

###---Amazon Glacier---###

-> Glacier is an extremely low-cost storage service that provides durable storage with security features for data archiving and backup.
 -> Data stored in Amazon Glacier in Archives.
 -> An archive is any object, such a photos, videos or document, which u stored in vault.
 -> A single archive can be as long as 40terabytes.
 -> You can store unlimited number of archive and unlimited amount of data in Amazon Glacier.
 -> Vault is used for long term archive, so Glacier used Vault instead of Buckets.
 -> Glacier stored Archives, S3 store objects.
 -> It having larger size then S3.
 -> Amazon S3 Glacier (Glacier) provides a management console, which you can use to create and delete vaults. However, all other interactions (Ex: upload) with Glacier require that you use the AWS Command Line Interface (CLI) or write code.

Amazon S3 Glacier Data Model:

- Vault
- Archive
- Job
- Notification Configuration

-> In terms of REST, vaults and archives are the resources.

Vault:

-> a vault is a container for storing archives.
 -> When you create a vault, you specify a name and choose an AWS Region where you want to create the vault.
 -> Each vault resource has a unique address.
 <https://<region-specific endpoint>/<account-id>/vaults/<vaultname>>
 Ex: <https://glacier.us-west-2.amazonaws.com/111122223333/vaults/examplevault>
 -> You can store an unlimited number of archives in a vault.

Archive:

-> An archive can be any data such as a photo, video, or document and is a base unit of storage in Glacier.
 -> Each archive has a unique ID and an optional description.
 -> Each archive has a unique address.
 <https://<region-specific endpoint>/<account-id>/vaults/<vault-name>/archives/<archive-id>>
 https://glacier.us-west-2.amazonaws.com/111122223333/vaults/examplevault/archives/NkbByEejwEggmBz2fTHgJrg0XBoDfjP4q6iu87-TjhgG6eGoOY9Z8i1_AUyUsuhPAdTqLHy8pTl5nfCFJmDl2yEZONi5L260mw12vcs01MNGntHEQL8MBfGlqrEXAMPLEArchiveId
 -> You can store an unlimited number of archives in a vault.

Job:

-> Glacier jobs can perform a select query on an archive, retrieve an archive, or get an inventory of a vault.
 -> When performing a query on an archive, you initiate a job providing a SQL query and list of Glacier archive objects.
 -> Glacier Select runs the query in place and writes the output results to Amazon S3.
 -> Retrieving an archive and vault inventory (list of archives) are asynchronous operations in Glacier in which you first initiate a job, and then download the job output after Glacier completes the job.
 -> Glacier offers a cold storage data archival solution.

Notification Configuration:

-> Because jobs take time to complete, Glacier supports a notification mechanism to notify you when a job is complete.
 -> You can configure a vault to send notification to an Amazon Simple Notification Service (Amazon SNS) topic when jobs complete.
 -> Glacier stores the notification configuration as a JSON document.

Supported Operations in Glacier:

-> Amazon S3 Glacier (Glacier) supports a set of operations. Among all the supported operations, only the following operations are asynchronous:

- Retrieving an archive
- Retrieving a vault inventory (list of archives)

-> These operations require you to first initiate a job and then download the job output.

Vault Operations:

- > Glacier provides operations to create and delete vaults.
- > You can obtain a vault description for a specific vault or for all vaults in a region.
- > The vault description provides information such as creation date, number of archives in the vault, total size in bytes used by all the archives in the vault, and the date Glacier generated the vault inventory.

Archive Operations

- > Glacier provides operations for you to upload and delete archives.
- > You cannot update an existing archive; you must delete the existing archive and upload a new archive.
- > Note that each time you upload an archive, Glacier generates a new archive ID.

Jobs

- > You can initiate a Glacier job to perform a select query on an archive, retrieve an archive, or get an inventory of a vault.
- > The following are the types of Glacier jobs:
 - > select- Perform a select query on an archive.
 - > archive-retrieval- Retrieve an archive.
 - > inventory-retrieval- Inventory a vault.

