AWS STORAGE

AWS offers a complete range of cloud services to support both application and archival compliance requirements. Select from the objects, files, and block storage services as well as cloud data migration to start designing the foundation of your cloud IT environments.

Types of storage:

AWS offers 5 types of storage services such as:

1. Simple Storage Service (S3)
2. Elastic File System (EFS)
3. Elastic Block Store (EBS)
4. Glacier
5. Snowball

Difference between Object Storage and Block Storage:

Block Storage:

- Block storage is suitable for transitional databases, random read/write loads and structured database storage.
- Block storage divides the data to be stored in evenly sized blocks called data chunks for instance, a file can be split into evenly sized blocks before it is stored.
- Data blocks stored in block storage would not contain metadata. (Data created, data modified, content type etc.)
- Block storage only keeps the address (index number) where the data blocks are stored, it does not care what is in that block, just how to retrieve it when required.

Object Storage:

- Object storage stores the files as a whole and does not divide them.
- In object storage an object is: the file/ data itself, its Meta data, object global unique ID.
- The object global unique ID is a unique identifier for the object (can be the object name itself) and it must be unique such that it can be retrieved disregarding where it's physical storage location is.
- Object storage cannot be mounted as a drive.
- Example of object storage solutions are Dropbox, AWS S3, Facebook.

1. Simple Storage Service (S3):

- S3 is a storage for the internet. It has a simple web service interface for simple storing and retrieving of any amount of data, anytime from anywhere on the internet.
- S3 is object based storage.
- You cannot install operating system on S3.
- S3 has a distributed data store architecture where objects are redundantly stored in multiple locations. (minimum 3 locations in same region)
- Data is stored in bucket.
- A bucket is a flat container of objects.
- Maximum capacity of a bucket is 5TB.
- You can create folders in your bucket (available through console)
- You cannot create nested buckets.
- Bucket ownership is non transferrable.
- S3 bucket is region specific.
- You can have up to 100 buckets per account. (may expand on request)

S3 Bucket Naming Rules:

- S3 bucket names (keys) are globally unique across all AWS regions.
- Bucket names cannot be change after they are created.
- If bucket is deleted its name become available again to you or other account to use.
- Bucket names must be at least 3 and no more than 63 characters long.
- Bucket names are part of URL used to access a bucket.
- Bucket name must be a series of one or more labels (xyz bucket)
- Bucket names can contain lowercase, numbers and hyphen but cannot use uppercase letters.
- Bucket name should not be an IP address.
- Each label must start and end with a lowercase letter or a number.
- By default buckets and its objects are private, and by default only owner can access the bucket.

S3 Bucket Sub-Resources:

Sub-resources of S3 bucket includes:

Lifecycle: to decide on objects lifecycle management.

Website: to hold configurations related to static website hosted in S3 buckets.

Versioning: keep objects versions as it changes (set updated)

Access Control List: bucket policies

The name is simply two parts: bucket region’s end point / bucket name

Example: for S3 bucket named mybucket in Europe west region is

https://s3-eu-west1.amazonaws.com/mybucket

S3 Objects:

- An object size stored in an S3 bucket can be 0 byte to 5TB.
- Each object is stored and retrieve by unique key. (ID or name)
- An object in AWS S3 is uniquely identified and addressed through:
  - service endpoint
  - bucket name
  - object key (name)
  - optionally object version
- Object stored in a S3 bucket in a region will never leave that region unless you specifically move them to another region or CRR.
- A bucket owner can grant cross account permissions to another AWS account (or users in another account) to upload objects.
- You can grant S3 bucket / object permission to:
  - Individual users
  - AWS account
  - Make the resource public
  - To all authenticate user

S3 Bucket Versioning:

- Bucket versioning is a S3 bucket sub resource used to protect against accidental object/data deletion or overwrites.
- Versioning can also be used for data retention and archive.
- Once you enable versioning on a bucket it cannot be disabled however it can be suspended.
- When enable, bucket versioning will protect existing and new objects and maintains their versions as they are updated.
- Updating objects refers to PUT, POST, COPY, DELETE actions on objects.
- When versioning is enable and you try to delete an object a delete marker is placed on the object.
- You can still view the object and delete the marker.
- If you reconsider deleting the objects you can delete the delete marker and the object will be enable again.
- You will be charged for all S3 storage cost for all object versions stored.
- You can use versioning with S3 lifecycle policies to delete older version or you can move them to a cheaper S3 storage (Glacier.)
- Bucket version state:-
  - Enabled
  - Suspended

    - Un-versioned
- Versioning applies to all objects in a bucket and not partially applied.
- Object existing before enable versioning will have a version ID or NULL.
- If you have a bucket that is already versioned then you suspended versioning existing objects and their versions remain as it is.
- However they will not be updated/ version further with future updates while the bucket versioning is suspended.
- New objects (uploaded after suspension) they will have a version ID "null" if the same key (name) is used to stone another objects it will override the existing one.
- An object deletion in a suspended versioning buckets will only delete the objects with ID "null".

S3 Bucket Versioning-MFA Delete:

- Multifactor authentication delete is a versioning capacity that adds another level of security in case your account is compromised.
- This adds another layer of security for the following:
    - Changing your bucket's versioning state.
    - Permanently deleting on objects version.
- MFA delete requires:
    - Your security credentials.
    - The code displayed on an approved physical or s/w based authentication device.

S3 Multipart Upload:

- It is used to upload an object in parts.
- Parts are uploaded independently and in parallel in any order.
- It is recommended for objects sizes of 100MB or larger.
- You must use it for objects larger than 5GB.
- This is done though S3 multipart upload API.

Copying S3 Objects:

- The copy operation creates a copy of an objects that is already stored in Amazon S3.
- You can create a copy of your object up to 5GB in size a single atomic operation.
- However to copy an object greater then 5GB you must use the multipart upload API.
- Incur charges if copy to another region.

Use the copy operation to:

- Generate additional copies of the subjects.
- Renaming object (copy to a new name)
- Changing the copy's storage class or encrypt it at rest.
- Move object across AWS location/region.

- Change object metadata.

STORAGE CLASSES OF AMAZON S3:

There are 6 types of storage classes of Amazon S3 is available such as:

1. Amazon S3 Standard
2. Amazon S3 Glacier Deep Archive
3. Amazon Glacier
4. Amazon S3 Standard Infrequent Access
5. Amazon S3 one-zone-IA
6. Amazon S3 Intelligent Tiering

1. Amazon S3 Standard:

- S3 standard offers high durability, availability and performance object storage for frequently accessed data.
- Durability is 99.999999999%.
- Designed for 99.99% availability over a given year.
- Supports SSL for data in transit and encryption of data at rest.
- The storage cost for the object is fairly high but there is very less charge for accessing the objects.
- Largest object that can be uploaded in a single PUT in 5GB.

2. Amazon S3 IA (standard):

- S3-IA is for data that is accessed less frequently but requires rapid access when needed.
- The storage cost is much cheaper than S3-standard almost half the price, but you are charged more heavily for accessing your objects.
- Durability is 99.999999999%.
- Resilient against events that impact an entire AZ.
- Availability is 99.9% in a year.
- Supports SSL for data in transit and encryption of data at rest.
- Data that is deleted from S3-IA within 30 days will be charged for a full 30 days.
- Backed with the Amazon S3 service level agreement for availability.

3. Amazon S3 Intelligent Tiering:

- The S3 intelligent tiering storage class is designed to optimize cost by automatically moving data to the most cost effective access tier.

