aws

Developer Guide

Amazon Glacier

Versione API 2012-06-01

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

Amazon Glacier: Developer Guide

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

I marchi e il trade dress di Amazon non possono essere utilizzati in relazione ad alcun prodotto o servizio che non sia di Amazon, in alcun modo che possa causare confusione tra i clienti, né in alcun modo che possa denigrare o screditare Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

..... x
Che cos'è Amazon Glacier? 1
Utilizzi attualmente Amazon Glacier? 1
Modello di dati 3
Vault 3
Archive (Archivia) 4
Processo 5
Configurazione delle notifiche 6
Operazioni supportate 6
Operazioni di vault 7
Operazioni di archivio 7
Processi 7
Accesso ad Amazon Glacier 7
Regioni ed endpoint 8
Nozioni di base 9
Fase 1: prima di iniziare 10
Configura un Account AWS 10
Scarica l' AWS SDK appropriato 12
Fase 2. Creazione di un vault 13
Fase 3. Caricamento di un archivio in un vault 14
Caricamento di un archivio mediante Java 15
Caricamento di un archivio mediante .NET 21
Fase 4. Download di un archivio da un vault 22
Download di un archivio mediante Java 23
Download di un archivio mediante .NET 25
Fase 5. Eliminazione di un archivio da un vault 27
Sezioni correlate 28
Eliminazione di un archivio mediante Java 28
Eliminazione di un archivio mediante .NET 29
Eliminazione di un archivio utilizzando il AWS CLI 30
Fase 6. Eliminazione di un vault 33
A questo punto come si può procedere? 34
Utilizzo di vault 35
Operazioni di vault in Amazon Glacier 36

Creazione ed eliminazione di vault ... 36
Recupero di metadati di vault ... 36
Download di un inventario vault ... 36
Configurazione delle notifiche di vault ... 37
Creazione di un vault ... 38
Creazione di un vault mediante Java ... 38
Creazione di un vault mediante .NET ... 42
Creazione di un vault mediante REST ... 46
Creazione di un vault mediante la console ... 46
Creazione di un vault utilizzando il AWS CLI ... 47
Recupero di metadati di vault ... 48
Recupero di metadati di vault mediante Java ... 49
Recupero di metadati di vault mediante .NET ... 51
Recupero di metadati di vault mediante REST ... 54
Recupero dei metadati di Vault utilizzando il AWS CLI ... 54
Download di un inventario vault ... 55
Informazioni sull'inventario ... 57
Download di un inventario vault mediante Java ... 58
Download di un inventario vault mediante .NET ... 65
Download di un inventario vault mediante REST ... 73
Scaricamento di un inventario del vault utilizzando il AWS CLI ... 73
Configurazione delle notifiche di vault ... 76
Concetti generali ... 77
Configurazione delle notifiche di vault mediante Java ... 79
Configurazione delle notifiche di vault mediante .NET ... 82
Configurazione delle notifiche di vault mediante l'API REST ... 85
Configurazione delle notifiche di vault mediante la console ... 85
Configurazione delle notifiche di vault mediante la CLI ... 88
Eliminazione di un vault ... 89
Eliminazione di un vault mediante Java ... 90
Eliminazione di un vault mediante .NET ... 91
Eliminazione di un vault mediante REST ... 93
Eliminazione di una vault vuota mediante la console ... 93
Eliminazione di un vault utilizzando il AWS CLI ... 94
Tagging di vault ... 97
Etichettatura degli archivi tramite la console Amazon Glacier ... 98

Etichettatura degli archivi utilizzando il AWS CLI 100
Etichettatura degli archivi tramite l'API Amazon Glacier 100
Sezioni correlate 101
Vault Lock 101
Panoramica sul blocco di vault 101
Blocco di una vault mediante l'API 102
Blocco della vault mediante la CLI 104
Blocco di una vault mediante la console 106
Utilizzo di archivi 108
Operazioni di archivio 109
Caricamento di un archivio 109
Ricerca di un archivio 109
Download di un archivio 109
Eliminazione di un archivio 110
Aggiornamento di un archivio 110
Conservazione di metadati di archivio sul lato client 110
Caricamento di un archivio 110
Opzioni per il caricamento di un archivio 111
Caricamento di un archivio in un'unica operazione 112
Caricamento in parti di archivi di grandi dimensioni 122
Download di un archivio 140
Recupero degli archivi 141
Download di un archivio mediante Java 145
Download di un archivio mediante .NET 162
Scaricare un archivio di grandi dimensioni usando Python 179
Download di un archivio mediante REST 187
Scaricamento di un archivio utilizzando il AWS CLI 187
Eliminazione di un archivio 191
Eliminazione di un archivio mediante Java 192
Eliminazione di un archivio mediante .NET 194
Eliminazione di un archivio mediante REST 197
Eliminazione di un archivio utilizzando AWS CLI 197
Usando il AWS SDKs 201
AWS Librerie SDK per Java e.NET 201
Cos'è un'API di basso livello? 201
Cos'è un'API di alto livello? 202

Quando utilizzare l'API di alto livello e di basso livello 202
Lavorare con AWS SDKs 203
Usando il AWS SDK per Java 204
Utilizzo dell'API di basso livello 204
Utilizzo dell'API di alto livello 205
Esecuzione di esempi Java mediante Eclipse 206
Impostazione dell'endpoint 207
Usando il AWS SDK per .NET 207
Utilizzo dell'API di basso livello 208
Utilizzo dell'API di alto livello 209
Esecuzione di esempi .NET 210
Impostazione dell'endpoint 210
Esempi di codice 212
Nozioni di base 214
Ciao Amazon Glacier 215
Operazioni 216
Scenari 281
Archiviazione di un file, ricezione di notifiche e avvio di processo 282
Come ottenere i contenuti dell'archivio ed eliminare l'archivio 288
Sicurezza 294
Protezione dei dati 294
Crittografia dei dati 295
Gestione delle chiavi 296
Riservatezza del traffico Internet 296
Identity and Access Management 297
Destinatari 297
Autenticazione con identità 297
Gestione dell’accesso con l’utilizzo delle policy 299
Come funziona Amazon Glacier con IAM 301
Esempi di policy basate sull’identità 307
Esempi di policy basate su risorse 314
Risoluzione dei problemi 317
Riferimento alle autorizzazioni dell'API Amazon Glacier 319
Registrazione e monitoraggio 327
Convalida della conformità 328
Resilienza 330

Sicurezza dell'infrastruttura ... 331
Endpoint VPC ... 331
Policy di recupero dei dati ... 332
Scelta di una policy di recupero dei dati di Amazon Glacier ... 332
Policy Free Tier Only ... 333
Policy Max Retrieval Rate ... 334
Policy No Retrieval Limit ... 334
Utilizzo della console Amazon Glacier per configurare una policy di recupero dei dati ... 334
Utilizzo dell'API Amazon Glacier per configurare una politica di recupero dei dati ... 335
Utilizzo dell'API REST di Amazon Glacier per configurare una policy di recupero dei dati ... 335
Utilizzo di AWS SDKs per impostare una politica di recupero dei dati ... 336
Assegnazione di tag alle risorse ... 337
Nozioni di base sul tagging ... 337
Limitazioni applicate ai tag ... 338
Monitoraggio dei costi mediante il tagging ... 338
Gestione del controllo degli accessi con il tagging ... 339
Sezioni correlate ... 339
Registrazione di audit con AWS CloudTrail ... 340
Informazioni su Amazon Glacier in CloudTrail ... 340
Comprendere le voci dei file di log di Amazon Glacier ... 341
Documentazione di riferimento delle API ... 345
Intestazioni di richiesta comuni ... 346
Intestazioni di risposta comuni ... 349
Firmare le richieste ... 350
Esempio di calcolo di firma ... 351
Calcolo di firme per le operazioni di streaming ... 353
Calcolo di checksum ... 355
Esempio di struttura hash 1: caricamento di un archivio in un'unica richiesta ... 357
Esempio di struttura hash 2: caricamento di un archivio mediante un caricamento in più parti ... 358
Calcolo della struttura hash di un file ... 359
Ottenimento di checksum durante il download di dati ... 368
Risposte agli errori ... 371
Esempio 1: richiesta Describe Job con un job ID inesistente ... 373
Esempio 2: richiesta List Jobs con un valore non valido per il parametro di richiesta ... 375
Operazioni di vault ... 376

Abort Vault Lock ... 377
Add Tags To Vault ... 380
Create Vault ... 383
Complete Vault Lock ... 386
Delete Vault ... 389
Delete Vault Access Policy ... 392
Delete Vault Notifications ... 395
Describe Vault ... 397
Get Vault Access Policy ... 401
Get Vault Lock ... 405
Get Vault Notifications ... 410
Initiate Vault Lock ... 413
List Tags For Vault ... 418
List Vaults ... 421
Remove Tags From Vault ... 428
Set Vault Access Policy ... 431
Set Vault Notification Configuration ... 434
Operazioni di archivio ... 438
Delete Archive ... 439
Upload Archive ... 441
Operazioni di caricamento in più parti ... 447
Abort Multipart Upload ... 447
Completamento del caricamento in più parti ... 450
Avvio del caricamento in più parti ... 455
List Parts ... 460
Elenco dei caricamenti in più parti ... 467
Upload Part ... 474
Operazioni di processo ... 481
Describe Job ... 481
Get Job Output ... 492
Initiate Job ... 503
Visualizzazione di un elenco dei processi ... 514
Tipi di dati utilizzati nelle operazioni di processo ... 524
CSVInput ... 525
CSVOutput ... 526
Crittografia ... 527

GlacierJobDescription 528
Grant 532
Grantee 532
InputSerialization 534
InventoryRetrievalJobInput 534
jobParameters 536
OutputLocation 538
OutputSerialization 539
S3Location 539
SelectParameters 541
Operazioni di recupero di dati 542
Get Data Retrieval Policy 542
List Provision Capacity 546
Purchase Provisioned Capacity 550
Set Data Retrieval Policy 553
Cronologia dei documenti 559
Aggiornamenti precedenti 560
AWS Glossario 563

Questa pagina è riservata ai clienti esistenti del servizio Amazon Glacier che utilizzano Vaults e l'API REST originale del 2012.

Se stai cercando soluzioni di archiviazione, ti consigliamo di utilizzare le classi di storage Amazon Glacier in Amazon S3, S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval e S3 Glacier Deep Archive. Per ulteriori informazioni su queste opzioni di storage, consulta le classi di storage di Amazon Glacier.

Amazon Glacier (servizio autonomo originale basato su vault) non accetterà più nuovi clienti a partire dal 15 dicembre 2025, senza alcun impatto sui clienti esistenti. Amazon Glacier è un servizio APIs autonomo che archivia i dati in vault ed è distinto dalle classi di storage Amazon S3 e Amazon S3 Glacier. I dati esistenti rimarranno sicuri e accessibili in Amazon Glacier a tempo indeterminato. Non è richiesta alcuna migrazione. Per uno storage di archiviazione a lungo termine a basso costo, AWS consiglia le classi di storage Amazon S3 Glacier, che offrono un'esperienza cliente superiore con disponibilità Regione AWS completa, costi inferiori e integrazione dei servizi APIs basata su bucket S3. AWS Se desideri funzionalità avanzate, prendi in considerazione la migrazione alle classi di storage Amazon S3 Glacier utilizzando la AWS nostra Solutions Guidance per il trasferimento di dati dai vault Amazon Glacier alle classi di storage Amazon S3 Glacier.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.

Che cos'è Amazon Glacier?

Se attualmente utilizzi il servizio Amazon Glacier e desideri saperne di più, troverai le informazioni di cui hai bisogno in questa guida. Amazon Glacier è un servizio sicuro e durevole per l'archiviazione dei dati a basso costo e il backup a lungo termine tramite vault. Per ulteriori informazioni sui prezzi dei servizi Amazon Glacier, consulta i prezzi di Amazon Glacier.

Argomenti

- Utilizzi attualmente Amazon Glacier?
- Modello di dati Amazon Glacier
- Operazioni supportate in Amazon Glacier
- Accesso ad Amazon Glacier

Utilizzi attualmente Amazon Glacier?

> **Note**
>
> Questa sezione riguarda il servizio Amazon Glacier. Se attualmente utilizzi le classi di storage Amazon S3 Glacier (S3 Glacier Instant Retrieval, S3 Glacier Flexible Retrieval e S3 Glacier Deep Archive), consulta Classi di storage per l'archiviazione di oggetti nella Amazon S3 User Guide.

Se attualmente utilizzi il servizio Amazon Glacier e desideri saperne di più, ti consigliamo di iniziare leggendo le seguenti sezioni:

- Cos'è Amazon Glacier: il resto di questa sezione descrive il modello di dati sottostante, le operazioni che supporta e ciò che puoi utilizzare per interagire con AWS SDKs il servizio.
- Nozioni di base: la sezione Guida introduttiva ad Amazon Glacier descrive il processo per creare una vault, caricare archivi, creare processi per il download di archivi, recuperare l'output dei processi ed eliminare archivi.

> **Important**
>
> Amazon Glacier fornisce una console. Tuttavia, qualsiasi operazione di archiviazione, come il caricamento, il download o l'eliminazione, richiede l'uso del codice AWS Command Line Interface (AWS CLI) o la scrittura. La console non supporta le operazioni di archivio. Ad esempio, per caricare dati, come foto, video e altri documenti, è necessario utilizzare il codice AWS CLI o scrivere il codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il AWS SDKs.
> Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con, consulta AWS CLIAWS CLI la pagina di riferimento per Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Utilizzo di Amazon Glacier con. AWS Command Line Interface

Oltre alla sezione introduttiva, probabilmente vorrai saperne di più sulle operazioni di Amazon Glacier. Le seguenti sezioni forniscono informazioni dettagliate sull'utilizzo di Amazon Glacier utilizzando l'API REST e per Java e Microsoft AWS SDKs .NET:

- Utilizzo di AWS SDKs con Amazon Glacier

  Questa sezione fornisce una panoramica degli esempi di codice AWS SDKs utilizzati in vari esempi di codice in questa guida. La consultazione di questa sezione si rivelerà utile per la lettura delle sezioni seguenti. Include una panoramica del livello alto e basso APIs che SDKs offrono, su quando utilizzarli e sui passaggi più comuni per eseguire gli esempi di codice forniti in questa guida.

- Lavorare con Vaults in Amazon Glacier

  Questa sezione fornisce dettagli su varie operazioni di vault, come la creazione di una vault, il recupero di metadati di una vault, l'utilizzo di processi per recuperare inventari vault e la configurazione delle notifiche della vault. Oltre a utilizzare la console Amazon Glacier, puoi utilizzarla per varie operazioni di AWS SDKs vault. Questa sezione descrive l'API e fornisce esempi di lavoro utilizzando e il AWS SDK per Java . AWS SDK per .NET

- Lavorare con gli archivi in Amazon Glacier

  Questa sezione fornisce dettagli sulle operazioni di archivio come il caricamento di un archivio in un'unica richiesta oppure l'utilizzo di un'operazione di caricamento in più parti per caricare archivi di grandi dimensioni in parti. Viene inoltre descritta la creazione di processi per il download di

archivi in modo asincrono. La sezione fornisce esempi di utilizzo di AWS SDK per Java e AWS SDK per .NET.

- Riferimento API per Amazon Glacier

  Amazon Glacier è un servizio. RESTful Questa sezione descrive le operazioni REST, tra cui la sintassi e le richieste e le risposte di esempio per tutte le operazioni. Le librerie AWS SDK racchiudono questa API, semplificando le attività di programmazione.

Modello di dati Amazon Glacier

I componenti principali del modello di dati di Amazon Glacier includono vault e archivi. Amazon Glacier è un servizio Web basato su REST. In termini REST, vault e archivi sono le risorse. Inoltre, il modello di dati di Amazon Glacier include risorse per la configurazione dei processi e delle notifiche. che completano le risorse di base.

Argomenti

- Vault
- Archive (Archivia)
- Processo
- Configurazione delle notifiche

Vault

In Amazon Glacier, un vault è un contenitore per l'archiviazione di archivi. Una vault è simile a un bucket Amazon S3. Quando crei un vault, specifichi un nome e scegli Regione AWS dove vuoi creare il vault.

Ogni risorsa vault ha un indirizzo univoco. Il formato generale è:

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

Ad esempio, supponiamo di creare una vault (`examplevault`) nella regione Stati Uniti occidentali (Oregon) nel proprio account con l'ID 111122223333. È possibile accedere a tale vault mediante il seguente URI:

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

Ecco cosa significano i vari componenti dell'URI:

- `glacier.us-west-2.amazonaws.com` identifica la regione Stati Uniti occidentali (Oregon).
- `111122223333`è l' Account AWS ID che possiede il vault.
- `vaults` fa riferimento alla raccolta di vault di proprietà dell' Account AWS.
- `examplevault` identifica un determinato vault nella raccolta di vault.

An Account AWS può creare casseforti in qualsiasi ambiente supportato. Regione AWS Per un elenco di quelli supportati Regioni AWS, vedereAccesso ad Amazon Glacier. In una regione, un account deve utilizzare nomi di vault univoci. An Account AWS può creare casseforti con lo stesso nome in diverse regioni.

In una vault puoi conservare un numero illimitato di archivi. A seconda delle tue esigenze applicative o aziendali, puoi conservare tali archivi in uno o più vault.

Amazon Glacier supporta diverse operazioni di vault. Nota che queste operazioni sono specifiche per regione. Ad esempio, quando crei un vault, lo fai in una determinata regione. Quando richiedi un elenco di vault, lo richiedi a una determinata regione e l'elenco risultante include solo i vault creati in quella regione specifica Regione AWS.

Archive (Archivia)

Un archivio può contenere qualunque tipo di dati, ad esempio foto, video o documenti. Un archivio è simile a un oggetto Amazon S3 ed è l'unità di storage di base in Amazon Glacier. Ogni archivio dispone di un ID univoco e una descrizione facoltativa. Puoi specificare la descrizione facoltativa solo durante il caricamento di un archivio. Amazon Glacier assegna all'archivio un ID, che è unico nel luogo in cui è archiviato Regione AWS l'archivio.

Ogni archivio ha un indirizzo univoco. Il formato generale è il seguente:

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

Di seguito è riportato un esempio di URI di un archivio all'interno della vault `examplevault` nella regione Stati Uniti occidentali (Oregon) nell'account 111122223333:

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

In una vault puoi conservare un numero illimitato di archivi.

Processo

Un job di Amazon Glacier può recuperare un archivio o ottenere un inventario di un vault.

Il recupero di archivi e inventari di vault (elenchi di archivi) sono operazioni asincrone in Amazon Glacier, in cui prima si avvia un processo e poi si scarica l'output del lavoro dopo che Amazon Glacier ha completato il processo.

> **Note**
>
> Amazon Glacier offre una soluzione di archiviazione dei dati con storage a freddo. Se la tua applicazione necessita di una soluzione di archiviazione che richiede il recupero di dati in tempo reale, ti consigliamo di utilizzare Amazon S3. Per ulteriori informazioni, consulta Amazon Simple Storage Service (Amazon S3).

Per avviare un processo di inventario vault, devi fornire un nome di vault. I processi di recupero di archivio richiedono il nome della vault e l'ID archivio. Puoi anche fornire una descrizione facoltativa dei processi per facilitarne l'identificazione.

I processi di recupero di archivio e inventario vault sono associati a una vault. Un vault può avere più processi in corso in qualsiasi momento. Quando invii una richiesta di lavoro (avvia un lavoro), Amazon Glacier ti restituisce un ID lavoro per tracciare il lavoro. Ogni processo è identificato in modo univoco da un URI nel formato:

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

Di seguito è riportato un esempio di processo associato a una vault `examplevault` nella regione Stati Uniti occidentali (Oregon) nell'account 111122223333.

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

Per ogni lavoro, Amazon Glacier conserva informazioni come il tipo di lavoro, la descrizione, la data di creazione, la data di completamento e lo stato del lavoro. Puoi ottenere informazioni su un

determinato processo o ottenere un elenco di tutti i processi associati a un vault. L'elenco dei lavori restituiti da Amazon Glacier include tutti i lavori in corso e quelli terminati di recente.

Configurazione delle notifiche

Poiché l'esecuzione dei lavori richiede tempo, Amazon Glacier supporta un meccanismo di notifica per avvisarti quando un processo è completato. Puoi quindi configurare una vault per inviare una notifica a un argomento di Amazon Simple Notification Service (Amazon SNS) al completamento di un processo. Puoi specificare un argomento di Amazon SNS per ogni vault nella configurazione delle notifiche.

Amazon Glacier archivia la configurazione delle notifiche come documento JSON. Di seguito è riportato un esempio di configurazione delle notifiche di vault:

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

Le configurazioni delle notifiche sono associate alle vault; puoi impostare una sola configurazione per vault. Ogni risorsa di configurazione delle notifiche è identificata in modo univoco con un URI nel formato:

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

Amazon Glacier supporta le operazioni per impostare, ricevere ed eliminare una configurazione di notifica. Se elimini una configurazione, non viene inviata alcuna notifica al completamento di ogni operazione di recupero dati nella vault.

Operazioni supportate in Amazon Glacier

Per lavorare con vault e archivi (vediModello di dati Amazon Glacier), Amazon Glacier supporta una serie di operazioni. Tra tutte le operazioni supportate, solo quelle elencate di seguito sono asincrone:

- Recupero di un archivio
- Recupero di un inventario vault (elenco di archivi)

Queste operazioni richiedono dapprima l'avvio di un processo e quindi il download dell'output del processo. Le seguenti sezioni riassumono le operazioni di Amazon Glacier.

Operazioni di vault

Amazon Glacier fornisce operazioni per creare ed eliminare vault. Puoi ottenere una descrizione di un determinato vault o di tutte le vault in una Regione AWS. La descrizione del vault fornisce informazioni come la data di creazione, il numero di archivi nel vault, la dimensione totale in byte utilizzata da tutti gli archivi del vault e la data in cui Amazon Glacier ha generato l'inventario del vault. Amazon Glacier fornisce anche operazioni per impostare, recuperare ed eliminare una configurazione di notifica nel vault. Per ulteriori informazioni, consulta Lavorare con Vaults in Amazon Glacier.

Operazioni di archivio

Amazon Glacier fornisce operazioni per caricare ed eliminare archivi. L'aggiornamento di un archivio esistente non è consentito; devi eliminare quell'archivio e caricarne uno nuovo. Ogni volta che carichi un archivio, Amazon Glacier genera un nuovo ID di archivio. Per ulteriori informazioni, consulta Lavorare con gli archivi in Amazon Glacier.

Processi

Puoi avviare un processo Amazon Glacier per eseguire un recupero su un archivio o creare un inventario di un vault.

Di seguito sono riportati i tipi di lavori in Amazon Glacier:

- `archive-retrieval`: recupero di un archivio

  Per ulteriori informazioni, consulta Scaricare un archivio in Amazon Glacier.
- `inventory-retrieval`: inventario di una vault.

  Per ulteriori informazioni, consulta Scaricamento di un inventario Vault in Amazon Glacier.

Accesso ad Amazon Glacier

Amazon Glacier è RESTful un servizio Web che utilizza HTTP e HTTPS come protocollo di trasporto JavaScript e Object Notation (JSON) come formato di serializzazione dei messaggi. Il codice dell'applicazione può effettuare richieste direttamente all'API del servizio web Amazon Glacier.

Quando utilizzi la REST API direttamente, devi scrivere il codice necessario per firmare e autenticare le richieste. Per ulteriori informazioni sull'API, consulta Riferimento API per Amazon Glacier.

In alternativa, puoi semplificare lo sviluppo di applicazioni utilizzando AWS SDKs le chiamate API REST di Amazon Glacier. In tal caso, devi soltanto fornire le credenziali e le librerie eseguono automaticamente l'autenticazione e la firma delle richieste. Per ulteriori informazioni sull'uso di AWS SDKs, consulta Utilizzo di AWS SDKs con Amazon Glacier.

Amazon Glacier fornisce anche una console. Tuttavia, tutte le operazioni di archiviazione e di lavoro richiedono la scrittura di codice e l'invio di richieste utilizzando direttamente l'API REST o le librerie wrapper AWS SDK. Per accedere alla console Amazon Glacier, vai alla console Amazon Glacier.

Regioni ed endpoint

Crei un vault in un ambiente specifico. Regione AWS Invii sempre le tue richieste Amazon Glacier a un endpoint specifico di un. Regione AWSPer un elenco di quelli Regioni AWS supportati da Amazon Glacier, consulta gli endpoint e le quote di Amazon Glacier nella Guida generale.AWS

Guida introduttiva ad Amazon Glacier

Puoi iniziare a usare Amazon Glacier (Amazon Glacier) lavorando con vault e archivi. Una vault è un container di archivi e un archivio è un qualsiasi oggetto, ad esempio una foto, un video o un documento archiviato in una vault. Un archivio è l'unità di storage di base in Amazon Glacier. Questo esercizio introduttivo fornisce istruzioni per esplorare le operazioni di base di Amazon Glacier su vault e archivi. Per ulteriori informazioni su queste risorse, consulta la sezione Modello di dati Amazon Glacier.

Nel corso dell'esercizio, creerai una vault, caricherai e scaricherai un archivio e infine eliminerai l'archivio e la vault. Tutte queste operazioni possono essere eseguite a livello di codice, Tuttavia, l'esercizio introduttivo utilizza la console di gestione Amazon Glacier per creare ed eliminare un vault. Per caricare e scaricare un archivio, questa sezione introduttiva utilizza l'API di alto livello per and the. AWS SDK per Java AWS SDK per .NET L'API di alto livello offre un'esperienza di programmazione semplificata quando si lavora con Amazon Glacier. Per ulteriori informazioni sull'utilizzo dell'API di alto livello con, consulta. AWS SDKs Utilizzo di AWS SDKs con Amazon Glacier

> **Important**
>
> Amazon Glacier fornisce una console. Tuttavia, qualsiasi operazione di archiviazione, come il caricamento, il download o l'eliminazione, richiede l'utilizzo della AWS Command Line Interface (CLI) o la scrittura del codice. La console non supporta le operazioni di archivio. Ad esempio, per caricare dati, come foto, video e altri documenti, è necessario utilizzare il codice AWS CLI o scrivere il codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il AWS SDKs.
>
> Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con, consulta AWS CLIAWS CLI la pagina di riferimento per Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Usare Amazon Glacier con. AWS Command Line Interface

Questo esercizio fornisce esempi di codice in Java e C# per il caricamento e il download di un archivio. L'ultima sezione dell'esercizio introduttivo fornisce passaggi in cui puoi saperne di più sull'esperienza degli sviluppatori con Amazon Glacier.

Argomenti

- Fase 1: Prima di iniziare a usare Amazon Glacier
- Fase 2: creazione di un vault in Amazon Glacier
- Fase 3: caricare un archivio in un vault in Amazon Glacier
- Passaggio 4: scaricare un archivio da un vault in Amazon Glacier
- Passaggio 5: eliminare un archivio da un vault in Amazon Glacier
- Passaggio 6: eliminare un vault in Amazon Glacier
- A questo punto come si può procedere?

Fase 1: Prima di iniziare a usare Amazon Glacier

Prima di iniziare con questo esercizio, devi registrarti a un file Account AWS (se non ne hai già uno) e poi scaricarne uno. AWS SDKs Per le istruzioni, consulta le sezioni seguenti.

Argomenti

- Configurare un utente Account AWS e un utente amministratore
- Scarica l' AWS SDK appropriato

Configurare un utente Account AWS e un utente amministratore

Se non l'hai ancora fatto, devi registrarti Account AWS e creare un utente amministratore nell'account.

Per completare la configurazione, seguire le istruzioni nei seguenti argomenti.

Configurare Account AWS e creare un utente amministratore

Iscriviti per AWS

Quando ti iscrivi ad Amazon Web Services (AWS), ti iscrivi Account AWS automaticamente a tutti i servizi in AWS, incluso Amazon Glacier. Ti vengono addebitati solo i servizi che utilizzi. Per ulteriori informazioni sui tassi di utilizzo di Amazon Glacier, consulta la pagina dei prezzi di Amazon Glacier.

Se ne hai già uno Account AWS, passa a. Scarica l' AWS SDK appropriato Se non ne hai uno Account AWS, usa la procedura seguente per crearne uno.

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la https://portal.aws.amazon.com/billing/registrazione.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata o un messaggio di testo e ti verrà chiesto di inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, Utente root dell'account AWSviene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

Per creare un utente amministratore, scegli una delle seguenti opzioni.

Scelta di un modo per gestire il tuo amministratore	Per	Come	Puoi anche
In IAM Identity Center (Consigliato)	Usa credenziali a breve termine per accedere a AWS. Ciò è in linea con le best practice per la sicurezza. Per informazioni sulle best practice, consulta Best practice per la sicurezza in IAM nella Guida per l'utente di IAM.	Segui le istruzioni riportate in Nozioni di base nella Guida per l'utente di AWS IAM Identity Center .	Configura l'accesso programmatico configurando l'uso AWS IAM Identity Center nella Guida AWS CLI per l'AWS Command Line Interface utente.

Scelta di un modo per gestire il tuo amministratore	Per	Come	Puoi anche
In IAM (Non consigliato)	Usa credenziali a lungo termine per accedere a AWS.	Seguendo le istruzioni contenute in Creare un utente IAM per l'accesso di emergenza nella Guida per l'utente IAM.	Configura l'accesso programmatico tramite Manage access keys for IAM users nella IAM User Guide.

Scarica l' AWS SDK appropriato

Per provare l'esercizio introduttivo, devi decidere quale linguaggio di programmazione utilizzare e quindi scaricare l' AWS SDK appropriato per la tua piattaforma di sviluppo.

L'esercizio fornisce esempi in Java e C#.

Download del kit AWS SDK per Java

Per testare gli esempi Java contenuti in questa guida per sviluppatori, devi utilizzare il kit AWS SDK per Java. Per scaricarlo, hai a disposizione le seguenti opzioni di download:

- Se utilizzi Eclipse, puoi scaricarlo e installarlo AWS Toolkit for Eclipse utilizzando il sito di aggiornamento http://aws.amazon.com/eclipse/. Per ulteriori informazioni, consulta AWS Toolkit for Eclipse.
- Se intendi utilizzare qualsiasi altro IDE, scarica AWS SDK per Java.

Download del kit AWS SDK per .NET

Per testare gli esempi in C# contenuti in questa guida per gli sviluppatori, devi utilizzare il kit AWS SDK per .NET. Per scaricarlo, hai a disposizione le seguenti opzioni di download:

- Se utilizzi Visual Studio, puoi installare AWS SDK per .NET sia il AWS Toolkit for Visual Studio. Il toolkit fornisce AWS Explorer per Visual Studio e modelli di progetto che è possibile utilizzare

per lo sviluppo. Per scaricare il AWS SDK per .NET, vai a http://aws.amazon.com/sdkfornet. Per impostazione predefinita, lo script di installazione installa sia l' AWS SDK che il. AWS Toolkit for Visual Studio Per ulteriori informazioni sul toolkit, consulta la Guida per l'utente di AWS Toolkit for Visual Studio.

- Se intendi creare l'applicazione con qualsiasi altro IDE, puoi utilizzare lo stesso collegamento fornito nella fase precedente e installare solo il kit AWS SDK per .NET.

Fase 2: creazione di un vault in Amazon Glacier

Una vault è un container per l'archiviazione di archivi. Il primo passaggio consiste nel creare un vault in uno dei vault supportati. Regioni AWSPer un elenco di quelli Regioni AWS supportati da Amazon Glacier, consulta gli endpoint e le quote di Amazon Glacier nella Guida generale.AWS

Puoi creare vault a livello di codice o utilizzando la console Amazon Glacier. In questa fase, utilizziamo la console.

Creazione di un vault

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. Nel pannello di navigazione a sinistra, seleziona Vault.
3. Seleziona Crea vault.

   Viene visualizzata la pagina Crea vault.
4. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione. La vault si trova nella regione selezionata.
5. Per Nome vault, inserisci un nome per la vault.

   Di seguito sono riportati i requisiti di denominazione della vault:

   - Il nome di un archivio deve essere univoco all'interno dell'ambiente Regione AWS in cui è stato creato. Account AWS
   - Un nome della vault deve avere una lunghezza compresa tra 1 e 255 caratteri.
   - Un nome di una vault può contenere solo i seguenti caratteri: a-z, A-Z, 0-9, _ (trattino basso), - (trattino) e . (punto).
6. In Notifiche eventi, per attivare o disattivare le notifiche su una vault relative al completamento di un processo, scegli una delle seguenti impostazioni:

- Disattivazione delle notifiche: le notifiche sono disattivate e non vengono inviate a un argomento Amazon Simple Notification Service (Amazon SNS) al completamento di un processo specificato.

- Attivazione delle notifiche: le notifiche sono attivate e vengono inviate all'argomento di Amazon SNS al completamento di un processo specificato.

  Se hai scelto Attiva le notifiche, consulta Configurazione delle notifiche Vault utilizzando la console Amazon Glacier.

7. Se il nome del vault Regione AWS e è corretto, scegli Crea vault.

Il tuo nuovo vault è ora elencato nella pagina Vaults della console Amazon Glacier.

Fase 3: caricare un archivio in un vault in Amazon Glacier

In questo passaggio, carichi un archivio di esempio nella vault creata nella fase precedente (consulta Fase 2: creazione di un vault in Amazon Glacier). A seconda della piattaforma di sviluppo che utilizzi, fai clic su uno dei collegamenti alla fine di questa sezione.

> **Important**
>
> Qualsiasi operazione di archivio, ad esempio caricamento, download o eliminazione, richiede l'utilizzo dell' AWS Command Line Interface (CLI) o la scrittura di codice. La console non supporta le operazioni di archivio. Ad esempio, per caricare dati, come foto, video e altri documenti, devi utilizzare AWS CLI o scrivere codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il. AWS SDKs
> Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con AWS CLI, AWS CLI consulta Reference for Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Usare Amazon Glacier con. AWS Command Line Interface

Un archivio è un qualsiasi oggetto, ad esempio una foto, un video o un documento, che conservi in una vault. Un archivio è l'unità di storage di base in Amazon Glacier. Puoi caricare un archivio in una singola richiesta. Per archivi di grandi dimensioni, Amazon Glacier fornisce un'operazione API di caricamento multiparte che consente di caricare un archivio in parti.

In questa sezione, carichi un archivio di esempio in una singola richiesta. Per questo esercizio, specifichi un file di piccole dimensioni. Per i file di grandi dimensioni, il caricamento in più parti è la soluzione più appropriata. Per ulteriori informazioni, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Argomenti

- Carica un archivio in un vault in Amazon Glacier utilizzando AWS SDK per Java
- Carica un archivio in un vault in Amazon Glacier utilizzando il AWS SDK per .NET

Carica un archivio in un vault in Amazon Glacier utilizzando AWS SDK per Java

Il seguente esempio di codice Java utilizza l'API di alto livello di AWS SDK per Java per caricare un archivio di esempio nel vault. Nell'esempio di codice, nota quanto segue:

- L'esempio crea un'istanza della classe `AmazonGlacierClient`.
- Nell'esempio viene utilizzata l'operazione API `upload` della classe `ArchiveTransferManager` dell'API di alto livello dell' AWS SDK per Java.
- L'esempio utilizza la regione (`us-west-2`) Stati Uniti occidentali (Oregon).

Per step-by-step istruzioni su come eseguire questo esempio, vedere. Esecuzione di esempi Java per Amazon Glacier con Eclipse Devi aggiornare il codice con il nome dell'archivio di file che intendi caricare come indicato.

> **Note**
>
> Amazon Glacier conserva un inventario di tutti gli archivi nei tuoi vault. Quando carichi l'archivio nel seguente esempio, non sarà visualizzato in un vault nella console di gestione fino a quando l'inventario vault non viene aggiornato. Questo aggiornamento viene in genere eseguito una volta al giorno.

SDK per Java 2.x

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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();
    }
}
```

- Per i dettagli sull'API, consulta la UploadArchivesezione AWS SDK for Java 2.x API Reference.

Carica un archivio in un vault in Amazon Glacier utilizzando il AWS SDK per .NET

Il seguente esempio di codice C# utilizza l'API di alto livello di AWS SDK per .NET per caricare un archivio di esempio nel vault. Nell'esempio di codice, nota quanto segue:

- L'esempio crea un'istanza della `ArchiveTransferManager` classe per l'endpoint Amazon Glacier Region specificato.
- Il codice di esempio utilizzata la regione (`us-west-2`) Stati Uniti occidentali (Oregon).
- L'esempio utilizza l'operazione API `Upload` della classe `ArchiveTransferManager` per caricare l'archivio. Per archivi di piccole dimensioni, questa operazione carica l'archivio direttamente su Amazon Glacier. Per archivi di grandi dimensioni, questa operazione utilizza l'operazione API di caricamento multiparte di Amazon Glacier per suddividere il caricamento in più parti per un migliore recupero degli errori, se si riscontrano errori durante lo streaming dei dati su Amazon Glacier.

Per step-by-step istruzioni su come eseguire il seguente esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con il nome della vault e il nome del file di archivio da caricare come indicato.

> **Note**
>
> Amazon Glacier conserva un inventario di tutti gli archivi nei tuoi vault. Quando carichi l'archivio nel seguente esempio, non sarà visualizzato in una vault nella console di gestione fino a quando l'inventario vault non viene aggiornato. Questo aggiornamento viene in genere eseguito una volta al giorno.

Example — Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .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();
        }
    }
}
```

Passaggio 4: scaricare un archivio da un vault in Amazon Glacier

In questa fase, scarichi l'archivio di esempio caricato precedentemente in Fase 3: caricare un archivio in un vault in Amazon Glacier.

> **Important**
>
> Amazon Glacier fornisce una console. Tuttavia, qualsiasi operazione di archiviazione, come il caricamento, il download o l'eliminazione, richiede l'utilizzo della AWS Command Line Interface (CLI) o la scrittura del codice. La console non supporta le operazioni di archivio. Ad esempio, per caricare dati, come foto, video e altri documenti, è necessario utilizzare il codice AWS CLI o scrivere il codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il AWS SDKs.

Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con AWS CLI, AWS CLI consulta Reference for Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Usare Amazon Glacier con. AWS Command Line Interface

In generale, il recupero dei dati da Amazon Glacier è un processo in due fasi:

1. Avvia un processo di recupero.
2. Download dei dati dopo il completamento del processo.

Per recuperare un archivio da Amazon Glacier, devi prima avviare un processo. e quindi scaricare i dati dopo il completamento del processo. Per ulteriori informazioni sul recupero di archivi, consulta Recupero degli archivi Amazon Glacier.

Il tempo di accesso della tua richiesta dipende dall'opzione di recupero scelta: Expedited, Standard o Bulk. Per tutti gli archivi, tranne quelli più grandi (250 MB e oltre), gli archivi a cui si accede utilizzando i recuperi accelerati sono generalmente disponibili entro 1-5 minuti. Gli archivi recuperati con l'opzione Standard divengono in genere disponibili nel giro di 3-5 ore. I recuperi Bulk generalmente divengono disponibili entro 5-12 ore. Per ulteriori informazioni sulle varie opzioni di recupero, consulta le domande frequenti su Amazon Glacier. Per informazioni sui costi di recupero dei dati, consulta la pagina dei prezzi di Amazon Glacier.

Gli esempi di codice illustrati negli argomenti seguenti avviano il processo, ne attendono il completamento e quindi eseguono il download dei dati dell'archivio.

Argomenti

- Scarica un archivio da un vault in Amazon Glacier utilizzando AWS SDK per Java
- Scarica un archivio da un vault in Amazon Glacier utilizzando AWS SDK per .NET

Scarica un archivio da un vault in Amazon Glacier utilizzando AWS SDK per Java

Il seguente esempio di codice Java utilizza l'API di alto livello di AWS SDK per Java per scaricare l'archivio caricato nel passaggio precedente. Nell'esempio di codice, nota quanto segue:

- L'esempio crea un'istanza della classe `AmazonGlacierClient`.

- Il codice utilizza la regione Stati uniti occidentali (Oregon) (`us-west-2`), ovvero l'ubicazione in cui hai creato la vault in Fase 2: creazione di un vault in Amazon Glacier.
- Nell'esempio viene utilizzata l'operazione API `download` della classe `ArchiveTransferManager` dell'API di alto livello dell' AWS SDK per Java. L'esempio crea un argomento Amazon Simple Notification Service (Amazon SNS) e una coda Amazon Simple Queue Service (Amazon SQS) sottoscritta a quell'argomento. Se hai creato un utente amministratore AWS Identity and Access Management (IAM) come indicato inFase 1: Prima di iniziare a usare Amazon Glacier, l'utente dispone delle autorizzazioni IAM necessarie per la creazione e l'uso dell'argomento Amazon SNS e della coda Amazon SQS.

Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi Java per Amazon Glacier con Eclipse Devi aggiornare il codice con l'ID archivio del file che hai caricato in Fase 3: caricare un archivio in un vault in Amazon Glacier come indicato.

Example - Download di un archivio mediante l' AWS SDK per 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);
        }
    }
}
```

Scarica un archivio da un vault in Amazon Glacier utilizzando AWS SDK per .NET

Il seguente esempio di codice C# utilizza l'API di alto livello di AWS SDK per .NET per scaricare l'archivio in cui hai caricato in precedenza. Carica un archivio in un vault in Amazon Glacier utilizzando il AWS SDK per .NET Nell'esempio di codice, nota quanto segue:

- L'esempio crea un'istanza della `ArchiveTransferManager` classe per l'endpoint Amazon Glacier Region specificato.
- L'esempio di codice utilizza la regione Stati uniti occidentali (Oregon) (`us-west-2`), ovvero l'ubicazione in cui hai precedentemente creato la vault in Fase 2: creazione di un vault in Amazon Glacier.
- L'esempio utilizza l'operazione API `Download` della classe `ArchiveTransferManager` per eseguire il download dell'archivio. L'esempio crea un argomento Amazon Simple Notification Service (Amazon SNS) e una coda Amazon Simple Queue Service (Amazon SQS) sottoscritta a quell'argomento. Se hai creato un utente amministratore AWS Identity and Access Management (IAM) come indicato inFase 1: Prima di iniziare a usare Amazon Glacier, l'utente dispone delle autorizzazioni IAM necessarie per la creazione e l'uso dell'argomento Amazon SNS e della coda Amazon SQS.

- L'esempio quindi avvia il processo di recupero dell'archivio ed esegue il polling della coda affinché l'archivio sia disponibile. Quando l'archivio è disponibile, il download ha inizio. Per informazioni sui tempi di recupero, consulta Opzioni di recupero dall'archivio.

Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con l'ID archivio del file che hai caricato in Fase 3: caricare un archivio in un vault in Amazon Glacier come indicato.

Example — Scarica un archivio utilizzando l'API di alto livello di AWS SDK per .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);
            }
        }
    }
}
```

Passaggio 5: eliminare un archivio da un vault in Amazon Glacier

In questa fase, elimini l'archivio di esempio caricato in Fase 3: caricare un archivio in un vault in Amazon Glacier.

> **Important**
>
> Non puoi eliminare un archivio utilizzando la console Amazon Glacier. Qualsiasi operazione di archiviazione, come il caricamento, il download o l'eliminazione, richiede l'utilizzo della AWS Command Line Interface (CLI) o la scrittura del codice. Per caricare dati, come foto, video e altri documenti, è necessario utilizzare il codice AWS CLI o scrivere il codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il AWS SDKs.
> Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con AWS CLI, AWS CLI consulta Reference for Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Usare Amazon Glacier con. AWS Command Line Interface

Elimina l'archivio di esempio seguendo uno di questi o i seguenti passaggi: SDKs AWS CLI

- Eliminare un archivio da un vault in Amazon Glacier utilizzando il AWS SDK per Java
- Eliminare un archivio da un vault in Amazon Glacier utilizzando AWS SDK per .NET
- Eliminare un archivio in Amazon Glacier utilizzando AWS CLI

Sezioni correlate

- Fase 3: caricare un archivio in un vault in Amazon Glacier
- Eliminazione di un archivio in Amazon Glacier

Eliminare un archivio da un vault in Amazon Glacier utilizzando il AWS SDK per Java

Il seguente esempio di codice utilizza AWS SDK per Java per eliminare l'archivio. Nel codice, tenere presente quanto segue:

- L'oggetto `DeleteArchiveRequest` descrive la richiesta di eliminazione, tra cui il nome della vault in cui si trova l'archivio e l'ID archivio.
- L'operazione `deleteArchive` API invia la richiesta ad Amazon Glacier per eliminare l'archivio.
- L'esempio utilizza la regione (`us-west-2`) Stati Uniti occidentali (Oregon).

Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi Java per Amazon Glacier con Eclipse Devi aggiornare il codice con l'ID archivio del file che hai caricato in Fase 3: caricare un archivio in un vault in Amazon Glacier come indicato.

Example - Eliminazione di un archivio mediante l' AWS SDK per 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);
        }
    }
}
```

Eliminare un archivio da un vault in Amazon Glacier utilizzando AWS SDK per .NET

Il seguente esempio di codice C# utilizza l'API di alto livello di AWS SDK per .NET per eliminare l'archivio caricato nel passaggio precedente. Nell'esempio di codice, nota quanto segue:

- L'esempio crea un'istanza della `ArchiveTransferManager` classe per l'endpoint Amazon Glacier Region specificato.
- Il codice di esempio utilizzata la regione (`us-west-2`) Stati Uniti occidentali (Oregon).
- L'esempio utilizza l'operazione API `Delete` della classe `ArchiveTransferManager` fornita come parte dall'API di alto livello dell' AWS SDK per .NET.

Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con l'ID archivio del file che hai caricato in Fase 3: caricare un archivio in un vault in Amazon Glacier come indicato.

Example — Eliminazione di un archivio utilizzando l'API di alto livello di AWS SDK per .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();
    }
  }
}
```

Eliminare un archivio in Amazon Glacier utilizzando AWS CLI

Puoi eliminare gli archivi in Amazon Glacier utilizzando AWS Command Line Interface ().AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: eliminazione di un archivio utilizzando AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface
2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: eliminazione di un archivio utilizzando AWS CLI

1. Utilizzare il comando `initiate-job` per avviare un'attività di recupero inventario. Per ulteriori informazioni sul comando `initiate-job`, consulta Avvia processo.

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

   Output previsto:

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

```
}
```

2. Utilizza il comando `describe-job` per verificare lo stato dell'attività di recupero precedente. Per ulteriori informazioni sul comando `describe-job`, consulta Descrivi processo.

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

Output previsto:

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

3. Attendi il completamento del processo.

Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

Puoi impostare la configurazione delle notifiche per specifici eventi sulla vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier. Amazon Glacier invia un messaggio all'argomento Amazon SNS specificato ogni volta che si verifica l'evento specifico.

4. Al termine del processo, utilizza il comando `get-job-output` per eseguire il download del processo di recupero nel file `output.json`. Per ulteriori informazioni sul comando `get-job-output`, consulta Ottieni output processo.

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

Questo comando produce un file con i seguenti campi.

```
{
"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. Utilizzare il comando `delete-archive` per eliminare tutti gli archivi da una vault finché non ne rimane nessuno.

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

Per ulteriori informazioni sul comando `delete-archive`, consulta Elimina archivio.

Passaggio 6: eliminare un vault in Amazon Glacier

Una vault è un container per l'archiviazione di archivi. Per eliminare un vault Amazon Glacier, devi prima eliminare tutti gli archivi esistenti nel vault a partire dall'ultimo inventario calcolato da Amazon Glacier.

Puoi eliminare un vault a livello di codice o utilizzando la console Amazon Glacier. Per informazioni sull'eliminazione di un vault a livello di codice, consulta Eliminazione di un vault in Amazon Glacier.

> **Important**
>
> Se carichi un archivio in una vault o elimini un archivio da una vault nelle ultime 24 ore, devi attendere che l'ultimo inventario vault venga aggiornato per riflettere le informazioni più recenti. Amazon Glacier prepara un inventario per ogni vault ogni 24 ore.

Per eliminare un vault vuoto

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. Dal menu Seleziona una regione, scegli Regione AWS il vault che desideri eliminare.

   In questo esercizio introduttivo, la vault di esempio si trova nella regione degli Stati Uniti occidentali (Oregon).
3. Seleziona il pulsante di opzione accanto alla vault vuota che desideri eliminare. Se la vault non è vuota è necessario eliminare tutti gli archivi prima di eliminare la vault. Per ulteriori informazioni, consulta Eliminazione di un archivio in Amazon Glacier.

   > **Important**
   >
   > L'eliminazione di una vault non può essere annullata.
4. Scegli Elimina.
5. Viene visualizzata la finestra di dialogo Elimina vault. Scegli Elimina.

Per eliminare un vault non vuoto

1. Se si elimina una vault non vuota, è necessario innanzitutto eliminare tutti gli archivi esistenti prima di eliminare la vault. Puoi farlo scrivendo il codice per effettuare una richiesta di eliminazione dell'archivio utilizzando l'API REST, il AWS SDK per Java, AWS SDK per .NET o il AWS CLI. Per informazioni sull'eliminazione di archivi, consulta Passaggio 5: eliminare un archivio da un vault in Amazon Glacier.
2. Quando la vault è vuota, segui i passaggi per eliminare una vault vuota nella procedura precedente.

A questo punto come si può procedere?

Ora che hai provato l'esercizio introduttivo, puoi esplorare le seguenti sezioni per saperne di più su Amazon Glacier.

- Lavorare con Vaults in Amazon Glacier
- Lavorare con gli archivi in Amazon Glacier

Lavorare con Vaults in Amazon Glacier

Una vault è un container per l'archiviazione di archivi. Quando crei un vault, specifichi il nome del vault e il nome Regione AWS in cui desideri crearlo. Per un elenco di quelli Regioni AWS supportati da Amazon Glacier, consulta gli endpoint e le quote di Amazon Glacier nella Guida generale.AWS

In una vault puoi conservare un numero illimitato di archivi.

> **Important**
>
> Amazon Glacier fornisce una console. Tuttavia, qualsiasi operazione di archiviazione, come il caricamento, il download o l'eliminazione, richiede l'uso del codice AWS Command Line Interface (AWS CLI) o la scrittura. La console non supporta le operazioni di archivio. Ad esempio, per caricare dati, come foto, video e altri documenti, è necessario utilizzare il codice AWS CLI o scrivere il codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il AWS SDKs.
> Per installare AWS CLI, consulta AWS Command Line Interface. Per ulteriori informazioni sull'uso di Amazon Glacier con, consulta AWS CLIAWS CLI la pagina di riferimento per Amazon Glacier. Per esempi di utilizzo di per AWS CLI caricare archivi su Amazon Glacier, consulta Usare Amazon Glacier con. AWS Command Line Interface

Argomenti

- Operazioni di vault in Amazon Glacier
- Creazione di un vault in Amazon Glacier
- Recupero dei metadati del vault in Amazon Glacier
- Scaricamento di un inventario Vault in Amazon Glacier
- Configurazione delle notifiche Vault in Amazon Glacier
- Eliminazione di un vault in Amazon Glacier
- Etichettatura dei vault Amazon Glacier
- Serratura Amazon Glacier Vault

Operazioni di vault in Amazon Glacier

Amazon Glacier supporta diverse operazioni di vault. Le operazioni della vault sono specifiche a seconda della Regioni AWS. In altre parole, quando si crea una vault, lo si crea in una Regione AWS specifica. Quando elenchi i vault, Amazon Glacier restituisce l'elenco dei vault Regione AWS da quello specificato nella richiesta.

Creazione ed eliminazione di vault

An Account AWS può creare fino a 1.000 vault per. Regione AWSPer un elenco di quelli Regioni AWS supportati da Amazon Glacier, consulta gli endpoint e le quote di Amazon Glacier nella Guida generale.AWS

Puoi eliminare un vault solo se non ci sono archivi nel vault a partire dall'ultimo inventario calcolato da Amazon Glacier e se non ci sono state scritture nel vault dall'ultimo inventario.

> **Note**
>
> Amazon Glacier prepara un inventario per ogni vault ogni 24 ore. Poiché l'inventario potrebbe non riflettere le informazioni più recenti, Amazon Glacier garantisce che il vault sia effettivamente vuoto controllando se sono state eseguite operazioni di scrittura dall'ultimo inventario del vault.

Per ulteriori informazioni, consultare Creazione di un vault in Amazon Glacier e Eliminazione di un vault in Amazon Glacier.

Recupero di metadati di vault

Puoi recuperare informazioni relative a una vault, ad esempio data di creazione, numero di archivi nella vault e dimensione totale di tutti gli archivi nella vault. Amazon Glacier fornisce chiamate API per recuperare queste informazioni per un vault specifico o per tutti i vault in uno specifico account. Regione AWS Per ulteriori informazioni, consulta Recupero dei metadati del vault in Amazon Glacier.

Download di un inventario vault

Un inventario vault fa riferimento all'elenco di archivi in una vault. Per ogni archivio nell'elenco, l'inventario fornisce informazioni quali ID archivio, data di creazione e dimensione. Amazon Glacier

aggiorna l'inventario del vault una volta al giorno, a partire dal giorno in cui il primo archivio viene caricato nel vault. Un inventario vault deve esistere perché sia possibile scaricarlo.

Il download di un inventario vault è un'operazione asincrona. Devi dapprima avviare un processo per poter scaricare l'inventario. Dopo aver ricevuto la richiesta di lavoro, Amazon Glacier prepara l'inventario per il download. Quando il processo è completato, puoi eseguire il download dei dati dell'inventario.

Data la natura asincrona del processo, puoi utilizzare le notifiche di Amazon Simple Notification Service (Amazon SNS) per ricevere notifiche sul completamento del processo. Puoi specificare un argomento di Amazon SNS per ogni singola richiesta di processo o configurare la vault affinché venga inviata una notifica quando si verificano determinati eventi di vault.

Amazon Glacier prepara un inventario per ogni vault ogni 24 ore. Se nessun archivio è stato aggiunto o eliminato nel vault dall'ultimo inventario, la data dell'inventario non viene aggiornata.

Quando avvii un processo per un inventario del vault, Amazon Glacier restituisce l'ultimo inventario generato, che è point-in-time uno snapshot e non dati in tempo reale. A prima vista, recuperare un inventario vault per ogni caricamento di archivio può non sembrare utile. Tuttavia, supponiamo che tu gestisca un database sul lato client che contiene i metadati associati agli archivi che carichi su Amazon Glacier. In tal caso, l'inventario vault potrebbe rivelarsi utile per risolvere le differenze tra le informazioni nel database e l'inventario vault effettivo.

Per ulteriori informazioni sul recupero di un inventario vault, consulta Scaricamento di un inventario Vault in Amazon Glacier.

Configurazione delle notifiche di vault

Il recupero di qualsiasi elemento da Amazon Glacier, ad esempio un archivio da un vault o un inventario del vault, è un processo in due fasi. Innanzitutto, devi avviare un processo. Quando questo risulta completato, esegui il download dell'output. Per sapere quando il lavoro è completo, puoi utilizzare le notifiche di Amazon Glacier. Amazon Glacier invia messaggi di notifica a un argomento di Amazon Simple Notification Service (Amazon SNS) fornito da te.

Puoi configurare le notifiche per una vault e identificare gli eventi di vault e l'argomento di Amazon SNS per ricevere le notifiche relative agli eventi che si verificano. Ogni volta che si verifica l'evento vault, Amazon Glacier invia una notifica all'argomento Amazon SNS specificato. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier.

Creazione di un vault in Amazon Glacier

In seguito alla creazione di un vault, si aggiunge un vault al set di vault nel tuo account. An Account AWS può creare fino a 1.000 vault per regione. AWS Per un elenco delle AWS regioni supportate da Amazon Glacier (Amazon Glacier), consulta Regioni ed endpoint nella Guida generale.AWS

Quando crei un vault, devi fornire un nome di vault. Di seguito sono riportati i requisiti di denominazione di vault:

- I nomi possono contenere un numero di caratteri compreso tra 1 e 255.
- I caratteri validi sono a-z, , A-Z0-9, "_" (trattino basso), "-" (trattino) e "." (punto).

I nomi dei vault devono essere univoci all'interno di un account e della AWS regione in cui viene creato il vault. In altre parole, un account può creare casseforti con lo stesso nome in AWS regioni diverse ma non nella stessa regione. AWS

Argomenti

- Creazione di un vault in Amazon Glacier utilizzando il AWS SDK per Java
- Creazione di un vault in Amazon Glacier utilizzando il AWS SDK per .NET
- Creazione di un vault in Amazon Glacier utilizzando l'API REST
- Creazione di un vault utilizzando la console Amazon Glacier
- Creazione di un vault in Amazon Glacier utilizzando il AWS Command Line Interface

Creazione di un vault in Amazon Glacier utilizzando il AWS SDK per Java

L'API di basso livello fornisce metodi per tutte le operazioni di vault, tra cui la creazione e l'eliminazione di vault, l'ottenimento di una descrizione del vault e l'ottenimento di un elenco di vault creati in uno specifico. Regione AWS Di seguito è riportata la procedura per creare una vault mediante il kit AWS SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare un archivio Regione AWS in cui si desidera creare un vault. Tutte le operazioni che esegui utilizzando questo client si applicano a quella Regione AWS.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `CreateVaultRequest`.

Amazon Glacier (Amazon Glacier) richiede di fornire un nome per il vault e l'ID dell'account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.

3. Eseguire il metodo `createVault` fornendo l'oggetto della richiesta come parametro.

   La risposta restituita da Amazon Glacier è disponibile nell'oggetto. `CreateVaultResult`

Il seguente frammento di codice Java illustra la procedura precedente. Il frammento di codice crea un vault nella regione `us-west-2` `Location`Viene stampato l'URI relativo del vault che include l'ID dell'account, il e il nome del Regione AWS vault.

```
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**
>
> Per informazioni sull'API REST sottostante, consulta Create Vault (PUT vault).

Esempio: creazione di un vault utilizzando il AWS SDK per Java

Il seguente esempio di codice Java crea un vault nella `us-west-2` regione (per ulteriori informazioni Regioni AWS, vedereAccesso ad Amazon Glacier). Inoltre, l'esempio di codice recupera le informazioni del vault, elenca tutti gli archivi presenti nello stesso Regione AWS archivio e quindi elimina l'archivio creato.

Per step-by-step istruzioni su come eseguire l'esempio seguente, vedere. Esecuzione di esempi Java per Amazon Glacier con 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);
    }
```

```
}
```

Creazione di un vault in Amazon Glacier utilizzando il AWS SDK per .NET

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for .NET forniscono un metodo per creare un vault.

Argomenti

- Creazione di un vault mediante l'API di alto livello del kit AWS SDK per .NET
- Creazione di un vault utilizzando l'API di basso livello di AWS SDK per .NET

Creazione di un vault mediante l'API di alto livello del kit AWS SDK per .NET

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `CreateVault` che puoi utilizzare per creare una vault in una Regione AWS .

Esempio: operazioni di vault che utilizzano l'API di alto livello di AWS SDK per .NET

L'esempio di codice C# seguente crea ed elimina una vault nella regione Stati Uniti occidentali (Oregon). Per un elenco delle applicazioni Regioni AWS in cui è possibile creare casseforti, vedere. Accesso ad Amazon Glacier

Per step-by-step istruzioni su come eseguire l'esempio seguente, vedereEsecuzione di esempi di codice. Devi aggiornare il codice con un nome di vault come indicato.

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();
    }
  }
}
```

Creazione di un vault utilizzando l'API di basso livello di AWS SDK per .NET

L'API di basso livello fornisce metodi per tutte le operazioni del vault, tra cui la creazione e l'eliminazione di vault, l'ottenimento di una descrizione del vault e l'ottenimento di un elenco di vault creati in un archivio specifico. Regione AWS Di seguito è riportata la procedura per creare una vault mediante il kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare un archivio Regione AWS in cui si desidera creare un vault. Tutte le operazioni che esegui utilizzando questo client si applicano a quella Regione AWS.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `CreateVaultRequest`.