Accessing Amazon S3 Glacier

- > Amazon S3 Glacier (Glacier) is a RESTful web service that uses HTTP and HTTPS as a transport and JSON as a message serialization format.
- > Your application code can make requests directly to the Glacier web service API.
- > When using the REST API directly, you must write the necessary code to sign and authenticate your requests.

create a vault, upload and download an archive, and delete the archive and the vault:

Step 1: Create a Vault in Amazon S3 Glacier

- 1) Sign into the AWS Management Console and open the Glacier console at <https://console.aws.amazon.com/glacier/>.
- 2) Select a region from the region selector.
- 3) If you are using Glacier for the first time, click Get started. (Otherwise, you would click Create Vault.)
- 4) Enter examplevault as the vault name in the Vault Name field and then click Next Step.
- 5) Select Do not enable notifications. For this getting started exercise, you will not configure notifications for the vault.

NOTE: If you wanted to have notifications sent to you or your application whenever certain Glacier jobs complete, you would select Enable notifications and create a new SNS topic or Enable notifications and use an existing SNS topic to set up Amazon Simple Notification Service (Amazon SNS) notifications.

In subsequent steps, you upload an archive and then download it using the high-level API of the AWS SDK. Using the high-level API does not require that you configure vault notification to retrieve your data.
- 6) If the region and vault name are correct, then click Submit.
- 7) Your new vault is listed on the Glacier Vaults page.

(OR)

- 1) Create a vault with the create-vault command (CLI).


```
$ aws glacier create-vault --account-id - --vault-name myvault
{
  "location": "/123456789012/vaults/myvault"
}
```

NOTE: All S3 Glacier commands require an account ID parameter. Use the hyphen character (--account-id -) to use the current account.

Step-2: Upload an Archive to a Vault in Amazon S3 Glacier

(a) Preparing a File for Uploading:

- > Create a file for the test upload. The following commands create a file named largefile that contains exactly 3 MiB of random data.


```
$ dd if=/dev/urandom of=largefile bs=3145728 count=1
1+0 records in
```

```

1+0 records out
3145728 bytes (3.1 MB) copied, 0.205813 s, 15.3 MB/s
-> "dd" is a utility that copies a number of bytes from an input file to an output file.
-> Next, split the file into 1 MiB (1,048,576 byte) chunks.
$ split --bytes=1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'

```

(b) Initiating a Multipart Upload and Upload Files

(i) Create a multipart upload in Amazon S3 Glacier by using the initiate-multipart-upload command.

```

$ aws glacier initiate-multipart-upload --account-id - --archive-description
"multipart upload test" --part-size 1048576 --vault-name myvault
{
  "uploadId":
  "19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLqyFu7sY1_1R7vgFu
  JV6NtcV5zpsJ",
  "location":
  "/123456789012/vaults/myvault/multipart-uploads/19gaRezEXAMPLES6Ry5YYdqthHOC_
  kGRCT03L9yetr220UmPtBYKk-OssZtLqyFu7sY1_1R7vgFuJV6NtcV5zpsJ"
}

```

NOTE: S3 Glacier requires the size of each part in bytes (1 MiB in this example), your vault name, and an account ID to configure the multipart upload.

The AWS CLI outputs an upload ID when the operation is complete. Save the upload ID to a shell variable for later use.

```

$
UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLqyFu7sY1_1R
7vgFuJV6NtcV5zpsJ"

```

(ii) Next, use the upload-multipart-part command to upload each of the three parts.

```

$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkaa --range
'bytes 0-1048575/*' --account-id - --vault-name myvault

```

```

{
  "checksum":
  "elf2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}

```

```

$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkab
--range 'bytes 1048576-2097151/*' --account-id - --vault-name myvault

```

```

{
  "checksum":
  "elf2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}

```

```

$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkac
--range 'bytes 2097152-3145727/*' --account-id - --vault-name myvault

```

```

{
  "checksum":
  "elf2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}

```

(c) Completing the Upload:

-> Amazon S3 Glacier requires a tree hash of the original file to confirm that all of the uploaded pieces reached AWS intact.

-> To calculate a tree hash, you must split the file into 1 MiB parts and calculate a binary SHA-256 hash of each piece.

-> Then you split the list of hashes into pairs, combine the two binary hashes in each pair, and take hashes of the results.

-> Repeat this process until there is only one hash left. If there is an odd number of hashes at any level, promote it to the next level without modifying it.

-> The key to calculating a tree hash correctly when using command line utilities is to store each hash in binary format and convert to hexadecimal only at the last step.

-> Combining or hashing the hexadecimal version of any hash in the tree will cause an incorrect result.