- It works by storing objects in two access tiers.
- If an object in the frequent access tier is accessed it is automatically moved back to the frequent access tier.
- There is no retrieval fees when using the S3 intelligent tiering storage class and no additional tering fees when objects are moved between access tiers.
- Same low latency and high performance of S3 standard.
- Objects less than 128kb cannot move to IA.
- Durability is99.999999999%.
- Availability is 99.9%.

4. Amazon One-Zone IA

- S3 one zone IA is for data that is accessed less frequently but requires rapid access when needed.
- Data store is single AZ.
- Ideal for those who want lower cost option of IA data.
- It is good choice for storing secondary backup copies of on-premise data of easily re-creatable data.
- You can use S3 lifecycle policies.
- Durability is 99.999999999%.
- Availability is 99.5%.
- Because S3 one zone IA stores data in a single AZ, data stored in this storage class will be lost in the event of AZ destruction.

5. Amazon S3 Glacier:

- S3 glacier is a secure, durable, low cost storage class for data archiving.
- To keep cost low yet suitable for varying needs S3 glacier provides three retrieval options that ranges from a few minutes to hours.
- You can upload object directly to glacier or use lifecycle policies.
- Durability is 99.999999999%.
- Data is resilient in the event of one entire AZ destruction.
- Supports SSL for data in transit and encryption data at rest.
- You can retrieve 10GB of your amazon S3 glacier data per month for free with free tier account.

6. Amazon S3 Glacier Deep Archive:

- S3 glacier deep archive is amazon S3 cheapest storage.
- Design to retain data for long period even if for 10 years.
- All objects stored in S3 glacier deep archive are replicated and stored across at least at three geographically AZ.

- Durability is 99.999999999%.
- Ideal alternative to magnetic tape libraries.
- Retrieval time within 12 hours.
- Storage cost is up to 75% less than for the existing S3 glacier storage class.
- Availability is 99.9%.

ELASTIC BLOCK STORE (EBS)

There are two types of block store devices are available for EC2.

1. Elastic Block Store (persistent, network attached virtual drive)
2. Instances Store Backed EC2:
   - Basically the virtual hard drive on the host allocated to this EC2 instance.
   - Limit to 10GB per device
   - Ephemeral storage (non-persistent storage)
   - The EC2 instance can’t be stopped, can only be rebooted or terminated. Terminate will delete data.

- ➢ EBS volume behaves like RAW, unformatted, external block storage devices that you can attached to your EC2 instance.
- ➢ EBS volumes are block storage devices suitable for database style data that requires frequent reads and writes.
- ➢ EBS volumes are attached to your EC2 instances through the AWS network, like virtual hard drive.
- ➢ An EBS volume can attach to a single EC2 instances only at a time.
- ➢ Both EBS volumes and EC2 instances must be in the same AZ.
- ➢ An EBS volume data is replicated by AWS across multiple servers in the same AZ to prevent data loss resulting from any single AWS component failure.

EBS Volume Types:

1. SSD backed volume
2. HDD backed volume
3. Magnetic standard

SSD backed volume is also two types:

A. General purpose SSD (GP2)
B. Provisioned IOPS SSD (io1)

HDD backed volume is also two types:

A. Throughput optimized HDD (st1)
B. Cold HDD (SC1)

A. General Purposed SSD (gp2)

- GP2 is the default EBS volume type for the amazon EC2 instance.
- GP2 volumes are backed by SSDs.
- General purpose balances both price and performances.
- Ratio of 3IOPS/GB with up to 10,000 IOPS.
- Boot volume having low latency.
- Volume size: 1 GB to 16 GB.
- Price: $0.10/ GB/month

B. Provisioned IOPS SSD (io1)

- These volumes are ideal for both IOPS intensive and throughput intensive workloads that requires extremely low latency or for mission critical applications.
- Designed for I/O intensive applications such as large relational or NoSQL databases.
- Use if you need more than 10,000 IOPS.
- Can provision up to 32,000 IOPS per volume.
- Volume size: 4GB to 16TB
- Price : $ 0.125/GB/month

