verrouillage du coffre est passé à l'état `Locked` en appelant Achèvement du verrouillage du coffre (POST lockId). Vous pouvez arrêter le processus de verrouillage de coffre en appelant Interruption du verrouillage du coffre (DELETE lock-policy). Pour plus d'informations sur le processus de verrouillage de coffre, consultez la section Verrou Amazon Glacier Vault.

Si aucune stratégie de verrouillage de coffre n'est définie sur le coffre, l'opération renvoie une erreur `404 Not found`. Pour plus d'informations sur les stratégies de verrouillage de coffre, consultez la page Stratégies de verrouillage de coffre.

Requêtes

Pour renvoyer la stratégie de verrouillage de coffre actuelle et d'autres attributs, envoyez une demande `GET` HTTP à l'URI de la sous-ressource `lock-policy` du coffre, comme illustré dans l'exemple de syntaxe suivant.

Syntaxe

```
GET /AccountId/vaults/vaultName/lock-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

En réponse, Amazon Glacier (Amazon Glacier) renvoie la politique d'accès au coffre-fort au format JSON dans le corps de la réponse.

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length

{
  "Policy": "string",
  "State": "string",
  "ExpirationDate": "string",
```

```
  "CreationDate":"string"
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

Stratégie

Stratégie de verrouillage de coffre sous la forme une chaîne JSON, qui utilise « \ » comme caractère d'échappement.

Type : Chaîne

État

État du verrouillage du coffre.

Type : Chaîne

Valeurs valides : `InProgress|Locked`

ExpirationDate

Date et heure UTC auxquelles l'ID de verrouillage expire. Cette valeur peut être `null`si le verrouillage de coffre se trouve dans un état `Locked`.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

CreationDate

Date et heure UTC auxquelles le verrouillage de coffre est passé à l'état `InProgresst`.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant montre comment obtenir une stratégie de verrouillage de coffre.

Exemple de requête

Dans cet exemple, une demande GET est envoyée à l'URI de la sous-ressource `lock-policy` d'un coffre.

```
GET /-/vaults/examplevault/lock-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier renvoie la politique d'accès au coffre-fort sous forme de chaîne JSON dans le corps de la réponse. La chaîne JSON renvoyée utilise « \ » comme caractère d'échappement, comme illustré dans l'exemple de demande Lancement de verrouillage de coffre (POST lock-policy). Toutefois, l'exemple suivant montre la chaîne JSON retournée sans caractères d'échappement pour faciliter la lecture.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: length

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

```
        {
          "Sid": "Define-vault-lock",
          "Principal": {
            "AWS": "arn:aws:iam::999999999999:root"
          },
          "Effect": "Deny",
          "Action": "glacier:DeleteArchive",
          "Resource": [
            "arn:aws:glacier:us-west-2:999999999999:vaults/examplevault"
          ],
          "Condition": {
            "NumericLessThanEquals": {
              "glacier:ArchiveAgeInDays": "365"
            }
          }
        }
      ]
    }
  ",
  "State": "InProgress",
  "ExpirationDate": "exampledate",
  "CreationDate": "exampledate"
}
```

Sections connexes

- Interruption du verrouillage du coffre (DELETE lock-policy)
- Achèvement du verrouillage du coffre (POST lockId)
- Lancement de verrouillage de coffre (POST lock-policy)

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Obtenir les notifications de coffre (GET notification-configuration)

Description

Cette opération extrait la sous-ressource `notification-configuration` définie sur le coffre (voir la section Définition d'une configuration de notification de coffre (PUT notification-configuration). Si la configuration des notifications pour un coffre n'est pas définie, l'opération retourne une erreur `404 Not Found`. Pour plus d'informations sur les notifications de coffre, consultez la page Configuration des notifications de coffre-fort dans Amazon Glacier.

Requêtes

Pour extraire les informations de configuration des notifications, envoyez une demande `GET` à l'URI de la sous-ressource `notification-configuration` d'un coffre.

Syntaxe

```
GET /AccountId/vaults/VaultName/notification-configuration HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length
{
  "Events": [
    String,
    ...
  ],
  "SNSTopic": String
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

Événements

Liste d'un ou de plusieurs événements pour lesquels Amazon Glacier (Amazon Glacier) enverra une notification à la rubrique Amazon SNS spécifiée. Pour plus d'informations sur les événements

de coffre pour lesquels vous pouvez configurer un coffre afin de publier des notifications, consultez la section Définition d'une configuration de notification de coffre (PUT notification-configuration).

Type : tableau

SNSTopic

Amazon Resource Name (ARN) de la rubrique Amazon Simple Notification Service (Amazon SNS). Pour plus d'informations, consultez Démarrage avec Amazon SNS dans le Guide de démarrage Amazon Simple Notification Service.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant illustre l'extraction de la configuration de notification pour un coffre.

Exemple de requête

Dans cet exemple, une demande `GET` est envoyée à la sous-ressource `notification-configuration` d'un coffre.

```
GET /-/vaults/examplevault/notification-configuration HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Une réponse positive présente le document de configuration de la journalisation d'audit dans le corps de la réponse au format JSON. Dans cet exemple, la configuration montre que les notifications pour

deux événements (`ArchiveRetrievalCompleted` et `InventoryRetrievalCompleted`) sont envoyées à la rubrique Amazon SNS `arn:aws:sns:us-west-2:012345678901:mytopic`.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 150

{
  "Events": [
    "ArchiveRetrievalCompleted",
    "InventoryRetrievalCompleted"
  ],
  "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic"
}
```

Sections connexes

- Supprimer les notifications de coffre (DELETE notification-configuration)
- Définition d'une configuration de notification de coffre (PUT notification-configuration)
- Identity and Access Management pour Amazon Glacier

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Lancement de verrouillage de coffre (POST lock-policy)

Description

Cette opération lance le processus de verrouillage de coffre en procédant comme suit :

- Installation d'une stratégie de verrouillage de coffre sur le coffre spécifié.
- Affectation de l'état `InProgress` au verrouillage de coffre.

- Renvoi d'un ID de verrouillage qui est utilisé pour terminer le processus de verrouillage de coffre.

Vous pouvez définir une stratégie de verrouillage de coffre pour chaque coffre et la taille de cette stratégie peut atteindre 20 Ko. Pour plus d'informations sur les stratégies de verrouillage de coffre, consultez la page Stratégies de verrouillage de coffre.

Vous devez terminer le processus de verrouillage de coffre dans un délai de 24 heures après le passage à l'état `InProgress` du verrouillage de coffre. Au terme de la fenêtre de 24 heures, l'ID de verrouillage expire, le coffre quitte automatiquement l'état `InProgress` et la stratégie de verrouillage de coffre est retirée du coffre. Vous appelez Achèvement du verrouillage du coffre (POST lockId) pour terminer le processus de verrouillage de coffre en affectant l'état `Locked` au verrouillage de coffre.

> **Note**
>
> Lorsqu'un verrouillage de coffre est à l'état `Locked`, vous ne pouvez pas lancer un nouveau verrouillage de coffre pour le coffre.

Vous pouvez arrêter le processus de verrouillage de coffre en appelant Interruption du verrouillage du coffre (DELETE lock-policy). Vous pouvez obtenir l'état du verrouillage de coffre en appelant Obtenir le verrouillage de coffre (GET lock-policy). Pour plus d'informations sur le processus de verrouillage de coffre, consultez la section Verrou Amazon Glacier Vault.

Si cette opération est appelée alors que le verrouillage de coffre est à l'état `InProgress`, l'opération renvoie une erreur `AccessDeniedException`. Lorsque le verrouillage de coffre est à l'état `InProgress`, vous devez appeler Interruption du verrouillage du coffre (DELETE lock-policy) avant de pouvoir lancer une nouvelle stratégie de verrouillage de coffre.

Requêtes

Pour lancer le processus de verrouillage de coffre, envoyez une demande HTTP `POST` à l'URI de la sous-ressource `lock-policy` du coffre, comme illustré dans l'exemple de syntaxe suivant.

Syntaxe

```
POST /AccountId/vaults/vaultName/lock-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
```

```
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01

{
  "Policy": "string"
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande contient les champs JSON suivants.

Stratégie

Stratégie de verrouillage de coffre sous la forme une chaîne JSON, qui utilise « \ » comme caractère d'échappement.

Type : String

Obligatoire : oui

Réponses

Amazon Glacier (Amazon Glacier) renvoie une `HTTP 201 Created` réponse si la politique est acceptée.

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-lock-id: lockId
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`x-amz-lock-id`	ID de verrouillage qui est utilisé pour terminer le processus de verrouillage de coffre. Type : String

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP PUT à l'URI de la sous-ressource `lock-policy` du coffre. La chaîne JSON `Policy` utilise « \ » comme caractère d'échappement.

```
PUT /-/vaults/examplevault/lock-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: length
x-amz-glacier-version: 2012-06-01

{"Policy":"{\"Version\":\"2012-10-17\",       \"Statement\":[{\"Sid\":\"Define-vault-
lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"arn:aws:iam::999999999999:root
\"},\"Action\":\"glacier:DeleteArchive\",\"Resource\":\"arn:aws:glacier:us-
west-2:999999999999:vaults/examplevault\",\"Condition\":{\"NumericLessThanEquals\":
{\"glacier:ArchiveAgeinDays\":\"365\"}}}]}"}
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier renvoie une `HTTP 201 Created` réponse, comme illustré dans l'exemple suivant.

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
x-amz-lock-id: AE863rKkWZU53SLW5be4DUcW
```

Sections connexes

- Interruption du verrouillage du coffre (DELETE lock-policy)
- Achèvement du verrouillage du coffre (POST lockId)
- Obtenir le verrouillage de coffre (GET lock-policy)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Affichage des balises d'un coffre (GET tags)

Cette opération affiche la liste de toutes les balises attachées à un coffre. En l'absence de balises, elle renvoie un mappage vide. Pour en savoir plus sur les identifications, consultez Balisage des ressources Amazon Glacier.

Syntaxe de la requête

Pour afficher la liste des balises attachées à un coffre, envoyez une demande HTTP GET à l'URI des balises, comme illustré dans l'exemple de syntaxe suivant.

```
GET /AccountId/vaults/vaultName/tags HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Si l'opération aboutit, le service renvoie une réponse HTTP `200 OK`.

Syntaxe de la réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length
{
   "Tags":
      {
         "string" : "string",
         "string" : "string"
      }
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

Balises

Balises attachées au coffre. Chaque balise est constituée d'une clé et d'une valeur.

Type : mappage chaîne/chaîne

Obligatoire : oui

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d'erreur.

Exemples

Exemple : Affichage de la liste des balises attachées à un coffre

L'exemple suivant affiche la liste des balises attachées à un coffre.

Exemple de requête

Dans cet exemple, une demande GET est envoyée pour extraire une liste de balises du coffre spécifié.

```
GET /-/vaults/examplevault/tags HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier (Amazon Glacier) renvoie un `HTTP 200 OK` contenant une liste de balises pour le coffre-fort, comme indiqué dans l'exemple suivant.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Content-Type: application/json
Content-Length: length

{
   "Tags",
      {
         "examplekey1": "examplevalue1",
         "examplekey2": "examplevalue2"
      }
}
```

Sections connexes

- Ajout de balises à un coffre (POST tags add)

- Suppression des balises d'un coffre (POST tags remove)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Affichage des coffres (GET vaults)

Description

Cette opération affiche la liste de tous les coffres dont le compte de l'utilisateur appelant est propriétaire. La liste renvoyée dans la réponse est triée en fonction des valeurs ASCII des noms de coffre.

Par défaut, cette opération renvoie jusqu'à 10 éléments par demande. S'il existe d'autres coffres à afficher, le champ `marker` du corps de la réponse contient l'Amazon Resource Name (ARN) du coffre à partir duquel poursuivre l'affichage de la liste avec une nouvelle demande d'affichage des coffres ; sinon, la valeur du champ `marker` est `null`. Dans votre prochaine demande List Vaults, vous définissez le `marker` paramètre sur la valeur renvoyée par Amazon Glacier (Amazon Glacier) dans les réponses à votre précédente demande List Vaults. Vous pouvez également limiter le nombre de coffres renvoyés dans la réponse en spécifiant le paramètre `limit` dans la demande.

Requêtes

Pour obtenir une liste de coffres, vous envoyez une demande `GET` à la ressource vaults.

Syntaxe

```
GET /AccountId/vaults HTTP/1.1
```

```
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération utilise les paramètres de demande suivants.

Nom	Description	Obligatoire
`limit`	Le nombre maximum de coffres à renvoyer. La limite par défaut est de 10. Il peut arriver que le nombre de coffres renvoyés soit inférieur à la limite spécifiée, mais il ne sera jamais supérieur à cette valeur. Type : String Contraintes : la valeur d'entier minimale est 1. La valeur d'entier maximale est 10.	Non
`marker`	Chaîne opaque utilisée pour la pagination. `marker` spécifie l'ARN du coffre à partir duquel l'affichage de la liste de coffres doit débuter. (Le coffre spécifié pour le paramètre `marker` n'est pas inclus dans la liste renvoyée.) Obtenez la valeur du paramètre `marker` à partir d'une réponse à une demande	Non

Nom	Description	Obligatoire
	d'affichage des coffres précédente. Vous ne devez inclure la valeur du paramètre `marker` que si vous poursuivez la pagination de résultats commencée dans une demande d'affichage des coffres précédente. Si une valeur vide ("") est spécifiée pour le paramètre marker, la liste de coffres renvoyée débute à partir du premier coffre. Type : String Contraintes: Aucune	

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length

{
  "Marker": String
  "VaultList": [
   {
    "CreationDate": String,
    "LastInventoryDate": String,
    "NumberOfArchives": Number,
```

```
    "SizeInBytes": Number,
    "VaultARN": String,
    "VaultName": String
   },
   ...
  ]
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

CreationDate

Date à laquelle le coffre a été créé, exprimée en heure UTC (temps universel coordonné).

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

LastInventoryDate

Date du dernier inventaire du coffre, exprimée en heure UTC (temps universel coordonné). Ce champ peut avoir une valeur null si un inventaire n'a pas encore été exécuté sur le coffre, par exemple, si celui-ci vient d'être créé. Pour plus d'informations sur le lancement d'un inventaire pour un coffre, consultez la section Lancement d'une tâche (POST jobs).

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

Marker

Champ `vaultARN` représentant l'endroit à partir duquel poursuivre la pagination des résultats. Vous utilisez la valeur du paramètre `marker` dans une autre demande d'affichage des coffres pour obtenir davantage de coffres dans la liste. S'il n'y a pas d'autres coffres, cette valeur est `null`.

Type : chaîne

NumberOfArchives

Nombre d'archives présentes dans le coffre à la date du dernier inventaire.

Type : nombre

SizeInBytes

Taille totale, exprimée en octets, de toutes les archives présentes dans le coffre, y compris l'éventuelle surcharge de chaque archive, à la date du dernier inventaire.

Type : nombre

VaultARN

Amazon Resource Name (ARN) du coffre.

Type : chaîne

VaultList

Tableau d'objets, chacun d'eux fournissant une description d'un coffre.

Type : tableau

VaultName

Nom du coffre.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple : Affichage de la liste de tous les coffres

L'exemple suivant affiche la liste des coffres. Dans la mesure où les paramètres `marker` et `limit` ne sont pas spécifiés dans la demande, jusqu'à 10 coffres peuvent être renvoyés.

Exemple de requête

```
GET /-/vaults HTTP/1.1
```

```
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

La valeur du champ `Marker` est `null`, ce qui signifie qu'il n'y a pas d'autres coffres à afficher.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Content-Type: application/json
Content-Length: 497

{
  "Marker": null,
  "VaultList": [
   {
    "CreationDate": "2012-03-16T22:22:47.214Z",
    "LastInventoryDate": "2012-03-21T22:06:51.218Z",
    "NumberOfArchives": 2,
    "SizeInBytes": 12334,
    "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault1",
    "VaultName": "examplevault1"
   },
   {
    "CreationDate": "2012-03-19T22:06:51.218Z",
    "LastInventoryDate": "2012-03-21T22:06:51.218Z",
    "NumberOfArchives": 0,
    "SizeInBytes": 0,
    "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault2",
    "VaultName": "examplevault2"
   },
   {
    "CreationDate": "2012-03-19T22:06:51.218Z",
    "LastInventoryDate": "2012-03-25T12:14:31.121Z",
    "NumberOfArchives": 0,
    "SizeInBytes": 0,
    "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault3",
    "VaultName": "examplevault3"
   }
```

```
  ]
}
```

Exemple : Affichage d'une liste partielle de coffres

L'exemple suivant renvoie deux coffres à partir du coffre spécifié par le paramètre `marker`.

Exemple de requête

```
GET /-/vaults?limit=2&marker=arn:aws:glacier:us-west-2:012345678901:vaults/
examplevault1 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Deux coffres sont renvoyés dans la liste. Le champ `Marker` contient l'ARN de coffre à utiliser afin de poursuivre la pagination dans une autre demande d'affichage des coffres.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Content-Type: application/json
Content-Length: 497

{
  "Marker": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault3",
  "VaultList": [
   {
    "CreationDate": "2012-03-16T22:22:47.214Z",
    "LastInventoryDate": "2012-03-21T22:06:51.218Z",
    "NumberOfArchives": 2,
    "SizeInBytes": 12334,
    "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault1",
    "VaultName": "examplevault1"
   },
   {
    "CreationDate": "2012-03-19T22:06:51.218Z",
    "LastInventoryDate": "2012-03-21T22:06:51.218Z",
```

```
    "NumberOfArchives": 0,
    "SizeInBytes": 0,
    "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault2",
    "VaultName": "examplevault2"
   }
  ]
}
```

Sections connexes

- Création de coffre (PUT vault)
- Suppression de coffre (DELETE vault)
- Lancement d'une tâche (POST jobs)
- Identity and Access Management pour Amazon Glacier

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Suppression des balises d'un coffre (POST tags remove)

Cette opération supprime une ou plusieurs balises dans la série de balises attachées à un coffre. Pour en savoir plus sur les identifications, consultez Balisage des ressources Amazon Glacier.

Cette opération est idempotente. L'opération aboutira même s'il n'y a pas de balises attachées au coffre.

Syntaxe de la requête

Pour supprimer des balises d'un coffre, envoyez une demande HTTP POST à l'URI des balises, comme il est illustré dans l'exemple de syntaxe suivant.

```
POST /AccountId/vaults/vaultName/tags?operation=remove HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
```

```
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01
{
   "TagKeys": [
      "string",
      "string"
   ]
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Nom	Description	Obligatoire
`operation=remove`	Paramètre de chaîne de requête unique `operation` doté de la valeur `remove` pour le distinguer de l'opération Ajout de balises à un coffre (POST tags add).	Oui

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande contient les champs JSON suivants.

TagKeys

Une liste de clés de balise. Chaque balise correspondante est supprimée du coffre.

Type : tableau de chaînes

Contrainte de longueur : Minimum de 1 élément dans la liste. Maximum de 10 éléments dans la liste.

Obligatoire : oui

Réponses

Si l'action aboutit, le service renvoie une réponse HTTP `204 No Content` avec un corps HTTP vide.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP POST pour supprimer les balises spécifiées.

```
POST /-/vaults/examplevault/tags?operation=remove HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: length
x-amz-glacier-version: 2012-06-01

{
   "TagsKeys": [
      "examplekey1",
      "examplekey2"
   ]
}
```

Exemple de réponse

Si la demande est acceptée, Amazon Glacier (Amazon Glacier) renvoie un `HTTP 204 No Content` comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Ajout de balises à un coffre (POST tags add)
- Affichage des balises d'un coffre (GET tags)

consultez aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Définir la stratégie d'accès au coffre (PUT access-policy)

Description

Cette opération configure une stratégie d'accès à un coffre et remplace toute stratégie existante. Pour configurer une stratégie d'accès aux coffres, envoyez une demande PUT à la sous-ressource `access-policy` du coffre. Vous pouvez définir une stratégie d'accès par coffre et cette stratégie peut atteindre 20 Ko. Pour plus d'informations sur les stratégies d'accès au coffre, consultez la section Politiques d'accès au coffre-fort.

Requêtes

Syntaxe

Pour définir une stratégie d'accès aux coffres, envoyez une demande PUT HTTP à l'URI de la sous-ressource `access-policy` du coffre, comme illustré dans l'exemple de syntaxe suivant.

```
PUT /AccountId/vaults/vaultName/access-policy HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01

{
  "Policy": "string"
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande contient les champs JSON suivants.

Stratégie

Stratégie d'accès aux coffres sous forme de chaîne JSON, qui utilise « \ » comme caractère d'échappement.

