S3-Simple Storage Service

- Amazon Simple Storage Service – S3 is a simple key, value object store designed for the Internet
- provides unlimited storage space and works on the pay-as-you-use model. Service rates get cheaper as the usage volume increases
- offers an extremely durable, highly available, and infinitely scalable data storage infrastructure at very low costs.
- is Object-level storage (not Block level storage like EBS volumes) and cannot be used to host OS or dynamic websites.
- S3 resources *e.g. buckets and objects* are private by default.

S3 Buckets & Objects

S3 Buckets

- A bucket is a container for objects stored in S3
- Buckets help organize the S3 namespace.
- A bucket is owned by the AWS account that creates it and helps identify the account responsible for storage and data transfer charges.
- Bucket names are globally unique, regardless of the AWS region in which it was created and the namespace is shared by all AWS accounts
- Even though S3 is a global service, buckets are created within a region specified during the creation of the bucket.
- Every object is contained in a bucket
- There is no limit to the number of objects that can be stored in a bucket and no difference in performance whether a single bucket or multiple buckets are used to store all the objects
- The S3 data model is a flat structure i.e. there are no hierarchies or folders within the buckets. However, logical hierarchy can be inferred using the key name prefix e.g. Folder1/Object1
- Restrictions
  - 100 buckets (soft limit) and a maximum of 1000 buckets can be created in each AWS account
  - Bucket names should be globally unique and DNS compliant
  - Bucket ownership is not transferable
  - Buckets cannot be nested and cannot have a bucket within another bucket
  - Bucket name and region cannot be changed, once created
- Empty or a non-empty buckets can be deleted

Objects

Objects are the fundamental entities stored in a bucket

- An object is uniquely identified within a bucket by a key name and version ID (if S3 versioning is enabled on the bucket)
- Object metadata cannot be modified after the object is uploaded and it can be only modified by performing copy operation and setting the metadata
- Objects belonging to a bucket that reside in a specific AWS region never leave that region, unless explicitly copied using Cross Region Replication
- Each object can be up to 5 TB in size
- An object can be retrieved as a whole or a partially
- With Versioning enabled, current as well as previous versions of an object can be retrieved

AWS S3 Storage Classes

- S3 storage classes allow lifecycle management for automatic transition of objects for cost savings.
- All S3 storage classes provide the same durability, first-byte latency, and support SSL encryption of data in transit, and data encryption at rest.
- S3 also regularly verifies the integrity of the data using checksums and provides the auto-healing capability.

S3 Standard

- STANDARD is the **default storage class**, if none specified during upload
- Low latency and high throughput performance
- Designed for **99.999999999% i.e. 11 9's Durability** of objects across AZs
- Designed for 99.99% availability over a given year
- Resilient against events that impact an entire Availability Zone and is designed to **sustain the loss of data in a two facilities**
- Ideal for performance-sensitive use cases and frequently accessed data
- S3 Standard is appropriate for a wide variety of use cases, including cloud applications, dynamic websites, content distribution, mobile and gaming applications, and big data analytics.

S3 Intelligent Tiering (S3 Intelligent-Tiering)

- S3 Intelligent Tiering storage class is designed to **optimize storage costs** by automatically moving data to the most cost-effective storage access tier, without performance impact or operational overhead.
- Delivers automatic cost savings by moving data on a granular object-level between two access tiers
  - one tier that is optimized for frequent access and
  - another lower-cost tier that is optimized for infrequently accessed data.
- a frequent access tier and a lower-cost infrequent access tier, when access patterns change.
- For a small monthly monitoring and automation fee per object, S3 monitors access patterns of the objects and moves objects that have not been accessed for 30 consecutive days to the infrequent access tier.
- There are no separate retrieval fees when using the Intelligent Tiering storage class. If an object in the infrequent access tier is accessed, it is automatically moved back to the frequent access tier.

S3 Standard-Infrequent Access (S3 Standard-IA)

- S3 Standard-Infrequent Access storage class is optimized for long-lived and **less frequently accessed** data. for e.g. *for backups and older data where access is limited, but the use case still demands high performance*

- Ideal for use for the primary or only copy of data that can't be recreated.
- Data stored redundantly across multiple geographically separated AZs and are resilient to the **loss of an Availability Zone**.
- offers greater availability and resiliency than the ONEZONE_IA class.
- Objects are available for real-time access.

S3 One Zone-Infrequent Access (S3 One Zone-IA)

- S3 One Zone-Infrequent Access storage classes are designed for long-lived and infrequently accessed data, but available for millisecond access (similar to the STANDARD and STANDARD_IA storage class).
- Ideal when the data can be recreated if the AZ fails, and for object replicas when setting cross-region replication (CRR).
- Stores the object data in **only one AZ**, which makes it less expensive than Standard-Infrequent Access
- Data is **not resilient** to the physical loss of the AZ resulting from disasters, such as earthquakes and floods.
- One Zone-Infrequent Access storage class is as durable as Standard-Infrequent Access, but it is less available and less resilient.