(1) To calculate a tree hash:

(i) If you haven't already, split the original file into 1 MiB parts.

```

$ split --bytes=1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'

```

- ```
(ii) Calculate and store the binary SHA-256 hash of each chunk.
$ openssl dgst -sha256 -binary chunkaa > hash1
$ openssl dgst -sha256 -binary chunkab > hash2
$ openssl dgst -sha256 -binary chunkac > hash3
(iii) Combine the first two hashes and take the binary hash of the result.
$ cat hash1 hash2 > hash12
$ openssl dgst -sha256 -binary hash12 > hash12hash
(iv) Combine the parent hash of chunks aa and ab with the hash of chunk ac and hash
the result, this time outputting hexadecimal. Store the result in a shell variable.
$ cat hash12hash hash3 > hash123
$ openssl dgst -sha256 hash123
SHA256(hash123)= 9628195fcdcbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
$ TREEHASH=9628195fcdcbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
(2) Finally, complete the upload with the complete-multipart-upload command.
-> This command takes the original file's size in bytes, the final tree hash value
in hexadecimal, and your account ID and vault name.
$ aws glacier complete-multipart-upload --checksum $TREEHASH --archive-size 3145728
--upload-id $UPLOADID --account-id - --vault-name myvault
{
 "archiveId":
 "d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrAllGAAONJAzo5QdP-N83MKqd96Unspoa5H51ItWX-sK8-QS0ZhwsyGiu9-R-kwWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg",
 "checksum": "9628195fcdcbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
 "location":
 "/123456789012/vaults/myvault/archives/d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrAllGAAONJAzo5QdP-N83MKqd96Unspoa5H51ItWX-sK8-QS0ZhwsyGiu9-R-kwWUyS1dSB1mgPPWkEbeFfqDSav053rU7FvVLHfRc6hg"
}
(3) You can also check the status of the vault using the describe-vault command.
$ aws glacier describe-vault --account-id - --vault-name myvault
{
 "SizeInBytes": 3178496,
 "VaultARN": "arn:aws:glacier:us-west-2:123456789012:vaults/myvault",
 "LastInventoryDate": "2018-12-07T00:26:19.028Z",
 "NumberOfArchives": 1,
 "CreationDate": "2018-12-06T21:23:45.708Z",
 "VaultName": "myvault"
}

NOTE: Vault status is updated about once per day.
(4) Now it's safe to remove the chunk and hash files that you created.
$ rm chunk* hash*
```

(OR)

## Step-2: Upload an Archive to a Vault in Amazon S3 Glacier

- ```
-> Upload an Archive to a Vault using "upload-archive" command
-> This operation adds an archive to a vault.
-> This is a synchronous operation, and for a successful upload, your data is durably
persisted.
-> Amazon S3 Glacier returns the archive ID in the x-amz-archive-id header of the
response.
-> You must use the archive ID to access your data in Amazon S3 Glacier.
-> After you upload an archive, you should save the archive ID returned so that you can
retrieve or delete the archive later.
```

```
$ aws glacier upload-archive --account-id - --vault-name my-vault --body archive.zip
{
  "archiveId":
    "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtrDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-AJVLu2ccmDSyDUMZwKwbpbAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
  "checksum": "969fb39823836d81f0cc028195fcdcbcbbe76cdde932d4646fa7de5f21e18aa67",
  "location":
    "/0123456789012/vaults/my-vault/archives/kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtrDvc2VkJPSDtfKmQrj0IRQLSGsNuDp-AJVLu2ccmDSyDUMZwKwbpbAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
}
```

Scenario: Moving the objects from S3 to Glacier

- 1) Create the S3 bucket (This can be used for backup)
- 2) upload the backup files into this bucket, (select and upload into this bucket)
- 3) Now you want some rule/machinesum to move into Glacier from time to time.
 - > This should be versioning turned ON
 - > Goto bucket properties -> Versioning -> Enable Versioning
 - > Goto bucket properties -> Lifecycle -> Create a rule (Add rule)
 - > Choose rule target: whole bucket:<backup_bucket_name>
 - > Configure Rule:
 - > Transition to the standard - infrequent access storage class [30]days after object creation date.
 - > Archive to the Glacier Storage class [60]days after object creation date.
 - > Expire [360]Days after object creation date.
- 4) Review and Name -> Create and Activate Rule.