Type : Chaîne

Obligatoire : oui

Réponses

En réponse, Amazon Glacier revient `204 No Content` si la politique est acceptée.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP PUT à l'URI de la sous-ressource `access-policy` du coffre. La chaîne JSON `Policy` utilise « \ » comme caractère d'échappement.

```
PUT /-/vaults/examplevault/access-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
Content-Length: length
x-amz-glacier-version: 2012-06-01

{"Policy":"{\"Version\":\"2012-10-17\",       \"Statement\":[{\"Sid\":
\"Define-owner-access-rights\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"arn:aws:iam::999999999999:root\"},\"Action\":\"glacier:DeleteArchive\",\"Resource\":
\"arn:aws:glacier:us-west-2:999999999999:vaults/examplevault\"}]}"}
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier (Amazon Glacier) renvoie un `HTTP 204 No Content` comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Suppression de la stratégie d'accès au coffre (DELETE access-policy)
- Obtention de la stratégie d'accès au coffre (GET access-policy)

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Définition d'une configuration de notification de coffre (PUT notification-configuration)

Description

La récupération d'une archive et d'un inventaire du coffre est une opération asynchrone dans Amazon Glacier (Amazon Glacier) pour laquelle vous devez d'abord lancer une tâche et attendre que la tâche soit terminée avant de pouvoir télécharger le résultat de la tâche. Vous pouvez configurer un coffre pour qu'il publie un message dans une rubrique Amazon Simple Notification Service (Amazon SNS) une fois que ces tâches sont terminées. Vous pouvez utiliser cette opération pour définir une configuration de notification sur le coffre. Pour de plus amples informations, veuillez consulter Configuration des notifications de coffre-fort dans Amazon Glacier.

Pour configurer des notifications de coffre, envoyez une demande PUT à la sous-ressource `notification-configuration` du coffre. Une configuration de notification est propre à un coffre ; par conséquent, elle est également appelée sous-ressource de coffre. La demande doit inclure un document JSON qui fournit une rubrique Amazon Simple Notification Service (Amazon SNS) et les événements pour lesquels vous souhaitez qu'Amazon Glacier envoie des notifications à cette rubrique.

Vous pouvez configurer un coffre afin qu'il publie une notification pour les événements de coffre suivants :

- **`ArchiveRetrievalCompleted`** : cet événement se produit lorsqu'une tâche de lancement d'extraction d'archive se termine (Lancement d'une tâche (POST jobs)). L'état de la tâche terminée peut être `Succeeded` ou `Failed`. La notification envoyée à la rubrique SNS est identique à la sortie renvoyée par une opération Description de la tâche (GET JobID).
- **`InventoryRetrievalCompleted`** : cet événement se produit lorsqu'une tâche de lancement d'extraction d'inventaire se termine (Lancement d'une tâche (POST jobs)). L'état de la tâche

terminée peut être `Succeeded` ou `Failed`. La notification envoyée à la rubrique SNS est identique à la sortie renvoyée par une opération Description de la tâche (GET JobID).

Les rubriques Amazon SNS doivent autoriser le coffre à publier des notifications dans la rubrique.

Requêtes

Pour définir une configuration de notification sur votre coffre, envoyez une demande PUT à l'URI de la sous-ressource `notification-configuration` du coffre. Vous spécifiez la configuration dans le corps de la demande. La configuration comprend le nom de la rubrique Amazon SNS et un tableau des événements qui déclenchent l'envoi d'une notification à chaque rubrique.

Syntaxe

```
PUT /AccountId/vaults/VaultName/notification-configuration HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01

{
   "SNSTopic": String,
   "Events":[String, ...]
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le code JSON contient les champs suivants dans le corps de la demande :

Événements

Ensemble d'un ou de plusieurs événements pour lesquels vous souhaitez qu'Amazon Glacier envoie une notification.

Valeurs valides : `ArchiveRetrievalCompleted` | `InventoryRetrievalCompleted`

Obligatoire : oui

Type : tableau

SNSTopic

ARN de la rubrique Amazon SNS. Pour plus d'informations, accédez à Démarrage avec Amazon SNS dans le Guide de démarrage Amazon Simple Notification Service.

Obligatoire : oui

Type : chaîne

Réponses

En réponse, Amazon Glacier (Amazon Glacier) revient `204 No Content` si la configuration des notifications est acceptée.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant montre comment configurer la notification de coffre.

Exemple de requête

La demande suivante définit la configuration des notifications de `examplevault` qui prévoit l'envoi de notifications pour deux événements (`ArchiveRetrievalCompleted` et `InventoryRetrievalCompleted`) à la rubrique Amazon SNS `arn:aws:sns:us-west-2:012345678901:mytopic`.

```
PUT /-/vaults/examplevault/notification-policy HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
   "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"],
   "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic"
}
```

Exemple de réponse

Une réponse positive renvoie le code de réponse `204 No Content`.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Obtenir les notifications de coffre (GET notification-configuration)
- Supprimer les notifications de coffre (DELETE notification-configuration)
- Identity and Access Management pour Amazon Glacier

Voir aussi

Pour plus d'informations sur l'utilisation de cette API dans l'un des langages Amazon spécifiques SDKs, consultez ce qui suit :

- AWS Command Line Interface

Opérations d'archivage

Les opérations d'archivage disponibles pour une utilisation dans Amazon Glacier sont les suivantes.

Rubriques

- Suppression d'une archive (DELETE archive)
- Chargement d'archive (POST archive)

Suppression d'une archive (DELETE archive)

Description

Cette opération supprime une archive d'un coffre. Vous pouvez supprimer une archive à la fois à partir d'un coffre. Pour supprimer l'archive, vous devez fournir son ID d'archive dans la demande de suppression. Vous pouvez obtenir l'ID de l'archive en téléchargeant l'inventaire du coffre qui contient l'archive. Pour plus d'informations sur le téléchargement de l'inventaire de coffre, consultez la page Téléchargement d'un inventaire de coffre-fort dans Amazon Glacier.

Une fois que vous avez supprimé une archive, vous pouvez toujours faire une demande de lancement d'une tâche afin d'extraire l'archive supprimée, mais la tâche d'extraction de l'archive échouera.

Les extractions d'archives en cours pour un ID d'archivage lorsque vous supprimez l'archive peuvent ou non selon les scénarios suivants :

- Si la tâche de récupération des archives prépare activement les données pour le téléchargement lorsqu'Amazon Glacier (Amazon Glacier) reçoit la demande de suppression de l'archive, l'opération de récupération des archives risque d'échouer.
- Si la tâche de récupération de l'archive a correctement préparé l'archive pour le téléchargement lorsqu'Amazon Glacier reçoit la demande de suppression de l'archive, vous pourrez télécharger le résultat.

Pour plus d'informations sur l'extraction de l'archive, consultez la section Téléchargement d'une archive dans Amazon Glacier.

Cette opération est idempotente. Une tentative de suppression d'une archive déjà supprimée ne génère aucune erreur.

Requêtes

Pour supprimer une archive, envoyez une demande `DELETE` à l'URI de la ressource d'archive.

Syntaxe

```
DELETE /AccountId/vaults/VaultName/archives/ArchiveID HTTP/1.1
Host: glacier.Region.amazonaws.com
x-amz-Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations

d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant montre comment supprimer une archive du coffre nommé `examplevault`.

Exemple de requête

L'ID de l'archive à supprimer est spécifié comme étant une sous-ressource de `archives`.

```
DELETE /-/vaults/examplevault/archives/NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier répond `204 No Content` pour indiquer que l'archive est supprimée.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)
- Chargement d'archive (POST archive)
- Identity and Access Management pour Amazon Glacier

Chargement d'archive (POST archive)

Description

Cette opération ajoute une archive à un coffre. Pour qu'un chargement aboutisse, vos données sont conservées durablement. En réponse, Amazon Glacier (Amazon Glacier) renvoie l'ID d'archive dans

l'`x-amz-archive-id`en-tête de la réponse. Vous devez enregistrer l'ID d'archive ainsi renvoyé afin de l'utiliser pour accéder ultérieurement à l'archive.

Vous devez fournir un hachage SHA256 arborescent des données que vous téléchargez. Pour plus d'informations sur le calcul d'un hachage SHA256 arborescent, consultezCalcul des totaux de contrôle.

> **Note**
>
> Le hachage de l' SHA256 arbre n'est requis que pour l'action Upload Archive (archive POST) lors de l'utilisation de l'API. Cela n'est pas obligatoire lors de l'utilisation du AWS CLI.

Lorsque vous chargez une archive, vous avez la possibilité de décrire l'archive dans les limites de 1 024 caractères ASCII imprimables. Amazon Glacier renvoie la description de l'archive lorsque vous récupérez l'archive ou que vous obtenez l'inventaire du coffre-fort. Amazon Glacier n'interprète en aucune façon la description. Une description d'archive ne doit pas nécessairement être unique. Vous ne pouvez pas utiliser la description pour extraire ou trier la liste d'archives.

À l'exception de la description facultative de l'archive, Amazon Glacier ne prend en charge aucune métadonnée supplémentaire pour les archives. L'ID d'archive est une séquence de caractères opaque qui ne permet de déduire aucune information concernant l'archive. Par conséquent, vous pouvez conserver les métadonnées sur les archives côté client. Pour de plus amples informations, veuillez consulter Utilisation des archives dans Amazon Glacier.

Les archives sont immuables. Une fois que vous avez chargé une archive, vous ne pouvez pas modifier son contenu ni sa description.

Requêtes

Pour charger une archive, utilisez la méthode HTTP `POST` et limitez la portée de la demande à la sous-ressource `archives` du coffre dans lequel vous souhaitez enregistrer l'archive. La demande doit inclure la taille de la charge utile de l'archive, la somme de contrôle (hachage de l'SHA256 arbre) et peut éventuellement inclure une description de l'archive.

Syntaxe

```
POST /AccountId/vaults/VaultName/archives
Host: glacier.Region.amazonaws.com
x-amz-glacier-version: 2012-06-01
```

```
Date: Date
Authorization: SignatureValue
x-amz-archive-description: Description
x-amz-sha256-tree-hash: SHA256 tree hash
x-amz-content-sha256: SHA256 linear hash
Content-Length: Length

<Request body.>
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette implémentation de l'opération n'utilise pas de paramètres de demande.

En-têtes de demande

Cette opération utilise les en-têtes de demande suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Nom	Description	Obligatoire
`Content-Length`	Taille de l'objet, exprimée en octets. Pour plus d'informations, rendez-vous sur http://www.w3. org/Protocols/rfc2616/rfc2616-sec14.html #sec14 .13. Type : nombre Par défaut : aucun	Oui

Nom	Description	Obligatoire
	Contraintes: Aucune	
`x-amz-archive-description`	Description facultative de l'archive que vous chargez. Il peut s'agir d'une description rédigée dans un style simple ou d'un identifiant que vous choisissez d'attribuer. La description ne doit pas nécessairement être unique d'une archive à l'autre. Lorsque vous procédez à l'extraction d'un inventaire de coffre (consultez la section Lancement d'une tâche (POST jobs)), l'inventaire inclut cette description pour chacune des archives renvoyées dans la réponse. Type : String Par défaut : aucun Contraintes : la description doit être inférieure ou égale à 1 024 caractères. Les caractères autorisés sont les caractères ASCII 7 bits sans code de contrôle, plus précisément les valeurs ASCII décimales 32-126 ou hexadécimales 0x20-0x7E.	Non
`x-amz-content-sha256`	La SHA256 somme de contrôle (un hachage linéaire) de la charge utile. Il ne s'agit pas de la valeur que vous spécifiez dans l'en-tête `x-amz-sha256-tree-hash` . Type : String Par défaut : aucun Contraintes: Aucune	Oui

Nom	Description	Obligatoire
`x-amz-sha256-tree-hash`	Somme de contrôle calculée par l'utilisateur, hachage de SHA256 l'arbre, de la charge utile. Pour plus d'informations sur le calcul du hachage de l' SHA256 arbre, consultez Calcul des totaux de contrôle. Si Amazon Glacier calcule une somme de contrôle différente de la charge utile, il rejettera la demande. Type : String Par défaut : aucun Contraintes: Aucune	Oui

Corps de la requête

Le corps de la demande contient les données à charger.

Réponses

En réponse, Amazon Glacier stocke l'archive de manière durable et renvoie un chemin URI vers l'ID de l'archive.

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-sha256-tree-hash: ChecksumComputedByAmazonGlacier
Location: Location
x-amz-archive-id: ArchiveId
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`Location`	Chemin d'accès par URI relatif de la ressource d'archive nouvellement ajoutée. Type : String
`x-amz-archive-id`	ID de l'archive. Cette valeur fait également partie de l'en-tête `Location`. Type : String
`x-amz-sha256-tree-hash`	Somme de contrôle de l'archive calculée par Amazon Glacier. Type : String

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant illustre une demande de chargement d'une archive.

```
POST /-/vaults/examplevault/archives HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-sha256-tree-hash:
 beb0fe31a1c7ca8c6c04d574ea906e3f97b31fdca7571defb5b44dca89b5af60
x-amz-content-sha256: 7f2fe580edb35154041fa3d4b41dd6d3adaef0c85d2ff6309f1d4b520eeecda3
Content-Length: 2097152
x-amz-glacier-version: 2012-06-01
Authorization: Authorization=AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-content-sha256;x-amz-date;x-
```

```
amz-glacier-
version,Signature=16b9a9e220a37e32f2e7be196b4ebb87120ca7974038210199ac5982e792cace

<Request body (2097152 bytes).>
```

Exemple de réponse

La réponse positive ci-dessous comporte un `Location` en-tête dans lequel vous pouvez obtenir l'identifiant attribué par Amazon Glacier à l'archive.

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
x-amz-sha256-tree-hash:
 beb0fe31a1c7ca8c6c04d574ea906e3f97b31fdca7571defb5b44dca89b5af60
Location: /111122223333/vaults/examplevault/archives/
NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
x-amz-archive-id: NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
```

Sections connexes

- Utilisation des archives dans Amazon Glacier
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Suppression d'une archive (DELETE archive)
- Identity and Access Management pour Amazon Glacier

Opérations de chargement partitionné

Les opérations de téléchargement partitionné disponibles dans Amazon Glacier sont les suivantes.

Rubriques

- Annulation de chargement partitionné (DELETE uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Lancement de chargement partitionné (POST multipart-uploads)
- Répertorier les parties (GET uploadID)

- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Partie chargement (PUT uploadID)

Annulation de chargement partitionné (DELETE uploadID)

Description

Cette commande arrête le chargement partitionné identifié par l'ID de chargement.

Une fois que la demande d'annulation de chargement partitionné a abouti, vous ne pouvez pas utiliser l'ID de chargement pour charger d'autres parties ou effectuer d'autres opérations. L'arrêt d'un chargement partitionné terminé échoue. En revanche, l'arrêt d'un chargement qui a déjà été arrêté aboutit pour un temps limité.

Cette opération est idempotente.

Pour plus d'informations sur le chargement partitionné, consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Requêtes

Pour arrêter un chargement partitionné, envoyez une demande HTTP `DELETE` à l'URI de la sous-ressource `multipart-uploads` du coffre et identifiez l'ID de chargement partitionné spécifique dans l'URI.

Syntaxe

```
DELETE /AccountId/vaults/VaultName/multipart-uploads/uploadID HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations

d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemple

Exemple de requête

Dans l'exemple suivant, une demande DELETE est envoyée à l'URI d'une ressource d'ID de chargement partitionné.

```
DELETE /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE  HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)
- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Répertorier les parties (GET uploadID)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Identity and Access Management pour Amazon Glacier

Achèvement du chargement partitionné (POST uploadID)

Description

Vous appelez cette opération de téléchargement en plusieurs parties pour informer Amazon Glacier (Amazon Glacier) que toutes les parties de l'archive ont été chargées et qu'Amazon Glacier peut désormais assembler l'archive à partir des parties chargées.

Pour plus d'informations sur le chargement partitionné, consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Après avoir assemblé et enregistré l'archive dans le coffre, Amazon Glacier renvoie l'ID d'archive de la ressource d'archive nouvellement créée. Une fois que vous avez chargé une archive, vous devez enregistrer l'ID de l'archive renvoyé pour extraire l'archive ultérieurement.

Dans la demande, vous devez inclure le hachage SHA256 arborescent calculé de l'ensemble de l'archive que vous avez chargée. Pour plus d'informations sur le calcul d'un hachage SHA256 arborescent, consultezCalcul des totaux de contrôle. Côté serveur, Amazon Glacier construit également le hachage SHA256 arborescent de l'archive assemblée. Si les valeurs correspondent, Amazon Glacier enregistre l'archive dans le coffre ; dans le cas contraire, une erreur est renvoyée et l'opération échoue. L'opération Répertorier les parties (GET uploadID) renvoie la liste des parties chargées pour un chargement partitionné spécifique. Cette liste comporte les informations de total de contrôle pour chaque partie chargée, lesquelles peuvent être utilisées pour déboguer un problème de total de contrôle incorrect.

En outre, Amazon Glacier vérifie également la présence de plages de contenu manquantes. Lors du chargement des parties, vous spécifiez les valeurs de plage qui identifient l'emplacement auquel se trouve chaque partie dans l'assemblage final de l'archive. Lors de l'assemblage de l'archive finale, Amazon Glacier vérifie la présence de plages de contenu manquantes. Si des plages de contenu sont manquantes, Amazon Glacier renvoie un message d'erreur et l'opération de téléchargement partitionné complet échoue.

L'opération d'achèvement du chargement partitionné est idempotente. Une fois que vous avez effectué correctement le chargement partitionné, si vous appelez à nouveau l'opération dans un court délai, celle-ci aboutit et renvoie le même ID d'archive. Cela est très utile si vous rencontrez un problème réseau ou si vous recevez une erreur serveur 500, auquel cas vous pouvez répéter votre demande d'achèvement du chargement partitionné et obtenir le même ID d'archive sans créer d'archives en double. Notez toutefois qu'une fois le chargement partitionné terminé, vous ne pouvez pas appeler l'opération d'affichage de la liste des parties et que le chargement partitionné

n'apparaîtra pas dans la réponse d'affichage de la liste des chargements partitionnés même si une opération idempotente est possible.

Requêtes

Pour terminer un téléchargement partitionné, vous envoyez une requête HTTP POST à l'URI de l'ID de téléchargement créé par Amazon Glacier en réponse à votre demande d'initiation de téléchargement partitionné. Il s'agit de l'URI que vous avez utilisé lors du chargement des parties. Outre les en-têtes obligatoires courants, vous devez inclure le résultat du hachage SHA256 arborescent de l'ensemble de l'archive et la taille totale de l'archive en octets.

Syntaxe

```
POST /AccountId/vaults/VaultName/multipart-uploads/uploadID
Host: glacier.Region.amazonaws.com
Date: date
Authorization: SignatureValue
x-amz-sha256-tree-hash: SHA256 tree hash of the archive
x-amz-archive-size: ArchiveSize in bytes
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise les en-têtes de demande suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Nom	Description	Obligatoire
`x-amz-archive-size`	Taille totale, en octets, de la totalité de l'archive. Cette valeur doit être la somme de toutes les tailles des différentes parties que vous avez chargées. Type : String Par défaut : aucun Contraintes: Aucune	Oui
`x-amz-sha256-tree-hash`	Le hachage SHA256 arborescent de l'ensemble de l'archive. Il s'agit du hachage SHA256 arborescent des différentes parties. Si la valeur que vous spécifiez dans la demande ne correspond pas au hachage SHA256 arborescent de l'archive assemblée finale tel que calculé par Amazon Glacier, Amazon Glacier renvoie une erreur et la demande échoue. Type : String Par défaut : aucun Contraintes: Aucune	Oui

Elements de la demande

Cette opération n'utilise pas les éléments de la demande.

Réponses

Amazon Glacier (Amazon Glacier) crée un hachage SHA256 arborescent de l'intégralité de l'archive. Si la valeur correspond au hachage SHA256 arborescent de l'intégralité de l'archive que vous avez spécifiée dans la demande, Amazon Glacier ajoute l'archive au coffre-fort. En réponse, il renvoie l'en-tête HTTP `Location` avec le chemin d'accès de l'URL de la ressource d'archivage nouvellement ajoutée. Si la taille de l'archive ou SHA256 celle que vous avez envoyée dans la demande ne correspond pas, Amazon Glacier renvoie un message d'erreur et le chargement reste incomplet. Il

est possible de renouveler ultérieurement l'opération d'achèvement du chargement partitionné avec les valeurs correctes pour créer correctement une archive. Si un chargement en plusieurs parties n'aboutit pas, Amazon Glacier finira par récupérer l'ID de téléchargement.

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
x-amz-archive-id: ArchiveId
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`Location`	Chemin d'accès URI relatif de l'archive nouvellement créée. Cette URL inclut l'ID d'archive généré par Amazon Glacier. Type : String
`x-amz-archive-id`	ID de l'archive. Cette valeur fait également partie de l'en-tête `Location`. Type : String

Champs de réponse

Cette opération ne renvoie pas de corps de réponse.

Exemple

Exemple de requête

Dans cet exemple, une demande HTTP POST est envoyée à l'URI qui a été renvoyé par une demande de lancement du chargement partitionné. La demande indique à la fois le hachage SHA256 arborescent de l'archive entière et la taille totale de l'archive.

```
POST /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
z-amz-Date: 20170210T120000Z
x-amz-sha256-tree-hash:1ffc0f54dd5fdd66b62da70d25edacd0
x-amz-archive-size:8388608
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

L'exemple de réponse suivant montre qu'Amazon Glacier a créé avec succès une archive à partir des éléments que vous avez chargés. La réponse inclut l'ID de l'archive avec le chemin d'accès complet.