   Amazon Glacier (Amazon Glacier) richiede di fornire un nome per il vault e l'ID dell'account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.
3. Eseguire il metodo `CreateVault` fornendo l'oggetto della richiesta come parametro.

   La risposta restituita da Amazon Glacier è disponibile nell'oggetto. `CreateVaultResponse`

Esempio: operazioni di vault che utilizzano l'API di basso livello di AWS SDK per .NET

L'esempio di codice C# seguente illustra la procedura precedente. In questo esempio viene creato una vault nella regione Stati Uniti occidentali (Oregon). Inoltre, l'esempio di codice recupera le informazioni sull'archivio, elenca tutti gli archivi nello stesso archivio e quindi elimina Regione AWS l'archivio creato. Quello `Location` stampato è l'URI relativo del vault che include l'ID dell'account, il e il nome del Regione AWS vault.

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Create Vault (PUT vault).

Per step-by-step istruzioni su come eseguire l'esempio seguente, vedere. Esecuzione di esempi di codice Devi aggiornare il codice con un nome di vault come indicato.

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);
    }
  }
}
```

Creazione di un vault in Amazon Glacier utilizzando l'API REST

Per creare un vault mediante l'API REST, consulta Create Vault (PUT vault).

Creazione di un vault utilizzando la console Amazon Glacier

Per creare un vault utilizzando la console Amazon Glacier (Amazon Glacier)Fase 2: creazione di un vault in Amazon Glacier, consulta il tutorial Getting Started.

Creazione di un vault in Amazon Glacier utilizzando il AWS Command Line Interface

Segui questi passaggi per creare un vault in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: creazione di un deposito utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: creazione di un deposito utilizzando il AWS CLI

1. Utilizzare il `create-vault` comando per creare un deposito denominato *awsexamplevault* account. *111122223333*

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

   Output previsto:

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

2. Verificare la creazione utilizzando il comando `describe-vault`.

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

Recupero dei metadati del vault in Amazon Glacier

Puoi recuperare informazioni relative a una vault, ad esempio data di creazione, numero di archivi nella vault e dimensione totale di tutti gli archivi nella vault. Amazon Glacier (Amazon Glacier) fornisce chiamate API per recuperare queste informazioni per un vault specifico o per tutti i vault in una regione specifica del tuo account. AWS

Se recuperi un elenco di vault, Amazon Glacier restituisce l'elenco ordinato in base ai valori ASCII dei nomi dei vault. L'elenco contiene fino a 1.000 vault. Devi sempre verificare che nella risposta sia presente un contrassegno indicante il punto da cui continuare l'elenco; se non vi sono altri elementi, il valore del campo relativo al contrassegno è `null`. Puoi eventualmente limitare il numero di vault restituiti nella risposta. Se esistono più vault di quelli restituiti nella risposta, viene eseguita la paginazione del risultato. Devi inviare ulteriori richieste per recuperare il set di vault successivo.

Argomenti

- Recupero dei metadati del vault in Amazon Glacier utilizzando AWS SDK per Java
- Recupero dei metadati del vault in Amazon Glacier utilizzando AWS SDK per .NET
- Recupero di metadati di vault mediante l'API REST

- Recupero dei metadati del vault in Amazon Glacier utilizzando AWS Command Line Interface

Recupero dei metadati del vault in Amazon Glacier utilizzando AWS SDK per Java

Argomenti

- Recupero di metadati di vault per una vault
- Recupero di metadati di vault per tutti i vault in una regione
- Esempio: recupero di metadati di vault mediante Amazon SDK per Java

Recupero di metadati di vault per una vault

Puoi recuperare i metadati per un vault specifico o per tutti i vault in una regione specifica. AWS Di seguito è riportata la procedura per recuperare metadati di vault per una vault specifica utilizzando l'API di basso livello dell'Amazon SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui risiede il vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DescribeVaultRequest`.

   Amazon Glacier (Amazon Glacier) richiede di fornire un nome per il vault e l'ID dell'account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.
3. Eseguire il metodo `describeVault` fornendo l'oggetto della richiesta come parametro.

   Le informazioni sui metadati del vault restituite da Amazon Glacier sono disponibili nell'oggetto. `DescribeVaultResult`

Il seguente frammento di codice Java illustra la procedura precedente.

```
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**
>
> Per informazioni sull'API REST sottostante, consulta Describe Vault (GET vault).

Recupero di metadati di vault per tutti i vault in una regione

Puoi anche utilizzare il `listVaults` metodo per recuperare i metadati per tutti i vault in una regione specifica. AWS

Il seguente frammento di codice Java recupera un elenco di vault nella regione `us-west-2`. La richiesta limita a 5 il numero di vault restituiti nella risposta. Il frammento di codice effettua quindi una serie di `listVaults` chiamate per recuperare l'intero elenco dei vault dalla Regione. AWS

```
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);
```

Nel segmento di codice precedente, se non specifichi il `Limit` valore nella richiesta, Amazon Glacier restituisce fino a 10 vault, come impostato dall'API Amazon Glacier. Se sono presenti più vault da elencare, il campo `marker` della risposta contiene l'ARN (Amazon Resource Name) del vault in cui continuare l'elenco con una nuova richiesta. In caso contrario, il campo `marker` è null.

Nota che le informazioni restituite per ogni vault nell'elenco sono le stesse che ottieni chiamando il metodo `describeVault` per un determinato vault.

> **Note**
>
> Il metodo `listVaults` chiama l'API REST sottostante (vedi List Vaults (GET vaults)).

Esempio: recupero di metadati di vault mediante Amazon SDK per Java

Per un esempio di codice, consulta Esempio: creazione di un vault utilizzando il AWS SDK per Java. L'esempio di codice Java crea un vault e recupera i metadati di vault.

Recupero dei metadati del vault in Amazon Glacier utilizzando AWS SDK per .NET

Argomenti

- Recupero di metadati di vault per una vault
- Recupero di metadati di vault per tutti i vault in una regione
- Esempio: recupero dei metadati di Vault utilizzando l'API di basso livello del AWS SDK per .NET

Recupero di metadati di vault per una vault

Puoi recuperare i metadati per un archivio specifico o per tutti gli archivi in una regione specifica. AWS Di seguito è riportata la procedura per recuperare metadati per un determinato vault utilizzando l'API di basso livello del kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui risiede il vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DescribeVaultRequest`.

   Amazon Glacier (Amazon Glacier) richiede di fornire un nome per il vault e l'ID dell'account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.

3. Eseguire il metodo `DescribeVault` fornendo l'oggetto della richiesta come parametro.

   Le informazioni sui metadati del vault restituite da Amazon Glacier sono disponibili nell'oggetto. `DescribeVaultResult`

Il seguente frammento di codice C# illustra la procedura precedente. Il frammento di codice recupera informazioni sui metadati di una vault esistente nella regione Stati Uniti occidentali (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**
>
> Per informazioni sull'API REST sottostante, consulta Describe Vault (GET vault).

Recupero di metadati di vault per tutti i vault in una regione

Puoi anche utilizzare il `ListVaults` metodo per recuperare i metadati per tutti i vault in una regione specifica. AWS

Il seguente frammento di codice C# recupera un elenco di vault nella regione Stati Uniti occidentali (Oregon). La richiesta limita a 5 il numero di vault restituiti nella risposta. Il frammento di codice effettua quindi una serie di `ListVaults` chiamate per recuperare l'intero elenco dei vault dalla Regione. AWS

```
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);
```

Nel segmento di codice precedente, se non specifichi il `Limit` valore nella richiesta, Amazon Glacier restituisce fino a 10 vault, come impostato dall'API Amazon Glacier.

Nota che le informazioni restituite per ogni vault nell'elenco sono le stesse che ottieni chiamando il metodo `DescribeVault` per un determinato vault.

> **Note**
>
> Il metodo `ListVaults` chiama l'API REST sottostante (vedi List Vaults (GET vaults)).

Esempio: recupero dei metadati di Vault utilizzando l'API di basso livello del AWS SDK per .NET

Per un esempio di codice, consulta Esempio: operazioni di vault che utilizzano l'API di basso livello di AWS SDK per .NET. L'esempio di codice C# crea un vault e ne recupera i metadati.

Recupero di metadati di vault mediante l'API REST

Per elencare i vault mediante l'API REST, consulta List Vaults (GET vaults). Per ottenere informazioni relative a un vault, consulta Describe Vault (GET vault).

Recupero dei metadati del vault in Amazon Glacier utilizzando AWS Command Line Interface

Questo esempio mostra come recuperare le informazioni e i metadati del vault in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: recupero dei metadati del Vault utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface
2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

- Prova a utilizzare il comando help.

```
aws help
```

- Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

- Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

```
aws configure list
```

Esempio: recupero dei metadati del Vault utilizzando il AWS CLI

- Usa il `describe-vault` comando per descrivere un archivio denominato account. *awsexamplevault* *111122223333*

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

Scaricamento di un inventario Vault in Amazon Glacier

Dopo aver caricato il primo archivio nel vault, Amazon Glacier (Amazon Glacier) crea automaticamente un inventario del vault e quindi lo aggiorna circa una volta al giorno. Dopo che Amazon Glacier ha creato il primo inventario, in genere è necessaria mezza giornata e fino a un giorno prima che l'inventario sia disponibile per il recupero. Puoi recuperare un inventario del vault da Amazon Glacier con la seguente procedura in due fasi:

1. Avvio di un processo di recupero di un inventario mediante l'operazione Initiate Job (POST jobs).

> **Important**
>
> L'utilizzo di una policy di recupero dati può causare un errore nella richiesta di avvio di un processo di recupero e generare l'eccezione `PolicyEnforcedException`. Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon

Glacier. Per ulteriori informazioni sull'eccezione `PolicyEnforcedException`, consulta Risposte agli errori.

2. Download dei byte mediante l'operazione Get Job Output (GET output) dopo il completamento del processo.

Ad esempio, il recupero di un archivio o di un inventario vault richiede dapprima l'avvio di un processo di recupero. La richiesta di processo viene eseguita in modo asincrono. Quando avvii un processo di recupero, Amazon Glacier crea un processo e restituisce un ID del lavoro nella risposta. Quando Amazon Glacier completa il processo, puoi ottenere l'output del lavoro, i byte di archivio o i dati di inventario del vault.

Per ottenere l'output di un processo, è necessario che il processo sia stato completato. Per determinare lo stato di un processo, puoi procedere come segue:

- Attendi la notifica di completamento del lavoro: puoi specificare un argomento di Amazon Simple Notification Service (Amazon SNS) su cui Amazon Glacier può pubblicare una notifica dopo il completamento del lavoro. Puoi specificare l'argomento di Amazon SNS utilizzando i metodi seguenti:
  - Specifica un argomento di Amazon SNS per processo.

    Quando avvii un processo, puoi eventualmente specificare un argomento di Amazon SNS.
  - Imposta la configurazione delle notifiche per il vault.

    Puoi impostare la configurazione delle notifiche per specifici eventi sulla vault (consulta Configurazione delle notifiche Vault in Amazon Glacier). Amazon Glacier invia un messaggio all'argomento SNS specificato ogni volta che si verifica l'evento specifico.

  Se hai impostato la configurazione delle notifiche nel vault e specifichi anche un argomento di Amazon SNS quando avvii un processo, Amazon Glacier invia un messaggio di completamento del processo a entrambi gli argomenti.

  Puoi configurare l'argomento di SNS per ricevere notifiche tramite e-mail oppure memorizzare il messaggio in una coda Amazon Simple Queue Service (Amazon SQS) per eseguirne il polling mediante l'applicazione. Quando un messaggio è visualizzato nella coda, puoi verificare se il completamento del processo è riuscito e quindi scaricare l'output del processo.

- Richiedi informazioni sul lavoro in modo esplicito: Amazon Glacier fornisce anche una descrizione dell'operazione di lavoro (Describe Job (GET JobID)) che ti consente di effettuare sondaggi per ottenere informazioni sul lavoro. Puoi inviare periodicamente questa richiesta per ottenere tali informazioni. Tuttavia, l'utilizzo delle notifiche di Amazon SNS è l'opzione consigliata.

> **Note**
>
> Le informazioni che ottieni via le notifiche di SNS sono identiche a quelle che ottieni chiamando un'operazione Describe Job.

Argomenti

- Informazioni sull'inventario
- Scaricare un inventario Vault in Amazon Glacier utilizzando il AWS SDK per Java
- Scaricamento di un inventario Vault in Amazon Glacier utilizzando il AWS SDK per .NET
- Download di un inventario vault mediante l'API REST
- Scaricamento di un inventario Vault in Amazon Glacier utilizzando AWS Command Line Interface

Informazioni sull'inventario

Amazon Glacier aggiorna un inventario del vault almeno una volta al giorno, a partire dal giorno in cui carichi per la prima volta un archivio nel vault. Se nessun archivio è stato aggiunto o eliminato nel vault dall'ultimo inventario, la data dell'inventario non viene aggiornata. Quando avvii un processo per un inventario del vault, Amazon Glacier restituisce l'ultimo inventario generato, che è point-in-time uno snapshot e non dati in tempo reale. Tieni presente che dopo che Amazon Glacier ha creato il primo inventario per il vault, in genere è necessaria mezza giornata e fino a un giorno prima che l'inventario sia disponibile per il recupero.

A prima vista, recuperare un inventario vault per ogni caricamento di archivio può non sembrare utile. Tuttavia, supponiamo che tu gestisca un database sul lato client che associa i metadati sugli archivi che carichi su Amazon Glacier. In tal caso, l'inventario vault potrebbe essere utilizzato per risolvere le differenze tra le informazioni nel database e l'inventario vault effettivo. Puoi limitare il numero di elementi di inventario recuperati filtrandoli per data di creazione dell'archivio o impostando una quota. Per ulteriori informazioni sulla limitazione del recupero di inventario, consulta Recupero di un intervallo di inventario.

L'inventario può essere restituito in due formati: CSV o JSON. Puoi eventualmente specificare il formato quando avvii il processo di inventario. Il formato di default è JSON. Per ulteriori informazioni sui campi dati restituiti nell'output di un processo di inventario, consulta la sezione Corpo della risposta nella pagina relativa all'API Get Job Output.

Scaricare un inventario Vault in Amazon Glacier utilizzando il AWS SDK per Java

Di seguito è riportata la procedura per recuperare un inventario vault utilizzando l'API di basso livello AWS SDK per Java. L'API di alto livello non supporta il recupero di un inventario vault.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui risiede il vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Avvia un processo di recupero di inventario eseguendo il metodo `initiateJob`.

   Esegui `initiateJob` fornendo informazioni sul processo in un oggetto `InitiateJobRequest`.

   > **Note**
   >
   > Nota che se un inventario non è stato completato per il vault, viene restituito un errore. Amazon Glacier (Amazon Glacier) prepara un inventario per ogni vault periodicamente, ogni 24 ore.

   Amazon Glacier restituisce un Job ID in risposta. La risposta è disponibile in un'istanza della 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. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

   Puoi anche interrogare Amazon Glacier `describeJob` chiamando il metodo per determinare lo stato di completamento del lavoro. La soluzione consigliata è comunque quella di utilizzare un argomento di Amazon SNS per le notifiche. L'esempio di codice fornito nella sezione seguente utilizza Amazon SNS per Amazon Glacier per pubblicare un messaggio.

4. Scarica l'output del processo (dati dell'inventario vault) eseguendo il metodo `getJobOutput`.

   Per fornire l'ID del tuo account, il nome del vault e il job ID, devi creare un'istanza della classe `GetJobOutputRequest`. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.

   L'output restituito da Amazon Glacier è disponibile nell'oggetto. `GetJobOutputResult`

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

> **Note**
>
> Per informazioni sull'API REST sottostante relativa al processo, consulta Operazioni di processo.

Esempio: recupero di un inventario vault mediante Amazon SDK per Java

L'esempio di codice Java seguente recupera l'inventario vault per il vault specificato.

L'esempio esegue le seguenti operazioni:

- Crea un argomento ARN Amazon Simple Notification Service (Amazon SNS).

  Amazon Glacier invia una notifica su questo argomento una volta completato il processo.
- Crea una coda Amazon Simple Queue Service (Amazon SQS).

  L'esempio collega una policy alla coda per consentire all'argomento di Amazon SNS di pubblicare messaggi nella coda.
- Avvia un processo per scaricare l'archivio specificato.

  Nella richiesta di lavoro, viene specificato l'argomento Amazon SNS che è stato creato in modo che Amazon Glacier possa pubblicare una notifica sull'argomento dopo il completamento del processo.
- Verifica la presenza di un messaggio contenente l'ID processo nella coda di Amazon SQS.

  Se il messaggio esiste, analizza il codice JSON e verifica se il completamento del processo è riuscito. Se è il caso, scarica l'archivio.
- Esegue una pulizia eliminando l'argomento di Amazon SNS e la coda di Amazon SQS creata.

```
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));
    }
}
```

Scaricamento di un inventario Vault in Amazon Glacier utilizzando il AWS SDK per .NET

Di seguito è riportata la procedura per recuperare un inventario vault utilizzando l'API di basso livello AWS SDK per .NET. L'API di alto livello non supporta il recupero di un inventario vault.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una regione in cui risiede il vault. AWS Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Avvia un processo di recupero di inventario eseguendo il metodo `InitiateJob`.

   Le informazioni sul processo devono essere fornite in un oggetto `InitiateJobRequest`. Amazon Glacier (Amazon Glacier) restituisce un Job ID in risposta. La risposta è disponibile in un'istanza della 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. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica sul vault che identifica un argomento di Amazon Simple Notification Service (Amazon SNS) o hai specificato un argomento Amazon SNS quando hai avviato un lavoro, Amazon Glacier invia un messaggio a quell'argomento dopo il completamento del processo. L'esempio di codice fornito nella sezione seguente utilizza Amazon SNS per Amazon Glacier per pubblicare un messaggio.

   Puoi anche interrogare Amazon Glacier `DescribeJob` chiamando il metodo per determinare lo stato di completamento del lavoro. La soluzione consigliata è comunque quella di utilizzare un argomento di Amazon SNS per le notifiche.

4. Scarica l'output del processo (dati dell'inventario vault) eseguendo il metodo `GetJobOutput`.

   Per fornire l'ID del tuo account, il nome di vault e il job ID, devi creare un'istanza della classe `GetJobOutputRequest`. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.

   L'output restituito da Amazon Glacier è disponibile nell'oggetto. `GetJobOutputResponse`

```
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**
>
> Per informazioni sull'API REST sottostante relativa al processo, consulta Operazioni di processo.

Esempio: recupero di un inventario Vault utilizzando l'API di basso livello di AWS SDK per .NET

L'esempio di codice C# seguente recupera un inventario vault per il vault specificato.

L'esempio esegue le seguenti operazioni:

- Configura un argomento di Amazon SNS.

  Amazon Glacier invia una notifica su questo argomento una volta completato il processo.
- Configura una coda Amazon SQS.

  L'esempio collega una policy alla coda per consentire all'argomento di Amazon SNS di pubblicare messaggi.
- Avvia un processo per scaricare l'archivio specificato.

  Nella richiesta di lavoro, l'esempio specifica l'argomento Amazon SNS in modo che Amazon Glacier possa inviare un messaggio dopo il completamento del processo.
- Verifica periodicamente la presenza di un messaggio nella coda di Amazon SQS.

Se il messaggio esiste, analizza il codice JSON e verifica se il completamento del processo è riuscito. Se è il caso, scarica l'archivio. L'esempio di codice utilizza la libreria JSON.NET (vedi JSON.NET) per analizzare il codice JSON.

- Esegue una pulizia eliminando l'argomento di Amazon SNS e la coda di Amazon SQS creata.

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);
      }
    }
  }
}
```

Download di un inventario vault mediante l'API REST

Download di un inventario vault mediante l'API REST

Il download di un inventario vault è un processo in due fasi.

1. Avvia un processo di tipo `inventory-retrieval`. Per ulteriori informazioni, consulta Initiate Job (POST jobs).
2. Una volta completato il processo, scarica i dati dell'inventario. Per ulteriori informazioni, consulta Get Job Output (GET output).

Scaricamento di un inventario Vault in Amazon Glacier utilizzando AWS Command Line Interface

Segui questi passaggi per scaricare un inventario del vault in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: scaricamento di un inventario del vault utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: scaricamento di un inventario del vault utilizzando il AWS CLI

1. Utilizzare il comando `initiate-job` per avviare un'attività di recupero inventario.

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

   Output previsto:

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

2. Utilizzare il comando `describe-job` per verificare lo stato dell'attività di recupero precedente.

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

Output previsto:

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

3. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. L'ID del lavoro non scade per almeno 24 ore dopo il completamento del lavoro da parte di Amazon Glacier. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

   Puoi impostare la configurazione delle notifiche per specifici eventi sul vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier. Amazon Glacier invia un messaggio all'argomento SNS specificato ogni volta che si verificano eventi specifici.

4. Al termine, utilizzare il comando `get-job-output` per scaricare l'attività di recupero nel file `output.json`.

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

Questo comando produce un file con i seguenti campi.

```
{
"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":
...
]}
```

Configurazione delle notifiche Vault in Amazon Glacier

Il recupero di qualsiasi elemento da Amazon Glacier, ad esempio un archivio da un vault o un inventario del vault, è un processo in due fasi.

1. Avvia un processo di recupero.
2. Quando il processo risulta completato, esegui il download dell'output del processo.

Puoi impostare la configurazione delle notifiche per una vault di modo che un messaggio venga inviato a un argomento di Amazon Simple Notification Service (Amazon SNS) al completamento del processo.

Argomenti

- Configurazione delle notifiche Vault in Amazon Glacier: concetti generali
- Configurazione delle notifiche del vault in Amazon Glacier utilizzando AWS SDK per Java
- Configurazione delle notifiche Vault in Amazon Glacier utilizzando AWS SDK per .NET
- Configurazione delle notifiche Vault in Amazon Glacier utilizzando l'API REST
- Configurazione delle notifiche Vault utilizzando la console Amazon Glacier
- Configurazione delle notifiche di vault mediante l' AWS Command Line Interface

Configurazione delle notifiche Vault in Amazon Glacier: concetti generali

Una richiesta di lavoro di recupero di Amazon Glacier viene eseguita in modo asincrono. È necessario attendere che Amazon Glacier completi il processo prima di poterne ottenere l'output. Puoi effettuare sondaggi periodici su Amazon Glacier per determinare lo stato del lavoro, ma questo non è un approccio ottimale. Amazon Glacier supporta anche le notifiche. Al completamento di un processo, il processo può pubblicare un messaggio su un argomento di Amazon Simple Notification Service (Amazon SNS). L'utilizzo di questa funzionalità richiede la configurazione delle notifiche sulla vault. Nella configurazione, identifichi uno o più eventi e un argomento Amazon SNS a cui desideri che Amazon Glacier invii un messaggio quando si verifica l'evento.

Amazon Glacier definisce gli eventi specificamente correlati al completamento del lavoro `ArchiveRetrievalCompleted` (`InventoryRetrievalCompleted`,) che puoi aggiungere alla configurazione delle notifiche del vault. Quando viene completato un processo specifico, Amazon Glacier pubblica un messaggio di notifica sull'argomento SNS.

La configurazione delle notifiche è un documento JSON come mostrato nell'esempio seguente.

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

Nota che puoi configurare un solo argomento di Amazon SNS per una vault.

> **Note**
>
> L'aggiunta di una configurazione di notifica a un vault fa sì che Amazon Glacier invii una notifica ogni volta che si verifica l'evento specificato nella configurazione di notifica. Puoi anche eventualmente specificare un argomento di Amazon SNS in ogni richiesta di avvio di processo. Se aggiungi sia la configurazione delle notifiche nel vault che specifichi anche un argomento Amazon SNS nella richiesta di avvio del lavoro, Amazon Glacier invia entrambe le notifiche.

Il messaggio di completamento del lavoro inviato da Amazon Glacier include informazioni come il tipo di processo `InventoryRetrieval` (`ArchiveRetrieval`,), lo stato di completamento del lavoro, il

nome dell'argomento SNS, il codice di stato del processo e l'ARN del vault. Di seguito è riportato un esempio di notifica inviata da Amazon Glacier a un argomento SNS dopo `InventoryRetrieval` il completamento di un processo.

```
{
 "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"
}
```

Se il valore del campo `Completed` è true, devi verificare anche il valore del campo `StatusCode` per determinare se il processo è stato completato o presenta degli errori.

> **Note**
>
> Nota che l'argomento di Amazon SNS deve consentire alla vault di pubblicare una notifica. Per impostazione predefinita, solo il proprietario dell'argomento di Amazon SNS può pubblicare un messaggio nell'argomento. Tuttavia, se l'argomento Amazon SNS e il vault sono di proprietà di diversi Account AWS, è necessario configurare l'argomento Amazon SNS per accettare le pubblicazioni dal vault. Puoi configurare la policy di argomenti di Amazon SNS nella console Amazon SNS.

Per ulteriori informazioni su Amazon SNS, consulta Nozioni di base su Amazon SNS.

Configurazione delle notifiche del vault in Amazon Glacier utilizzando AWS SDK per Java

Di seguito è riportata la procedura per configurare le notifiche per una vault utilizzando l'API di basso livello del kit AWS SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui risiede il vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Fornisci informazioni sulla configurazione delle notifiche creando un'istanza della classe `SetVaultNotificationsRequest`.

   Devi fornire il nome della vault, informazioni sulla configurazione delle notifiche e l'ID account. Quando specifichi una configurazione delle notifiche, fornisci il nome della risorsa Amazon (ARN) di un argomento di Amazon SNS esistente e uno o più eventi per i quali desideri ricevere una notifica. Per un elenco degli eventi supportati, consulta Set Vault Notification Configuration (PUT notification-configuration).

3. Eseguire il metodo `setVaultNotifications` fornendo l'oggetto della richiesta come parametro.

Il seguente frammento di codice Java illustra la procedura precedente. Il frammento imposta una configurazione delle notifiche per un vault. La configurazione richiede ad Amazon Glacier (Amazon Glacier) di inviare una notifica all'argomento Amazon SNS specificato quando si verifica l'evento o l'evento. `ArchiveRetrievalCompleted` `InventoryRetrievalCompleted`

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

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Operazioni di vault.

Esempio: impostazione della configurazione delle notifiche su un vault utilizzando AWS SDK per Java

L'esempio di codice Java seguente imposta la configurazione delle notifiche per un vault, elimina la configurazione e quindi la ripristina. Per step-by-step istruzioni su come eseguire l'esempio seguente, vedereUtilizzo di AWS SDK per Java con 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);
    }
}
```

Configurazione delle notifiche Vault in Amazon Glacier utilizzando AWS SDK per .NET

Di seguito è riportata la procedura per configurare le notifiche per una vault utilizzando l'API di basso livello del kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una regione in cui risiede il vault AWS . Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Fornisci informazioni sulla configurazione delle notifiche creando un'istanza della classe `SetVaultNotificationsRequest`.

   Devi fornire il nome della vault, informazioni sulla configurazione delle notifiche e l'ID account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.

   Quando specifichi una configurazione delle notifiche, fornisci il nome della risorsa Amazon (ARN) di un argomento di Amazon SNS esistente e uno o più eventi per i quali desideri ricevere una notifica. Per un elenco degli eventi supportati, consulta Set Vault Notification Configuration (PUT notification-configuration).

3. Eseguire il metodo `SetVaultNotifications` fornendo l'oggetto della richiesta come parametro.

4. Dopo l'impostazione della configurazione delle notifiche per un vault, puoi recuperare le informazioni sulla configurazione chiamando il metodo `GetVaultNotifications` e rimuoverle chiamando il metodo `DeleteVaultNotifications` fornito dal client.

Esempio: impostazione della configurazione delle notifiche su un vault utilizzando il AWS SDK per .NET

L'esempio di codice C# seguente illustra la procedura precedente. Questo esempio imposta la configurazione delle notifiche per la vault ("`examplevault`") nella regione Stati Uniti occidentali (Oregon), recupera la configurazione e quindi la elimina. La configurazione richiede ad Amazon Glacier (Amazon Glacier) di inviare una notifica all'argomento Amazon SNS specificato quando si verifica l'evento o l'evento. `ArchiveRetrievalCompleted` `InventoryRetrievalCompleted`

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Operazioni di vault.

Per step-by-step istruzioni su come eseguire il seguente esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice e fornire un nome di vault esistente e un argomento di Amazon SNS come indicato.

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);
    }
```

```
  }
}
```

Configurazione delle notifiche Vault in Amazon Glacier utilizzando l'API REST

Per configurare notifiche di vault mediante l'API REST, consulta Set Vault Notification Configuration (PUT notification-configuration). Inoltre, puoi anche ottenere (Get Vault Notifications (GET notification-configuration)) ed eliminare (Delete Vault Notifications (DELETE notification-configuration)) notifiche di vault.

Configurazione delle notifiche Vault utilizzando la console Amazon Glacier

Questa sezione descrive come configurare le notifiche del vault utilizzando la console Amazon Glacier. Quando configuri le notifiche, specifichi eventi di completamento di processi che inviano una notifica a un argomento di Amazon Simple Notification Service (Amazon SNS). Oltre a configurare le notifiche per la vault, puoi anche specificare un argomento in cui pubblicare notifiche quando avvii un processo. Se la vault è configurata per inviare una notifica per un determinato evento e configuri anche le notifiche nella richiesta di avvio di un processo, vengono inviate due notifiche.

Configurazione di una notifica di vault

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. Nel pannello di navigazione a sinistra, seleziona Vault.
3. Nell'elenco Vault, seleziona una vault.
4. Nella sezione Notifiche, scegli Modifica.
5. Nella pagina Notifiche evento, seleziona Attivazione delle notifiche.
6. Nella sezione Notifiche, seleziona una delle seguenti opzioni di Amazon Simple Notification Service (Amazon SNS), quindi segui i passaggi corrispondenti:

Opzioni Amazon SNS	Azione
Crea un nuovo argomento SNS	1. Seleziona Crea un nuovo argomento. 2.

Opzioni Amazon SNS	Azione
	Per Nome argomento immetti un nome per il nuovo argomento. I nomi argomento possono contenere fino a 256 caratteri. Sono consentiti caratteri alfanumerici, trattini (-) e trattini bassi (_). I nomi degli argomenti devono essere univoci all'interno dell'account e. Regione AWS 3. (Facoltativo) Se desideri iscriverti all'argomento utilizzando i messaggi SMS, inserisci un nome per Nome visualizzato. Un nome visualizzato può avere fino a 100 caratteri.

Opzioni Amazon SNS	Azione
Scegli un argomento SNS esistente	1. Seleziona Scegli un argomento SNS esistente. 2. In Specifica argomento SNS, seleziona una delle seguenti opzioni: • Scegli tra i tuoi argomenti SNS Viene visualizzato l'elenco a tendina Argomenti SNS. Seleziona un argomento esistente dall'elenco a tendina. • Inserisci ARN dell'argomento SNS Viene visualizzata la casella di testo ARN dell'argomento di Amazon SNS. Inserisci il nome della risorsa Amazon (ARN) per l'argomento Amazon SNS L'ARN di un argomento SNS ha il formato seguente: `arn:aws:sns: region:account-id :topic-name` L'ARN di un argomento SNS si trova nella console Amazon SNS.

7. In Eventi, seleziona uno o entrambi gli eventi a cui desideri inviare notifiche:

   - Per inviare una notifica solo quando i lavori di recupero dell'archivio sono completi, seleziona Processo di recupero archivio completato.
   - Per inviare una notifica solo quando i lavori di inventario vault sono completi, seleziona Processo di recupero di inventario vault completato.

Configurazione delle notifiche di vault mediante l' AWS Command Line Interface

Questa sezione descrive come configurare le notifiche di vault utilizzando l' AWS Command Line Interface. Quando configuri le notifiche, specifichi eventi di completamento di processi che inviano notifiche a un argomento di Amazon Simple Notification Service (Amazon SNS). Oltre a configurare le notifiche per il vault, puoi anche specificare un argomento in cui pubblicare notifiche quando avvii un processo. Se il vault è configurato per segnalare un determinato evento e specifichi una notifica nella richiesta di avvio di processo, vengono inviate due notifiche.

Segui questi passaggi per configurare la notifica della vault utilizzando l' AWS CLI.

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: configura le notifiche del Vault utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

- Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

```
aws configure list
```

Esempio: configura le notifiche del Vault utilizzando il AWS CLI

1. Utilizza il comando `set-vault-notifications` per configurare le notifiche che verranno inviate quando si verificano eventi specifici in una vault. Per impostazione predefinita, non ricevi alcuna notifica.

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

2. La configurazione delle notifiche è un documento JSON come mostrato nell'esempio seguente.

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

   Per ulteriori informazioni sull'utilizzo di Amazon SNS per Amazon Glacier, consulta, Configurazione delle notifiche Vault in Amazon Glacier: concetti generali

   Per ulteriori informazioni su Amazon SNS, consulta Nozioni di base su Amazon SNS.

Eliminazione di un vault in Amazon Glacier

Amazon Glacier (Amazon Glacier) elimina un vault solo se non ci sono archivi nel vault a partire dall'ultimo inventario calcolato e non ci sono state operazioni di scrittura nel vault dall'ultimo inventario. Per informazioni sull'eliminazione di archivi, consulta Eliminazione di un archivio in Amazon Glacier. Per informazioni su come scaricare un inventario vault, consulta Scaricamento di un inventario Vault in Amazon Glacier.

> **Note**
>
> Amazon Glacier prepara un inventario per ogni vault ogni 24 ore. Poiché l'inventario potrebbe non riflettere le informazioni più recenti, Amazon Glacier garantisce che il vault sia effettivamente vuoto controllando se sono state eseguite operazioni di scrittura dall'ultimo inventario del vault.

> **Note**
>
> Per l'eliminazione automatica degli archivi del vault, consulta Eliminazione automatica degli archivi del vault in Amazon S3 Glacier.

Argomenti

- Eliminazione di un vault in Amazon Glacier utilizzando il AWS SDK per Java
- Eliminazione di un vault in Amazon Glacier utilizzando AWS SDK per .NET
- Eliminazione di un vault in Amazon Glacier utilizzando l'API REST
- Eliminazione di un vault vuoto utilizzando la console Amazon Glacier
- Eliminazione di un vault in Amazon Glacier utilizzando il AWS Command Line Interface

Eliminazione di un vault in Amazon Glacier utilizzando il AWS SDK per Java

Di seguito è riportata la procedura per eliminare un vault mediante l'API di basso livello del kit AWS SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   Devi specificare una AWS regione da cui desideri eliminare un vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DeleteVaultRequest`.

   Devi fornire il nome della vault e l'ID account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.

3. Eseguire il metodo `deleteVault` fornendo l'oggetto della richiesta come parametro.

   Amazon Glacier (Amazon Glacier) elimina il vault solo se è vuoto. Per ulteriori informazioni, consulta Delete Vault (DELETE vault).

Il seguente frammento di codice Java illustra la procedura precedente.

```
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**
>
> Per informazioni sull'API REST sottostante, consulta Delete Vault (DELETE vault).

Esempio: eliminazione di un vault utilizzando il AWS SDK per Java

Per un esempio di codice, consulta Esempio: creazione di un vault utilizzando il AWS SDK per Java. L'esempio di codice Java illustra le operazioni di base relative ai vault, inclusa la creazione e l'eliminazione.

Eliminazione di un vault in Amazon Glacier utilizzando AWS SDK per .NET

Sia il livello alto che quello basso APIs forniti da Amazon SDK for .NET forniscono un metodo per eliminare un vault.

Argomenti

- Eliminazione di un vault utilizzando l'API di alto livello di AWS SDK per .NET
- Eliminazione di un vault utilizzando l'API di basso livello di AWS SDK per .NET

Eliminazione di un vault utilizzando l'API di alto livello di AWS SDK per .NET

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `DeleteVault` che puoi utilizzare per eliminare un vault.

Esempio: eliminazione di un vault utilizzando l'API di alto livello di AWS SDK per .NET

Per un esempio di codice, consulta Esempio: operazioni di vault che utilizzano l'API di alto livello di AWS SDK per .NET. L'esempio di codice C# illustra le operazioni di base relative alle vault, inclusa la creazione e l'eliminazione.

Eliminazione di un vault utilizzando l'API di basso livello di AWS SDK per .NET

Di seguito è riportata la procedura per eliminare un vault mediante il kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione da cui si desidera eliminare un vault. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DeleteVaultRequest`.

   Devi fornire il nome della vault e l'ID account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.
3. Eseguire il metodo `DeleteVault` fornendo l'oggetto della richiesta come parametro.

   Amazon Glacier (Amazon Glacier) elimina il vault solo se è vuoto. Per ulteriori informazioni, consulta Delete Vault (DELETE vault).

Il seguente frammento di codice C# illustra la procedura precedente. Lo snippet recupera le informazioni sui metadati di un archivio esistente nella regione predefinita. AWS

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

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

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

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Delete Vault (DELETE vault).

Esempio: eliminazione di un vault utilizzando l'API di basso livello del AWS SDK per .NET

Per un esempio di codice, consulta Esempio: operazioni di vault che utilizzano l'API di basso livello di AWS SDK per .NET. L'esempio di codice C# illustra le operazioni di base relative alle vault, inclusa la creazione e l'eliminazione.

Eliminazione di un vault in Amazon Glacier utilizzando l'API REST

Per eliminare un vault mediante l'API REST, consulta Delete Vault (DELETE vault).

Eliminazione di un vault vuoto utilizzando la console Amazon Glacier

> **Note**
>
> Prima di eliminare una vault, è necessario innanzitutto eliminare tutti gli archivi esistenti all'interno della vault. Puoi farlo scrivendo il codice per effettuare una richiesta di eliminazione dell'archivio utilizzando l'API REST, the AWS SDK per Java AWS SDK per .NET, o usando il AWS Command Line Interface ().AWS CLI Per informazioni sull'eliminazione di archivi, consulta Passaggio 5: eliminare un archivio da un vault in Amazon Glacier.

Dopo che la vault è vuota, puoi eliminarla utilizzando la procedura seguente.

Per eliminare un vault vuoto utilizzando la console Amazon Glacier

1. Accedi Console di gestione AWS e apri la console Amazon Glacier su Amazon Glacier Console.
2. In Seleziona una regione, scegli Regione AWS dove si trova il vault.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nell'elenco Vault, seleziona l'opzione accanto al nome della vault che desideri eliminare, quindi scegli Elimina nella parte superiore della pagina.

5. Nella finestra di dialogo Elimina vault, conferma che desideri eliminare la vault scegliendo Elimina.

> **Important**
>
> L'eliminazione di una vault non può essere annullata.

6. Per verificare di aver eliminato la vault, apri l'elenco Vault e inserisci il nome della vault eliminata. Se la vault non appare tra i risultati, la cancellazione si è conclusa correttamente.

Eliminazione di un vault in Amazon Glacier utilizzando il AWS Command Line Interface

Puoi eliminare vault vuoti e non vuoti in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: eliminazione di un archivio vuoto utilizzando il AWS CLI
- Esempio: eliminazione di un deposito non vuoto utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

```
aws help
```

- Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

- Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

  ```
  aws configure list
  ```

Esempio: eliminazione di un archivio vuoto utilizzando il AWS CLI

- Utilizzare il comando `delete-vault` per eliminare un vault che non contiene archivi.

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

Esempio: eliminazione di un deposito non vuoto utilizzando il AWS CLI

Amazon Glacier elimina un vault solo se non ci sono archivi nel vault all'ultimo inventario calcolato e non sono state effettuate operazioni di scrittura nel vault dall'ultimo inventario. L'eliminazione di un vault non vuoto è un processo in tre fasi: recupero dell'archivio IDs dal rapporto di inventario di un vault, eliminazione di ogni archivio e quindi eliminazione del vault.

1. Utilizzare il comando `initiate-job` per avviare un'attività di recupero inventario.

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

   Output previsto:

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

2. Utilizzare il comando `describe-job` per verificare lo stato dell'attività di recupero precedente.
   ```

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

Output previsto:

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

3. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

   Puoi impostare la configurazione delle notifiche per specifici eventi sulla vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier. Amazon Glacier invia un messaggio all'argomento SNS specificato ogni volta che si verifica l'evento specifico.

4. Al termine, utilizzare il comando `get-job-output` per scaricare l'attività di recupero nel file `output.json`.

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

   Questo comando produce un file con i seguenti campi.

   ```
   {
   ```

```
"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. Utilizzare il comando `delete-archive` per eliminare tutti gli archivi da una vault finché non ne rimane nessuno.

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

> **Note**
>
> Se l'ID dell'archivio inizia con un trattino o un altro carattere speciale, devi racchiudere l'ID dell'archivio tra virgolette per eseguire questo comando.

6. Utilizzare il comando `initiate-job` per avviare una nuova attività di recupero inventario.

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

7. Al termine, utilizzare il comando `delete-vault` per eliminare un vault senza archivi.

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

Etichettatura dei vault Amazon Glacier

Puoi assegnare i tuoi metadati ai vault di Amazon Glacier sotto forma di tag. Un tag è una coppia chiave-valore che definisci per un vault. Per informazioni di base sul tagging, incluse le limitazioni relative ai tag, consulta Etichettare le risorse di Amazon Glacier.

Gli argomenti seguenti descrivono come aggiungere, elencare e rimuovere tag per i vault.

Argomenti

- Etichettatura degli archivi tramite la console Amazon Glacier
- Etichettatura degli archivi utilizzando il AWS CLI
- Etichettatura degli archivi tramite l'API Amazon Glacier
- Sezioni correlate

Etichettatura degli archivi tramite la console Amazon Glacier

Puoi aggiungere, elencare e rimuovere tag utilizzando la console Amazon Glacier, come descritto nelle seguenti procedure.

Visualizzazione di tag per un vault

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nell'elenco Vault, seleziona una vault.
5. Seleziona la scheda Proprietà vault. Scorri fino alla sezione Tag per visualizzare i tag associati alla vault.

Aggiunta di un tag a un vault

È possibile associare fino a 50 tag a una vault. I tag associati a una vault devono avere chiavi di tag univoche.

Per ulteriori informazioni sulle restrizioni relative ai tag, consulta Tagging Amazon Glacier Resources.

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nell'elenco Vault, scegli il nome della vault a cui desideri aggiungere i tag.

5. Seleziona la scheda Proprietà vault.
6. Nel riquadro Tag, selezionare Aggiungi. Viene visualizzata la pagina Aggiungi tag.
7. Nella pagina Aggiungi tag, specifica la chiave tag nel campo Chiave e, facoltativamente, specifica un valore di tag nel campo Valore.
8. Scegli Save changes (Salva modifiche).

Per modificare un tag

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nell'elenco Vault, scegli un nome di vault.
5. Seleziona la scheda Proprietà vault, quindi scorri verso il basso fino alla sezione Tag.
6. In Tag, seleziona la casella di controllo accanto ai tag che desideri modificare, quindi scegli Modifica. Viene visualizzata la pagina Modifica tag.
7. Aggiorna la chiave del tag nel campo Chiave e, facoltativamente, aggiorna il valore del tag nel campo Valore.
8. Scegli Save changes (Salva modifiche).

Rimozione di un tag da un vault

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nell'elenco Vault, seleziona il nome della vault dal quale desideri rimuovere i tag.
5. Seleziona la scheda Proprietà vault. Scorri verso il basso fino alla sezione Tag.
6. In Tag, seleziona la casella di controllo accanto ai tag che desideri rimuovere, quindi seleziona Elimina.
7. Viene visualizzata la finestra di dialogo Elimina tag. Per confermare che desideri eliminare i tag selezionati, seleziona Elimina.

Etichettatura degli archivi utilizzando il AWS CLI

Segui questi passaggi per aggiungere, elencare o rimuovere tag utilizzando AWS Command Line Interface ()AWS CLI.

Ogni tag consiste di una chiave e un valore. Ogni vault può avere fino a 50 tag.

1. Per aggiungere i tag alla vault, usa il comando `add-tags-to-vault`.

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

   Per ulteriori informazioni su questa operazione di vault, consulta Aggiungi tag al vault.

2. Per elencare tutti i tag associati a una vault, utilizza il comando `list-tags-for-vault`.

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

   Per ulteriori informazioni su questa azione di vault, consulta Elenca tag per vault.

3. Per rimuovere uno o più tag dal set di tag associato a una vault, utilizza il comando `remove-tags-from-vault`.

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

   Per ulteriori informazioni su questa operazione di vault, consulta Rimuovi tag dal vault.

Etichettatura degli archivi tramite l'API Amazon Glacier

Puoi aggiungere, elencare e rimuovere tag utilizzando l'API Amazon Glacier. Per alcuni esempi, consultare la seguente documentazione:

Add Tags To Vault (POST tags add)

Aggiunge o aggiorna i tag per il vault specificato.

List Tags For Vault (GET tags)

Elenca i tag per il vault specificato.

Remove Tags From Vault (POST tags remove)

Rimuove i tag dal vault specificato.

Sezioni correlate

- Etichettare le risorse di Amazon Glacier

Serratura Amazon Glacier Vault

I seguenti argomenti descrivono come bloccare un vault in Amazon Glacier e come utilizzare le policy di Vault Lock.

Argomenti

- Panoramica sul blocco di vault
- Bloccare un vault utilizzando l'API Amazon Glacier
- Bloccare un vault utilizzando il AWS Command Line Interface
- Bloccare un vault utilizzando la console Amazon Glacier

Panoramica sul blocco di vault

Amazon Glacier Vault Lock ti aiuta a implementare e applicare facilmente i controlli di conformità per singoli vault Amazon Glacier con una policy Vault Lock. Puoi specificare controlli in formato "write-once-read-many" (WORM) in una policy di blocco della vault e bloccare la policy per impedire modifiche future.

> **Important**
>
> Una volta bloccata, una policy di blocco della vault non può più essere modificata o eliminata.

Amazon Glacier applica i controlli impostati nella policy Vault Lock per aiutarti a raggiungere gli obiettivi di conformità. Ad esempio, puoi utilizzare le policy di blocco della vault per imporre la conservazione dei dati. Puoi implementare una varietà di controlli di conformità in una policy Vault

Lock utilizzando il linguaggio di policy (IAM). AWS Identity and Access Management Per ulteriori informazioni sulle policy di blocco della vault, consulta Policy di Vault Lock.

Una policy di blocco della vault è differente da una policy di accesso a vault. Entrambe regolamentano il controllo degli accessi alle vault, Tuttavia, una policy di blocco della vault può essere bloccata per evitare modifiche future, consentendo quindi un'applicazione rigorosa dei controlli di conformità. Puoi utilizzare la policy di blocco della vault per implementare controlli normativi e di conformità, che in genere richiedono un monitoraggio scrupoloso dell'accesso ai dati.

> **Important**
>
> Ti consigliamo di creare prima una vault, quindi di completare una policy di blocco della vault e poi caricare gli archivi nella vault in modo che la policy venga applicata ad essi.

Una policy di accesso a vault viene invece utilizzata per implementare controlli degli accessi non correlati alla conformità, temporanei e soggetti a frequenti modifiche. I due tipi di policy, di accesso e di blocco della vault, possono essere utilizzati congiuntamente. Ad esempio, puoi implementare regole di conservazione di dati basate sul tempo nella policy di blocco della vault (negare le eliminazioni) e concedere l'accesso in lettura a determinate terze parti o ai tuoi partner aziendali (consentire le letture) nella tua policy di accesso a volt.

Il blocco di un vault comporta due fasi:

1. Avvia il blocco associando alla vault una policy di blocco della vault che imposta lo stato In corso del blocco e restituisce un ID di blocco. Quando lo stato del blocco è In corso, hai 24 ore di tempo per convalidare la policy di blocco della vault prima della scadenza dell'ID di blocco. Per evitare che la vault esca dallo stato in corso, è necessario completare processo di blocco della vault entro queste 24 ore. In caso contrario, la policy di blocco della vault verrà eliminata.
2. Utilizza l'ID di blocco per completare il processo di blocco. Se la policy di blocco della vault non funziona come previsto, puoi arrestare il processo di blocco della vault e ricominciare dall'inizio. Per informazioni su come utilizzare l'API Amazon Glacier per bloccare un vault, consulta. Bloccare un vault utilizzando l'API Amazon Glacier

Bloccare un vault utilizzando l'API Amazon Glacier

Per bloccare il tuo vault con l'API Amazon Glacier, devi Initiate Vault Lock (POST lock-policy) prima chiamare con una policy Vault Lock che specifica i controlli che desideri distribuire. L'operazione

`Initiate Vault Lock` associa la policy alla vault, attiva lo stato In corso del blocco della vault e restituisce un ID di blocco univoco. Dopo l'attivazione dello stato In corso del blocco della vault, hai 24 ore di tempo per completare il blocco chiamando Complete Vault Lock (POST lockId) con l'ID di blocco restituito dalla chiamata di `Initiate Vault Lock`.

> **Important**
>
> - Ti consigliamo di creare prima una vault, quindi di completare una policy di blocco della vault e poi caricare gli archivi nella vault in modo che la policy venga applicata ad essi.
> - Una volta bloccata, la policy di blocco della vault non può essere modificata o eliminata.

Se non completi il processo di blocco della vault entro 24 ore dopo il passaggio allo stato In corso, lo stato della vault esce automaticamente dallo stato In corso e la policy di blocco della vault viene rimossa. Puoi chiamare `Initiate Vault Lock` nuovamente per implementare una nuova policy di blocco della vault e passare allo stato In corso.

Lo stato In corso fornisce l'opportunità di verificare la policy di blocco della vault prima di bloccarla. La policy di blocco della vault diventa effettiva quando lo stato è In corso come se la vault fosse stata bloccata, tranne per il fatto che puoi rimuovere la policy chiamando Abort Vault Lock (DELETE lock-policy). Per ottimizzare la policy, puoi ripetere la combinazione `Abort Vault Lock`/`Initiate Vault Lock` più volte come necessario allo scopo di convalidare le modifiche alla policy di blocco della vault.

Dopo la convalida della policy di blocco della vault, puoi chiamare Complete Vault Lock (POST lockId) con l'ID di blocco più recente per completare il processo di blocco della vault. Lo stato della vault diventa Bloccato e di conseguenza la policy di blocco della vault non è più modificabile e non può più essere rimossa chiamando `Abort Vault Lock`.

Sezioni correlate

- Policy di Vault Lock
- Abort Vault Lock (DELETE lock-policy)
- Complete Vault Lock (POST lockId)
- Get Vault Lock (GET lock-policy)
- Initiate Vault Lock (POST lock-policy)

Bloccare un vault utilizzando il AWS Command Line Interface

Puoi bloccare il tuo vault utilizzando. AWS Command Line Interface Questo installerà una policy di blocco della vault sulla vault specificata e restituirà l'ID di blocco. È necessario completare il processo di blocco della vault entro 24 ore, altrimenti la policy di blocco della vault viene rimossa dalla vault.

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface
2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

1. Usa `initiate-vault-lock` per installare una policy di blocco della vault e imposta lo stato di blocco della volt su `InProgress`.

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

2. La configurazione di blocco è un documento JSON come mostrato nell'esempio seguente. Prima di utilizzare questo comando, sostituite *VAULT_ARN* and *Principal* con i valori appropriati per il vostro caso d'uso.

   Per trovare l'ARN della vault che desideri bloccare, puoi usare il comando `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. Dopo aver avviato il blocco della vault, puoi vedere il file `lockId` restituito.

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

Per completare il blocco della vault, devi eseguire `complete-vault-lock` entro 24 ore, altrimenti la policy di blocco della vault viene rimossa dalla vault.

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

Sezioni correlate

- initiate-vault-lock in Riferimento ai comandi AWS CLI
- list-vaults nella pagina relativa al riferimento ai comandi della AWS CLI
- complete-vault-lock in Riferimento ai comandi AWS CLI
- Policy di Vault Lock
- Abort Vault Lock (DELETE lock-policy)
- Complete Vault Lock (POST lockId)
- Get Vault Lock (GET lock-policy)
- Initiate Vault Lock (POST lock-policy)

Bloccare un vault utilizzando la console Amazon Glacier

Amazon Glacier Vault Lock ti aiuta a implementare e applicare facilmente i controlli di conformità per singoli vault Amazon Glacier con una policy Vault Lock. Per ulteriori informazioni su Amazon Glacier Vault Lock, consulta Amazon Glacier Access Control with Vault Lock Policies.

> **Important**
>
> - Ti consigliamo di creare prima una vault, quindi di completare una policy di blocco della vault e poi caricare gli archivi nella vault in modo che la policy venga applicata ad essi.
> - Una volta bloccata, la policy di blocco della vault non può essere modificata o eliminata.

Per avviare una policy Vault Lock sul tuo vault utilizzando la console Amazon Glacier

Avvia il blocco associando alla vault una policy di blocco della vault che attiva lo stato In corso del blocco e restituisce un ID di blocco. Quando lo stato del blocco è In corso, hai 24 ore di tempo per convalidare la policy di blocco della vault prima della scadenza dell'ID di blocco.

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, seleziona una regione Regione AWS dal selettore della regione.
3. Nel pannello di navigazione a sinistra, seleziona Vault.
4. Nella pagina Vault, seleziona Crea vault.
5. Crea un nuovo vault.

   > **Important**
   >
   > Ti consigliamo di creare prima una vault, quindi di completare una policy di blocco della vault e poi caricare gli archivi nella vault in modo che la policy venga applicata ad essi.

6. Seleziona la tua nuova vault dall'elenco Vault.
7. Seleziona la scheda Policy di vault.
8. Nella sezione Policy di Vault Lock, seleziona Avvia policy di Vault Lock.
9. Nella pagina Avvia policy di Vault Lock, specifica i controlli di conservazione dei record nella policy di blocco della vault in formato testo nella casella di testo standard.

> **Note**
>
> Puoi specificare i controlli di conservazione dei record in una policy Vault Lock in formato testo e avviare il Vault Lock chiamando l'operazione dell'`Initiate Vault Lock`API o tramite l'interfaccia utente interattiva nella console Amazon Glacier. Per informazioni sulla formattazione della policy Vault Lock, consulta Esempi di policy di Amazon Glacier Vault Lock.

10. Scegli Save changes (Salva modifiche).
11. Nella finestra di dialogo Registra ID di Vault Lock, copia il tuo ID blocco e salvalo in un posto sicuro.

> **Important**
>
> Dopo l'avvio della policy di blocco della vault, hai 24 ore di tempo per convalidare la policy e completare il processo di blocco. Per completare il processo di blocco, è necessario fornire l'ID blocco. Se non viene fornito entro 24 ore, l'ID blocco scade e la policy in corso viene eliminata.

12. Una volta salvato l'ID blocco in una posizione sicura, seleziona Chiudi.
13. Verifica la tua policy di blocco della vault entro le prossime 24 ore. Se la policy funziona come previsto, seleziona Completa policy di Vault Lock.
14. Nella finestra di dialogo Completa Vault Lock, seleziona la casella di controllo per prendere atto del fatto che il completamento del processo della policy di blocco della vault è irreversibile.
15. Inserisci l'ID blocco fornito nella casella di testo.
16. Seleziona Completa Vault Lock.

Lavorare con gli archivi in Amazon Glacier

Un archivio è un qualsiasi oggetto, ad esempio una foto, un video o un documento, che conservi in una vault. È un'unità di storage di base in Amazon Glacier (Amazon Glacier). Ogni archivio dispone di un ID univoco e una descrizione facoltativa. Quando carichi un archivio, Amazon Glacier restituisce una risposta che include un ID di archivio. Questo ID di archivio è unico nella AWS regione in cui è archiviato l'archivio. Di seguito è riportato un esempio di ID archivio.

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

IDs Gli archivi hanno una lunghezza di 138 byte. Quando carichi un archivio, puoi fornire una descrizione facoltativa. Puoi recuperare un archivio utilizzando il relativo ID, ma non la descrizione.

> **Important**
>
> Amazon Glacier fornisce una console di gestione. che puoi utilizzare per creare ed eliminare vault. Tuttavia, tutte le altre interazioni con Amazon Glacier richiedono l'uso della AWS Command Line Interface (CLI) o la scrittura di codice. Ad esempio, per caricare dati, come foto, video e altri documenti, devi utilizzare AWS CLI o scrivere codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando Amazon SDKs. Per ulteriori informazioni sull'uso di Amazon Glacier con, consulta AWS CLI Reference for Amazon AWS CLI Glacier. Per installare, vai a. AWS CLIAWS Command Line Interface

Argomenti

- Operazioni di archiviazione in Amazon Glacier
- Conservazione di metadati di archivio sul lato client
- Caricamento di un archivio in Amazon Glacier
- Scaricare un archivio in Amazon Glacier
- Eliminazione di un archivio in Amazon Glacier

Operazioni di archiviazione in Amazon Glacier

Amazon Glacier supporta le seguenti operazioni di archiviazione di base: caricamento, download ed eliminazione. Il download di un archivio è un'operazione asincrona.

Caricamento di un archivio in Amazon Glacier

Puoi caricare un archivio in un'unica operazione oppure in parti. La chiamata API utilizzata per caricare un archivio in parti è denominata caricamento in più parti. Per ulteriori informazioni, consulta Caricamento di un archivio in Amazon Glacier.

> **Important**
>
> Amazon Glacier fornisce una console di gestione. che puoi utilizzare per creare ed eliminare vault. Tuttavia, tutte le altre interazioni con Amazon Glacier richiedono l'uso della AWS Command Line Interface (CLI) o la scrittura di codice. Ad esempio, per caricare dati, come foto, video e altri documenti, devi utilizzare AWS CLI o scrivere codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando Amazon SDKs. Per ulteriori informazioni sull'uso di Amazon Glacier con, consulta AWS CLI Reference for Amazon AWS CLI Glacier. Per installare, vai a. AWS CLIAWS Command Line Interface

Ricerca di un ID di archivio in Amazon Glacier

Per ottenere l'ID archivio, scarica l'inventario vault per la vault contenente l'archivio. Per ulteriori informazioni sul download di un inventario vault, consulta Scaricamento di un inventario Vault in Amazon Glacier.

Scaricare un archivio in Amazon Glacier

Il download di un archivio è un'operazione asincrona. Per scaricare un archivio specifico, devi innanzitutto avviare un processo. Dopo aver ricevuto la richiesta di lavoro, Amazon Glacier prepara l'archivio per il download. Una volta completato il processo, puoi scaricare i dati dell'archivio. A causa della natura asincrona del processo, puoi richiedere ad Amazon Glacier di inviare una notifica a un argomento di Amazon Simple Notification Service (Amazon SNS) al termine del processo. Puoi specificare un argomento di SNS per ogni singola richiesta di processo o configurare il vault affinché venga inviata una notifica quando si verificano determinati eventi. Per ulteriori informazioni sul caricamento di un archivio, consulta Scaricare un archivio in Amazon Glacier.

Eliminazione di un archivio in Amazon Glacier

Amazon Glacier fornisce una chiamata API che puoi utilizzare per eliminare un archivio alla volta. Per ulteriori informazioni, consulta Eliminazione di un archivio in Amazon Glacier.

Aggiornamento di un archivio in Amazon Glacier

Dopo il caricamento di un archivio, non puoi aggiornarne il contenuto o la descrizione. Il solo modo per farlo è eliminare l'archivio e caricarne un altro. Tieni presente che ogni volta che carichi un archivio, Amazon Glacier ti restituisce un ID di archivio univoco.

Conservazione di metadati di archivio sul lato client

Ad eccezione della descrizione opzionale dell'archivio, Amazon Glacier non supporta metadati aggiuntivi per gli archivi. Quando carichi un archivio, Amazon Glacier assegna un ID, una sequenza opaca di caratteri, dalla quale non puoi dedurre alcun significato sull'archivio. Puoi conservare i metadati relativi agli archivi sul lato client. I metadati possono includere il nome dell'archivio e altre informazioni utili sull'archivio.

> **Note**
>
> Se sei un cliente di Amazon Simple Storage Service (Amazon S3), sai che quando carichi un oggetto in un bucket puoi assegnargli una chiave di oggetto, ad esempio `MyDocument.txt` o `SomePhoto.jpg`. In Amazon Glacier, non puoi assegnare una chiave oggetto agli archivi che carichi.

Se gestisci i metadati di archiviazione sul lato client, tieni presente che Amazon Glacier mantiene un inventario del vault che include l'archivio IDs e tutte le descrizioni fornite durante il caricamento dell'archivio. Di tanto in tanto, puoi scaricare l'inventario vault per risolvere eventuali problemi nel database lato client che gestisci per i metadati degli archivi. Tuttavia, Amazon Glacier effettua l'inventario del vault all'incirca ogni giorno. Quando richiedi un inventario vault, Amazon Glacier restituisce l'ultimo inventario preparato, ovvero uno snapshot point-in-time.

Caricamento di un archivio in Amazon Glacier

Amazon Glacier (Amazon Glacier) fornisce una console di gestione che puoi utilizzare per creare ed eliminare vault. Tuttavia, non puoi caricare archivi su Amazon Glacier utilizzando la console di

gestione. Per caricare dati, come foto, video e altri documenti, devi utilizzare AWS CLI o scrivere codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando Amazon SDKs.

Per informazioni sull'uso di Amazon Glacier con, consulta AWS CLI Reference for Amazon AWS CLI Glacier. Per installare, vai a. AWS CLIAWS Command Line Interface I seguenti argomenti sul caricamento descrivono come caricare archivi su Amazon Glacier utilizzando Amazon SDK for Java, Amazon SDK for .NET e l'API REST.

Argomenti

- Opzioni per caricare un archivio su Amazon Glacier
- Caricamento di un archivio in un'unica operazione
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)

Opzioni per caricare un archivio su Amazon Glacier

A seconda della dimensione dei dati che stai caricando, Amazon Glacier offre le seguenti opzioni:

- Carica archivi in un'unica operazione: la dimensione degli archivi che possono essere caricati in un'unica operazione deve essere compresa tra 1 byte a 4 GB. Tuttavia, incoraggiamo i clienti di Amazon Glacier a utilizzare il caricamento in più parti per caricare archivi di dimensioni superiori a 100 MB. Per ulteriori informazioni, consulta Caricamento di un archivio in un'unica operazione.
- Upload archives in parts (Carica archivi in parti) – Mediante l'API di caricamento in più parti, puoi caricare archivi con dimensioni fino a 40.000 GB (10.000 * 4 GB).

  L'API di caricamento in più parti è concepita per migliorare l'esperienza di caricamento di archivi di grandi dimensioni. Puoi caricare gli archivi in parti. Queste parti possono essere caricate in modo indipendente, in qualsiasi ordine e in parallelo. Se il caricamento di una parte non riesce, è sufficiente caricare di nuovo quella parte e non l'intero archivio. Puoi utilizzare un caricamento in più parti per archivi con una dimensione compresa tra 1 byte e 40.000 GB. Per ulteriori informazioni, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

> **Important**
>
> L'inventario del vault Amazon Glacier viene aggiornato solo una volta al giorno. Un archivio caricato non sarà immediatamente visibile nel vault (nella console o nell'elenco di inventari vault scaricato) fino a quando l'inventario vault non viene aggiornato.

Utilizzo del servizio AWS Snowball Edge

AWS Snowball Edge accelera lo spostamento di grandi quantità di dati da e verso i dispositivi di proprietà di AWS Amazon, bypassando Internet. Per ulteriori informazioni, consulta la pagina dei dettagli di AWS Snowball Edge.

Per caricare dati esistenti su Amazon Glacier (Amazon Glacier), potresti prendere in considerazione l'utilizzo di uno dei tipi di dispositivo per importare AWS Snowball Edge i dati in Amazon S3 e quindi spostarli nella classe di storage Amazon Glacier per l'archiviazione utilizzando le regole del ciclo di vita. Quando trasferisci oggetti Amazon S3 alla classe di storage Amazon Glacier, Amazon S3 utilizza internamente Amazon Glacier per uno storage durevole a costi inferiori. Anche se gli oggetti vengono archiviati in Amazon Glacier, rimangono oggetti di Amazon S3 gestiti in Amazon S3 e non è possibile accedervi direttamente tramite Amazon Glacier.

Per ulteriori informazioni sulla configurazione del ciclo di vita di Amazon S3 e sulla transizione degli oggetti alla classe di storage Amazon Glacier, consulta Object Lifecycle Management and Transitioning Objects nella Amazon Simple Storage Service User Guide.

Caricamento di un archivio in un'unica operazione

Come descritto in Caricamento di un archivio in Amazon Glacier, puoi caricare archivi di dimensioni ridotte in un'unica operazione. Tuttavia, incoraggiamo i clienti di Amazon Glacier (Amazon Glacier) a utilizzare Multipart Upload per caricare archivi di dimensioni superiori a 100 MB.

Argomenti

- Caricamento di un archivio in un'unica operazione utilizzando il AWS Command Line Interface
- Caricamento di un archivio in un'unica operazione utilizzando AWS SDK per Java
- Caricamento di un archivio in un'unica operazione utilizzando Amazon AWS SDK per .NET Glacier
- Caricamento di un archivio in un'unica operazione mediante l'API REST

Caricamento di un archivio in un'unica operazione utilizzando il AWS Command Line Interface

Puoi caricare un archivio in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: caricare un archivio utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: caricare un archivio utilizzando il AWS CLI

Per caricare un archivio è necessario creare una vault. Per ulteriori informazioni sulla creazione di vault, consulta Creazione di un vault in Amazon Glacier.

1. Utilizza il comando `upload-archive` per aggiungere un archivio a una vault esistente. Nell'esempio seguente sostituisci `vault name` e `account ID`. Per il parametro `body`, specifica il percorso del file che desideri caricare.

