**Amazon S3**

**What is S3?**

S3 is one of the first services that has been produced by aws.

S3 stands for Simple Storage Service.

S3 provides developers and IT teams with secure, durable, highly scalable object storage.

It is easy to use with a simple web services interface to store and retrieve any amount of data from anywhere on the web.

S3 is a safe place to store the files.

It is Object-based storage, i.e., you can store the images, word files, pdf files, etc.

The files which are stored in S3 can be from 0 Bytes to 5 TB.

It has unlimited storage means that you can store the data as much you want.

Files are stored in Bucket. A bucket is like a folder available in S3 that stores the files.

S3 is a universal namespace, i.e., the names must be unique globally. Bucket contains a DNS address. Therefore, the bucket must contain a unique name to generate a unique DNS address.

**Advantages of Amazon S3**

**Create Buckets:** Firstly, we create a bucket and provide a name to the bucket. Buckets are the containers in S3 that stores the data. Buckets must have a unique name to generate a unique DNS address.

**Storing data in buckets**: Bucket can be used to store an infinite amount of data. You can upload the files as much you want into an Amazon S3 bucket, i.e., there is no maximum limit to store the files. Each object can contain up to 5 TB of data. Each object can be stored and retrieved by using a unique developer assigned-key.

**Download data:** You can also download your data from a bucket and can also give permission to others to download the same data. You can download the data at any time whenever you want.

**Permissions:** You can also grant or deny access to others who want to download or upload the data from your Amazon S3 bucket. Authentication mechanism keeps the data secure from unauthorized access.

**Standard interfaces:** S3 is used with the standard interfaces REST and SOAP interfaces which are designed in such a way that they can work with any development toolkit.

**Security:** Amazon S3 offers security features by protecting unauthorized users from accessing your data.

**Amazon S3 Concepts**

* Buckets
* Objects
* Keys
* Regions
* Data Consistency Model

**Buckets**

A bucket is a container used for storing the objects.

Every object is incorporated in a bucket.

For example, if the object named photos/tree.jpg is stored in the tree image bucket, then it can be addressed by using the URL http://treeimage.s3.amazonaws.com/photos/tree.jpg.

A bucket has no limit to the amount of objects that it can store. No bucket can exist inside of other buckets.

S3 performance remains the same regardless of how many buckets have been created.

The AWS user that creates a bucket owns it, and no other AWS user cannot own it. Therefore, we can say that the ownership of a bucket is not transferrable.

The AWS account that creates a bucket can delete a bucket, but no other AWS user can delete the bucket.

**Objects**

Objects are the entities which are stored in an S3 bucket.

An object consists of object data and metadata where metadata is a set of name-value pair that describes the data.

An object consists of some default metadata such as date last modified, and standard HTTP metadata, such as Content type. Custom metadata can also be specified at the time of storing an object.

It is uniquely identified within a bucket by key and version ID.

**Key**

A key is a unique identifier for an object.

Every object in a bucket is associated with one key.

An object can be uniquely identified by using a combination of bucket name, the key, and optionally version ID.

For example, in the URL http://jtp.s3.amazonaws.com/2019-01-31/Amazons3.wsdl where "jtp" is the bucket name, and key is "2019-01-31/Amazons3.wsdl"

**Regions**

You can choose a geographical region in which you want to store the buckets that you have created.

A region is chosen in such a way that it optimizes the latency, minimize costs or address regulatory requirements.

Objects will not leave the region unless you explicitly transfer the objects to another region.

**Data Consistency Model**

Amazon S3 replicates the data to multiple servers to achieve high availability.

* Two types of model:

**Read-after-write consistency for PUTS of new objects**.

For a PUT request, S3 stores the data across multiple servers to achieve high availability.

A process stores an object to S3 and will be immediately available to read the object.

A process stores a new object to S3, it will immediately list the keys within the bucket.

It does not take time for propagation, the changes are reflected immediately.

**Eventual consistency for overwrite PUTS and DELETES**

For PUTS and DELETES to objects, the changes are reflected eventually, and they are not available immediately.

If the process replaces an existing object with the new object, you try to read it immediately. Until the change is fully propagated, the S3 might return prior data.

If the process deletes an existing object, immediately try to read it. Until the change is fully propagated, the S3 might return the deleted data.

If the process deletes an existing object, immediately list all the keys within the bucket. Until the change is fully propagated, the S3 might return the list of the deleted key.

**AWS Storage Classes**

S3 storage classes are used to assist the concurrent loss of data in one or two facilities.

S3 storage classes maintain the integrity of the data using checksums.

S3 provides lifecycle management for the automatic migration of objects for cost savings.

S3 contains four types of storage classes:

* S3 Standard
* S3 Standard IA
* S3 one zone-infrequent access
* S3 Glacier

**S3 Standard**

Standard storage class stores the data redundantly across multiple devices in multiple facilities.

It is designed to sustain the loss of 2 facilities concurrently.

Standard is a default storage class if none of the storage class is specified during upload.

It provides low latency and high throughput performance.

It designed for 99.99% availability and 99.999999999% durability

**S3 Standard IA**

IA stands for infrequently accessed.