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Location: /111122223333/vaults/examplevault/archives/
NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
x-amz-archive-id: NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
```

Vous pouvez maintenant envoyer des demandes HTTP à l'URI de la ressource/archive nouvellement ajoutée. Par exemple, vous pouvez envoyer une demande GET pour extraire l'archive.

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)
- Partie chargement (PUT uploadID)

- Annulation de chargement partitionné (DELETE uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Répertorier les parties (GET uploadID)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Suppression d'une archive (DELETE archive)
- Identity and Access Management pour Amazon Glacier

Lancement de chargement partitionné (POST multipart-uploads)

Description

Cette opération lance un chargement partitionné (consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)). Amazon Glacier (Amazon Glacier) crée une ressource de téléchargement en plusieurs parties et renvoie son identifiant dans la réponse. Vous utilisez cet ID de chargement dans les opérations de chargement partitionné suivantes.

Lorsque vous lancez un chargement partitionné, vous spécifiez la taille de la partie en indiquant le nombre d'octets correspondant. La taille de la partie doit être exprimée en mébioctet (Mio) (1 024 kibioctets [Kio]) multiplié par une puissance de 2 (par exemple, 1 048 576 (1 Mio), 2 097 152 (2 Mio), 4 194 304 (4 Mio), 8 388 608 (8 Mio), et ainsi de suite). La taille minimale autorisée pour les parties est de 1 Mio et la taille maximale est de 4 gibioctets (Gio).

Chacune des parties que vous chargez à l'aide de cet ID de chargement, à l'exception de la dernière, doit être de taille identique. La dernière partie peut être de taille identique ou plus petite. Par exemple, supposez que vous souhaitez charger un fichier de 16,2 Mio. Si vous lancez le chargement partitionné avec une taille de partie de 4 Mio, vous allez charger quatre parties de 4 Mio chacune et une partie de 0,2 Mio.

> **Note**
>
> Vous n'avez pas besoin de connaître la taille de l'archive lorsque vous lancez un téléchargement partitionné, car Amazon Glacier ne vous oblige pas à spécifier la taille globale de l'archive.

Une fois le téléchargement partitionné terminé, Amazon Glacier supprime la ressource de téléchargement partitionné référencée par l'ID. Amazon Glacier supprimera également la ressource

de téléchargement partitionné si vous annulez le téléchargement partitionné ou elle peut être supprimée s'il n'y a aucune activité pendant une période de 24 heures. Il se peut que l'ID soit toujours disponible après 24 heures, mais ce n'est pas le comportement auquel les applications doivent s'attendre.

Requêtes

Pour lancer un chargement partitionné, vous envoyez une demande HTTP `POST` à l'URI de la sous-ressource `multipart-uploads` du coffre dans lequel vous souhaitez enregistrer l'archive. La demande doit inclure la taille de la partie et peut éventuellement comporter une description de l'archive.

Syntaxe

```
POST /AccountId/vaults/VaultName/multipart-uploads
Host: glacier.us-west-2.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
x-amz-archive-description: ArchiveDescription
x-amz-part-size: PartSize
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise les en-têtes de demande suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Nom	Description	Obligatoire
`x-amz-part-size`	Taille de chaque partie, à l'exception de la dernière, exprimée en octets. La taille de la dernière partie peut être inférieure à celle de cette partie. Type : Chaîne Par défaut : aucun Contraintes : la taille de la partie doit être exprimée en mébioctet (1 024 Kio) multiplié par une puissance de 2 (par exemple, 1 048 576 (1 Mio), 2 097 152 (2 Mio), 4 194 304 (4 Mio), 8 388 608 (8 Mio), et ainsi de suite). La taille minimale autorisée pour les parties est de 1 Mio et la taille maximale est de 4 Gio (4 096 Mio).	Oui
`x-amz-archive-description`	Description de l'archive que vous chargez en plusieurs parties. Il peut s'agir d'une description rédigée dans un style simple ou d'un identifiant unique que vous choisissez d'attribuer. Lorsque vous procédez à l'extraction d'un inventaire de coffre (consultez la section Lancement d'une tâche (POST jobs)), l'inventaire inclut cette description pour chacune des archives renvoyées dans la réponse. Les espaces de début présents dans les descriptions d'archive sont retirés. Type : Chaîne Par défaut : aucun	Non

Nom	Description	Obligatoire
	Contraintes : la description doit être inférieur e ou égale à 1 024 octets. Les caractères autorisés sont des caractères ASCII 7 bits sans code de contrôle, plus spécifiquement les valeurs ASCII décimales 32 à 126 ou hexadécimales 0x20 à 0x7E.	

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Dans la réponse, Amazon Glacier crée une ressource de téléchargement partitionné identifiée par un ID et renvoie le chemin URI relatif de l'ID de téléchargement partitionné.

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
x-amz-multipart-upload-id: multiPartUploadId
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`Location`	Le chemin d'URI relatif de l'ID de téléchargement en plusieurs parties créé par Amazon Glacier. Vous utilisez ce chemin d'accès par URI pour définir la portée de vos demandes de chargement de parties et pour terminer le chargement partitionné.

Nom	Description
	Type : Chaîne
`x-amz-multipart-upload-id`	ID du chargement partitionné. Cette valeur fait également partie de l'en-tête `Location`. Type : Chaîne

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d'erreur.

Exemple

Exemple de requête

L'exemple suivant lance un chargement partitionné en envoyant une demande HTTP `POST` à l'URI de la sous-ressource `multipart-uploads` d'un coffre nommé `examplevault`. La demande comprend des en-têtes pour spécifier une taille de partie de 4 Mio (4 194 304 octets) une description facultative de l'archive.

```
POST /-/vaults/examplevault/multipart-uploads
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-archive-description: MyArchive-101
x-amz-part-size: 4194304
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Amazon Glacier crée une ressource de téléchargement en plusieurs parties et l'ajoute à la `multipart-uploads` sous-ressource du coffre-fort. L'en-tête de réponse `Location` inclut le chemin d'accès par URI relatif à l'ID de chargement partitionné.

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Location: /111122223333/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE
x-amz-multipart-upload-id:
 OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE
```

Pour plus d'informations sur le chargement de plusieurs parties, consultez la section Partie chargement (PUT uploadID).

Sections connexes

- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Annulation de chargement partitionné (DELETE uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Répertorier les parties (GET uploadID)
- Suppression d'une archive (DELETE archive)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Identity and Access Management pour Amazon Glacier

Répertorier les parties (GET uploadID)

Description

Cette opération de chargement partitionné répertorie les parties d'une archive qui ont été chargées dans le cadre d'un chargement partitionné spécifique identifié par un ID de chargement. Pour plus

d'informations sur le chargement partitionné, consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Vous pouvez faire cette demande à tout moment au cours d'un chargement partitionné en cours avant d'avoir terminé le chargement partitionné. Amazon Glacier retourne la liste des parties triée en fonction de la plage que vous avez spécifiée dans chaque chargement de partie. Si vous envoyez une demande List Parts après avoir effectué le téléchargement en plusieurs parties, Amazon Glacier (Amazon Glacier) renvoie un message d'erreur.

L'opération de création d'une liste des parties prend en charge la pagination. Vous devez toujours vérifier le champ `Marker` dans le corps de la réponse pour repérer un marqueur à partir duquel poursuivre l'affichage de la liste. S'il n'y a aucun autre élément, le champ `marker` est `null`. Si `marker` ce n'est pas nul, pour récupérer le prochain ensemble de pièces, vous avez envoyé une autre demande List Parts avec le paramètre de `marker` demande défini sur la valeur de marqueur renvoyée par Amazon Glacier en réponse à votre précédente demande List Parts.

Vous pouvez également limiter le nombre de parties renvoyées dans la réponse en spécifiant le paramètre `limit` dans la demande.

Requêtes

Syntaxe

Pour afficher les parties d'un chargement partitionné en cours, vous envoyez une demande `GET` à l'URI de la ressource de l'ID de chargement partitionné. L'ID de chargement partitionné est retourné lorsque vous lancez un chargement partitionné (Lancement de chargement partitionné (POST multipart-uploads)). Vous pouvez éventuellement spécifier les paramètres `marker` et `limit`.

```
GET /AccountId/vaults/VaultName/multipart-uploads/uploadID HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations

d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Nom	Description	Obligatoire
`limit`	Le nombre maximum de parties à renvoyer. La limite par défaut est de 50. Il peut arriver que le nombre de parties renvoyées soit inférieur à la limite spécifiée, mais il ne sera jamais supérieur à cette valeur. Type : Chaîne Contraintes : la valeur d'entier minimale est 1. La valeur d'entier maximale est `50`.	Non
`marker`	Chaîne opaque utilisée pour la pagination. `marker` spécifie la partie à partir de laquelle la liste des parties doit commencer . Obtenez la valeur `marker` dans une réponse précédente concernant l'établissement d'une liste des parties. Vous devez simplement inclure la valeur du paramètre `marker` si vous poursuivez la pagination de résultats commencée dans une demande précédente d'affichage de la liste des parties. Type : Chaîne Contraintes: Aucune	Non

En-têtes de demande

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length

{
    "ArchiveDescription" : String,
    "CreationDate" : String,
    "Marker": String,
    "MultipartUploadId" : String,
    "PartSizeInBytes" : Number,
    "Parts" :
    [ {
      "RangeInBytes" : String,
      "SHA256TreeHash" : String
      },
      ...
     ],
    "VaultARN" : String
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

ArchiveDescription

La description de l'archive qui a été spécifiée dans la demande de lancement des chargements partitionnés. Ce champ est `null` si aucune description d'archive n'a été spécifiée dans l'opération de lancement des chargements partitionnés.

Type : chaîne

CreationDate

L'heure UTC à laquelle le chargement partitionné a été lancé.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

Marker

Chaîne opaque représentant l'endroit à partir duquel poursuivre la pagination des résultats. Vous utilisez le `marker` dans une nouvelle demande d'affichage des parties afin d'obtenir davantage de tâches dans la liste. S'il n'y a pas d'autres parties, cette valeur est `null`.

Type : chaîne

MultipartUploadId

L'ID du chargement auquel les parties sont associées.

Type : chaîne

PartSizeInBytes

La taille de la partie en octets. Il s'agit de la même valeur que celle qui a été spécifiée dans la demande de lancement d'une demande de chargement partitionné.

Type : nombre

Parties

Une liste des tailles des parties pour le chargement partitionné. Chaque objet du tableau contient une paire nom/valeur `RangeBytes` et `sha256-tree-hash`.

Type : tableau

RangeInBytes

La plage d'octets d'une partie, y compris la valeur supérieure de la plage.

Type : chaîne

SHA256TreeHash

Valeur de hachage de l' SHA256 arbre calculée par Amazon Glacier pour la pièce. Ce champ n'est jamais `null`.

Type : chaîne

VaultARN

L'Amazon Resource Name (ARN) du coffre pour lequel le chargement partitionné a été lancé.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple : Liste différentes parties d'un chargement partitionné

L'exemple suivant répertorie toutes les parties d'un chargement. L'exemple envoie une demande `GET` HTTP à l'URI de l'ID de chargement partitionné d'un chargement partitionné en cours et retourne jusqu'à 1 000 parties.

Exemple de requête

```
GET /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Dans la réponse, Amazon Glacier renvoie une liste des pièces téléchargées associées à l'ID de téléchargement en plusieurs parties spécifié. Cet exemple contient deux parties seulement. Le

champ `Marker` retourné est `null`, ce qui indique qu'il n'y a plus aucune partie pour le chargement partitionné.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 412

{
    "ArchiveDescription" : "archive description",
    "CreationDate" : "2012-03-20T17:03:43.221Z",
    "Marker": null,
    "MultipartUploadId" :
 "OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE",
    "PartSizeInBytes" : 4194304,
    "Parts" :
    [ {
      "RangeInBytes" : "0-4194303",
      "SHA256TreeHash" : "01d34dabf7be316472c93b1ef80721f5d4"
      },
      {
      "RangeInBytes" : "4194304-8388607",
      "SHA256TreeHash" : "0195875365afda349fc21c84c099987164"
      }],
    "VaultARN" : "arn:aws:glacier:us-west-2:012345678901:vaults/demo1-vault"
}
```

Exemple : Répertorier les parties d'un chargement partitionné (indiquer le marqueur et les paramètres de limite de la demande)

L'exemple suivant illustre l'utilisation de la pagination pour obtenir un nombre limité de résultats. L'exemple envoie une demande GET HTTP à l'URI de l'ID de chargement partitionné d'un chargement partitionné en cours pour retourner une partie. Un paramètres `marker` de départ spécifie à partir de quelle partie la liste doit commencer. Vous pouvez obtenir la valeur `marker` dans la réponse d'une demande précédente de liste de parties. En outre, dans cet exemple, le paramètre `limit`est défini sur 1 et retourne une partie. Notez que le champ `Marker` n'est pas `null`, ce qui indique qu'il reste encore au moins une partie à obtenir.

Exemple de requête

```
GET /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE?marker=1001&limit=1 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Dans la réponse, Amazon Glacier renvoie une liste des pièces chargées associées à l'ID de téléchargement partitionné en cours spécifié.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: text/json
Content-Length: 412

{
    "ArchiveDescription" : "archive description 1",
    "CreationDate" : "2012-03-20T17:03:43.221Z",
    "Marker": "MfgsKHVjbQ6EldVl72bn3_n5h2TaGZQUO-Qb3B9j3TITf7WajQ",
    "MultipartUploadId" :
 "OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE",
    "PartSizeInBytes" : 4194304,
    "Parts" :
    [ {
      "RangeInBytes" : "4194304-8388607",
      "SHA256TreeHash" : "01d34dabf7be316472c93b1ef80721f5d4"
      }],
    "VaultARN" : "arn:aws:glacier:us-west-2:012345678901:vaults/demo1-vault"
}
```

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)

- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Annulation de chargement partitionné (DELETE uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Identity and Access Management pour Amazon Glacier

Afficher la liste des chargements partitionnés (GET multipart-uploads)

Description

Cette opération de chargement partitionné affiche la liste des chargements partitionnés en cours pour le coffre spécifié. Un chargement partitionné en cours est un chargement partitionné qui a été lancé par une demande Lancement de chargement partitionné (POST multipart-uploads), mais qui n'a pas encore été terminé ou arrêté. La liste renvoyée dans la réponse de chargement partitionné n'a aucun ordre garanti.

L'opération d'affichage de la liste des chargements partitionnés prend en charge la pagination. Par défaut, cette opération renvoie jusqu'à 50 chargements partitionnés dans la réponse. Vous devez toujours vérifier le champ `marker` dans le corps de la réponse pour repérer un marqueur à partir duquel poursuivre l'affichage de la liste. S'il n'y a aucun autre élément, le champ `marker` est `null`.

Si ce n'`marker`est pas nul, pour récupérer le prochain ensemble de téléchargements partitionnés, vous avez envoyé une autre demande List Multipart Uploads avec le paramètre de requête défini sur la valeur de marqueur qu'Amazon Glacier (Amazon Glacier) a renvoyée en réponse à votre précédente `marker` demande List Multipart Uploads.

Notez la différence entre cette opération et l'opération Répertorier les parties (GET uploadID)). L'opération d'affichage des chargements partitionnés affiche la liste de tous les chargements partitionnés pour un coffre. L'opération d'affichage de la liste des parties renvoie les parties d'un chargement partitionné spécifique identifié par un ID de chargement.

Pour plus d'informations sur le chargement partitionné, consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Requêtes

Syntaxe

Pour afficher la liste des chargements partitionnés, envoyez une demande `GET` à l'URI de la sous-ressource `multipart-uploads` du coffre. Vous pouvez éventuellement spécifier les paramètres `marker` et `limit`.

```
GET /AccountId/vaults/VaultName/multipart-uploads HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Nom	Description	Obligatoire
`limit`	Spécifie le nombre maximal de chargements renvoyés dans le corps de la réponse. Si ce paramètre n'est pas spécifié, l'opération d'affichage de la liste des chargements renvoie jusqu'à 50 chargements. Type : Chaîne Contraintes : la valeur d'entier minimale est 1. La valeur d'entier maximale est `50`.	Non

Nom	Description	Obligatoire
`marker`	Chaîne opaque utilisée pour la pagination. `marker` spécifie le chargement auquel doit commencer la liste des chargements. Obtenez la valeur du paramètre `marker` à partir d'une réponse précédente d'affichage de la liste des chargements. Il suffit d'inclure la valeur du paramètre `marker` si vous poursuivez la pagination de résultats commencée dans une demande précédente d'affichage de la liste des chargements. Type : Chaîne Contraintes: Aucune	Non

En-têtes de demande

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length

{
  "Marker": String,
  "UploadsList" : [
    {
```

```
      "ArchiveDescription": String,
      "CreationDate": String,
      "MultipartUploadId": String,
      "PartSizeInBytes": Number,
      "VaultARN": String
    },
   ...
  ]
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

ArchiveDescription

La description de l'archive qui a été spécifiée dans la demande de lancement des chargements partitionnés. Ce champ est `null` si aucune description d'archive n'a été spécifiée dans l'opération de lancement des chargements partitionnés.

Type : chaîne

CreationDate

L'heure UTC à laquelle le chargement partitionné a été lancé.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

Marker

Chaîne opaque représentant l'endroit à partir duquel poursuivre la pagination des résultats. Vous utilisez le paramètre `marker` dans une nouvelle demande d'affichage de la liste des chargements partitionnés en vue d'obtenir davantage de chargements dans la liste. S'il n'y a pas d'autres chargements, cette valeur est `null`.

Type : chaîne

PartSizeInBytes

Taille de la partie spécifiée dans la demande Lancement de chargement partitionné (POST multipart-uploads). Il s'agit de la taille de toutes les parties comprises dans le téléchargement excepté la dernière partie, dont la taille peut être inférieure à cette taille.

Type : nombre

MultipartUploadId

ID du chargement partitionné.

Type : chaîne

UploadsList

Liste des métadonnées relatives aux objets des chargements partitionnés. Chaque élément de la liste contient un ensemble de paires nom-valeur pour le chargement correspondant, notamment `ArchiveDescription`, `CreationDate`, `MultipartUploadId`, `PartSizeInBytes` et `VaultARN`.

Type : tableau

VaultARN

Amazon Resource Name (ARN) du coffre qui contient l'archive.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple : Affichage de la liste de tous les chargements partitionnés

L'exemple suivant affiche la liste de tous les chargements partitionnés en cours pour le coffre. L'exemple illustre une demande HTTP `GET` envoyée à l'URI de la sous-ressource `multipart-uploads` d'un coffre spécifié. Etant donné que les paramètres `marker` et `limit` ne sont pas spécifiés dans la demande, jusqu'à 1 000 chargements partitionnés en cours sont renvoyés.

Exemple de requête

```
GET /-/vaults/examplevault/multipart-uploads HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Dans la réponse, Amazon Glacier renvoie une liste de tous les téléchargements partitionnés en cours pour le coffre spécifié. Le champ `marker` est `null`, ce qui indique qu'il n'y a pas plus de chargements à afficher dans la liste.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 1054

{
  "Marker": null,
  "UploadsList": [
    {
      "ArchiveDescription": "archive 1",
      "CreationDate": "2012-03-19T23:20:59.130Z",
      "MultipartUploadId":
 "xsQdFIRsfJr20CW2AbZBKpRZAFTZSJIMtL2hYf8mvp8dM0m4RUzlaqoEye6g3h3ecqB_zqwB7zLDMeSWhwo65re4C4Ev"
      "PartSizeInBytes": 4194304,
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    },
    {
      "ArchiveDescription": "archive 2",
      "CreationDate": "2012-04-01T15:00:00.000Z",
      "MultipartUploadId": "nPyGOnyFcx67qqX7E-0tSGiRi88hHMOwOxR-
_jNyM6RjVMFfV29lFqZ3rNsSaWBugg6OP92pRtufeHdQH7ClIpSF6uJc",
      "PartSizeInBytes": 4194304,
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    },
    {
      "ArchiveDescription": "archive 3",
```

```
      "CreationDate": "2012-03-20T17:03:43.221Z",
      "MultipartUploadId": "qt-RBst_7yO8gVIonIBsAxr2t-db0pE4s8MNeGjKjGdNpuU-
cdSAcqG62guwV9r5jh5mLyFPzFEitTpNE7iQfHiu1XoV",
      "PartSizeInBytes": 4194304,
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    }
  ]
}
```

Exemple : Liste partielle des chargements partitionnés

L'exemple suivant illustre l'utilisation de la pagination pour obtenir un nombre limité de résultats. L'exemple illustre une demande HTTP `GET` envoyée à l'URI de la sous-ressource `multipart-uploads` pour un coffre spécifié. Dans cet exemple, le paramètre `limit` est défini sur 1, ce qui signifie qu'un seul chargement est renvoyé dans la liste, et le paramètre `marker` indique l'ID de chargement partitionné auquel commence la liste renvoyée.