```
aws glacier upload-archive --vault-name awsexamplevault --account-id 123456789012
 --body archive.zip
```

2. Output previsto:

```
{
    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
    "checksum": "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/123456789012/vaults/awsexamplevault/archives/
kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
}
```

   Al termine, il comando restituirà l'ID dell'archivio, il checksum e la posizione in Amazon Glacier. Per ulteriori informazioni sul comando upload-archive, consulta upload-archive nel Riferimento ai comandi AWS CLI .

Caricamento di un archivio in un'unica operazione utilizzando AWS SDK per Java

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for Java forniscono un metodo per caricare un archivio.

Argomenti

- Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java
- Caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello di AWS SDK per Java

Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `upload` che puoi utilizzare per caricare un archivio in una vault.

> **Note**
>
> Il metodo `upload` ti consente di caricare archivi di qualsiasi dimensione. A seconda della dimensione dell'archivio, determina se questo deve essere caricato in un'unica operazione oppure mediante un caricamento in più parti con l'API corrispondente.

Esempio: caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java

L'esempio di codice Java seguente carica un archivio in una vault (`examplevault`) nella regione Stati Uniti occidentali (Oregon) (`us-west-2`). Per un elenco delle AWS regioni e degli endpoint supportati, consulta. Accesso ad Amazon Glacier

Per step-by-step istruzioni su come eseguire questo esempio, consultaEsecuzione di esempi Java per Amazon Glacier con Eclipse. Devi aggiornare il codice con il nome della vault in cui intendi effettuare il caricamento e il nome del file da caricare come indicato.

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);
        }
    }
}
```

Caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello di AWS SDK per Java

L'interfaccia API di basso livello fornisce metodi per tutte le operazioni di archivio. Di seguito è riportata la procedura per caricare un archivio mediante il kit AWS SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui si desidera caricare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `UploadArchiveRequest`.

   Oltre ai dati da caricare, devi fornire un checksum (struttura hash SHA-256) del payload, il nome della vault, la lunghezza del contenuto dei dati e l'ID del tuo account.

   Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.
3. Eseguire il metodo `uploadArchive` fornendo l'oggetto della richiesta come parametro.

   In risposta, Amazon Glacier (Amazon Glacier) restituisce un ID di archivio dell'archivio appena caricato.

Il seguente frammento di codice Java illustra la procedura precedente.

```
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());
```

Esempio: caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello del AWS SDK per Java

Il seguente esempio di codice Java utilizza il AWS SDK per Java per caricare un archivio in un vault (). `examplevault` Per step-by-step istruzioni su come eseguire questo esempio, vedereEsecuzione di esempi Java per Amazon Glacier con Eclipse. Devi aggiornare il codice con il nome della vault in cui intendi effettuare il caricamento e il nome del file da caricare come indicato.

```
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);
        }
    }
}
```

Caricamento di un archivio in un'unica operazione utilizzando Amazon AWS SDK per .NET Glacier

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for .NET forniscono un metodo per caricare un archivio in un'unica operazione.

Argomenti

- Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET
- Caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello di AWS SDK per .NET

Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `Upload` che puoi utilizzare per caricare un archivio in una vault.

> **Note**
>
> Il metodo `Upload` ti consente di caricare file di qualsiasi dimensione. A seconda della dimensione del file, determina se questo deve essere caricato in un'unica operazione oppure mediante un caricamento in più parti con l'API corrispondente.

Esempio: caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

L'esempio di codice C# seguente carica un archivio in una vault (`examplevault`) nella regione Stati Uniti occidentali (Oregon).

Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con il nome del file che intendi caricare come indicato.

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();
    }
  }
}
```

Caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello di AWS SDK per .NET

L'interfaccia API di basso livello fornisce metodi per tutte le operazioni di archivio. Di seguito è riportata la procedura per caricare un archivio mediante il kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui si desidera caricare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `UploadArchiveRequest`.

   Oltre ai dati da caricare, devi fornire un checksum (struttura hash SHA-256) del payload, il nome della vault e l'ID del tuo account.

   Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per .NET con Amazon Glacier.
3. Eseguire il metodo `UploadArchive` fornendo l'oggetto della richiesta come parametro.

   In risposta, Amazon Glacier restituisce un ID di archivio dell'archivio appena caricato.

Esempio: caricamento di un archivio in un'unica operazione utilizzando l'API di basso livello del AWS SDK per .NET

L'esempio di codice C# seguente illustra la procedura precedente. L'esempio utilizza AWS SDK per .NET per caricare un archivio in un vault (). `examplevault`

> **Note**
>
> Per informazioni sull'utilizzo dell'API REST sottostante per caricare un archivio in un'unica richiesta, consulta Upload Archive (POST archive).

Per step-by-step istruzioni su come eseguire questo esempio, vediEsecuzione di esempi di codice. Devi aggiornare il codice con il nome del file che intendi caricare come indicato.

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;
      }
    }
  }
}
```

Caricamento di un archivio in un'unica operazione mediante l'API REST

Puoi utilizzare la chiamata API Upload Archive per caricare un archivio in un'unica operazione. Per ulteriori informazioni, consulta Upload Archive (POST archive).

Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)

Argomenti

- Processo di caricamento in più parti
- Dati rapidi
- Caricamento di archivi di grandi dimensioni mediante l' AWS CLI
- Caricamento in parti di archivi di grandi dimensioni mediante Amazon SDK per Java
- Caricamento di archivi di grandi dimensioni utilizzando AWS SDK per .NET

- Caricamento in parti di archivi di grandi dimensioni mediante l'API REST

Processo di caricamento in più parti

Come descritto inCaricamento di un archivio in Amazon Glacier, incoraggiamo i clienti di Amazon Glacier (Amazon Glacier) a utilizzare Multipart Upload per caricare archivi superiori a 100 mebibyte (MiB).

1. Avvio del caricamento in più parti

   Quando invii una richiesta per avviare un caricamento in più parti, Amazon Glacier restituisce un ID di caricamento multiparte, che è un identificatore univoco per il caricamento multiparte. Questo ID è necessario per qualsiasi operazione di caricamento in più parti successiva Questo ID non scade per almeno 24 ore dopo il completamento del lavoro da parte di Amazon Glacier.

   Nella richiesta di avviare un caricamento in più parti, specificare la dimensione della parte in numero di byte. Ogni parte che carichi, tranne l'ultima, deve essere di quella dimensione.

   > **Note**
   >
   > Non è necessario conoscere la dimensione globale dell'archivio quando si utilizza il caricamento in più parti. Ciò significa che è possibile utilizzare caricamenti in più parti nei casi in cui non si conosce la dimensione dell'archivio quando si avvia il caricamento dell'archivio. Devi solo decidere la dimensione della parte al momento in cui inizi il caricamento in più parti.

   Nella richiesta Initiate Multipart Upload, puoi anche fornire una descrizione facoltativa dell'archivio.

2. Caricamento delle parti

   Per ogni richiesta di caricamento, devi includere l'ID di caricamento in più parti ottenuto nella fase 1. Nella richiesta, devi anche specificare l'intervallo di contenuto in byte che identifica la posizione della parte nell'archivio finale. Amazon Glacier utilizza successivamente le informazioni sull'intervallo di contenuti per assemblare l'archivio nella sequenza corretta. Poiché fornisci l'intervallo di contenuto per ogni parte che carichi, questo determina la posizione della parte nell'assemblaggio finale dell'archivio e di conseguenza puoi caricare le parti in qualsiasi ordine.

Puoi anche caricare le parti in parallelo. Se carichi una nuova parte che utilizza lo stesso intervallo di contenuto di una parte caricata in precedenza, quest'ultima viene sovrascritta.

3. Completamento (o arresto) del caricamento in più parti

   Dopo il caricamento di tutte le parti dell'archivio, devi eseguire l'operazione di completamento. Anche in questo caso, devi specificare l'ID di caricamento nella tua richiesta. Amazon Glacier crea un archivio concatenando le parti in ordine crescente in base all'intervallo di contenuti fornito. La risposta di Amazon Glacier a una richiesta di caricamento multiparte completo include un ID di archivio per l'archivio appena creato. Se hai fornito una descrizione opzionale dell'archivio nella richiesta Initiate Multipart Upload, Amazon Glacier la associa all'archivio assemblato. Una volta completato un caricamento in più parti, non puoi fare riferimento all'ID di caricamento in più parti. Ciò significa che non puoi accedere alle parti associate a quell'ID.

   Se arresti un caricamento in più parti, non puoi caricare altre parti utilizzando quell'ID. Tutto lo spazio di archiviazione utilizzato dalle parti associate al caricamento in più parti arrestato viene liberato. Se eventuali caricamenti di parti erano in corso, possono essere eseguiti correttamente o meno anche dopo l'arresto.

Operazioni di caricamento in più parti supplementari

Amazon Glacier (Amazon Glacier) fornisce le seguenti chiamate API di caricamento multiparte aggiuntive.

- List Parts: questa operazione ti consente di elencare le parti di un determinato caricamento in più parti e restituisce informazioni sulle parti che hai caricato nell'ambito di tale caricamento. Per ogni richiesta di parti di un elenco, Amazon Glacier restituisce informazioni per un massimo di 1.000 parti. Se vi sono altre parti da elencare per il caricamento in più parti, viene eseguita la paginazione del risultato e la risposta include un contrassegno che segnala il punto da cui continuare l'elenco. Per recuperare le parti successive, devi inviare ulteriori richieste. Tenere presente che l'elenco di parti restituito non include le parti per le quali non è stato completato il caricamento.
- List Multipart Uploads: questa operazione consente di ottenere un elenco dei caricamenti in più parti in corso. Un caricamento in più parti in corso è un caricamento avviato, ma non ancora completato o annullato. Per ogni richiesta List Multipart Uploads, Amazon Glacier restituisce fino a 1.000 caricamenti in più parti. Se vi sono altri caricamenti in più parti da elencare, viene eseguita la paginazione dei risultati e la risposta include un contrassegno che segnala il punto da cui continuare l'elenco. Per recuperare i caricamenti in più parti rimanenti, devi inviare ulteriori richieste.

Dati rapidi

La tabella riportata di seguito fornisce le specifiche di base di un caricamento in più parti.

Elemento	Specifica
Dimensione massima dell'archivio	10.000 x 4 gibibyte (GiB)
Numero massimo di parti per caricamento	10.000
Dimensione parte	Da 1 MiB a 4 GiB, l'ultima parte può essere < 1 MiB. Devi specificare la dimensione in byte. La dimensione della parte deve essere un mebibyte (1024 kibibyte [KiB]) moltiplicato per una potenza di 2. Ad esempio, `1048576` (1 MiB), `2097152` (2 MiB), `4194304` (4 MiB), `8388608` (8 MiB).
Numero massimo di parti restituite per una richiesta di elenco delle parti	1.000
Numero massimo di caricamenti in più parti restituiti per una richiesta di elenco dei caricamenti in più parti	1.000

Caricamento di archivi di grandi dimensioni mediante l' AWS CLI

Puoi caricare un archivio in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI Per migliorare l'esperienza di caricamento per archivi di grandi dimensioni, Amazon Glacier fornisce diverse operazioni API per supportare caricamenti multiparte. Utilizzando queste operazioni API, puoi caricare gli archivi in parti. Queste parti possono essere caricate in modo indipendente, in qualsiasi ordine e in parallelo. Se il caricamento di una parte non ha esito positivo, è sufficiente caricare di nuovo quella parte e non l'intero archivio. Puoi utilizzare i caricamenti in più parti per archivi con una dimensione compresa tra 1 byte e 40.000 gibibyte (GiB).

Per ulteriori informazioni sui caricamenti multiparte di Amazon Glacier, consulta. Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- (Prerequisito) Installa Python
- (Prerequisito) Creare un Amazon Glacier Vault
- Esempio: caricamento di archivi di grandi dimensioni in parti utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface
2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

(Prerequisito) Installa Python

Per completare un caricamento in più parti, devi calcolare l'hash SHA256 dell'albero dell'archivio che stai caricando. Questa operazione è diversa dal calcolo dell'hash SHA256 ad albero del file che desideri caricare. Per calcolare l'hash dell' SHA256 albero dell'archivio che stai caricando, puoi usare

C# (Javacon.NET) o. Python In questo esempio verrà utilizzato Python. Per istruzioni sull'uso di Java o C#, consulta Calcolo di checksum.

Per informazioni sull'installazione di Python, consulta la pagina Installazione o aggiornamento di Python nella Guida per lo sviluppatore di Boto3.

(Prerequisito) Creare un Amazon Glacier Vault

Per utilizzare l'esempio seguente, devi avere almeno un vault Amazon Glacier creato. Per ulteriori informazioni sulla creazione di vault, consulta Creazione di un vault in Amazon Glacier.

Esempio: caricamento di archivi di grandi dimensioni in parti utilizzando il AWS CLI

In questo esempio, creerai un file e utilizzerai operazioni API di caricamento in più parti per caricare questo file, in parti su Amazon Glacier.

> **Important**
>
> Prima di intraprendere questa procedura, accertati di avere eseguito tutti i passaggi necessari. Per caricare un archivio, è necessario aver creato un archivio, AWS CLI configurarlo ed essere pronti a utilizzare Java C# o Python a calcolare l'hash di un SHA256 albero.

La procedura seguente utilizza i comandi `initiate-multipart-upload` e`upload-multipart-part`. `complete-multipart-upload` AWS CLI

Per ulteriori informazioni dettagliate su ognuno di questi comandi, consulta initiate-multipart-upload, upload-multipart-part e complete-multipart-upload nella Guida di riferimento ai comandi AWS CLI .

1. Usa il comando initiate-multipart-upload per creare una risorsa di caricamento in più parti. Nella richiesta, specifica la dimensione della parte in numero di byte. Ogni parte che carichi, tranne l'ultima, deve essere di quella dimensione. Non è necessario conoscere la dimensione globale dell'archivio quando si avvia un caricamento. Tuttavia, avrai bisogno della dimensione totale, in byte, di ogni parte per completare il caricamento nella fase finale.

   Nel comando seguente, sostituisci i valori per i parametri `--vault-name` e `--account-ID` con le tue informazioni. Questo comando specifica che caricherai un archivio con una dimensione parziale di 1 mebibyte (MiB) (1024 x 1024 byte) per file. Se necessario, sostituisci il valore di questo parametro `--part-size`.

```
aws glacier initiate-multipart-upload --vault-name awsexamplevault --part-
size 1048576 --account-id 123456789012
```

Output previsto:

```
{
"location": "/123456789012/vaults/awsexamplevault/multipart-uploads/uploadId",
"uploadId": "uploadId"
}
```

Al termine, il comando restituirà l'ID e la posizione di caricamento della risorsa di caricamento multiparte in Amazon Glacier. Sarà necessario usare questo ID di caricamento nei passaggi successivi.

2. Per questo esempio, è possibile utilizzare i seguenti comandi per creare un file da 4,4 MiB, dividerlo in blocchi da 1 MiB e caricare ogni blocco. Per caricare i propri file, è possibile seguire una procedura simile suddividendo i dati in blocchi e caricando ogni parte.

   Linux o macOS:

   Il comando seguente crea un file da 4,4 MiB, denominato `file_to_upload`, su Linux o macOS.

   ```
   mkfile -n 9000b file_to_upload
   ```

   Windows

   Il comando seguente crea un file da 4,4 MiB, denominato `file_to_upload`, su Windows.

   ```
   fsutil file createnew file_to_upload 4608000
   ```

3. Successivamente, dividerai questo file in blocchi da 1 MiB.

   ```
   split -b 1048576 file_to_upload chunk
   ```

   Ora hai le seguenti cinque parti. Le prime quattro sono da 1 MiB e l'ultima è di circa 400 kibibyte (KiB).

   ```
   chunkaa
   ```

```
chunkab
chunkac
chunkad
chunkae
```

4. Usa il comandi upload-multipart-part per caricare una parte di un archivio. Puoi caricare gli archivi in parti in un qualsiasi ordine. Puoi anche caricare le parti in parallelo. In un caricamento in più parti è consentito caricare fino a 10.000 parti.

   Nel comando seguente, sostituisci i valori per i parametri `--vault-name`, `--account-ID` e `--upload-id`. L'ID di caricamento deve corrispondere all'ID fornito come output del comando `initiate-multipart-upload`. Il parametro `--range` specifica che caricherai una parte con una dimensione di 1 MiB (1024 x 1024 byte). Questa dimensione deve corrispondere a quella specificata nel comando `initiate-multipart-upload`. Regola il valore della dimensione, se necessario. Il parametro `--body` specifica il nome della parte che stai caricando.

   ```
   aws glacier upload-multipart-part --body chunkaa --range='bytes 0-1048575/*' --
   vault-name awsexamplevault --account-id 123456789012 --upload-id upload_ID
   ```

   In caso di successo, il comando produrrà un output contenente il checksum per la parte caricata.

5. Esegui nuovamente il comando `upload-multipart-part` per caricare le parti rimanenti del caricamento in più parti. Aggiorna i valori dei parametri `--range` e `--body` per ogni comando in modo che corrispondano alla parte che stai caricando.

   ```
   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**
>
> Il valore del parametro `--range` del comando finale è inferiore perché la parte finale del nostro caricamento è inferiore a 1 MiB. In caso di successo, ogni comando produrrà un output contenente il checksum per ogni parte caricata.

6. Successivamente, assemblerai l'archivio e completerai il caricamento. È necessario includere la dimensione totale e l'hash SHA256 dell'albero dell'archivio.

   Per calcolare l'hash ad SHA256 albero dell'archivio, puoi usare Java C# o. Python In questo esempio verrà utilizzato Python. Per istruzioni sull'uso di Java o C#, consulta Calcolo di checksum.

   Crea il file `checksum.py` di Python e inserisci il codice seguente. Se necessario, sostituisci il nome del file originale.

   ```
   from botocore.utils import calculate_tree_hash

   checksum = calculate_tree_hash(open('file_to_upload', 'rb'))
   print(checksum)
   ```

7. Esegui `checksum.py` per calcolare l'hash dell' SHA256 albero. Il seguente hash potrebbe non corrispondere al tuo output.

   ```
   $ python3 checksum.py
   $ 3d760edb291bfc9d90d35809243de092aea4c47b308290ad12d084f69988ae0c
   ```

8. Usa il comando complete-multipart-upload per completare il caricamento dell'archivio. Sostituisci i valori per i parametri `--vault-name`, `--account-ID`, `--upload-ID` e `--checksum`. Il valore del parametro `--archive` specifica la dimensione totale, in byte, dell'archivio. Questo valore deve essere la somma di tutte le dimensioni delle singole parti caricate. Se necessario, sostituisci questo valore.

   ```
   aws glacier complete-multipart-upload --archive-size 4608000 --vault-
   name awsexamplevault --account-id 123456789012 --upload-id upload_ID --
   checksum checksum
   ```

   Al termine, il comando restituirà l'ID, il checksum e la posizione dell'archivio in Amazon Glacier.

Caricamento in parti di archivi di grandi dimensioni mediante Amazon SDK per Java

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for Java forniscono un metodo per caricare un archivio di grandi dimensioni (vedi). Caricamento di un archivio in Amazon Glacier

- L'API di alto livello fornisce un metodo che puoi utilizzare per caricare archivi di qualsiasi dimensione. A seconda del file che stai caricando, il metodo carica un archivio in un'unica operazione o utilizza il supporto per il caricamento multiparte in Amazon Glacier (Amazon Glacier) per caricare l'archivio in parti.
- L'API di basso livello esegue la mappatura all'implementazione REST sottostante. Di conseguenza, fornisce un metodo per caricare archivi di dimensioni ridotte in un'unica operazione e un gruppo di metodi che supportano il caricamento in più parti per gli archivi di grandi dimensioni. Questa sezione illustra il caricamento in parti di archivi di grandi dimensioni mediante l'API di basso livello.

Per ulteriori informazioni sul livello alto e su quello di basso livello, consulta. APIs Utilizzo di AWS SDK per Java con Amazon Glacier

Argomenti

- Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di alto livello di AWS SDK per Java
- Carica archivi di grandi dimensioni in parti utilizzando l'API di basso livello di AWS SDK per Java

Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di alto livello di AWS SDK per Java

Per caricare archivi di piccole o grandi dimensioni, utilizzi gli stessi metodi dell'API di alto livello. In base alle dimensioni dell'archivio, i metodi API di alto livello decidono se caricare l'archivio in un'unica operazione o utilizzare l'API di caricamento multiparte fornita da Amazon Glacier. Per ulteriori informazioni, consulta Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java.

Carica archivi di grandi dimensioni in parti utilizzando l'API di basso livello di AWS SDK per Java

Per un controllo granulare del caricamento, puoi utilizzare l'API di basso livello, che consente di configurare la richiesta ed elaborare la risposta. Di seguito è riportata la procedura per caricare archivi di grandi dimensioni in parti utilizzando il kit AWS SDK per Java.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui si desidera salvare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Avvia il caricamento in più parti chiamando il metodo `initiateMultipartUpload`.

   Devi fornire il nome della vault in cui desideri caricare l'archivio, la dimensione di parte da utilizzare per caricare le parti dell'archivio ed eventualmente una descrizione. Queste informazioni devono essere specificate quando si crea un'istanza della classe `InitiateMultipartUploadRequest`. In risposta, Amazon Glacier restituisce un ID di caricamento.
3. Carica le parti chiamando il metodo `uploadMultipartPart`.

   Per ogni parte caricata, devi fornire il nome della vault, l'intervallo di byte nell'archivio assemblato finale che sarà caricato nella parte, il checksum dei dati della parte e l'ID di caricamento.
4. Completa il caricamento in più parti chiamando il metodo `completeMultipartUpload`.

   Devi fornire l'ID di caricamento, il checksum dell'intero archivio, la dimensione dell'archivio (la dimensione combinata di tutte le parti caricate) e il nome della vault. Amazon Glacier crea l'archivio a partire dalle parti caricate e restituisce un ID di archivio.

Esempio: caricamento di un archivio di grandi dimensioni in un componente utilizzando AWS SDK per Java

Il seguente esempio di codice Java utilizza AWS SDK per Java per caricare un archivio in un vault ()`examplevault`. Per step-by-step istruzioni su come eseguire questo esempio, vedereEsecuzione di esempi Java per Amazon Glacier con Eclipse. Devi aggiornare il codice con il nome del file che intendi caricare come indicato.

> **Note**
>
> Questo esempio è valido per dimensioni di parte comprese tra 1 MB e 1 GB. Nota tuttavia che Amazon Glacier supporta dimensioni di parte fino a 4 GB.

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();
    }
}
```

Caricamento di archivi di grandi dimensioni utilizzando AWS SDK per .NET

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for .NET forniscono un metodo per caricare archivi di grandi dimensioni in parti (vedi). Caricamento di un archivio in Amazon Glacier

- L'API di alto livello fornisce un metodo che puoi utilizzare per caricare archivi di qualsiasi dimensione. A seconda del file che stai caricando, il metodo carica l'archivio in un'unica operazione o utilizza il supporto di caricamento multiparte in Amazon Glacier (Amazon Glacier) per caricare l'archivio in parti.
- L'API di basso livello esegue la mappatura all'implementazione REST sottostante. Di conseguenza, fornisce un metodo per caricare archivi di dimensioni ridotte in un'unica operazione e un gruppo di metodi che supportano il caricamento in più parti per gli archivi di grandi dimensioni. Questa sezione illustra il caricamento in parti di archivi di grandi dimensioni mediante l'API di basso livello.

Per ulteriori informazioni sul livello alto e su quello di basso livello, consulta. APIs Utilizzo di AWS SDK per .NET con Amazon Glacier

Argomenti

- Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di alto livello di AWS SDK per .NET
- Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di basso livello di AWS SDK per .NET

Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di alto livello di AWS SDK per .NET

Per caricare archivi di piccole o grandi dimensioni, utilizzi gli stessi metodi dell'API di alto livello. In base alle dimensioni dell'archivio, i metodi API di alto livello decidono se caricare l'archivio in un'unica operazione o utilizzare l'API di caricamento multiparte fornita da Amazon Glacier. Per ulteriori informazioni, consulta Caricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET.

Caricamento di archivi di grandi dimensioni in parti utilizzando l'API di basso livello di AWS SDK per .NET

Per un controllo granulare del caricamento, puoi utilizzare l'API di basso livello, che consente di configurare la richiesta ed elaborare la risposta. Di seguito è riportata la procedura per caricare archivi di grandi dimensioni in parti utilizzando il kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

È necessario specificare una AWS regione in cui si desidera salvare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Avvia il caricamento in più parti chiamando il metodo `InitiateMultipartUpload`.

   Devi fornire il nome della vault in cui desideri caricare l'archivio, la dimensione di parte da utilizzare per caricare le parti dell'archivio ed eventualmente una descrizione. Queste informazioni devono essere specificate quando si crea un'istanza della classe `InitiateMultipartUploadRequest`. In risposta, Amazon Glacier restituisce un ID di caricamento.

3. Carica le parti chiamando il metodo `UploadMultipartPart`.

   Per ogni parte caricata, devi fornire il nome della vault, l'intervallo di byte nell'archivio assemblato finale che sarà caricato nella parte, il checksum dei dati della parte e l'ID di caricamento.

4. Completa il caricamento in più parti chiamando il metodo `CompleteMultipartUpload`.

   Devi fornire l'ID di caricamento, il checksum dell'intero archivio, la dimensione dell'archivio (la dimensione combinata di tutte le parti caricate) e il nome della vault. Amazon Glacier crea l'archivio a partire dalle parti caricate e restituisce un ID di archivio.

Esempio: caricamento in parti di un archivio di grandi dimensioni mediante Amazon SDK per .NET

Il seguente esempio di codice C# utilizza il AWS SDK per .NET per caricare un archivio in un vault (). `examplevault` Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con il nome del file che intendi caricare come indicato.

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;
    }
  }
}
```

Caricamento in parti di archivi di grandi dimensioni mediante l'API REST

Come descritto in Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti), il caricamento in più parti fa riferimento a un set di operazioni che consentono di caricare un archivio in parti e di eseguire operazioni correlate. Per ulteriori informazioni su queste operazioni, consulta i seguenti argomenti di riferimento dell'API:

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- Abort Multipart Upload (DELETE uploadID)
- List Parts (GET uploadID)
- List Multipart Uploads (GET multipart-uploads)

Scaricare un archivio in Amazon Glacier

Amazon Glacier fornisce una console di gestione che puoi utilizzare per creare ed eliminare vault. Tuttavia, non puoi scaricare archivi da Amazon Glacier utilizzando la console di gestione. Per scaricare dati, come foto, video e altri documenti, devi usare AWS Command Line Interface (AWS CLI) o scrivere codice per effettuare richieste, utilizzando direttamente l'API REST o utilizzando il. AWS SDKs

Per informazioni sull'uso di Amazon Glacier con, consulta AWS CLIAWS CLI la pagina di riferimento per Amazon Glacier. Per installare, consulta. AWS CLIAWS Command Line Interface I seguenti

argomenti descrivono come scaricare archivi su Amazon Glacier utilizzando AWS SDK per Java l'API REST di Amazon Glacier. AWS SDK per .NET

Argomenti

- Recupero degli archivi Amazon Glacier
- Scaricamento di un archivio in Amazon Glacier utilizzando il AWS SDK per Java
- Scaricamento di un archivio in Amazon Glacier utilizzando il AWS SDK per .NET
- Scaricare un archivio di grandi dimensioni utilizzando l'elaborazione parallela con Python
- Download di un archivio mediante REST API
- Scaricare un archivio in Amazon Glacier utilizzando il AWS CLI

Recupero degli archivi Amazon Glacier

Il recupero di un archivio da Amazon Glacier è un'operazione asincrona in cui si avvia prima un processo e quindi si scarica l'output al termine del processo. Per avviare un processo di recupero dell'archivio, si utilizza l'operazione API Initiate Job (POST jobs) REST o l'equivalente in, o il. AWS CLI AWS SDKs

Argomenti

- Opzioni di recupero dall'archivio
- Recupero di intervalli di archivio

Il recupero di un archivio da Amazon Glacier è un processo in due fasi. Di seguito è riportata una panoramica di questo processo.

Recupero di un archivio

1. Avvia un processo di recupero di archivio.

    a. Ottieni l'ID dell'archivio che intendi recuperare. Puoi ottenere questo ID da un inventario del vault. È possibile ottenere l'ID dell'archivio con l'API REST AWS CLI, oppure AWS SDKs. Per ulteriori informazioni, consulta Scaricamento di un inventario Vault in Amazon Glacier.

    b. Avvia un processo che richieda ad Amazon Glacier di preparare un intero archivio o una parte di esso per il download successivo utilizzando l'operazione. Initiate Job (POST jobs)

Quando avvii un processo, Amazon Glacier restituisce un ID del lavoro nella risposta e lo esegue in modo asincrono. Puoi eseguire il download dell'output del processo solo dopo il completamento del processo come descritto nel passaggio 2.

> **Important**
>
> Solo nei recuperi di tipo Standard, l'utilizzo di una policy di recupero dati può causare un errore nella richiesta `Initiate Job` e generare l'eccezione `PolicyEnforcedException`. Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon Glacier. Per ulteriori informazioni sull'eccezione `PolicyEnforcedException`, consulta Risposte agli errori.

Se necessario, puoi ripristinare ampi segmenti di dati archiviati in Amazon Glacier. Per ulteriori informazioni sul ripristino dei dati dalle classi di storage Amazon Glacier, consulta Storage Classes for Archiving Objects nella Amazon Simple Storage Service User Guide.

2. Al completamento del processo, esegui il download dei byte mediante l'operazione Get Job Output (GET output).

   Puoi scaricare tutti i byte o specificare un intervallo di byte per scaricare solo una parte dell'output del processo. Per gli output più voluminosi, il download in blocchi è utile in caso di errore durante l'operazione, ad esempio un errore di rete. Se scarichi l'output del processo in un'unica richiesta e si verifica un errore di rete, devi ripetere il download dell'output dall'inizio. Se invece esegui un download in blocchi, in caso di errore, devi solo ripetere il download della parte in questione e non dell'intero output.

Amazon Glacier deve completare un processo prima di poterne ottenere l'output. Dopo il completamento, un processo rimane valido per almeno 24 ore, il che significa che puoi eseguire il download dell'output entro quel periodo di tempo. Il ripristino può scadere in qualsiasi momento dopo 24 ore dal completamento del processo. Per determinare se il processo è stato completato, controllane lo stato in uno dei seguenti modi:

- Attendi una notifica di completamento del lavoro: puoi specificare un argomento di Amazon Simple Notification Service (Amazon SNS) su cui Amazon Glacier può pubblicare una notifica dopo il completamento del lavoro. Amazon Glacier invia una notifica solo dopo aver completato il processo.

Puoi specificare un argomento di Amazon SNS all'avvio del processo. Oltre a specificare un argomento di Amazon SNS nella richiesta di lavoro, se nel vault sono impostate notifiche per gli eventi di recupero degli archivi, Amazon Glacier pubblica anche una notifica su quell'argomento SNS. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier.

- Richiedi informazioni sul lavoro in modo esplicito: puoi anche utilizzare l'API Amazon Glacier `Describe Job` Describe Job (GET JobID) operation () per ottenere periodicamente informazioni sulla mansione. Ti consigliamo tuttavia di utilizzare le notifiche di Amazon SNS.

> **Note**
>
> Le informazioni che ottieni utilizzando le notifiche di Amazon SNS sono identiche a quelle che ottieni chiamando l'operazione API `Describe Job`.

Opzioni di recupero dall'archivio

Per recuperare un archivio in funzione delle tue esigenze quanto a costi e tempo di accesso, puoi specificare una delle opzioni descritte di seguito all'avvio di un processo. Per informazioni sui prezzi di recupero, consulta la pagina dei prezzi di Amazon Glacier.

- Expedited: i recuperi rapidi consentono di accedere rapidamente ai dati archiviati nella classe di archiviazione S3 Glacier Flexible Retrieval o nel livello S3 Intelligent-Tiering Archive Access quando sono necessarie richieste occasionali e urgenti di ripristino degli archivi. Per tutti gli archivi, ad eccezione di quelli più grandi (oltre 250 MB), i dati accessibili tramite i recuperi accelerati sono in genere disponibili entro 1-5 minuti. La capacità assegnata assicura che la capacità di recupero per ripristini di tipo Expedited sia disponibile quando richiesta. Per ulteriori informazioni, consulta capacità con provisioning.
- Standard: i recuperi di tipo Standard ti consentono di accedere a qualsiasi archivio entro alcune ore. I recuperi Standard generalmente vengono completati entro 3-5 ore. Questa è l'opzione predefinita per le richieste di recupero che non specificano l'opzione di recupero.
- In blocco: i recuperi in blocco sono l'opzione di recupero Amazon Glacier più economica, che puoi utilizzare per recuperare grandi quantità, anche petabyte, di dati in modo economico in un giorno. I recuperi Bulk generalmente vengono completati entro 5-12 ore.

La tabella seguente riepiloga le opzioni di recupero archivi. Per informazioni sui prezzi, consultare Prezzi di Amazon Glacier.

Per creare o `Bulk` recuperare `ExpeditedStandard`, imposta l'elemento di richiesta nella `Tier` richiesta di operazione API RestoreObjectREST sull'opzione che desideri, o sull'equivalente in () o. AWS Command Line Interface AWS CLI AWS SDKs Se hai acquistato capacità con provisioning, tutti i recuperi Expedited vengono serviti automaticamente mediante la capacità con provisioning.

capacità con provisioning

La capacità assegnata assicura che la capacità di recupero per effettuare recuperi di tipo Expedited sia disponibile al bisogno. Ogni unità di capacità consente di eseguire almeno tre recuperi Express ogni 5 minuti e fornisce fino a 150 megabyte al secondo () di velocità di recupero. MBps

Se il carico di lavoro richiede un accesso altamente affidabile e predicibile a un sottoinsieme di dati nell'arco di pochi minuti, è consigliabile acquistare capacità di recupero con provisioning. In assenza di capacità con provisioning, i recuperi Expedited vengono generalmente accettati, salvo in rare situazioni di richiesta insolitamente elevata. Tuttavia, se è necessario l'accesso ai recuperi Expedited in qualsiasi circostanza, è necessario acquistare capacità di recupero assegnata.

Acquisto di capacità con provisioning

Puoi acquistare unità di capacità assegnate utilizzando la console Amazon Glacier, il funzionamento AWS SDKs dell'API RESTPurchase Provisioned Capacity (POST provisioned-capacity), o il. AWS CLI Per informazioni sui prezzi della capacità assegnata, consulta la pagina dei prezzi di Amazon Glacier.

Un'unità di capacità con provisioning dura per un mese a partire dalla data e dall'ora dell'acquisto.

Se la data di inizio è il trentunesimo giorno di un mese, la data di scadenza sarà l'ultimo giorno del mese successivo. Ad esempio, se la data di inizio è il 31 agosto, la data di scadenza sarà il 30 settembre. Se la data di inizio è il 31 gennaio, la data di scadenza sarà il 28 febbraio.

Per acquistare capacità assegnata utilizzando la console Amazon Glacier

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. Nel riquadro di navigazione sinistro, seleziona Impostazioni di recupero dati.
3. In Unità di capacità fornite (PCUs), scegli Acquista PCU. Viene visualizzata la finestra di dialogo Acquista PCU.

4. Se desideri acquistare capacità assegnata, inserisci **`confirm`** nella casella Per confermare l'acquisto.
5. Scegli Acquista PCU.

Recupero di intervalli di archivio

Quando recuperi un archivio da Amazon Glacier, puoi facoltativamente specificare un intervallo o una parte dell'archivio da recuperare. L'impostazione di default prevede il recupero dell'intero archivio. Specificare un intervallo di byte può rivelarsi utile se intendi:

- Gestisci i download dei dati: Amazon Glacier consente di scaricare i dati recuperati per 24 ore dopo il completamento della richiesta di recupero. Di conseguenza, puoi decidere di recuperare solo alcune parti dell'archivio per gestire la pianificazione dei download entro la finestra di download specificata.
- Recupera una determinata parte di un archivio di grandi dimensioni: supponiamo, ad esempio, che in precedenza tu abbia aggregato molti file e li abbia caricati come un unico archivio, ma ora desideri recuperarne solo alcuni. In questo caso, puoi specificare l'intervallo dell'archivio che contiene i file che desideri utilizzando una richiesta di recupero. oppure avviare più richieste di recupero, ognuna con un intervallo di uno o più file.

Quando avvii un processo di recupero utilizzando recuperi di intervalli, devi fornire un intervallo che è allineato in termini di megabyte. In altre parole, l'intervallo di byte può iniziare da zero (inizio dell'archivio) o da qualsiasi incremento di 1 MB successivo (1 MB, 2 MB, 3 MB e così via).

La fine di tale intervallo può essere la fine dell'archivio oppure qualsiasi incremento di 1 MB più grande dell'inizio dell'intervallo. Inoltre, se desideri ottenere valori di checksum quando esegui il download dei dati (dopo il completamento del processo di recupero), l'intervallo che richiedi all'avvio del processo deve essere allineato anche alla struttura hash. Puoi utilizzare i checksum per determinare se i dati sono stati danneggiati durante la trasmissione. Per ulteriori informazioni sull'allineamento in termini di megabyte e alla struttura hash, consulta Ottenimento di checksum durante il download di dati.

Scaricamento di un archivio in Amazon Glacier utilizzando il AWS SDK per Java

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for Java forniscono un metodo per scaricare un archivio.

Argomenti

- Scaricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java
- Scaricamento di un archivio utilizzando l'API di basso livello di AWS SDK per Java

Scaricamento di un archivio utilizzando l'API di alto livello di AWS SDK per Java

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `download` che puoi utilizzare per scaricare un archivio.

> **Important**
>
> La classe `ArchiveTransferManager` crea un argomento Amazon Simple Notification Service (Amazon SNS) e una coda Amazon Simple Queue Service (Amazon SQS) sottoscritta a quell'argomento. Avvia quindi il processo di recupero di archivio ed esegue il polling della coda affinché l'archivio sia disponibile. Quando l'archivio è disponibile, il download ha inizio. Per informazioni sui tempi di recupero, consulta Opzioni di recupero dall'archivio.

Esempio: download di un archivio utilizzando l'API di alto livello di AWS SDK per Java

L'esempio di codice Java seguente esegue il download di un archivio da una vault (`examplevault`) nella regione Stati Uniti occidentali (Oregon) (`us-west-2`).

Per step-by-step istruzioni sull'esecuzione di questo esempio, consultaEsecuzione di esempi Java per Amazon Glacier con Eclipse. Devi aggiornare il codice con un ID archivio esistente e il percorso di file locale in cui intendi salvare l'archivio scaricato come indicato.

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);
        }
    }
}
```

Scaricamento di un archivio utilizzando l'API di basso livello di AWS SDK per Java

Di seguito è riportata la procedura per recuperare un inventario vault mediante l'API di basso livello del kit AWS SDK per Java .

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

È necessario specificare una AWS regione da cui si desidera scaricare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Avvia un processo `archive-retrieval` eseguendo il metodo `initiateJob`.

   Fornisci informazioni sul lavoro, come l'ID di archivio dell'archivio che desideri scaricare e l'argomento opzionale di Amazon SNS in cui desideri che Amazon Glacier (Amazon Glacier) pubblichi un messaggio di completamento del lavoro, creando un'istanza della classe. `InitiateJobRequest` Amazon Glacier restituisce un Job ID in risposta. La risposta è disponibile in un'istanza della 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();
```

   Facoltativamente, puoi specificare un intervallo di byte per richiedere ad Amazon Glacier di preparare solo una parte dell'archivio. Ad esempio, puoi aggiornare la richiesta precedente aggiungendo la seguente istruzione affinché Amazon Glacier prepari unicamente la parte dell'archivio compresa tra 1 MB e 2 MB.

```
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. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica sul vault che identifica un argomento di Amazon Simple Notification Service (Amazon SNS) o hai specificato un argomento Amazon SNS quando hai avviato un processo, Amazon Glacier invia un messaggio a quell'argomento dopo il completamento del processo.

   Puoi anche interrogare Amazon Glacier `describeJob` chiamando il metodo per determinare lo stato di completamento del lavoro. La soluzione consigliata è comunque quella di utilizzare un argomento di Amazon SNS per le notifiche.

4. Scarica l'output del processo (dati dell'archivio) eseguendo il metodo `getJobOutput`.

   Per fornire le informazioni sulla richiesta come il job ID e il nome di vault, crea un'istanza della classe `GetJobOutputRequest`. L'output restituito da Amazon Glacier è disponibile nell'oggetto. `GetJobOutputResult`

```
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.
```

   Il frammento di codice precedente scarica l'intero processo di output. Puoi eventualmente recuperare solo una parte dell'output oppure scaricarlo interamente in blocchi più piccoli specificando l'intervallo di byte in `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 ****");
```

In risposta alla tua `GetJobOutput` chiamata, Amazon Glacier restituisce il checksum della parte dei dati scaricati, se vengono soddisfatte determinate condizioni. Per ulteriori informazioni, consulta Ottenimento di checksum durante il download di dati.

Per verificare che non vi siano errori nel download, puoi quindi calcolare il checksum sul lato client e confrontarlo con il checksum di Amazon Glacier inviato nella risposta.

Per un processo di recupero dall'archivio con l'intervallo facoltativo specificato, quando ottieni la descrizione del lavoro, include il checksum dell'intervallo che stai recuperando (). SHA256 TreeHash Puoi utilizzare questo valore per verificare ulteriormente la precisione dell'intero intervallo di byte che scarichi in seguito. Ad esempio, se avvii un processo per recuperare un intervallo di archivio allineato a una struttura hash e quindi scarichi l'output in blocchi di modo che ogni richiesta `GetJobOutput` restituisca un checksum, puoi calcolare il checksum di ogni parte che scarichi sul lato client e quindi calcolare la struttura hash. Puoi confrontarlo con il checksum restituito da Amazon Glacier in risposta alla tua richiesta di descrizione del lavoro per verificare che l'intero intervallo di byte scaricato corrisponda all'intervallo di byte archiviato in Amazon Glacier.

Per un esempio di utilizzo, consulta Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per Java.

Esempio 1: recupero di un archivio utilizzando l'API di basso livello di AWS SDK per Java

L'esempio di codice Java seguente scarica un archivio dal vault specificato. Una volta completato il processo, l'esempio scarica l'intero output in un'unica chiamata `getJobOutput`. Per un esempio di download di output in blocchi, consulta Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per Java.

L'esempio esegue le seguenti operazioni:

- Crea un argomento ARN Amazon Simple Notification Service (Amazon SNS).

  Amazon Glacier invia una notifica su questo argomento una volta completato il processo.
- Crea una coda Amazon Simple Queue Service (Amazon SQS).

  L'esempio collega una policy alla coda per consentire all'argomento di Amazon SNS di pubblicare messaggi nella coda.

- Avvia un processo per scaricare l'archivio specificato.

  Nella richiesta di lavoro, viene specificato l'argomento Amazon SNS che è stato creato in modo che Amazon Glacier possa pubblicare una notifica sull'argomento dopo il completamento del processo.

- Verifica periodicamente la presenza di un messaggio contenente l'ID processo nella coda di Amazon SQS.

  Se il messaggio esiste, analizza il codice JSON e verifica se il completamento del processo è riuscito. Se è il caso, scarica l'archivio.

- Esegue una pulizia eliminando l'argomento di Amazon SNS e la coda di Amazon SQS creata.

```
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));
    }
}
```

Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per Java

Il seguente esempio di codice Java recupera un archivio da Amazon Glacier. L'esempio di codice scarica l'output del processo in blocchi specificando l'intervallo di byte in un oggetto `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));
    }
}
```

Scaricamento di un archivio in Amazon Glacier utilizzando il AWS SDK per .NET

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for .NET forniscono un metodo per scaricare un archivio.

Argomenti

- Scaricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET
- Scaricamento di un archivio utilizzando l'API di basso livello di AWS SDK per .NET

Scaricamento di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `Download` che puoi utilizzare per scaricare un archivio.

> **Important**
>
> La classe `ArchiveTransferManager` crea un argomento Amazon Simple Notification Service (Amazon SNS) e una coda Amazon Simple Queue Service (Amazon SQS) sottoscritta a quell'argomento. Avvia quindi il processo di recupero di archivio ed esegue il polling della coda affinché l'archivio sia disponibile. Quando l'archivio è disponibile, il download ha inizio. Per informazioni sui tempi di recupero, consulta Opzioni di recupero dall'archivio.

Esempio: download di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

L'esempio di codice C# seguente esegue il download di un archivio da una vault (`examplevault`) nella regione Stati Uniti occidentali (Oregon).

Per step-by-step istruzioni su come eseguire questo esempio, consultaEsecuzione di esempi di codice. Devi aggiornare il codice con un ID archivio esistente e il percorso di file locale in cui intendi salvare l'archivio scaricato come indicato.

```
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);
      }
    }
  }
}
```

Scaricamento di un archivio utilizzando l'API di basso livello di AWS SDK per .NET

Di seguito sono riportati i passaggi per scaricare un archivio Amazon Glacier (Amazon Glacier) utilizzando l'API di basso livello di. AWS SDK per .NET

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   Devi specificare una AWS regione da cui vuoi scaricare l'archivio. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.

2. Avvia un processo `archive-retrieval` eseguendo il metodo `InitiateJob`.

Fornisci informazioni sul lavoro, come l'ID di archivio dell'archivio che desideri scaricare e l'argomento opzionale di Amazon SNS in cui desideri che Amazon Glacier pubblichi un messaggio di completamento del lavoro, creando un'istanza della classe. `InitiateJobRequest` Amazon Glacier restituisce un Job ID in risposta. La risposta è disponibile in un'istanza della 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;
```

Facoltativamente, puoi specificare un intervallo di byte per richiedere ad Amazon Glacier di preparare solo una parte dell'archivio, come mostrato nella seguente richiesta. La richiesta specifica che Amazon Glacier preparerà solo la parte dell'archivio compresa tra 1 MB e 2 MB.

```
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. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica sul vault che identifica un argomento di Amazon Simple Notification Service (Amazon SNS) o hai specificato un argomento Amazon SNS quando hai avviato un processo, Amazon Glacier invia un messaggio a quell'argomento dopo il completamento del processo. L'esempio di codice fornito nella sezione seguente utilizza Amazon SNS per Amazon Glacier per pubblicare un messaggio.

   Puoi anche interrogare Amazon Glacier `DescribeJob` chiamando il metodo per determinare lo stato di completamento del lavoro. La soluzione consigliata è comunque quella di utilizzare un argomento di Amazon SNS per le notifiche.

4. Scarica l'output del processo (dati dell'archivio) eseguendo il metodo `GetJobOutput`.

   Per fornire le informazioni sulla richiesta come il job ID e il nome di vault, crea un'istanza della classe `GetJobOutputRequest`. L'output restituito da Amazon Glacier è disponibile nell'oggetto. `GetJobOutputResponse`

```
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);
  }
}
```

Il frammento di codice precedente scarica l'intero processo di output. Puoi eventualmente recuperare solo una parte dell'output oppure scaricarlo interamente in blocchi più piccoli specificando l'intervallo di byte in `GetJobOutputRequest`.

```
GetJobOutputRequest getJobOutputRequest = new GetJobOutputRequest()
{
  JobId = jobId,
  VaultName = vaultName
};
getJobOutputRequest.SetRange(0, 1048575); // Download only the first 1 MB chunk of
 the output.
```

In risposta alla tua `GetJobOutput` chiamata, Amazon Glacier restituisce il checksum della parte dei dati scaricati, se vengono soddisfatte determinate condizioni. Per ulteriori informazioni, consulta Ottenimento di checksum durante il download di dati.

Per verificare che non vi siano errori nel download, puoi quindi calcolare il checksum sul lato client e confrontarlo con il checksum di Amazon Glacier inviato nella risposta.

Per un processo di recupero dall'archivio con l'intervallo opzionale specificato, quando si ottiene la descrizione del lavoro, include il checksum dell'intervallo che si sta recuperando (SHA256TreeHash). È possibile utilizzare questo valore per verificare ulteriormente la precisione dell'intero intervallo di byte che verrà scaricato in seguito. Ad esempio, se avvii un processo per recuperare un intervallo di archivio allineato a una struttura hash e quindi scarichi l'output in blocchi di modo che ogni richiesta `GetJobOutput` restituisca un checksum, puoi calcolare il checksum di ogni parte che scarichi sul lato client e quindi calcolare la struttura hash. Puoi confrontarlo con il checksum restituito da Amazon Glacier in risposta alla tua richiesta di descrizione del lavoro per verificare che l'intero intervallo di byte scaricato corrisponda all'intervallo di byte archiviato in Amazon Glacier.

Per un esempio di utilizzo, consulta Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per .NET.

Esempio 1: recupero di un archivio utilizzando l'API di basso livello di AWS SDK per .NET

L'esempio di codice C# seguente scarica un archivio dal vault specificato. Una volta completato il processo, l'esempio scarica l'intero output in un'unica chiamata `GetJobOutput`. Per un esempio di

download di output in blocchi, consulta Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per .NET.

L'esempio esegue le seguenti operazioni:

- Configura un argomento Amazon Simple Notification Service (Amazon SNS)

  Amazon Glacier invia una notifica su questo argomento una volta completato il processo.
- Configura una coda Amazon Simple Queue Service (Amazon SQS)

  L'esempio collega una policy alla coda per consentire all'argomento di Amazon SNS di pubblicare messaggi.
- Avvia un processo per scaricare l'archivio specificato.

  Nella richiesta di lavoro, l'esempio specifica l'argomento Amazon SNS in modo che Amazon Glacier possa inviare un messaggio dopo il completamento del processo.
- Verifica periodicamente la presenza di un messaggio nella coda di Amazon SQS.

  Se il messaggio esiste, analizza il codice JSON e verifica se il completamento del processo è riuscito. Se è il caso, scarica l'archivio. L'esempio di codice utilizza la libreria JSON.NET (vedi JSON.NET) per analizzare il codice JSON.
- Esegue una pulizia eliminando l'argomento di Amazon SNS e la coda di Amazon SQS creata.

```
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);
      }
    }
  }
}
```

Esempio 2: recupero di un archivio utilizzando l'API di basso livello di —Download Output in Chunks AWS SDK per .NET

Il seguente esempio di codice C# recupera un archivio da Amazon Glacier. L'esempio di codice scarica l'output del processo in blocchi specificando l'intervallo di byte in un oggetto `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);
      }
    }
  }
}
```

Scaricare un archivio di grandi dimensioni utilizzando l'elaborazione parallela con Python

Questo argomento descrive come scaricare un archivio di grandi dimensioni da Amazon S3 Glacier (S3 Glacier) utilizzando l'elaborazione parallela con Python. Questo approccio consente di scaricare in modo affidabile archivi di qualsiasi dimensione suddividendoli in parti più piccole che possono essere elaborate in modo indipendente.

Panoramica

Lo script Python fornito in questo esempio esegue le seguenti attività:

1. Imposta le AWS risorse necessarie (argomento Amazon SNS e code Amazon SQS) per le notifiche
2. Avvia un processo di recupero dell'archivio con Amazon Glacier
3. Monitora una coda Amazon SQS per le notifiche di completamento dei lavori
4. Divide l'archivio di grandi dimensioni in blocchi gestibili
5. Scarica blocchi in parallelo utilizzando più thread di lavoro
6. Salva ogni blocco su disco per un successivo riassemblaggio

Prerequisiti

Prima di iniziare, assicurati di avere:

- Python 3.6 o successivo installato
- AWS SDK per Python (Boto3) installato
- AWS credenziali configurate con autorizzazioni appropriate per Amazon Glacier, Amazon SNS e Amazon SQS
- Spazio su disco sufficiente per archiviare i blocchi di archivio scaricati

Esempio: scaricare un archivio utilizzando l'elaborazione parallela con Python

Il seguente script Python dimostra come scaricare un archivio di grandi dimensioni da Amazon Glacier utilizzando l'elaborazione parallela:

```
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)
```

Utilizzo dello script

Per utilizzare questo script, procedi nel seguente modo:

1. Sostituisci i valori segnaposto nello script con le tue informazioni specifiche:
   - *output_file_path*: Directory in cui verranno salvati i file frammentari
   - *vault_name*: Nome del vault S3 Glacier
   - *notify_queue_name*: nome per la coda di notifica dei lavori
   - *chunk_download_queue_name*: nome per la coda di download dei blocchi
   - *sns_topic_name*: nome dell'argomento SNS
   - *region*: AWS regione in cui si trova il tuo vault
   - *archive_id*: ID dell'archivio da recuperare
2. Esegui lo script :

   ```
   python download_large_archive.py
   ```
3. Dopo aver scaricato tutti i blocchi, puoi combinarli in un unico file usando un comando come:

   ```
   cat /path/to/chunks/*.chunk > complete_archive.file
   ```

Considerazioni importanti

Quando usi questo script, tieni presente quanto segue:

- Il completamento del recupero dell'archivio da S3 Glacier può richiedere diverse ore, a seconda del livello di recupero selezionato.
- Lo script viene eseguito all'infinito, interrogando continuamente le code. Potresti voler aggiungere una condizione di cessazione in base ai tuoi requisiti specifici.
- Assicurati di avere spazio su disco sufficiente per archiviare tutti i blocchi dell'archivio.

- Se lo script viene interrotto, è possibile riavviarlo `retrieve_archive=False` per continuare a scaricare i blocchi senza avviare un nuovo processo di recupero.
- Regola i *`workers`* parametri *`chunk_size`* and in base alla larghezza di banda della rete e alle risorse di sistema.
- AWS Si applicano le tariffe standard per i recuperi di Amazon S3, Amazon SNS e l'utilizzo di Amazon SQS.

Download di un archivio mediante REST API

Download di un archivio mediante REST API

Il download di un archivio è un processo in due fasi.

1. Avvia un processo di tipo `archive-retrieval`. Per ulteriori informazioni, consulta Initiate Job (POST jobs).
2. Al completamento del processo, esegui il download dei dati dell'archivio. Per ulteriori informazioni, consulta Get Job Output (GET output).

Scaricare un archivio in Amazon Glacier utilizzando il AWS CLI

Puoi scaricare archivi in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: scaricare un archivio utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: scaricare un archivio utilizzando il AWS CLI

> **Note**
>
> Per eseguire il download dei tuoi archivi devi conoscere i tuoi ID archivio. I passaggi 1-4 recupereranno gli ID archivio. Se conosci già gli ID archivio di cui desideri eseguire il download, vai al passaggio 5.

1. Utilizza il comando `initiate-job` per avviare un processo di recupero inventario. Il report sull'inventario elencherà gli ID archivio.

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

   Output previsto:

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

```
    "jobId": "*** jobid ***"
}
```

2. Utilizza il comando `describe-job` per verificare lo stato del comando del processo precedente.

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

Output previsto:

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

3. Attendi il completamento del processo .

Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

Puoi impostare la configurazione delle notifiche per specifici eventi sulla vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier. Amazon Glacier invia un messaggio all'argomento SNS specificato ogni volta che si verifica l'evento specifico.

4. Al termine, utilizzare il comando `get-job-output` per scaricare l'attività di recupero nel file `output.json`. Questo file conterrà gli ID archivio.

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

Questo comando produce un file con i seguenti campi.