Standard IA storage class is used when data is accessed less frequently but requires rapid access when needed.

It has a lower fee than S3, but you will be charged for a retrieval fee.

It is designed to sustain the loss of 2 facilities concurrently.

It is mainly used for larger objects greater than 128 KB kept for atleast 30 days.

It provides low latency and high throughput performance.

It designed for 99.99% availability and 99.999999999% durability

**S3 one zone-infrequent access**

S3 one zone-infrequent access storage class is used when data is accessed less frequently but requires rapid access when needed.

It stores the data in a single availability zone while other storage classes store the data in a minimum of three availability zones. Due to this reason, its cost is 20% less than Standard IA storage class.

It is an optimal choice for the less frequently accessed data but does not require the availability of Standard or Standard IA storage class.

It is a good choice for storing the backup data.

It is cost-effective storage which is replicated from other AWS region using S3 Cross Region replication.

It has the same durability, high performance, and low latency, with a low storage price and low retrieval fee.

It designed for 99.5% availability and 99.999999999% durability of objects in a single availability zone.

It provides lifecycle management for the automatic migration of objects to other S3 storage classes.

The data can be lost at the time of the destruction of an availability zone as it stores the data in a single availability zone.

**S3 Glacier**

S3 Glacier storage class is the cheapest storage class, but it can be used for archive only.

You can store any amount of data at a lower cost than other storage classes.

S3 Glacier provides three types of models:

Expedited: In this model, data is stored for a few minutes, and it has a very higher fee.

Standard: The retrieval time of the standard model is 3 to 5 hours.

Bulk: The retrieval time of the bulk model is 5 to 12 hours.

You can upload the objects directly to the S3 Glacier.

It is designed for 99.999999999% durability of objects across multiple availability zones.

**Features:**

**Versioning**

Versioning is a means of keeping the multiple forms of an object in the same S3 bucket. Versioning can be used to retrieve, preserve and restore every version of an object in S3 bucket.

For example, bucket consists of two objects with the same key but with different version ID's such as photo.jpg (version ID is 11) and photo.jpg (version ID is 12).

Versioning-enabled buckets allow you to recover the objects from the deletion or overwrite. It serves two purposes:

If you delete an object, instead of deleting the object permanently, it creates a delete marker which becomes a current version of an object.

If you overwrite an object, it creates a new version of the object and also restores the previous version of the object.

**Cross Region Replication**

Cross Region Replication is a feature that replicates the data from one bucket to another bucket which could be in a different region.

It provides asynchronous copying of objects across buckets. Suppose X is a source bucket and Y is a destination bucket. If X wants to copy its objects to Y bucket, then the objects are not copied immediately.

**Some points to be remembered for Cross Region Replication**

Create two buckets: Create two buckets within AWS Management Console, where one bucket is a source bucket, and other is a destination bucket.

Enable versioning: Cross Region Replication can be implemented only when the versioning of both the buckets is enabled.

Amazon S3 encrypts the data in transit across AWS regions using SSL: It also provides security when data traverse across the different regions.

Already uploaded objects will not be replicated: If any kind of data already exists in the bucket, then that data will not be replicated when you perform the cross region replication.

**Use cases of Cross Region Replication**

Compliance Requirements

By default, Amazon S3 stores the data across different geographical regions or availability zone to have the availability of data. Sometimes there could be compliance requirements that you want to store the data in some specific region. Cross Region Replication allows you to replicate the data at some specific region to satisfy the requirements.

Minimize Latency

Suppose your customers are in two geographical regions. To minimize latency, you need to maintain the copies of data in AWS region that are geographically closer to your users.

Maintain object copies under different ownership: Regardless of who owns the source bucket, you can tell to Amazon S3 to change the ownership to AWS account user that owns the destination bucket. This is referred to as an owner override option.

**Lifecycle Management**

Lifecycle Management is used so that objects are stored cost-effectively throughout their lifecycle. A lifecycle configuration is a set of rules that define the actions applied by S3 to a group of objects.

The lifecycle defines two types of actions:

**Transition actions**: When you define the transition to another storage class. For example, you choose to transit the objects to Standard IA storage class 30 days after you have created them or archive the objects to the Glacier storage class 60 days after you have created them.

**Expiration actions**: You need to define when objects expire, the Amazon S3 deletes the expired object on your behalf.

Suppose business generates a lot of data in the form of test files, images, audios or videos and the data is relevant for 30 days only. After that, you might want to transition from standard to standard IA as storage cost is lower. After 60 days, you might want to transit to Glacier storage class for the longtime archival. Perhaps you want to expire the object after 60 days completely, so Amazon has a service known as Lifecycle Management, and this service exist within S3 bucket.

**Lifecycle policies:**

Use Lifecycle rules to manage your object: You can manage the Lifecycle of an object by using a Lifecycle rule that defines how Amazon S3 manages objects during their lifetime.

**Automate transition to tiered storage**: Lifecycle allows you to transition objects to Standard IA storage class automatically and then to the Glacier storage class.

**Expire your objects**: Using Lifecycle rule, you can automatically expire your objects.