

# SIMPLE STORAGE SERVICE (S3)

## What is Amazon S3?

Amazon S3 is a **Simple Storage Service** in AWS that stores files of different types like Photos, Audio, and Videos as **Objects** providing more scalability and security to. It allows the users to store and retrieve any amount of data at any point of time from anywhere on the web. It facilitates features such as **extremely high availability, security, and simple connection to other AWS Services**.

## Why Amazon S3 is Used?

Amazon S3 is used for various purposes in the Cloud because of its robust features with scaling and securing of data. It helps people with all kinds of use cases from fields such as **Mobile/Web applications, Big data, Machine Learning** and many more. The following are a few Wide Usage of Amazon S3 service.

- **Data Storage**: Amazon S3 acts as the best option for scaling both small and large storage applications. It helps in **storing and retrieving the data-intensive applications** as per needs in ideal time.
- **Backup and Recovery**: Many Organizations are using Amazon S3 to **backup** their critical data and **maintain the data durability and availability** for recovery needs.
- **Hosting Static Websites**: Amazon S3 facilitates in storing **HTML, CSS and other web content** from Users/developers allowing them for hosting Static Websites benefiting with low-latency access and cost-effectiveness.
- **Data Archiving**: **Amazon S3 Glacier** service integration helps as a **cost-effective solution for long-term data storing which are less frequently accessed applications**.

- **Big Data Analytics**: Amazon S3 is often considered as data lake because of its capacity to store large amounts of both structured and unstructured data offering seamless integration with other AWS Analytics and AWS Machine Learning Services.

## **What is an Amazon S3 bucket?**

Amazon S3 bucket is a fundamental Storage Container feature in AWS S3 Service. It provides a secure and scalable repository for storing of Objects such as Text data, Images, Audio and Video files over AWS Cloud. Each S3 bucket name should be named globally unique and should be configured with ACL (Access Control List).

## **How Does Amazon S3 works?**

Amazon S3 works on organizing the data into unique S3 Buckets, customizing the buckets with Access controls. It allows the users to store objects inside the S3 buckets with facilitating features like versioning and lifecycle management of data storage with scaling. The following are a few main features of Amazon s3:

### **1. Amazon S3 Buckets and Objects**

**Amazon S3 Bucket**: Data in S3 is stored in containers called buckets. Each bucket will have its own set of policies and configurations. This enables users to have more control over their data. Bucket Names must be unique. Can be thought of as a parent folder of data. There is a limit of 100 buckets per AWS account. But it can be increased if requested by AWS support.

**Amazon S3 Objects**: Fundamental entity type stored in AWS S3. You can store as many objects as you want to store. The maximum size of an AWS S3 bucket is **5TB**. It consists of the following:

- Key
- Version ID
- Value

- Metadata
- Subresources
- Access control information
- Tags

## 2. Amazon S3 Versioning and Access Control

**S3 Versioning:** Versioning means always keeping a record of previously uploaded files in S3. Points to Versioning are not enabled by default. Once enabled, it is enabled for all objects in a bucket. Versioning keeps all the copies of your file, so, it adds cost for storing multiple copies of your data. For example, 10 copies of a file of size 1GB will have you charged for using 10GBs for S3 space. Versioning is helpful to prevent unintended overwrites and deletions. Objects with the same key can be stored in a bucket if versioning is enabled (since they have a unique version ID).

**Access control lists (ACLs):** A document for verifying access to S3 buckets from outside your AWS account. An ACL is specific to each bucket. You can utilize S3 Object Ownership, an Amazon S3 bucket-level feature, to manage who owns the objects you upload to your bucket and to enable or disable ACLs.

## 3. Bucket policies and Life Cycles

**Bucket Policies:** A document for verifying the access to S3 buckets from within your AWS account, controls which services and users have what kind of access to your S3 bucket. Each bucket has its own Bucket Policies.

**Lifecycle Rules:** This is a cost-saving practice that can move your files to AWS Glacier (The AWS Data Archive Service) or to some other S3 storage class for cheaper storage of old data or completely delete the data after the specified time.