```
{
"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. Usa il comando `initiate-job` per avviare il processo di recupero di ogni archivio da una vault. È necessario specificare il parametro del processo come `archive-retrieval`, come da indicazione seguente.

```
aws glacier initiate-job --vault-name awsexamplevault --account-id 111122223333
 --job-parameters="{\"Type\":\"archive-retrieval\",\"ArchiveId\":\"*** archiveId
 ***\"}"
```

6. Attendi il completamento del processo `archive-retrieval`. Utilizza il comando `describe-job` per verificare lo stato del comando precedente.

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

7. Quando il processo precedente è completo, usa il comando `get-job-output` per eseguire il download dell'archivio.

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

Eliminazione di un archivio in Amazon Glacier

Non puoi eliminare un archivio utilizzando la console di gestione Amazon Glacier (Amazon Glacier). Per eliminare un archivio è necessario utilizzare la AWS Command Line Interface (CLI) o scrivere il codice per effettuare una richiesta di eliminazione utilizzando direttamente l'API REST o le AWS SDK per Java librerie wrapper e.NET. I seguenti argomenti spiegano come utilizzare le librerie wrapper e.NET, l'API REST AWS SDK per Java e la. AWS CLI

Argomenti

- Eliminazione di un archivio in Amazon Glacier utilizzando AWS SDK per Java
- Eliminazione di un archivio in Amazon Glacier utilizzando AWS SDK per .NET
- Eliminazione di un archivio Amazon Glacier utilizzando l'API REST
- Eliminazione di un archivio in Amazon Glacier utilizzando AWS Command Line Interface

Puoi eliminare un archivio alla volta da una vault. Per eliminare un archivio, devi indicare il relativo ID nella richiesta di eliminazione. Per ottenere l'ID archivio, scarica l'inventario vault per la vault contenente l'archivio. Per ulteriori informazioni sul download di un inventario vault, consulta Scaricamento di un inventario Vault in Amazon Glacier.

Dopo l'eliminazione di un archivio, puoi ancora eseguire una richiesta di avvio di un processo per recuperare l'archivio eliminato, ma il processo di recupero non riuscirà.

Le operazioni di recupero di archivio che sono in corso per un ID archivio nel momento in cui elimini l'archivio possono riuscire o meno a seconda dei seguenti scenari:

- Se il processo di recupero dell'archivio sta preparando attivamente i dati per il download quando Amazon Glacier riceve la richiesta di eliminazione dell'archivio, l'operazione di recupero dell'archivio potrebbe non riuscire.
- Se il processo di recupero dell'archivio ha preparato correttamente l'archivio per il download quando Amazon Glacier riceve la richiesta di eliminazione dell'archivio, potrai scaricare l'output.

Per ulteriori informazioni sul recupero di archivi, consulta Scaricare un archivio in Amazon Glacier.

Questa operazione è idempotente. L'eliminazione di un archivio già eliminato non genera un errore.

Dopo aver eliminato un archivio, se scarichi immediatamente l'inventario del vault, questo potrebbe includere l'archivio eliminato nell'elenco perché Amazon Glacier prepara l'inventario del vault solo una volta al giorno.

> **Note**
>
> Per l'eliminazione automatica degli archivi del vault, consulta Eliminazione automatica degli archivi del vault in Amazon S3 Glacier.

Eliminazione di un archivio in Amazon Glacier utilizzando AWS SDK per Java

Di seguito sono riportati i passaggi per eliminare un archivio utilizzando l'API di basso livello. AWS SDK per Java

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui è archiviato l'archivio che si desidera eliminare. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DeleteArchiveRequest`.

   Devi fornire un ID archivio, un nome di vault e l'ID del tuo account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDK per Java con Amazon Glacier.
3. Eseguire il metodo `deleteArchive` fornendo l'oggetto della richiesta come parametro.

Il seguente frammento di codice Java illustra la procedura precedente.

```
AmazonGlacierClient client;

DeleteArchiveRequest request = new DeleteArchiveRequest()
    .withVaultName("*** provide a vault name ***")
    .withArchiveId("*** provide an archive ID ***");

client.deleteArchive(request);
```

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Delete Archive (DELETE archive).

Esempio: eliminazione di un archivio utilizzando il AWS SDK per Java

Il seguente esempio di codice Java utilizza AWS SDK per Java per eliminare un archivio. Per step-by-step istruzioni su come eseguire questo esempio, vedereEsecuzione di esempi Java per Amazon Glacier con Eclipse. Devi aggiornare il codice con un nome di vault e l'ID archivio che intendi eliminare come indicato.

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);
        }
    }
}
```

Eliminazione di un archivio in Amazon Glacier utilizzando AWS SDK per .NET

Sia il livello alto che quello di basso livello APIs forniti da Amazon SDK for .NET forniscono un metodo per eliminare un archivio.

Argomenti

- Eliminazione di un archivio utilizzando l'API di alto livello di AWS SDK per .NET
- Eliminazione di un archivio utilizzando l'API di basso livello AWS SDK per .NET

Eliminazione di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

La classe `ArchiveTransferManager` dell'API di alto livello fornisce il metodo `DeleteArchive` che puoi utilizzare per eliminare un archivio.

Esempio: eliminazione di un archivio utilizzando l'API di alto livello di AWS SDK per .NET

Il seguente esempio di codice C# utilizza l'API di alto livello di AWS SDK per .NET per eliminare un archivio. Per step-by-step istruzioni su come eseguire questo esempio, consulta. Esecuzione di esempi di codice Devi aggiornare il codice con l'ID archivio che intendi eliminare come indicato.

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();
    }
  }
}
```

Eliminazione di un archivio utilizzando l'API di basso livello AWS SDK per .NET

Di seguito è riportata la procedura per eliminare un archivio mediante il kit AWS SDK per .NET.

1. Crea un'istanza della classe `AmazonGlacierClient` (client).

   È necessario specificare una AWS regione in cui è archiviato l'archivio che si desidera eliminare. Tutte le operazioni eseguite utilizzando questo client si applicano a quella AWS regione.
2. Fornisci informazioni sulla richiesta creando un'istanza della classe `DeleteArchiveRequest`.

   Devi fornire un ID archivio, un nome di vault e l'ID del tuo account. Se non specifichi un ID account, viene utilizzato l'ID account associato alle credenziali che hai fornito per firmare la richiesta. Per ulteriori informazioni, consulta Utilizzo di AWS SDKs con Amazon Glacier.
3. Eseguire il metodo `DeleteArchive` fornendo l'oggetto della richiesta come parametro.

Esempio: eliminazione di un archivio utilizzando l'API di basso livello del AWS SDK per .NET

L'esempio di codice C# seguente illustra la procedura precedente. L'esempio utilizza l'API di basso livello di AWS SDK per .NET per eliminare un archivio.

> **Note**
>
> Per informazioni sull'API REST sottostante, consulta Delete Archive (DELETE archive).

Per step-by-step istruzioni su come eseguire questo esempio, consultaEsecuzione di esempi di codice. Devi aggiornare il codice con l'ID archivio che intendi eliminare come indicato.

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);
    }
  }
}
```

Eliminazione di un archivio Amazon Glacier utilizzando l'API REST

Puoi utilizzare l'API Delete Archive per eliminare un archivio.

- Per informazioni sull'API Delete Archive, consulta Delete Archive (DELETE archive).
- Per informazioni sull'utilizzo dell'API REST, consulta Riferimento API per Amazon Glacier.

Eliminazione di un archivio in Amazon Glacier utilizzando AWS Command Line Interface

Puoi eliminare gli archivi in Amazon Glacier (Amazon Glacier) utilizzando (). AWS Command Line Interface AWS CLI

Argomenti

- (Prerequisito) Configurazione di AWS CLI
- Esempio: eliminazione di un archivio utilizzando il AWS CLI

(Prerequisito) Configurazione di AWS CLI

1. Scarica e configura la AWS CLI. Per le istruzioni, consulta i seguenti argomenti nella Guida per l'utente dell'AWS Command Line Interface :

   Installazione di AWS Command Line Interface

   Configurazione di AWS Command Line Interface

2. Verifica la AWS CLI configurazione inserendo i seguenti comandi al prompt dei comandi. Poiché entrambi questi comandi non forniscono credenziali in modo esplicito, vengono utilizzate le credenziali del profilo predefinito.

   - Prova a utilizzare il comando help.

     ```
     aws help
     ```

   - Per ottenere un elenco dei vault Amazon Glacier sull'account configurato, usa il comando. `list-vaults` Sostituiscilo *123456789012* con il tuo ID. Account AWS

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

   - Per vedere i dati di configurazione correnti per AWS CLI, usa il `aws configure list` comando.

     ```
     aws configure list
     ```

Esempio: eliminazione di un archivio utilizzando il AWS CLI

1. Utilizzare il comando initiate-job per avviare un'attività di recupero inventario.

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

   Output previsto:

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

2. Utilizzare il comando describe-job per verificare lo stato dell'attività di recupero precedente.

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

Output previsto:

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

3. Attendi il completamento del processo .

   Devi attendere che l'output del processo sia pronto per poter eseguire il download. Se hai impostato una configurazione di notifica nel vault o hai specificato un argomento di Amazon Simple Notification Service (Amazon SNS) quando hai avviato il processo, Amazon Glacier invia un messaggio all'argomento una volta completato il processo.

   Puoi impostare la configurazione delle notifiche per specifici eventi sulla vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier. Amazon Glacier invia un messaggio all'argomento SNS specificato ogni volta che si verifica l'evento specifico.

4. Al termine, utilizzare il comando get-job-output per scaricare l'attività di recupero nel file `output.json`.

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

   Questo comando produce un file con i seguenti campi.

```
{
"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. Utilizzare il comando `delete-archive` per eliminare tutti gli archivi da una vault finché non ne rimane nessuno.

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

Utilizzo di AWS SDKs con Amazon Glacier

AWS ti consente SDKs di sviluppare applicazioni per Amazon Glacier. Le librerie SDK racchiudono l'API Amazon Glacier sottostante, semplificando le attività di programmazione. Ad esempio, per ogni richiesta inviata ad Amazon Glacier, devi includere una firma per autenticare le tue richieste. Quando utilizzi le librerie SDK, devi fornire solo le tue credenziali AWS di sicurezza nel codice e le librerie calcolano la firma necessaria e la includono nella richiesta inviata ad Amazon Glacier. AWS SDKs Forniscono librerie mappate all'API REST sottostante e forniscono oggetti che puoi usare per creare richieste ed elaborare risposte con facilità.

Argomenti

- AWS Librerie SDK per Java e.NET
- Utilizzo di Amazon Glacier con un SDK AWS
- Utilizzo di AWS SDK per Java con Amazon Glacier
- Utilizzo di AWS SDK per .NET con Amazon Glacier

Il AWS Command Line Interface (AWS CLI) è uno strumento unificato per gestire i tuoi file Servizi AWS, incluso Amazon Glacier. Per informazioni sul download di, consulta. AWS CLIAWS Command Line Interface Per un elenco dei comandi CLI di Amazon Glacier, consulta Command Reference.AWS CLI

AWS Librerie SDK per Java e.NET

Le versioni AWS SDKs per Java e.NET offrono librerie wrapper di alto e basso livello.

Puoi trovare esempi di utilizzo di Amazon Glacier utilizzando e in tutta questa AWS SDK per Java guida per AWS SDK per .NET sviluppatori.

Cos'è un'API di basso livello?

Le librerie wrapper di basso livello mappano fedelmente l'API REST sottostante (Riferimento API per Amazon Glacier) supportata da Amazon Glacier. Per ogni operazione REST di Amazon Glacier, l'API di basso livello fornisce un metodo corrispondente, un oggetto di richiesta per fornire le informazioni sulla richiesta e un oggetto di risposta per elaborare la risposta di Amazon Glacier. Le librerie wrapper di basso livello sono l'implementazione più completa delle operazioni sottostanti di Amazon Glacier.

Per informazioni su queste librerie SDK, consulta Utilizzo di AWS SDK per Java con Amazon Glacier e Utilizzo di AWS SDK per .NET con Amazon Glacier.

Cos'è un'API di alto livello?

Per semplificare ulteriormente lo sviluppo di applicazioni, queste librerie offrono un'astrazione di alto livello per alcune delle operazioni. Ad esempio:

- Caricamento di un archivio: per caricare un archivio utilizzando l'API di basso livello, oltre al nome di file e al nome di vault in cui desideri salvare l'archivio, devi fornire anche un checksum (struttura hash SHA-256) del payload. Con l'API di alto livello il checksum viene calcolato automaticamente.
- Download di un archivio o di un inventario vault: per eseguire il download di un archivio utilizzando l'API di basso livello, devi avviare un processo, attenderne il completamento e quindi ottenere l'output del processo. Devi scrivere codice aggiuntivo per configurare un argomento Amazon Simple Notification Service (Amazon SNS) per Amazon Glacier per ricevere una notifica quando il processo è completato. Inoltre, per controllare se un messaggio relativo al completamento del processo è stato pubblicato nell'argomento, devi disporre di un meccanismo di polling. L'API di alto livello fornisce un metodo per il download di un archivio che esegue tutte queste operazioni. In pratica, devi soltanto specificare un ID archivio e il percorso della cartella in cui desideri salvare i dati scaricati.

Per informazioni su queste librerie SDK, consulta Utilizzo di AWS SDK per Java con Amazon Glacier e Utilizzo di AWS SDK per .NET con Amazon Glacier.

Quando utilizzare l'API di alto livello e di basso livello

In generale, dovresti utilizzare l'API di alto livello per la sua semplicità e non perché fornisce i metodi necessari per eseguire un'operazione. Tuttavia, se l'API di alto livello non offre la funzionalità ricercata, puoi utilizzare l'API di basso livello. L'API di basso livello consente inoltre il controllo granulare dell'operazione, come la logica di ripetizione dei tentativi in caso di problemi. Ad esempio, quando carichi un archivio, l'API di alto livello utilizza la dimensione del file per determinare se l'archivio deve essere caricato in un'unica operazione oppure mediante un caricamento in più parti con l'API corrispondente. L'API include anche la logica di ripetizione dei tentativi in caso di errori nel caricamento. Tuttavia, se nel caso della tua applicazione è più importante disporre del controllo granulare, è consigliabile utilizzare l'API di basso livello.

Utilizzo di Amazon Glacier con un SDK AWS

AWS i kit di sviluppo software (SDKs) sono disponibili per molti linguaggi di programmazione più diffusi. Ogni SDK fornisce un'API, esempi di codice, e documentazione che facilitano agli sviluppatori la creazione di applicazioni nel loro linguaggio preferito.

Documentazione sugli SDK	Esempi di codice
AWS SDK per C++	AWS SDK per C++ esempi di codice
AWS CLI	AWS CLI esempi di codice
AWS SDK per Go	AWS SDK per Go esempi di codice
AWS SDK per Java	AWS SDK per Java esempi di codice
AWS SDK per JavaScript	AWS SDK per JavaScript esempi di codice
AWS SDK per Kotlin	AWS SDK per Kotlin esempi di codice
AWS SDK per .NET	AWS SDK per .NET esempi di codice
AWS SDK per PHP	AWS SDK per PHP esempi di codice
AWS Strumenti per PowerShell	AWS Strumenti per PowerShell esempi di codice
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) esempi di codice
AWS SDK per Ruby	AWS SDK per Ruby esempi di codice
AWS SDK per Rust	AWS SDK per Rust esempi di codice
AWS SDK for SAP ABAP	AWS SDK for SAP ABAP esempi di codice
AWS SDK per Swift	AWS SDK per Swift esempi di codice

Per esempi specifici di Amazon Glacier, consulta. Esempi di codice per l'utilizzo di Amazon Glacier AWS SDKs

> **Esempio di disponibilità**
>
> Non riesci a trovare quello che ti serve? Richiedi un esempio di codice utilizzando il link Provide feedback (Fornisci un feedback) nella parte inferiore di questa pagina.

Utilizzo di AWS SDK per Java con Amazon Glacier

AWS SDK per Java Fornisce sia il livello alto che quello basso APIs per Amazon Glacier (Amazon Glacier) come descritto in. Utilizzo di AWS SDKs con Amazon Glacier Per informazioni sul download di AWS SDK per Java, consulta Amazon SDK for Java.

> **Note**
>
> AWS SDK per Java Fornisce client thread-safe per l'accesso ad Amazon Glacier. Come best practice, le tue applicazioni dovrebbero creare un client e riutilizzarlo tra i thread.

Argomenti

- Utilizzo dell'API di basso livello
- Utilizzo dell'API di alto livello
- Esecuzione di esempi Java per Amazon Glacier con Eclipse
- Impostazione dell'endpoint

Utilizzo dell'API di basso livello

La `AmazonGlacierClient` classe di basso livello fornisce tutti i metodi associati alle operazioni REST sottostanti di Amazon Riferimento API per Amazon Glacier Glacier (). Quando chiami uno di questi metodi, devi creare un oggetto di richiesta corrispondente e fornire un oggetto di risposta in cui il metodo possa restituire la risposta di Amazon Glacier all'operazione.

Ad esempio, la classe `AmazonGlacierClient` fornisce il metodo `createVault` per creare una vault. Questo metodo esegue la mappatura all'operazione REST Create Vault sottostante (vedi Create Vault (PUT vault)). Per utilizzare questo metodo, devi creare istanze dell'`CreateVaultResult`oggetto che riceve la risposta Amazon Glacier, come mostrato nel seguente frammento di codice Java:

```
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);
```

Tutti gli esempi di basso livello nella guida utilizzano questo modello.

> **Note**
>
> Il segmento di codice precedente specifica `AccountID` quando si crea la richiesta. Tuttavia, quando si utilizza il AWS SDK per Java, la `AccountId` richiesta è facoltativa e pertanto tutti gli esempi di basso livello in questa guida non impostano questo valore. `AccountId`Questo è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare l' Account AWS ID o, facoltativamente, un «-», nel qual caso Amazon Glacier Account AWS utilizza l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non includere trattini nell'ID. Quando si utilizza AWS SDK per Java, se non si fornisce l'ID dell'account, la libreria imposta l'ID dell'account su '-'.

Utilizzo dell'API di alto livello

Per semplificare ulteriormente lo sviluppo dell'applicazione, AWS SDK per Java fornisce la `ArchiveTransferManager` classe che implementa un'astrazione di livello superiore per alcuni metodi dell'API di basso livello. Questa classe offre metodi utili, ad esempio `upload` e `download` per le operazioni di archivio.

Ad esempio, il frammento di codice Java seguente utilizza il metodo `upload` di alto livello per caricare un archivio.

```
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();
```

Nota che tutte le operazioni eseguite si applicano alla AWS regione specificata durante la creazione dell'oggetto. `ArchiveTransferManager` Se non specificate alcuna AWS Regione, viene AWS SDK per Java impostata `us-east-1` come AWS Regione predefinita.

Tutti gli esempi di alto livello in questa guida utilizzano questo modello.

> **Note**
>
> La classe `ArchiveTransferManager` di alto livello può essere creata con un'istanza `AmazonGlacierClient` o `AWSCredentials`.

Esecuzione di esempi Java per Amazon Glacier con Eclipse

Il modo più semplice di iniziare a utilizzare i codici di esempio Java è installare l'ultima versione di AWS Toolkit for Eclipse. Per informazioni sull'installazione del toolkit più recente o sull'aggiornamento all'ultima versione, visita http://aws.amazon.com/eclipse. Le attività seguenti mostrano in dettaglio come creare e testare i codici di esempio Java forniti in questa sezione.

Procedura generale per la creazione di codici di esempio Java

1	Crea un profilo di credenziali predefinito per AWS le tue credenziali come descritto nell' AWS SDK per Java argomento Fornire AWS credenziali in Amazon SDK for Java.
2	Crea un nuovo progetto AWS Java in Eclipse. Il progetto è pre-configurato con l' AWS SDK per Java.
3	Copiare il codice dalla sezione che si sta leggendo nel progetto.
4	Aggiornare il codice con i dati eventualmente richiesti. Ad esempio, per il caricamento di un file, indicare il percorso del file e il nome del bucket.
5	Eseguire il codice. Verificare l'avvenuta creazione dell'oggetto utilizzando la Console di gestione AWS. Per ulteriori informazioni su Console di gestione AWS, vai a http://aws.amazon.com/console/.

Impostazione dell'endpoint

Per impostazione predefinita, AWS SDK per Java utilizza l'endpoint`https://glacier.us-east-1.amazonaws.com`. Puoi impostare l'endpoint in modo esplicito come illustrato nei frammenti di codice Java in questa sezione.

Il frammento seguente mostra come impostare l'endpoint sulla regione Stati Uniti occidentali (Oregon) (`us-west-2`) nell'API di basso livello.

Example

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

Il frammento seguente mostra come impostare l'endpoint sulla regione Stati Uniti occidentali (Oregon) nell'API di alto livello.

```
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);
```

Per un elenco delle AWS regioni e degli endpoint supportati, consulta. Accesso ad Amazon Glacier

Utilizzo di AWS SDK per .NET con Amazon Glacier

L' AWS SDK per .NET API è disponibile in. `AWSSDK.dll` Per informazioni sul download di AWS SDK per .NET, vai a Sample Code Libraries. Come descritto inUtilizzo di AWS SDKs con Amazon Glacier, AWS SDK per .NET fornisce sia il livello alto che quello basso APIs.

> **Note**
>
> L'API di basso livello e l'API di alto livello forniscono client thread-safe per l'accesso ad Amazon Glacier. Come best practice, le tue applicazioni dovrebbero creare un client e riutilizzarlo tra i thread.

Argomenti

- Utilizzo dell'API di basso livello
- Utilizzo dell'API di alto livello
- Esecuzione di esempi di codice
- Impostazione dell'endpoint

Utilizzo dell'API di basso livello

La `AmazonGlacierClient` classe di basso livello fornisce tutti i metodi associati alle operazioni REST sottostanti di Amazon Glacier (Amazon Glacier) (). Riferimento API per Amazon Glacier Quando chiami uno di questi metodi, devi creare un oggetto di richiesta corrispondente e fornire un oggetto di risposta in cui il metodo possa restituire una risposta Amazon Glacier all'operazione.

Ad esempio, la classe `AmazonGlacierClient` fornisce il metodo `CreateVault` per creare una vault. Questo metodo esegue la mappatura all'operazione REST Create Vault sottostante (vedi Create Vault (PUT vault)). Per utilizzare questo metodo, devi creare istanze delle `CreateVaultResponse` classi `CreateVaultRequest` and per fornire informazioni sulla richiesta e ricevere una risposta Amazon Glacier come mostrato nel seguente frammento di codice C#:

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

CreateVaultRequest request = new CreateVaultRequest()
{
  AccountId = "-",
  VaultName = "*** Provide vault name ***"
};

CreateVaultResponse response = client.CreateVault(request);
```

Tutti gli esempi di basso livello nella guida utilizzano questo modello.

> **Note**
>
> Il segmento di codice precedente specifica `AccountId` quando si crea la richiesta. Tuttavia, quando si utilizza il AWS SDK per .NET, la `AccountId` nella richiesta è facoltativa e pertanto tutti gli esempi di basso livello in questa guida non impostano questo valore. `AccountId`Questo è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare l' Account AWS ID o, facoltativamente, un «-», nel qual caso Amazon Glacier Account AWS utilizza l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non includere trattini nell'ID. Quando si utilizza AWS SDK per .NET, se non si fornisce l'ID dell'account, la libreria imposta l'ID dell'account su '-'.

Utilizzo dell'API di alto livello

Per semplificare ulteriormente lo sviluppo dell'applicazione, AWS SDK per .NET fornisce la `ArchiveTransferManager` classe che implementa un'astrazione di livello superiore per alcuni metodi dell'API di basso livello. Questa classe offre metodi utili, ad esempio `Upload` e `Download`, per le operazioni di archivio.

Ad esempio, il frammento di codice C# seguente utilizza il metodo `Upload` di alto livello per caricare un archivio.

```
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;
```

Nota che tutte le operazioni eseguite si applicano alla AWS regione specificata durante la creazione dell'oggetto. `ArchiveTransferManager` Tutti gli esempi di alto livello in questa guida utilizzano questo modello.

> **Note**
>
> La classe `ArchiveTransferManager` di alto livello necessita ancora del client `AmazonGlacierClient` di basso livello, che puoi passare in modo esplicito, altrimenti `ArchiveTransferManager` crea il client.

Esecuzione di esempi di codice

Il modo più semplice di iniziare a utilizzare i codici di esempio .NET è installare l' AWS SDK per .NET. Per ulteriori informazioni, consulta Amazon SDK per .NET.

La procedura seguente ti consente di testare gli esempi di codice forniti in questa guida.

Procedura generale per la creazione di esempi di codice .NET (mediante Visual Studio)

1	Crea un profilo di credenziali per AWS le tue credenziali come descritto nell'argomento Amazon SDK for .NET Configurazione delle credenziali. AWS
2	Crea un nuovo progetto di Visual Studio utilizzando il modello Progetto AWS vuoto.
3	Sostituire il codice nel file di progetto, `Program.cs` , con il codice riportato nella sezione che si sta leggendo.
4	Eseguire il codice. Verificare l'avvenuta creazione dell'oggetto utilizzando la Console di gestione AWS. Per ulteriori informazioni su Console di gestione AWS, vai a http://aws.amazon.com/console/.

Impostazione dell'endpoint

Per impostazione predefinita, AWS SDK per .NET imposta l'endpoint sulla regione Stati Uniti occidentali (Oregon) (`https://glacier.us-west-2.amazonaws.com`). È possibile impostare l'endpoint su altre AWS regioni, come illustrato nei seguenti frammenti di C#.

Il frammento seguente mostra come impostare l'endpoint sulla regione Stati Uniti occidentali (Oregon) (`us-west-2`) nell'API di basso livello.

Example

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

Il frammento seguente mostra come impostare l'endpoint sulla regione Stati Uniti occidentali (Oregon) nell'API di alto livello.

```
var manager = new ArchiveTransferManager(Amazon.RegionEndpoint.USWest2);
```

Per un elenco aggiornato delle AWS regioni e degli endpoint supportati, consulta. Accesso ad Amazon Glacier

Esempi di codice per l'utilizzo di Amazon Glacier AWS SDKs

I seguenti esempi di codice mostrano come usare Amazon Glacier con AWS un kit di sviluppo software (SDK).

Le operazioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le operazioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Gli scenari sono esempi di codice che mostrano come eseguire un'attività specifica richiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Per un elenco completo di guide ed esempi di codice per sviluppatori AWS SDK, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Nozioni di base

Ciao Amazon Glacier

Il seguente esempio di codice mostra come iniziare a usare Amazon Glacier.

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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}");
        }
    }
}
```

- Per i dettagli sull'API, consulta la ListVaultssezione AWS SDK per .NET API Reference.

Esempi di codice

- Esempi di base per l'utilizzo di Amazon Glacier AWS SDKs
  - Ciao Amazon Glacier
  - Azioni per l'utilizzo di Amazon Glacier AWS SDKs
    - Utilizzo AddTagsToVault con un AWS SDK o una CLI
    - Utilizzo CreateVault con un AWS SDK o una CLI
    - Utilizzo DeleteArchive con un AWS SDK o una CLI
    - Utilizzo DeleteVault con un AWS SDK o una CLI
    - Utilizzo DeleteVaultNotifications con un AWS SDK o una CLI
    - Utilizzo DescribeJob con un AWS SDK o una CLI
    - Utilizzo DescribeVault con un AWS SDK o una CLI
    - Utilizzo GetJobOutput con un AWS SDK o una CLI
    - Utilizzo GetVaultNotifications con un AWS SDK o una CLI
    - Utilizzo InitiateJob con un AWS SDK o una CLI
    - Utilizzo ListJobs con un AWS SDK o una CLI

    - Utilizzo ListTagsForVault con un AWS SDK o una CLI
    - Utilizzo ListVaults con un AWS SDK o una CLI
    - Utilizzo SetVaultNotifications con un AWS SDK o una CLI
    - Utilizzo UploadArchive con un AWS SDK o una CLI
    - Utilizzo UploadMultipartPart con un AWS SDK o una CLI
- Scenari per l'utilizzo di Amazon Glacier AWS SDKs
    - Archivia un file in Amazon Glacier, ricevi notifiche e avvia un processo utilizzando un SDK AWS
    - Ottieni contenuti di archivio Amazon Glacier ed elimina l'archivio utilizzando un SDK AWS

Esempi di base per l'utilizzo di Amazon Glacier AWS SDKs

I seguenti esempi di codice mostrano come utilizzare le nozioni di base di Amazon AWS SDKs Glacier con.

Esempi

- Ciao Amazon Glacier
- Azioni per l'utilizzo di Amazon Glacier AWS SDKs
    - Utilizzo AddTagsToVault con un AWS SDK o una CLI
    - Utilizzo CreateVault con un AWS SDK o una CLI
    - Utilizzo DeleteArchive con un AWS SDK o una CLI
    - Utilizzo DeleteVault con un AWS SDK o una CLI
    - Utilizzo DeleteVaultNotifications con un AWS SDK o una CLI
    - Utilizzo DescribeJob con un AWS SDK o una CLI
    - Utilizzo DescribeVault con un AWS SDK o una CLI
    - Utilizzo GetJobOutput con un AWS SDK o una CLI
    - Utilizzo GetVaultNotifications con un AWS SDK o una CLI
    - Utilizzo InitiateJob con un AWS SDK o una CLI
    - Utilizzo ListJobs con un AWS SDK o una CLI
    - Utilizzo ListTagsForVault con un AWS SDK o una CLI
    - Utilizzo ListVaults con un AWS SDK o una CLI
    - Utilizzo SetVaultNotifications con un AWS SDK o una CLI

- Utilizzo UploadArchive con un AWS SDK o una CLI
- Utilizzo UploadMultipartPart con un AWS SDK o una CLI

Ciao Amazon Glacier

Il seguente esempio di codice mostra come iniziare a usare Amazon Glacier.

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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}");
```

```
        }
    }
}
```

- Per i dettagli sull'API, consulta la ListVaultssezione AWS SDK per .NET API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo di Amazon Glacier con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Azioni per l'utilizzo di Amazon Glacier AWS SDKs

I seguenti esempi di codice mostrano come eseguire singole azioni di Amazon Glacier con. AWS SDKs Ogni esempio include un collegamento a GitHub, dove puoi trovare le istruzioni per la configurazione e l'esecuzione del codice.

Questi estratti chiamano l'API Amazon Glacier e sono estratti di codice da programmi più grandi che devono essere eseguiti nel contesto. È possibile visualizzare le azioni nel contesto in Scenari per l'utilizzo di Amazon Glacier AWS SDKs .

Gli esempi seguenti includono solo le operazioni più comunemente utilizzate. Per un elenco completo, consulta l'Amazon Glacier API Reference.

Esempi

- Utilizzo AddTagsToVault con un AWS SDK o una CLI
- Utilizzo CreateVault con un AWS SDK o una CLI
- Utilizzo DeleteArchive con un AWS SDK o una CLI
- Utilizzo DeleteVault con un AWS SDK o una CLI
- Utilizzo DeleteVaultNotifications con un AWS SDK o una CLI
- Utilizzo DescribeJob con un AWS SDK o una CLI
- Utilizzo DescribeVault con un AWS SDK o una CLI
- Utilizzo GetJobOutput con un AWS SDK o una CLI
- Utilizzo GetVaultNotifications con un AWS SDK o una CLI
- Utilizzo InitiateJob con un AWS SDK o una CLI

- Utilizzo ListJobs con un AWS SDK o una CLI
- Utilizzo ListTagsForVault con un AWS SDK o una CLI
- Utilizzo ListVaults con un AWS SDK o una CLI
- Utilizzo SetVaultNotifications con un AWS SDK o una CLI
- Utilizzo UploadArchive con un AWS SDK o una CLI
- Utilizzo UploadMultipartPart con un AWS SDK o una CLI

Utilizzo **`AddTagsToVault`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `AddTagsToVault`.

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
    }
```

- Per i dettagli sull'API, consulta la AddTagsToVaultsezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente aggiunge due tag a un vault denominato `my-vault`:

```
aws glacier add-tags-to-vault --account-id - --vault-name my-vault --
tags id=1234,date=july2015
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta AddTagsToVault AWS CLICommand Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo di Amazon Glacier con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`CreateVault`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `CreateVault`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
}
```

- Per i dettagli sull'API, consulta la CreateVaultsezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente crea un nuovo vault denominato `my-vault`:

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta CreateVault AWS CLICommand Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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);
        }
    }
}
```

- Per i dettagli sull'API, consulta la CreateVaultsezione AWS SDK for Java 2.x API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Crea il 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 };
```

Crea la vault.

```
// 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();
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori di AWS SDK per JavaScript.
- Per i dettagli sull'API, consulta la CreateVaultsezione AWS SDK per JavaScript API Reference.

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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!");
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori di AWS SDK per JavaScript.
- Per i dettagli sull'API, consulta la CreateVaultsezione AWS SDK per JavaScript API Reference.

PowerShell

Strumenti per PowerShell V4

Esempio 1: crea un nuovo archivio per l'account dell'utente. Poiché non è stato fornito alcun valore al AccountId parametro -, i cmdlet utilizzano un valore predefinito di «-» che indica l'account corrente.

```
New-GLCVault -VaultName myvault
```

Output:

```
/01234567812/vaults/myvault
```

- Per i dettagli sull'API, vedere CreateVaultin AWS Strumenti per PowerShell Cmdlet Reference (V4).

Strumenti per V5 PowerShell

Esempio 1: crea un nuovo archivio per l'account dell'utente. Poiché non è stato fornito alcun valore al AccountId parametro -, i cmdlet utilizzano un valore predefinito di «-» che indica l'account corrente.

```
New-GLCVault -VaultName myvault
```

Output:

```
/01234567812/vaults/myvault
```

- Per i dettagli sull'API, vedere CreateVaultin AWS Strumenti per PowerShell Cmdlet Reference (V5).

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta CreateVault AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`DeleteArchive`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `DeleteArchive`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Come ottenere i contenuti dell'archivio ed eliminare l'archivio

CLI

AWS CLI

Eliminazione di un archivio da un vault

L'esempio `delete-archive` seguente rimuove l'archivio specificato da `example_vault`.

```
aws glacier delete-archive \
    --account-id 111122223333  \
    --vault-name example_vault \
    --archive-id Sc0u9ZP8yaWkmh-XGlIvAVprtLhaLCGnNwNl5I5x9HqPIkX5mjc0DrId3Ln-
Gi_k2HzmlIDZUz117KSdVMdMXLuFWi9PJUitxWO73edQ43eTlMWkH0pd9zVSAuV_XXZBVhKhyGhJ7w
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteArchive AWS CLICommand Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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);
        }
    }
}
```

- Per i dettagli sull'API, consulta la DeleteArchivesezione AWS SDK for Java 2.x API Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta DeleteArchive AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`DeleteVault`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `DeleteVault`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Come ottenere i contenuti dell'archivio ed eliminare l'archivio

CLI

AWS CLI

Il comando seguente elimina un vault denominato `my-vault`:

```
aws glacier delete-vault --vault-name my-vault --account-id -
```

Il comando non produce output. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DeleteVault AWS CLICommand Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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);
        }
    }
}
```

- Per i dettagli sull'API, consulta la DeleteVaultsezione AWS SDK for Java 2.x API Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta DeleteVault AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`DeleteVaultNotifications`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `DeleteVaultNotifications`.

CLI

AWS CLI

Rimuovere le notifiche SNS per un vault

L'esempio `delete-vault-notifications` seguente rimuove le notifiche inviate da Amazon Simple Notification Service (Amazon SNS) per il vault specificato.

```
aws glacier delete-vault-notifications \
    --account-id 111122223333 \
    --vault-name example_vault
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteVaultNotifications AWS CLICommand Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta DeleteVaultNotifications AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`DescribeJob`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `DescribeJob`.

CLI

AWS CLI

Il comando seguente recupera le informazioni su un processo di recupero dell'inventario su un vault denominato `my-vault`:

```
aws glacier describe-job --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW
```

Output:

```
{
    "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 del processo si trova nell'output di `aws glacier initiate-job` e `aws glacier list-jobs`. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DescribeJob AWS CLICommand Reference.

PowerShell

Strumenti per PowerShell V4

Esempio 1: restituisce i dettagli del lavoro specificato. Quando il processo viene completato correttamente, è possibile utilizzare il cmdlet GCJob Read-Output per recuperare il contenuto del processo (un archivio o un elenco di inventario) nel file system locale.

```
Get-GLCJob -VaultName myvault -JobId "op1x...JSbthM"
```

Output:

```
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
```

- Per i dettagli sull'API, vedere DescribeJobin AWS Strumenti per PowerShell Cmdlet Reference (V4).

Strumenti per V5 PowerShell

Esempio 1: restituisce i dettagli del lavoro specificato. Quando il processo viene completato correttamente, è possibile utilizzare il cmdlet GCJob Read-Output per recuperare il contenuto del processo (un archivio o un elenco di inventario) nel file system locale.

```
Get-GLCJob -VaultName myvault -JobId "op1x...JSbthM"
```

Output:

```
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
```

- Per i dettagli sull'API, vedere DescribeJobin AWS Strumenti per PowerShell Cmdlet Reference (V5).

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta DescribeJob AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`DescribeVault`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `DescribeVault`.

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
    }
```

- Per i dettagli sull'API, consulta la DescribeVaultsezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente recupera i dati relativi a un vault denominato `my-vault`:

```
aws glacier describe-vault --vault-name my-vault --account-id -
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DescribeVault AWS CLICommand Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo di Amazon Glacier con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`GetJobOutput`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `GetJobOutput`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Come ottenere i contenuti dell'archivio ed eliminare l'archivio

CLI

AWS CLI

Il comando seguente salva l'output di un processo di inventario del vault in un file nella directory corrente denominata `output.json`:

```
aws glacier get-job-output --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61ds04CnMW output.json
```

`job-id` è disponibile nell'output di `aws glacier list-jobs`. Nota che il nome del file di output è un argomento posizionale non preceduto dal nome di un'opzione. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Output:

```
{
    "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
```

- Per i dettagli sull'API, consulta GetJobOutput AWS CLICommand Reference.

PowerShell

Strumenti per PowerShell V4

Esempio 1: scarica il contenuto dell'archivio di cui è stato pianificato il recupero nel processo specificato e lo archivia in un file su disco. Il download convalida automaticamente il checksum, se disponibile. Se lo si desidera, è possibile restituire l'intera risposta, incluso il checksum, specificando. **`-Select '*'`**

```
Read-GLCJobOutput -VaultName myvault -JobId "HSWjArc...Zq2XLiW" -FilePath "c:
\temp\blue.bin"
```

- Per i dettagli sull'API, vedere GetJobOutputin AWS Strumenti per PowerShell Cmdlet Reference (V4).

Strumenti per V5 PowerShell

Esempio 1: scarica il contenuto dell'archivio pianificato per il recupero nel processo specificato e lo archivia in un file su disco. Il download convalida automaticamente il checksum, se disponibile. Se lo si desidera, è possibile restituire l'intera risposta, incluso il checksum, specificando. **`-Select '*'`**

```
Read-GLCJobOutput -VaultName myvault -JobId "HSWjArc...Zq2XLiW" -FilePath "c:
\temp\blue.bin"
```

- Per i dettagli sull'API, vedere GetJobOutputin AWS Strumenti per PowerShell Cmdlet Reference (V5).

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta GetJobOutput AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`GetVaultNotifications`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `GetVaultNotifications`.

CLI

AWS CLI

Il comando seguente ottiene una descrizione della configurazione della notifica per un vault denominato `my-vault`:

```
aws glacier get-vault-notifications --account-id - --vault-name my-vault
```

Output:

```
{
    "vaultNotificationConfig": {
        "Events": [
            "InventoryRetrievalCompleted",
            "ArchiveRetrievalCompleted"
        ],
        "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault"
    }
}
```

Se non è stata configurata alcuna notifica per il vault, viene restituito un errore. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta GetVaultNotifications AWS CLICommand Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta GetVaultNotifications AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`InitiateJob`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `InitiateJob`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Recupera un archivio da un caveau. Questo esempio utilizza la classe. ArchiveTransferManager Per i dettagli sull'API, vedere 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}%");
        }
    }
```

- Per i dettagli sulle API, consulta la InitiateJobsezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente avvia un processo per ottenere un inventario del `my-vault` vault:

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

Output:

```
{
    "location": "/0123456789012/vaults/my-vault/jobs/
zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
    "jobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW"
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Il comando seguente avvia un processo per recuperare un archivio dal vault: `my-vault`

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-
parameters file://job-archive-retrieval.json
```

`job-archive-retrieval.json`è un file JSON nella cartella locale che specifica il tipo di lavoro, l'ID di archivio e alcuni parametri opzionali:

```
{
  "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 Gli archivi sono disponibili nell'output di `aws glacier upload-archive` e. `aws glacier get-job-output`

Output:

```
{
    "location": "/011685312445/vaults/mwunderl/jobs/l7IL5-
EkXyEY9Ws95fClzIbk2O5uLYaFdAYOi-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav",
    "jobId": "l7IL5-EkXy2O5uLYaFdAYOiEY9Ws95fClzIbk-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav"
}
```

Per informazioni sul formato dei parametri del processo, consulta Initiate Job nell'Amazon Glacier API Reference.

- Per i dettagli sull'API, consulta Command InitiateJobReference AWS CLI .

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Recupera un inventario del caveau.

```
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);

        }
    }
```

```
}
```

- Per i dettagli sull'API, consulta la sezione AWS SDK for Java 2.x API InitiateJobReference.

PowerShell

Strumenti per PowerShell V4

Esempio 1: avvia un processo per recuperare un archivio dal vault specificato di proprietà dell'utente. Lo stato del processo può essere verificato utilizzando il cmdlet Get-GLCJob . Quando il processo viene completato correttamente, è possibile utilizzare il cmdlet GCJob Read-Output per recuperare il contenuto dell'archivio nel file system locale.

```
Start-GLCJob -VaultName myvault -JobType "archive-retrieval" -JobDescription
 "archive retrieval" -ArchiveId "o9O9j...TX-TpIhQJw"
```

Output:

```
JobId            JobOutputPath Location
-----            ------------- --------
op1x...JSbthM                  /012345678912/vaults/test/jobs/
op1xe...I4HqCHkSJSbthM
```

- Per i dettagli sull'API, vedere InitiateJobin AWS Strumenti per PowerShell Cmdlet Reference (V4).

Strumenti per V5 PowerShell

Esempio 1: avvia un processo per recuperare un archivio dal vault specificato di proprietà dell'utente. Lo stato del processo può essere verificato utilizzando il cmdlet Get-GLCJob . Quando il processo viene completato correttamente, è possibile utilizzare il cmdlet GCJob Read-Output per recuperare il contenuto dell'archivio nel file system locale.

```
Start-GLCJob -VaultName myvault -JobType "archive-retrieval" -JobDescription
 "archive retrieval" -ArchiveId "o9O9j...TX-TpIhQJw"
```

Output:

```
JobId            JobOutputPath Location
-----            ------------- --------
```

```
op1x...JSbthM                  /012345678912/vaults/test/jobs/
op1xe...I4HqCHkSJSbthM
```

- Per i dettagli sull'API, vedere InitiateJobin AWS Strumenti per PowerShell Cmdlet Reference (V5).

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Recupera un inventario del caveau.

```
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
```

Recupera un archivio da un vault.

```
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
```

- Per i dettagli sull'API, consulta InitiateJob AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`ListJobs`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `ListJobs`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nei seguenti esempi di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo
- Come ottenere i contenuti dell'archivio ed eliminare l'archivio

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
    }
```

- Per i dettagli sull'API, consulta la ListJobssezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente elenca i lavori in corso e quelli completati di recente per un vault denominato `my-vault`:

```
aws glacier list-jobs --account-id - --vault-name my-vault
```

Output:

```
{
    "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 richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListJobs AWS CLICommand Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta ListJobs AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`ListTagsForVault`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `ListTagsForVault`.

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
    }
```

- Per i dettagli sull'API, consulta la ListTagsForVaultsezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente consente di elencare i tag applicati a un vault denominato `my-vault`:

```
aws glacier list-tags-for-vault --account-id - --vault-name my-vault
```

Output:

```
{
    "Tags": {
        "date": "july2015",
        "id": "1234"
    }
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListTagsForVault AWS CLICommand Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo di Amazon Glacier con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`ListVaults`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `ListVaults`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
}
```

- Per i dettagli sull'API, consulta la ListVaultssezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente consente di elencare i vault nell'account e nella regione predefiniti:

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

Output:

```
{
    "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 richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListVaults AWS CLICommand Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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);
        }
    }
}
```

- Per i dettagli sull'API, consulta la ListVaultssezione AWS SDK for Java 2.x API Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta ListVaults AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`SetVaultNotifications`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `SetVaultNotifications`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo

CLI

AWS CLI

Il comando seguente configura le notifiche SNS per un vault denominato `my-vault`:

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

`notificationconfig.json` è un file JSON nella cartella corrente che specifica un argomento SNS e gli eventi da pubblicare:

```
{
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
  "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta SetVaultNotifications AWS CLICommand Reference.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta SetVaultNotifications AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`UploadArchive`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `UploadArchive`.

Gli esempi di operazioni sono estratti di codice da programmi più grandi e devono essere eseguiti nel contesto. È possibile visualizzare questa operazione nel contesto nel seguente esempio di codice:

- Archiviazione di un file, ricezione di notifiche e avvio di processo

.NET

SDK per .NET

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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;
       }
   }
```

- Per i dettagli sull'API, consulta la UploadArchivesezione AWS SDK per .NET API Reference.

CLI

AWS CLI

Il comando seguente carica un archivio nella cartella corrente denominata `archive.zip` in un archivio denominato `my-vault`:

```
aws glacier upload-archive --account-id - --vault-name my-vault --
body archive.zip
```

Output:

```
{
    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--
zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
    "checksum":
 "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/0123456789012/vaults/my-vault/archives/
kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Per recuperare un archivio caricato, avvia un processo di recupero con il comando initiate-job di aws glacier.

- Per i dettagli sull'API, consulta UploadArchive AWS CLICommand Reference.

Java

SDK per Java 2.x

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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();
    }
}
```

- Per i dettagli sull'API, consulta la UploadArchivesezione AWS SDK for Java 2.x API Reference.

JavaScript

SDK per JavaScript (v3)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Crea il 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 };
```

Caricamento dell'archivio.

```
// 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();
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori di AWS SDK per JavaScript.
- Per i dettagli sull'API, consulta la UploadArchivesezione AWS SDK per JavaScript API Reference.

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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);
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori di AWS SDK per JavaScript.
- Per i dettagli sull'API, consulta la UploadArchivesezione AWS SDK per JavaScript API Reference.

PowerShell

Strumenti per PowerShell V4

Esempio 1: carica un singolo file nel vault specificato, restituendo l'ID dell'archivio e il checksum calcolato.

```
Write-GLCArchive -VaultName myvault -FilePath c:\temp\blue.bin
```

Output:

```
FilePath                  ArchiveId             Checksum
--------                  ---------             --------
C:\temp\blue.bin          o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
```

Esempio 2: carica il contenuto di una gerarchia di cartelle nell'archivio specificato nell'account dell'utente. Per ogni file caricato, il cmdlet emette il nome del file, l'ID di archivio corrispondente e il checksum calcolato dell'archivio.

```
Write-GLCArchive -VaultName myvault -FolderPath . -Recurse
```

Output:

```
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
```

- Per i dettagli sull'API, vedere UploadArchivein Cmdlet Reference (V4).AWS Strumenti per PowerShell

Strumenti per V5 PowerShell

Esempio 1: carica un singolo file nel vault specificato, restituendo l'ID dell'archivio e il checksum calcolato.

```
Write-GLCArchive -VaultName myvault -FilePath c:\temp\blue.bin
```

Output:

```
FilePath                    ArchiveId             Checksum
--------                    ---------             --------
C:\temp\blue.bin            o9O9jUUs...TTX-TpIhQJw 79f3e...f4395b
```

Esempio 2: carica il contenuto di una gerarchia di cartelle nell'archivio specificato nell'account dell'utente. Per ogni file caricato, il cmdlet emette il nome del file, l'ID di archivio corrispondente e il checksum calcolato dell'archivio.

```
Write-GLCArchive -VaultName myvault -FolderPath . -Recurse
```

Output:

```
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
```

- Per i dettagli sull'API, vedere UploadArchivein Cmdlet Reference (V5).AWS Strumenti per PowerShell

Python

SDK per Python (Boto3)

> **Note**
>
> C'è di più su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice 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
```

- Per i dettagli sull'API, consulta UploadArchive AWSSDK for Python (Boto3) API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Utilizzo **`UploadMultipartPart`** con un AWS SDK o una CLI

Gli esempi di codice seguenti mostrano come utilizzare `UploadMultipartPart`.

CLI

AWS CLI

Il comando seguente carica la prima parte da 1 MiB (1024 x 1024 byte) di un archivio:

```
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 richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Il parametro body indica il percorso di un file delle parti sul filesystem locale. Il parametro range utilizza un intervallo di contenuti HTTP che indica i byte occupati dalla parte nell'archivio completato. L'ID di caricamento viene restituito dal comando `aws glacier initiate-multipart-upload` e può essere ottenuto anche utilizzando `aws glacier list-multipart-uploads`.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. UploadMultipartPartAWS CLI

JavaScript

SDK per JavaScript (v2)

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Creazione di un caricamento in più parti di blocchi da 1 megabyte di un oggetto 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);
        }
      });
    });
  }
});
```

- Per ulteriori informazioni, consulta la Guida per sviluppatori di AWS SDK per JavaScript.
- Per i dettagli sull'API, consulta la UploadMultipartPartsezione AWS SDK per JavaScript API Reference.

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consultaUtilizzo di Amazon Glacier con un SDK AWS. Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Scenari per l'utilizzo di Amazon Glacier AWS SDKs

I seguenti esempi di codice mostrano come implementare scenari comuni in Amazon Glacier con. AWS SDKs Questi scenari mostrano come eseguire attività specifiche richiamando più funzioni all'interno di Amazon Glacier o combinandole con altre. Servizi AWS Ogni scenario include un collegamento al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice.

Gli scenari sono relativi a un livello intermedio di esperienza per aiutarti a comprendere le azioni di servizio nel contesto.

Esempi

- Archivia un file in Amazon Glacier, ricevi notifiche e avvia un processo utilizzando un SDK AWS
- Ottieni contenuti di archivio Amazon Glacier ed elimina l'archivio utilizzando un SDK AWS

Archivia un file in Amazon Glacier, ricevi notifiche e avvia un processo utilizzando un SDK AWS

L'esempio di codice seguente mostra come:

- Crea un vault Amazon Glacier.
- Configurare la vault per pubblicare le notifiche in un argomento Amazon SNS.
- Caricare il file di archivio sulla vault.
- Avvia un processo di recupero di archivio.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Crea una classe che racchiuda le operazioni di 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
```

Chiama le funzioni sulla classe wrapper per creare una vault e caricare un file, quindi configura la vault per pubblicare notifiche e avviare un processo per recuperare l'archivio.

```
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.")
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella Documentazione di riferimento delle API SDK AWS per Python (Boto3).
  - CreateVault
  - InitiateJob
  - ListJobs
  - ListVaults
  - SetVaultNotifications
  - UploadArchive

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Ottieni contenuti di archivio Amazon Glacier ed elimina l'archivio utilizzando un SDK AWS

L'esempio di codice seguente mostra come:

- Elenca le offerte di lavoro per un vault Amazon Glacier e visualizza lo stato del lavoro.
- Ottenere l'output di un processo di recupero dell'archivio completato.
- Eliminare un archivio.
- Eliminare una vault.

Python

SDK per Python (Boto3)

> **Note**
>
> C'è altro su. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Crea una classe che racchiuda le operazioni di 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
```

Chiama le funzioni della classe wrapper per ottenere il contenuto dell'archivio da un processo completato ed eliminare l'archivio.

```
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)
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella Documentazione di riferimento delle API SDK AWS per Python (Boto3).
  - DeleteArchive
  - DeleteVault
  - GetJobOutput
  - ListJobs

Per un elenco completo delle guide per sviluppatori AWS SDK e degli esempi di codice, consulta. Utilizzo di Amazon Glacier con un SDK AWS Questo argomento include anche informazioni su come iniziare e dettagli sulle versioni precedenti dell'SDK.

Sicurezza in Amazon Glacier

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di un data center e di un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS te e te. Il modello di responsabilità condivisa descrive questo come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi in Cloud AWS. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. L'efficacia della nostra sicurezza è regolarmente testata e verificata da revisori di terze parti come parte dei programmi di conformitàAWS. Per informazioni sui programmi di conformità applicabili ad Amazon Glacier (Amazon Glacier), consulta Services in Scope by Compliance Program.AWS
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile per altri fattori, tra cui la riservatezza dei dati, i requisiti dell'azienda e leggi e normative applicabili.

Questa documentazione ti aiuterà a capire come applicare il modello di responsabilità condivisa quando usi Amazon Glacier. I seguenti argomenti mostrano come configurare Amazon Glacier per raggiungere i tuoi obiettivi di sicurezza e conformità. Imparerai anche a utilizzare altri AWS servizi che possono aiutarti a monitorare e proteggere le tue risorse Amazon Glacier.

Argomenti

- Protezione dei dati in Amazon Glacier
- Identity and Access Management per Amazon Glacier
- Registrazione e monitoraggio in Amazon Glacier
- Convalida della conformità per Amazon Glacier
- Resilienza in Amazon Glacier
- Sicurezza dell'infrastruttura in Amazon Glacier

Protezione dei dati in Amazon Glacier

Amazon Glacier (Amazon Glacier) offre uno storage cloud altamente durevole per l'archiviazione dei dati e il backup a lungo termine. Amazon Glacier è progettato per offrire una durabilità del 99,76%

e offre funzionalità complete di sicurezza e conformità che possono aiutarti a soddisfare i severi requisiti normativi. Amazon Glacier archivia i dati in modo ridondante in AWS più zone di disponibilità (AZ) e su più dispositivi all'interno di ciascuna zona di disponibilità. Per aumentare la durabilità, Amazon Glacier archivia i dati in modo sincrono su AZs più file prima di confermare il corretto caricamento.

Per ulteriori informazioni sull'infrastruttura cloud AWS globale, consulta Global Infrastructure.

Ai fini della protezione dei dati, ti consigliamo di proteggere Account AWS le credenziali e di concedere a singoli utenti, gruppi o ruoli solo le autorizzazioni necessarie per svolgere le loro mansioni lavorative.

Se hai bisogno di moduli crittografici convalidati FIPS 140-2 per l'accesso AWS tramite un'interfaccia a riga di comando o un'API, utilizza un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il Federal Information Processing Standard (FIPS) 140-2.

Argomenti

- Crittografia dei dati
- Gestione delle chiavi
- Riservatezza del traffico Internet

Crittografia dei dati

La protezione dei dati si riferisce alla protezione dei dati durante il transito (mentre viaggiano da e verso Amazon Glacier) e a riposo (mentre sono archiviati nei data center). AWS Puoi proteggere i dati in transito che vengono caricati direttamente su Amazon Glacier utilizzando Secure Sockets Layer (SSL) o la crittografia lato client.

Puoi accedere ad Amazon Glacier anche tramite Amazon S3. Amazon S3 supporta la configurazione del ciclo di vita su un bucket Amazon S3, che consente di trasferire gli oggetti alla classe di storage Amazon Glacier per l'archiviazione. I dati in transito tra Amazon S3 e Amazon Glacier tramite politiche del ciclo di vita sono crittografati tramite SSL.

I dati inattivi archiviati in Amazon Glacier vengono automaticamente crittografati lato server utilizzando Advanced Encryption Standard (AES-256) a 256 bit con chiavi gestite da. AWS Se preferisci gestire le tue chiavi, puoi anche utilizzare la crittografia lato client prima di archiviare i dati in Amazon Glacier. Per ulteriori informazioni sulla configurazione della crittografia predefinita di Amazon

S3, consulta Crittografia predefinita di Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

Gestione delle chiavi

La crittografia lato server affronta la crittografia dei dati a riposo, ovvero Amazon Glacier crittografa i tuoi dati mentre li scrive nei suoi data center e li decrittografa per te quando accedi. Se la richiesta è autenticata e sono disponibili le autorizzazioni per l'accesso, non c'è differenza nelle modalità di accesso ai dati, crittografati o meno.

I dati inattivi archiviati in Amazon Glacier vengono automaticamente crittografati lato server utilizzando AES-256, utilizzando chiavi gestite da. AWS Come ulteriore protezione, AWS cripta la chiave stessa con una chiave radice che ruotiamo regolarmente.

Riservatezza del traffico Internet

L'accesso ad Amazon Glacier tramite la rete avviene tramite published. AWS APIs I client devono supportare Transport Layer Security (TLS) 1.2. È consigliabile TLS 1.3 o versioni successive. I client devono inoltre supportare le suite di cifratura con PFS (Perfect Forward Secrecy), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Diffie-Hellman Ephemeral (ECDHE). La maggior parte dei sistemi moderni come Java 7 e versioni successive, supporta tali modalità. Inoltre, è necessario firmare le richieste utilizzando un ID chiave di accesso e una chiave di accesso segreta associate a un principale IAM, oppure è possibile utilizzare AWS Security Token Service (AWS STS) per generare le credenziali di sicurezza temporanee per firmare le richieste.

Endpoint VPC

Un endpoint di cloud privato virtuale (VPC) ti consente di connettere privatamente il tuo VPC ai servizi supportati AWS e ai servizi endpoint VPC basati su AWS PrivateLink senza richiedere un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect Sebbene Amazon Glacier non supporti direttamente gli endpoint VPC, puoi sfruttare gli endpoint VPC di Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) VPC se accedi ad Amazon Glacier come livello di storage integrato con Amazon S3.

Per ulteriori informazioni sulla configurazione del ciclo di vita di Amazon S3 e sulla transizione degli oggetti alla classe di storage Amazon Glacier, consulta Object Lifecycle Management and Transitioning Objects nella Amazon Simple Storage Service User Guide. Per ulteriori informazioni sugli endpoint VPC, consulta Endpoint VPC nella Guida per l'utente di Amazon VPC.

Identity and Access Management per Amazon Glacier

AWS Identity and Access Management (IAM) è un software Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle risorse. AWS Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse Amazon Glacier. IAM è uno strumento Servizio AWS che puoi utilizzare senza costi aggiuntivi.