C. Throughput optimized HDD (st1)

- ST1 is backed by hard disk drives and is ideal for frequently accessed, throughput intensive workloads with large datasets.
- ST1 volumes deliver performance in term of throughput, measured in MB/S.
- Big data, data warehouse, log processing.
- It cannot be a boot volume.
- Can provisioned up to 500 IOPS per volume.
- Volume size: 500GB to 16 TB
- Price: $0.045/GB/month

D. Cold HDD (SC1)

- SC1 is also backed by HDD and provides the lowest cost per GB of all EBS volume types.
- Lowest cost storage for infrequent access workloads.
- Used in file servers.
- Cannot be a boot volume.
- Can provisioned up to 250 IOPS per volume.
- Volume size: 500 GB to 16TB

- Price: $0.025/GB/Month

Magnetic Standard:

- Lowest cost per GB of all EBS volume type that is bootable.
- Magnetic volumes are ideal for workloads where data is accessed infrequently and applications where the lowest storage cost is important.
- Price: $0.05/GB/month
- Volume size: 1GB to 1TB
- Max IOPS/volume: 40-200

EBS Snapshot of Root Volume and Non-root Volume:

- EBS snapshots are point-in-time images/copies of your EBS volume.
- Any data written to the volume after the snapshot process is initiated, will not be included in the resulting snapshot (but will be included in future incremental update.)
- Per AWS account up to 5000 EBS volumes can be created.
- Per account up to 10,000 EBS snapshots can be created.
- EBS snapshots are stored on S3, however you cannot access them directly. You can only access them through EC2 APIs.
- While EBS volumes are AZ specific, snapshots are region specific.
- Any AZ in region can use snapshot to create EBS volume.
- To migrate an EBS from one AZ to another, create a snapshot (region specific) and create an EBS volume from the Snapshot in the intended AZ.
- You can create a snapshot to an EBS volume of the same or larger size than the original volumes size from which the snapshot was initially created.

- You can take a snapshot of a non-root EBS volume while the volumes is in use on a running EC2 instance.
- This means, you can still access it while the snapshot is being processed.
- However the snapshot will only include data that is already written to your volume.
- The snapshot is created immediately but it may stay in pending status until the full snapshot is completed. This may takes few hours to complete specially for the first time snapshot if a volume.
- During the period when the snapshot status is pending you can still access the volume (non-root) but I/O might be slower because of the snapshot activity.
- While in pending state, an in progress snapshot will not include data from ongoing reads and writes to the volume.
- To take complete snapshot of your non-root EBS volume: stop of unmounts the volume.
- To create a snapshot for a root EBS volume you must stop the instance first then take the snapshot.

Incremental Snapshot:

- EBS snapshots are stored incrementally.
- For low cost storage on S3 and a guarantee to be able to able fully restore data from the snapshot.
- What you need is a single snapshot then further snapshot will only carry the changed blocks (incremental updates).
- Therefore you do not need to have multiple full/complete copies of the snapshot.
- You are charged for:
  - Data transferred to S3 from your EBS volume you are taking snapshot.
  - Snapshot stored in S3.
  - First snapshot is a clone, subsequent snapshots are incremental.
  - Deleting snapshot will only remove data exclusive to that snapshot.

EBS Encryption:

- EBS encryption is supported on all EBS volume types and all EC2 instance families.
- Snapshots of encrypted volumes are also encrypted.
- Creating an EBS volume from an encrypted snapshot will result in an encrypted volume.
- Data encryption at rest means encrypting data while it is stored on the data storage device.
- There are many ways you can encrypt data on an EBS volume at rest, while the volume is attached to an EC2 instance:
  - Use 3rd party EBS volume
  - Encryption tods.
  - Use encrypted EBS volumes.
  - Use encrypted at the O.S level.