Exemple de requête

```
GET /-/vaults/examplevault/multipart-uploads?
limit=1&marker=xsQdFIRsfJr20CW2AbZBKpRZAFTZSJIMtL2hYf8mvp8dM0m4RUzlaqoEye6g3h3ecqB_zqwB7zLDMeSw
 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Dans la réponse, Amazon Glacier (Amazon Glacier) renvoie une liste ne contenant pas plus de deux téléchargements partitionnés en cours pour le coffre spécifié, en commençant par le marqueur spécifié et en renvoyant deux résultats.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 470

{
```

```
  "Marker": "qt-RBst_7yO8gVIonIBsAxr2t-db0pE4s8MNeGjKjGdNpuU-
cdSAcqG62guwV9r5jh5mLyFPzFEitTpNE7iQfHiu1XoV",
  "UploadsList" : [
    {
      "ArchiveDescription": "archive 2",
      "CreationDate": "2012-04-01T15:00:00.000Z",
      "MultipartUploadId": "nPyGOnyFcx67qqX7E-0tSGiRi88hHMOwOxR-
_jNyM6RjVMFfV29lFqZ3rNsSaWBugg6OP92pRtufeHdQH7ClIpSF6uJc",
      "PartSizeInBytes": 4194304,
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    }
  ]
}
```

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)
- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Annulation de chargement partitionné (DELETE uploadID)
- Répertorier les parties (GET uploadID)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Identity and Access Management pour Amazon Glacier

Partie chargement (PUT uploadID)

Description

Cette opération de chargement partitionné permet de charger une partie d'une archive. Vous pouvez charger des parties de l'archive dans n'importe quel ordre car vous spécifiez la plage d'octets de l'archive assemblée qui sera chargée dans cette partie dans votre demande de chargement partiel. Vous pouvez également charger ces différentes parties en parallèle. Vous pouvez charger jusqu'à 10 000 parties pour un chargement partitionné.

Pour plus d'informations sur le chargement partitionné, consultez Chargement d'archives volumineuses en plusieurs parties (chargement partitionné).

Amazon Glacier (Amazon Glacier) rejette votre demande de partie chargée si l'une des conditions suivantes est remplie :

- SHA256 le hachage de l'arbre ne correspond pas : pour vous assurer que les données de la pièce ne sont pas corrompues lors de la transmission, vous calculez un hachage SHA256 arborescent de la pièce et vous l'incluez dans votre demande. Dès réception des données de pièce, Amazon Glacier calcule également un hachage SHA256 arborescent. Si les deux valeurs de hachage ne correspondent pas, l'opération échoue. Pour plus d'informations sur le calcul d'un hachage SHA256 arborescent, consultezCalcul des totaux de contrôle.
- SHA256 le hachage linéaire ne correspond pas : pour obtenir l'autorisation, vous calculez un hachage SHA256 linéaire de l'ensemble de la charge utile téléchargée et vous l'incluez dans votre demande. Pour plus d'informations sur le calcul d'un hachage SHA256 linéaire, consultezCalcul des totaux de contrôle.
- La taille de la partie ne correspond pas – La taille de chaque partie, à l'exception de la dernière, doit correspondre à la taille spécifiée dans la demande Lancement de chargement partitionné (POST multipart-uploads) correspondante. La taille de la dernière partie doit être identique, ou inférieure, à la taille spécifiée.

  > **Note**
  >
  > Si vous chargez une partie dont la taille est inférieure à la taille de la partie spécifiée dans votre première demande de chargement partitionné et si cette partie n'est pas la dernière, la demande de chargement de la partie est réussie. Toutefois, la demande suivante de chargement partitionné complet échouera.

- La plage ne correspond pas – La valeur de la plage d'octets de la demande ne correspond pas à la taille de la partie spécifiée dans la demande de lancement correspondante. Par exemple, si vous spécifiez une taille de partie de 4 194 304 octets (4 Mo), les plages de parties comprises entre 0 et 4 194 303 octets (4 Mo - 1) et entre 4 194 304 (4 Mo) et 8 388 607 (8 Mo - 1) sont valides. Toutefois, si vous définissez une valeur de plage de 2 Mo à 6 Mo, la plage ne correspond pas à la taille de la partie et le chargement échouera.

Cette opération est idempotente. Si vous téléchargez la même partie plusieurs fois, les données figurant dans la demande plus récente remplace les données chargées précédemment.

Requêtes

Vous envoyez cette demande PUT HTTP à l'URI de l'ID de chargement qui a été renvoyé par votre demande de lancement de chargement partitionné. Amazon Glacier utilise l'ID de téléchargement pour associer les chargements partiels à un chargement partitionné spécifique. La demande doit inclure un hachage SHA256 arborescent des données de pièce (`x-amz-SHA256-tree-hash`en-tête), un hachage SHA256 linéaire de l'ensemble de la charge utile (`x-amz-content-sha256`en-tête), la plage d'octets (`Content-Range`en-tête) et la longueur de la partie en octets (`Content-Length`en-tête).

Syntaxe

```
PUT /AccountId/vaults/VaultName/multipart-uploads/uploadID HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Content-Range: ContentRange
Content-Length: PayloadSize
Content-Type: application/octet-stream
x-amz-sha256-tree-hash: Checksum of the part
x-amz-content-sha256: Checksum of the entire payload
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise les en-têtes de demande suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Nom	Description	Obligatoire
`Content-Length`	Identifie la longueur de la partie en octets. Type : String Par défaut : aucun Contraintes: Aucune	Non
`Content-Range`	Identifie la plage d'octets de l'archive assemblée qui sera chargée dans cette partie. Amazon Glacier utilise ces informations pour assembler l'archive dans le bon ordre. Le format de cet en-tête suit RFC 2616. Voici un exemple d'en-tête : `Content-Range:bytes 0-4194303/*` . Type : String Par défaut : aucun Contraintes : la plage ne peut pas être supérieure à la taille de la partie que vous avez spécifiée lorsque vous avez lancé le chargement partitionné.	Oui
`x-amz-content-sha256`	La SHA256 somme de contrôle (un hachage linéaire) de la charge utile téléchargée. Il ne s'agit pas de la valeur que vous spécifiez dans l'en-tête `x-amz-sha256-tree-hash` . Type : String	Oui

Nom	Description	Obligatoire
	Par défaut : aucun Contraintes: Aucune	
`x-amz-sha256-tree-hash`	Spécifie un hachage SHA256 arborescent des données en cours de téléchargement. Pour plus d'informations sur le calcul d'un hachage SHA256 arborescent, consultezCalcul des totaux de contrôle. Type : String Par défaut : aucun Contraintes: Aucune	Oui

Corps de la requête

Le corps de la demande contient les données à charger.

Réponses

Une fois le chargement partiel réussi, Amazon Glacier renvoie une `204 No Content` réponse.

Syntaxe

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-sha256-tree-hash: ChecksumComputedByAmazonGlacier
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`x-amz-sha256-tree-hash`	Le hachage de l' SHA256 arbre calculé par Amazon Glacier pour la partie chargée. Type : String

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Exemple

La demande suivante charge une partie de 4 Mo. La demande définit la plage d'octets afin que ce soit la première partie de l'archive.

Exemple de requête

L'exemple envoie une demande PUT HTTP pour charger une partie de 4 Mo. La demande est envoyée à l'URI de l'ID de chargement qui a été retournée par la demande de lancement du chargement partitionné. L'en-tête `Content-Range` identifie la partie en tant que première partie de 4 Mo de l'archive.

```
PUT /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Range:bytes 0-4194303/*
x-amz-sha256-tree-hash:c06f7cd4baacb087002a99a5f48bf953
x-amz-content-sha256:726e392cb4d09924dbad1cc0ba3b00c3643d03d14cb4b823e2f041cff612a628
Content-Length: 4194304
Authorization: Authorization=AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-content-sha256;x-amz-date;x-
amz-glacier-
version,Signature=16b9a9e220a37e32f2e7be196b4ebb87120ca7974038210199ac5982e792cace
```

Pour télécharger la pièce suivante, la procédure est la même ; toutefois, vous devez calculer un nouveau hachage SHA256 arborescent de la pièce que vous chargez et également spécifier une nouvelle plage d'octets pour indiquer où la pièce sera placée dans l'assemblage final. La demande

suivante charge une autre partie en utilisant le même ID de chargement. La demande spécifie les 4 Mo suivants de l'archive après la demande précédente et une taille de partie de 4 Mo.

```
PUT /-/vaults/examplevault/multipart-uploads/
OW2fM5iVylEpFEMM9_HpKowRapC3vn5sSL39_396UW9zLFUWVrnRHaPjUJddQ5OxSHVXjYtrN47NBZ-
khxOjyEXAMPLE HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Range:bytes 4194304-8388607/*
Content-Length: 4194304
x-amz-sha256-tree-hash:f10e02544d651e2c3ce90a4307427493
x-amz-content-sha256:726e392cb4d09924dbad1cc0ba3b00c3643d03d14cb4b823e2f041cff612a628
x-amz-glacier-version: 2012-06-01
Authorization: Authorization=AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20120525/
us-west-2/glacier/aws4_request, SignedHeaders=host;x-amz-content-sha256;x-amz-date;x-
amz-glacier-version,
 Signature=16b9a9e220a37e32f2e7be196b4ebb87120ca7974038210199ac5982e792cace
```

Les pièces peuvent être chargées dans n'importe quelle commande ; Amazon Glacier utilise les spécifications de plage de chaque pièce pour déterminer l'ordre dans lequel les assembler.

Exemple de réponse

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
x-amz-sha256-tree-hash: c06f7cd4baacb087002a99a5f48bf953
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sections connexes

- Lancement de chargement partitionné (POST multipart-uploads)
- Partie chargement (PUT uploadID)
- Achèvement du chargement partitionné (POST uploadID)
- Annulation de chargement partitionné (DELETE uploadID)
- Afficher la liste des chargements partitionnés (GET multipart-uploads)
- Répertorier les parties (GET uploadID)
- Chargement d'archives volumineuses en plusieurs parties (chargement partitionné)
- Identity and Access Management pour Amazon Glacier

Opérations de tâche

Les opérations de travail disponibles dans Amazon Glacier sont les suivantes.

Rubriques

- Description de la tâche (GET JobID)
- Génération de sortie de tâche (GET output)
- Lancement d'une tâche (POST jobs)
- Affichage des tâches (GET jobs)

Description de la tâche (GET JobID)

Description

Cette opération renvoie des informations sur une tâche que vous avez initiée précédemment, notamment la date de lancement de la tâche, l'utilisateur qui a initié la tâche, le code/message de statut de la tâche et la rubrique Amazon Simple Notification Service (Amazon SNS) à laquelle vous devez vous informer une fois qu'Amazon Glacier (Amazon Glacier) a terminé la tâche. Pour plus d'informations sur le lancement d'une tâche, consultez Lancement d'une tâche (POST jobs).

> **Note**
>
> Cette opération vous permet de vérifier l'état de votre tâche. Cependant, nous vous recommandons vivement de configurer une rubrique Amazon SNS et de la spécifier dans votre demande de travail d'initiation afin qu'Amazon Glacier puisse informer le sujet une fois la tâche terminée.

Un identifiant de tâche n'expirera pas pendant au moins 24 heures après qu'Amazon Glacier ait terminé la tâche.

Requêtes

Syntaxe

Pour obtenir des informations sur une tâche, utilisez la méthode GET HTTP et limitez la portée de la demande à la tâche spécifique. Notez que le chemin d'URI relatif est le même que celui qu'Amazon Glacier vous a renvoyé lorsque vous avez lancé la tâche.

```
GET /AccountID/vaults/VaultName/jobs/JobID HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: date
Authorization: signatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

> **Note**
>
> Dans la demande, si vous omettez le `JobID`, la réponse renvoie la liste de toutes les tâches actives sur le coffre spécifié. Pour plus d'informations sur l'affichage de la liste des tâches, consultez Affichage des tâches (GET jobs).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length

{
    "Action": "string",
    "ArchiveId": "string",
    "ArchiveSHA256TreeHash": "string",
    "ArchiveSizeInBytes": number,
    "Completed": boolean,
    "CompletionDate": "string",
    "CreationDate": "string",
    "InventoryRetrievalParameters": {
        "EndDate": "string",
        "Format": "string",
        "Limit": "string",
        "Marker": "string",
        "StartDate": "string"
    },
    "InventorySizeInBytes": number,
    "JobDescription": "string",
    "JobId": "string",
    "JobOutputPath": "string",
    "OutputLocation": {
        "S3": {
            "AccessControlList": [
                {
                    "Grantee": {
                        "DisplayName": "string",
                        "EmailAddress": "string",
                        "ID": "string",
                        "Type": "string",
                        "URI": "string"
                    },
```

```
                    "Permission": "string"
                }
            ],
            "BucketName": "string",
            "CannedACL": "string",
            "Encryption": {
                "EncryptionType": "string",
                "KMSContext": "string",
                "KMSKeyId": "string"
            },
            "Prefix": "string",
            "StorageClass": "string",
            "Tagging": {
                "string": "string"
            },
            "UserMetadata": {
                "string": "string"
            }
        }
    },
    "RetrievalByteRange": "string",
    "SelectParameters": {
        "Expression": "string",
        "ExpressionType": "string",
        "InputSerialization": {
            "csv": {
                "Comments": "string",
                "FieldDelimiter": "string",
                "FileHeaderInfo": "string",
                "QuoteCharacter": "string",
                "QuoteEscapeCharacter": "string",
                "RecordDelimiter": "string"
            }
        },
        "OutputSerialization": {
            "csv": {
                "FieldDelimiter": "string",
                "QuoteCharacter": "string",
                "QuoteEscapeCharacter": "string",
                "QuoteFields": "string",
                "RecordDelimiter": "string"
            }
        }
    },
```

```
    "SHA256TreeHash": "string",
    "SNSTopic": "string",
    "StatusCode": "string",
    "StatusMessage": "string",
    "Tier": "string",
    "VaultARN": "string"
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

Action

Le type de tâche. Il s'agit de `ArchiveRetrieval`, `InventoryRetrieval` ou `Select`.

Type : chaîne

ArchiveId

ID d'archive demandé pour une tâche de sélection ou d'extraction d'archive. Sinon, ce champ est `null`.

Type : chaîne

ArchivageSHA256TreeHash

Le hachage SHA256 arborescent de l'ensemble de l'archive pour une tâche de récupération d'archives. Pour les tâches d'extraction d'inventaire, ce champ est `null`.

Type : chaîne

ArchiveSizeInBytes

Pour une tâche `ArchiveRetrieval`, il s'agit de la taille en octets de l'archive demandée au téléchargement. Pour la tâche `InventoryRetrieval`, la valeur est `null`.

Type : nombre

Terminé

L'état de la tâche. Lorsqu'une tâche d'extraction d'archive ou d'inventaire est terminée, vous obtenez la sortie de la tâche à l'aide de Génération de sortie de tâche (GET output).

Type : valeur booléenne

CompletionDate

La date UTC lors de la fin de la demande de tâche. Lorsque la tâche est en cours, cette valeur est null.

Type : chaîne

CreationDate

L'heure UTC à laquelle la tâche a été créée.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

InventoryRetrievalParameters

Paramètres d'entrée utilisés pour l'extraction d'un inventaire par plage.

Type : objet InventoryRetrievalJobInput

InventorySizeInBytes

Pour une tâche `InventoryRetrieval`, il s'agit de la taille en octets de l'inventaire dont le téléchargement est demandé. Pour la tâche `ArchiveRetrieval` ou `Select`, la valeur est `null`.

Type : nombre

JobDescription

La description de tâche que vous avez fournie lorsque vous avez lancé la tâche.

Type : chaîne

JobId

ID identifiant la tâche dans Amazon Glacier.

Type : chaîne

JobOutputPath

Contient l'emplacement de la sortie de la tâche.

Type : chaîne

OutputLocation

Objet qui contient des informations sur l'emplacement auquel les erreurs et résultats de tâches de sélection sont stockés.

Type : objet OutputLocation

RetrievalByteRange

La plage d'octets récupérée pour les tâches de récupération d'archives sous la forme « *StartByteValue* - »*EndByteValue*. Si vous ne spécifiez pas de plage dans la récupération de l'archive, l'archive entière est récupérée ; elle est également StartByteValueégale à 0 et EndByteValueégale à la taille de l'archive moins 1. Pour les tâches d'extraction d'inventaire ou de sélection, ce champ est `null`.

Type : chaîne

SelectParameters

Objet qui contient les informations sur les paramètres utilisés pour une tâche de sélection.

Type : objet SelectParameters

SHA256TreeHash

La valeur de hachage de l' SHA256 arbre pour la plage demandée d'une archive. Si la demande Lancement d'une tâche (POST jobs) définie pour une archive spécifiait une plage alignée avec le hachage d'arborescence, ce champ renvoie une valeur. Pour plus d'informations sur l'alignement avec le hachage d'arborescence dans le cas d'extractions par plage d'archive, consultez Réception des totaux de contrôle lors du téléchargement de données.

Lorsque la totalité de l'archive est extraite, cette valeur est identique à celle de `ArchiveSHA256TreeHash`.

La valeur de ce champ est `null` dans les cas suivants :

- Tâches d'extraction d'archive qui spécifient une plage qui n'est pas alignée avec le hachage d'arborescence.
- Tâches d'archivage qui spécifient une plage qui correspond à la totalité de l'archive et dont l'état est `InProgress`.
- Tâches d'inventaire.

- Tâches de sélection.

Type : chaîne

SNSTopic

Rubrique Amazon SNS qui reçoit une notification.

Type : chaîne

StatusCode

Code indiquant l'état de la tâche.

Valeurs Valides : `InProgress` | `Succeeded` | `Failed`

Type : chaîne

StatusMessage

Message convivial qui décrit l'état de la tâche.

Type : chaîne

Palier

Niveau d'accès aux données à utiliser pour la tâche de sélection ou d'extraction d'archive.

Valeurs Valides : `Bulk` | `Expedited` | `Standard`

Type : chaîne

VaultARN

Amazon Resource Name (ARN) du coffre dont la tâche est une sous-ressource.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant illustre une demande de tâche visant à extraire une archive.

Exemple de demande : Obtention de la description de la tâche

```
GET /-/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Le corps de réponse inclut du code JSON qui décrit la tâche spécifiée. Notez que les champs JSON sont identiques pour la tâche d'extraction d'inventaire et la tâche d'extraction d'archive. Cependant, lorsqu'un champ ne s'applique pas au type de tâche, sa valeur est `null`. Voici un exemple de réponse pour une tâche d'extraction d'archive. Notez ce qui suit :

- La valeur du champ `Action` est `ArchiveRetrieval`.
- Le champ `ArchiveSizeInBytes` affiche la taille de l'archive demandée dans la tâche d'extraction d'archive.
- Le `ArchiveSHA256TreeHash` champ indique le hachage de l' SHA256 arborescence pour l'ensemble de l'archive.
- Le champ `RetrievalByteRange` affiche la plage demandée dans la demande de lancement de tâche. Dans cet exemple, l'archive entière est demandée.
- Le `SHA256TreeHash` champ indique le hachage SHA256 arborescent pour la plage demandée dans la demande Initiate Job. Dans cet exemple, il est identique au champ `ArchiveSHA256TreeHash`, ce qui signifie que l'archive entière a été demandée.
- La valeur du champ `InventorySizeInBytes` est `null`, car elle ne s'applique pas à une tâche d'extraction d'archive.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 419
{
```

```
  "Action": "ArchiveRetrieval",
  "ArchiveId": "NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
  "ArchiveSizeInBytes": 16777216,
  "ArchiveSHA256TreeHash":
 "beb0fe31a1c7ca8c6c04d574ea906e3f97b31fdca7571defb5b44dca89b5af60",
  "Completed": false,
  "CompletionDate": null,
  "CreationDate": "2012-05-15T17:21:39.339Z",
  "InventorySizeInBytes": null,
  "JobDescription": "My ArchiveRetrieval Job",
  "JobId": "HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID",
  "RetrievalByteRange": "0-16777215",
  "SHA256TreeHash": "beb0fe31a1c7ca8c6c04d574ea906e3f97b31fdca7571defb5b44dca89b5af60",
  "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic",
  "StatusCode": "InProgress",
  "StatusMessage": "Operation in progress.",
  "Tier": "Bulk",
  "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
}
```

Voici un exemple de réponse pour une tâche d'extraction d'inventaire. Notez ce qui suit :

- La valeur du champ `Action` est `InventoryRetrieval`.
- Les valeurs de champ `ArchiveSizeInBytes`, `ArchiveSHA256TreeHash` et `RetrievalByteRange` sont nulles, car ces champs ne s'appliquent pas à une tâche d'extraction d'inventaire.
- La valeur de champ `InventorySizeInBytes` est `null`, car la tâche est encore en cours et n'a pas encore entièrement préparé l'inventaire au téléchargement. Si la tâche était terminée avant votre demande de description de la tâche, ce champ vous indiquerait la taille de la sortie.

```
{
   "Action": "InventoryRetrieval",
   "ArchiveId": null,
   "ArchiveSizeInBytes": null,
   "ArchiveSHA256TreeHash": null,
   "Completed": false,
   "CompletionDate": null,
   "CreationDate": "2012-05-15T23:18:13.224Z",
```

```
   "InventorySizeInBytes": null,
   "JobDescription": "Inventory Description",
   "JobId": "HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID",
   "RetrievalByteRange": null,
   "SHA256TreeHash": null,
   "SNSTopic": "arn:aws:sns:us-west-2:012345678901:mytopic",
   "StatusCode": "InProgress",
   "StatusMessage": "Operation in progress.",
   "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
}
```

Voici un exemple de réponse pour une tâche d'extraction d'inventaire terminée qui contient un marqueur utilisé pour poursuivre la pagination de l'extraction de l'inventaire du coffre.

```
{
    "Action": "InventoryRetrieval",
    "ArchiveId": null,
    "ArchiveSHA256TreeHash": null,
    "ArchiveSizeInBytes": null,
    "Completed": true,
    "CompletionDate": "2013-12-05T21:51:13.591Z",
    "CreationDate": "2013-12-05T21:51:12.281Z",
    "InventorySizeInBytes": 777062,
    "JobDescription": null,
    "JobId": "sCC2RZNBF2nildYD_roe0J9bHRdPQUbDRkmTdg-mXi2u3lc49uW6TcEhDF2D9pB2phx-
BN30JaBru7PMyOlfXHdStzu8",
    "NextInventoryRetrievalMarker": null,
    "RetrievalByteRange": null,
    "SHA256TreeHash": null,
    "SNSTopic": null,
    "StatusCode": "Succeeded",
    "StatusMessage": "Succeeded",
    "Tier": "Bulk",
    "VaultARN": "arn:aws:glacier-devo:us-west-2:836579025725:vaults/inventory-
icecube-2",
    "InventoryRetrievalParameters": {
        "StartDate": "2013-11-12T13:43:12Z",
        "EndDate": "2013-11-20T08:12:45Z",
        "Limit": "120000",
        "Format": "JSON",
        "Marker":
 "vyS0t2jHQe5qbcDggIeD50chS1SXwYMrkVKo0KHiTUjEYxBGCqRLKaiySzdN7QXGVVV5XZpNVG67pCZ_uykQXFMLaxOSu
```

```
    },
}
```

Sections connexes

- Génération de sortie de tâche (GET output)
- Identity and Access Management pour Amazon Glacier

Génération de sortie de tâche (GET output)

Description

Cette opération télécharge la sortie de la tâche que vous avez lancée à l'aide de l'opération Lancement d'une tâche (POST jobs). Selon le type de tâche que vous avez spécifié lors du lancement de la tâche, la sortie représente le contenu d'une archive ou d'un inventaire de coffre.