Argomenti

- Destinatari
- Autenticazione con identità
- Gestione dell’accesso con l’utilizzo delle policy
- Come funziona Amazon Glacier con IAM
- Esempi di policy basate sull'identità per Amazon Glacier
- Esempi di policy basate sulle risorse per Amazon Glacier
- Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Glacier
- Riferimento alle autorizzazioni API

Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia in base al tuo ruolo:

- Utente del servizio: richiedi le autorizzazioni all'amministratore se non riesci ad accedere alle funzionalità (vedi) Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Glacier
- Amministratore del servizio: determina l'accesso degli utenti e invia le richieste di autorizzazione (vediCome funziona Amazon Glacier con IAM)
- Amministratore IAM: scrivi politiche per gestire l'accesso (vediEsempi di policy basate sull'identità per Amazon Glacier)

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM. Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come AWS IAM Identity Center (IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/ Facebook Per ulteriori informazioni sull'accesso, consulta Come accedere al tuo nella Guida per l'utente Account AWS.Accedi ad AWS

Per l'accesso programmatico, AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta AWS Signature Version 4 per le richieste API nella IAM User Guide.

Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Ti consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali dell'utente root, consulta Attività che richiedono le credenziali dell'utente root nella Guida per l'utente IAM.

Identità federata

Come best practice, richiedi agli utenti umani di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory aziendale, del provider di identità Web o Directory Service che accede Servizi AWS utilizzando le credenziali di una fonte di identità. Le identità federate assumono ruoli che forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo. AWS IAM Identity Center Per ulteriori informazioni, consulta Cos'è IAM Identity Center? nella Guida per l'utente di AWS IAM Identity Center .

Utenti e gruppi IAM

Un utente IAM è un'identità con autorizzazioni specifiche per una singola persona o applicazione. Consigliamo di utilizzare credenziali temporanee anziché utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee nella Guida per l'utente IAM.

Un gruppo IAM specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per grandi gruppi di utenti. Per ulteriori informazioni, consulta Casi d'uso per utenti IAM nella Guida per l'utente IAM.

Ruoli IAM

Un ruolo IAM è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo passando da un utente a un ruolo IAM (console) o chiamando un'operazione AWS CLI o AWS API. Per ulteriori informazioni, consulta Metodi per assumere un ruolo nella Guida per l'utente IAM.

I ruoli IAM sono utili per l'accesso federato degli utenti, le autorizzazioni utente IAM temporanee, l'accesso tra account, l'accesso tra servizi e le applicazioni in esecuzione su Amazon. EC2 Per maggiori informazioni, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente di IAM.

Gestione dell'accesso con l'utilizzo delle policy

Puoi controllare l'accesso AWS creando policy e collegandole a identità o risorse. AWS Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa. AWS valuta queste politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sui documenti relativi alle policy JSON, consulta Overview of JSON policy nella IAM User Guide.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguire l'operazione.

Policy basate sull'identità

Le policy basate sull'identità sono documenti di policy di autorizzazione JSON che alleghi a un'identità (utente, gruppo o ruolo). Queste politiche controllano quali azioni possono eseguire le identità, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consultare Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente di IAM.

Le politiche basate sull'identità possono essere politiche in linea (incorporate direttamente in una singola identità) o politiche gestite (politiche autonome collegate a più identità). Per scoprire come scegliere tra politiche gestite e politiche in linea, consulta Choose between managed policy e inline policy nella IAM User Guide.

Policy basate sulle risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio specificare un'entità principale.

Le policy basate su risorse sono policy inline che risiedono in tale servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse dai tipi di policy più comuni:

- Limiti delle autorizzazioni: imposta le autorizzazioni massime che una policy basata sull'identità può concedere a un'entità IAM. Per ulteriori informazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente di IAM .
- Politiche di controllo del servizio (SCPs): specifica le autorizzazioni massime per un'organizzazione o un'unità organizzativa in. AWS Organizations Per ulteriori informazioni, consultare Policy di controllo dei servizi nella Guida per l'utente di AWS Organizations .
- Politiche di controllo delle risorse (RCPs): imposta le autorizzazioni massime disponibili per le risorse nei tuoi account. Per ulteriori informazioni, consulta Politiche di controllo delle risorse (RCPs) nella Guida per l'AWS Organizations utente.
- Criteri di sessione: criteri avanzati passati come parametro durante la creazione di una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare Policy di sessione nella Guida per l'utente di IAM.

Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per sapere come si AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta la logica di valutazione delle policy nella IAM User Guide.

Come funziona Amazon Glacier con IAM

Prima di utilizzare IAM per gestire l'accesso ad Amazon Glacier, scopri quali funzionalità IAM sono disponibili per l'uso con Amazon Glacier.

Funzionalità IAM che puoi utilizzare con Amazon Glacier

Funzionalità IAM	Supporto per Amazon Glacier
Policy basate sull'identità	Sì
Policy basate su risorse	Sì
Operazioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione della policy (specifica del servizio)	Sì
ACLs	No
ABAC (tag nelle policy)	No
Credenziali temporanee	Sì
Autorizzazioni del principale	No
☻Ruoli di servizio	No
Ruoli collegati al servizio	No

Per avere una visione di alto livello di come Amazon Glacier AWS e altri servizi funzionano con la maggior parte delle funzionalità IAM AWS , consulta i servizi che funzionano con IAM nella IAM User Guide.

Politiche basate sull'identità per Amazon Glacier

Supporta le policy basate sull'identità: sì

Le policy basate sull'identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consultare Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente di IAM.

Con le policy basate sull'identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta Guida di riferimento agli elementi delle policy JSON IAM nella Guida per l'utente di IAM.

Esempi di policy basate sull'identità per Amazon Glacier

Per visualizzare esempi di politiche basate sull'identità di Amazon Glacier, consulta. Esempi di policy basate sull'identità per Amazon Glacier

Politiche basate sulle risorse all'interno di Amazon Glacier

Supporta le policy basate sulle risorse: sì

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli di IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. In una policy basata sulle risorse è obbligatorio specificare un'entità principale. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, è possibile specificare un intero account o entità IAM in un altro account come entità principale in una policy basata sulle risorse. Per maggiori informazioni, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente di IAM.

Il servizio Amazon Glacier supporta solo un tipo di policy basata sulle risorse chiamata vault policy, che è collegata a un vault. Questa policy definisce quali principali possono eseguire azioni sulla vault.

Le policy del vault di Amazon Glacier gestiscono le autorizzazioni nei seguenti modi:

- Gestione delle autorizzazioni utente in un account mediante un'unica policy di vault anziché di più policy utente singole.

- Gestione di autorizzazioni tra account come alternativa all'utilizzo di ruoli IAM.

Esempi di policy basate sulle risorse all'interno di Amazon Glacier

Per visualizzare esempi di politiche basate sulle risorse di Amazon Glacier, consulta. Esempi di policy basate sulle risorse per Amazon Glacier

Azioni politiche per Amazon Glacier

Supporta le operazioni di policy: si

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso a un criterio. Includere le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco delle azioni di Amazon Glacier, consulta Azioni definite da Amazon Glacier nel Service Authorization Reference.

Le azioni politiche in Amazon Glacier utilizzano il seguente prefisso prima dell'azione:

```
glacier
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [
            "glacier:CreateVault",
            "glacier:DescribeVault",
            "glacier:ListVaults"
         ]
```

È possibile specificare più azioni tramite caratteri jolly (*). Ad esempio, per specificare tutte le azioni che iniziano con la parola `Describe`, includi la seguente azione:

```
"Action": "glacier:GetVault*"
```

Per visualizzare esempi di politiche basate sull'identità di Amazon Glacier, consulta. Esempi di policy basate sull'identità per Amazon Glacier

Risorse relative alle policy per Amazon Glacier

Supporta le risorse relative alle policy: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Come best practice, specificare una risorsa utilizzando il suo nome della risorsa Amazon (ARN). Per le azioni che non supportano le autorizzazioni a livello di risorsa, usa un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

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

Per visualizzare un elenco dei tipi di risorse Amazon Glacier e ARNs relativi, consulta Risorse definite da Amazon Glacier nel Service Authorization Reference. Per informazioni su quali azioni è possibile specificare l'ARN di ciascuna risorsa, consulta Azioni definite da Amazon Glacier.

In Amazon Glacier, la risorsa principale è un vault. Amazon Glacier supporta le policy solo a livello di vault. In altre parole, in una policy IAM, il `Resource` valore specificato può essere un vault specifico o un insieme di vault in una regione specifica. AWS Amazon Glacier non supporta le autorizzazioni a livello di archivio.

Per tutte le azioni di Amazon Glacier`Resource`, specifica il vault su cui desideri concedere le autorizzazioni. A queste risorse sono associati Amazon Resource Names (ARNs) univoci, come mostrato nella tabella seguente, e puoi utilizzare un carattere jolly (*) nell'ARN per abbinare i nomi dei vault che iniziano con lo stesso prefisso.

Amazon Glacier fornisce una serie di operazioni per lavorare con le risorse Amazon Glacier. Per informazioni sulle operazioni disponibili, consulta Riferimento API per Amazon Glacier.

Alcune azioni dell'API Amazon Glacier supportano più risorse. Ad esempio, `glacier:AddTagsToVault` accede a examplevault1 e examplevault2, quindi un principale deve

disporre delle autorizzazioni per accedere a entrambe le risorse. Per specificare più risorse in una singola istruzione, separale ARNs con virgole.

```
"Resource": [
""arn:aws:glacier:us-west-2:123456789012:vaults/examplevault1",",
""arn:aws:glacier:us-west-2:123456789012:vaults/examplevault2","
]
```

Chiavi relative alle condizioni delle politiche per Amazon Glacier

Supporta le chiavi di condizione delle policy specifiche del servizio: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'`Condition`elemento specifica quando le istruzioni vengono eseguite in base a criteri definiti. È possibile compilare espressioni condizionali che utilizzano operatori di condizione, ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di contesto delle condizioni AWS globali nella Guida per l'utente IAM.

Per visualizzare un elenco di chiavi di condizione di Amazon Glacier, consulta Condition keys for Amazon Glacier nel Service Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, consulta Azioni definite da Amazon Glacier.

Per esempi di utilizzo delle chiavi di condizione specifiche di Glacier, consulta Policy di Vault Lock.

ACLs in Amazon Glacier

Supporti ACLs: no

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLs sono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

ABAC con Amazon Glacier

Supporta ABAC (tag nelle policy): No

Il controllo degli accessi basato sugli attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base ad attributi chiamati tag. Puoi allegare tag a entità e AWS risorse IAM, quindi progettare politiche ABAC per consentire operazioni quando il tag del principale corrisponde al tag sulla risorsa.

Per controllare l'accesso basato su tag, fornire informazioni sui tag nell'elemento condizione di una policy utilizzando le chiavi di condizione `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Sì. Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per maggiori informazioni su ABAC, consultare Definizione delle autorizzazioni con autorizzazione ABAC nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consultare Utilizzo del controllo degli accessi basato su attributi (ABAC) nella Guida per l'utente di IAM.

Utilizzo di credenziali temporanee con Amazon Glacier

Supporta le credenziali temporanee: sì

Le credenziali temporanee forniscono l'accesso a breve termine alle AWS risorse e vengono create automaticamente quando si utilizza la federazione o si cambia ruolo. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta Credenziali di sicurezza temporanee in IAM e compatibili Servizi AWS con IAM nella IAM User Guide.

Autorizzazioni principali multiservizio per Amazon Glacier

Supporta l'inoltro delle sessioni di accesso (FAS): no

Le sessioni di accesso inoltrato (FAS) utilizzano le autorizzazioni del principale che chiama an Servizio AWS, combinate con la richiesta di effettuare richieste ai servizi Servizio AWS downstream. Per i dettagli delle policy relative alle richieste FAS, consultare Forward access sessions.

Ruoli di servizio per Amazon Glacier

Supporta i ruoli di servizio: no

Un ruolo di servizio è un ruolo IAM che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per maggiori informazioni, consultare la sezione Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS nella Guida per l'utente di IAM.

> **Warning**
>
> La modifica delle autorizzazioni per un ruolo di servizio potrebbe interrompere la funzionalità di Amazon Glacier. Modifica i ruoli di servizio solo quando Amazon Glacier fornisce indicazioni in tal senso.

Ruoli collegati ai servizi per Amazon Glacier

Supporta i ruoli collegati ai servizi: no

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un. Servizio AWS Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati al servizio, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta Servizi AWS supportati da IAM. Trova un servizio nella tabella che include un `Yes` nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Esempi di policy basate sull'identità per Amazon Glacier

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse Amazon Glacier. Per concedere agli utenti l'autorizzazione a eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consultare Creazione di policy IAM (console) nella Guida per l'utente di IAM.

Per informazioni dettagliate sulle azioni e sui tipi di risorse definiti da Amazon Glacier, incluso il formato di per ogni tipo di ARNs risorsa, consulta Azioni, risorse e chiavi di condizione per Amazon Glacier nel Service Authorization Reference.

Di seguito è riportato un esempio di policy che concede le autorizzazioni per tre azioni relative al vault Amazon Glacier `glacier:CreateVault` (e`glacier:ListVaults`) su una risorsa`glacier:DescribeVault`, utilizzando l'Amazon Resource Name (ARN) che identifica tutti i vault nella regione. `us-west-2` AWS ARNs AWS identifica in modo univoco le risorse. Per ulteriori informazioni sull' ARNs utilizzo con Amazon Glacier, consulta. Risorse relative alle policy per 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 policy concede autorizzazioni per creare, elencare e ottenere descrizioni di vault nella regione `us-west-2`. Il carattere jolly (*) alla fine dell'ARN significa che questa istruzione può corrispondere a qualsiasi nome di vault.

> **Important**
>
> Quando concedi autorizzazioni per creare un vault utilizzando l'operazione `glacier:CreateVault`, devi specificare un carattere jolly (*) in quanto non puoi conoscere il nome di vault fino a che non crei il vault.

Argomenti

- Best practice per le policy

- Utilizzo della console Amazon Glacier
- Consentire agli utenti di visualizzare le loro autorizzazioni
- Esempi di policy gestite dal cliente

Best practice per le policy

Le policy basate sull'identità determinano se qualcuno può creare, accedere o eliminare risorse Amazon Glacier nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l' Account AWS. Quando si creano o modificano policy basate sull'identità, seguire queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche gestite che concedono le autorizzazioni per molti casi d'uso comuni.AWS Sono disponibili nel tuo. Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti specifiche per i tuoi casi d'uso. Per maggiori informazioni, consultare Policy gestite da AWS o Policy gestite da AWS per le funzioni dei processi nella Guida per l'utente di IAM.
- Applicazione delle autorizzazioni con privilegio minimo - Quando si impostano le autorizzazioni con le policy IAM, concedere solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegio minimo. Per maggiori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consultare Policy e autorizzazioni in IAM nella Guida per l'utente di IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso - Per limitare l'accesso a operazioni e risorse è possibile aggiungere una condizione alle policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio CloudFormation. Per maggiori informazioni, consultare la sezione Elementi delle policy JSON di IAM: condizione nella Guida per l'utente di IAM.
- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali - IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per maggiori informazioni, consultare Convalida delle policy per il Sistema di analisi degli accessi IAM nella Guida per l'utente di IAM.

- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungere le condizioni MFA alle policy. Per maggiori informazioni, consultare Protezione dell'accesso API con MFA nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consultare Best practice di sicurezza in IAM nella Guida per l'utente di IAM.

Utilizzo della console Amazon Glacier

Per accedere alla console Amazon Glacier, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle risorse Amazon Glacier presenti nel tuo. Account AWS Se si cra una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso o l' AWS CLI API. AWS Al contrario, è opportuno concedere l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

La console Amazon Glacier fornisce un ambiente integrato per creare e gestire vault Amazon Glacier. Come minimo, alle identità IAM che crei devono essere concesse le autorizzazioni per l'`glacier:ListVaults`azione di visualizzazione della console Amazon Glacier, come mostrato nell'esempio seguente.

JSON

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

```
        }
```

AWS affronta molti casi d'uso comuni fornendo politiche IAM autonome create e amministrate da. AWS Le policy gestite concedono le autorizzazioni necessarie per i casi di utilizzo comune in modo da non dover cercare quali sono le autorizzazioni richieste. Per ulteriori informazioni, consulta Policy gestite da AWS nella Guida per l'utente di IAM.

Le seguenti politiche AWS gestite, che puoi allegare agli utenti del tuo account, sono specifiche di Amazon Glacier:

- AmazonGlacierReadOnlyAccess— Concede l'accesso in sola lettura ad Amazon Console di gestione AWS Glacier tramite.
- AmazonGlacierFullAccess— Garantisce l'accesso completo ad Amazon Console di gestione AWS Glacier tramite.

Puoi anche creare policy IAM personalizzate per consentire le autorizzazioni per le azioni e le risorse dell'API Amazon Glacier. Puoi collegare queste policy personalizzate ai ruoli IAM personalizzati che crei per i tuoi vault Amazon Glacier.

Entrambe le politiche gestite di Amazon AWS Glacier discusse nella sezione successiva concedono autorizzazioni per. `glacier:ListVaults`

Per maggiori informazioni, consultare Aggiunta di autorizzazioni a un utente nella Guida per l'utente di IAM.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono cpllegate alla relativa identità utente. Questa policy include le autorizzazioni per completare questa azione sulla console o utilizzando l'API or a livello di codice. AWS CLI AWS

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

```
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Esempi di policy gestite dal cliente

In questa sezione, puoi trovare esempi di politiche utente che concedono autorizzazioni per varie azioni di Amazon Glacier. Queste policy funzionano quando utilizzi l'API REST di Amazon Glacier, Amazon, la console di gestione SDKs Amazon Glacier o, se applicabile, AWS CLI la console di gestione Amazon Glacier.

> **Note**
>
> Tutti gli esempi utilizzano la regione Stati Uniti occidentali (Oregon) () e contengono account fittizius-west-2. IDs

Esempi

- Esempio 1: consentire a un utente di scaricare archivi da un vault
- Esempio 2: consentire a un utente di creare un vault e di configurare le notifiche
- Esempio 3: consentire a un utente di caricare archivi in un determinato vault
- Esempio 4: concedere autorizzazioni complete a un utente su un determinato vault

Esempio 1: consentire a un utente di scaricare archivi da un vault

Per scaricare un archivio, devi dapprima avviare un processo per recuperare l'archivio. Quando il processo è completato, puoi scaricare i dati. L'esempio di policy che segue concede autorizzazioni per l'operazione di avvio di un processo `glacier:InitiateJob` (che consente all'utente di recuperare un archivio o un inventario vault dal vault) e autorizzazioni per l'operazione di download di dati recuperati `glacier:GetJobOutput`. La policy concede inoltre autorizzazioni per l'esecuzione dell'operazione `glacier:DescribeJob` di modo che l'utente possa ottenere lo stato del processo. Per ulteriori informazioni, consulta Initiate Job (POST jobs).

La policy concede tali autorizzazioni su un vault denominato `examplevault`. Puoi ottenere l'ARN del vault dalla console Amazon Glacier o a livello di codice chiamando le azioni o le API. Describe Vault (GET vault) List Vaults (GET vaults)

Esempio 2: consentire a un utente di creare un vault e di configurare le notifiche

L'esempio di policy seguente concede autorizzazioni per creare una vault nella regione us-west-2 come specificato nell'elemento `Resource` e per configurare le notifiche. Per ulteriori informazioni sull'utilizzo delle notifiche, consulta Configurazione delle notifiche Vault in Amazon Glacier. La policy concede anche le autorizzazioni per elencare i vault nella regione e ottenere una descrizione specifica del vault. AWS

> **Important**
>
> Quando concedi autorizzazioni per creare un vault utilizzando l'operazione `glacier:CreateVault`, devi specificare un carattere jolly (*) nel valore `Resource` in quanto non puoi conoscere il nome di vault fino a che non crei il vault.

Esempio 3: consentire a un utente di caricare archivi in un determinato vault

L'esempio di policy seguente concede autorizzazioni per caricare archivi in una determinata vault nella regione us-west-2. Queste autorizzazioni consentono a un utente di caricare tutto l'archivio in

una sola volta utilizzando l'operazione API Upload Archive (POST archive) oppure in parti utilizzando l'operazione API Initiate Multipart Upload (POST multipart-uploads).

Esempio 4: concedere autorizzazioni complete a un utente su un determinato vault

La seguente policy di esempio concede le autorizzazioni per tutte le azioni di Amazon Glacier su un vault denominato. `examplevault`

Esempi di policy basate sulle risorse per Amazon Glacier

Un vault Amazon Glacier può avere una policy di accesso al vault e una policy Vault Lock ad esso associata. Una policy di accesso al vault di Amazon Glacier è una policy basata sulle risorse che puoi utilizzare per gestire le autorizzazioni al tuo vault. Una policy di Vault Lock è una policy di accesso a vault che può essere bloccata. Dopo il blocco di una policy di Vault Lock, la policy non può essere modificata. Puoi utilizzare una policy di Vault Lock per applicare controlli di conformità.

Argomenti

- Policy di accesso delle vault
- Policy di Vault Lock

Policy di accesso delle vault

Una policy di accesso al vault di Amazon Glacier è una policy basata sulle risorse che puoi utilizzare per gestire le autorizzazioni al tuo vault.

Puoi creare una policy di accesso a vault per ogni vault allo scopo di gestire le autorizzazioni, nonché modificare le autorizzazioni in una policy di accesso a vault in qualsiasi momento. Amazon Glacier supporta anche una policy Vault Lock su ogni vault che, dopo averlo bloccato, non può essere modificata. Per ulteriori informazioni sull'utilizzo delle policy di Vault Lock, consulta Policy di Vault Lock.

Esempi

- Esempio 1: concedere autorizzazioni per più account per azioni specifiche di Amazon Glacier
- Esempio 2: concessione di autorizzazioni tra account per operazioni di cancellazione MFA

Esempio 1: concedere autorizzazioni per più account per azioni specifiche di Amazon Glacier

La seguente politica di esempio concede autorizzazioni multiaccount a due Account AWS per una serie di operazioni Amazon Glacier su un vault denominato. `examplevault`

> **Note**
>
> Tutti i costi associati al vault sono fatturati all'account proprietario del vault. Tutte i costi relativi a richieste, trasferimenti di dati e recuperi generati da account esterni autorizzati sono fatturati all'account proprietario del vault.

Esempio 2: concessione di autorizzazioni tra account per operazioni di cancellazione MFA

Puoi utilizzare l'autenticazione a più fattori (MFA) per proteggere le tue risorse Amazon Glacier. Per fornire un ulteriore livello di protezione, MFA richiede agli utenti di dimostrare di possedere materialmente un dispositivo MFA fornendo un codice MFA valido. Per ulteriori informazioni sulla configurazione dell'accesso MFA, consulta la pagina Configurazione dell'accesso a API protetto con MFA nella Guida per l'utente IAM.

La politica di esempio concede un'autorizzazione Account AWS con credenziali temporanee per eliminare gli archivi da un archivio denominato examplevault, a condizione che la richiesta sia autenticata con un dispositivo MFA. La policy utilizza la chiave di condizione `aws:MultiFactorAuthPresent` per specificare questo requisito aggiuntivo. Per ulteriori informazioni, consulta la pagina Chiavi disponibili per le condizioni nella Guida per l'utente IAM.

Policy di Vault Lock

Un vault Amazon Glacier (Amazon Glacier) può avere una policy di accesso al vault basata sulle risorse e una policy Vault Lock associata. Una policy di Vault Lock è una policy di accesso a vault che può essere bloccata. L'utilizzo di una policy di blocco della vault può facilitare l'implementazione di requisiti normativi e di conformità. Amazon Glacier fornisce una serie di operazioni API per gestire le policy di Vault Lock, vedi. Bloccare un vault utilizzando l'API Amazon Glacier

Per illustrare la funzione delle policy di blocco della vault, supponiamo che gli archivi vengano conservati per un anno prima di poterli eliminare. Per implementare questo requisito, puoi creare una policy di Vault Lock che nega agli utenti l'autorizzazione di eliminare un archivio prima di un anno. Puoi testare questa policy prima di bloccarla, in quanto, una volta bloccata, non può essere modificata. Per ulteriori informazioni sul processo di blocco, consulta Policy di Vault Lock. Se desideri

gestire altre autorizzazioni utente che possono essere modificate, puoi utilizzare la policy di accesso a vault (vedi Policy di accesso delle vault).

Puoi utilizzare l'API Amazon Glacier, Amazon o la console SDKs Amazon AWS CLI Glacier per creare e gestire le policy Vault Lock. Per un elenco delle azioni di Amazon Glacier consentite per le policy basate sulle risorse del vault, consulta. Riferimento alle autorizzazioni API

Esempi

- Esempio 1: negare autorizzazioni di eliminazione per archivi di meno di 365 giorni
- Esempio 2: negare autorizzazioni di eliminazione in base a un tag

Esempio 1: negare autorizzazioni di eliminazione per archivi di meno di 365 giorni

Supponi che un requisito normativo ti obblighi a conservare archivi per un anno prima di poterli eliminare. Puoi soddisfare tale requisito implementando la policy di Vault Lock illustrata di seguito. La policy nega l'operazione `glacier:DeleteArchive` sul vault examplevault se l'archivio che si tenta di eliminare ha meno di un anno. La policy utilizza la chiave di condizione specifica di Amazon Glacier `ArchiveAgeInDays` per applicare il requisito di conservazione di un anno.

Esempio 2: negare autorizzazioni di eliminazione in base a un tag

Supponi che sia implementata una regola di conservazione basata sul tempo secondo la quale un archivio può essere eliminato se ha meno di un anno. Allo stesso tempo, supponi di dover applicare un blocco a fini giudiziari agli archivi per evitarne l'eliminazione o la modifica per una durata indeterminata durante un'accertamento legale. In tal caso, il blocco a fini giudiziari prevale sulla regola di conservazione basata sul tempo specificata nella policy di Vault Lock.

Per implementare queste due regole, il seguente esempio di policy comporta due istruzioni:

- La prima istruzione nega le autorizzazioni di eliminazione a tutti gli utenti, bloccando il vault. Questo blocco viene eseguito utilizzando il tag `LegalHold`.
- La seconda istruzione concede le autorizzazioni di eliminazione quando l'archivio ha meno di 365 giorni. Tuttavia, anche quando gli archivi hanno meno di 365 giorni, non possono essere eliminati se la condizione della prima istruzione viene soddisfatta.

Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Glacier

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con Amazon Glacier e IAM.

Argomenti

- Non sono autorizzato a eseguire un'azione in Amazon Glacier
- Non sono autorizzato a eseguire iam: PassRole
- Desidero consentire a persone esterne a me di accedere Account AWS alle mie risorse Amazon Glacier

Non sono autorizzato a eseguire un'azione in Amazon Glacier

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa *`my-example-widget`* fittizia ma non dispone di autorizzazioni `glacier:`*`GetWidget`* fittizie.

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

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa *`my-example-widget`* utilizzando l'azione `glacier:`*`GetWidget`*.

Se hai bisogno di assistenza, contatta il tuo AWS amministratore. L'amministratore è la persona che ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam:PassRole`azione, le tue politiche devono essere aggiornate per consentirti di trasferire un ruolo ad Amazon Glacier.

Alcuni Servizi AWS consentono di trasferire un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

Il seguente errore di esempio si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in Amazon Glacier. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere il ruolo al servizio.

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

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ha fornito le credenziali di accesso.

Desidero consentire a persone esterne a me di accedere Account AWS alle mie risorse Amazon Glacier

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACLs), puoi utilizzare tali politiche per consentire alle persone di accedere alle tue risorse.

Per maggiori informazioni, consultare gli argomenti seguenti:

- Per sapere se Amazon Glacier supporta queste funzionalità, consulta. Come funziona Amazon Glacier con IAM
- Per sapere come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta Fornire l'accesso a soggetti Account AWS di proprietà di terze parti nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consultare Fornire l'accesso a utenti autenticati esternamente (Federazione delle identità) nella Guida per l'utente di IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente di IAM.

Riferimento alle autorizzazioni API

Quando configuri Come funziona Amazon Glacier con IAM e scrivi una policy di autorizzazione che puoi collegare a un'identità IAM (policy basate su identità) o a una risorsa (policy basate su risorse), puoi utilizzare la tabella seguente come riferimento. L' include ogni operazione dell'API Amazon Glacier, le azioni corrispondenti per le quali è possibile concedere le autorizzazioni per eseguire l'azione e la risorsa per la quale è possibile concedere AWS le autorizzazioni.

Puoi specificare le operazioni nell'elemento `Action` della policy e il valore della risorsa nell'elemento `Resource` della policy. Inoltre, puoi utilizzare l'elemento `Condition` del linguaggio della policy IAM per specificare quando una policy deve essere applicata.

Per specificare un'operazione, utilizza il prefisso `glacier:` seguito dal nome dell'operazione API (ad esempio, `glacier:CreateVault`). Per la maggior parte delle azioni di Amazon Glacier`Resource`, è il vault su cui desideri concedere le autorizzazioni. Puoi specificare un vault come valore `Resource` utilizzando l'ARN del vault. Per esprimere le condizioni è necessario utilizzare chiavi di condizione predefinite. Per ulteriori informazioni, consulta Politiche basate sulle risorse all'interno di Amazon Glacier.

La tabella seguente elenca le operazioni utilizzabili con le policy basate su identità e le policy basate su risorse.

> **Note**
>
> Alcune operazioni possono essere utilizzate solo con le policy basate su identità. Queste azioni sono contrassegnate da un asterisco (*) accanto al nome dell'operazione API nella prima colonna.

API Amazon Glacier e autorizzazioni richieste per le azioni

Abort Multipart Upload (DELETE uploadID)

Autorizzazioni necessarie (operazioni API): `glacier:AbortMultipartUpload`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Abort Vault Lock (DELETE lock-policy)

Autorizzazioni necessarie (operazioni API): `glacier:AbortVaultLock`

Risorse:

Chiavi delle condizioni di Amazon Glacier:

Add Tags To Vault (POST tags add)

Autorizzazioni necessarie (operazioni API):`glacier:AddTagsToVault`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Complete Multipart Upload (POST uploadID)

Autorizzazioni necessarie (operazioni API):`glacier:CompleteMultipartUpload`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Complete Vault Lock (POST lockId)

Autorizzazioni necessarie (operazioni API):`glacier:CompleteVaultLock`

Risorse:

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Create Vault (PUT vault) *

Autorizzazioni necessarie (operazioni API):`glacier:CreateVault`

Risorse:

Chiavi delle condizioni di Amazon Glacier:

Delete Archive (DELETE archive)

Autorizzazioni necessarie (operazioni API):`glacier:DeleteArchive`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:, `glacier:ArchiveAgeInDays` `glacier:ResourceTag/`*`TagKey`*

Delete Vault (DELETE vault)

Autorizzazioni necessarie (operazioni API):`glacier:DeleteVault`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Delete Vault Access Policy (DELETE access-policy)

Autorizzazioni necessarie (operazioni API):`glacier:DeleteVaultAccessPolicy`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Delete Vault Notifications (DELETE notification-configuration)

Autorizzazioni necessarie (operazioni API):`glacier:DeleteVaultNotifications`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Describe Job (GET JobID)

Autorizzazioni necessarie (operazioni API):`glacier:DescribeJob`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Describe Vault (GET vault)

Autorizzazioni necessarie (operazioni API):`glacier:DescribeVault`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Get Data Retrieval Policy (GET policy) *

Autorizzazioni necessarie (operazioni API):`glacier:GetDataRetrievalPolicy`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:policies/retrieval-limit-policy`

Chiavi delle condizioni di Amazon Glacier:

Get Job Output (GET output)

Autorizzazioni necessarie (operazioni API):`glacier:GetJobOutput`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Get Vault Access Policy (GET access-policy)

Autorizzazioni necessarie (operazioni API):`glacier:GetVaultAccessPolicy`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Get Vault Lock (GET lock-policy)

Autorizzazioni necessarie (operazioni API):`glacier:GetVaultLock`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Get Vault Notifications (GET notification-configuration)

Autorizzazioni necessarie (operazioni API):`glacier:GetVaultNotifications`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

Initiate Job (POST jobs)

Autorizzazioni necessarie (operazioni API):`glacier:InitiateJob`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier:, `glacier:ArchiveAgeInDays` `glacier:ResourceTag/`*`TagKey`*

Initiate Multipart Upload (POST multipart-uploads)

Autorizzazioni necessarie (operazioni API):`glacier:InitiateMultipartUpload`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Initiate Vault Lock (POST lock-policy)

Autorizzazioni necessarie (operazioni API):`glacier:InitiateVaultLock`

Risorse:

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

List Jobs (GET jobs)

Autorizzazioni necessarie (operazioni API):`glacier:ListJobs`

Risorse: `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

List Multipart Uploads (GET multipart-uploads)

Autorizzazioni necessarie (operazioni API):`glacier:ListMultipartUploads`

Risorse: `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

List Parts (GET uploadID)

Autorizzazioni necessarie (operazioni API):`glacier:ListParts`

Risorse: `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

List Tags For Vault (GET tags)

Autorizzazioni necessarie (operazioni API):`glacier:ListTagsForVault`

Risorse: `arn:aws:glacier:region:account-id:vaults/vault-name`, `arn:aws:glacier:region:account-id:vaults/example*`, `arn:aws:glacier:region:account-id:vaults/*`

Chiavi delle condizioni di Amazon Glacier:

List Vaults (GET vaults)

Autorizzazioni necessarie (operazioni API):`glacier:ListVaults`

Risorse:

Chiavi delle condizioni di Amazon Glacier:

Remove Tags From Vault (POST tags remove)

Autorizzazioni necessarie (operazioni API):`glacier:RemoveTagsFromVault`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Set Data Retrieval Policy (PUT policy) *

Autorizzazioni necessarie (operazioni API):`glacier:SetDataRetrievalPolicy`

Risorse:`arn:aws:glacier:`*`region`*`:`*`account-id`*`:policies/retrieval-limit-policy`

Chiavi delle condizioni di Amazon Glacier:

Set Vault Access Policy (PUT access-policy)

Autorizzazioni necessarie (operazioni API):`glacier:SetVaultAccessPolicy`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Set Vault Notification Configuration (PUT notification-configuration)

Autorizzazioni necessarie (operazioni API):`glacier:SetVaultNotifications`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Upload Archive (POST archive)

Autorizzazioni necessarie (operazioni API):`glacier:UploadArchive`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Upload Part (PUT uploadID)

Autorizzazioni necessarie (operazioni API):`glacier:UploadMultipartPart`

Risorse: `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/vault-name`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/example*`, `arn:aws:glacier:`*`region`*`:`*`account-id`*`:vaults/*`

Chiavi delle condizioni di Amazon Glacier: `glacier:ResourceTag/`*`TagKey`*

Registrazione e monitoraggio in Amazon Glacier

Il monitoraggio è una parte importante per mantenere l'affidabilità, la disponibilità e le prestazioni di Amazon Glacier (Amazon Glacier) e delle tue soluzioni. AWS È necessario raccogliere i dati di monitoraggio da tutte le parti della AWS soluzione in modo da poter identificare ed eseguire più facilmente il debug dell'origine di un eventuale errore. AWS fornisce i seguenti strumenti per monitorare le tue risorse Amazon Glacier e rispondere a potenziali incidenti:

CloudWatch Allarmi Amazon

Quando usi Amazon Glacier tramite Amazon S3, puoi utilizzare gli allarmi CloudWatch Amazon per monitorare una singola metrica per un periodo di tempo specificato. Se la metrica supera una determinata soglia, viene inviata una notifica a un argomento o una policy di Amazon SNS. AWS Auto Scaling CloudWatch gli allarmi non richiamano azioni perché si trovano in uno stato particolare. È necessario invece cambiare lo stato e mantenerlo per un numero di periodi specificato. Per ulteriori informazioni, consulta Monitoring Metrics with Amazon CloudWatch.

AWS CloudTrail Registri

CloudTrail fornisce un registro delle azioni intraprese da un utente, un ruolo o un AWS servizio in Amazon Glacier. CloudTrail acquisisce tutte le chiamate API per Amazon Glacier come eventi, incluse le chiamate dalla console Amazon Glacier e le chiamate di codice ad Amazon Glacier. APIs Per ulteriori informazioni, consulta Registrazione delle chiamate API Amazon Glacier con AWS CloudTrail.

AWS Trusted Advisor

Trusted Advisor attinge alle best practice apprese servendo centinaia di migliaia di clienti. AWS Trusted Advisor ispeziona l' AWS ambiente e quindi formula raccomandazioni quando esistono opportunità per risparmiare denaro, migliorare la disponibilità e le prestazioni del sistema o contribuire a colmare le lacune di sicurezza. Tutti i AWS clienti hanno accesso a cinque Trusted Advisor controlli. I clienti con un piano di supporto Business o Enterprise possono visualizzare tutti i Trusted Advisor controlli.

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente di Supporto .

Convalida della conformità per Amazon Glacier

La sicurezza e la conformità di Amazon Glacier (Amazon Glacier) vengono valutate da revisori di terze parti nell'ambito di diversi programmi di conformità, tra cui: AWS

- System and Organization Controls (SOC)
- Payment Card Industry Data Security Standard (PCI DSS)
- Federal Risk and Authorization Management Program (FedRAMP)
- Health Insurance Portability and Accountability Act (HIPAA)

AWS fornisce un elenco frequentemente aggiornato di servizi nell'ambito di specifici programmi di conformità nella pagina AWS Services in Scope by Compliance Program AWS .

I report di audit di terze parti possono essere scaricati utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricare i report in AWS Artifact nella AWS Artifact Guida per l'utente.

Per ulteriori informazioni sui programmi di AWS conformità, consulta Programmi di AWS conformità.

La tua responsabilità di conformità quando usi Amazon Glacier è determinata dalla sensibilità dei tuoi dati, dagli obiettivi di conformità dell'organizzazione e dalle leggi e dai regolamenti applicabili. Se l'uso di Amazon Glacier è soggetto alla conformità a standard come HIPAA, PCI o FedRAMP, fornisce risorse per aiutarti a: AWS

- Serratura Amazon Glacier Vaultconsente di implementare e applicare facilmente i controlli di conformità per singoli vault Amazon Glacier con una politica di blocco del vault. Puoi specificare controlli in formato WORM (Write Once Read Many) in una policy di Vault Lock e bloccare la policy per impedire modifiche future. Dopo il blocco della policy, non è possibile modificarla. Le politiche di blocco dei vault possono aiutarti a rispettare i quadri normativi come -4 e HIPAA. SEC17a
- Le guide rapide su sicurezza e conformità illustrano le considerazioni architettoniche e i passaggi da seguire per implementare ambienti di base incentrati sulla sicurezza e la conformità su. AWS
- Architecting for HIPAA Security and Compliance descrive in che modo le aziende utilizzano per aiutarle a soddisfare i requisiti HIPAA. AWS
- AWS Well-Architected Tool AWS (WA Tool) è un servizio nel cloud che fornisce un processo coerente per rivedere e misurare l'architettura AWS utilizzando le migliori pratiche. Lo strumento AWS WA fornisce consigli per rendere i carichi di lavoro più affidabili, sicuri, efficienti ed economici.
- AWS Le risorse per la conformità forniscono diverse cartelle di lavoro e guide che potrebbero essere applicabili al settore e alla località in cui operate.

- AWS Config: consente di valutare il livello di conformità delle configurazioni delle risorse con practice interne, linee guida e regolamenti industriali.
- AWS Security Hub ti offre una visione completa del tuo stato di sicurezza interno AWS e ti aiuta a verificare la tua conformità agli standard e alle migliori pratiche del settore della sicurezza.

Resilienza in Amazon Glacier

L'infrastruttura AWS globale è costruita attorno a regioni e zone di disponibilità. AWS forniscono più zone di disponibilità fisicamente separate e isolate che sono connesse tramite reti altamente ridondanti, a bassa latenza e throughput elevato. Queste zone di disponibilità offrono un modo efficace per progettare e gestire le applicazioni e i database. Sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali. Amazon Glacier archivia i dati in modo ridondante su più dispositivi in almeno tre zone di disponibilità. Per aumentare la durabilità, Amazon Glacier archivia i dati in modo sincrono su AZs più file prima di confermare il corretto caricamento.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta Global Infrastructure.AWS

Sicurezza dell'infrastruttura in Amazon Glacier

In quanto servizio gestito, Amazon Glacier (Amazon Glacier) è protetto dalle AWS procedure di sicurezza di rete globali descritte in Amazon Web Services: panoramica dei processi di sicurezza.

L'accesso ad Amazon Glacier tramite la rete avviene tramite published. AWS APIs I client devono supportare Transport Layer Security (TLS) 1.2. È consigliabile TLS 1.3 o versioni successive. I client devono inoltre supportare le suite di cifratura con PFS (Perfect Forward Secrecy), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Diffie-Hellman Ephemeral (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità. Inoltre, è necessario firmare le richieste utilizzando un ID chiave di accesso e la chiave di accesso segreta associati a un principale IAM, oppure è possibile utilizzare l'AWS Security Token Service (AWS STS) per generare le credenziali di sicurezza temporanee per firmare le richieste.

Endpoint VPC

Un endpoint di cloud privato virtuale (VPC) ti consente di connettere privatamente il tuo VPC ai servizi supportati AWS e ai servizi endpoint VPC basati su AWS PrivateLink senza richiedere un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect Sebbene Amazon Glacier non supporti direttamente gli endpoint VPC, puoi sfruttare gli endpoint VPC di Amazon S3 se accedi ad Amazon Glacier come livello di storage integrato con Amazon S3.

Per ulteriori informazioni sulla configurazione del ciclo di vita di Amazon S3 e sulla transizione degli oggetti alla classe di storage Amazon Glacier, consulta Object Lifecycle Management and Transitioning Objects nella Amazon Simple Storage Service User Guide. Per ulteriori informazioni sugli endpoint VPC, consulta Endpoint VPC nella Guida per l'utente di Amazon VPC.

Politiche di recupero dei dati di Amazon Glacier

Con le policy di recupero dei dati di Amazon Glacier, puoi impostare facilmente quote di recupero dei dati e gestire le attività di recupero dei dati in ogni area. Account AWS Regione AWSPer ulteriori informazioni sui costi di recupero dei dati di Amazon Glacier, consulta i prezzi di Amazon Glacier.

> **Important**
>
> Una politica di recupero dei dati si applica solo ai recuperi Standard e gestisce le richieste di recupero effettuate direttamente su Amazon Glacier.
> Per ulteriori informazioni sulle classi di storage di Amazon Glacier, consulta Storage classes for archiving objects e Transitioning objects nella Amazon Simple Storage Service User Guide.

Argomenti

- Scelta di una policy di recupero dei dati di Amazon Glacier
- Utilizzo della console Amazon Glacier per configurare una policy di recupero dei dati
- Utilizzo dell'API Amazon Glacier per configurare una politica di recupero dei dati

Scelta di una policy di recupero dei dati di Amazon Glacier

Puoi scegliere tra tre tipi di policy di recupero dei dati di Amazon Glacier: No Retrieval Limit, Free Tier Only e Max Retrieval Rate.

Nessun limite di recupero è la policy sul recupero dati predefinita utilizzata per le operazioni di recupero. Se si utilizza la policy Nessun limite di recupero, non è impostata alcuna quota di recupero valida e vengono accettate tutte le richieste valide di recupero dati.

Utilizzando una politica Free Tier Only, puoi mantenere i dati recuperati entro l'indennità giornaliera del piano AWS gratuito e non incorrere in alcun costo per il recupero dei dati. Se desideri recuperare più dati di quelli previsti dal piano AWS gratuito, puoi utilizzare la politica Max Retrieval Rate per impostare una quota relativa alla tariffa di recupero. bytes-per-hour La politica Max Retrieval Rate garantisce che la frequenza massima di recupero per tutti i lavori di recupero del tuo account non superi la quota che hai impostato. Regione AWS bytes-per-hour

Con le policy Solo piano gratuito e Velocità di recupero massima, le richieste di recupero dati che superano la quota di recupero specificata non sono accettate. Se utilizzi una politica dedicata esclusivamente al piano gratuito, Amazon Glacier rifiuta in modo sincrono le richieste di recupero che superano la quota del piano gratuito. AWS Se utilizzi una politica di Max Retrieval Rate, Amazon Glacier rifiuta le richieste di recupero che fanno sì che il tasso di recupero di picco dei lavori in corso superi la quota stabilita dalla policy. bytes-per-hour Queste policy ti consentono di semplificare la gestione dei costi inerenti al recupero di dati.

Di seguito sono riportate alcune considerazioni utili sulle policy di recupero dati:

- Le impostazioni delle politiche di recupero dei dati non modificano il periodo da 3 a 5 ore necessario per recuperare i dati da Amazon Glacier utilizzando i recuperi Standard.
- L'impostazione di una nuova policy di recupero dati non influisce sui processi di recupero accettati in precedenza che sono già in corso.
- Se una richiesta di processo di recupero viene respinta a causa di una policy sul recupero dati, non ti sarà addebitato alcun costo per il processo o la richiesta.
- Puoi impostare una politica di recupero dei dati per ciascuno Regione AWS, che regolerà tutte le attività di recupero dei dati nel tuo account. Regione AWS Una politica di recupero dei dati è specifica per un particolare Regione AWS dato che i costi di recupero dei dati variano da un paese all'altro. Regioni AWS Per ulteriori informazioni, consulta i prezzi di Amazon Glacier.

Policy Free Tier Only

Puoi impostare una politica di recupero dei dati su Free Tier Only per garantire che i recuperi rimangano sempre entro il limite del piano gratuito, in modo da non incorrere in AWS costi per il recupero dei dati. Se una richiesta di recupero viene respinta, ricevi un messaggio di errore indicante che la richiesta è stata rifiutata dalla policy sul recupero dati corrente.

La policy sul recupero dati viene impostata su Solo piano gratuito su una base per regione. Una volta impostata la policy, non puoi recuperare in un giorno una quantità di dati superiore alla quota giornaliera proporzionale di recupero del Piano gratuito AWS per quella Regione AWS. Inoltre, non sono previsti costi per il recupero dati.

Puoi passare alla policy Solo piano gratuito dopo aver sostenuto costi di recupero dati per un mese. La policy Solo piano gratuito verrà applicata alle nuove richieste di recupero, ma non avrà alcun effetto sulle richieste precedenti. I costi sostenuti in precedenza ti verranno fatturati.

Policy Max Retrieval Rate

Puoi impostare la tua politica di recupero dei dati su Max Retrieval Rate per controllare la frequenza di picco di recupero specificando una quota di recupero dati che abbia un massimo. bytes-per-hour Quando si imposta la politica di recupero dei dati su Max Retrieval Rate, una nuova richiesta di recupero viene rifiutata se ciò fa sì che la frequenza massima di recupero dei lavori in corso superi la quota specificata dalla policy. bytes-per-hour Se una richiesta di processo di recupero viene respinta, viene visualizzato un messaggio di errore indicante che la richiesta è stata rifiutata dalla policy di recupero dati corrente.

L'impostazione della politica di recupero dei dati sulla politica Max Retrieval Rate può influire sulla quantità di franchigia del piano gratuito che è possibile utilizzare in un giorno. AWS Supponiamo, ad esempio, di impostare Max Retrieval Rate su 1 MB per ora. Questa tariffa è inferiore alla tariffa prevista dalla politica Free Tier AWS . Per assicurarti di utilizzare al meglio l'indennità giornaliera del piano AWS gratuito, puoi prima impostare la tua politica su Free Tier Only, quindi passare alla politica Max Retrieval Rate in un secondo momento, se necessario. Per ulteriori informazioni su come viene calcolata l'indennità di recupero, consulta Amazon Glacier. FAQs

Policy No Retrieval Limit

Se la tua policy di recupero dati è impostata su Nessun limite di recupero, tutte le richieste di recupero dati valide vengono accettate e i tuoi costi di recupero dati variano in funzione dell'utilizzo.

Utilizzo della console Amazon Glacier per configurare una policy di recupero dei dati

Per creare una politica di recupero dei dati utilizzando la console Amazon Glacier

1. Accedi Console di gestione AWS e apri la console Amazon Glacier da casa. https://console.aws.amazon.com/glacier/
2. In Seleziona una regione, scegline una Regione AWS dal menu a discesa. Puoi configurare una politica di recupero dei dati per ciascuno. Regione AWS
3. Nel riquadro di navigazione sinistro, seleziona Impostazioni di recupero dati.
4. Scegli Modifica. Viene visualizzata la pagina Modifica policy di recupero dati.
5. In Policy di recupero dati, seleziona una policy.

Puoi selezionare una delle tre policy di recupero dati: Solo piano gratuito, Velocità di recupero massima e Nessun limite di recupero.

- Se selezioni Nessun limite di recupero, vengono accettate tutte le richieste di recupero dati valide.
- Se scegli solo il piano gratuito, le richieste di recupero dati che superano il piano AWS gratuito non vengono accettate.
- Se selezioni Specifica una velocità di recupero massima, le richieste di recupero dati vengono rifiutate se fanno sì che la velocità di recupero di picco dei processi in corso superi la velocità di recupero massima specificata. È necessario specificare un valore in gigabyte (GB) all'ora nella casella GB/ora sotto Velocità di recupero massima. Quando digiti un valore in GB/ora, la console calcola automaticamente un costo stimato.

6. Scegli Save changes (Salva modifiche).

Utilizzo dell'API Amazon Glacier per configurare una politica di recupero dei dati

Puoi visualizzare e impostare una policy di recupero dei dati utilizzando l'API REST di Amazon Glacier o utilizzando il. AWS SDKs

Utilizzo dell'API REST di Amazon Glacier per configurare una policy di recupero dei dati

Puoi visualizzare e impostare una politica di recupero dei dati utilizzando l'API REST di Amazon Glacier. Per visualizzare una policy di recupero dati esistente, utilizza l'operazione Get Data Retrieval Policy (GET policy). Puoi configurare una policy di recupero dati utilizzando l'operazione Set Data Retrieval Policy (PUT policy).

Se utilizzi l'operazione di policy PUT, seleziona il tipo di policy di recupero dati impostando il valore del campo `Strategy` JSON su `BytesPerHour`, `FreeTier` o `None`. `BytesPerHour`equivale alla selezione di Specifica una velocità di recupero massima nella console, `FreeTier` equivale alla selezione di Solo piano gratuito, mentre `None` equivale alla selezione di Nessun limite di recupero.

Se utilizzi l'operazione Initiate Job (POST jobs) per avviare un processo di recupero dati che supererà la velocità di recupero massima definita nella tua policy, l'operazione Initiate Job viene interrotta e viene generata un'eccezione.

Utilizzo di AWS SDKs per impostare una politica di recupero dei dati

AWS ti consente SDKs di sviluppare applicazioni per Amazon Glacier. Questi SDKs forniscono librerie mappate all'API REST sottostante e forniscono oggetti che consentono di creare richieste ed elaborare risposte con facilità. Per ulteriori informazioni, consulta Utilizzo di AWS SDKs con Amazon Glacier.

Etichettare le risorse di Amazon Glacier

Un tag è un'etichetta che assegni a una risorsa. AWS Ogni tag consiste di una chiave e di un valore, entrambi personalizzabili. Puoi assegnare i tag che definisci alle risorse del vault Amazon Glacier (Amazon Glacier). L'utilizzo dei tag è un modo semplice ma efficace per gestire AWS le risorse e organizzare i dati, inclusi i dati di fatturazione.

Argomenti

- Nozioni di base sul tagging
- Limitazioni applicate ai tag
- Monitoraggio dei costi mediante il tagging
- Gestione del controllo degli accessi con il tagging
- Sezioni correlate

Nozioni di base sul tagging

Utilizza la console Amazon Glacier AWS Command Line Interface ,AWS CLI() o l'API Amazon Glacier per completare le seguenti attività:

- Aggiunta di tag a un vault
- Generazione di un elenco di tag per un vault
- Rimozione di tag da un vault

Per informazioni su come aggiungere, elencare e rimuovere tag, consulta Etichettatura dei vault Amazon Glacier.

Puoi utilizzare i tag per categorizzare i vault, ad esempio in base a scopo, proprietario o ambiente. Poiché definisci una chiave e un valore per ogni tag, puoi creare un set di categorie personalizzate per soddisfare esigenze specifiche. Ad esempio, puoi definire un set di tag che consente di monitorare i vault per proprietario e scopo. Di seguito sono riportati alcuni esempi di tag:

- Proprietario: nome
- Scopo: archivi video
- Ambiente: produzione

Limitazioni applicate ai tag

Le limitazioni di base relative ai tag sono:

- Il numero massimo di tag per una risorsa (vault) è 50.
- Per chiavi e valori di tag viene fatta la distinzione tra maiuscole e minuscole.

Le limitazioni relative alle chiavi di tag sono:

- In un set di tag per un vault, ogni chiave di tag deve essere univoca. Se aggiungi un tag con una chiave già in uso, il nuovo tag sovrascrive la coppia chiave-valore esistente.
- Le chiavi dei tag non possono iniziare con `aws:` perché questo prefisso è riservato all'uso di. AWS AWS puoi creare tag che iniziano con questo prefisso per tuo conto, ma non puoi modificarli o eliminarli.
- Le chiavi di tag devono contenere da 1 a 128 caratteri Unicode.
- Le chiavi di tag devono contenere i seguenti caratteri: numeri, spazi lettere Unicode e i seguenti caratteri speciali: `_ . / = + - @`.

Le limitazioni applicate ai valori di tag sono:

- I valori di tag devono contenere da 0 a 255 caratteri Unicode.
- I valori dei tag possono essere vuoti. In caso contrario, devono contenere i seguenti caratteri: numeri, spazi, lettere Unicode e qualsiasi carattere speciale tra quelli elencati di seguito: `_ . / = + - @`.

Monitoraggio dei costi mediante il tagging

Puoi utilizzare i tag per classificare e tenere traccia AWS dei costi. Quando applichi i tag a qualsiasi AWS risorsa, compresi gli archivi, il report sull'allocazione AWS dei costi include l'utilizzo e i costi aggregati per tag. Puoi applicare i tag che rappresentano categorie di business (come centri di costo, nomi di i e proprietari) per organizzare i costi tra più servizi. Per ulteriori informazioni, consulta Utilizzo dei tag per l'allocazione dei costi ai fini dei report di fatturazione personalizzati nella AWS Billing User Guide (Guida per l'utente di Amazon API Gateway).

Gestione del controllo degli accessi con il tagging

Puoi utilizzare i tag come condizione in un'istruzione di policy di accesso. Ad esempio, puoi impostare un tag di blocco a fini giudiziari e includerlo come condizione in una policy di conservazione di dati secondo cui "l'eliminazione dell'archivio da parte di qualsiasi utente sarà negata se il valore del tag di blocco a fini giudiziari è impostato su `True`". In condizioni normali, puoi distribuire la policy di conservazione di dati e impostare il tag di blocco a fini giudiziari su `False`. Se i dati devono essere bloccati per assistere un'indagine, puoi facilmente attivare il blocco a fini giudiziari impostando il valore del tag su `True` e rimuovere il blocco in modo analogo in seguito. Per ulteriori informazioni, consulta Controllo dell'accesso alle risorse AWS mediante i tag nella Guida per l'utente di .

Sezioni correlate

- Etichettatura dei vault Amazon Glacier

Registrazione delle chiamate API Amazon Glacier con AWS CloudTrail

Amazon Glacier (Amazon Glacier) è integrato AWS CloudTrail con un servizio che fornisce una registrazione delle azioni intraprese da un utente, ruolo o servizio in Amazon Glacier. AWS CloudTrail acquisisce tutte le chiamate API per Amazon Glacier come eventi, incluse le chiamate dalla console Amazon Glacier e le chiamate di codice ad Amazon Glacier. APIs Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per Amazon Glacier. Se non configuri un trail, è comunque possibile visualizzare gli eventi più recenti nella console di CloudTrail in Event history (Cronologia eventi). Utilizzando le informazioni raccolte da CloudTrail, puoi determinare la richiesta che è stata effettuata ad Amazon Glacier, l'indirizzo IP da cui è stata effettuata, chi ha effettuato la richiesta, quando è stata effettuata e ulteriori dettagli.

Per ulteriori informazioni CloudTrail, consulta la Guida per l'AWS CloudTrail utente.

Informazioni su Amazon Glacier in CloudTrail

CloudTrail è abilitato sul tuo account al Account AWS momento della creazione dell'account. Quando si verifica un'attività in Amazon Glacier, tale attività viene registrata in CloudTrail un evento insieme ad AWS altri eventi di servizio nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare eventi recenti in. Account AWS Per ulteriori informazioni, consulta Visualizzazione degli eventi con la cronologia degli CloudTrail eventi.

Per una registrazione continua degli eventi del tuo sito Account AWS, compresi gli eventi per Amazon Glacier, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando crei un percorso nella console, il percorso si applica a tutte le AWS regioni. Il trail registra gli eventi di tutte le AWS regioni della AWS partizione e consegna i file di log al bucket Amazon S3 specificato. Inoltre, puoi configurare altri AWS servizi per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei log. CloudTrail Per ulteriori informazioni, consulta:

- Panoramica della creazione di un trail
- CloudTrail Servizi e integrazioni supportati
- Configurazione delle notifiche Amazon SNS per CloudTrail

- Ricezione di file di CloudTrail registro da più regioni e ricezione di file di CloudTrail registro da più account

Tutte le azioni di Amazon Glacier vengono registrate e CloudTrail documentate in. Riferimento API per Amazon Glacier Ad esempio, le chiamate a Create Vault (PUT vault)Delete Vault (DELETE vault), e le List Vaults (GET vaults) azioni generano voci nei file di registro. CloudTrail

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con credenziali utente root o con altre credenziali.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro AWS servizio.

Per ulteriori informazioni, consulta Elemento CloudTrail userIdentity.

Comprendere le voci dei file di log di Amazon Glacier

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'azione richiesta, la data e l'ora dell'azione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia ordinata dello stack delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail registro che mostra le azioniCreate Vault (PUT vault), Delete Vault (DELETE vault)List Vaults (GET vaults), eDescribe Vault (GET vault).


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

Riferimento API per Amazon Glacier

Amazon Glacier supporta una serie di operazioni, in particolare una serie di chiamate API, che consentono RESTful di interagire con il servizio.

Puoi utilizzare qualsiasi libreria di programmazione in grado di inviare richieste HTTP per inviare le tue richieste REST ad Amazon Glacier. Quando si invia una richiesta REST, Amazon Glacier richiede l'autenticazione di ogni richiesta firmando la richiesta. Inoltre, quando carichi un archivio, devi anche calcolare il checksum del payload e includerlo nella richiesta. Per ulteriori informazioni, consulta Firmare le richieste.

Se si verifica un errore, devi sapere cosa invia Amazon Glacier in una risposta di errore in modo da poterlo elaborare. Oltre a documentare le operazioni REST, questa sezione fornisce tutte le informazioni necessarie per effettuare chiamate API REST direttamente.

Puoi utilizzare direttamente le chiamate API REST o utilizzare Amazon SDKs che fornisce librerie wrapper. Queste librerie firmano ogni richiesta inviata e calcolano il checksum del payload nella tua richiesta. Pertanto, l'utilizzo di Amazon SDKs semplifica le attività di codifica. Questa guida per sviluppatori fornisce esempi pratici di operazioni di base di Amazon Glacier utilizzando AWS SDK per Java e.NET. Per ulteriori informazioni, consultare Utilizzo di AWS SDKs con Amazon Glacier.

Argomenti

- Intestazioni di richiesta comuni
- Intestazioni di risposta comuni
- Firmare le richieste
- Calcolo di checksum
- Risposte agli errori
- Operazioni di vault
- Operazioni di archivio
- Operazioni di caricamento in più parti
- Operazioni di processo
- Tipi di dati utilizzati nelle operazioni di processo
- Operazioni di recupero di dati

Intestazioni di richiesta comuni

Le richieste REST di Amazon Glacier (Amazon Glacier) includono intestazioni che contengono informazioni di base sulla richiesta. La tabella seguente descrive le intestazioni che possono essere utilizzate da tutte le richieste REST di Amazon Glacier.

Header Name (Nome intestazione)	Descrizione	Richiesto
`Authorization`	L'intestazione necessaria per firmare le richieste. Amazon Glacier richiede la versione 4 di Signature . Per ulteriori informazioni, consulta Firmare le richieste. -Tipo: stringa	Sì
`Content-Length`	La lunghezza del corpo della richiesta (senza intestazioni). -Tipo: stringa Condizione: necessaria solo per l'API Upload Archive (POST archive).	Condizionale
`Date`	La data che può essere utilizzata per creare la firma contenuta nell'intestazione `Authorization` . Se per la firma è necessario utilizzare l'intestazione `Date`, questa deve essere specificata nel formato di base ISO 8601. In questo caso, l'intestazione `x-amz-date` non è necessaria. Nota che quando `x-amz-date` è presente, sovrascrive sempre il valore dell'intestazione `Date`. Se l'intestazione Date non viene utilizzata per la registrazione, può essere uno dei formati di data completa specificati dal protocollo RFC 2616, sezione 3.3. Ad esempio, la seguente data/ora `Wed, 10 Feb 2017 12:00:00 GMT` è un' date/	Condizionale

Header Name (Nome intestazione)	Descrizione	Richiesto
	time intestazione valida per l'uso con Amazon Glacier. Se utilizzi l'intestazione `Date` per la firma, deve essere nel formato di base ISO 8601 `YYYYMMDD'T'HHMMSS'Z'` . -Tipo: stringa Condizione: se l'intestazione `Date` è specifica ta, ma non è nel formato di base ISO 8601, devi anche includere l'intestazione `x-amz-date` . Se `Date` è nel formato di base ISO 8601, è sufficien te per firmare le richieste e l'intestazione `x-amz-date` non è necessaria. Per ulteriori informazioni, consulta Gestione delle date in Signature Version 4 nel glossario di Amazon Web Services.	
`Host`	Questa intestazione specifica l'endpoint del servizio a cui invii le richieste. Il valore deve avere il formato "`glacier.`*`region`*`.amazonaws.com` «, dove *`region`* viene sostituito con una designazi one di regione, ad esempio. AWS `us-west-2` -Tipo: stringa	Sì

Header Name (Nome intestazione)	Descrizione	Richiesto
`x-amz-content-sha256`	Il SHA256 checksum calcolato di un intero payload caricato con o. Upload Archive (POST archive) Upload Part (PUT uploadID) Questa intestazione non è identica all'intestazione `x-amz-sha256-tree-hash` , anche se per alcuni payload di piccole dimensioni i valori sono gli stessi. Quando l'intestazione `x-amz-content-sha256` è richiesta, devi specificare `x-amz-content-sha256` e `x-amz-sha256-tree-hash` . ▪Tipo: stringa Condizione: necessaria per l'API di streaming , Upload Archive (POST archive) e Upload Part (PUT uploadID).	Condizionale
`x-amz-date`	Data utilizzata per creare la firma nell'intestazione di autorizzazione. Deve essere nel formato di base ISO 8601 (`YYYYMMDD'T'HHMMSS'Z'`). Ad esempio, quanto segue date/time `20170210T120000Z` è valido `x-amz-date` per l'uso con Amazon Glacier. ▪Tipo: stringa Condizione: `x-amz-date` è facoltativa per tutte le richieste; può essere utilizzata per sovrascrivere la data utilizzata per firmare le richieste. Se l'intestazione `Date` è specificata nel formato di base ISO 8601, l'intestazione `x-amz-date` non è necessaria. Quando `x-amz-date` è presente, sovrascrive sempre il valore dell'intestazione `Date`. Per ulteriori informazioni, consulta Gestione delle date in Signature Version 4 nel glossario di Amazon Web Services.	Condizionale

Header Name (Nome intestazione)	Descrizione	Richiesto
`x-amz-glacier-version`	La versione dell'API Amazon Glacier da usare. La versione corrente è `2012-06-01` . -Tipo: stringa	Sì
`x-amz-sha256-tree-hash`	Il checksum SHA256 tree-hash calcolato per un archivio () o una parte di archivio () Upload Archive (POST archive) caricati. Upload Part (PUT uploadID) Per ulteriori informazioni sul calcolo di questo checksum, consulta Calcolo di checksum. -Tipo: stringa Impostazione predefinita: nessuna Condizione: necessaria per Upload Archive (POST archive) e Upload Part (PUT uploadID).	Condizionale

Intestazioni di risposta comuni

La tabella seguente descrive le intestazioni di risposta comuni alla maggior parte delle risposte dell'API.

Nome	Descrizione
`Content-Length`	La lunghezza in byte del corpo della risposta. -Tipo: stringa
`Date`	La data e l'ora in cui Amazon Glacier (Amazon Glacier) ha risposto, ad esempio,. `Wed, 10 Feb 2017 12:00:00 GMT` Il formato deve essere uno dei formati di data completa indicato dal protocollo RFC 2616, sezione 3.3. L'elemento `Date` restituito può essere leggermente diverso da altre date, pertanto la data restituita da una richiesta Upload Archive (POST archive)

Nome	Descrizione
	potrebbe ad esempio non corrispondere alla data visualizzata per l'archivio in un elenco di inventario per il vault. -Tipo: stringa
`x-amzn-RequestId`	Un valore creato da Amazon Glacier che identifica in modo univoco la tua richiesta. Nel caso in cui tu abbia un problema con Amazon Glacier AWS , puoi utilizzare questo valore per risolvere il problema. Ti consigliamo di registrare tali valori. -Tipo: stringa
`x-amz-sha256-tree-hash`	Il checksum SHA256 tree-hash dell'archivio o del corpo dell'inventario. Per ulteriori informazioni sul calcolo di questo checksum, consulta Calcolo di checksum. -Tipo: stringa

Firmare le richieste

Amazon Glacier richiede l'autenticazione di ogni richiesta inviata firmando la richiesta. Per firmare una richiesta, è necessario calcolare una firma digitale utilizzando una funzione hash crittografica. Una funzione hash crittografica è una funzione che restituisce un valore hash univoco basato sull'input. L'input alla funzione hash include il testo della richiesta e la tua chiave di accesso segreta. La funzione hash restituisce un valore hash che includi nella richiesta come firma. La firma è parte dell'intestazione `Authorization` della richiesta.

Dopo aver ricevuto la richiesta, Amazon Glacier ricalcola la firma utilizzando la stessa funzione di hash e lo stesso input utilizzati per firmare la richiesta. Se la firma risultante corrisponde alla firma nella richiesta, Amazon Glacier elabora la richiesta. In caso contrario, la richiesta viene respinta.

Amazon Glacier supporta l'autenticazione AWS tramite Signature versione 4. La procedura per il calcolo di una firma può essere suddivisa in tre fasi:

- Fase 1. Creazione di una richiesta canonica

Riorganizza la richiesta HTTP in un formato canonico. L'utilizzo di un modulo canonico è necessario perché Amazon Glacier utilizza la stessa forma canonica quando ricalcola una firma per confrontarla con quella che hai inviato.

- Fase 2. Creazione di una stringa di firma

  Crea una stringa che utilizzerai come uno dei valori di input per la funzione hash crittografica. La stringa, denominata stringa di firma, è una concatenazione del nome dell'algoritmo hash, della data della richiesta, di una stringa di ambito credenziali e della richiesta in formato canonico creata nella fase precedente. La stringa dell'ambito delle credenziali è di per sé una concatenazione di informazioni su data, regione e servizio. AWS

- Fase 3. Creazione di una firma

  Crea una firma per la tua richiesta utilizzando una funzione hash crittografica che accetta due stringhe di input: la tua stringa di firma e una chiave derivata. La chiave derivata viene calcolata partendo dalla chiave di accesso segreta e utilizzando la stringa di ambito delle credenziali per creare una serie di codici di autenticazione dei messaggi basati su hash (). HMACs Tieni presente che la funzione hash utilizzata in questa fase di firma non è l'algoritmo tree-hash utilizzato in Amazon Glacier per caricare i dati. APIs

Argomenti

- Esempio di calcolo di firma
- Calcolo di firme per le operazioni di streaming

Esempio di calcolo di firma

L'esempio in questa sezione mostra come creare una firma per Create Vault (PUT vault). L'esempio può essere utilizzato come riferimento per verificare il metodo di calcolo della firma. Per ulteriori informazioni, consulta Signing AWS API request nella IAM User Guide.

L'esempio presuppone quanto segue:

- Il time stamp della richiesta è `Fri, 25 May 2012 00:24:53 GMT`.
- L'endpoint è la regione Stati Uniti orientali (Virginia settentrionale), `us-east-1`.

La sintassi generale della richiesta (incluso il corpo JSON) è:

```
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
```

Il formato canonico della richiesta calcolata per Fase 1: creazione di una richiesta canonica è:

```
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
```

L'ultima riga della richiesta canonica è l'hash del corpo della richiesta. Nota inoltre la terza riga vuota nella richiesta canonica. Ciò è dovuto alla mancanza di parametri di query per questa API.

La stringa di firma per Fase 2. Creazione di una stringa di firma è:

```
AWS4-HMAC-SHA256
20120525T002453Z
20120525/us-east-1/glacier/aws4_request
5f1da1a2d0feb614dd03d71e87928b8e449ac87614479332aced3a701f916743
```

La prima riga della stringa di firma è l'algoritmo, la seconda è il timestamp, la terza è l'ambito credenziali e l'ultima è un hash del formato della richiesta canonica in Fase 1: creazione di una richiesta canonica. Il nome del servizio da utilizzare nell'ambito credenziali è `glacier`.

In Fase 3. Creazione di una firma, la chiave derivata può essere rappresentata come segue:

```
derived key = HMAC(HMAC(HMAC(HMAC("AWS4" + YourSecretAccessKey,"20120525"),"us-
east-1"),"glacier"),"aws4_request")
```

Se viene utilizzata la chiave di accesso segreta `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`, la firma calcolata è:

```
3ce5b2f2fffac9262b4da9256f8d086b4aaf42eba5f111c21681a65a127b7c2a
```

La fase finale consiste nel creare l'intestazione `Authorization`. Per la chiave di accesso `AKIAIOSFODNN7EXAMPLE`, l'intestazione (con interruzioni di riga aggiunte per facilitare la lettura) è:

```
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20120525/us-east-1/
glacier/aws4_request,
SignedHeaders=host;x-amz-date;x-amz-glacier-version,
Signature=3ce5b2f2fffac9262b4da9256f8d086b4aaf42eba5f111c21681a65a127b7c2a
```

Calcolo di firme per le operazioni di streaming

Upload Archive (POST archive) e Upload Part (PUT uploadID) sono operazioni di streaming che richiedono l'inclusione di un'ulteriore intestazione `x-amz-content-sha256` per la firma e l'invio della richiesta. Le fasi relative alla firma per le operazioni di streaming sono esattamente le stesse delle altre operazioni, con l'aggiunta dell'intestazione di streaming.

Il calcolo dell'intestazione dello streaming si `x-amz-content-sha256` basa sull'SHA256 hash dell'intero contenuto (payload) da caricare. Nota che questo calcolo è diverso dal SHA256 tree hash (). Calcolo di checksum Oltre ai casi banali, il valore hash SHA 256 dei dati del payload sarà diverso dall'hash ad SHA256 albero dei dati del payload.

Se i dati del payload sono specificati come array di byte, è possibile utilizzare il seguente frammento di codice Java per calcolare l'hash. 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();
}
```

Allo stesso modo, in C# è possibile calcolare l' SHA256 hash dei dati del payload come mostrato nel seguente frammento di codice.

```
public static byte[] CalculateSHA256Hash(byte[] payload)
{
    SHA256 sha256 = System.Security.Cryptography.SHA256.Create();
    byte[] hash = sha256.ComputeHash(payload);

    return hash;
}
```

Esempio di calcolo di firma per l'API di streaming

L'esempio seguente illustra i dettagli della creazione di una firma perUpload Archive (POST archive), uno dei due streaming APIs in Amazon Glacier. L'esempio presuppone quanto segue:

- Il time stamp della richiesta è `Mon, 07 May 2012 00:00:00 GMT`.
- L'endpoint è la regione Stati Uniti orientali (Virginia settentrionale), us-east-1.
- Il payload del contenuto è una stringa «Benvenuto in Amazon Glacier».

La sintassi generale della richiesta (incluso il corpo JSON) è illustrata nell'esempio seguente. Nota che è inclusa l'intestazione `x-amz-content-sha256`. In questo esempio semplificato, `x-amz-sha256-tree-hash` e `x-amz-content-sha256` hanno lo stesso valore, ma nel caso di caricamenti di archivi superiori a 1 MB sono differenti.

```
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
```

Il formato canonico della richiesta calcolata per Fase 1. Creazione di una richiesta canonica è riportato di seguito. Nota che l'intestazione di streaming `x-amz-content-sha256` è inclusa con il relativo valore. Ciò significa che devi prima leggere il payload e calcolare l' SHA256 hash, quindi calcolare la firma.

```
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
```

Il resto della procedura di calcolo della firma segue le fasi descritte in Esempio di calcolo di firma. L'intestazione `Authorization` che utilizza la Secret Access Key `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY` e la chiave di accesso `AKIAIOSFODNN7EXAMPLE` è esposta di seguito (con interruzioni di riga aggiunte per facilitare la lettura):

```
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
```

Calcolo di checksum

Durante il caricamento di un archivio, devi includere le intestazioni `x-amz-sha256-tree-hash` e `x-amz-content-sha256`. L'intestazione `x-amz-sha256-tree-hash` è un checksum del payload nel corpo della richiesta. Questo argomento descrive come calcolare l'intestazione `x-amz-sha256-tree-hash`. L'intestazione `x-amz-content-sha256` è un hash dell'intero payload ed è necessaria per l'autorizzazione. Per ulteriori informazioni, consulta Esempio di calcolo di firma per l'API di streaming.

Il payload della richiesta può essere:

- Archivio completo: durante il caricamento di un archivio in un'unica richiesta mediante l'API Upload Archive, l'intero archivio viene inviato nel corpo della richiesta. In questo caso, devi includere il checksum dell'intero archivio.
- Parte di un archivio: durante il caricamento di un archivio in parti mediante l'API corrispondente, solo una parte dell'archivio viene inviata nel corpo della richiesta. In questo caso, devi includere il checksum di quella parte dell'archivio. Dopo aver caricato tutte le parti, è necessario inviare una richiesta Complete Multipart Upload che deve includere il checksum dell'intero archivio.

Il checksum del payload è una struttura hash SHA-256. È denominata struttura hash in quanto nel processo di calcolo del checksum viene calcolata una struttura di valori hash SHA-256. Il valore hash nella radice è il checksum dell'intero archivio.

> **Note**
>
> Questa sezione descrive un metodo di calcolo della struttura hash SHA-256. Tuttavia, puoi utilizzare qualsiasi altra procedura purché produca lo stesso risultato.

Per calcolare la struttura hash SHA-256, procedi come segue:

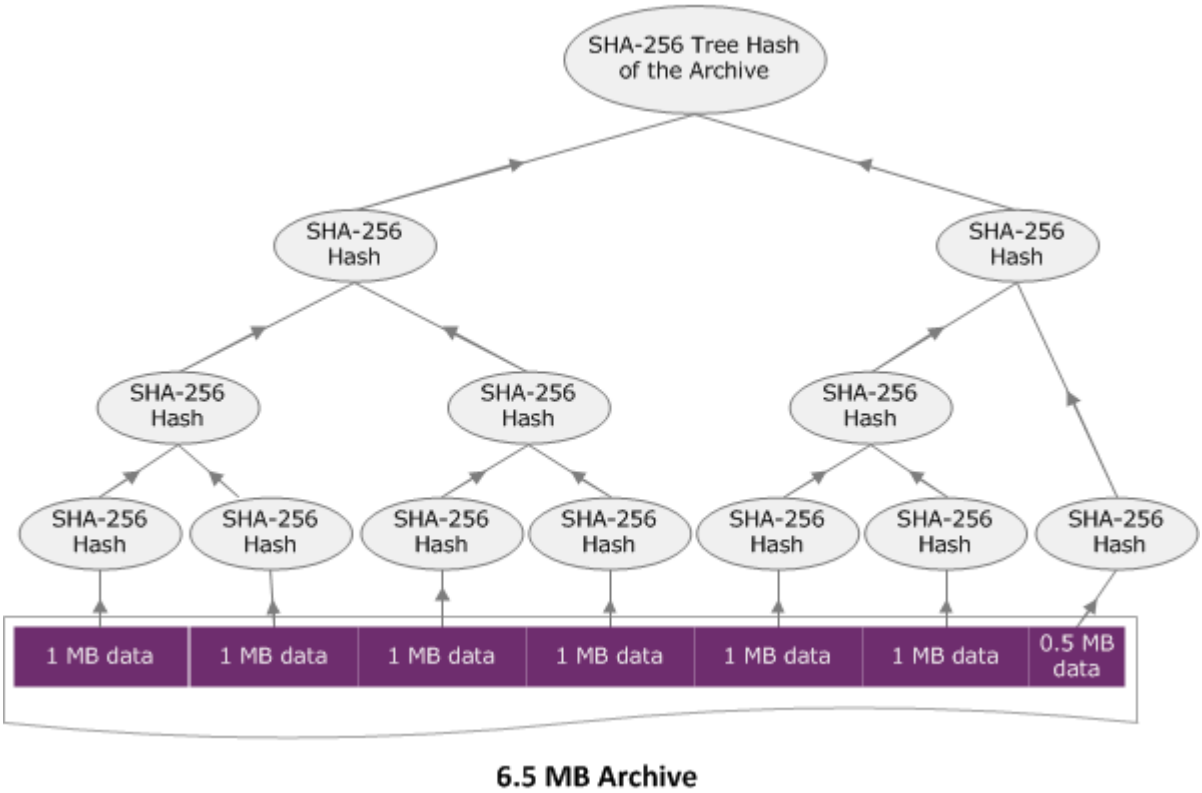
1. Per ogni blocco di 1 MB di dati di payload, calcola l'hash SHA-256. L'ultimo blocco di dati può essere inferiore a 1 MB. Ad esempio, se stai caricando un archivio di 3,2 MB, calcoli i valori hash SHA-256 per ognuno dei primi tre blocchi di 1 MB di dati, quindi calcoli l'hash SHA-256 dei restanti 0,2 MB di dati. Questi valori hash formano i nodi foglia della struttura.
2. Crea il livello successivo della struttura.
   a. Concatena i valori hash di due nodi figlio consecutivi e calcola l'hash SHA-256 dei valori hash concatenati. Questa concatenazione e generazione dell'hash SHA-256 produce un nodo padre per i due nodi figlio.
   b. Quando rimane un solo nodo figlio, alza il valore hash al livello successivo nella struttura.
3. Ripeti la fase 2 fino a che la struttura risultante ha una radice. La radice della struttura fornisce un hash dell'intero archivio e una radice della sottostruttura appropriata fornisce l'hash della parte in un caricamento in più parti.

Argomenti

- Esempio di struttura hash 1: caricamento di un archivio in un'unica richiesta
- Esempio di struttura hash 2: caricamento di un archivio mediante un caricamento in più parti
- Calcolo della struttura hash di un file
- Ottenimento di checksum durante il download di dati

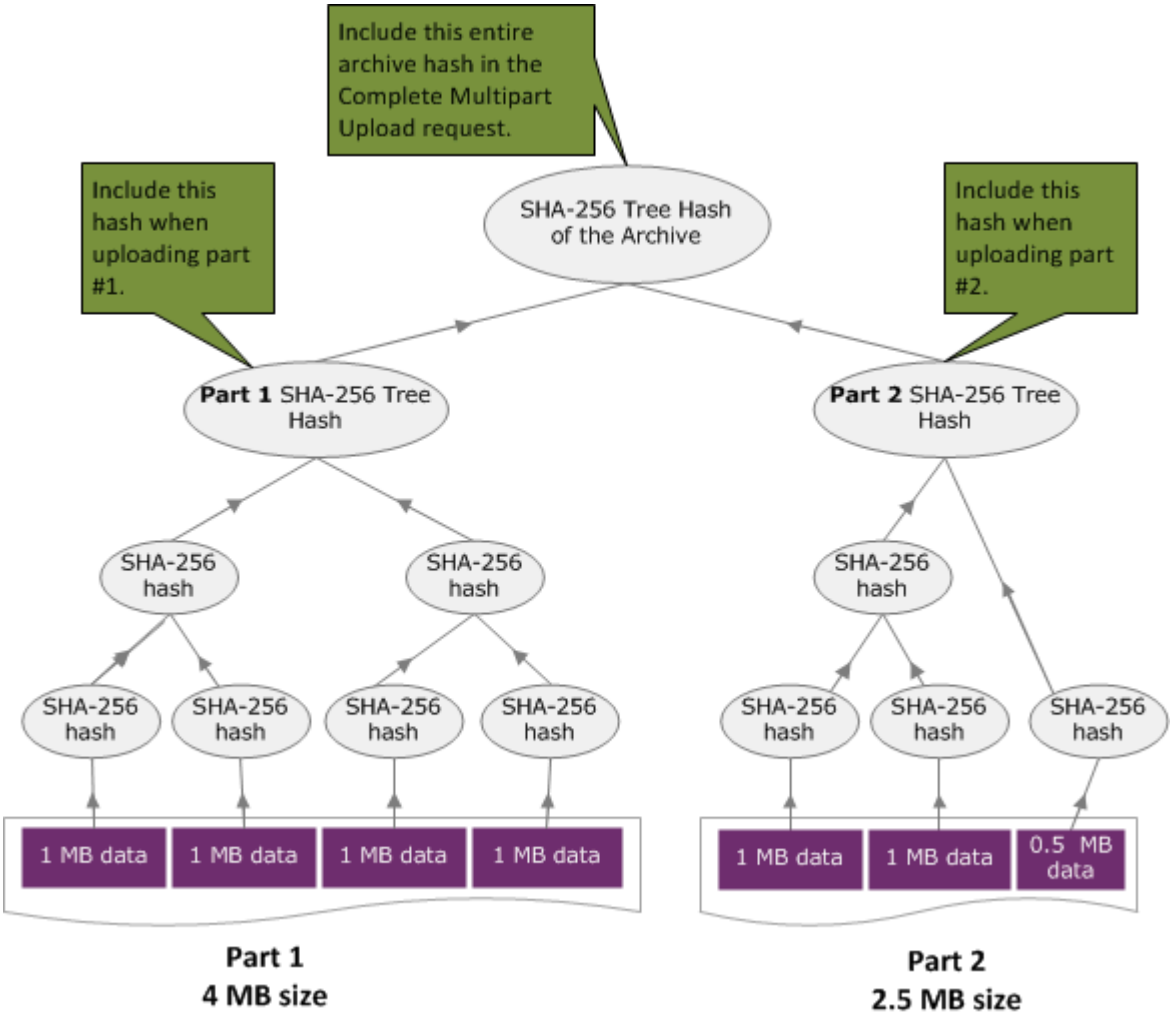
Esempio di struttura hash 1: caricamento di un archivio in un'unica richiesta

Quando carichi un archivio in un'unica richiesta mediante l'API Upload Archive (consulta Upload Archive (POST archive)), il payload della richiesta include l'intero archivio. Di conseguenza, devi includere la struttura hash dell'intero archivio nell'intestazione di richiesta `x-amz-sha256-tree-hash`. Supponi di voler caricare un archivio di 6,5 MB. Il diagramma seguente illustra il processo di creazione dell'hash SHA-256 dell'archivio. Leggi l'archivio e calcola l'hash SHA-256 per ogni blocco di 1 MB. Calcola anche l'hash per i restanti 0,5 MB di dati e quindi crea la struttura come descritto nella procedura precedente.



Esempio di struttura hash 2: caricamento di un archivio mediante un caricamento in più parti

Il processo di calcolo della struttura hash durante il caricamento di un archivio mediante il caricamento in più parti è identico a quello per il caricamento dell'archivio in un'unica richiesta. L'unica differenza è che in un caricamento in più parti carichi solo una parte dell'archivio in ogni richiesta (utilizzando l'API Upload Part (PUT uploadID)), di conseguenza fornisci soltanto il checksum della parte nell'intestazione di richiesta `x-amz-sha256-tree-hash`. Tuttavia, dopo il caricamento di tutte le parti, devi inviare la richiesta Complete Multipart Upload (vedi Complete Multipart Upload (POST uploadID)) con una struttura hash dell'intero archivio nell'intestazione di richiesta `x-amz-sha256-tree-hash`.



Calcolo della struttura hash di un file

Gli algoritmi illustrati di seguito sono selezionati a scopo dimostrativo. Puoi ottimizzare il codice in funzione delle esigenze del tuo scenario di implementazione. Se utilizzi un Amazon SDK per la programmazione con Amazon Glacier (Amazon Glacier), il calcolo dell'hash dell'albero viene eseguito automaticamente e devi solo fornire il riferimento al file.

Example 1. Esempio con Java

L'esempio seguente mostra come calcolare l'hash dell' SHA256 albero di un file utilizzando Java. Puoi eseguire questo esempio specificando una percorso di file come argomento oppure utilizzare il metodo `TreeHashExample.computeSHA256TreeHash` direttamente nel codice.

```
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. Esempio con C# .NET

L'esempio seguente mostra come calcolare l'hash dell' SHA256 albero di un file. Puoi eseguire questo esempio specificando un percorso di file come argomento.

```
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;
        }
    }
}
```

Ottenimento di checksum durante il download di dati

Quando recuperi un archivio utilizzando l'API Initiate Job (vedi Initiate Job (POST jobs)), puoi eventualmente specificare un intervallo di archivio da recuperare. Analogamente, quando scarichi i dati utilizzando l'API Get Job Output (vedi Get Job Output (GET output)), puoi eventualmente specificare un intervallo di dati da scaricare. È necessario comprendere due caratteristiche importanti di questi intervalli quando si recuperano e scaricano i dati di un archivio. L'intervallo da recuperare deve essere allineato in termini di megabyte all'archivio. L'intervallo da recuperare e quello da scaricare devono essere allineati alla struttura hash per poter ricevere valori di checksum durante il download dei dati. La definizione di questi due tipi di allineamenti di intervalli è riportata di seguito:

- Megabyte allineato - Un intervallo [StartByte, EndBytes] è megabyte (1024*1024) allineato quando StartBytesè divisibile per 1 MB e EndBytespiù 1 è divisibile per 1 MB o è uguale alla fine dell'archivio specificato (dimensione del byte dell'archivio meno 1). Se si specifica l'API Initiate Job, l'intervallo deve essere obbligatoriamente allineato in termini di megabyte.
- Tree-hash aligned - Un range [StartBytes, EndBytes] è l'hash dell'albero allineato rispetto a un archivio se e solo se la radice dell'hash dell'albero costruito sull'intervallo è equivalente a un nodo nell'hash dell'albero dell'intero archivio. L'intervallo da recuperare e quello da scaricare devono essere allineati alla struttura hash per poter ricevere i valori di checksum dei dati scaricati. Per un esempio di intervalli e della relativa relazione con la struttura hash di un archivio, consulta Esempio di struttura hash: recupero di un intervallo di archivio allineato alla struttura hash.

  Nota che un intervallo allineato alla struttura hash è anche allineato in termini di megabyte. Tuttavia, un intervallo allineato in termini di megabyte non è necessariamente allineato alla struttura hash.

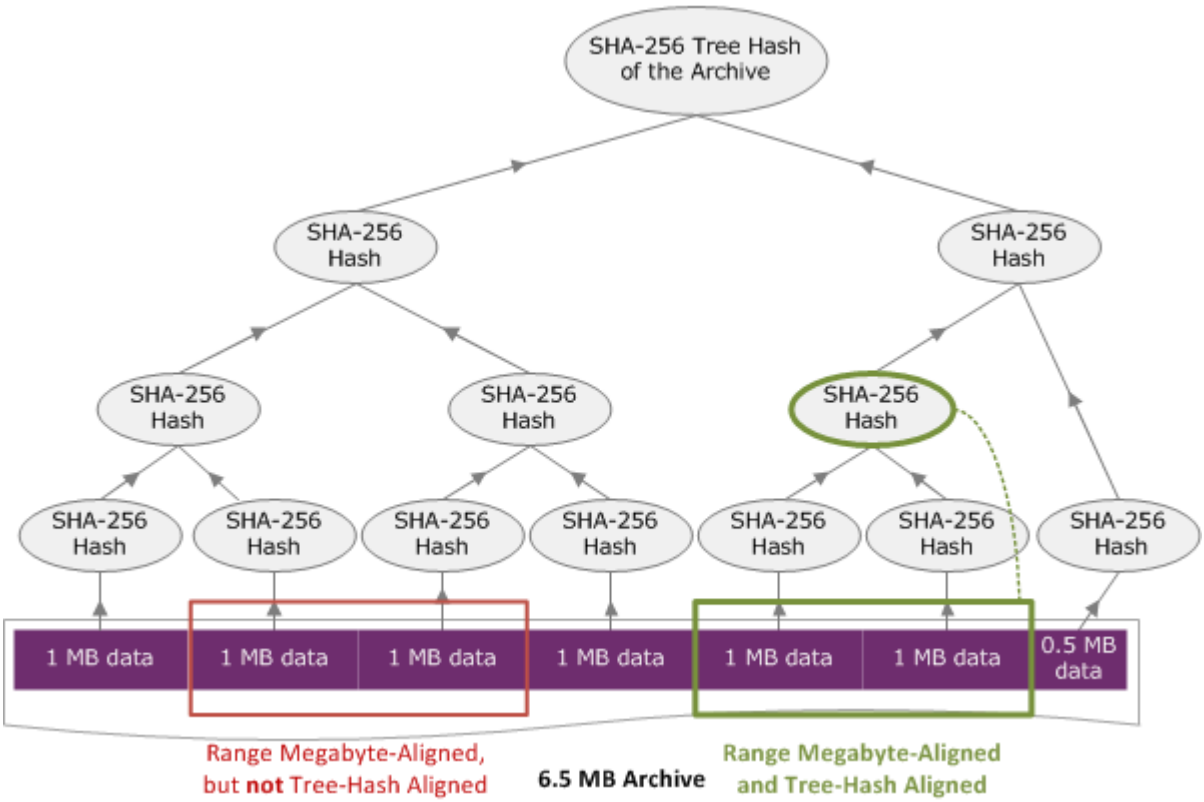
Di seguito sono riportati i casi in cui si ottiene un valore di checksum durante il download dei dati in archivio:

- Se non specifichi un intervallo da recuperare nella richiesta Initiate Job e scarichi l'intero archivio nella richiesta Get Job.
- Se non specifichi un intervallo da recuperare nella richiesta Initiate Job e specifichi un intervallo allineato alla struttura hash da scaricare nella richiesta Get Job.
- Se specifichi un intervallo allineato alla struttura hash da recuperare nella richiesta Initiate Job e scarichi l'intero intervallo nella richiesta Get Job.
- Se specifichi un intervallo allineato alla struttura hash da recuperare nella richiesta Initiate Job e specifichi un intervallo allineato alla struttura hash da scaricare nella richiesta Get Job.

Se specifichi un intervallo da recuperare nella richiesta Initiate Job che non è allineato alla struttura hash, puoi comunque ottenere i dati in archivio ma nessun valore di checksum viene restituito quando scarichi i dati nella richiesta Get Job.

Esempio di struttura hash: recupero di un intervallo di archivio allineato alla struttura hash

Supponi di avere un archivio di 6,5 MB nel tuo vault e che intendi recuperare 2 MB dell'archivio. Il modo in cui specifichi l'intervallo di 2 MB nella richiesta Initiate Job determina se ricevi i valori di checksum dei dati quando scarichi i dati. Il diagramma seguente illustra due intervalli di 2 MB per l'archivio di 6,5 MB che puoi scaricare. Entrambi gli intervalli sono allineati in termini di megabyte, ma solo uno è allineato alla struttura hash.



Specifica di un intervallo allineato alla struttura hash

Questa sezione fornisce l'esatta specifica di un intervallo allineato a una struttura hash. Gli intervalli allineati alla struttura hash sono importanti quando scarichi una parte di un archivio e specifichi l'intervallo di dati da recuperare e l'intervallo di dati recuperati da scaricare. Se entrambi questi intervalli sono allineati alla struttura hash, riceverai i valori di checksum quando scarichi i dati.

Un intervallo [A, B] è allineato alla struttura hash rispetto a un archivio se e solo se, quando una nuova struttura hash viene creata su [A, B], la radice della struttura hash di quell'intervallo è equivalente a un nodo nella struttura hash dell'intero archivio. Ciò è illustrato nel diagramma in Esempio di struttura hash: recupero di un intervallo di archivio allineato alla struttura hash. In questa sezione forniamo la specifica per l'allineamento alla struttura hash.

Considera [P, Q) come la query di intervallo di un archivio di N megabyte (MB) e che P e Q sono multipli di 1 MB. Nota che l'intervallo inclusivo effettivo è [P MB, Q MB – 1 byte], ma per semplicità, lo presentiamo come [P, Q). Sulla base di queste considerazioni:

- Se P è un numero dispari, un solo intervallo allineato alla struttura hash è possibile, ovvero [P, P +1 MB).

- Se P è un numero pari e k è il numero massimo, dove P può essere scritto come 2k * X, ci sono al massimo k intervalli allineati alla struttura hash che iniziano con P. X è un numero intero maggiore di 0. Gli intervalli allineati alla struttura hash rientrano nelle seguenti categorie:
  - In ogni i, dove (0 <= i <= k) e dove $P + 2^i < N$, $[P, Q + 2^i)$ è un intervallo allineato alla struttura hash.
  - P = 0 è il caso speciale dove A = 2[lgN]*0

Risposte agli errori

In caso di errore, l'API restituisce una delle seguenti eccezioni:

Codice	Descrizione	Codice di stato HTTP	Tipo
`AccessDeniedException`	Restituito se si è verificato un tentativo di accesso a una risorsa non consentito da una policy AWS Identity and Access Management (IAM) o se è stato utilizzato un Account AWS ID errato nell'URI della richiesta. Per ulteriori informazioni, consulta Identity and Access Management per Amazon Glacier.	`403 Forbidden`	Client
`BadRequest`	Restituita se la richiesta non può essere elaborata.	`400 Bad Request`	Client
`ExpiredTokenException`	Restituita se il token di sicurezza utilizzato nella richiesta è scaduto.	`403 Forbidden`	Client
`InsufficientCapacityException`	Restituita se la capacità non è sufficiente per elaborare la richiesta di tipo Expedited. Questo errore è relativo solo ai	`503 Service Unavailable`	Server

Codice	Descrizione	Codice di stato HTTP	Tipo
	recuperi di tipo Expedited e non a quelli di tipo Standard o Bulk.		
`InvalidParameterValueException`	Restituita se un parametro della richiesta non è stato specificato correttamente.	`400 Bad Request`	Client
`InvalidSignatureException`	Restituita se la firma della richiesta non è valida.	`403 Forbidden`	Client
`LimitExceededException`	Restituita se la richiesta supera uno dei seguenti limiti: limite di vault, limite di tag o limite di capacità con provisioning.	`400 Bad Request`	Client
`MissingAuthenticationTokenException`	Restituita se non vengono trovati dati di autenticazione della richiesta.	`400 Bad Request`	Client
`MissingParameterValueException`	Restituita se un'intestazione o un parametro necessario non è nella richiesta.	`400 Bad Request`	Client
`PolicyEnforcedException`	Restituita se un processo di recupero supera il limite per il tasso di recupero della policy di recupero dati corrente. Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon Glacier.	`400 Bad Request`	Client
`ResourceNotFoundException`	Restituita se la risorsa specifica ta, ad esempio un vault, un ID di caricamento o un job ID non esiste.	`404 Not Found`	Client

Codice	Descrizione	Codice di stato HTTP	Tipo
`RequestTimeoutException`	Restituito se si carica un archivio e Amazon Glacier (Amazon Glacier) scade durante la ricezione del caricamento.	`408 Request Timeout`	Client
`SerializationException`	Restituita se il corpo della richiesta non è valido. Se è incluso un payload JSON, verifica se il formato è corretto.	`400 Bad Request`	Client
`ServiceUnavailableException`	Restituita se il servizio non può completare la richiesta.	`500 Internal Server Error`	Server
`ThrottlingException`	Restituito se devi ridurre la frequenza delle richieste ad Amazon Glacier.	`400 Bad Request`	Client
`UnrecognizedClientException`	Restituita se l'ID chiave di accesso o il token di sicurezza non è valido.	`400 Bad Request`	Client

Diversi Amazon APIs Glacier restituiscono la stessa eccezione, ma con messaggi di eccezione diversi per aiutarti a risolvere l'errore specifico riscontrato.

Amazon Glacier restituisce informazioni sugli errori nel corpo della risposta. I seguenti esempi mostrano alcune delle risposte agli errori.

Esempio 1: richiesta Describe Job con un job ID inesistente

Supponiamo che invii una richiesta Describe Job (GET JobID) per un processo che non esiste, ovvero che specifichi un job ID inesistente.

```
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
```

In risposta, Amazon Glacier restituisce la seguente risposta di errore.

```
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"
  }
```

Dove:

Codice

Una delle eccezioni generali.

- Tipo: stringa

Message

Una descrizione generica della condizione di errore specifica dell'API che restituisce l'errore.

- Tipo: stringa

Tipo

L'origine dell'errore. Il campo può essere uno dei seguenti valori: `Client`, `Server` o `Unknown`.

Tipo: stringa.

Nota quanto segue in relazione alla risposta precedente:

- Per la risposta all'errore, Amazon Glacier restituisce i valori del codice di stato di and. `4xx` `5xx` In questo esempio, il codice di stato è `404 Not Found`.
- Il valore `Content-Type` dell'intestazione `application/json` indica JSON nel corpo.
- Il codice JSON nel corpo fornisce le informazioni sull'errore.

Nella richiesta precedente, anziché un job ID non valido, supponiamo di specificare un vault inesistente. La risposta restituisce un messaggio differente.

```
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"
}
```

Esempio 2: richiesta List Jobs con un valore non valido per il parametro di richiesta

In questo esempio, invii una richiesta List Jobs (GET jobs) per recuperare processi di vault con uno specifico `statuscode` e per `statuscode` fornisci il valore non valido `finished` anziché i valori accettabili `InProgress`, `Succeeded` o `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 restituisce `InvalidParameterValueException` il file con un messaggio appropriato.

```
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"
}
```

Operazioni di vault

Di seguito sono elencate le operazioni di vault disponibili in Amazon Glacier.

Argomenti

- Abort Vault Lock (DELETE lock-policy)
- Add Tags To Vault (POST tags add)
- Create Vault (PUT vault)
- Complete Vault Lock (POST lockId)
- Delete Vault (DELETE vault)
- Delete Vault Access Policy (DELETE access-policy)
- Delete Vault Notifications (DELETE notification-configuration)
- Describe Vault (GET vault)
- Get Vault Access Policy (GET access-policy)
- Get Vault Lock (GET lock-policy)
- Get Vault Notifications (GET notification-configuration)
- Initiate Vault Lock (POST lock-policy)
- List Tags For Vault (GET tags)
- List Vaults (GET vaults)
- Remove Tags From Vault (POST tags remove)
- Set Vault Access Policy (PUT access-policy)
- Set Vault Notification Configuration (PUT notification-configuration)

Abort Vault Lock (DELETE lock-policy)

Descrizione

Questa operazione arresta il processo di blocco della vault se lo stato del blocco della vault non è `Locked`. Se questa operazione viene richiesta quando lo stato del blocco di vault è `Locked`, l'operazione restituisce un errore `AccessDeniedException`. L'interruzione del processo di blocco della vault elimina la policy di blocco della vault dalla vault specificata.

Lo stato `InProgress` di un blocco della vault viene attivato chiamando Initiate Vault Lock (POST lock-policy). Lo stato `Locked` di un blocco della vault viene attivato chiamando Complete Vault Lock (POST lockId). Puoi ottenere lo stato di un blocco di vault chiamando Get Vault Lock (GET lock-policy). Per ulteriori informazioni sul processo di blocco della vault, consulta Serratura Amazon Glacier Vault. Per ulteriori informazioni sulle policy di blocco della vault, consulta Policy di Vault Lock.

Questa operazione è idempotente. Se lo stato del blocco di vault è `InProgress` oppure se nessuna policy è associata al vault, puoi richiamare questa operazione più volte senza errori.

Richieste

Per eliminare la policy di Vault Lock, invia una richiesta HTTP `DELETE` all'URI della sottorisorsa `lock-policy` del vault.

Sintassi

```
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**
>
> Il valore è l'ID. `AccountId` Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Se la policy viene eliminata con successo, Amazon Glacier restituisce una risposta. `HTTP 204 No Content`

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come arrestare un processo di blocco della vault.

Richiesta di esempio

In questo esempio, una richiesta `DELETE` è inviata alla sottorisorsa `lock-policy` della vault denominata **`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
```

Risposta di esempio

Se la policy viene eliminata con successo, Amazon Glacier restituisce `HTTP 204 No Content` una risposta, come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sezioni correlate

- Complete Vault Lock (POST lockId)
- Get Vault Lock (GET lock-policy)
- Initiate Vault Lock (POST lock-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Add Tags To Vault (POST tags add)

Questa operazione aggiunge i tag specificati a un vault. Ogni tag consiste di una chiave e un valore. Ogni vault può avere fino a 50 tag. Se la tua richiesta comporta il superamento del limite di tag per il vault, l'operazione genera l'errore `LimitExceededException`.

Se un tag con una chiave specificata è già presente nel vault, il valore di chiave esistente verrà sovrascritto. Per ulteriori informazioni sui tag, consulta Etichettare le risorse di Amazon Glacier.

Sintassi della richiesta

Per aggiungere tag a un vault, invia una richiesta HTTP POST all'URI dei tag come illustrato nell'esempio di sintassi seguente.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Nome	Descrizione	Richiesto
`operation=add`	Un singolo parametro di stringa di query `operation` con un valore `add` per differenziarlo da Remove Tags From Vault (POST tags remove).	Sì

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il corpo della richiesta contiene i campi JSON descritti di seguito.

Tags

I tag da aggiungere al vault. Ogni tag consiste di una chiave e un valore. Il valore può essere una stringa vuota.

Tipo: mappatura stringa a stringa

Vincoli della lunghezza: lunghezza minima 1. Lunghezza massima 10.

Campo obbligatorio: sì

Risposte

Se la richiesta riesce, il servizio restituisce una risposta HTTP `204 No Content`.

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP POST con i tag da aggiungere al vault.

```
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"
    }
}
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier restituisce `HTTP 204 No Content` un come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- List Tags For Vault (GET tags)
- Remove Tags From Vault (POST tags remove)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Create Vault (PUT vault)

Descrizione

Questa operazione crea un nuovo vault con il nome specificato. Il nome del vault deve essere univoco all'interno di una AWS regione per un Account AWS. È possibile creare fino a 1.000 vault per account. Per informazioni sulla creazione di altri vault, consulta la pagina dei dettagli del prodotto Amazon Glacier.

Quando assegni un nome a un vault, devi considerare quanto segue.

- I nomi possono contenere un numero di caratteri compreso tra 1 e 255.
- I caratteri validi sono a-z, , A-Z0-9, "_" (trattino basso), "-" (trattino) e "." (punto).

Questa operazione è idempotente, puoi inviare la stessa richiesta più volte e non ha ulteriori effetti dopo la prima volta che Amazon Glacier (Amazon Glacier) crea il vault specificato.

Richieste

Sintassi

Per creare un vault, invia una richiesta HTTP PUT all'URI del vault da creare.

```
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**
>
> Il valore è l'ID. `AccountId` Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Per questa operazione, il corpo della richiesta deve essere vuoto (0 byte).

Risposte

Sintassi

```
HTTP/1.1 201 Created
```

```
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`Location`	Il percorso URI relativo del vault che è stato creato. -Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP PUT per creare un vault denominato `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
```

Risposta di esempio

Amazon Glacier crea il vault e ne restituisce il percorso URI relativo nell'intestazione. `Location` L'ID account è sempre visualizzato nell'intestazione `Location` indipendentemente da quanto specificato nella richiesta, ovvero l'ID account o un trattino ('-').

```
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
```

Sezioni correlate

- List Vaults (GET vaults)
- Delete Vault (DELETE vault)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Complete Vault Lock (POST lockId)

Descrizione

Questa operazione completa il processo di blocco di vault cambiando lo stato del blocco di vault da `InProgress` a `Locked` e rendendo la policy di Vault Lock non modificabile. Lo stato `InProgress` di un blocco della vault viene attivato chiamando Initiate Vault Lock (POST lock-policy). Puoi ottenere lo stato del blocco della vault chiamando Get Vault Lock (GET lock-policy). Per ulteriori informazioni sul processo di blocco della vault, consulta Serratura Amazon Glacier Vault.

Questa operazione è idempotente. La richiesta ha sempre un esito positivo se lo stato del blocco di vault è `Locked` e l'ID di blocco fornito corrisponde all'ID di blocco utilizzato inizialmente per bloccare il vault.

Se un ID di blocco non valido viene passato nella richiesta quando lo stato del blocco di vault è `Locked`, l'operazione restituisce un errore `AccessDeniedException`. Se un ID di blocco non valido viene passato nella richiesta quando lo stato del blocco della vault è `InProgress`, l'operazione genera un errore `InvalidParameter`.

Richieste

Per completare il processo di blocco di vault, invia una richiesta HTTP `POST` all'URI della sottorisorsa `lock-policy` del vault con un ID di blocco valido.

Sintassi

```
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**
>
> Il valore è l'ID. `AccountId` Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Il valore `lockId` è l'ID di blocco ottenuto da una richiesta Initiate Vault Lock (POST lock-policy).

Parametri della richiesta

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Se la richiesta riesce, il servizio restituisce una risposta HTTP `204 No Content`.

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP POST con l'ID di blocco per completare il processo di blocco di vault.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 204 No Content` una risposta, come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- Abort Vault Lock (DELETE lock-policy)
- Get Vault Lock (GET lock-policy)
- Initiate Vault Lock (POST lock-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Delete Vault (DELETE vault)

Descrizione

Questa operazione elimina una vault. Amazon Glacier (Amazon Glacier) eliminerà un vault solo se non ci sono archivi nel vault come nell'ultimo inventario e non ci sono state scritture nel vault dall'ultimo inventario. Se una di queste condizioni non è soddisfatta, l'eliminazione del vault non riesce (ovvero, il vault non viene rimosso) e Amazon Glacier restituisce un errore.

Puoi utilizzare l'Describe Vault (GET vault)operazione che fornisce informazioni sul vault, incluso il numero di archivi nel vault; tuttavia, le informazioni si basano sull'ultimo inventario del vault generato da Amazon Glacier.

Questa operazione è idempotente.

> **Note**
>
> Quando elimini un vault, viene eliminata anche la policy di accesso a vault associata al vault. Per ulteriori informazioni sulle policy di accesso a vault, consulta Policy di accesso delle vault.

Richieste

Per eliminare un vault, invia una richiesta `DELETE` all'URI della risorsa vault.

Sintassi

```
DELETE /AccountId/vaults/VaultName HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente elimina un vault denominato `examplevault`. L'esempio di richiesta rappresenta una richiesta `DELETE` inviata all'URI della risorsa (il vault) da eliminare.

```
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
```

Risposta di esempio

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
```

```
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- Create Vault (PUT vault)
- List Vaults (GET vaults)
- Initiate Job (POST jobs)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Delete Vault Access Policy (DELETE access-policy)

Descrizione

Questa operazione consente di eliminare la policy di accesso associata al vault specificato. Alla fine l'operazione è coerente, ovvero potrebbe essere necessario del tempo prima che Amazon Glacier (Amazon Glacier) rimuova completamente la politica di accesso e potresti comunque vedere l'effetto della politica per un breve periodo dopo aver inviato la richiesta di eliminazione.

Questa operazione è idempotente. Puoi richiamare l'operazione di eliminazione più volte, anche se nessuna policy è associata al vault. Per ulteriori informazioni sulle policy di accesso a vault, consulta Policy di accesso delle vault.

Richieste

Per eliminare la policy di accesso a vault corrente, invia una richiesta HTTP `DELETE` all'URI della sottorisorsa `access-policy` del vault.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

In risposta, Amazon Glacier restituisce `204 No Content` se l'eliminazione della policy riesce.

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come eliminare una policy di accesso a vault.

Richiesta di esempio

In questo esempio, una richiesta `DELETE` è inviata alla sottorisorsa `access-policy` della vault denominata **`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
```

Risposta di esempio

In risposta, se la policy viene eliminata con successo, Amazon Glacier restituisce `204 No Content` un messaggio come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sezioni correlate

- Get Vault Access Policy (GET access-policy)

- Set Vault Access Policy (PUT access-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Delete Vault Notifications (DELETE notification-configuration)

Descrizione

Questa operazione elimina la configurazione delle notifiche impostata per un vault Set Vault Notification Configuration (PUT notification-configuration). Alla fine l'operazione è coerente, ovvero Amazon Glacier (Amazon Glacier) potrebbe impiegare del tempo per disabilitare completamente le notifiche e potresti continuare a ricevere alcune notifiche per un breve periodo dopo aver inviato la richiesta di eliminazione.

Richieste

Per eliminare la configurazione delle notifiche per un vault, invia una richiesta `DELETE` alla sottorisorsa `notification-configuration` del vault.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come rimuovere la configurazione delle notifiche per un vault.

Richiesta di esempio

In questo esempio, una richiesta `DELETE` è inviata alla sottorisorsa `notification-configuration` del vault denominato `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
```

Risposta di esempio

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sezioni correlate

- Get Vault Notifications (GET notification-configuration)
- Set Vault Notification Configuration (PUT notification-configuration)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Describe Vault (GET vault)

Descrizione

Questa operazione restituisce informazioni relative a un vault, tra cui l'ARN (Amazon Resource Name), la data di creazione, il numero di archivi che contiene e la dimensione totale di tutti quegli archivi. Il numero di archivi e la loro dimensione totale si riferiscono all'ultimo inventario del vault generato da Amazon Glacier (Amazon Glacier) (vedi). Lavorare con Vaults in Amazon Glacier Amazon Glacier genera inventari di vault all'incirca ogni giorno. Ciò significa che se aggiungi o

rimuovi un archivio in un vault e quindi invii immediatamente una richiesta Describe Vault, è possibile che la risposta non rifletta le modifiche.

Richieste

Per ottenere informazioni su un vault, invia una richiesta `GET` all'URI della risorsa vault specifica.

Sintassi

```
GET /AccountId/vaults/VaultName HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

CreationDate

La data di creazione del vault in formato UTC.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

LastInventoryDate

La data UTC in cui Amazon Glacier ha completato l'ultimo inventario del vault. Per informazioni sulla generazione di un inventario per una vault, consulta Initiate Job (POST jobs).

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

NumberOfArchives

Il numero di archivi nel vault alla data dell'ultimo inventario vault. Questo campo restituirà null se un inventario non è ancora stato generato per il vault, ad esempio, se hai appena creato il vault.

Tipo: numero

SizeInBytes

La dimensione totale in byte degli archivi nel vault, incluso l'overhead di ogni archivio, alla data dell'ultimo inventario. Questo campo restituirà `null` se un inventario non è ancora stato generato per il vault, ad esempio, se hai appena creato il vault.

Tipo: numero

VaultARN

L'ARN della vault.

- Tipo: stringa

VaultName

Il nome di vault specificato al momento della creazione. Questo nome è incluso anche nell'ARN del vault.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente mostra come ottenere informazioni sul vault denominato `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
```

Risposta di esempio

```
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"
}
```

Sezioni correlate

- Create Vault (PUT vault)
- List Vaults (GET vaults)
- Delete Vault (DELETE vault)
- Initiate Job (POST jobs)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Get Vault Access Policy (GET access-policy)

Descrizione

Questa operazione recupera la sottorisorsa `access-policy` impostata nella vault. Per ulteriori informazioni sull'impostazione di questa sottorisorsa, consulta Set Vault Access Policy (PUT access-policy). Se non vi è alcuna policy di accesso impostata sul vault, l'operazione restituisce un errore `404 Not found`. Per ulteriori informazioni sulle policy di accesso a vault, consulta Policy di accesso delle vault.

Richieste

Per restituire la policy di accesso a vault corrente, invia una richiesta HTTP `GET` all'URI della sottorisorsa `access-policy` del vault.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

In risposta, Amazon Glacier (Amazon Glacier) restituisce la policy di accesso al vault in formato JSON nel corpo della risposta.

Sintassi

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length

{
  "Policy": "string"
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

Policy

La policy di accesso a vault in forma di stringa JSON che utilizza "\" come carattere di escape.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come ottenere una policy di accesso a vault.

Richiesta di esempio

In questo esempio, una richiesta `GET` è inviata all'URI di una sottorisorsa `access-policy` della vault.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier restituisce la policy di accesso al vault come stringa JSON nel corpo della risposta. La stringa JSON restituita utilizza "\" come carattere di escape, come illustrato negli esempi Set Vault Access Policy (PUT access-policy). Nell'esempio seguente la stringa JSON è invece restituita senza caratteri escape per una migliore leggibilità.

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

```
  "
}
```

Sezioni correlate

- Delete Vault Access Policy (DELETE access-policy)
- Set Vault Access Policy (PUT access-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Get Vault Lock (GET lock-policy)

Descrizione

Questa operazione consente di recuperare i seguenti attributi dalla sottorisorsa `lock-policy` impostata nel vault specificato:

- La policy di Vault Lock impostata nel vault.
- Lo stato del blocco di vault, ovvero `InProgess` o `Locked`.
- La scadenza dell'ID del blocco. L'ID di blocco è utilizzato per completare il processo di blocco di vault.
- Quando il blocco di vault è stato avviato ed è passato allo stato `InProgress`.

Lo stato `InProgress` di un blocco della vault viene attivato chiamando Initiate Vault Lock (POST lock-policy). Lo stato `Locked` di un blocco della vault viene attivato chiamando Complete Vault Lock (POST lockId). Puoi arrestare il processo di blocco della vault chiamando Abort Vault Lock (DELETE lock-policy) Per ulteriori informazioni sul processo di blocco della vault, consulta Serratura Amazon Glacier Vault.

Se nessuna policy di Vault Lock è impostata nel vault, l'operazione restituisce un errore `404 Not found`. Per ulteriori informazioni sulle policy di blocco della vault, consulta Policy di Vault Lock.

Richieste

Per restituire la policy di Vault Lock corrente e altri attributi, invia una richiesta HTTP `GET` all'URI della sottorisorsa `lock-policy` del vault come illustrato nell'esempio di sintassi seguente.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

In risposta, Amazon Glacier (Amazon Glacier) restituisce la policy di accesso al vault in formato JSON nel corpo della risposta.

Sintassi

```
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"
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

Policy

La policy di Vault Lock in forma di stringa JSON, che utilizza "\" come carattere di escape.

- Tipo: stringa

Stato

Lo stato del blocco di vault.

- Tipo: stringa

Valori validi: `InProgress|Locked`

ExpirationDate

La data e l'ora di scadenza (in formato UTC) dell'ID del blocco. Questo valore può essere `null` se lo stato del blocco di vault è `Locked`.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

CreationDate

La data e l'ora (in formato UTC) del passaggio del blocco di vault allo stato `InProgress`.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come ottenere una policy di Vault Lock.

Richiesta di esempio

In questo esempio, una richiesta `GET` è inviata all'URI di una sottorisorsa `lock-policy` della vault.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier restituisce la policy di accesso al vault come stringa JSON nel corpo della risposta. La stringa JSON restituita utilizza "\" come carattere di escape, come illustrato nella richiesta di esempio Initiate Vault Lock (POST lock-policy). Nell'esempio seguente la stringa JSON è invece restituita senza caratteri escape per una migliore leggibilità.

```
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"
}
```

Sezioni correlate

- Abort Vault Lock (DELETE lock-policy)
- Complete Vault Lock (POST lockId)
- Initiate Vault Lock (POST lock-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Get Vault Notifications (GET notification-configuration)

Descrizione

Questa operazione recupera la sottorisorsa `notification-configuration` impostata nel vault (vedi Set Vault Notification Configuration (PUT notification-configuration)). Se la configurazione delle notifiche per un vault non è impostata, l'operazione restituisce un errore `404 Not Found`. Per ulteriori informazioni sulle notifiche di vault, consulta Configurazione delle notifiche Vault in Amazon Glacier.

Richieste

Per recuperare le informazioni sulla configurazione delle notifiche, invia una richiesta `GET` all'URI della sottorisorsa `notification-configuration` di un vault.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
HTTP/1.1 200 OK
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Content-Type: application/json
Content-Length: length
{
  "Events": [
    String,
    ...
  ],
  "SNSTopic": String
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

Eventi

Un elenco di uno o più eventi per i quali Amazon Glacier (Amazon Glacier) invierà una notifica all'argomento Amazon SNS specificato. Per informazioni sugli eventi di vault per i quali è possibile configurare la pubblicazione di notifiche in un vault, consulta Set Vault Notification Configuration (PUT notification-configuration).

Tipo: matrice

SNSTopic

Il nome della risorsa Amazon (ARN) dell'argomento Amazon Simple Notification Service (Amazon SNS). Per ulteriori informazioni, consulta Nozioni di base su Amazon SNS nella Guida per gli sviluppatori di Amazon Simple Notification Service.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come recuperare la configurazione delle notifiche per un vault.

Richiesta di esempio

In questo esempio, una richiesta `GET` è inviata alla sottorisorsa `notification-configuration` di un vault.

```
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
```

Risposta di esempio

In una risposta corretta, il documento di configurazione della registrazione di controllo nel corpo della risposta è in formato JSON. In questo esempio, la configurazione mostra che le notifiche per

due eventi (`ArchiveRetrievalCompleted` e `InventoryRetrievalCompleted`) sono inviate all'argomento `arn:aws:sns:us-west-2:012345678901:mytopic` di Amazon SNS.

```
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"
}
```

Sezioni correlate

- Delete Vault Notifications (DELETE notification-configuration)
- Set Vault Notification Configuration (PUT notification-configuration)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Initiate Vault Lock (POST lock-policy)

Descrizione

Questa operazione avvia la procedura di blocco di vault effettuando le seguenti operazioni:

- Installazione di una policy di Vault Lock nel vault specificato.
- Impostazione dello stato del blocco di vault su `InProgress`.

- Restituzione dell'ID di blocco utilizzato per completare il processo di blocco di vault.

Puoi impostare una policy di Vault Lock per vault, la cui dimensione non deve essere superiore a 20 KB. Per ulteriori informazioni sulle policy di blocco della vault, consulta Policy di Vault Lock.

Devi completare il processo di blocco di vault entro 24 ore dall'attivazione dello stato `InProgress` del blocco. Al termine delle 24 ore, l'ID del blocco scade, il vault passa automaticamente dallo stato `InProgress` a un altro stato e la policy di Vault Lock viene rimossa dal vault. Chiama Complete Vault Lock (POST lockId) per completare il processo di blocco di vault impostando lo stato del blocco di vault su `Locked`.

> **Note**
>
> Quando lo stato di un blocco di vault è `Locked`, non puoi avviare un nuovo blocco di vault per il vault.

Puoi arrestare il processo di blocco della vault chiamando Abort Vault Lock (DELETE lock-policy) e ottenere lo stato del blocco della vault chiamando Get Vault Lock (GET lock-policy). Per ulteriori informazioni sul processo di blocco della vault, consulta Serratura Amazon Glacier Vault.

Se questa operazione viene chiamata quando lo stato del blocco di vault è `InProgress`, l'operazione restituisce un errore `AccessDeniedException`. Quando lo stato del blocco di vault è `InProgress`, devi dapprima chiamare Abort Vault Lock (DELETE lock-policy) per poter avviare una nuova policy di Vault Lock.

Richieste

Per avviare un processo di blocco di vault, invia una richiesta HTTP `POST` all'URI della sottorisorsa `lock-policy` del vault, come illustrato nell'esempio di sintassi seguente.