- Encrypt data at the application level before storing it to the volume.
- Use encrypt file system on the top of the EBS volume.
- Encrypt volume area accessed exactly like unencrypted ones, basically encryption is handled transparently.
- You can attach an encrypted and unencrypted volumes to the same EC2 instance.
- Remember that the EBS volumes area not physically attached to the EC2 instance, rather they are virtually attached through the EBS infrastructure.
- This means when you encrypt data on an EBS volume data is actually encrypted on the EC2 instance then transferred, encrypted to be stored on the EBS volume.
- This means data in transit between EC2 and encrypted EBS volume is also encrypted.
- There is no direct way to change the encryption state of the volume.
- To change the state you need to follow either of the following two ways:
  - Attach a new encrypted EBS volume to the EC2 instance that has the data to be encrypted.
  - Mount the new volume to the EC2 instance.
  - Copy the data for the un-encrypted volume to the new volume.

- Both volumes must be on the same EC2 instance.

Or

- Create a snapshot of the unencrypted volume.
- Copy the snapshot and choose encryption for the new copy, this will create an encrypted copy of the snapshot.
- Use this new copy to create an EBS volume which will be encrypt too.
- Attach the new encrypted EBS volume to the EC2 instance.

Root EBS Volume Encryption:

- There is no direct way to change the encryption state of a volume.
- There is an indirect work around to this:
    - Launch the instance with the EBS volume required.
    - Do whatever patching of install applications.
    - Create an AMI from the EC2 instance.
    - Copy the AMI and choose encryption while copying.
    - This results it an encrypted AMI that is private (yours only).
    - Use the encrypted AMI to launch new EC2 instances which will have their EBS root volume3 encrypted.

EBS Encryption Key:

- To encrypt a volume or snapshot, you need an encryption key, these keys are called customer master key (CMK) and are managed by AWS key management service (KMS).
- When encrypting the first EBS volume, AWS KMS creates a default CMK key.
- This key is used for your first volume encryption of snapshots created from this volumes and subsequent volumes created from these snapshots.
- After that each newly encrypted volume is encrypted with a unique/ separate AES-256 bit encryption key. This key is used to encrypt the volume, its snapshot and any volumes created of its snapshots.

Changing Encryption Key:

- You cannot change the encryption (CMK) key used to encrypt an existing encrypted snapshot or encrypted EBS volume.
- If you want to change the key, create a copy of the snapshot and specify during the copy process that you want to re-encrypt the copy with a different key.

- This comes in handy when you have a snapshot that was encrypted using your default CMK key and you want to change the key in order to able to share the snapshot with other accounts.

Sharing EBS Snapshot:

- By default only the account owner can create volumes from the account snapshots.
- You can share your unencrypted snapshots with the AWS community by making them public.
- Also you can share your unencrypted snapshots with a selected AWS account by making them private then selecting the AWS accounts to share with.
- You cannot make your encrypted snapshots public.
- You cannot make a snapshot of an encrypted EBS volume public on AWS.
- You can share your encrypted snapshot with specific AWS account as follows:
    - Make sure that you use a non-default/custom CMK key to encrypt the snapshot, not the default CMK key (AWS will not allow the sharing if default CMK is used.)
    - Configure cross account permissions in order to give the account with which you want to share the snapshot access to the custom CMK key used to encrypt the snapshot.
    - Without this the other account will not be able to copy the snapshots nor will be able to create volumes of the snapshots.

- AWS will not allow you to share snapshots encrypted using your default CMK key.
- For the AWS account with whom an encrypted snapshot is shared.
    - They must first create their own copies of the snapshot.
    - Then they use that copy to restore/create EBS volume.
- You can make a copy of the snapshot when it has been fully saved to S3 (its status show as complete) and not during the snapshot’s pending status (when data blocks are being moved to S3).
- Amazon S3 server side encryption (SSE) protect the snapshot data-in-transit while copying.
- You can have up to 5 snapshots copy request running in a single destination per account.