Vous pouvez télécharger la totalité de la sortie d'une tâche ou une partie seulement de la sortie en spécifiant une plage d'octets. Pour les tâches d'extraction d'archive et les tâches d'extraction d'inventaire, vous devez comparer la taille téléchargée à la taille renvoyée dans les en-têtes de la réponse Génération de sortie de tâche.

Pour les tâches d'extraction d'archive, vous devez également vérifier que la taille obtenue est conforme à ce que vous aviez prévu. Si vous téléchargez une partie de la sortie, la taille prévue est basée sur la plage d'octets que vous avez spécifiée. Par exemple, si vous spécifiez la plage `bytes=0-1048575`, vous devez vérifier que la taille de votre téléchargement est 1 048 576 octets. Si vous téléchargez une archive complète, la taille attendue est la taille de l'archive lorsque vous l'avez chargée sur Amazon Glacier (Amazon Glacier). La taille prévue est également renvoyée dans les en-têtes de la réponse Génération de sortie de tâche.

Dans le cas d'une tâche de récupération d'archives, en fonction de la plage d'octets que vous spécifiez, Amazon Glacier renvoie la somme de contrôle pour la partie des données. Afin de vous assurer que la partie téléchargée est correcte, calculez le total de contrôle sur le client et vérifiez que les valeurs correspondent et que la taille est conforme à ce que vous aviez prévu.

Un identifiant de tâche n'expire pas pendant au moins 24 heures après qu'Amazon Glacier ait terminé la tâche. En d'autres termes, vous pouvez télécharger le résultat de la tâche dans les 24 heures suivant la fin de la tâche par Amazon Glacier.

Requêtes

Syntaxe

Pour extraire une sortie de tâche, vous envoyez la demande HTTP `GET` à l'URI de la sortie `output` de la tâche spécifique.

```
GET /AccountId/vaults/VaultName/jobs/JobID/output HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
Range: ByteRangeToRetrieve
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise les en-têtes de demande suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Nom	Description	Obligatoire
`Range`	Plage d'octets à extraire de la sortie. Par exemple, si vous souhaitez télécharger les 1 048 576 premiers octets, spécifiez	Non

Nom	Description	Obligatoire
	la plage `bytes=0-1048575` . Pour plus d'informations, consultez le document Range Header Field Definition. La plage est relative par rapport à n'importe quelle plage spécifiée dans la demande de lancement de tâche. Par défaut, cette opération télécharge la totalité de la sortie. Si la sortie de la tâche est volumineuse, vous pouvez utiliser l'en-tête de demande `Range` pour extraire une partie de la sortie. Cela vous permet de télécharger la totalité de la sortie en blocs d'octets plus petits. Imaginons par exemple que vous souhaitiez télécharger une sortie de tâche de 1 Go et que vous décidiez d'effectuer ce téléchargement par blocs de 128 Mo via huit demandes de génération de sortie de tâche au total. Vous allez utiliser le processus suivant pour télécharger la sortie de la tâche : 1. Téléchargez un bloc de sortie de 128 Mo en spécifiant la plage d'octets appropriée à l'aide de l'en-tête `Range`. Vérifiez que les 128 Mo de données ont bien tous été reçus. 2. Outre les données, la réponse inclut un total de contrôle de la charge utile. Vous calculez le total de contrôle de la charge utile sur le client et vous le comparez au total de contrôle que vous avez reçu dans la réponse pour vous assurer que vous avez bien reçu toutes les données prévues. 3. Répétez les étapes 1 et 2 pour les huit blocs de données de sortie de 128 Mo, en spécifiant à chaque fois la plage d'octets appropriée. 4. Après le téléchargement de toutes les parties de la sortie de la tâche, vous obtenez une liste de huit valeurs de total de contrôle. Calculez le hachage d'arborescence de ces valeurs pour obtenir le total de contrôle de la totalité de la sortie. A l'aide de l'opération Description de la tâche (GET JobID), récupérez des informations sur la tâche dont est	

Nom	Description	Obligatoire
	extraite la sortie. La réponse inclut la somme de contrôle de l'ensemble de l'archive stockée dans Amazon Glacier. Comparez cette valeur au total de contrôle que vous avez calculé pour vous assurer que la totalité de l'archive a été téléchargée sans erreur. Type : Chaîne Par défaut : aucun Contraintes: Aucune	

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

Pour une demande d'extraction qui renvoie toutes les données d'une tâche, la réponse de la sortie de tâche renvoie le code de réponse `200 OK`. Lorsqu'un contenu partiel est demandé, par exemple, si vous avez spécifié l'en-tête `Range` dans la demande, le code de réponse `206 Partial Content` est renvoyé.

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: ContentType
Content-Length: Length
x-amz-sha256-tree-hash: ChecksumComputedByAmazonGlacier

[Body containing job output.]
```

En-têtes de réponse

En-tête	Description
`Content-Range`	La plage d'octets renvoyée par Amazon Glacier. Si seule une partie de la sortie est téléchargée, la réponse fournit la plage d'octets renvoyée par Amazon Glacier. Par exemple, `bytes 0-1048575/8388608` renvoie le premier bloc de 1 Mo des 8 Mo. Pour plus d'informations sur l'en-tête `Content-Range` , consultez le document Content-Range Header Field Definition. Type : Chaîne
`Content-Type`	L'en-tête Content-Type varie selon que la sortie de tâche est une archive ou un inventaire de coffre. • Pour des données archivées, l'en-tête Content-Type est `application/octet-stream` . • Pour un inventaire de coffre, si vous avez demandé le format CSV lors du lancement de la tâche, l'en-tête Content-Type est `text/csv`. Sinon, par défaut, l'inventaire de coffre est renvoyé au format JSON, et l'en-tête Content-Type est `application/json` . Type : Chaîne
`x-amz-sha256-tree-hash`	Somme de contrôle des données dans la réponse. Cet en-tête est renvoyé uniquement lors de l'extraction de la sortie pour une tâche d'extraction d'archive. En outre, cet en-tête s'affiche lorsque la plage de données extraite demandée dans la demande de lancement de tâche est alignée avec le hachage d'arborescence et que la plage à télécharg

En-tête	Description
	er demandée dans la demande de génération de sortie de tâche est également alignée avec le hachage d'arborescence. Pour plus d'informations sur les plages alignées avec le hachage d'arborescence, consultez la section Réception des totaux de contrôle lors du téléchargement de données. Par exemple, si dans votre demande de lancement de tâche, vous avez spécifié une plage à extraire qui est alignée avec le hachage d'arborescence (incluant la totalité de l'archive), vous recevrez le total de contrôle des données que vous téléchargez dans les conditions suivantes : • Vous obtenez la totalité de la plage des données extraites. • Vous demandez une plage d'octets des données extraites dont la taille est égale à un méga-octet (1 024 Ko) multiplié par la puissance 2 et qui commence et se termine sur un multiple de la taille de la plage demandée. Par exemple, si vous avez 3,1 Mo de données extraites et que vous spécifiez une plage à renvoyer qui commence à 1 Mo et se termine à 2 Mo, `x-amz-sha256-tree-hash` est renvoyé comme en-tête de réponse. • Vous demandez une plage de données extraites à renvoyer qui va jusqu'à la fin des données, et le début de la plage est un multiple de la taille de la plage à extraire qui est arrondi à la puissance deux suivante, mais qui n'est pas inférieur à un méga-octet (1 024 Ko). Par exemple, si vous avez 3,1 Mo de données extraites et que vous spécifiez une plage qui commence à 2 Mo et se termine à 3,1 Mo (fin des données), `x-amz-sha256-tree-hash` est renvoyé comme en-tête de réponse. Type : Chaîne

Corps de la réponse

Amazon Glacier renvoie le résultat de la tâche dans le corps de la réponse. Selon le type de tâche spécifié, la sortie représente un contenu d'archive ou un inventaire de coffre. Dans le cas d'un inventaire de coffre, par défaut, la liste de l'inventaire est renvoyée sous la forme du corps JSON suivant.

```
{
 "VaultARN": String,
 "InventoryDate": String,
 "ArchiveList": [
      {"ArchiveId": String,
       "ArchiveDescription": String,
       "CreationDate": String,
       "Size": Number,
       "SHA256TreeHash": String
      },
      ...
    ]
}
```

Si vous avez spécifié le format de sortie CSV lors du lancement de la tâche d'inventaire de coffre, l'inventaire de coffre est renvoyé au format CSV dans le corps. Le format CSV comporte cinq colonnes ArchiveId « », ArchiveDescription « », CreationDate « Size » et « SHA256 TreeHash » avec les mêmes définitions que les champs JSON correspondants.

> **Note**
>
> Dans le format CSV renvoyé, la totalité du contenu de certains champs renvoyés peut être placé entre guillemets. Les champs renvoyés qui contiennent une virgule ou des guillemets sont toujours placés entre guillemets. Par exemple, `my archive description,1` est renvoyé comme suit : `"my archive description,1"`. Les guillemets présents dans les champs renvoyés entre guillemets sont placés dans une séquence d'échappement et précédés d'une barre oblique inverse. Par exemple, `my archive description,1"2` est renvoyé comme suit : `"my archive description,1\"2"` et `my archive description,1\"2` est renvoyé comme suit : `"my archive description,1\\"2"`. La barre oblique inverse n'est pas placée dans une séquence d'échappement.

Le corps de réponse JSON contient les champs JSON suivants.

ArchiveDescription

Description d'une archive.

Type : chaîne

ArchiveId

ID d'une archive.

Type : chaîne

ArchiveList

Tableau de métadonnées d'archive. Chaque objet du tableau représente les métadonnées d'une archive contenue dans le coffre.

Type : tableau

CreationDate

Date et heure de création de l'archive, exprimées en heure UTC (temps universel coordonné).

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

InventoryDate

Date et heure UTC du dernier inventaire du coffre réalisé après que des modifications aient été apportées à celui-ci. Même si Amazon Glacier prépare un inventaire du coffre une fois par jour, la date d'inventaire n'est mise à jour que s'il y a eu des ajouts ou des suppressions d'archives dans le coffre depuis le dernier inventaire.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

SHA256TreeHash

Hachage d'arborescence de l'archive.

Type : chaîne

Taille

Taille de l'archive, exprimée en octets.

Type : nombre

VaultARN

Amazon Resource Name (ARN) à partir duquel l'extraction d'archive a été demandée.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant illustre une demande de tâche visant à extraire une archive.

Exemple 1 : Téléchargement d'une sortie

Cet exemple extrait les données préparées par Amazon Glacier en réponse à votre demande de lancement d'une tâche de récupération d'archives.

Exemple de requête

```
GET /-/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID/output
 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Voici un exemple de réponse à une tâche d'extraction d'archive. Notez que l'en-tête `Content-Type` est `application/octet-stream` et que l'en-tête `x-amz-sha256-tree-hash` est inclus dans la réponse, ce qui signifie que toutes les données de la tâche sont renvoyées.

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
x-amz-sha256-tree-hash:
 beb0fe31a1c7ca8c6c04d574ea906e3f97b31fdca7571defb5b44dca89b5af60
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/octet-stream
Content-Length: 1048576

[Archive data.]
```

Voici un exemple de réponse à une tâche d'extraction d'inventaire. Notez que l'en-tête `Content-Type` est `application/json`. Notez également que l'en-tête `x-amz-sha256-tree-hash` ne figure pas dans la réponse.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 906

{
 "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault",
 "InventoryDate": "2011-12-12T14:19:01Z",
 "ArchiveList": [
   {
     "ArchiveId": "DMTmICA2n5Tdqq5BV2z7og-
A20xnpAPKt3UXwWxdWsn_D6auTUrW6kwy5Qyj9xd1MCE1mBYvMQ63LWaT8yTMzMaCxB_9VBWrW4Jw4zsvg5kehAPDVKcppU
oA",
     "ArchiveDescription": "my archive1",
     "CreationDate": "2012-05-15T17:19:46.700Z",
     "Size": 2140123,
     "SHA256TreeHash":
 "6b9d4cf8697bd3af6aa1b590a0b27b337da5b18988dbcc619a3e608a554a1e62"
   },
   {
     "ArchiveId": "2lHzwhKhgF2JHyvCS-
ZRuF08IQLuyB4265Hs3AXj9MoAIhz7tbXAvcFeHusgU_hViO1WeCBe0N5lsYYHRyZ7rrmRkNRuYrXUs_sjl2K8ume_7mKO_
uHE1oHqaW9d37pabXrSA",
     "ArchiveDescription": "my archive2",
     "CreationDate": "2012-05-15T17:21:39.339Z",
     "Size": 2140123,
     "SHA256TreeHash":
 "7f2fe580edb35154041fa3d4b41dd6d3adaef0c85d2ff6309f1d4b520eeecda3"
   }
```

```
  ]
}
```

Exemple 2 : Téléchargement d'une sortie partielle

Cet exemple ne récupère qu'une partie de l'archive préparée par Amazon Glacier en réponse à votre demande initiale de récupération d'archives. La demande utilise l'en-tête `Range` facultatif pour extraire uniquement les 1 024 premiers octets.

Exemple de requête

```
GET /-/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID/output
 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Range: bytes=0-1023
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

La réponse positive suivante contient la réponse `206 Partial Content`. Dans ce cas, la réponse inclut également un `Content-Range` en-tête qui indique la plage d'octets renvoyée par Amazon Glacier.

```
HTTP/1.1 206 Partial Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Range: bytes 0-1023/8388608
Content-Type: application/octet-stream
Content-Length: 1024

[Archive data.]
```

Sections connexes

- Description de la tâche (GET JobID)

- Lancement d'une tâche (POST jobs)
- Identity and Access Management pour Amazon Glacier

Lancement d'une tâche (POST jobs)

Cette opération lance les types de tâches Amazon Glacier (Amazon Glacier) suivants :

- `archive-retrieval` : extrait une archive
- `inventory-retrieval` : inventorie un coffre

Rubriques

- Initialisation d'une tâche d'extraction d'archive ou d'inventaire de coffre
- Requêtes
- Réponses
- Exemples
- Sections connexes

Initialisation d'une tâche d'extraction d'archive ou d'inventaire de coffre

L'extraction d'une archive ou d'un inventaire de coffre sont des opérations asynchrones qui exigent que vous lanciez une tâche. Une fois lancée, la tâche ne peut pas être annulée. L'extraction est un processus en deux étapes :

1. Initiez une tâche d'extraction à l'aide de l'opération Lancement d'une tâche (POST jobs).

   > **Important**
   >
   > Une stratégie d'extraction de données peut entraîner l'échec de votre demande de lancement de tâche d'extraction et générer une `PolicyEnforcedException`. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier. Pour plus d'informations sur l'exception `PolicyEnforcedException`, consultez la section Réponses d'erreur.

2. Une fois la tâche terminée, téléchargez les octets à l'aide de l'opération Génération de sortie de tâche (GET output).

La demande d'extraction est exécutée de manière asynchrone. Lorsque vous lancez une tâche d'extraction, Amazon Glacier crée une tâche et renvoie un ID de tâche dans la réponse. Lorsque Amazon Glacier a terminé la tâche, vous pouvez obtenir le résultat de la tâche (archives ou données d'inventaire). Pour plus d'informations sur l'obtention de la sortie de la tâche, consultez l'opération Génération de sortie de tâche (GET output).

Pour recevoir la sortie de la tâche, la tâche doit être terminée. Pour déterminer si une tâche est terminée, vous disposez des options suivantes :

- Utiliser une notification Amazon SNS : vous pouvez spécifier une rubrique Amazon SNS sur laquelle Amazon Glacier peut publier une notification une fois la tâche terminée. Vous pouvez spécifier une rubrique SNS par demande de tâche. La notification n'est envoyée qu'une fois qu'Amazon Glacier a terminé le travail. En plus de spécifier une rubrique SNS par demande de tâche, vous pouvez configurer des notifications de coffre pour un coffre afin que les notifications de tâche soient envoyées pour toutes les extractions. Pour de plus amples informations, veuillez consulter Définition d'une configuration de notification de coffre (PUT notification-configuration).
- Obtenir les détails de la tâche – Vous pouvez effectuer une demande Description de la tâche (GET JobID) pour obtenir les informations concernant le statut de la tâche pendant que celle-ci est en cours. Cependant, il est préférable d'utiliser une notification Amazon SNS pour déterminer à quel moment une tâche se termine.

> **Note**
>
> Les informations que vous obtenez via une notification sont identiques à celles que vous obtenez en appelant Description de la tâche (GET JobID).

Si, pour un événement spécifique, vous ajoutez à la fois la configuration des notifications dans le coffre et que vous spécifiez une rubrique SNS dans votre demande de travail d'initiation, Amazon Glacier envoie les deux notifications. Pour de plus amples informations, veuillez consulter Définition d'une configuration de notification de coffre (PUT notification-configuration).

L'inventaire du coffre

Amazon Glacier met à jour l'inventaire d'un coffre-fort environ une fois par jour, à compter du jour où vous chargez une archive dans le coffre-fort pour la première fois. Si aucun ajout ou aucune

suppression d'archive n'a été effectué dans le coffre depuis le dernier inventaire, la date de l'inventaire n'est pas mise à jour. Lorsque vous lancez une tâche pour un inventaire de coffre-fort, Amazon Glacier renvoie le dernier inventaire généré, qui est un point-in-time instantané et non des données en temps réel.

Une fois qu'Amazon Glacier a créé le premier inventaire pour le coffre, il faut généralement une demi-journée, voire un jour, avant que cet inventaire ne soit disponible pour extraction.

Vous ne trouverez peut-être pas utile d'extraire un inventaire de coffre pour chaque chargement d'archive. Supposons toutefois que vous mainteniez une base de données côté client associant des métadonnées relatives aux archives que vous chargez sur Amazon Glacier. Ensuite, vous trouverez peut-être l'inventaire de coffre très utile pour rapprocher, au besoin, les informations contenues dans votre base de données avec l'inventaire de coffre réel. Pour plus d'informations sur les champs de données renvoyés dans la sortie d'une tâche d'inventaire, consultez Corps de la réponse.

Extraction d'un inventaire par plage

Vous pouvez limiter le nombre d'éléments dans l'inventaire en filtrant par date de création de l'archive ou en définissant une limite.

Filtrage par la date de création de l'archive

Vous pouvez extraire des éléments d'inventaire pour des archives créées entre la valeur de `StartDate` et la valeur de `EndDate` en spécifiant les valeurs de ces paramètres dans la demande Lancement d'une tâche. Les archives créées à la ou après la `StartDate` et avant la `EndDate` sont renvoyées. Si vous n'indiquez que le paramètre `StartDate` sans le paramètre `EndDate`, vous extrayez l'inventaire pour toutes les archives créées à la ou après la `StartDate`. Si vous n'indiquez que le paramètre `EndDate` sans le paramètre `StartDate`, vous recevez l'inventaire correspondant à toutes les archives créées avant la `EndDate`.