Sintassi

```
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**
>
> Il valore è l'ID. `AccountId` Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il corpo della richiesta contiene i campi JSON descritti di seguito.

Policy

La policy di Vault Lock in forma di stringa JSON, che utilizza "\" come carattere di escape.

Tipo: stringa

Campo obbligatorio: sì

Risposte

Amazon Glacier (Amazon Glacier) restituisce `HTTP 201 Created` una risposta, se la politica viene accettata.

Sintassi

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-lock-id: lockId
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`x-amz-lock-id`	L'ID di blocco utilizzato per completare il processo di blocco di vault. ▪Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP PUT all'URI della sottorisorsa `lock-policy` della vault. La stringa JSON `Policy` utilizza "\" come carattere di escape.

```
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\"}}}]}"}
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier restituisce `HTTP 201 Created` una risposta, come illustrato nell'esempio seguente.

```
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
```

Sezioni correlate

- Abort Vault Lock (DELETE lock-policy)
- Complete Vault Lock (POST lockId)
- Get Vault Lock (GET lock-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

List Tags For Vault (GET tags)

Questa operazione elenca tutti i tag associati a un vault. L'operazione restituisce una mappa vuota se non sono presenti tag. Per ulteriori informazioni sui tag, consulta Etichettare le risorse di Amazon Glacier.

Sintassi della richiesta

Per elencare i tag per un vault, invia una richiesta HTTP GET all'URI dei tag come illustrato nell'esempio di sintassi seguente.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Se l'operazione riesce, il servizio restituisce una risposta HTTP `200 OK`.

Sintassi della risposta

```
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"
      }
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

Tags

I tag associati al vault. Ogni tag consiste di una chiave e un valore.

Tipo: mappatura stringa a stringa

Campo obbligatorio: sì

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Esempio: elenco di tag per un vault

L'esempio seguente elenca i tag per un vault.

Richiesta di esempio

In questo esempio, una richiesta GET viene inviata per recuperare un elenco di tag dal vault specificato.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 200 OK` un file con un elenco di tag per il vault, come illustrato nell'esempio seguente.

```
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"
      }
}
```

Sezioni correlate

- Add Tags To Vault (POST tags add)

- Remove Tags From Vault (POST tags remove)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

List Vaults (GET vaults)

Descrizione

Questa operazione elenca tutti i vault appartenenti all'account dell'utente che effettua la richiesta. L'elenco restituito nella risposta è ordinato in base ai valori ASCII dei nomi di vault.

Per impostazione predefinita, questa operazione restituisce fino a 10 elementi per richiesta. Se sono presenti più vault da elencare, il campo `marker` nel corpo della risposta contiene l'ARN (Amazon Resource Name) del vault da cui continuare l'elenco con una nuova richiesta List Vaults. In caso contrario, il campo `marker` è `null`. Nella tua prossima richiesta List Vaults imposti il `marker` parametro sul valore che Amazon Glacier (Amazon Glacier) ha restituito nelle risposte alla tua precedente richiesta List Vaults. Puoi anche limitare il numero di vault restituiti nella risposta specificando il parametro `limit` nella richiesta.

Richieste

Per ottenere un elenco, invia una richiesta `GET` alla risorsa vaults.

Sintassi

```
GET /AccountId/vaults HTTP/1.1
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> Il valore è l'ID. `AccountId` Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione utilizza i seguenti parametri di richiesta.

Nome	Descrizione	Richiesto
`limit`	Il numero massimo di vault da restituire. Il limite di default è 10. Il numero di vault restituiti può essere inferiore al limite specificato, ma mai superiore. -Tipo: stringa Vincoli: il valore intero minimo è 1. Il valore intero massimo è 10.	No
`marker`	Una stringa utilizzata per la paginazione. `marker` specifica l'ARN dopo il quale deve iniziare l'elenco. (il vault specifica to da `marker` non è incluso nell'elenco restituito). Ottieni il valore `marker` da una precedente risposta di List Vaults. Devi includere `marker` solo se continui la paginazione dei risultati iniziata in una precedente richiesta List Vaults. Se per il contrassegno specifichi un valore vuoto (""), viene restituito un elenco di vault che inizia dal primo vault. -Tipo: stringa Vincoli: nessuno	No

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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
   },
   ...
  ]
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

CreationDate

La data di creazione dell'archivio in formato UTC.

Tipo: stringa. Una rappresentazione di stringa nel formato di data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

LastInventoryDate

La data dell'ultimo inventario vault in formato UTC. Questo campo può essere null se un inventario non è ancora stato generato per il vault, ad esempio, se hai appena creato il vault. Per informazioni sulla generazione di un inventario per una vault, consulta Initiate Job (POST jobs).

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

Marker

`vaultARN` rappresenta il punto da cui continuare la paginazione dei risultati. Puoi utilizzare il parametro `marker` in un'altra richiesta List Vaults per ottenere ulteriori vault nell'elenco. Se non vi sono altri vault, questo valore è `null`.

- Tipo: stringa

NumberOfArchives

Il numero di archivi nel vault alla data dell'ultimo inventario.

Tipo: numero

SizeInBytes

La dimensione totale in byte di tutti gli archivi nel vault, incluso l'overhead di ogni archivio, alla data dell'ultimo inventario.

Tipo: numero

VaultARN

L'ARN della vault.

- Tipo: stringa

VaultList

Un matrice di oggetti, con ogni oggetto che fornisce una descrizione di un vault.

Tipo: matrice

VaultName

Il nome del vault.

-Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Esempio: elenco di tutti i vault

L'esempio seguente elenca tutti i vault. Poiché i parametri `marker` e `limit` non sono specificati nella richiesta, vengono restituiti fino a 10 vault.

Richiesta di esempio

```
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
```

Risposta di esempio

Il parametro `Marker` è `null` a indicare che non ci sono altri vault da elencare.

```
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"
   }
  ]
}
```

Esempio: elenco parziale di vault

L'esempio seguente restituisce due vault a partire dal vault specificato dal parametro `marker`.

Richiesta di esempio

```
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
```

Risposta di esempio

Nell'elenco vengono restituiti due vault. Il parametro `Marker` contiene l'ARN del vault per continuare la paginazione in un'altra richiesta List Vaults.

```
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"
   }
  ]
}
```

Sezioni correlate

- Create Vault (PUT vault)
- Delete Vault (DELETE vault)

- Initiate Job (POST jobs)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Remove Tags From Vault (POST tags remove)

Questa operazione consente di rimuovere uno o più tag dal set di tag associato a un vault. Per ulteriori informazioni sui tag, consulta Etichettare le risorse di Amazon Glacier.

Questa operazione è idempotente. L'operazione riuscirà anche se non vi sono tag associati al vault.

Sintassi della richiesta

Per rimuovere tag da un vault, devi inviare una richiesta HTTP POST all'URI dei tag come illustrato nell'esempio di sintassi seguente.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un

Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Nome	Descrizione	Richiesto
`operation=remove`	Un singolo parametro di stringa di query `operation` con un valore `remove` per differenziarlo da Add Tags To Vault (POST tags add).	Sì

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il corpo della richiesta contiene i campi JSON descritti di seguito.

TagKeys

Un elenco delle chiavi di tag. Ogni tag corrispondente viene rimosso dal vault.

Tipo: matrice di stringhe

Vincolo di lunghezza: l'elenco deve contenere almeno 1 elemento e al massimo 10.

Campo obbligatorio: sì

Risposte

Se l'operazione viene riesce, il servizio invia una risposta HTTP `204 No Content` con un corpo HTTP vuoto.

Sintassi

```
HTTP/1.1 204 No Content
```

```
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP POST per rimuovere i tag specificati.

```
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"
   ]
}
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 204 No Content` un come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- Add Tags To Vault (POST tags add)
- List Tags For Vault (GET tags)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Set Vault Access Policy (PUT access-policy)

Descrizione

Questa operazione consente di configurare una policy di accesso per un vault e sovrascrive una policy esistente. Per configurare una policy di accesso a vault, invia una richiesta PUT alla sottorisorsa `access-policy` del vault. Puoi impostare una sola policy di accesso per vault, la cui dimensione non deve essere superiore a 20 KB. Per ulteriori informazioni sulle policy di accesso a vault, consulta Policy di accesso delle vault.

Richieste

Sintassi

Per impostare una policy di accesso a vault, invia una richiesta HTTP PUT all'URI della sottorisorsa `access-policy` del vault come illustrato nell'esempio di sintassi seguente.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il corpo della richiesta contiene i campi JSON descritti di seguito.

Policy

La policy di accesso a vault in forma di stringa JSON che utilizza "\" come carattere di escape.

Tipo: stringa

Campo obbligatorio: sì

Risposte

In risposta, Amazon Glacier `204 No Content` restituisce il prodotto se la politica viene accettata.

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP PUT all'URI della sottorisorsa `access-policy` della vault. La stringa JSON `Policy` utilizza "\" come carattere di escape.

```
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\"}]}"}
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 204 No Content` un come illustrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- Delete Vault Access Policy (DELETE access-policy)
- Get Vault Access Policy (GET access-policy)

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Set Vault Notification Configuration (PUT notification-configuration)

Descrizione

Il recupero di un archivio e di un inventario del vault sono operazioni asincrone in Amazon Glacier (Amazon Glacier), per le quali è necessario innanzitutto avviare un processo e attendere il completamento del processo prima di poter scaricare l'output del lavoro. Puoi quindi configurare una vault per pubblicare un messaggio su un argomento di Amazon Simple Notification Service (Amazon SNS) quando i processi sono completati. Questa operazione ti consente di impostare la

configurazione delle notifiche per il vault. Per ulteriori informazioni, consulta Configurazione delle notifiche Vault in Amazon Glacier.

Per configurare le notifiche per il vault, invia una richiesta PUT alla sottorisorsa `notification-configuration` del vault. Una configurazione delle notifiche è specifica di un vault, di conseguenza è definita anche sottorisorsa di vault. La richiesta deve includere un documento JSON che fornisca un argomento di Amazon Simple Notification Service (Amazon SNS) e gli eventi per i quali desideri che Amazon Glacier invii notifiche all'argomento.

Puoi configurare un vault affinché pubblichi una notifica per i seguenti eventi di vault:

- **`ArchiveRetrievalCompleted`**: questo evento si verifica al completamento di un processo avviato per il recupero di un archivio (Initiate Job (POST jobs)). Lo stato del processo completato può essere `Succeeded` o `Failed`. La notifica inviata all'argomento di SNS è identica all'output restituito da Describe Job (GET JobID).
- **`InventoryRetrievalCompleted`** – Questo evento si verifica al completamento di un processo avviato per il recupero di un inventario (Initiate Job (POST jobs)). Lo stato del processo completato può essere `Succeeded` o `Failed`. La notifica inviata all'argomento di SNS è identica all'output restituito da Describe Job (GET JobID).

Gli argomenti di Amazon SNS devono concedere alla vault l'autorizzazione per pubblicare notifiche nell'argomento.

Richieste

Per impostare la configurazione delle notifiche per il vault, invia una richiesta PUT all'URI della sottorisorsa `notification-configuration` del vault. La configurazione deve essere specificata nel corpo della richiesta. La configurazione include il nome dell'argomento di Amazon SNS e una matrice di eventi che attivano l'invio della notifica a ogni argomento.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il codice JSON nel corpo della richiesta contiene i campi descritti di seguito.

Eventi

Una serie di uno o più eventi per i quali desideri che Amazon Glacier invii una notifica.

Valori validi: `ArchiveRetrievalCompleted` | `InventoryRetrievalCompleted`

Obbligatorio: sì

Tipo: matrice

SNSTopic

L'ARN dell'argomento Amazon SNS. Per ulteriori informazioni, consulta Nozioni di base su Amazon SNS nella Guida per gli sviluppatori di Amazon Simple Notification Service.

Obbligatorio: sì

- Tipo: stringa

Risposte

In risposta, Amazon Glacier (Amazon Glacier) `204 No Content` restituisce se la configurazione di notifica è stata accettata.

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come configurare una notifica di vault.

Richiesta di esempio

La richiesta seguente imposta la configurazione delle notifiche `examplevault` di modo che le notifiche per due eventi (`ArchiveRetrievalCompleted` e `InventoryRetrievalCompleted`) siano inviate all'argomento `arn:aws:sns:us-west-2:012345678901:mytopic` di Amazon SNS.

```
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"
}
```

Risposta di esempio

Una risposta corretta restituisce `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
```

Sezioni correlate

- Get Vault Notifications (GET notification-configuration)
- Delete Vault Notifications (DELETE notification-configuration)
- Identity and Access Management per Amazon Glacier

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in una delle lingue SDKs Amazon specifiche per lingua, consulta quanto segue:

- AWS Command Line Interface

Operazioni di archivio

Di seguito sono elencate le operazioni di archiviazione disponibili per l'uso in Amazon Glacier.

Argomenti

- Delete Archive (DELETE archive)
- Upload Archive (POST archive)

Delete Archive (DELETE archive)

Descrizione

Questa operazione elimina un archivio da un vault. Puoi eliminare un archivio alla volta da una vault. Per eliminare un archivio, devi indicare il relativo ID nella richiesta di eliminazione. Per ottenere l'ID archivio, scarica l'inventario vault per la vault contenente l'archivio. Per ulteriori informazioni sul download di un inventario vault, consulta Scaricamento di un inventario Vault in Amazon Glacier.

Dopo l'eliminazione di un archivio, puoi ancora eseguire una richiesta di avvio di un processo per recuperare l'archivio eliminato, ma il processo di recupero non riuscirà.

Le operazioni di recupero di archivio che sono in corso per un ID archivio nel momento in cui elimini l'archivio possono riuscire o meno a seconda dei seguenti scenari:

- Se il processo di recupero dell'archivio sta preparando attivamente i dati per il download quando Amazon Glacier (Amazon Glacier) riceve la richiesta di eliminazione dell'archivio, l'operazione di recupero dell'archivio potrebbe non riuscire.
- Se il processo di recupero dell'archivio ha preparato correttamente l'archivio per il download quando Amazon Glacier riceve la richiesta di eliminazione dell'archivio, potrai scaricare l'output.

Per ulteriori informazioni sul recupero di archivi, consulta Scaricare un archivio in Amazon Glacier.

Questa operazione è idempotente. Il tentativo di eliminazione di un archivio già eliminato non genera un errore.

Richieste

Per eliminare un archivio, devi inviare una richiesta `DELETE` all'URI della risorsa dell'archivio.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come eliminare un archivio dal vault denominato `examplevault`.

Richiesta di esempio

L'ID dell'archivio da eliminare è specificato come sottorisorsa di `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
```

Risposta di esempio

Se la richiesta ha esito positivo, Amazon Glacier risponde indicando che l'`204 No Content`archivio è stato eliminato.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Archive (POST archive)
- Identity and Access Management per Amazon Glacier

Upload Archive (POST archive)

Descrizione

Questa operazione aggiunge un archivio a un vault. Affinché un caricamento venga eseguito correttamente, i dati sono resi persistenti in modo durevole. In risposta, Amazon Glacier (Amazon

Glacier) restituisce l'ID dell'archivio nell'intestazione della risposta. `x-amz-archive-id` Salva questo ID archivio per poter accedere all'archivio in seguito.

Devi fornire un hash ad SHA256 albero dei dati che stai caricando. Per informazioni sul calcolo dell'hash di un SHA256 albero, consulta. Calcolo di checksum

> **Note**
>
> L'hash SHA256 dell'albero è richiesto solo per l'azione Upload Archive (archivio POST) quando si utilizza l'API. Non è necessario quando si utilizza. AWS CLI

Quando carichi un archivio, puoi eventualmente specificare una descrizione che può contenere fino a 1.024 caratteri ASCII stampabili. Amazon Glacier restituisce la descrizione dell'archivio quando recuperi l'archivio o ottieni l'inventario del vault. Amazon Glacier non interpreta in alcun modo la descrizione. Una descrizione di archivio non deve essere univoca e non può essere utilizzata per recuperare o ordinare l'elenco di archivi.

Ad eccezione della descrizione opzionale dell'archivio, Amazon Glacier non supporta metadati aggiuntivi per gli archivi. L'ID archivio è una sequenza di caratteri opaca da cui non è possibile dedurre alcuna informazione relativa all'archivio. Di conseguenza, puoi conservare i metadati sugli archivi sul lato client. Per ulteriori informazioni, consulta Lavorare con gli archivi in Amazon Glacier.

Gli archivi non sono modificabili. Dopo il caricamento di un archivio, non è possibile modificarne il contenuto o la descrizione.

Richieste

Per caricare un archivio, utilizzi il metodo HTTP POST e definisci l'ambito della richiesta per la sottorisorsa `archives` del vault in cui desideri salvare l'archivio. La richiesta deve includere la dimensione del payload dell'archivio, il checksum (SHA256 tree hash) e può facoltativamente includere una descrizione dell'archivio.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa implementazione dell'operazione non utilizza parametri di richiesta.

Intestazioni di richiesta

Questa operazione utilizza le seguenti intestazioni di richiesta, oltre alle intestazioni di richiesta comuni a tutte le operazioni. Per ulteriori informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Nome	Descrizione	Richiesto
`Content-Length`	La dimensione dell'oggetto in byte. Per ulteriori informazioni, vai a http://www.w3. org/Protocols/rfc2616/rfc2616-sec14. html #sec14 .13. Tipo: numero Impostazione predefinita: nessuna Vincoli: nessuno	Sì
`x-amz-archive-description`	La descrizione facoltativa dell'archivio che stai caricando . Può trattarsi di una descrizione in linguaggio normale o	No

Nome	Descrizione	Richiesto
	di un identificatore che scegli di assegnare. La descrizione di archivio non deve essere univoca. Quando recuperi un inventario vault (vedi Initiate Job (POST jobs)), l'inventario include questa descrizione per ogni archivio che restituisce nella risposta. -Tipo: stringa Impostazione predefinita: nessuna Vincoli: la descrizione non deve contenere più di 1.024 caratteri. I caratteri consentiti sono caratteri ASCII a 7 bit senza codici di controllo, in particolare valori ASCII da 32 a 126 in decimale o da 0x20 a 0x7E in esadecimale.	
`x-amz-content-sha256`	Il SHA256 checksum (un hash lineare) del payload. Non si tratta dello stesso valore che specifichi nell'intestazione `x-amz-sha256-tree-hash` . -Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì
`x-amz-sha256-tree-hash`	Il checksum, l'hash dell' SHA256 albero, calcolato dall'uten te del payload. Per informazioni sul calcolo dell'hash dell' SHA256 albero, vedere. Calcolo di checksum Se Amazon Glacier calcola un checksum diverso del payload, rifiuterà la richiesta. -Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì

Corpo della richiesta

Il corpo della richiesta contiene i dati da caricare.

Risposte

In risposta, Amazon Glacier archivia l'archivio in modo duraturo e restituisce un percorso URI all'ID dell'archivio.

Sintassi

```
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
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`Location`	Il percorso URI relativo della risorsa di archivio aggiunta. ▪Tipo: stringa
`x-amz-archive-id`	L'ID dell'archivio. Questo valore è incluso anche nell'intestazione `Location`. ▪Tipo: stringa
`x-amz-sha256-tree-hash`	Il checksum dell'archivio calcolato da Amazon Glacier. ▪Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente mostra una richiesta per il caricamento di un archivio.

```
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).>
```

Risposta di esempio

La risposta corretta riportata di seguito presenta un'`Location`intestazione in cui è possibile ottenere l'ID assegnato da Amazon Glacier all'archivio.

```
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
```

Sezioni correlate

- Lavorare con gli archivi in Amazon Glacier
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Delete Archive (DELETE archive)
- Identity and Access Management per Amazon Glacier

Operazioni di caricamento in più parti

Di seguito sono elencate le operazioni di caricamento in più parti disponibili per l'uso in Amazon Glacier.

Argomenti

- Abort Multipart Upload (DELETE uploadID)
- Complete Multipart Upload (POST uploadID)
- Initiate Multipart Upload (POST multipart-uploads)
- List Parts (GET uploadID)
- List Multipart Uploads (GET multipart-uploads)
- Upload Part (PUT uploadID)

Abort Multipart Upload (DELETE uploadID)

Descrizione

Questo comando di caricamento in più parti arresta un caricamento in più parti identificato dall'ID di caricamento.

Se la richiesta Abort Multipart Upload viene eseguita correttamente, non puoi utilizzare l'ID di caricamento per caricare altre parti o eseguire altre operazioni. L'arresto di un caricamento in più parti completato non è consentito. Tuttavia, l'arresto di un caricamento già interrotto riesce per un breve periodo di tempo.

Questa operazione è idempotente.

Per informazioni sul caricamento in più parti, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Richieste

Per arrestare un caricamento in più parti, invia una richiesta `DELETE` HTTP all'URI della sottorisorsa `multipart-uploads` della vault e identifica l'ID di caricamento in più parti come parte dell'URI.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
HTTP/1.1 204 No Content
```

```
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempio

Richiesta di esempio

Nell'esempio seguente, una richiesta DELETE è inviata all'URI di una risorsa di ID di caricamento in più parti.

```
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
```

Risposta di esempio

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:00:00 GMT
```

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- List Multipart Uploads (GET multipart-uploads)
- List Parts (GET uploadID)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Identity and Access Management per Amazon Glacier

Complete Multipart Upload (POST uploadID)

Descrizione

Si chiama operazione di caricamento in più parti per informare Amazon Glacier (Amazon Glacier) che tutte le parti dell'archivio sono state caricate e Amazon Glacier può ora assemblare l'archivio dalle parti caricate.

Per informazioni sul caricamento in più parti, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Dopo aver assemblato e salvato l'archivio nel vault, Amazon Glacier restituisce l'ID di archivio della risorsa di archivio appena creata. Dopo il caricamento di un archivio, è consigliabile salvare l'ID archivio restituito allo scopo di recuperare l'archivio successivamente.

Nella richiesta, devi includere l'hash dell' SHA256 albero calcolato dell'intero archivio che hai caricato. Per informazioni sul calcolo dell'hash di un SHA256 albero, consulta. Calcolo di checksum Sul lato server, Amazon Glacier crea anche l'hash ad albero SHA256 dell'archivio assemblato. Se i valori corrispondono, Amazon Glacier salva l'archivio nel vault; in caso contrario, restituisce un errore e l'operazione ha esito negativo. L'operazione List Parts (GET uploadID) restituisce un elenco delle parti caricate per uno specifico caricamento in più parti. Include le informazioni sul checksum per ogni parte caricata che può essere utilizzata per il debug di un errore di checksum.

Inoltre, Amazon Glacier verifica anche eventuali intervalli di contenuti mancanti. Durante il caricamento delle parti, devi specificare i valori di intervallo che identificano la posizione di ogni parte nell'assemblaggio finale dell'archivio. Durante l'assemblaggio dell'archivio finale, Amazon Glacier verifica la presenza di eventuali intervalli di contenuti mancanti e, se sono presenti, Amazon Glacier restituisce un errore e l'operazione Complete Multipart Upload non riesce.

L'operazione Complete Multipart Upload è idempotente. Dopo il primo completamento riuscito di un caricamento in più parti, se chiami di nuovo l'operazione dopo un breve periodo, l'operazione riesce e restituisce lo stesso ID archivio. Ciò è utile in caso di un problema di rete o di un errore di server 500, in quanto puoi ripetere la richiesta Complete Multipart Upload e ottenere lo stesso ID archivio senza creare archivi duplicati. Nota, tuttavia, che dopo il completamento del caricamento in più parti, non puoi chiamare l'operazione List Parts e il caricamento in più parti non apparirà nella risposta List Multipart Uploads, anche se un'operazione idempotente è possibile.

Richieste

Per completare un caricamento in più parti, invii una richiesta HTTP POST all'URI dell'ID di caricamento creato da Amazon Glacier in risposta alla tua richiesta Initiate Multipart Upload. Si tratta dello stesso URI che hai utilizzato durante il caricamento delle parti. Oltre alle intestazioni comuni richieste, devi includere il risultato dell'hash ad SHA256 albero dell'intero archivio e la dimensione totale dell'archivio in byte.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza le seguenti intestazioni di richiesta, oltre alle intestazioni di richiesta comuni a tutte le operazioni. Per ulteriori informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Nome	Descrizione	Richiesto
`x-amz-archive-size`	La dimensione totale in byte dell'intero archivio. Questo valore deve essere la somma di tutte le dimensioni delle singole parti caricate. ▪Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì
`x-amz-sha256-tree-hash`	L'hash ad SHA256 albero dell'intero archivio. È l'hash dell'albero dell' SHA256 albero dell'hash delle singole parti. Se il valore specificato nella richiesta non corrisponde all'hash dell' SHA256 albero dell'archivio assemblato finale calcolato da Amazon Glacier, Amazon Glacier restituisce un errore e la richiesta ha esito negativo. ▪Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì

Elementi della richiesta

Questa operazione non utilizza elementi di richiesta.

Risposte

Amazon Glacier (Amazon Glacier) crea SHA256 un hash ad albero dell'intero archivio. Se il valore corrisponde all'hash dell' SHA256albero dell'intero archivio specificato nella richiesta, Amazon Glacier

aggiunge l'archivio al vault. In risposta, restituisce l'intestazione HTTP `Location` con il percorso URL della risorsa di archivio appena aggiunta. Se la dimensione dell'archivio o SHA256 quella che hai inviato nella richiesta non corrisponde, Amazon Glacier restituirà un errore e il caricamento rimane incompleto. È possibile riprovare l'operazione Complete Multipart Upload successivamente con valori corretti e riuscire a creare un archivio. Se un caricamento in più parti non viene completato, alla fine Amazon Glacier recupererà l'ID di caricamento.

Sintassi

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
x-amz-archive-id: ArchiveId
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`Location`	Il percorso URI relativo dell'archivio creato. Questo URL include l'ID di archivio generato da Amazon Glacier. ▪Tipo: stringa
`x-amz-archive-id`	L'ID dell'archivio. Questo valore è incluso anche nell'intestazione `Location`. ▪Tipo: stringa

Campi di risposta

Questa operazione non restituisce un corpo della risposta.

Esempio

Richiesta di esempio

In questo esempio, la richiesta HTTP POST è inviata all'URI restituito da una richiesta Initiate Multipart Upload. La richiesta specifica sia l'hash dell' SHA256 albero dell'intero archivio che la dimensione totale dell'archivio.

```
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
```

Risposta di esempio

Il seguente esempio di risposta mostra che Amazon Glacier ha creato con successo un archivio dalle parti che hai caricato. La risposta include l'ID archivio con il percorso completo.

```
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
```

A questo punto puoi inviare le richieste HTTP all'URI della risorsa o dell'archivio aggiunto. Ad esempio, puoi inviare una richiesta GET per recuperare l'archivio.

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)

- Abort Multipart Upload (DELETE uploadID)
- List Multipart Uploads (GET multipart-uploads)
- List Parts (GET uploadID)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Delete Archive (DELETE archive)
- Identity and Access Management per Amazon Glacier

Initiate Multipart Upload (POST multipart-uploads)

Descrizione

Questa operazione avvia una caricamento in più parti (consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)). Amazon Glacier (Amazon Glacier) crea una risorsa di caricamento multiparte e ne restituisce l'ID nella risposta. Questo ID può essere utilizzato nelle operazioni di caricamento in più parti successive.

Quando avvii un caricamento in più parti, specifichi la dimensione di parte in byte. La dimensione della parte deve essere un mebibyte (MiB) (1024 kibibyte [KiB]) moltiplicato per una potenza di 2, ad esempio 1048576 (1 MiB), 2097152 (2 MiB), 4194304 (4 MiB), 8388608 (8 MiB) e così via. La dimensione di parte minima autorizzata è 1 MiB, mentre quella massima è 4 gibibyte (GiB).

Ogni parte che carichi utilizzando questo ID di caricamento, tranne l'ultima, deve avere la stessa dimensione. L'ultimo parte può avere la stessa dimensione o una dimensione inferiore. Ad esempio, supponi di voler caricare un file di 16,2 MiB. Se avvii il caricamento in più parti con una dimensione di parte di 4 MiB, caricherai quattro parti di 4 MiB ciascuna e una parte di 0,2 MiB.

> **Note**
>
> Non è necessario conoscere la dimensione dell'archivio quando si avvia un caricamento in più parti, poiché Amazon Glacier non richiede di specificare la dimensione complessiva dell'archivio.

Dopo aver completato il caricamento multiparte, Amazon Glacier rimuove la risorsa di caricamento multiparte a cui fa riferimento l'ID. Amazon Glacier rimuoverà anche la risorsa di caricamento multiparte se annulli il caricamento multiparte o potrebbe essere rimossa se non c'è attività per un

periodo di 24 ore. È possibile che l'ID sia ancora disponibile dopo 24 ore, ma è consigliabile non affidarsi a questa eventualità.

Richieste

Per avviare un caricamento in più parti, devi inviare una richiesta HTTP POST all'URI della sottorisorsa `multipart-uploads` del vault in cui desideri salvare l'archivio. La richiesta deve includere la dimensione di parte ed eventualmente una descrizione dell'archivio.

Sintassi

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza le seguenti intestazioni di richiesta, oltre alle intestazioni di richiesta comuni a tutte le operazioni. Per ulteriori informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Nome	Descrizione	Richiesto
`x-amz-part-size`		Sì

Nome	Descrizione	Richiesto
	La dimensione in byte di ogni parte eccetto l'ultima. La dimensione dell'ultima parte può essere inferiore a questa dimensione. • Tipo: stringa Impostazione predefinita: nessuna Vincoli: la dimensione della parte deve essere un mebibyte (1024 KiB) moltiplicato per una potenza di 2, ad esempio 1048576 (1 MiB), 2097152 (2 MiB), 4194304 (4 MiB), 8388608 (8 MiB) e così via. La dimensione di parte minima autorizzata è 1 MiB, mentre quella massima è 4 gibibyte (4096 MiB).	
`x-amz-archive-description`	La descrizione dell'archivio di cui stai eseguendo il caricamento in parti. Può essere una descrizione in linguaggio normale o un identificatore univoco che scegli di assegnare . Quando recuperi un inventario vault (vedi Initiate Job (POST jobs)), l'inventario include questa descrizione per ogni archivio che restituisce nella risposta. Gli spazi iniziali nelle descrizioni degli archivi vengono rimossi. • Tipo: stringa Impostazione predefinita: nessuna Vincoli: la descrizione deve essere minore o uguale a 1024 byte. I caratteri consentiti sono caratteri ASCII a 7 bit senza codici di controllo , in particolare valori ASCII da 32 a 126 in decimale o da 0x20 a 0x7E in esadecimale.	No

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Nella risposta, Amazon Glacier crea una risorsa di caricamento multiparte identificata da un ID e restituisce il percorso URI relativo dell'ID di caricamento multiparte.

Sintassi

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
Location: Location
x-amz-multipart-upload-id: multiPartUploadId
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`Location`	Il percorso URI relativo dell'ID di caricamento multiparte creato da Amazon Glacier. Questo percorso URI viene utilizzato per definire l'ambito delle tue richieste di caricamento di parti e per completare il caricamento in più parti. ▪Tipo: stringa
`x-amz-multipart-upload-id`	L'ID del caricamento in più parti. Questo valore è incluso anche nell'intestazione `Location`. ▪Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempio

Richiesta di esempio

L'esempio seguente avvia un caricamento in più parti inviando una richiesta HTTP POST all'URI della sottorisorsa `multipart-uploads` di un vault denominato `examplevault`. La richiesta include intestazioni che consentono di specificare una dimensione di parte di 4 MiB (4194304 byte) e la descrizione facoltativa dell'archivio.

```
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
```

Risposta di esempio

Amazon Glacier crea una risorsa di caricamento multiparte e la aggiunge alla sottorisorsa `multipart-uploads` del vault. L'intestazione di risposta `Location` include il percorso URI relativo all'ID di caricamento in più parti.

```
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
```

Per ulteriori informazioni sul caricamento di singole parti, consulta Upload Part (PUT uploadID).

Sezioni correlate

- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- Abort Multipart Upload (DELETE uploadID)
- List Multipart Uploads (GET multipart-uploads)
- List Parts (GET uploadID)
- Delete Archive (DELETE archive)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Identity and Access Management per Amazon Glacier

List Parts (GET uploadID)

Descrizione

Questa operazione di caricamento in più parti elenca le parti di un archivio che sono state caricate mediante un determinato caricamento in più parti identificato da un ID di caricamento. Per informazioni sul caricamento in più parti, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Puoi effettuare questa richiesta in qualsiasi momento durante un caricamento in più parti prima che venga completato. Amazon Glacier restituisce l'elenco di parti ordinato per intervallo specificato in ogni caricamento di parte. Se invii una richiesta List Parts dopo aver completato il caricamento multiparte, Amazon Glacier (Amazon Glacier) restituisce un errore.

L'operazione List Parts supporta la paginazione. Devi sempre verificare se il campo `Marker` nel corpo della risposta include un contrassegno che segnala il punto dal quale continuare l'elenco. Se non vi sono altri elementi, il campo `marker` è `null`. Se non `marker` è nullo, per recuperare il prossimo set di parti hai inviato un'altra richiesta List Parts con il parametro `marker` request impostato sul valore marker restituito da Amazon Glacier in risposta alla tua precedente richiesta List Parts.

Puoi inoltre limitare il numero di parti restituite nella risposta specificando il parametro `limit` nella richiesta.

Richieste

Sintassi

Per elencare le parti di un caricamento in più parti in corso, devi inviare una richiesta `GET` all'URI della risorsa ID di caricamento in più parti. L'ID di caricamento in più parti viene restituito quando avvii un caricamento in più parti (Initiate Multipart Upload (POST multipart-uploads)). Puoi eventualmente specificare i parametri `marker` e `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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Nome	Descrizione	Richiesto
`limit`	Il numero massimo di parti da restituire. Il limite di default è 50. Il numero di parti restituite può essere inferiore al limite specificato, ma mai superiore. ▪Tipo: stringa Vincoli: il valore intero minimo è 1. Il valore intero massimo è 50.	No
`marker`		No

Nome	Descrizione	Richiesto
	Una stringa opaca utilizzata per la paginazione. Il parametro `marker` specifica la parte da cui deve cominciare l'elenco di parti. Ottieni il valore `marker` da una risposta List Parts precedente. Devi includere `marker` solo se continui la paginazione dei risultati iniziata in una richiesta List Parts precedente. -Tipo: stringa Vincoli: nessuno	

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

ArchiveDescription

La descrizione dell'archivio specificata nella richiesta Initiate Multipart Upload. Questo campo è `null` se la descrizione dell'archivio non è stata specificata nell'operazione Initiate Multipart Upload.

- Tipo: stringa

CreationDate

La data e l'ora in formato UTC in cui il caricamento in più parti è stato avviato.

Tipo: stringa. Una rappresentazione di stringa nel formato di data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

Marker

Una stringa opaca che rappresenta il punto da cui continuare la paginazione dei risultati. Utilizza l'elenco `marker` in una nuova richiesta List Parts per ottenere ulteriori processi nell'elenco. Se non vi sono altre parti, questo valore è `null`.

- Tipo: stringa

MultipartUploadId

L'ID del caricamento a cui sono associate le parti.

- Tipo: stringa

PartSizeInBytes

La dimensione di parte in byte. Si tratta dello stesso valore specificato nella richiesta Initiate Multipart Upload.

Tipo: numero

Parti

Un elenco di dimensioni di parte del caricamento in più parti. Ogni oggetto nella matrice contiene una coppia nome/valore `RangeBytes` e `sha256-tree-hash`.

Tipo: matrice

RangeInBytes

L'intervallo di byte di una parte, compreso il valore superiore dell'intervallo.

- Tipo: stringa

SHA256TreeHash

Il valore hash SHA256 dell'albero calcolato da Amazon Glacier per la parte. Questo campo non è mai `null`.

- Tipo: stringa

VaultARN

L'ARN (Amazon Resource Name) del vault in cui è stato avviato il caricamento in più parti.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Esempio: elenco di parti di un caricamento in più parti

L'esempio seguente elenca tutte le parti di un caricamento. L'esempio invia una richiesta HTTP `GET` all'URI dell'ID di un caricamento in più parti in corso e restituisce fino a 1.000 parti.

Richiesta di esempio

```
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
```

Risposta di esempio

Nella risposta, Amazon Glacier restituisce un elenco di parti caricate associate all'ID di caricamento multiparte specificato. In questo esempio, ci sono solo due parti. Il campo `Marker` restituito è `null` a indicare che non ci sono ulteriori parti del caricamento in più parti.

```
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"
```

```
}
```

Esempio: elenco di parti di un caricamento in più parti (specificare i parametri di richiesta di contrassegno e limite)

L'esempio seguente mostra come utilizzare la paginazione per ottenere un numero limitato di risultati. L'esempio invia una richiesta HTTP `GET` all'URI dell'ID di un caricamento in più parti in corso per restituire una parte. Un parametro `marker` iniziale specifica la parte da cui cominciare l'elenco. Puoi ottenere il valore `marker` da una risposta di una richiesta precedente per un elenco di parti. Inoltre, in questo esempio, il parametro `limit` è impostato su 1 e restituisce una parte. Nota che il campo `Marker` non è `null`, a indicare che è presente almeno un'altra parte da ottenere.

Richiesta di esempio

```
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
```

Risposta di esempio

Nella risposta, Amazon Glacier restituisce un elenco di parti caricate associate all'ID di caricamento multiparte in corso specificato.

```
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"
}
```

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- Abort Multipart Upload (DELETE uploadID)
- List Multipart Uploads (GET multipart-uploads)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Identity and Access Management per Amazon Glacier

List Multipart Uploads (GET multipart-uploads)

Descrizione

Questa operazione di caricamento in più parti elenca i caricamenti di questo tipo in corso per il vault specificato. Un caricamento in più parti in corso è un caricamento che è stato avviato da una richiesta Initiate Multipart Upload (POST multipart-uploads), ma che non è ancora stato completato o arrestato. L'elenco restituito nella risposta List Multipart Uploads non è ordinato.

L'operazione List Multipart Uploads supporta la paginazione. Per impostazione predefinita, questa operazione restituisce fino a 50 caricamenti in più parti nella risposta. Devi sempre verificare se il campo `marker` nel corpo della risposta include un contrassegno che segnala il punto dal quale continuare l'elenco; se non vi sono altri elementi, il campo `marker` è `null`.

Se non `marker` è nullo, per recuperare il set successivo di caricamenti multiparte hai inviato un'altra richiesta List Multipart Uploads con il parametro `marker` request impostato sul valore marker che Amazon Glacier (Amazon Glacier) aveva restituito in risposta alla precedente richiesta List Multipart Uploads.

Nota la differenza tra questa operazione e l'operazione List Parts (GET uploadID). L'operazione List Multipart Uploads elenca tutti i caricamenti in più parti per un vault. L'operazione List Parts restituisce le parti di un determinato caricamento in più parti identificato da un ID di caricamento.

Per informazioni sul caricamento in più parti, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Richieste

Sintassi

Per elencare i caricamenti in più parti, invia una richiesta `GET` all'URI della sottorisorsa `multipart-uploads` del vault. Puoi eventualmente specificare i parametri `marker` e `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**
>
> Il valore è l'ID dell'account proprietario del vault. `AccountId` Account AWS Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Nome	Descrizione	Richiesto
`limit`	Specifica Il numero massimo di caricamenti restituiti nel corpo della risposta. Se non è specificato, l'operazione List Uploads restituisce fino a 50 caricamenti. ▪Tipo: stringa	No

Nome	Descrizione	Richiesto
	Vincoli: il valore intero minimo è 1. Il valore intero massimo è 50.	
marker	Una stringa opaca utilizzata per la paginazione. Il parametro marker specifica il caricamento da cui deve iniziare l'elenco di caricamenti. Ottieni il valore marker da una risposta List Uploads precedente. Devi includere marker solo se continui la paginazione dei risultati iniziata in una richiesta List Uploads precedente. -Tipo: stringa Vincoli: nessuno	No

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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
    },
   ...
  ]
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

ArchiveDescription

La descrizione dell'archivio specificata nella richiesta Initiate Multipart Upload. Questo campo è `null` se la descrizione dell'archivio non è stata specificata nell'operazione Initiate Multipart Upload.

- Tipo: stringa

CreationDate

La data e l'ora in formato UTC in cui il caricamento in più parti è stato avviato.

Tipo: stringa. Una rappresentazione di stringa nel formato di data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

Marker

Una stringa opaca che rappresenta il punto da cui continuare la paginazione dei risultati. Utilizza l'elenco `marker` in una nuova richiesta List Multipart Uploads per ottenere ulteriori caricamenti nell'elenco. Se non vi sono altri caricamenti, questo valore è `null`.

- Tipo: stringa

PartSizeInBytes

La dimensione di parte specificata nella richiesta Initiate Multipart Upload (POST multipart-uploads). Si tratta della dimensione di tutte le parti nel caricamento tranne l'ultima parte, che può essere più piccola di questa dimensione.

Tipo: numero

MultipartUploadId

L'ID del caricamento in più parti.

- Tipo: stringa

UploadsList

Un elenco di metadati relativi agli oggetti del caricamento in più parti. Ogni elemento nell'elenco contiene un set di coppie nome-valore per il caricamento corrispondente, come `ArchiveDescription`, `CreationDate`, `MultipartUploadId`, `PartSizeInBytes` e `VaultARN`.

Tipo: matrice

VaultARN

L'ARN (Amazon Resource Name) del vault che contiene l'archivio.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Esempio: elenco di tutti i caricamenti in più parti

L'esempio seguente elenca tutti i caricamenti in più parti in corso per il vault. L'esempio mostra una richiesta HTTP `GET` inviata all'URI della sottorisorsa `multipart-uploads` di un determinato vault. Poiché i parametri `marker` e `limit` non sono specificati nella richiesta, vengono restituiti fino a 1.000 caricamenti in più parti.

Richiesta di esempio

```
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
```

Risposta di esempio

Nella risposta Amazon Glacier restituisce un elenco di tutti i caricamenti multiparte in corso per il vault specificato. Il campo `marker` è `null`, a indicare che non vi sono altri caricamenti da elencare.

```
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"
    }
  ]
}
```

Esempio: elenco parziale dei caricamenti in più parti

L'esempio seguente mostra come utilizzare la paginazione per ottenere un numero limitato di risultati. L'esempio mostra una richiesta HTTP GET inviata all'URI della sottorisorsa `multipart-uploads` per un determinato vault. In questo esempio, il parametro `limit` è impostato su 1, il che significa che un solo caricamento viene restituito nell'elenco; il parametro `marker` indica l'ID di caricamento in più parti da cui inizia l'elenco restituito.

Richiesta di esempio

```
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
```

Risposta di esempio

Nella risposta, Amazon Glacier (Amazon Glacier) restituisce un elenco di non più di due caricamenti multiparte in corso per il vault specificato, a partire dal marker specificato e restituendo due risultati.

```
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"
    }
  ]
}
```

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- Abort Multipart Upload (DELETE uploadID)
- List Parts (GET uploadID)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Identity and Access Management per Amazon Glacier

Upload Part (PUT uploadID)

Descrizione

Questa operazione di caricamento in più parti consente di caricare una parte di un archivio. Puoi caricare parti di un archivio in qualsiasi ordine in quanto nella richiesta Upload Part specifichi l'intervallo di byte dell'archivio assemblato che verrà caricato in questa parte. Puoi anche caricare le parti in parallelo. In un caricamento in più parti è consentito caricare fino a 10.000 parti.

Per informazioni sul caricamento in più parti, consulta Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti).

Amazon Glacier (Amazon Glacier) rifiuta la richiesta di caricamento di una parte se si verifica una delle seguenti condizioni:

- SHA256 l'hash dell'albero non corrisponde: per garantire che i dati della parte non vengano danneggiati durante la trasmissione, si calcola un hash ad SHA256 albero della parte e lo si include nella richiesta. Dopo aver ricevuto i dati del componente, Amazon Glacier calcola SHA256 anche un tree hash. Se i due valori hash non corrispondono, l'operazione non riesce. Per informazioni sul calcolo dell'hash di un SHA256 albero, consulta. Calcolo di checksum
- SHA256 l'hash lineare non corrisponde: per l'autorizzazione è necessario calcolare un hash SHA256 lineare dell'intero payload caricato e includerlo nella richiesta. Per informazioni sul calcolo di un hash SHA256 lineare, vedere. Calcolo di checksum
- La dimensione di parte non corrisponde: la dimensione di ogni parte eccetto l'ultima deve corrispondere alla dimensione specificata nella richiesta Initiate Multipart Upload (POST multipart-uploads) corrispondente. La dimensione dell'ultima parte deve essere uguale o inferiore alla dimensione specificata.

  > **Note**
  >
  > Se carichi una parte la cui dimensione è inferiore alla dimensione specificata nella richiesta Initiate Multipart Upload e quella parte non è l'ultima parte, la richiesta Upload Part verrà eseguita senza problemi. Tuttavia, la richiesta Complete Multipart Upload successiva non riuscirà.

- L'intervallo non è allineato: il valore dell'intervallo di byte nella richiesta non è allineato alla dimensione di parte specificata nella richiesta di avvio corrispondente. Ad esempio, se specifichi una dimensione di parte di 4194304 byte (4 MB), gli intervalli da 0 a 4194303 byte (4 MB – 1) e da 4194304 (4 MB) a 8388607 (8 MB – 1) sono intervalli di parte validi. Tuttavia, se imposti un valore di intervallo da 2 MB a 6 MB, l'intervallo non è allineato alla dimensione di parte e il caricamento non riuscirà.

Questa operazione è idempotente. Se carichi la stessa parte più volte, i dati inclusi nella richiesta più recente sovrascrivono i dati caricati precedentemente.

Richieste

Devi inviare questa richiesta PUT HTTP all'URI dell'ID di caricamento restituito dalla richiesta Initiate Multipart Upload. Amazon Glacier utilizza l'ID di caricamento per associare i caricamenti di parti a un caricamento multiparte specifico. La richiesta deve includere un hash ad SHA256 albero dei dati della parte (`x-amz-SHA256-tree-hash`intestazione), un hash SHA256 lineare dell'intero payload

(`x-amz-content-sha256`header), l'intervallo di byte (header) e la lunghezza della parte in byte (`Content-Range`header). `Content-Length`

Sintassi

```
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**
>
> Il `AccountId` valore è l'ID dell'account proprietario del vault. Account AWS Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza le seguenti intestazioni di richiesta, oltre alle intestazioni di richiesta comuni a tutte le operazioni. Per ulteriori informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Nome	Descrizione	Richiesto
`Content-Length`	Identifica la lunghezza della parte in byte. ▪Tipo: stringa	No

Nome	Descrizione	Richiesto
	Impostazione predefinita: nessuna Vincoli: nessuno	
`Content-Range`	Identifica l'intervallo di byte nell'archivio assemblato che verrà caricato in questa parte. Amazon Glacier utilizza queste informazioni per assemblare l'archivio nella sequenza corretta. Il formato di questa intestazione segue la specifica RFC 2616. Un esempio è `Content-Range:bytes 0-4194303/*` . ▪Tipo: stringa Impostazione predefinita: nessuna Vincoli: l'intervallo non può essere maggiore della dimensione di parte specificata all'avvio del caricamento in più parti.	Sì
`x-amz-content-sha256`	Il SHA256 checksum (un hash lineare) del payload caricato. Non si tratta dello stesso valore che specifichi nell'intestazione `x-amz-sha256-tree-hash` . ▪Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì

Nome	Descrizione	Richiesto
`x-amz-sha256-tree-hash`	Specifica un hash SHA256 ad albero dei dati da caricare. Per informazioni sul calcolo dell'hash di un SHA256 albero, vedere. Calcolo di checksum -Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	Sì

Corpo della richiesta

Il corpo della richiesta contiene i dati da caricare.

Risposte

Una volta completato il caricamento di un componente, Amazon Glacier restituisce una risposta. `204 No Content`

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-sha256-tree-hash: ChecksumComputedByAmazonGlacier
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`x-amz-sha256-tree-hash`	L'hash SHA256 dell'albero calcolato da Amazon Glacier per la parte caricata.

Nome	Descrizione
	-Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Esempio

La richiesta seguente carica una parte di 4 MB. La richiesta imposta l'intervallo di byte di modo che sia la prima parte nell'archivio.

Richiesta di esempio

L'esempio invia una richiesta HTTP PUT per caricare una parte di 4 MB. La richiesta è inviata all'URI dell'ID di caricamento restituito dalla richiesta Initiate Multipart Upload. L'intestazione `Content-Range` identifica la parte come la prima parte di dati di 4 MB dell'archivio.

```
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
```

Per caricare la parte successiva, la procedura è la stessa; tuttavia, è necessario calcolare un nuovo hash ad SHA256 albero della parte che si sta caricando e specificare anche un nuovo intervallo di byte per indicare dove andrà la parte nell'assemblaggio finale. La richiesta seguente carica un'altra parte utilizzando lo stesso ID di caricamento. La richiesta specifica i 4 MB successivi dell'archivio dopo la richiesta precedente e una dimensione di parte di 4 MB.

```
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
```

Le parti possono essere caricate in qualsiasi ordine; Amazon Glacier utilizza le specifiche dell'intervallo per ciascuna parte per determinare l'ordine in cui assemblarle.

Risposta di esempio

```
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
```

Sezioni correlate

- Initiate Multipart Upload (POST multipart-uploads)
- Upload Part (PUT uploadID)
- Complete Multipart Upload (POST uploadID)
- Abort Multipart Upload (DELETE uploadID)
- List Multipart Uploads (GET multipart-uploads)
- List Parts (GET uploadID)
- Caricamento in parti di archivi di grandi dimensioni (caricamento in più parti)
- Identity and Access Management per Amazon Glacier

Operazioni di processo

Di seguito sono elencate le operazioni di lavoro disponibili in Amazon Glacier.

Argomenti

- Describe Job (GET JobID)
- Get Job Output (GET output)
- Initiate Job (POST jobs)
- List Jobs (GET jobs)

Describe Job (GET JobID)

Descrizione

Questa operazione restituisce informazioni su un processo avviato in precedenza, tra cui la data di inizio del lavoro, l'utente che ha avviato il processo, il codice/messaggio di stato del lavoro e l'argomento Amazon Simple Notification Service (Amazon SNS) per la notifica dopo che Amazon Glacier (Amazon Glacier) ha completato il processo. Per ulteriori informazioni sull'avvio di un processo, consulta Initiate Job (POST jobs).

> **Note**
>
> Questa operazione ti consente di verificare lo stato del processo. Tuttavia, ti consigliamo vivamente di impostare un argomento di Amazon SNS e di specificarlo nella richiesta di avvio del lavoro in modo che Amazon Glacier possa notificare l'argomento una volta completato il processo.

L'ID del lavoro non scadrà per almeno 24 ore dopo il completamento del lavoro da parte di Amazon Glacier.

Richieste

Sintassi

Per ottenere informazioni su un processo, utilizzi il metodo HTTP `GET` e definisci l'ambito della richiesta per il processo specifico. Tieni presente che il percorso URI relativo è lo stesso che Amazon Glacier ti ha restituito quando hai avviato il processo.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '`-`' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

> **Note**
>
> Se ometti `JobID` nella richiesta, la risposta restituisce un elenco di tutti i processi attivi nel vault specificato. Per ulteriori informazioni sugli elenchi di processi, consulta List Jobs (GET jobs).

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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"
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

Azione

Il tipo di processo. Il valore è `ArchiveRetrieval`, `InventoryRetrieval` o `Select`.

- Tipo: stringa

ArchiveId

L'ID archivio richiesto per un processo di recupero di archivio o Select. In caso contrario, il campo è `null`.

- Tipo: stringa

Archive (Archivia)SHA256TreeHash

L'hash SHA256 ad albero dell'intero archivio per un processo di recupero dell'archivio. Per i processi di recupero di inventario, questo campo è `null`.

- Tipo: stringa

ArchiveSizeInBytes

Per un processo `ArchiveRetrieval`, si tratta della dimensione in byte dell'archivio richiesto per il download. Per un processo `InventoryRetrieval`, il valore è `null`.

Tipo: numero

Completato

Stato del processo. Al completamento di un processo di recupero di archivio o inventario, puoi ottenere l'output del processo utilizzando Get Job Output (GET output).

Tipo: Booleano

CompletionDate

La data in formato UTC alla quale il processo è stato completato. Quando il processo è in corso, il valore è null.

- Tipo: stringa

CreationDate

La data di creazione del processo in formato UTC.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

InventoryRetrievalParameters

I parametri di input utilizzati per un recupero di intervallo di inventario.

Tipo: oggetto InventoryRetrievalJobInput

InventorySizeInBytes

Per un processo `InventoryRetrieval`, si tratta della dimensione in byte dell'inventario richiesto per il download. Per un processo `ArchiveRetrieval` o `Select`, il valore è `null`.

Tipo: numero

JobDescription

La descrizione del processo che hai fornito all'avvio del processo.

- Tipo: stringa

JobId

L'ID che identifica il lavoro in Amazon Glacier.

- Tipo: stringa

JobOutputPath

Contiene la posizione dell'output del processo.

- Tipo: stringa

OutputLocation

Un oggetto che contiene informazioni sulla posizione in cui vengono archiviati i risultati e gli errori relativi al processo Select.

Tipo: oggetto OutputLocation

RetrievalByteRange

L'intervallo di byte recuperato per i processi di recupero dell'archivio nel formato "-». *`StartByteValue EndByteValue`* Se non si specifica un intervallo nel recupero dell'archivio, viene recuperato l'intero archivio; inoltre è uguale a 0 ed EndByteValueè StartByteValueuguale alla dimensione dell'archivio meno 1. Per i processi di recupero di inventario o Select, questo campo è `null`.

- Tipo: stringa

SelectParameters

Un oggetto che contiene informazioni sui parametri utilizzati per una selezione.

Tipo: oggetto SelectParameters

SHA256TreeHash

Il valore hash dell' SHA256 albero per l'intervallo richiesto di un archivio. Se nella richiesta Initiate Job (POST jobs) per un archivio è stato specificato un intervallo allineato alla struttura hash questo campo restituisce un valore. Per ulteriori informazioni sull'allineamento alla struttura hash per recuperi di intervallo di archivio, consulta Ottenimento di checksum durante il download di dati.

Per i casi in cui viene recuperato l'intero archivio, questo valore è identico al valore di `ArchiveSHA256TreeHash`.

Questo campo è `null` nei seguenti casi:

- Processi di recupero di archivio che specificano un intervallo non allineato alla struttura hash.
- Processi di archivio che specificano un intervallo corrispondente all'intero archivio e il cui stato è `InProgress`.
- Processi di inventario.
- Processi Select.

- Tipo: stringa

SNSTopic

Un argomento di Amazon SNS che riceve notifiche.

- Tipo: stringa

StatusCode

Il codice indicante lo stato del processo.

Valori validi: `InProgress` | `Succeeded` | `Failed`

- Tipo: stringa

StatusMessage

Un messaggio che descrive lo stato del processo.

- Tipo: stringa

Livello

Il livello di accesso ai dati da utilizzare per il processo di recupero di archivio o Select.

Valori validi: `Bulk` | `Expedited` | `Standard`

- Tipo: stringa

VaultARN

L'ARN del vault di cui il processo è una sottorisorsa.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra la richiesta per un processo che recupera un archivio.

Richiesta di esempio: ottenimento della descrizione di un processo

```
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
```

Risposta di esempio

Il corpo della risposta include codice JSON che descrive il processo specificato. Nota che per i processi di recupero di inventario e di archivio, i campi JSON sono identici. Tuttavia, quando un campo non si applica al tipo di processo, il relativo valore è `null`. Di seguito è riportata una risposta di esempio per un processo di recupero di archivio. Tieni presente quanto segue:

- Il valore del campo `Action` è `ArchiveRetrieval`.
- Il campo `ArchiveSizeInBytes` indica la dimensione dell'archivio richiesto nel processo di recupero di archivio.
- Il `ArchiveSHA256TreeHash` campo mostra l'hash dell' SHA256 albero per l'intero archivio.
- Il campo `RetrievalByteRange` indica l'intervallo desiderato nella richiesta Initiate Job. In questo esempio, viene richiesto l'intero archivio.
- Il `SHA256TreeHash` campo mostra l'hash SHA256 dell'albero per l'intervallo richiesto nella richiesta Initiate Job. In questo esempio, è identico al campo `ArchiveSHA256TreeHash`, il che significa che è stato richiesto l'intero archivio.
- Il valore del campo `InventorySizeInBytes` è `null` in quanto non si applica a un processo di recupero di archivio.

```
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"
}
```

Di seguito è riportata una risposta di esempio per un processo di recupero di inventario. Tieni presente quanto segue:

- Il valore del campo `Action` è `InventoryRetrieval`.
- I valori dei campi `ArchiveSizeInBytes`, `ArchiveSHA256TreeHash` e `RetrievalByteRange` sono null in quanto questi campi non si applicano a un processo di recupero di inventario.
- Il valore del campo `InventorySizeInBytes` è `null` in quanto il processo è ancora in corso e non ha preparato completamente l'inventario per il download. Se il processo fosse completato prima della richiesta Describe Job, questo campo indicherebbe la dimensione dell'output.

```
{
   "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"
}
```

Di seguito viene fornita una risposta di esempio per un processo di recupero di archivio completato che contiene un contrassegno utilizzato per continuare la paginazione del recupero di inventario vault.

```
{
    "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
    },
}
```

Sezioni correlate

- Get Job Output (GET output)
- Identity and Access Management per Amazon Glacier

Get Job Output (GET output)

Descrizione

Questa operazione scarica l'output del processo avviato utilizzando Initiate Job (POST jobs). A seconda del tipo di processo specificato all'avvio del processo, l'output sarà il contenuto di un archivio o di un inventario vault.

Puoi scaricare tutto l'output del processo o soltanto una parte specificando un intervallo di byte. Per i processi di recupero di archivio e di inventario, devi confrontare la dimensione scaricata e quella restituita nelle intestazioni della risposta Get Job Output.

Per i processi di recupero di archivio, devi inoltre verificare che la dimensione sia quella prevista. Se scarichi una parte dell'output, la dimensione prevista è basata sull'intervallo di byte specificato. Ad esempio, se specifichi un intervallo `bytes=0-1048575`, la dimensione del download deve essere 1.048.576 byte. Se scarichi un intero archivio, la dimensione prevista è la dimensione dell'archivio quando lo hai caricato su Amazon Glacier (Amazon Glacier). La dimensione prevista è inoltre restituita nelle intestazioni dalla risposta Get Job Output.

Nel caso di un processo di recupero dell'archivio, a seconda dell'intervallo di byte specificato, Amazon Glacier restituisce il checksum per la parte di dati. Per assicurarti che la parte scaricata sia quella corretta, calcola il checksum sul client, verifica che i valori corrispondano e che la dimensione sia quella prevista.

L'ID del lavoro non scade per almeno 24 ore dopo il completamento del lavoro da parte di Amazon Glacier. In altre parole, puoi scaricare l'output del lavoro entro 24 ore dal completamento del processo da parte di Amazon Glacier.