-> You can move the things from one storage class to other automatically.
->

Glacier Commands:

- > abort-multipart-upload:
 - > This operation aborts a multipart upload identified by the upload ID.
 - \$ aws glacier abort-multipart-upload --account-id - --vault-name my-vault --upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ
 - > This command does not produce any output.
- > abort-vault-lock:
 - > This operation aborts the vault locking process if the vault lock is not in the Locked state.
 - > If the vault lock is in the Locked state when this operation is requested, the operation returns an AccessDeniedException error.
 - > Aborting the vault locking process removes the vault lock policy from the specified vault.
 - \$aws glacier abort-vault-lock --account-id - --vault-name MyVaultName
 - > This command produces no output.
- > add-tags-to-vault:
 - > This operation adds the specified tags to a vault. Each tag is composed of a key and a value. Each vault can have up to 10 tags. If your request would cause the tag limit for the vault to be exceeded, the operation throws the LimitExceededException error. If a tag already exists on the vault under a specified key, the existing key value will be overwritten.
 - \$ aws glacier add-tags-to-vault --account-id - --vault-name my-vault --tags id=1234,date=july2015
- > complete-multipart-upload:
 - > You call this operation to inform Amazon S3 Glacier (Glacier) that all the archive parts have been uploaded and that Glacier can now assemble the archive from the uploaded parts
 - \$ aws glacier complete-multipart-upload --archive-size 3145728 --checksum 9628195fcdcbcbbe76cdde456d4646fa7de5f219fb39823836d81f0cc0e18aa67 --upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ --account-id - --vault-name my-vault
- > complete-vault-lock: This operation completes the vault locking process by transitioning the vault lock from the InProgress state to the Locked state
 - \$ aws glacier complete-vault-lock --account-id - --vault-name MyVaultName --lock-id 9QZgEXAMPLEPhvL6xEXAMPLE

```

-> create-vault: You can create up to 1,000 vaults per account.
    $ aws glacier create-vault --vault-name my-vault --account-id -

-> delete-archive:
    $ aws glacier delete-archive --account-id <value> --vault-name <value> --archive-id
    <value>

-> delete-vault:
    $ aws glacier delete-vault --vault-name my-vault --account-id -
-> delete-vault-access-policy

-> delete-vault-notifications
-> describe-job
-> describe-vault
-> get-data-retrieval-policy
-> get-job-output
-> get-vault-access-policy
-> get-vault-lock
-> get-vault-notifications
-> initiate-job:
    -> The following command initiates a job to get an inventory of the vault my-vault:
        $ aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
        '{"Type": "inventory-retrieval"}'
        {
            "location":
            "/0123456789012/vaults/my-vault/jobs/zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-RO4
            7Yc6FxsdsGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
            "jobId":
            "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-RO47Yc6FxsdsGBgf_Q2DK5Ejh18CnTS5XW4_XqlN
            HS61dsO4CnMW"
        }
    -> The following command initiates a job to retrieve an archive from the vault my-vault:
        $ aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
        file://job-archive-retrieval.json
    -> job-archive-retrieval.json
        {
            "Type": "archive-retrieval",
            "ArchiveId":
            "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtrDvc2VkpSDtfKmQrj0IRQLSGsNuDp-A
            JVlu2ccmDSyDumZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
            "Description": "Retrieve archive on 2015-07-17",
            "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-topic"
        }

-> initiate-multipart-upload
-> initiate-vault-lock
-> list-jobs:
    $ aws glacier list-jobs --account-id - --vault-name my-vault

-> list-multipart-uploads
-> list-parts
-> list-provisioned-capacity
-> list-tags-for-vault
-> list-vaults
    $ aws glacier list-vaults --account-id -

-> purchase-provisioned-capacity
-> remove-tags-from-vault
-> set-data-retrieval-policy
-> set-vault-access-policy
-> set-vault-notifications
-> upload-archive
-> upload-multipart-part
-> wait

```

NOTE: Retrieving an archive and vault inventory (list of archives) are asynchronous operations in Glacier in which you first initiate a job, and then download the job output after Glacier completes the job.

Q) Amazon Glacier is designed for: [C]

- (a) Active database Storage (b) Cached session data (C) Data archive (d) Frequently accessed data.

Q) What storage Classe(s) is/are available in S3 to save data? [B]

- (a) Reduced Redundancy (b) Amazon Glacier (C) Bucket (D) Objects

Reference Links:

-> Getting Started with Amazon S3 Glacier

->

<https://docs.aws.amazon.com/amazonglacier/latest/dev/amazon-glacier-getting-started.html?shortFooter=true>

-> Glacier CLI

-> <https://docs.aws.amazon.com/cli/latest/reference/glacier/index.html>