Limitation des éléments de l'inventaire par extraction

Vous pouvez limiter le nombre d'éléments d'inventaire renvoyés en définissant le paramètre `Limit` dans la demande Lancement d'une tâche. La sortie de la tâche d'inventaire contient tous les éléments d'inventaire jusqu'à la valeur `Limit` spécifiée. Si plusieurs éléments d'inventaire sont disponibles, le résultat est paginé. Une fois qu'une tâche est terminée, vous pouvez utiliser l'opération Description de la tâche (GET JobID) pour obtenir un marqueur que vous utiliserez dans une demande Lancement d'une tâche ultérieure. Le marqueur indique le point de départ de l'extraction de la prochaine série d'éléments d'inventaire. Vous pouvez parcourir l'ensemble de votre inventaire en faisant des demandes de Lancement d'une tâche avec le marqueur de la sortie Description de la

tâche précédente. Vous procédez ainsi jusqu'à ce que vous obteniez un marqueur de Description de la tâche renvoyant la valeur null, ce qui indique qu'il ne reste plus d'élément d'inventaire disponible.

Vous pouvez utiliser le paramètre `Limit` avec les paramètres de plage de dates.

Extraction d'une plage d'archive

Vous pouvez lancer l'extraction d'une archive pour l'archive entière ou une plage de l'archive. Pour extraire une plage d'archive, vous spécifiez une plage d'octets à renvoyer ou l'archive entière. Si ce champ est spécifié, la plage d'octets doit être alignée en termes de méga-octets (1024*1024). Autrement dit, la valeur de début de la plage doit être divisible par 1 Mo et la valeur de fin de la plage plus 1 doit être divisible par 1 Mo ou égale à la fin de l'archive. Si l'extraction de la plage de l'archive n'est pas alignée en termes de méga-octets, cette opération renvoie une réponse `400`. En outre, afin de garantir que vous obteniez les valeurs de total de contrôle pour les données que vous téléchargez à l'aide de la demande Génération de sortie de tâche (Génération de sortie de tâche (GET output)), la plage doit être alignée avec le hachage d'arborescence. Pour plus d'informations sur les plages alignées avec le hachage d'arborescence, consultez la section Réception des totaux de contrôle lors du téléchargement de données.

Niveau Rapide, Standard et Groupé

Lorsque vous lancez une tâche d'extraction d'archive, vous pouvez spécifier l'une des options suivantes dans le champ `Tier` du corps de la demande :

- **`Expedited`** : cette option vous permet d'accéder rapidement à vos données lorsque des demandes urgentes occasionnelles de restauration d'archives sont nécessaires. Pour toutes les archives à l'exception des plus volumineuses (plus de 250 Mo), les données auxquelles vous accédez du niveau Rapide sont généralement disponibles en 1 à 5 minutes.
- **`Standard`** : cette option vous permet d'accéder à vos archives en quelques heures. L'accès aux données à l'aide du niveau Standard est généralement rendu disponible en 3 à 5 heures. Cette option par défaut correspond aux demandes de tâches qui ne spécifient pas l'option de niveau.
- **`Bulk`**— Le stockage en masse est le niveau le moins coûteux pour Amazon Glacier. Il vous permet de récupérer de grandes quantités, voire des pétaoctets, de données à moindre coût par jour. L'accès aux données à l'aide du niveau Groupé est généralement rendu disponible en 5 à 12 heures.

Pour plus d'informations sur les récupérations rapides et en bloc, consultez Récupération des archives Amazon Glacier.

Requêtes

Pour lancer une tâche, vous utilisez la méthode HTTP `POST` et limitez la portée de la demande à la sous-ressource `jobs` du coffre. Vous spécifiez les détails de la demande de tâche dans le document JSON de votre demande. Le type de tâche est spécifié dans le champ `Type`. Vous pouvez éventuellement spécifier un `SNSTopic` champ pour indiquer une rubrique Amazon SNS sur laquelle Amazon Glacier peut publier une notification une fois la tâche terminée.

> **Note**
>
> Pour publier une notification dans Amazon SNS, vous devez créer la rubrique vous-même si elle n'existe pas déjà. Amazon Glacier ne crée pas le sujet pour vous. Le sujet doit être autorisé à recevoir des publications depuis un coffre-fort Amazon Glacier. Amazon Glacier ne vérifie pas si le coffre-fort est autorisé à publier sur le sujet. Si les autorisations ne sont pas configurées de façon appropriée, vous risquez de ne pas recevoir la notification même lorsque la tâche est terminée.

Syntaxe

Voici la syntaxe de la requête pour l'initiation d'une tâche.

```
POST /AccountId/vaults/VaultName/jobs HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01

{
   "jobParameters": {
      "ArchiveId": "string",
      "Description": "string",
      "Format": "string",
      "InventoryRetrievalParameters": {
         "EndDate": "string",
         "Limit": "string",
         "Marker": "string",
         "StartDate": "string"
      },
      "OutputLocation": {
         "S3": {
```

```
            "AccessControlList": [
               {
                  "Grantee": {
                     "DisplayName": "string",
                     "EmailAddress": "string",
                     "ID": "string",
                     "Type": "string",
                     "URI": "string"
                  },
                  "Permission": "string"
               }
            ],
            "BucketName": "string",
            "CannedACL": "string",
            "Encryption": {
               "EncryptionType": "string",
               "KMSContext": "string",
               "KMSKeyId": "string"
            },
            "Prefix": "string",
            "StorageClass": "string",
            "Tagging": {
               "string" : "string"
            },
            "UserMetadata": {
               "string" : "string"
            }
         }
      },
      "RetrievalByteRange": "string",
      "SelectParameters": {
         "Expression": "string",
         "ExpressionType": "string",
         "InputSerialization": {
            "csv": {
               "Comments": "string",
               "FieldDelimiter": "string",
               "FileHeaderInfo": "string",
               "QuoteCharacter": "string",
               "QuoteEscapeCharacter": "string",
               "RecordDelimiter": "string"
            }
         },
         "OutputSerialization": {
```

```
            "csv": {
               "FieldDelimiter": "string",
               "QuoteCharacter": "string",
               "QuoteEscapeCharacter": "string",
               "QuoteFields": "string",
               "RecordDelimiter": "string"
            }
         }
      },
      "SNSTopic": "string",
      "Tier": "string",
      "Type": "string"
   }
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Corps de la requête

La demande accepte les données suivantes au format JSON dans le corps de la demande.

jobParameters

Fournit des options de spécification des informations d'une tâche.

Type : objet jobParameters

Obligatoire : oui

Réponses

Amazon Glacier crée l'emploi. Dans la réponse, il retourne l'URI de la tâche.

Syntaxe

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: location
x-amz-job-id: jobId
x-amz-job-output-path: jobOutputPath
```

En-têtes de réponse

En-tête	Description
`Location`	Le chemin d'accès par URI relatif de la tâche. Vous pouvez utiliser ce chemin d'accès par URI pour trouver l'état de la tâche. Pour de plus amples informations, veuillez consulter Description de la tâche (GET JobID). Type : Chaîne Par défaut : aucun
`x-amz-job-id`	L'ID de la tâche. Cette valeur fait également partie de l'en-tête `Location`. Type : Chaîne Par défaut : aucun
`x-amz-job-output-path`	Chemin vers l'emplacement auquel les résultats de tâches de sélection sont stockés. Type : Chaîne Par défaut : aucun

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Cette opération inclut l'erreur ou les erreurs suivantes, en plus des erreurs possibles communes à toutes les opérations Amazon Glacier. Pour plus d'informations sur les erreurs Amazon Glacier et obtenir la liste des codes d'erreur, consultezRéponses d’erreur.

Code	Description	HTTP Status Code	Type
`InsufficientCapacityException`	Retournée s'il existe une capacité insuffisante pour traiter cette demande rapide. Cette erreur s'applique uniquement aux récupérations rapides et non aux récupérations standard ou en bloc.	`503 Service Unavailable`	Serveur

Exemples

Exemple de demande : Lancement d'une tâche d'extraction d'archive

```
POST /-/vaults/examplevault/jobs HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
  "Type": "archive-retrieval",
  "ArchiveId": "NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
  "Description": "My archive description",
  "SNSTopic": "arn:aws:sns:us-west-2:111111111111:Glacier-ArchiveRetrieval-topic-
Example",
  "Tier" : "Bulk"
}
```

Voici un exemple de corps d'une demande qui spécifie une plage de l'archive à extraire à l'aide du champ `RetrievalByteRange`.

```
{
  "Type": "archive-retrieval",
  "ArchiveId": "NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-
TjhqG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L26Omw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchi
  "Description": "My archive description",
  "RetrievalByteRange": "2097152-4194303",
  "SNSTopic": "arn:aws:sns:us-west-2:111111111111:Glacier-ArchiveRetrieval-topic-
Example",
  "Tier" : "Bulk"
}
```

Exemple de réponse

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Location: /111122223333/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
x-amz-job-id: HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
```

Exemple de demande : Lancement d'une tâche d'extraction d'inventaire

La demande suivante lance une tâche d'extraction d'inventaire pour obtenir une liste d'archives du coffre `examplevault`. Le paramètre `Format` défini sur `CSV` dans le corps de la demande indique que l'inventaire est renvoyé au format CSV.

```
POST /-/vaults/examplevault/jobs HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Content-Type: application/x-www-form-urlencoded
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
  "Type": "inventory-retrieval",
```

```
  "Description": "My inventory job",
  "Format": "CSV",
  "SNSTopic": "arn:aws:sns:us-west-2:111111111111:Glacier-InventoryRetrieval-topic-
Example"
}
```

Exemple de réponse

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Location: /111122223333/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
x-amz-job-id: HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
```

Exemples de demandes : Initier une tâche d'extraction d'inventaire à l'aide d'un filtrage par date avec une limité définie, et demande consécutive d'extraction de la page suivante des éléments de l'inventaire.

La demande suivante lance une tâche d'extraction d'inventaire en utilisant le filtrage de dates et en définissant une limite.

```
{
    "ArchiveId": null,
    "Description": null,
    "Format": "CSV",
    "RetrievalByteRange": null,
    "SNSTopic": null,
    "Type": "inventory-retrieval",
    "InventoryRetrievalParameters": {
        "StartDate": "2013-12-04T21:25:42Z",
        "EndDate": "2013-12-05T21:25:42Z",
        "Limit" : "10000"
    },
}
```

La demande suivante est un exemple de demande suivante d'extraction de la page suivante des éléments de l'inventaire à l'aide d'un marqueur obtenu de Description de la tâche (GET JobID).

```
{
```

```
    "ArchiveId": null,
    "Description": null,
    "Format": "CSV",
    "RetrievalByteRange": null,
    "SNSTopic": null,
    "Type": "inventory-retrieval",
    "InventoryRetrievalParameters": {
        "StartDate": "2013-12-04T21:25:42Z",
        "EndDate": "2013-12-05T21:25:42Z",
        "Limit": "10000",
        "Marker":
 "vyS0t2jHQe5qbcDggIeD50chS1SXwYMrkVKo0KHiTUjEYxBGCqRLKaiySzdN7QXGVVV5XZpNVG67pCZ_uykQXFMLaxOSu
    },
}
```

Exemple de réponse

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Location: /111122223333/vaults/examplevault/jobs/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
x-amz-job-id: HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID
x-amz-job-output-path: test/HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID/
```

Sections connexes

- Description de la tâche (GET JobID)
- Génération de sortie de tâche (GET output)
- Identity and Access Management pour Amazon Glacier

Affichage des tâches (GET jobs)

Description

Cette opération affiche la liste des tâches relatives à un coffre, y compris les tâches en cours et les tâches récemment terminées.

> **Note**
>
> Amazon Glacier (Amazon Glacier) conserve les tâches récemment terminées pendant un certain temps avant de les supprimer ; toutefois, il finit par supprimer les tâches terminées. La sortie des tâches terminées peut être extraite. Les tâches terminées étant conservées pendant un certain temps, vous pouvez ainsi obtenir les sorties correspondantes si vous manquez la notification d'achèvement de ces tâches ou si votre première tentative de téléchargement de ces tâches échoue. Par exemple, imaginons que vous démarriez une tâche d'extraction d'archive afin de télécharger une archive. Une fois la tâche terminée, vous commencez à télécharger l'archive, mais une erreur réseau se produit. Dans ce scénario, vous pouvez réessayer de télécharger l'archive tant que la tâche existe.

L'opération `List Jobs` prend en charge la pagination. Vous devez toujours vérifier le champ `Marker` de la réponse. S'il n'y a pas d'autres tâches à afficher, le champ `Marker` est défini sur `null`. S'il y a d'autres tâches à afficher, le champ `Marker` est défini sur une valeur non null que vous pouvez utiliser afin de poursuivre la pagination de la liste. Pour renvoyer une liste de tâches à partir d'une tâche donnée, définissez le paramètre de demande `marker` sur la valeur `Marker` pour la tâche que vous avez obtenue à partir d'une demande `List Jobs` précédente.

Vous pouvez définir une limite maximale pour le nombre de tâches renvoyées dans la réponse en spécifiant le paramètre `limit` dans la demande. La limite par défaut est de 50. Il peut arriver que le nombre de tâches renvoyées soit inférieur à la limite, mais il ne sera jamais supérieur à cette valeur.

En outre, vous pouvez filtrer la liste des tâches renvoyées en spécifiant le paramètre facultatif `statuscode` ou `completed`, ou les deux. Le paramètre `statuscode` vous permet de demander à ce que seules les tâches qui correspondent à l'état `InProgress`, `Succeeded` ou `Failed` soient retournées. Le paramètre `completed` vous permet de demander à ce que seules les tâches terminées (`true`) ou non terminées (`false`) soient retournées.

Requêtes

Syntaxe

Pour renvoyer une liste de tâches de tous types, envoyez une demande `GET` à l'URI de la sous-ressource `jobs` du coffre.

```
GET /AccountId/vaults/VaultName/jobs HTTP/1.1
```

```
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID du compte propriétaire du coffre-fort. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous utilisez un ID de compte, évitez d'y inclure des traits d'union (« - »).

Paramètres de demande

Nom	Description	Obligatoire
`completed`	L'état des tâches à renvoyer. Vous pouvez spécifier `true` ou `false`. Type : Boolean Contraintes: Aucune	Non
`limit`	Le nombre maximum de tâches à renvoyer. La limite par défaut est de 50. Il peut arriver que le nombre de tâches renvoyées soit inférieur à la limite spécifiée, mais il ne sera jamais supérieur à cette valeur. Type : Chaîne Contraintes : la valeur d'entier minimale est 1. La valeur d'entier maximale est 50.	Non
`marker`		Non

Nom	Description	Obligatoire
	Chaîne opaque utilisée pour la pagination afin de spécifier la tâche à partir de laquelle l'affichage de la liste de tâches doit débuter. Vous obtenez la valeur du paramètre `marker` à partir d'une réponse à une demande `List Jobs` précédente. Vous ne devez inclure la valeur `marker` que si vous poursuivez la pagination de résultats commencée dans une demande `List Jobs` précédente. Type : Chaîne Contraintes: Aucune	
`statuscode`	Le type de l'état de tâche à renvoyer. Type : Chaîne Contraintes : l'une des valeurs suivantes : `InProgress` , `Succeeded` ou`Failed`.	Non

En-têtes de demande

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
```

```
Content-Type: application/json
Content-Length: Length

{
    "JobList": [
        {
            "Action": "string",
            "ArchiveId": "string",
            "ArchiveSHA256TreeHash": "string",
            "ArchiveSizeInBytes": number,
            "Completed": boolean,
            "CompletionDate": "string",
            "CreationDate": "string",
            "InventoryRetrievalParameters": {
                "EndDate": "string",
                "Format": "string",
                "Limit": "string",
                "Marker": "string",
                "StartDate": "string"
            },
            "InventorySizeInBytes": number,
            "JobDescription": "string",
            "JobId": "string",
            "JobOutputPath": "string",
            "OutputLocation": {
                "S3": {
                    "AccessControlList": [
                        {
                            "Grantee": {
                                "DisplayName": "string",
                                "EmailAddress": "string",
                                "ID": "string",
                                "Type": "string",
                                "URI": "string"
                            },
                            "Permission": "string"
                        }
                    ],
                    "BucketName": "string",
                    "CannedACL": "string",
                    "Encryption": {
                        "EncryptionType": "string",
                        "KMSContext": "string",
                        "KMSKeyId": "string"
```

```
                    },
                    "Prefix": "string",
                    "StorageClass": "string",
                    "Tagging": {
                        "string": "string"
                    },
                    "UserMetadata": {
                        "string": "string"
                    }
                }
            },
            "RetrievalByteRange": "string",
            "SelectParameters": {
                "Expression": "string",
                "ExpressionType": "string",
                "InputSerialization": {
                    "csv": {
                        "Comments": "string",
                        "FieldDelimiter": "string",
                        "FileHeaderInfo": "string",
                        "QuoteCharacter": "string",
                        "QuoteEscapeCharacter": "string",
                        "RecordDelimiter": "string"
                    }
                },
                "OutputSerialization": {
                    "csv": {
                        "FieldDelimiter": "string",
                        "QuoteCharacter": "string",
                        "QuoteEscapeCharacter": "string",
                        "QuoteFields": "string",
                        "RecordDelimiter": "string"
                    }
                }
            },
            "SHA256TreeHash": "string",
            "SNSTopic": "string",
            "StatusCode": "string",
            "StatusMessage": "string",
            "Tier": "string",
            "VaultARN": "string"
        }
    ],
    "Marker": "string"
```

```
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

JobList

Liste d'objets de tâche. Chaque objet de tâche contient des métadonnées décrivant la tâche.

Type : tableau d'objets GlacierJobDescription

Marker

Chaîne opaque représentant l'endroit à partir duquel poursuivre la pagination des résultats. Vous utilisez la valeur `marker` dans une nouvelle demande `List Jobs` afin d'obtenir davantage de tâches dans la liste. S'il n'y a pas d'autres tâches à afficher, cette valeur est `null`.

Type : chaîne

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Les exemples suivants décrivent comment renvoyer des informations relatives à des tâches de coffre. Le premier exemple renvoie une liste comprenant deux tâches et le deuxième exemple renvoie un sous-ensemble de tâches.