S3 Glacier Instant Retrieval

- Use for archiving data that is rarely accessed and requires milliseconds retrieval.
- Storage class has a **minimum storage duration period of 90 days**

S3 Glacier Flexible Retrieval – S3 Glacier

- S3 GLACIER storage class is suitable for **low-cost data archiving** where data access is infrequent and retrieval time of minutes to hours is acceptable.
- Storage class has a **minimum storage duration period of 90 days**
- Provides **configurable retrieval times**, from minutes to hours
  - Expedited retrieval: 1-5 mins
  - Standard retrieval: 3-5 hours
  - Bulk retrieval: 5-12 hours

S3 Glacier Deep Archive

- Glacier Deep Archive storage class provides **the lowest-cost data archiving** where data access is infrequent and retrieval time of hours is acceptable.
- Has a **minimum storage duration period of 180 days** and can be accessed at a default retrieval time of 12 hours.
- Supports long-term retention and digital preservation for data that may be accessed once or twice a year
- Ideal alternative to magnetic tape libraries

S3 Versioning

- S3 Versioning helps to keep multiple variants of an object in the same bucket and can be used to preserve, retrieve, and restore every version of every object stored in the S3 bucket.
- S3 Object Versioning can be used to protect from unintended overwrites and accidental deletions
- As Versioning maintains multiple copies of the same objects as a whole and charges accrue for multiple versions *for e.g. for a 1GB file with 5 copies with minor differences would consume 5GB of S3 storage space and you would be charged for the same.*
- Buckets can be in one of the three states
  - Unversioned (the default)
  - Versioning-enabled

    - Versioning-suspended
- S3 Object Versioning is not enabled by default and has to be explicitly enabled for each bucket.
- Versioning once enabled, cannot be disabled and can only be suspended
- Versioning enabled on a bucket applies to all the objects within the bucket
- Permissions are set at the version level. Each version has its own object owner; an AWS account that creates the object version is the owner. So, you can set different permissions for different versions of the same object.
- Irrespective of the Versioning, each object in the bucket has a version.
    - For Non Versioned bucket, the version ID for each object is null
    - For Versioned buckets, a unique version ID is assigned to each object

Object Retrieval

- For Non Versioned bucket
    - An Object retrieval always returns the only object available.
- For Versioned bucket
    - An object retrieval returns the Current latest object.
    - Non-Current objects can be retrieved by specifying the version ID.

Object Addition

- For Non Versioned bucket
    - If an object with the same key is uploaded again it overwrites the object
- For Versioned bucket
    - If an object with the same key is uploaded, the newly uploaded object becomes the current version and the previous object becomes the non-current version.
    - A non-current versioned object can be retrieved and restored hence protecting against **accidental overwrites**

Object Deletion

- For Non Versioned bucket
    - An object is permanently deleted and cannot be recovered
- For the Versioned bucket,
    - All versions remain in the bucket and Amazon inserts a **delete marker** which becomes the Current version
    - A non-current versioned object can be retrieved and restored hence protecting against **accidental overwrites**
    - If an Object with a specific version ID is deleted, a permanent deletion happens and the object cannot be recovered

Delete marker

- Delete Marker object does not have any data or ACL associated with it, just the key and the version ID
- An object retrieval on a bucket with a delete marker as the Current version would return a 404
- Only a DELETE operation is allowed on the Delete Marker object
- If the Delete marker object is deleted by specifying its version ID, the previous non-current version object becomes the current version object

S3 Object Lifecycle Management

- S3 Object lifecycle can be managed by using a lifecycle configuration, which defines how S3 manages objects during their lifetime.
- Lifecycle configuration enables simplification of object lifecycle management, *for e.g. moving of less frequently access objects, backup or archival of data for several years, or permanent deletion of objects,*
- S3 controls all transitions automatically
- S3 Object lifecycle management allows 2 types of behavior
  - **Transition** in which the storage class for the objects changes
  - **Expiration** where the objects expire and are permanently deleted
- Lifecycle Management can be configured with Versioning, which allows storage of one current object version and zero or more non-current object versions

AWS S3 Security

- AWS S3 Security is a shared responsibility between AWS and the Customer
- S3 is a fully managed service that is protected by the AWS global network security procedures
- S3 Encryption supports both data at rest and data in transit encryption.
  - Data in transit encryption can be provided by enabling communication via SSL or using client-side encryption
  - Data at rest encryption can be provided using Server Side or Client Side encryption

Resource-Based policies

- Bucket policies and access control lists (ACLs) are resource-based because they are attached to the S3 resources

Bucket Policies

- Bucket policy can be used to grant cross-account access to **other AWS accounts or IAM users in other accounts** for the bucket and objects in it.
- Bucket policies provide centralized, access control to buckets and objects based on a variety of conditions, including S3 operations, requesters, resources, and aspects of the request (e.g. IP address)

Access Control Lists (ACLs)

- Each bucket and object has an ACL associated with it
- An ACL is a list of grants identifying grantee and permission granted
- ACLs are used to grant basic read/write permissions on resources to **other AWS accounts**.

AWS S3 Encryption

- AWS S3 Encryption supports both data at rest and data in transit encryption.
- Data in-transit
  - S3 allows protection of data in transit by enabling communication via SSL or using client-side encryption
- Data at Rest
  - Server-Side Encryption
    - S3 encrypts the object before saving it on disks in its data centers and decrypt it when the objects are downloaded
  - Client-Side Encryption
    - data is encrypted at the client-side and uploaded to S3.
    - the encryption process, the encryption keys, and related tools are managed by the user.

S3 Object Lock

- S3 Object Lock helps to store objects using a write-once-read-many (WORM) model.
- can help prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely.

S3 Access Points

- S3 access points simplify data access for any AWS service or customer application that stores data in S3.
- Access points are named network endpoints that are attached to buckets and can be used to perform S3 object operations, such as GetObject and PutObject.
- Each access point has distinct permissions and network controls that S3 applies for any request that is made through that access point.

- Each access point enforces a customized access point policy that works in conjunction with the bucket policy, attached to the underlying bucket.
- An access point can be configured to accept requests only from a VPC to restrict S3 data access to a private network.
- Custom block public access settings can be configured for each access point.

S3 Pricing

- S3 costs vary by Region
- Charges are incurred for
  - Storage – cost is per GB/month
  - Requests – per request cost varies depending on the request type GET, PUT
  - Data Transfer
    - **data transfer-in is free**
    - data transfer out is charged per GB/month (except in the same region or to Amazon CloudFront)