Richieste

Sintassi

Per recuperare un processo di output, invii la richiesta HTTP `GET` all'URI dell'`output` del processo.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza le seguenti intestazioni di richiesta, oltre alle intestazioni di richiesta comuni a tutte le operazioni. Per ulteriori informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Nome	Descrizione	Richiesto
`Range`	L'intervallo di byte da recuperare dall'output. Ad esempio, se desideri scaricare i primi 1.048.576 byte, specifica l'intervallo `bytes=0-1048575` . Per ulteriori informazioni, consulta la sezione relativa alla definizione del campo dell'intestazione Range. L'intervallo è relativo a qualsiasi intervallo specifica to nella richiesta Initiate Job. Per impostazione predefinita, questa operazione scarica l'intero output. Se l'output del processo è di grandi dimensioni, puoi utilizzar e l'intestazione della richiesta `Range` per recuperare una parte dell'output. Ciò ti consente di scaricare l'intero output in blocchi di byte più piccoli. Ad esempio, supponi di avere un output di processo di 1 GB e che decidi di scaricarlo in blocchi di	No

Nome	Descrizione	Richiesto
	128 MB di dati mediante otto richieste Get Job Output. Per scaricare l'output, devi procedere come segue: 1. Scarica un blocco di 128 MB di output specificando l'interva llo di byte appropriato mediante l'intestazione `Range`. Verifica che tutti i 128 MB di dati siano stati scaricati. 2. Oltre ai dati, la risposta includerà un checksum del payload. Per assicurarti di aver ricevuto tutti i dati previsti, calcola il checksum del payload sul client e confrontalo con il checksum ricevuto nella risposta. 3. Ripeti le fasi 1 e 2 per tutti i blocchi di 128 MB di dati dell'output, specificando ogni volta l'intervallo di byte appropriato. 4. Dopo il download di tutte le parti dell'output, hai un elenco di otto valori di checksum. Calcola la struttura hash di questi valori per trovare il checksum dell'intero output. Con l'operazione Describe Job (GET JobID), ottieni le informazi oni sul processo relativo all'output. La risposta include il checksum dell'intero archivio archiviato in Amazon Glacier. Confronta questo valore con il checksum che hai calcolato per accertarti di aver scaricato l'intero contenuto dell'archivio senza errori. ▪Tipo: stringa Impostazione predefinita: nessuna Vincoli: nessuno	

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

Per una richiesta di recupero che restituisce tutti i dati del processo, la risposta del relativo output restituisce un codice di risposta `200 OK`. Quando viene richiesto un contenuto parziale, ad esempio, se hai specificato l'intestazione `Range` nella richiesta, viene restituito il codice di risposta `206 Partial Content`.

```
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.]
```

Intestazioni di risposta

Header	Descrizione
`Content-Range`	L'intervallo di byte restituito da Amazon Glacier. Se viene scaricato solo un output parziale, la risposta fornisce l'intervallo di byte restituito da Amazon Glacier. Ad esempio, `bytes 0-1048575/8388608` restituisce il primo blocco da 1 MB di 8 MB. Per ulteriori informazioni sull'intestazione `Content-Range` , consulta la pagina relativa alla definizione del campo dell'intestazione Content-Range. ▪Tipo: stringa
`Content-Type`	Content-Type varia a seconda se l'output del processo è un archivio o un inventario vault.

Header	Descrizione
	• Per i dati dell'archivio, Content-Type è `application/octet-stream` . • Per un inventario vault, se hai richiesto il formato CSV all'avvio del processo, Content-Type è `text/csv`. In caso contrario, per impostazi one predefinita, l'inventario vault viene restituito in formato JSON e Content-Type è `application/json` . -Tipo: stringa

Header	Descrizione
`x-amz-sha256-tree-hash`	Il checksum dei dati nella risposta. Questa intestazione viene restituit a solo per il recupero dell'output di un processo di recupero di archivio. Viene inoltre visualizzata quando l'intervallo di dati richiesto in Initiate Job è allineato alla struttura hash e l'intervallo da scaricare richiesto nelle richiesta Get Job Output è anch'esso allineato alla struttura hash. Per ulteriori informazioni sugli intervalli allineati alla struttura hash, vedi Ottenimento di checksum durante il download di dati. Ad esempio, se nella richiesta Initiate Job hai specificato un intervallo da recuperare che è allineato alla struttura hash (include l'intero archivio), riceverai il checksum dei dati che scarichi quando: • Ottieni tutto l'intervallo dei dati recuperati. • Richiedi un intervallo di byte dei dati recuperati che ha una dimension e di un megabyte (1024 KB) moltiplicato per una potenza di 2 e che inizia e finisce con un multiplo della dimensione dell'intervallo richiesto . Ad esempio, se hai 3,1 MB di dati recuperati e specifichi un intervall o da restituire che inizia a 1 MB e finisce a 2 MB, viene restituito `x-amz-sha256-tree-hash` come intestazione della risposta. • Richiedi un intervallo di dati recuperati da restituire che va fino alla fine dei dati e l'inizio dell'intervallo è un multiplo della dimensione dell'inte rvallo da recuperare arrotondato alla successiva potenza di due ma non inferiore a un megabyte (1024 KB). Ad esempio, se hai 3,1 MB di dati recuperati e specifichi un intervallo che inizia a 2 MB e finisce a 3,1 MB (la fine dei dati), viene restituito `x-amz-sha256-tree-hash` come intestazione della risposta. ➧Tipo: stringa

Corpo della risposta

Amazon Glacier restituisce l'output del lavoro nel corpo della risposta. A seconda del tipo di processo, l'output può essere il contenuto dell'archivio o l'inventario vault. Nel caso di un inventario vault, per impostazione predefinita l'elenco viene restituito come corpo JSON, come mostrato di seguito.

```
{
 "VaultARN": String,
 "InventoryDate": String,
 "ArchiveList": [
      {"ArchiveId": String,
       "ArchiveDescription": String,
       "CreationDate": String,
       "Size": Number,
       "SHA256TreeHash": String
      },
      ...
    ]
}
```

Se hai richiesto l'output in formato CSV all'avvio del processo di inventario vault, l'inventario vault viene restituito in formato CSV nel corpo della risposta. Il formato CSV ha cinque colonne "ArchiveId«," ArchiveDescription «," CreationDate «, «Size» e "SHA256TreeHash" con le stesse definizioni dei campi JSON corrispondenti.

> **Note**
>
> Nel formato CSV restituito, i campi possono essere restituiti racchiusi tra virgolette. I campi che contengono una virgola o virgolette vengono sempre restituiti racchiusi tra virgolette. Ad esempio, `my archive description,1` viene restituito come `"my archive description,1"`. Le virgolette presenti nei campi restituiti tra virgolette sono precedute da un carattere di escape, ovvero una barra rovesciata. Ad esempio, `my archive description,1"2` viene restituito come `"my archive description,1\"2"` e `my archive description,1\"2` viene restituito come `"my archive description,1\\\"2"`. La barra rovesciata non è preceduta da un carattere di escape.

Il corpo della risposta JSON contiene i campi JSON descritti di seguito.

ArchiveDescription

La descrizione di un archivio.

- Tipo: stringa

ArchiveId

L'ID di un archivio.

- Tipo: stringa

ArchiveList

Una matrice di metadati dell'archivio. Ogni oggetto nella matrice rappresenta metadati per un archivio contenuto nel vault.

Tipo: matrice

CreationDate

La data e l'ora di creazione dell'archivio in formato UTC.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

InventoryDate

La data e l'ora in formato UTC dell'ultimo inventario del vault completato dopo le modifiche al vault. Anche se Amazon Glacier prepara un inventario del vault una volta al giorno, la data dell'inventario viene aggiornata solo se sono state aggiunte o eliminate dall'archivio dopo l'ultimo inventario.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

SHA256TreeHash

La struttura hash dell'archivio.

- Tipo: stringa

Dimensioni

La dimensione in byte dell'archivio.

Tipo: numero

VaultARN

La risorsa ARN (Amazon Resource Name) da cui è stato richiesto il recupero dell'archivio.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra la richiesta per un processo che recupera un archivio.

Esempio 1: download dell'output

Questo esempio recupera i dati preparati da Amazon Glacier in risposta alla tua richiesta di lavoro di avvio del recupero dell'archivio.

Richiesta di esempio

```
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
```

Risposta di esempio

Di seguito è riportato un esempio di risposta di un processo di recupero di archivio. Nota che l'intestazione `Content-Type` è `application/octet-stream` e che l'intestazione `x-amz-sha256-tree-hash` è inclusa nella risposta, il che significa che vengono restituiti tutti i dati del processo.

```
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.]
```

Di seguito è riportato un esempio di risposta di un processo di recupero di inventario. Nota che l'intestazione `Content-Type` è `application/json`. Considera inoltre che la risposta non include l'intestazione `x-amz-sha256-tree-hash`.

```
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"
   }
```

```
  ]
}
```

Esempio 2: download unicamente di un output parziale

Questo esempio recupera solo una parte dell'archivio preparato da Amazon Glacier in risposta alla tua richiesta di lavoro di avvio del recupero dell'archivio. La richiesta utilizza l'intestazione facoltativa `Range` per recuperare solo i primi 1.024 byte.

Richiesta di esempio

```
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
```

Risposta di esempio

La risposta corretta seguente mostra la risposta `206 Partial Content`. In questo caso, la risposta include anche un'`Content-Range`intestazione che specifica l'intervallo di byte restituito da 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.]
```

Sezioni correlate

- Describe Job (GET JobID)

- Initiate Job (POST jobs)
- Identity and Access Management per Amazon Glacier

Initiate Job (POST jobs)

Questa operazione avvia i seguenti tipi di processi Amazon Glacier (Amazon Glacier):

- `archive-retrieval`: recupero di un archivio
- `inventory-retrieval`: inventario di una vault

Argomenti

- Avvio di un processo di recupero di un archivio o di un inventario vault
- Richieste
- Risposte
- Esempi
- Sezioni correlate

Avvio di un processo di recupero di un archivio o di un inventario vault

Il recupero di un archivio o di un inventario vault è un'operazione asincrona che richiede l'avvio di un processo. Una volta avviato, il processo non può essere annullato. Un processo di recupero comporta due fasi:

1. Avvio di un processo di recupero mediante l'operazione Initiate Job (POST jobs).

   > **Important**
   >
   > L'utilizzo di una policy di recupero dati può causare un errore nella richiesta di avvio di un processo di recupero e generare l'eccezione `PolicyEnforcedException`. Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon Glacier. Per ulteriori informazioni sull'eccezione `PolicyEnforcedException`, consulta Risposte agli errori.

2. Download dei byte mediante l'operazione Get Job Output (GET output) dopo il completamento del processo.

La richiesta di recupero viene eseguita in modo asincrono. Quando avvii un processo di recupero, Amazon Glacier crea un processo e restituisce un ID del lavoro nella risposta. Quando Amazon Glacier completa il lavoro, puoi ottenere l'output del lavoro (dati di archivio o di inventario). Per informazioni su come ottenere l'output di un processo, consulta la sezione sull'operazione Get Job Output (GET output).

Per ottenere l'output di un processo, è necessario che il processo sia stato completato. Per determinare se un processo è completato, disponi delle seguenti opzioni:

- Usa una notifica Amazon SNS: puoi specificare un argomento Amazon SNS su cui Amazon Glacier può pubblicare una notifica dopo il completamento del processo. Puoi specificare un argomento di SNS per ogni richiesta di processo. La notifica viene inviata solo dopo che Amazon Glacier ha completato il processo. Oltre a specificare un argomento di SNS per ogni richiesta di processo, puoi configurare notifiche per un vault di modo che siano inviate per tutte le operazioni di recupero. Per ulteriori informazioni, consulta Set Vault Notification Configuration (PUT notification-configuration).
- Ottenere dettagli sul processo: puoi effettuare una richiesta Describe Job (GET JobID) per ottenere informazioni sullo stato del processo durante la sua esecuzione. È comunque consigliato utilizzare un notifica di Amazon SNS per stabilire se un processo è completato.

> **Note**
>
> Le informazioni che ottieni tramite le notifiche sono le stesse che ottieni chiamando Describe Job (GET JobID).

Se per un evento specifico, aggiungi sia la configurazione delle notifiche sul vault sia specifichi anche un argomento SNS nella richiesta di avvio del lavoro, Amazon Glacier invia entrambe le notifiche. Per ulteriori informazioni, consulta Set Vault Notification Configuration (PUT notification-configuration).

Inventario vault

Amazon Glacier aggiorna un inventario del vault circa una volta al giorno, a partire dal giorno in cui carichi per la prima volta un archivio nel vault. Se nessun archivio è stato aggiunto o eliminato nel vault dall'ultimo inventario, la data dell'inventario non viene aggiornata. Quando avvii un processo per un inventario del vault, Amazon Glacier restituisce l'ultimo inventario generato, che è point-in-time uno snapshot e non dati in tempo reale.

Dopo che Amazon Glacier ha creato il primo inventario per il vault, in genere occorre mezza giornata e fino a un giorno prima che l'inventario sia disponibile per il recupero.

A prima vista, recuperare un inventario vault per ogni caricamento di archivio può non sembrare utile. Tuttavia, supponiamo che tu gestisca un database sul lato client che associ i metadati sugli archivi che carichi su Amazon Glacier. In tal caso, l'inventario vault potrebbe essere utilizzato per risolvere le differenze tra le informazioni nel database e l'inventario vault effettivo. Per ulteriori informazioni sui campi dati restituiti nell'output di un processo di inventario, consulta Corpo della risposta.

Recupero di un intervallo di inventario

Puoi limitare il numero di elementi di inventario recuperati filtrandoli per data di creazione dell'archivio o impostando un limite.

Filtro in base alla data di creazione dell'archivio

Puoi recuperare elementi di inventario degli archivi creati tra `StartDate` ed `EndDate` specificando i valori di tali parametri nella richiesta Initiate Job. In questo modo, vengono restituiti gli archivi creati dopo `StartDate` e prima di `EndDate`. Se indichi solo il valore `StartDate` senza `EndDate`, recuperi l'inventario per tutti gli archivi creati a partire da `StartDate`. Se indichi solo il valore `EndDate` senza `StartDate`, recuperi l'inventario per tutti gli archivi creati prima di `EndDate`.

Limitazione degli elementi di inventario recuperati

Puoi limitare il numero di elementi di inventario restituiti impostando il parametro `Limit` nella richiesta Initiate Job. L'output del processo di inventario contiene elementi di inventario fino al valore `Limit` specificato. Se sono disponibili ulteriori elementi di inventario, viene eseguita la paginazione del risultato. Dopo il completamento di un processo, puoi utilizzare l'operazione Describe Job (GET JobID) per ottenere un contrassegno da utilizzare in una richiesta Initiate Job successiva. Il contrassegno indica il punto iniziale per il recupero del set di elementi di inventario successivo. Puoi scorrere l'intero inventario effettuando ripetutamente richieste Initiate Job con il contrassegno dell'output della richiesta Describe Job precedente. Procedi in questo modo fino a ottenere un contrassegno da Describe Job che restituisce null, a indicare che non ci sono altri elementi di inventario disponibili.

Puoi utilizzare il parametro `Limit` con i parametri relativi agli intervalli di date.

Recupero di un intervallo di archivio

Puoi avviare il recupero di un intero archivio oppure di un intervallo dell'archivio. Per il recupero di un intervallo di archivio, devi specificare un intervallo di byte da restituire o l'intero archivio. L'intervallo

specificato deve essere allineato in termini di megabyte (MB). In altre parole, il valore di inizio dell'intervallo deve essere divisibile per 1 MB e quello di fine più 1 deve essere divisibile per 1 MB o uguale alla fine dell'archivio. Se il recupero di un intervallo di archivio non è allineato in termini di megabyte, l'operazione restituisce una risposta `400`. Inoltre, per ottenere i valori di checksum dei dati scaricati utilizzando la richiesta Get Job Output (Get Job Output (GET output)), l'intervallo deve essere allineato alla struttura hash. Per ulteriori informazioni sugli intervalli allineati alla struttura hash, vedi Ottenimento di checksum durante il download di dati.

Recuperi di tipo Expedited, Standard e Bulk

Quando avvii un processo di recupero di archivio, puoi specificare una delle seguenti opzioni nel campo `Tier` del corpo della richiesta:

- **`Expedited`**: i recuperi di tipo Expedited consentono di accedere rapidamente ai tuoi dati in caso di richieste occasionali e urgenti di ripristino degli archivi. Per gli archivi di dimensioni inferiori a 250 MB, i tempi di recupero con il livello Expedited sono in genere compresi tra 1 e 5 minuti.
- **`Standard`**: i recuperi di tipo Standard consentono l'accesso a qualsiasi archivio nel giro di qualche ora. I tempi di recupero sono in genere inclusi tra le 3 e le 5 ore. Questa è l'opzione predefinita per le richieste di processo in cui non viene specificato il tipo di recupero.
- **`Bulk`**— Bulk è il livello più economico per Amazon Glacier, che consente di recuperare grandi quantità di dati, anche petabyte, a costi contenuti in un giorno. I tempi di recupero sono inclusi in genere tra le 5 e le 12 ore.

Per ulteriori informazioni sui recuperi di tipo Expedited e Bulk, consulta Recupero degli archivi Amazon Glacier.

Richieste

Per avviare un processo, utilizzi il metodo HTTP `POST` e definisci l'ambito della richiesta per la sottorisorsa `jobs` del vault. I dettagli relativi alla richiesta di processo devono essere indicati nel documento JSON della richiesta. Il tipo di processo è specificato nel campo `Type`. Facoltativamente, puoi specificare un `SNSTopic` campo per indicare un argomento di Amazon SNS su cui Amazon Glacier può inviare notifiche dopo il completamento del processo.

> **Note**
>
> Per pubblicare una notifica in Amazon SNS, devi creare l'argomento se ancora non esiste, Amazon Glacier non crea l'argomento per te. L'argomento deve disporre delle autorizzazioni

per ricevere pubblicazioni da un vault Amazon Glacier. Amazon Glacier non verifica se il vault è autorizzato a pubblicare sull'argomento. Se le autorizzazioni non sono configurate in modo appropriato, potresti non ricevere la notifica anche dopo il completamento del processo.

Sintassi

Di seguito viene riportata la sintassi della richiesta di avvio di un processo.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON nel corpo della richiesta.

jobParameters

Fornisce opzioni per specificare informazioni sul processo.

Tipo: oggetto jobParameters

Campo obbligatorio: sì

Risposte

Amazon Glacier crea posti di lavoro. e restituisce l'URI del processo nella risposta.

Sintassi

```
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
```

Intestazioni di risposta

Header	Descrizione
`Location`	

Header	Descrizione
	Il percorso URI relativo del processo. Puoi utilizzare questo percorso per trovare lo stato del processo. Per ulteriori informazioni, consulta Describe Job (GET JobID). ▪Tipo: stringa Impostazione predefinita: nessuna
`x-amz-job-id`	L'ID del processo. Questo valore è incluso anche nell'intestazione `Location`. ▪Tipo: stringa Impostazione predefinita: nessuna
`x-amz-job-output-path`	Il percorso alla posizione in cui sono archiviati i risultati del processo Select. ▪Tipo: stringa Impostazione predefinita: nessuna

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Questa operazione include l'errore o gli errori seguenti, oltre ai possibili errori comuni a tutte le operazioni di Amazon Glacier. Per informazioni sugli errori di Amazon Glacier e un elenco di codici di errore, consulta. Risposte agli errori

Codice	Descrizione	Codice di stato HTTP	Tipo
`InsufficientCapacityException`	Restituito in caso di capacità insufficiente per elaborare la richiesta di tipo Expedited.	`503 Service Unavailable`	Server

Codice	Descrizione	Codice di stato HTTP	Tipo
	Questo errore è relativo solo ai recuperi di tipo Expedited e non a quelli di tipo Standard o Bulk.		

Esempi

Esempio di richiesta: avvio di un processo di recupero di archivio

```
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"
}
```

Di seguito viene riportato un esempio del corpo di una richiesta che specifica un intervallo dell'archivio da recuperare mediante il campo `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"
}
```

Risposta di esempio

```
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
```

Esempio di richiesta: avvio di un processo di recupero di inventario

La richiesta seguente avvia un processo di recupero di inventario per ottenere un elenco di archivi dal vault `examplevault`. Il parametro `Format` impostato su `CSV` nel corpo della richiesta indica che l'inventario viene restituito in formato 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"
}
```

Risposta di esempio

```
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
```

Esempio di richieste: avvio di un processo di recupero di inventario utilizzando un filtro data con un limite definito e richiesta di recupero successiva della pagina di elementi di inventario seguente.

La richiesta che segue avvia un processo di recupero di inventario vault mediante un filtro data e un limite definito.

```
{
    "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 richiesta che segue è un esempio di richiesta di recupero successiva della pagina di elementi di inventario seguente mediante un contrassegno ottenuto da Describe Job (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
```

```
    },
}
```

Risposta di esempio

```
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/
```

Sezioni correlate

- Describe Job (GET JobID)
- Get Job Output (GET output)
- Identity and Access Management per Amazon Glacier

List Jobs (GET jobs)

Descrizione

Questa operazione visualizza un elenco dei processi per un vault, inclusi i processi in corso e quelli completati di recente.

> **Note**
>
> Amazon Glacier (Amazon Glacier) conserva i lavori completati di recente per un periodo prima di eliminarli; tuttavia, alla fine rimuove i lavori completati. L'output dei processi completati può essere recuperato. La conservazione per un certo periodo di tempo dei processi completati ti consente di ottenere l'output corrispondente nel caso di mancata lettura della notifica relativa al completamento del processo o di errore durante il primo tentativo di download. Supponi, ad esempio, di avviare un processo di recupero di archivio per scaricare

un archivio. Dopo il completamento del processo, esegui il download dell'archivio ma si verifica un errore di rete. In questo scenario, puoi ancora riprovare a scaricare l'archivio fino a che il processo esiste.

L'operazione `List Jobs` supporta la paginazione. Devi sempre verificare il campo `Marker` nella risposta. Se non vi sono altri processi da elencare, il campo `Marker` è impostato su `null`. Se invece sono presenti altri processi, il campo `Marker` è impostato su un valore non null che puoi utilizzare per continuare la paginazione dell'elenco. Per restituire un elenco di processi che inizia da un determinato processo, imposta il parametro di richiesta `marker` sul valore `Marker` di quel processo che hai ottenuto da una precedente richiesta `List Jobs`.

Puoi impostare un limite massimo per il numero di processi restituiti nella risposta specificando il parametro `limit` nella richiesta. Il limite di default è 50. Il numero di processi restituiti può essere inferiore al limite specificato, ma mai superiore.

Inoltre, specificando il parametro facoltativo `statuscode` o `completed` oppure entrambi puoi filtrare l'elenco di processi restituito. Con il parametro `statuscode` puoi scegliere di restituire solo i processi il cui stato è `InProgress`, `Succeeded`o `Failed`, mentre il parametro `completed` ti consente di restituire unicamente i processi completati (`true`) o quelli non completati (`false`).

Richieste

Sintassi

Per restituire un elenco con tutti i tipi di processi, invia una richiesta `GET` all'URI della sottorisorsa `jobs` del vault.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID dell'account proprietario del vault. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon

Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se utilizzi un ID account, non includere alcun trattino ("-") nell'ID.

Parametri della richiesta

Nome	Descrizione	Richiesto
`completed`	Lo stato dei processi da restituire. Puoi specificare `true` o `false`. Tipo: Booleano Vincoli: nessuno	No
`limit`	Il numero massimo di processi da restituire. Il limite di default è 50. Il numero di processi restituiti può essere inferiore al limite specificato, ma mai superiore. -Tipo: stringa Vincoli: il valore intero minimo è 1. Il valore intero massimo è 50.	No
`marker`	Una stringa opaca utilizzata per la paginazione che indica il processo da cui deve iniziare l'elenco di processi. Il valore `marker` è ottenuto da una risposta `List Jobs` precedente. Devi soltanto includere il valore `marker` se continui la paginazione dei risultati iniziata in una richiesta `List Jobs` precedente. -Tipo: stringa Vincoli: nessuno	No
`statuscode`	Il tipo di stato di processo da restituire. -Tipo: stringa	No

Nome	Descrizione	Richiesto
	Vincoli: uno dei valori `InProgress` , `Succeeded` o `Failed`.	

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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"
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

JobList

Un elenco di oggetti del processo. Ogni oggetto del processo contiene metadati che descrivono il processo.

Tipo: matrice di oggetti GlacierJobDescription

Marker

Una stringa opaca che rappresenta il punto da cui continuare la paginazione dei risultati. Puoi utilizzare il valore `marker` in una nuova richiesta `List Jobs` per includere ulteriori processi nell'elenco. Se non vi sono altri processi da elencare, questo valore è `null`.

- Tipo: stringa

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come restituire informazioni sui processi del vault. Il primo esempio restituisce un elenco di due processi e il secondo restituisce un sottoinsieme di processi.

Esempio: restituzione di tutti i processi

Richiesta di esempio

La richiesta `GET` seguente restituisce i processi di un vault.

```
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
```

Risposta di esempio

La risposta seguente include un processo di recupero di archivio e un processo di recupero di inventario che contiene un contrassegno utilizzato per continuare la paginazione del recupero di inventario vault. La risposta mostra anche il campo `Marker` impostato su `null`, a indicare che non vi sono altri processi da elencare.

```
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
}
```

Esempio: restituzione di un elenco parziale di processi

Richiesta di esempio

La richiesta GET seguente restituisce il processo specificato mediante il parametro `marker`. L'impostazione del parametro `limit` su 2 indica che vengono restituiti fino a due processi.

```
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
```

Risposta di esempio

La risposta seguente mostra due processi restituiti e il campo `Marker` impostato su un valore non null che può essere utilizzato per continuare la paginazione dell'elenco di processi.

```
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"
}
```

Sezioni correlate

- Describe Job (GET JobID)
- Identity and Access Management per Amazon Glacier

Tipi di dati utilizzati nelle operazioni di processo

Di seguito sono riportati i tipi di dati utilizzati nelle operazioni di lavoro in Amazon Glacier.

Argomenti

- CSVInput
- CSVOutput
- Crittografia
- GlacierJobDescription
- Grant
- Grantee
- InputSerialization
- InventoryRetrievalJobInput
- jobParameters
- OutputLocation
- OutputSerialization
- S3Location
- SelectParameters

CSVInput

Contiene informazioni sul file CSV.

Indice

Commenti

Un singolo carattere utilizzato per indicare che una riga deve essere ignorata quando il carattere è presente all'inizio della riga.

Tipo: Stringa

Obbligatorio: no

FieldDelimiter

Un singolo carattere utilizzato per separare campi in un record. Il carattere deve essere `\n`, `\r` oppure un carattere ASCII nell'intervallo 32-126. L'impostazione predefinita è una virgola (`,`).

Tipo: Stringa

Default: ,

Obbligatorio: no

FileHeaderInfo

Un valore che descrive come utilizzare la prima riga dell'input.

Tipo: Stringa

Valori validi: `Use` | `Ignore` | `None`

Obbligatorio: no

QuoteCharacter

Un singolo carattere utilizzato come carattere di escape dove il delimitatore di campo è parte del valore.

Tipo: Stringa

Obbligatorio: no

QuoteEscapeCharacter

Un singolo carattere utilizzato come carattere di escape per le virgolette in un valore già preceduto da un carattere di escape.

Tipo: Stringa

Obbligatorio: no

RecordDelimiter

Un singolo carattere utilizzato per separare record.

Tipo: Stringa

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

CSVOutput

Contiene informazioni sul formato CSV nel quale vengono memorizzati i risultati.

Indice

FieldDelimiter

Un singolo carattere utilizzato per separare campi in un record.

Tipo: Stringa

Obbligatorio: no

QuoteCharacter

Un singolo carattere utilizzato come carattere di escape dove il delimitatore di campo è parte del valore.

Tipo: Stringa

Obbligatorio: no

QuoteEscapeCharacter

Un singolo carattere utilizzato come carattere di escape per le virgolette in un valore già preceduto da un carattere di escape.

Tipo: Stringa

Obbligatorio: no

QuoteFields

Un valore che indica se tutti i campi dell'output devono essere racchiusi tra virgolette.

Valori validi: `ALWAYS` | `ASNEEDED`

Tipo: Stringa

Obbligatorio: no

RecordDelimiter

Un singolo carattere utilizzato per separare record.

Tipo: Stringa

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

Crittografia

Contiene informazioni sulla crittografia utilizzata per l'archiviazione dei risultati dei processi in Amazon S3.

Indice

Encryption (Crittografia)

L'algoritmo di crittografia lato server utilizzato per l'archiviazione dei risultati dei processi in Amazon S3. Il valore di default non prevede la crittografia.

Tipo: Stringa

Valori validi: `aws:kms` | `AES256`

Obbligatorio: no

KMSContext

Facoltativo. Se il tipo di crittografia è `aws:kms`,, puoi utilizzare questo valore per specificare il contesto di crittografia dei risultati dei processi.

Tipo: Stringa

Obbligatorio: no

KMSKeyId

L'ID della chiave AWS Key Management Service (AWS KMS) da utilizzare per la crittografia degli oggetti.

Tipo: Stringa

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

GlacierJobDescription

Contiene la descrizione di un job Amazon Glacier (Amazon Glacier).

Indice

Azione

Il tipo di processo. Il valore è `ArchiveRetrieval`, `InventoryRetrieval` o `Select`.

-Tipo: stringa

ArchiveId

L'ID archivio richiesto per un processo di recupero di archivio o Select. In caso contrario, il campo è `null`.

- Tipo: stringa

Archive (Archivia)SHA256TreeHash

L'hash SHA256 ad albero dell'intero archivio per il recupero dell'archivio. Per i processi di recupero di inventario, questo campo è `null`.

- Tipo: stringa

ArchiveSizeInBytes

Per un processo `ArchiveRetrieval`, si tratta della dimensione in byte dell'archivio richiesto per il download. Per un processo `InventoryRetrieval`, il valore è `null`.

Tipo: numero

Completato

`true` se il processo è completato, altrimenti è `false`.

Tipo: Booleano

CompletionDate

La data di completamento del processo.

La data in formato UTC alla quale il processo è stato completato. Quando il processo è in corso, il valore è null.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

CreationDate

La data di avvio del processo in formato UTC.

Tipo: una rappresentazione di stringa nel formato data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

InventoryRetrievalParameters

I parametri di input utilizzati per un recupero di intervallo di inventario.

Tipo: oggetto InventoryRetrievalJobInput

InventorySizeInBytes

Per un processo `InventoryRetrieval`, si tratta della dimensione in byte dell'inventario richiesto per il download. Per un processo `ArchiveRetrieval` o `Select`, il valore è `null`.

Tipo: numero

JobDescription

La descrizione del processo che hai fornito all'avvio del processo.

- Tipo: stringa

JobId

L'ID che identifica il lavoro in Amazon Glacier.

- Tipo: stringa

JobOutputPath

Contiene la posizione dell'output del processo.

- Tipo: stringa

OutputLocation

Un oggetto che contiene informazioni sulla posizione in cui vengono archiviati i risultati e gli errori relativi al processo Select.

Tipo: oggetto OutputLocation

RetrievalByteRange

L'intervallo di byte recuperato per i processi di recupero dell'archivio nel formato "-». *`StartByteValue EndByteValue`* Se non è stato specificato alcun intervallo nel recupero dell'archivio, viene recuperato l'intero archivio ed è StartByteValueuguale a 0 e EndByteValuealla dimensione dell'archivio meno 1. Per i processi di recupero di inventario, questo campo è `null`.

- Tipo: stringa

SelectParameters

Un oggetto che contiene informazioni sui parametri utilizzati per una selezione.

Tipo: oggetto SelectParameters

SHA256TreeHash

Il valore hash dell' SHA256 albero per l'intervallo richiesto di un archivio. Se nella richiesta Initiate Job (POST jobs) per un archivio è stato specificato un intervallo allineato alla struttura hash

questo campo restituisce un valore. Per ulteriori informazioni sull'allineamento alla struttura hash per recuperi di intervallo di archivio, consulta Ottenimento di checksum durante il download di dati.

Per i casi in cui viene recuperato l'intero archivio, questo valore è uguale al valore `ArchiveSHA256TreeHash`.

Questo campo è `null` nei seguenti casi:

- Processi di recupero di archivio che specificano un intervallo non allineato alla struttura hash.
- Processi di archivio che specificano un intervallo corrispondente all'intero archivio e il cui stato è `InProgress`.
- Processi di inventario.
- Processi Select.

- Tipo: stringa

SNSTopic

Il nome della risorsa Amazon (ARN) che rappresenta l'argomento di Amazon SNS a cui è inviata la notifica di completamento o di errore del processo, se la notifica è stata configurata all'avvio del processo (Initiate Job (POST jobs)).

- Tipo: stringa

StatusCode

Il codice indicante lo stato del processo.

Valori validi: `InProgress` | `Succeeded` | `Failed`

- Tipo: stringa

StatusMessage

Il messaggio sullo stato del processo.

- Tipo: stringa

Livello

Il livello di accesso ai dati da utilizzare per il processo di recupero di archivio o Select.

Valori validi: `Expedited` | `Standard` | `Bulk`

- Tipo: stringa

VaultARN

L'ARN del vault di cui il processo è una sottorisorsa.

Tipo: stringa

Ulteriori informazioni

- Initiate Job (POST jobs)

Grant

Contiene informazioni su un'assegnazione.

Indice

Grantee

L'assegnatario.

Tipo: oggetto Grantee

Obbligatorio: no

Autorizzazione

L'autorizzazione fornita all'assegnatario.

Tipo: Stringa

Valori validi: `FULL_CONTROL` | `WRITE` | `WRITE_ACP` | `READ` | `READ_ACP`

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

Grantee

Contiene informazioni su un assegnatario.

Indice

DisplayName

Il nome visualizzato dell'assegnatario.

Tipo: Stringa

Obbligatorio: no

EmailAddress

L'indirizzo e-mail dell'assegnatario.

Tipo: Stringa

Obbligatorio: no

ID

L'ID utente canonico ID dell'assegnatario.

-Tipo: stringa

Obbligatorio: no

Tipo

Il tipo dell'assegnatario.

Tipo: Stringa

Valori validi: `AmazonCustomerByEmail` | `CanonicalUser` | `Group`

Obbligatorio: no

URI

L'URI del gruppo di assegnatari.

Tipo: Stringa

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

InputSerialization

Descrive il modo in cui l'archivio viene serializzato.

Indice

CSV

Un oggetto che descrive la serializzazione di un oggetto con codifica CSV.

Tipo: oggetto CSVInput

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

InventoryRetrievalJobInput

Fornisce opzioni per specificare un processo di recupero di intervallo di inventario.

Indice

EndDate

La data di fine (in formato UTC) dell'intervallo di date per un recupero di inventario vault che include gli archivi creati prima di quella data.

Valori validi: una rappresentazione di stringa nel formato data ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) in secondi, ad esempio `2013-03-20T17:03:43Z`.

Tipo: stringa. Una rappresentazione di stringa nel formato data ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) in secondi, ad esempio `2013-03-20T17:03:43Z`.

Obbligatorio: no

Format (Formato)

Il formato di output dell'elenco di inventari vault, definito dalla richiesta Initiate Job (POST jobs) all'avvio di un processo di recupero di un inventario vault.

Valori validi: `CSV` | `JSON`

Obbligatorio: no

Tipo: Stringa

Limite

Il numero massimo di elementi di inventario che possono essere restituiti per ogni richiesta di recupero di inventario vault.

Valori validi: un valore intero superiore o uguale a 1.

Tipo: Stringa

Obbligatorio: no

Marker

Una stringa opaca che rappresenta il punto da cui continuare la paginazione dei risultati relativi al recupero di inventario vault. Utilizza questo marcatore in una nuova richiesta `Initiate Job` per ottenere ulteriori elementi di inventario. Se non vi sono altri elementi di inventario, questo valore è null.

Tipo: Stringa

Obbligatorio: no

StartDate

La data di inizio (in formato UTC) dell'intervallo di date per un recupero di inventario vault che include gli archivi creati in quella data o successivamente.

Valori validi: una rappresentazione di stringa nel formato data ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) in secondi, ad esempio `2013-03-20T17:03:43Z`.

Tipo: stringa. Una rappresentazione di stringa nel formato data ISO 8601 (`YYYY-MM-DDThh:mm:ssTZD`) in secondi, ad esempio `2013-03-20T17:03:43Z`.

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

jobParameters

Fornisce opzioni per definire un processo.

Indice

ArchiveId

L'ID dell'archivio desiderato. Questo campo è obbligatorio se il campo `Type` è impostato su `select` o `archive-retrieval`. Se specifichi questo campo per una richiesta di processo di recupero di inventario, si verifica un errore.

Valori validi: deve essere un ID di archivio valido ottenuto da una precedente richiesta ad Amazon Glacier (Amazon Glacier).

- Tipo: stringa

Obbligatorio: sì quando `Type` è impostato su `select` o `archive-retrieval`.

Descrizione

La descrizione facoltativa del processo.

Valori validi: la descrizione non può superare i 1.024 byte. I caratteri consentiti sono ASCII a 7 bit senza codici di controllo, in particolare valori ASCII da 32 a 126 in decimale o da 0x20 a 0x7E in esadecimale.

- Tipo: stringa

Obbligatorio: no

Format (Formato)

(Facoltativo) Il formato di output all'avvio di un processo di recupero di un inventario vault. Se avvii un processo di inventario e non specifichi un campo `Format`, JSON è il formato di default.

Valori validi: `CSV` | `JSON`

- Tipo: stringa

Obbligatorio: no

InventoryRetrievalParameters

I parametri di input utilizzati per un recupero di intervallo di inventario.

Tipo: oggetto InventoryRetrievalJobInput

Obbligatorio: no

OutputLocation

Un oggetto che contiene informazioni sulla posizione dove vengono archiviati i risultati del processo Select.

Tipo: oggetto OutputLocation

Obbligatorio: sì per i processi `select`.

RetrievalByteRange

L'intervallo di byte da recuperare per un file`archive-retrieval`, nel formato "-». *`StartByteValue EndByteValue`* Se questo campo non è specificato, viene recuperato l'intero archivio. Se è specificato, l'intervallo di byte deve essere allineato in termini di megabyte (1024*1024). Allineato ai megabyte significa che StartByteValuedeve essere divisibile per 1 MB e EndByteValuepiù 1 deve essere divisibile per 1 MB o essere la fine dell'archivio specificata come valore della dimensione in byte dell'archivio meno 1. Se RetrievalByteRangenon è allineato ai megabyte, questa operazione restituisce una risposta. `400`

Se specifichi questo campo per una richiesta di processo `inventory-retrieval` o `select`, si verifica un errore.

-Tipo: stringa

Obbligatorio: no

SelectParameters

Un oggetto che contiene informazioni sui parametri utilizzati per una selezione.

Tipo: oggetto SelectParameters

Obbligatorio: no

SNSTopic

L'Amazon Resource Name (ARN) dell'argomento Amazon SNS in cui Amazon Glacier invia una notifica quando il processo è completato e l'output è pronto per il download. L'argomento specificato invia la notifica ai sottoscrittori.

L'argomento di SNS deve esistere. In caso contrario, Amazon Glacier non lo crea per te. Inoltre, questo argomento deve avere una policy che consente all'account che ha creato il processo di

pubblicare messaggi nell'argomento. Per informazioni sui nomi degli argomenti SNS, consulta CreateTopicAmazon Simple Notification Service API Reference.

- Tipo: stringa

Obbligatorio: no

Livello

Il livello da utilizzare per un processo di recupero di archivio o Select. `Standard` è il valore predefinito.

Valori validi: `Expedited` | `Standard` | `Bulk`

- Tipo: stringa

Obbligatorio: no

Tipo

Il tipo di processo. Puoi avviare un processo per eseguire una query Select su un archivio, recuperare un archivio o ottenere l'inventario di un vault.

Valori validi: `select` | `archive-retrieval` | `inventory-retrieval`

- Tipo: stringa

Obbligatorio: sì

Ulteriori informazioni

- Initiate Job (POST jobs)

OutputLocation

Contiene informazioni sulla posizione in cui vengono archiviati risultati ed errori del processo.

Indice

S3

Un oggetto che descrive una posizione in Amazon S3 in cui archiviare i risultati della richiesta di ripristino.

Tipo: S3Location

Obbligatorio: sì

Ulteriori informazioni

- Initiate Job (POST jobs)

OutputSerialization

Descrive il modo in cui l'output è serializzato.

Indice

CSV

Un oggetto che descrive la serializzazione dei risultati della query con codifica CSV.

Tipo: oggetto CSVOutput

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

S3Location

Contiene informazioni sulla posizione in Amazon S3 dove vengono archiviati i risultati del processo.

Indice

AccessControlList

Un elenco di assegnazioni che controllano l'accesso ai risultati archiviati.

Tipo: matrice di oggetti Grant

Obbligatorio: no

BucketName

Il nome del bucket Amazon S3 dove vengono archiviati i risultati del processo. Il bucket deve trovarsi nella stessa AWS regione del vault che contiene l'oggetto di archivio di input.

Tipo: Stringa

Obbligatorio: sì

CannedACL

La lista di controllo accessi (ACL) predefinita da applicare ai risultati del processo.

Tipo: Stringa

Valori validi: `private` | `public-read` | `public-read-write` | `aws-exec-read` | `authenticated-read` | `bucket-owner-read` | `bucket-owner-full-control`

Obbligatorio: no

Encryption (Crittografia)

Un oggetto che contiene informazioni sulla crittografia utilizzata per l'archiviazione dei risultati del processo in Amazon S3.

Tipo: oggetto Crittografia

Obbligatorio: no

Prefisso

Il prefisso anteposto ai risultati di questa richiesta. La lunghezza massima del prefisso è 512 byte.

Tipo: Stringa

Obbligatorio: sì

StorageClass

La classe di storage utilizzata per archiviare i risultati del processo.

Tipo: Stringa

Valori validi: `STANDARD` | `REDUCED_REDUNDANCY` | `STANDARD_IA`

Obbligatorio: no

Assegnazione di tag

Il set di tag applicato ai risultati del processo.

Tipo: mappatura stringa a stringa

Obbligatorio: no

UserMetadata

Una mappa di metadati da archiviare con i risultati del processo in Amazon S3.

Tipo: mappatura stringa a stringa

Obbligatorio: no

Ulteriori informazioni

- Initiate Job (POST jobs)

SelectParameters

Contiene informazioni sui parametri utilizzati per la selezione.

Indice

Expression

L'espressione utilizzata per selezionare l'oggetto. L'espressione non deve superare la quota di 128.000 caratteri.

Tipo: Stringa

Obbligatorio: sì

ExpressionType

Il tipo di espressione fornito, ad esempio `SQL`.

Valori validi: `SQL`

Tipo: Stringa

Obbligatorio: sì

InputSerialization

Descrive il formato di serializzazione dell'oggetto nella selezione.

Tipo: oggetto InputSerialization

Obbligatorio: no

OutputSerialization

Descrive il modo in cui i risultati del processo Select sono serializzati.

Obbligatorio: no

Tipo: oggetto OutputSerialization

Ulteriori informazioni

- Initiate Job (POST jobs)

Operazioni di recupero di dati

Di seguito sono elencate le operazioni relative al recupero dei dati disponibili in Amazon Glacier.

Argomenti

- Get Data Retrieval Policy (GET policy)
- List Provisioned Capacity (GET provisioned-capacity)
- Purchase Provisioned Capacity (POST provisioned-capacity)
- Set Data Retrieval Policy (PUT policy)

Get Data Retrieval Policy (GET policy)

Descrizione

Questa operazione restituisce la politica corrente di recupero dei dati per la AWS regione Account AWS e specificata nella richiesta. `GET` Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon Glacier.

Richieste

Per restituire la policy di recupero dati corrente, invia una richiesta HTTP `GET` all'URI della policy di recupero dati come illustrato nell'esempio di sintassi seguente.

Sintassi

```
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**
>
> Il `AccountId` valore è l'ID. Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o facoltativamente un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Sintassi

```
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
         }
       ]
    }
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

BytesPerHour

Il numero massimo di byte che possono essere recuperati in un'ora.

Questo campo è disponibile solo se il valore del campo Strategy è `BytesPerHour`.

Tipo: numero

Regole

La regola di policy. Sebbene si tratti di un tipo di elenco, attualmente esiste una sola regola, che contiene un campo Strategia e facoltativamente un campo. BytesPerHour

Tipo: matrice

Strategia

Il tipo di policy di recupero dati.

- Tipo: stringa

  Valori validi: `BytesPerHour`|`FreeTier`|None. `BytesPerHour` equivale alla selezione di Max Retrieval Rate (Tasso di recupero massimo) nella console. `FreeTier` equivale alla selezione di Free Tier Only (Solo piano gratuito) nella console. None equivale alla selezione di No Retrieval Policy (Nessuna policy di recupero) nella console. Per ulteriori informazioni sulla selezione di policy di recupero dati nella console, consulta Politiche di recupero dei dati di Amazon Glacier.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente mostra come ottenere una policy di recupero dati.

Richiesta di esempio

In questo esempio, una richiesta `GET` è inviata all'URI della posizione di una policy.

```
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
```

Risposta di esempio

In una risposta corretta la policy di recupero dati è visualizzata nel corpo della risposta in formato 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"
          }
       ]
    }
}
```

Sezioni correlate

- Set Data Retrieval Policy (PUT policy)
- Initiate Job (POST jobs)

List Provisioned Capacity (GET provisioned-capacity)

Questa operazione genera un elenco delle unità di capacità con provisioning per l' Account AWS specificato. Per ulteriori informazioni sulla capacità con provisioning, consultare Opzioni di recupero dall'archivio.

Un'unità di capacità assegnata dura per un mese a partire dalla data e l'ora dell'acquisto, che è la data di inizio. L'unità scade in corrispondenza della data di scadenza, che cade esattamente un mese dopo la data di inizio approssimata al secondo più vicino.

Se la data di inizio è il trentunesimo giorno di un mese, la data di scadenza sarà l'ultimo giorno del mese successivo. Ad esempio, se la data di inizio è il 31 agosto, la data di scadenza sarà il 30 settembre. Se la data di inizio è il 31 gennaio, la data di scadenza sarà il 28 febbraio. Puoi vedere questa funzionalità nella Risposta di esempio.

Sintassi della richiesta

Per elencare la capacità di recupero con provisioning di un account, invia una richiesta HTTP GET all'URI della capacità con provisioning come illustrato nella sintassi di esempio seguente.

```
GET /AccountId/provisioned-capacity HTTP/1.1
```

```
Host: glacier.Region.amazonaws.com
Date: Date
Authorization: SignatureValue
x-amz-glacier-version: 2012-06-01
```

> **Note**
>
> Il `AccountId` valore è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Se l'operazione riesce, il servizio restituisce una risposta HTTP `200 OK`.

Sintassi della risposta

```
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"
      }
}
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Il corpo della risposta contiene i campi JSON descritti di seguito.

CapacityId

L'ID che identifica l'unità di capacità con provisioning.

Tipo: stringa.

StartDate

La data di acquisto (in formato UTC) dell'unità di capacità con provisioning.

Tipo: stringa. Una rappresentazione di stringa nel formato di data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

ExpirationDate

La data di scadenza (in formato UTC) dell'unità di capacità con provisioning.

Tipo: stringa. Una rappresentazione di stringa nel formato di data ISO 8601, ad esempio `2013-03-20T17:03:43.221Z`.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

L'esempio seguente elenca le unità di capacità con provisioning per un account.

Richiesta di esempio

In questo esempio, una richiesta GET viene inviata per recuperare un elenco delle unità di capacità con provisioning per l'account specificato.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 200 OK` un elenco di unità di capacità assegnate per l'account, come mostrato nell'esempio seguente.

L'unità di capacità assegnata elencata per prima è un esempio di unità con data di inizio al 31 gennaio 2017 e data di scadenza al 28 febbraio 2017. Come indicato in precedenza, se la data di inizio è il trentunesimo giorno di un mese, la data di scadenza sarà l'ultimo giorno del mese successivo.

```
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" ",
      },
      ...
```

```
}
```

Sezioni correlate

- Purchase Provisioned Capacity (POST provisioned-capacity)

Purchase Provisioned Capacity (POST provisioned-capacity)

Questa operazione consente di acquistare un'unità di capacità con provisioning per un Account AWS.

Un'unità di capacità assegnata dura per un mese a partire dalla data e l'ora dell'acquisto, che è la data di inizio. L'unità scade in corrispondenza della data di scadenza, che cade esattamente un mese dopo la data di inizio approssimata al secondo più vicino.

Se la data di inizio è il trentunesimo giorno di un mese, la data di scadenza sarà l'ultimo giorno del mese successivo. Ad esempio, se la data di inizio è il 31 agosto, la data di scadenza sarà il 30 settembre. Se la data di inizio è il 31 gennaio, la data di scadenza sarà il 28 febbraio.

La capacità assegnata assicura che la capacità di recupero per effettuare recuperi di tipo Expedited sia disponibile al bisogno. Ogni unità di capacità garantisce la possibilità di eseguire almeno tre recuperi rapidi ogni cinque minuti e fornisce fino al 150% di velocità MB/s di recupero. Per ulteriori informazioni sulla capacità con provisioning, consultare Opzioni di recupero dall'archivio.

> **Note**
>
> È previsto un limite di due unità di capacità predisposte per. Account AWS

Richieste

Per acquistare un'unità di capacità assegnata e Account AWS inviare una `POST` richiesta HTTP all'URI di capacità fornita.

Sintassi

```
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**
>
> Il `AccountId` valore è l'ID. Account AWS Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Questa operazione non ha un corpo della richiesta.

Risposte

Se la richiesta riesce, il servizio restituisce una risposta HTTP `201 Created`.

Sintassi

```
HTTP/1.1 201 Created
x-amzn-RequestId: x-amzn-RequestId
Date: Date
x-amz-capacity-id: CapacityId
```

Intestazioni di risposta

Una risposta corretta include le seguenti intestazioni di risposta, oltre alle intestazioni di risposta comuni a tutte le operazioni. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Nome	Descrizione
`x-amz-capacity-id`	L'ID che identifica l'unità di capacità con provisioning. -Tipo: stringa

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Questa operazione include l'errore o gli errori seguenti, oltre ai possibili errori comuni a tutte le operazioni di Amazon Glacier. Per informazioni sugli errori di Amazon Glacier e un elenco di codici di errore, consulta. Risposte agli errori

Codice	Descrizione	Codice di stato HTTP	Tipo
`LimitExceededException`	Restituito se la richiesta supera il limite di unità di capacità con provisioning dell'account.	`400 Bad Request`	Client

Esempi

L'esempio seguente acquista la capacità con provisioning per un account.

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP POST per l'acquisto di un'unità di capacità con provisioning.

```
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
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) restituisce `HTTP 201 Created` una risposta, come illustrato nell'esempio seguente.

```
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
```

Sezioni correlate

- List Provisioned Capacity (GET provisioned-capacity)

Set Data Retrieval Policy (PUT policy)

Descrizione

Questa operazione imposta e quindi implementa una politica di recupero dei dati nella regione specificata nella AWS richiesta. PUT È possibile impostare una politica per AWS regione per un. Account AWS La policy viene implementata alcuni minuti dopo un'operazione PUT riuscita.

L'operazione di impostazione di policy non ha effetto sui processi di recupero in corso prima dell'implementazione della policy. Per ulteriori informazioni sulle policy di recupero dati, consulta Politiche di recupero dei dati di Amazon Glacier.

Richieste

Sintassi

Per impostare una policy di recupero dati, invia una richiesta HTTP PUT all'URI della policy di recupero dati come illustrato nell'esempio di sintassi seguente.

```
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**
>
> Il `AccountId` valore è l' Account AWS ID. Questo valore deve corrispondere all' Account AWS ID associato alle credenziali utilizzate per firmare la richiesta. Puoi specificare un Account AWS ID o, facoltativamente, un singolo '-' (trattino), nel qual caso Amazon Glacier utilizza Account AWS l'ID associato alle credenziali utilizzate per firmare la richiesta. Se specifichi l'ID del tuo account, non devi includere trattini ("-") nell'ID.

Parametri della richiesta

Questa operazione non utilizza parametri richiesta.

Intestazioni di richiesta

Questa operazione utilizza solo le intestazioni richiesta che sono comuni a tutte le operazioni. Per informazioni sulle intestazioni richiesta comuni, consulta Intestazioni di richiesta comuni.

Corpo della richiesta

Il corpo della richiesta contiene i campi JSON descritti di seguito.

BytesPerHour

Il numero massimo di byte che possono essere recuperati in un'ora.

Questo campo è obbligatorio solo se il valore del campo Strategy è `BytesPerHour`. L'operazione PUT verrà rifiutata se il campo Strategy non è impostato su `BytesPerHour` e imposti questo campo.

Tipo: numero

Obbligatorio: sì, se il campo Strategy è impostato su `BytesPerHour`. In caso contrario, no.

Valori validi: il valore intero minimo 1. Il valore intero massimo è compreso tra 2^63 e 1 (inclusi).

Regole

La regola di policy. Sebbene si tratti di un tipo di elenco, attualmente deve esistere una sola regola, che contiene un campo Strategia e facoltativamente un campo. BytesPerHour

Tipo: matrice

Campo obbligatorio: sì

Strategia

Il tipo di policy di recupero dati da impostare.

-Tipo: stringa

Campo obbligatorio: sì

Valori validi: `BytesPerHour`|`FreeTier`|None. `BytesPerHour` equivale alla selezione di Max Retrieval Rate (Tasso di recupero massimo) nella console. `FreeTier` equivale alla selezione di Free Tier Only (Solo piano gratuito) nella console. None equivale alla selezione di No Retrieval Policy (Nessuna policy di recupero) nella console. Per ulteriori informazioni sulla selezione di policy di recupero dati nella console, consulta Politiche di recupero dei dati di Amazon Glacier.

Risposte

Sintassi

```
HTTP/1.1 204 No Content
x-amzn-RequestId: x-amzn-RequestId
Date: Date
```

Intestazioni di risposta

Questa operazione utilizza solo le intestazioni di risposta che sono comuni alla maggior parte delle risposte. Per informazioni sulle intestazioni di risposta comuni, consulta Intestazioni di risposta comuni.

Corpo della risposta

Questa operazione non restituisce un corpo della risposta.

Errori

Per informazioni sulle eccezioni e sui messaggi di errore di Amazon Glacier, consulta. Risposte agli errori

Esempi

Richiesta di esempio

L'esempio seguente invia una richiesta HTTP PUT con il campo Strategy impostato su `BytesPerHour`.

```
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'esempio seguente invia una richiesta HTTP PUT con il campo Strategy impostato su `FreeTier`.

```
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'esempio seguente invia una richiesta HTTP PUT con il campo Strategy impostato su None.

```
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"
          }
       ]
    }
}
```

Risposta di esempio

Se la richiesta ha avuto esito positivo, Amazon Glacier (Amazon Glacier) imposta la politica e restituisce `HTTP 204 No Content` un come mostrato nell'esempio seguente.

```
HTTP/1.1 204 No Content
x-amzn-RequestId: AAABZpJrTyioDC_HsOmHae8EZp_uBSJr6cnGOLKp_XJCl-Q
Date: Wed, 10 Feb 2017 12:02:00 GMT
```

Sezioni correlate

- Get Data Retrieval Policy (GET policy)
- Initiate Job (POST jobs)

Cronologia dei documenti

- Versione corrente del prodotto: 01-06-2012

La tabella seguente descrive le modifiche importanti in ogni versione dell'Amazon Glacier Developer Guide dal 5 luglio 2018 in poi. Per ricevere notifiche sugli aggiornamenti di questa documentazione, è possibile sottoscrivere un feed RSS.

Modifica	Descrizione	Data
Tempi di avvio migliorati per le richieste di ripristino Standard effettuate tramite Operazioni in batch Amazon S3	I recuperi standard per le richieste di ripristino effettuate tramite Operazioni in batch Amazon S3 ora possono iniziare in pochi minuti. Per ulteriori informazioni consulta Opzioni di recupero dall'archivio.	9 agosto 2023
Amazon S3 supporta richieste di ripristino a velocità più elevate per le classi di archiviazione S3 Glacier Flexible Retrieval (Recupero flessibile S3 Glacier) e S3 Glacier Deep Archive (Archiviazione profonda S3 Glacier)	Amazon S3 supporta richieste di ripristino a una velocità massima di 1.000 transazioni al secondo, per Account AWS per le classi di archiviazione S3 Glacier Flexible Retrieval (Recupero flessibile S3 Glacier) e S3 Glacier Deep Archive (Archiviazione profonda S3 Glacier).	15 novembre 2022
Modifica del nome Amazon Glacier	Amazon Glacier è ora Amazon Glacier per rispecchiare meglio l'integrazione di Glacier con Amazon S3.	20 novembre 2018
Aggiornamenti ora disponibili tramite RSS	Ora puoi iscriverti a un feed RSS per ricevere notifiche	5 luglio 2018

sugli aggiornamenti della guida Amazon Glacier Developer Guide.

Aggiornamenti precedenti

La tabella seguente descrive le modifiche importanti apportate in ogni versione dell'Amazon Glacier Developer Guide prima del 5 luglio 2018.

Modifica	Descrizione	Data di rilascio
Recuperi di dati di tipo Expedited e Bulk	Amazon Glacier ora supporta il recupero rapido e in blocco dei dati oltre ai recuperi standard. Per ulteriori informazioni, consulta Opzioni di recupero dall'archivio.	21 novembre 2016
Vault Lock	Amazon Glacier ora supporta Vault Lock, che consente di implementare e applicare facilmente i controlli di conformità su singoli vault Amazon Glacier con una policy Vault Lock. Per ulteriori informazioni, consultare Serratura Amazon Glacier Vault e Policy di Vault Lock.	8 luglio 2015
Tagging di vault	Amazon Glacier ora ti consente di etichettare i tuoi vault Amazon Glacier per una gestione più semplice delle risorse e dei costi. I tag sono etichette che puoi definire e associare ai tuoi vault e l'utilizzo dei tag aggiunge funzionalità di filtro a operazioni come i report sui costi. AWS Per ulteriori informazioni, consultare Etichettare le risorse di Amazon Glacier e Etichettatura dei vault Amazon Glacier.	22 giugno 2015
Policy di accesso dei vault	Amazon Glacier ora supporta la gestione dell'acce sso ai tuoi vault Amazon Glacier individuali utilizzan do policy di accesso al vault. Questa funzionalità ti consente di definire una policy di accesso direttame nte su un vault e quindi di autorizzare più facilment e l'accesso di utenti, gruppi commerciali dell'orga	27 Aprile 2015

Modifica	Descrizione	Data di rilascio
	nizzazione e partner commerciali ai vault. Per ulteriori informazioni, consulta Policy di accesso delle vault.	
Policy di recupero dati e registrazione di controllo	Amazon Glacier ora supporta le politiche di recupero dei dati e la registrazione degli audit. Le policy di recupero dati ti consentono di impostare facilmente limiti di recupero di dati e di semplificare la gestione dei costi inerenti al recupero di dati. Puoi definire i tuoi limiti di recupero dei dati con pochi clic in Console di gestione AWS o utilizzando l'API Amazon Glacier. Per ulteriori informazioni, consulta Politiche di recupero dei dati di Amazon Glacier. Inoltre, Amazon Glacier ora supporta l'audit logging AWS CloudTrail con, che registra le chiamate API Amazon Glacier per il tuo account e invia i file di log a un bucket Amazon S3 da te specificato. Per ulteriori informazioni, consulta Registrazione delle chiamate API Amazon Glacier con AWS CloudTrail.	11 dicembre 2014
Aggiornamenti degli esempi Java	Gli esempi di codice Java in questa guida che utilizzano l' AWS SDK per Java sono stati aggiornati.	27 giugno 2014
Limitazione degli elementi di inventario vault recuperati	Puoi ormai limitare il numero di elementi di inventario vault recuperati filtrando per data di creazione dell'archivio o impostando un limite. Per ulteriori informazioni sulla limitazione degli elementi di inventario recuperati, consulta Recupero di un intervallo di inventario nell'argomento Initiate Job (POST jobs).	31 dicembre 2013
Rimosso (obsoleto) URLs	È stata rimossa la pagina URLs che rimandava alla vecchia pagina delle credenziali di sicurezza dagli esempi di codice.	26 luglio 2013

Modifica	Descrizione	Data di rilascio
Supporto per il recupero di intervalli	Amazon Glacier ora supporta il recupero di intervalli specifici dei tuoi archivi. Puoi avviare un lavoro richiedendo ad Amazon Glacier di preparare un intero archivio o una parte dell'archivio per il download successivo. Quando un archivio è di dimensioni molto grandi, può risultare conveniente avviare vari processi sequenziali per preparare l'archivio. Per ulteriori informazioni, consulta Scaricare un archivio in Amazon Glacier.	13 Novembre 2012
Nuova guida	Questa è la prima versione della Amazon Glacier Developer Guide.	20 agosto 2012

AWS Glossario

Per la AWS terminologia più recente, consultate il AWS glossario nella sezione Reference.Glossario AWS