Exemple : Renvoi de toutes les tâches

Exemple de requête

La demande `GET` suivante renvoie les tâches associées à un coffre.

```
GET /-/vaults/examplevault/jobs  HTTP/1.1
```

```
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

La réponse suivante inclut une tâche d'extraction d'archive et une tâche d'extraction d'inventaire qui contient un marqueur permettant de poursuivre la pagination de l'extraction d'inventaire de coffre. La réponse illustre également le champ `Marker` défini sur `null`, ce qui signifie qu'il n'y a pas d'autres tâches à afficher.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 1444

{
  "JobList": [
    {
      "Action": "ArchiveRetrieval",
      "ArchiveId": "BDfaUQul0dVzYwAMr8YSa_6_8abbhZq-
i1oT69g8ByClfJyBgAGBkWl2QbF5os851P7Y7KdZDOHWJIn4rh1ZHaOYD3MgFhK_g0oDPesW34uHQoVGwoIqubf6BgUEfQm
      "ArchiveSizeInBytes": 1048576,
      "ArchiveSHA256TreeHash":
 "25499381569ab2f85e1fd0eb93c5406a178ab77c5933056eb5d6e7d4adda609b",
      "Completed": true,
      "CompletionDate": "2012-05-01T00:00:09.304Z",
      "CreationDate": "2012-05-01T00:00:06.663Z",
      "InventorySizeInBytes": null,
      "JobDescription": null,
      "JobId": "hDe9t9DTHXqFw8sBGpLQQOmIM0-
JrGtu1O_YFKLnzQ64548qJc667BRWTwBLZC76Ygy1jHYruqXkdcAhRsh0hYv4eVRU",
      "RetrievalByteRange": "0-1048575",
      "SHA256TreeHash":
 "25499381569ab2f85e1fd0eb93c5406a178ab77c5933056eb5d6e7d4adda609b",
      "SNSTopic": null,
      "StatusCode": "Succeeded",
      "StatusMessage": "Succeeded",
      "Tier": "Bulk",
```

```
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    },
    {
      "Action": "InventoryRetrieval",
      "ArchiveId": null,
      "ArchiveSizeInBytes": null,
      "ArchiveSHA256TreeHash": null,
      "Completed": true,
      "CompletionDate": "2013-05-11T00:25:18.831Z",
      "CreationDate": "2013-05-11T00:25:14.981Z",
      "InventorySizeInBytes": 1988,
      "JobDescription": null,
      "JobId":
 "2cvVOnBL36btzyP3pobwIceiaJebM1bx9vZOOUtmNAr0KaVZ4WkWgVjiPldJ73VU7imlm0pnZriBVBebnqaAcirZq_C5"
      "RetrievalByteRange": null,
      "SHA256TreeHash": null,
      "SNSTopic": null,
      "StatusCode": "Succeeded",
      "StatusMessage": "Succeeded",
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
      "InventoryRetrievalParameters": {
          "StartDate": "2013-11-12T13:43:12Z",
          "EndDate": "2013-11-20T08:12:45Z",
          "Limit": "120000",
          "Format": "JSON",
          "Marker":
 "vyS0t2jHQe5qbcDggIeD50chS1SXwYMrkVKo0KHiTUjEYxBGCqRLKaiySzdN7QXGVVV5XZpNVG67pCZ_uykQXFMLaxOSu
    }
  ],
  "Marker": null
}
```

Exemple : Renvoi d'une liste partielle de tâches

Exemple de requête

La demande GET suivante renvoie la tâche spécifiée par le paramètre `marker`. Définir le paramètre `limit` avec la valeur permet de spécifier que deux tâches au maximum peuvent être renvoyées.2

```
GET /-/vaults/examplevault/jobs?marker=HkF9p6o7yjhFx-
K3CGl6fuSm6VzW9T7esGQfco8nUXVYwS0jlb5gq1JZ55yHgt5vP54ZShjoQzQVVh7vEXAMPLEjobID&limit=2
 HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
```

```
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

La réponse suivante illustre deux tâches renvoyées et le champ `Marker` défini sur une valeur non null qui peut être utilisée afin de poursuivre la pagination de la liste de tâches.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 1744

{
  "JobList": [
    {
      "Action": "ArchiveRetrieval",
      "ArchiveId": "58-3KpZfcMPUznvMZNPaKyJx9wODCsWTnqcjtx2CjKZ6b-
XgxEuA8yvZOYTPQfd7gWR4GRm2XR08gcnWbLV4VPV_kDWtZJKi0TFhKKVPzwrZnA4-
FXuIBfViYUIVveeiBE51FO4bvg",
      "ArchiveSizeInBytes": 8388608,
      "ArchiveSHA256TreeHash":
 "106086b256ddf0fedf3d9e72f461d5983a2566247ebe7e1949246bc61359b4f4",
      "Completed": true,
      "CompletionDate": "2012-05-01T00:25:20.043Z",
      "CreationDate": "2012-05-01T00:25:16.344Z",
      "InventorySizeInBytes": null,
      "JobDescription": "aaabbbccc",
      "JobId": "s4MvaNHIh6mOa1f8iY4ioG2921SDPihXxh3Kv0FBX-
JbNPctpRvE4c2_BifuhdGLqEhGBNGeB6Ub-JMunR9JoVa8y1hQ",
      "RetrievalByteRange": "0-8388607",
      "SHA256TreeHash":
 "106086b256ddf0fedf3d9e72f461d5983a2566247ebe7e1949246bc61359b4f4",
      "SNSTopic": null,
      "StatusCode": "Succeeded",
      "StatusMessage": "Succeeded",
      "Tier": "Bulk",
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    },
```

```
    {
      "Action": "ArchiveRetrieval",
      "ArchiveId": "2NVGpf83U6qB9M2u-
Ihh61yoFLRDEoh7YLZWKBn80A2i1xG8uieBwGjAr4RkzOHA0E07ZjtI267R03Z-6Hxd8pyGQkBdciCSH1-
Lw63Kx9qKpZbPCdU0uTW_WAdwF6lR6w8iSyKdvw",
      "ArchiveSizeInBytes": 1048576,
      "ArchiveSHA256TreeHash":
 "3d2ae052b2978727e0c51c0a5e32961c6a56650d1f2e4ceccab6472a5ed4a0",
      "Completed": true,
      "CompletionDate": "2012-05-01T16:59:48.444Z",
      "CreationDate": "2012-05-01T16:59:42.977Z",
      "InventorySizeInBytes": null,
      "JobDescription": "aaabbbccc",
      "JobId":
 "CQ_tf6fOR4jrJCL61Mfk6VM03oY8lmnWK93KK4gLig1UPAbZiN3UV4G_5nq4AfmJHQ_dOMLOX5k8ItFv0wCPN0oaz5dG"
      "RetrievalByteRange": "0-1048575",
      "SHA256TreeHash":
 "3d2ae052b2978727e0c51c0a5e32961c6a56650d1f2e4ceccab6472a5ed4a0",
      "SNSTopic": null,
      "StatusCode": "Succeeded",
      "StatusMessage": "Succeeded",
      "Tier": "Standard",
      "VaultARN": "arn:aws:glacier:us-west-2:012345678901:vaults/examplevault"
    }
  ],
  "Marker":
 "CQ_tf6fOR4jrJCL61Mfk6VM03oY8lmnWK93KK4gLig1UPAbZiN3UV4G_5nq4AfmJHQ_dOMLOX5k8ItFv0wCPN0oaz5dG"
}
```


Sections connexes

- Description de la tâche (GET JobID)
- Identity and Access Management pour Amazon Glacier

Types de données utilisés dans les opérations de tâche

Les types de données suivants sont utilisés pour les opérations de travail dans Amazon Glacier.

Rubriques

- CSVInput

- CSVOutput
- Chiffrement
- GlacierJobDescription
- Octroi
- Bénéficiaire
- InputSerialization
- InventoryRetrievalJobInput
- jobParameters
- OutputLocation
- OutputSerialization
- Emplacement S3
- SelectParameters

CSVInput

Contient des informations sur le fichier CSV (valeurs séparées par des virgules).

Table des matières

Commentaires

Caractère nique utilisé pour indiquer qu'une ligne doit être ignorée lorsque le caractère est présent au début de cette ligne.

Type : chaîne

Obligatoire : non

FieldDelimiter

Caractère unique utilisé pour séparer des champs individuels les uns des autres au sein d'un enregistrement. Le caractère doit être un `\n`, un `\r`, ou un caractère ASCII compris dans la plage 32-126. La valeur par défaut est une virgule (`,`).

Type : chaîne

Par défaut : ,

Obligatoire : non

FileHeaderInfo

Valeur décrivant quoi faire avec la première ligne de l'entrée.

Type : chaîne

Valeurs Valides : `Use` | `Ignore` | `None`

Obligatoire : non

QuoteCharacter

Caractère unique utilisé comme caractère d'échappement lorsque le délimiteur de champ fait partie d'une valeur.

Type : chaîne

Obligatoire : non

QuoteEscapeCharacter

Caractère unique utilisé pour l'échappement du guillemet à l'intérieur d'une valeur déjà échappée.

Type : chaîne

Obligatoire : non

RecordDelimiter

Caractère unique utilisé pour séparer des enregistrements individuels les uns des autres.

Type : chaîne

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

CSVOutput

Contient des informations sur le format CSV (valeurs séparées par des virgules) dans lequel les résultats des tâches sont stockés.

Table des matières

FieldDelimiter

Caractère unique utilisé pour séparer des champs individuels les uns des autres au sein d'un enregistrement.

Type : chaîne

Obligatoire : non

QuoteCharacter

Caractère unique utilisé comme caractère d'échappement lorsque le délimiteur de champ fait partie d'une valeur.

Type : chaîne

Obligatoire : non

QuoteEscapeCharacter

Caractère unique utilisé pour l'échappement du guillemet à l'intérieur d'une valeur déjà échappée.

Type : chaîne

Obligatoire : non

QuoteFields

Valeur qui indique si tous les champs de sortie doivent être entourés de guillemets.

Valeurs valides : `ALWAYS` | `ASNEEDED`

Type : chaîne

Obligatoire : non

RecordDelimiter

Caractère unique utilisé pour séparer des enregistrements individuels les uns des autres.

Type : chaîne

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

Chiffrement

Contient des informations sur le chiffrement utilisé pour stocker les résultats des tâches dans Amazon S3.

Table des matières

Chiffrement

Algorithme de chiffrement côté serveur utilisé lors du stockage des résultats des tâches dans Amazon S3. La valeur par défaut est sans chiffrement.

Type : chaîne

Valeurs valides : `aws:kms` | `AES256`

Obligatoire : non

KMSContext

Facultatif. Si le type de chiffrement est `aws:kms`, vous pouvez utiliser cette valeur pour spécifier le contexte de chiffrement des résultats des tâches.

Type : chaîne

Obligatoire : non

KMSKeyId

L'ID de clé AWS Key Management Service (AWS KMS) à utiliser pour le chiffrement des objets.

Type : chaîne

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

GlacierJobDescription

Contient la description d'une tâche Amazon Glacier (Amazon Glacier).

Table des matières

Action

Le type de tâche. Il s'agit de `ArchiveRetrieval`, `InventoryRetrieval` ou `Select`.

Type : chaîne

ArchiveId

ID d'archive demandé pour une tâche de sélection ou d'extraction d'archive. Sinon, ce champ est `null`.

Type : chaîne

ArchivageSHA256TreeHash

Le hachage SHA256 arborescent de l'ensemble de l'archive pour une extraction d'archives. Pour les tâches d'extraction d'inventaire, ce champ est `null`.

Type : chaîne

ArchiveSizeInBytes

Pour une tâche `ArchiveRetrieval`, il s'agit de la taille en octets de l'archive demandée au téléchargement. Pour la tâche `InventoryRetrieval`, la valeur est `null`.

Type : nombre

Terminé

`true` si la tâche est terminée, sinon, `false`.

Type : valeur booléenne

CompletionDate

Date à laquelle la tâche s'est terminée.

La date UTC lors de la fin de la demande de tâche. Lorsque la tâche est en cours, cette valeur est null.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

CreationDate

La date UTC du début de la tâche.

Type : représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

InventoryRetrievalParameters

Paramètres d'entrée utilisés pour l'extraction d'un inventaire par plage.

Type : objet InventoryRetrievalJobInput

InventorySizeInBytes

Pour une tâche `InventoryRetrieval`, il s'agit de la taille en octets de l'inventaire dont le téléchargement est demandé. Pour la tâche `ArchiveRetrieval` ou `Select`, la valeur est `null`.

Type : nombre

JobDescription

Description de tâche que vous avez fournie lorsque vous avez lancé la tâche.

Type : chaîne

JobId

ID identifiant la tâche dans Amazon Glacier.

Type : chaîne

JobOutputPath

Contient l'emplacement de la sortie de la tâche.

Type : chaîne

OutputLocation

Objet qui contient des informations sur l'emplacement auquel les erreurs et résultats de tâches de sélection sont stockés.

Type : objet OutputLocation

RetrievalByteRange

La plage d'octets récupérée pour les tâches de récupération d'archives sous la forme « *StartByteValue* - »*EndByteValue*. Si aucune plage n'a été spécifiée lors de la récupération de l'archive, l'archive entière est récupérée et StartByteValuevaut 0 et est EndByteValueégale à la taille de l'archive moins 1. Pour les tâches d'extraction d'inventaire, ce champ est `null`.

Type : chaîne

SelectParameters

Objet qui contient les informations sur les paramètres utilisés pour une tâche de sélection.

Type : objet SelectParameters

SHA256TreeHash

La valeur de hachage de l' SHA256 arbre pour la plage demandée d'une archive. Si la demande Lancement d'une tâche (POST jobs) définie pour une archive spécifiait une plage alignée avec le hachage d'arborescence, ce champ renvoie une valeur. Pour plus d'informations sur l'alignement avec le hachage d'arborescence dans le cas d'extractions par plage d'archive, consultez Réception des totaux de contrôle lors du téléchargement de données.

Lorsque la totalité de l'archive est extraite, cette valeur est identique à celle de `ArchiveSHA256TreeHash`.

La valeur de ce champ est `null` dans les cas suivants :

- Tâches d'extraction d'archive qui spécifient une plage qui n'est pas alignée avec le hachage d'arborescence.
- Tâches d'archivage qui spécifient une plage qui correspond à la totalité de l'archive et dont l'état est `InProgress`.
- Tâches d'inventaire.
- Tâches de sélection.

Type : chaîne

SNSTopic

Amazon Resource Name (ARN) représentant une rubrique Amazon SNS à laquelle la notification d'achèvement ou d'échec de la tâche est envoyée, si une notification a été configurée lors du lancement de la tâche (Lancement d'une tâche (POST jobs)).

Type : chaîne

StatusCode

Code indiquant l'état de la tâche.

Valeurs Valides : `InProgress` | `Succeeded` | `Failed`

Type : chaîne

StatusMessage

Message d'état de la tâche.

Type : chaîne

Palier

Niveau d'accès aux données à utiliser pour la tâche de sélection ou d'extraction d'archive.

Valeurs Valides : `Expedited` | `Standard` | `Bulk`

Type : chaîne

VaultARN

ARN du coffre dont la tâche est une sous-ressource.

Type : chaîne

Plus d'informations

- Lancement d'une tâche (POST jobs)

Octroi

Contient des informations à propos d'une attribution.

Table des matières

Bénéficiaire

Bénéficiaire.

Type : objet Bénéficiaire

Obligatoire : non

Autorisation

Autorisation donnée au bénéficiaire.

Type : chaîne

Valeurs Valides: `FULL_CONTROL` | `WRITE` | `WRITE_ACP` | `READ` | `READ_ACP`

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

Bénéficiaire

Contient des informations à propos d'un bénéficiaire.

Table des matières

DisplayName

Nom de l'écran du bénéficiaire.

Type : chaîne

Obligatoire : non

EmailAddress

Adresse e-mail du bénéficiaire.

Type : chaîne

Obligatoire : non

ID

ID d'utilisateur canonique du bénéficiaire.

Type : chaîne

Obligatoire : non

Type

Type du bénéficiaire.

Type : chaîne

Valeurs Valides : `AmazonCustomerByEmail` | `CanonicalUser` | `Group`

Obligatoire : non

URI

URI du groupe du bénéficiaire.

Type : chaîne

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

InputSerialization

Décrit la manière dont l'archive est sérialisée.

Table des matières

CSV

Objet décrivant la sérialisation d'un objet codé CSV.

Type : objet CSVInput

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

InventoryRetrievalJobInput

Fournit des options de spécification de tâche d'extraction d'un inventaire par plage.

Table des matières

EndDate

Fin de la plage de dates au format UTC pour une extraction d'inventaire de coffre qui inclut les archives créées avant cette date.

Valeurs valides : représentation sous forme de chaîne au format de date ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) en secondes, par exemple `2013-03-20T17:03:43Z`.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) en secondes, par exemple `2013-03-20T17:03:43Z`.

Obligatoire : non

Format

Format de sortie de la liste de l'inventaire du coffre, défini par la demande Lancement d'une tâche (POST jobs) lors du lancement d'une tâche d'extraction d'un inventaire de coffre.

Valeurs valides : `CSV` | `JSON`

Obligatoire : non

Type : chaîne

Limite

Nombre maximal d'éléments d'inventaire pouvant être renvoyés pour chaque demande d'extraction d'inventaire de coffre.

Valeurs valides : valeur d'entier supérieure ou égale à 1.

Type : chaîne

Obligatoire : non

Marker

Chaîne opaque représentant l'endroit à partir duquel poursuivre la pagination des résultats d'extraction d'inventaire de coffre. Vous utilisez ce marqueur dans une nouvelle demande

`Initiate Job` afin d'obtenir des éléments d'inventaire supplémentaires. S'il n'y a pas d'autres éléments d'inventaire, cette valeur est null.

Type : chaîne

Obligatoire : non

StartDate

Début de la plage de dates au format UTC pour une extraction d'inventaire de coffre qui inclut les archives créées à cette date ou ultérieurement.

Valeurs valides : représentation sous forme de chaîne au format de date ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) en secondes, par exemple `2013-03-20T17:03:43Z`.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) en secondes, par exemple `2013-03-20T17:03:43Z`.

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

jobParameters

Fournit des options de définition d'une tâche.

Table des matières

ArchiveId

L'ID de l'archive que vous souhaitez. Ce champ est obligatoire uniquement si le champ `Type` est défini sur `select` ou `archive-retrieval`. Une erreur se produit si vous spécifiez ce champ pour une demande de tâche d'extraction d'inventaire.

Valeurs valides : il doit s'agir d'un identifiant d'archive valide que vous avez obtenu lors d'une précédente demande adressée à Amazon Glacier (Amazon Glacier).

Type : chaîne

Obligatoire : Oui lorsque `Type` est défini sur `select` ou `archive-retrieval`.

Description

La description facultative de la tâche.

Valeurs valides : la description doit être inférieure ou égale à 1 024 octets. Les caractères autorisés sont les caractères ASCII 7 bits sans code de contrôle, plus précisément les valeurs ASCII décimales ou hexadécimales comprises, respectivement, entre 32 et 126 et entre 0x20 et 0x7E.

Type : chaîne

Obligatoire : non

Format

(Facultatif) Format de sortie lors du lancement d'une tâche d'extraction d'un inventaire de coffre. Si vous lancez une tâche d'inventaire et que vous ne spécifiez pas de champ `Format`, JSON est le format par défaut.

Valeurs valides : `CSV` | `JSON`

Type : chaîne

Obligatoire : non

InventoryRetrievalParameters

Paramètres d'entrée utilisés pour l'extraction d'un inventaire par plage.

Type : objet InventoryRetrievalJobInput

Obligatoire : non

OutputLocation

Objet qui contient des informations sur l'emplacement auquel les résultats de tâches de sélection sont stockés.

Type : objet OutputLocation

Obligatoire : oui, pour les tâches `select`.

RetrievalByteRange

La plage d'octets à récupérer pour `unarchive-retrieval`, sous la forme « *`StartByteValue`* - *`EndByteValue`* ». Si ce champ n'est pas spécifié, l'archive entière est extraite. Si ce champ est

spécifié, la plage d'octets doit être alignée en termes de méga-octets (1024*1024). Aligné par mégaoctet signifie qu'StartByteValueil doit être divisible par 1 Mo, et EndByteValueplus 1 doit être divisible par 1 Mo ou être la fin de l'archive spécifiée comme valeur de taille d'octet d'archive moins 1. S'il n'RetrievalByteRangeest pas aligné en mégaoctets, cette opération renvoie une réponse. `400`

Une erreur se produit si vous spécifiez ce champ pour une demande de tâche `inventory-retrieval` ou `select`.

Type : chaîne

Obligatoire : non

SelectParameters

Objet qui contient les informations sur les paramètres utilisés pour une tâche de sélection.

Type : objet SelectParameters

Obligatoire : non

SNSTopic

Le nom de ressource Amazon (ARN) de la rubrique Amazon SNS dans laquelle Amazon Glacier envoie une notification lorsque le travail est terminé et que le résultat est prêt à être téléchargé. La rubrique spécifiée publie la notification à l'attention des abonnés.

La rubrique SNS doit exister. Si ce n'est pas le cas, Amazon Glacier ne le créera pas pour vous. En outre, la rubrique SNS doit avoir une stratégie qui permet au compte ayant créé la tâche de publier des messages dans la rubrique. Pour plus d'informations sur les noms des rubriques SNS, consultez CreateTopicle manuel Amazon Simple Notification Service API Reference.

Type : chaîne

Obligatoire : non

Palier

Niveau à utiliser pour une tâche de sélection ou d'extraction d'archive. `Standard` est la valeur par défaut utilisée.

Valeurs Valides : `Expedited` | `Standard` | `Bulk`

Type : chaîne

Obligatoire : non

Type

Le type de tâche. Vous pouvez lancer une tâche pour exécuter une requête de sélection sur une archive, d'extraction d'archive ou d'obtention d'un inventaire de coffre.

Valeurs Valides : `select` | `archive-retrieval` | `inventory-retrieval`

Type : chaîne

Obligatoire : oui

Plus d'informations

- Lancement d'une tâche (POST jobs)

OutputLocation

Contient des informations sur l'emplacement auquel les résultats des tâches et les erreurs sont stockés.

Table des matières

S3

Objet décrivant l'emplacement Amazon S3 où seront reçus les résultats de la demande de restauration.

Type : Emplacement S3

Obligatoire : oui

Plus d'informations

- Lancement d'une tâche (POST jobs)

OutputSerialization

Décrit la manière dont la sortie est sérialisée.

Table des matières

CSV

Objet décrivant la sérialisation des résultats de la demande codée CSV (valeurs séparées par des virgules).

Type : objet CSVOutput

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

Emplacement S3

Contient des informations sur l'emplacement dans Amazon S3 où sont stockés les résultats de la tâche.

Table des matières

AccessControlList

Liste des attributions contrôlant l'accès aux résultats stockés.

Type : tableau d'objets Octroi

Obligatoire : non

BucketName

Nom du compartiment Amazon S3 où sont stockés les résultats de la tâche. Le compartiment doit se trouver dans la même AWS région que le coffre contenant l'objet d'archive d'entrée.

Type : chaîne

Obligatoire : oui

CannedACL

Liste de contrôle d'accès (ACL) prédéfinie à appliquer aux résultats des tâches.

Type : chaîne

Valeurs Valides: `private` | `public-read` | `public-read-write` | `aws-exec-read` | `authenticated-read` | `bucket-owner-read` | `bucket-owner-full-control`

Obligatoire : non

Chiffrement

Objet qui contient des informations sur le chiffrement utilisé pour stocker les résultats de la tâche dans Amazon S3.

Type : objet Chiffrement

Obligatoire : non

Préfixe

Préfixe qui précède les résultats de cette demande. La longueur maximale du préfixe est de 512 caractères.

Type : chaîne

Obligatoire : oui

StorageClass

Classe de stockage utilisée pour stocker les résultats des tâches.

Type : chaîne

Valeurs Valides : `STANDARD` | `REDUCED_REDUNDANCY` | `STANDARD_IA`

Obligatoire : non

Balisage

Ensemble de balises appliqué aux résultats des tâches.

Type : mappage chaîne/chaîne

Obligatoire : non

UserMetadata

Mappage des métadonnées à stocker avec les résultats de la tâche dans Amazon S3.

Type : mappage chaîne/chaîne

Obligatoire : non

Plus d'informations

- Lancement d'une tâche (POST jobs)

SelectParameters

Contient les informations sur les paramètres utilisés pour la tâche de sélection.

Table des matières

Expression

Expression utilisée pour sélectionner l'objet. L'expression ne doit pas dépasser le quota de 128 000 caractères.

Type : chaîne

Obligatoire : oui

ExpressionType

Type de l'expression fournie, par exemple, `SQL`.

Valeurs valides : `SQL`

Type : chaîne

Obligatoire : oui

InputSerialization

Décrit le format de sérialisation de l'objet dans la sélection.

Type : objet InputSerialization

Obligatoire : non

OutputSerialization

Décrit la manière dont les résultats de la tâche de sélection sont sérialisés.

Obligatoire : non

Type : objet OutputSerialization

Plus d'informations

- Lancement d'une tâche (POST jobs)

Opérations de récupération des données

Les opérations liées à la récupération de données disponibles dans Amazon Glacier sont les suivantes.

Rubriques

- Obtention de la stratégie d'extraction de données (GET policy)
- Répertorier la capacité provisionnée (GET provisioned-capacity)
- Acheter la capacité provisionnée (POST provisioned-capacity)
- Définition de stratégie d'extraction de données (PUT policy)

Obtention de la stratégie d'extraction de données (GET policy)

Description

Cette opération renvoie la politique de récupération de données actuelle pour la AWS région Compte AWS et la région spécifiées dans la GET demande. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier.

Requêtes

Pour obtenir la stratégie actuelle d'extraction de données, envoyez une demande HTTP GET à l'URI de la stratégie d'extraction de données, comme illustré dans l'exemple de syntaxe suivant.

Syntaxe

```
GET /AccountId/policies/data-retrieval HTTP/1.1
```

```
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Syntaxe

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length
{
  "Policy":
```

```
    {
      "Rules":[
         {
            "BytesPerHour": Number,
            "Strategy": String
         }
       ]
    }
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

BytesPerHour

Nombre maximal d'octets pouvant être extraits en une heure.

Ce champ sera présent uniquement si la valeur du champ Stratégie est `BytesPerHour`.

Type : nombre

Règles

Règle de stratégie. Bien qu'il s'agisse d'un type de liste, il n'existe actuellement qu'une seule règle, qui contient un champ Stratégie et éventuellement un BytesPerHour champ.

Type : tableau

Stratégie

Le type de stratégie d'extraction de données.

Type : chaîne

Valeurs valides : `BytesPerHour|FreeTier|None`. `BytesPerHour` équivaut à sélectionner Taux d'extraction max. dans la console. `FreeTier` équivaut à sélectionner Offre gratuite uniquement

dans la console. None équivaut à sélectionner No Retrieval Policy (Pas de stratégie d'extraction) dans la console. Pour plus d'informations sur la sélection de stratégies d'extraction de données dans la console, consultez la section Politiques de récupération des données Amazon Glacier.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d'erreur.

Exemples

L'exemple suivant montre comment obtenir une stratégie d'extraction de données.

Exemple de requête

Dans cet exemple, une demande GET est envoyée à l'URI de l'emplacement d'une stratégie.

```
GET /-/policies/data-retrieval HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Une réponse positive affiche la stratégie d'extraction de données dans le corps de la réponse au format JSON.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
Content-Type: application/json
Content-Length: 85

{
  "Policy":
    {
      "Rules":[
         {
```

```
            "BytesPerHour":10737418240,
            "Strategy":"BytesPerHour"
          }
       ]
    }
}
```

Sections connexes

- Définition de stratégie d'extraction de données (PUT policy)
- Lancement d'une tâche (POST jobs)

Répertorier la capacité provisionnée (GET provisioned-capacity)

Cette opération répertorie les unités de capacité provisionnées pour le Compte AWS spécifié. Pour plus d'informations sur la capacité allouée, consultez Options de récupération des archives.

Une unité de capacité provisionnée dure un mois à compter de la date et de l'heure d'achat, qui correspondent à la date de début. L'unité expire à la date d'expiration, ce qui correspond à exactement un mois après la date de début, à la seconde près.

Si la date de début est le 31e jour du mois, la date d'expiration correspond au dernier jour du mois suivant. Par exemple, si la date de début est le 31 août, la date d'expiration est le 30 septembre. Si la date de début est le 31 janvier, la date d'expiration est le 28 février. Vous pouvez voir cette fonctionnalité sous Exemple de réponse.

Syntaxe de la requête

Pour afficher la capacité de récupération provisionnée pour un compte, envoyez une demande HTTP GET à l'URI de capacité provisionnée, comme illustré dans l'exemple de syntaxe suivant.

```
GET /AccountId/provisioned-capacity HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Si l'opération aboutit, le service renvoie une réponse HTTP `200 OK`.

Syntaxe de la réponse

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: Length
{
   "ProvisionedCapacityList":
      {
         "CapacityId" : "string",
         "StartDate" : "string"
```

```
         "ExpirationDate" : "string"
      }
}
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Le corps de la réponse contient les champs JSON suivants.

CapacityId

L'ID qui identifie l'unité de capacité provisionnée.

Type : chaîne.

StartDate

Date d'achat de la capacité provisionnée, fuseau horaire UTC.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

ExpirationDate

Date d'expiration de la capacité provisionnée, fuseau horaire UTC.

Type : chaîne. Représentation sous forme de chaîne au format de date ISO 8601, par exemple `2013-03-20T17:03:43.221Z`.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

L'exemple suivant répertorie les unités de capacité provisionnée pour un compte.

Exemple de requête

Dans cet exemple, une requête GET est envoyée pour récupérer la liste des unités de capacité provisionnée pour le compte spécifié.

```
GET /123456789012/priority-capacity HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier (Amazon Glacier) renvoie un `HTTP 200 OK` contenant la liste des unités de capacité allouées au compte, comme indiqué dans l'exemple suivant.

L'unité de capacité provisionnée répertoriée en premier est un exemple d'unité avec une date de début correspondant au 31 janvier 2017 et une date d'expiration correspondant au 28 février 2017. Comme stipulé auparavant, si la date de début est le 31e jour du mois, la date d'expiration correspond au dernier jour du mois suivant.

```
HTTP/1.1 200 OK
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
Content-Type: application/json
Content-Length: length

{
   "ProvisionedCapacityList",
      {
         "CapacityId": "zSaq7NzHFQDANTfQkDen4V7z",
         "StartDate": "2017-01-31T14:26:33.031Z",
         "ExpirationDate": "2017-02-28T14:26:33.000Z",
      },
      {
         "CapacityId": "yXaq7NzHFQNADTfQkDen4V7z",
         "StartDate": "2016-12-13T20:11:51.095Z"",
         "ExpirationDate": "2017-01-13T20:11:51.000Z" ",
      },
      ...
```

```
}
```

Sections connexes

- Acheter la capacité provisionnée (POST provisioned-capacity)

Acheter la capacité provisionnée (POST provisioned-capacity)

Cette opération achète une unité de capacité provisionnée pour un Compte AWS.

Une unité de capacité provisionnée dure un mois à compter de la date et de l'heure d'achat, qui correspondent à la date de début. L'unité expire à la date d'expiration, ce qui correspond à exactement un mois après la date de début, à la seconde près.

Si la date de début est le 31e jour du mois, la date d'expiration correspond au dernier jour du mois suivant. Par exemple, si la date de début est le 31 août, la date d'expiration est le 30 septembre. Si la date de début est le 31 janvier, la date d'expiration est le 28 février.

La capacité provisionnée garantit que la capacité de récupération pour les récupérations rapides est disponible lorsque vous en avez besoin. Chaque unité de capacité garantit qu'au moins trois extractions accélérées peuvent être effectuées toutes les cinq minutes et fournit un débit de récupération allant jusqu'à 150 MB/s %. Pour plus d'informations sur la capacité allouée, consultez Options de récupération des archives.

> **Note**
>
> Il y a une limite de deux unités de capacité provisionnées par. Compte AWS

Requêtes

Pour acheter une unité de capacité provisionnée pour Compte AWS envoyer une POST requête HTTP à l'URI de capacité provisionnée.

Syntaxe

```
POST /AccountId/provisioned-capacity HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
```

```
Content-Length: Length
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Cette opération n'a pas de corps de demande.

Réponses

Si la demande d'opération aboutit, le service renvoie une réponse HTTP `201 Created`.

Syntaxe

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-capacity-id: CapacityId
```

En-têtes de réponse

Une réponse positive comprend les en-têtes de réponse suivants, en plus de ceux communs à toutes les opérations. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Nom	Description
`x-amz-capacity-id`	L'ID qui identifie l'unité de capacité provisionnée. Type : Chaîne

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Cette opération inclut l'erreur ou les erreurs suivantes, en plus des erreurs possibles communes à toutes les opérations Amazon Glacier. Pour plus d'informations sur les erreurs Amazon Glacier et obtenir la liste des codes d'erreur, consultezRéponses d’erreur.

Code	Description	HTTP Status Code	Type
`LimitExceededException`	Retournée si la requête donnée dépasse la limite du compte pour les unités de capacité provision née.	`400 Bad Request`	Client

Exemples

L'exemple suivant illustre le cas d'un achat de capacité provisionnée pour un compte.

Exemple de requête

L'exemple suivant envoie une demande HTTP POST pour acheter une unité de capacité provisionnée.

```
POST /123456789012/provisioned-capacity HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2
```

```
Content-Length: length
x-amz-glacier-version: 2012-06-01
```

Exemple de réponse

Si la demande aboutit, Amazon Glacier (Amazon Glacier) renvoie une `HTTP 201 Created` réponse, comme indiqué dans l'exemple suivant.

```
HTTP/1.1 201 Created
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
x-amz-capacity-id: zSaq7NzHFQDANTfQkDen4V7z
```

Sections connexes

- Répertorier la capacité provisionnée (GET provisioned-capacity)

Définition de stratégie d'extraction de données (PUT policy)

Description

Cette opération définit puis met en œuvre une politique de récupération de données dans la AWS région spécifiée dans la PUT demande. Vous pouvez définir une politique par AWS région pour un Compte AWS. La stratégie est appliquée en quelques minutes dans le cadre d'une opération PUT réussie.

L'opération de définition de stratégie n'a pas d'incidence sur les tâches d'extraction qui étaient en cours avant l'application de la stratégie. Pour plus d'informations sur les stratégies d'extraction de données, consultez la section Politiques de récupération des données Amazon Glacier.

Requêtes

Syntaxe

Pour définir une stratégie d'extraction de données, envoyez une demande HTTP PUT à l'URI de stratégie d'extraction de données, comme illustré dans l'exemple de syntaxe suivant.

```
PUT /AccountId/policies/data-retrieval HTTP/1.1
Host: glacier.Region.amazonaws.com
```

```
Date: Date
Authorization: SignatureValue
Content-Length: Length
x-amz-glacier-version: 2012-06-01

{
  "Policy":
    {
      "Rules":[
         {
             "Strategy": String,
             "BytesPerHour": Number
         }
       ]
    }
}
```

> **Note**
>
> La `AccountId` valeur est l' Compte AWS ID. Cette valeur doit correspondre à l' Compte AWS ID associé aux informations d'identification utilisées pour signer la demande. Vous pouvez spécifier un Compte AWS identifiant ou éventuellement un seul « - » (trait d'union), auquel cas Amazon Glacier utilise l' Compte AWS identifiant associé aux informations d'identification utilisées pour signer la demande. Si vous spécifiez votre ID de compte, n'incluez pas de traits d'union (« - ») dans l'ID.

Paramètres de demande

Cette opération n'utilise pas les paramètres de demande.

En-têtes de demande

Cette opération utilise uniquement les en-têtes de demande qui sont communs à toutes les opérations. Pour plus d'informations sur les en-têtes de demande communs, consultez En-têtes de demande communs.

Corps de la requête

Le corps de la demande contient les champs JSON suivants.

BytesPerHour

Nombre maximal d'octets pouvant être extraits en une heure.

Ce champ n'est obligatoire que si la valeur du champ Strategy est `BytesPerHour`. Votre opération PUT est refusée si la valeur du champ Strategy n'est pas `BytesPerHour` et que vous définissez ce champ.

Type : nombre

Obligatoire : oui, si la valeur du champ Stratégie est `BytesPerHour`. Dans le cas contraire, ce champ n'est pas obligatoire.

Valeurs valides : la valeur d'entier minimale est 1. La valeur d'entier maximale est comprise entre 2^63 et 1 (inclus).

Règles

Règle de stratégie. Bien qu'il s'agisse d'un type de liste, il ne doit actuellement y avoir qu'une seule règle, qui contient un champ Stratégie et éventuellement un BytesPerHour champ.

Type : tableau

Obligatoire : oui

Stratégie

Type de stratégie d'extraction de données à définir.

Type : chaîne

Obligatoire : oui

Valeurs valides : `BytesPerHour`|`FreeTier`|None. `BytesPerHour` équivaut à sélectionner Taux d'extraction max. dans la console. `FreeTier` équivaut à sélectionner Offre gratuite uniquement dans la console. None équivaut à sélectionner No Retrieval Policy (Pas de stratégie d'extraction) dans la console. Pour plus d'informations sur la sélection de stratégies d'extraction de données dans la console, consultez la section Politiques de récupération des données Amazon Glacier.

Réponses

Syntaxe

```
HTTP/1.1 204 No Content
```

```
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

En-têtes de réponse

Cette opération utilise uniquement les en-têtes de réponse qui sont communs à la plupart des réponses. Pour plus d'informations sur les en-têtes de réponse communs, consultez En-têtes de réponse communs.

Corps de la réponse

Cette opération ne renvoie pas de corps de réponse.

Erreurs

Pour plus d'informations sur les exceptions et les messages d'erreur d'Amazon Glacier, consultezRéponses d’erreur.

Exemples

Exemple de requête

L'exemple suivant envoie une demande HTTP PUT avec la valeur `BytesPerHour` définie pour le champ Strategy.

```
PUT /-/policies/data-retrieval HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
  "Policy":
    {
      "Rules":[
         {
             "Strategy":"BytesPerHour",
             "BytesPerHour":10737418240
          }
       ]
    }
```

```
}
```

L'exemple suivant envoie une demande HTTP PUT avec la valeur `FreeTier` définie pour le champ Strategy.

```
PUT /-/policies/data-retrieval HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
  "Policy":
    {
      "Rules":[
         {
             "Strategy":"FreeTier"
          }
       ]
    }
}
```

L'exemple suivant envoie une demande HTTP PUT avec la valeur None définie pour le champ Strategy.

```
PUT /-/policies/data-retrieval HTTP/1.1
Host: glacier.us-west-2.amazonaws.com
x-amz-Date: 20170210T120000Z
x-amz-glacier-version: 2012-06-01
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20141123/
us-west-2/glacier/aws4_request,SignedHeaders=host;x-amz-date;x-amz-glacier-
version,Signature=9257c16da6b25a715ce900a5b45b03da0447acf430195dcb540091b12966f2a2

{
  "Policy":
    {
      "Rules":[
         {
             "Strategy":"None"
          }
```

```
        ]
    }
}
```

Exemple de réponse

Si la demande est acceptée, Amazon Glacier (Amazon Glacier) définit la politique et renvoie un `HTTP 204 No Content` comme indiqué dans l'exemple suivant.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sections connexes

- Obtention de la stratégie d'extraction de données (GET policy)
- Lancement d'une tâche (POST jobs)

Historique du document

- Version actuelle du produit : 2012-06-01

Le tableau suivant décrit les modifications importantes apportées à chaque version du guide du développeur Amazon Glacier à compter du 5 juillet 2018. Pour recevoir les notifications de mise à jour de cette documentation, abonnez-vous à un flux RSS.

Modification	Description	Date
Temps de démarrage améliorés pour les demandes de restauration standard effectuées via S3 Batch Operations	Les récupérations standard pour les demandes de restauration effectuées via les opérations par lot S3 peuvent désormais démarrer en quelques minutes. Pour plus d'informations, consultez Options de récupération des archives.	9 août 2023
Amazon S3 prend en charge des taux de demande de restauration plus élevés pour S3 Glacier Flexible Retrieval et S3 Glacier Deep Archive	Amazon S3 prend en charge les requêtes de restauration à un taux allant jusqu'à 1 000 transactions par seconde, par Compte AWS , pour les classes de stockage S3 Glacier Flexible Retrieval et S3 Glacier Deep Archive.	15 novembre 2022
Remplacement du nom Amazon Glacier	Amazon Glacier s'appelle désormais Amazon Glacier afin de mieux refléter l'intégration de Glacier à Amazon S3.	20 novembre 2018
Mises à jour disponibles sur RSS	Vous pouvez désormais vous abonner à un flux RSS pour recevoir des notifications	5 juillet 2018

concernant les mises à jour du guide Amazon Glacier Developer Guide.

Mises à jour antérieures

Le tableau suivant décrit les modifications importantes apportées à chaque version du guide du développeur Amazon Glacier avant le 5 juillet 2018.

Modifier	Description	Date de parution
Récupérations de données rapides et en bloc	Amazon Glacier prend désormais en charge les extractions de données accélérées et en masse en plus des extractions standard. Pour de plus amples informations, veuillez consulter Options de récupération des archives.	21 novembre 2016
Verrouillage du coffre-fort	Amazon Glacier prend désormais en charge Vault Lock, qui vous permet de déployer et d'appliquer facilement des contrôles de conformité sur des coffres-forts Amazon Glacier individuels avec une politique Vault Lock. Pour plus d'informations, consultez Verrou Amazon Glacier Vault et Stratégies de verrouillage de coffre.	8 juillet 2015
Balisage du coffre	Amazon Glacier vous permet désormais de baliser vos coffres-forts Amazon Glacier pour faciliter la gestion des ressources et des coûts. Les balises sont des étiquettes que vous pouvez définir et associer à vos coffres-forts, et leur utilisation ajoute des fonctionn alités de filtrage aux opérations telles que les rapports de AWS coûts. Pour plus d'informations, consultez Balisage des ressources Amazon Glacier et Marquer vos coffres-forts Amazon Glacier.	22 juin 2015
Stratégies d'accès aux coffres	Amazon Glacier prend désormais en charge la gestion de l'accès à vos coffres-forts Amazon Glacier individue	27 avril 2015

Modifier	Description	Date de parution
	ls en utilisant des politiques d'accès aux coffres-forts. Vous pouvez désormais définir une stratégie d'accès directement sur un coffre pour autoriser aisément les utilisateurs, les groupes commerciaux internes à votre organisation, ainsi que vos partenaires commerciaux externes, à accéder au coffre. Pour de plus amples informations, veuillez consulter Politiques d'accès au coffre-fort.	
Stratégies d'extraction des données et journalisation des audits	Amazon Glacier prend désormais en charge les politiques de récupération des données et la journalisation des audits. Les stratégies d'extraction des données vous permettent de définir facilement des limites de récupération des données et de simplifier la gestion du coût d'extraction des données. Vous pouvez définir vos propres limites de récupération de données en quelques clics dans AWS Management Console ou en utilisant l'API Amazon Glacier. Pour de plus amples informations, veuillez consulter Politiques de récupération des données Amazon Glacier. En outre, Amazon Glacier prend désormais en charge la journalisation des audits avec AWS CloudTrail, qui enregistre les appels d'API Amazon Glacier pour votre compte et envoie les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Pour de plus amples informations, veuillez consulter Journalisation des appels d'API Amazon Glacier avec AWS CloudTrail.	11 décembre 2014
Exemples de mises à jour en Java	Les exemples de code Java de ce guide qui utilisent l'AWS SDK pour Java ont été mis à jour.	27 juin 2014

Modifier	Description	Date de parution
Limite pour l'extraction d'inventaire du stock	Vous pouvez désormais limiter le nombre d'élément s d'inventaire de coffre extraits en filtrant par date de création de l'archive ou en définissant un nombre maximal d'éléments. Pour plus d'informations sur les limites d'extraction d'inventaire, consultez la page Extraction d'un inventaire par plagedans la rubrique Lancement d'une tâche (POST jobs).	31 décembre 2013
Supprimé, obsolète URLs	Suppression de la page URLs qui pointait vers l'ancienne page d'informations d'identification de sécurité dans les exemples de code.	26 juillet 2013
Prise en charge de l'extraction des plages	Amazon Glacier prend désormais en charge la récupération de plages spécifiques de vos archives. Vous pouvez lancer une tâche demandant à Amazon Glacier de préparer une archive complète ou une partie de celle-ci pour un téléchargement ultérieur. Si une archive est très volumineuse, vous trouverez peut-être plus économique de lancer plusieurs tâches séquentielles pour la préparer. Pour de plus amples informations, veuillez consulter Téléchargement d'une archive dans Amazon Glacier.	13 novembre 2012
Nouveau Guide	Il s'agit de la première version du guide du développe ur Amazon Glacier.	20 août 2012

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le AWS glossaire dans la Glossaire AWS référence.