## 4. Keys and Null Objects

**Keys:** The key in S3 is a unique identifier for an object in a bucket. For example in a bucket 'ABC' your GFG.java file is stored

at javaPrograms/GFG.java then 'javaPrograms/GFG.java' is your object key for GFG.java.

**Null Object:** Version ID for objects in a bucket where versioning is suspended is null. Such objects may be referred to as null objects.

## How To Use an Amazon S3 Bucket?

You can use the Amazon S3 buckets by following the simple steps which are mentioned below.

**Step 1:** Login into the Amazon account with your credentials and search for S3 and click on the S3. Now click on the option which is "Create bucket" and configure all the options which are shown while configuring.

**Step 2:** After configuring the AWS bucket now upload the objects into the buckets based upon your requirement. By using the AWS console or by using AWS CLI following is the command to upload the object into the AWS S3 bucket.

```
aws s3 cp <local-file-path> s3://<bucket-name>/
```

**Step 3:** You can control the permissions of the objects which was uploaded into the S3 buckets and also who can access the bucket. You can make the bucket public or private by default the S3 buckets will be in private mode.

**Step 4:** You can manage the S3 bucket lifecycle management by transitioning. Based upon the rules that you defined S3 bucket will be transitioning into different storage classes based on the age of the object which is uploaded into the S3 bucket.

**Step 5:** You need to turn to enable the services to monitor and analyze S3. You need to enable the S3 access logging to record who was requesting the objects which are in the S3 buckets.



## What are the types of S3 Storage Classes?

AWS S3 provides multiple storage types that offer different performance and features and different cost structures.

- **Standard**: Suitable for frequently accessed data, that needs to be highly available and durable.
- **Standard Infrequent Access (Standard IA)**: This is a cheaper data-storage class and as the name suggests, this class is best suited for storing infrequently accessed data like log files or data archives. Note that there may be a per GB data retrieval fee associated with the Standard IA class.
- **Intelligent Tiering**: This service class classifies your files automatically into frequently accessed and infrequently accessed and stores the infrequently accessed data in infrequent access storage to save costs. This is useful for unpredictable data access to an S3 bucket.
- **One Zone Infrequent Access (One Zone IA)**: All the files on your S3 have their copies stored in a minimum of 3 Availability Zones. One Zone IA stores this data in a single availability zone. It is only recommended to use this storage class for infrequently accessed, non-essential data. There may be a per GB cost for data retrieval.
- **Reduced Redundancy Storage (RRS)**: All the other S3 classes ensure the durability of 99.999999999%. RRS only ensures 99.99% durability. AWS no longer recommends RRS due to its less durability. However, it can be used to store non-essential data.

## How to Access Amazon S3 Bucket?

You can work and access the Amazon S3 bucket by using any one of the following methods

1. AWS Management Console

## 2. AWS CLI Commands

### 3. Programming Scripts ( Using boto3 library of Python )

#### 1. AWS Management Console

You can access the AWS S3 bucket using the AWS management console which is a **web-based user interface**. Firstly you need to create an AWS account and login to the Web console and from there you can choose the S3 bucket option from Amazon S3 service. ( AWS Console >> Amazon S3 >> S3 Buckets )

#### 2. AWS CLI Commands

In this methods firstly you have to **install the AWS CLI software in the terminal and try on configuring the aws account with access key, secret key and the default region**. Then on taking the ``aws - help``, you can figure out the s3 service usage. For example : To view try on running following command:

```
aws s3 ls
```

#### 3. Programming scripts

You can configure the Amazon S3 bucket by **using a scripting programming languages like Python and with using libraries such as boto3 library** you can perform the AWS S3 tasks.

## AWS S3 Bucket Permissions

You can manage the permission of S3 buckets by using several methods following are a few of them.

**1. Bucket Policies:** Bucket policies can be **attached directly to the S3 bucket and they are in JSON format which can perform the bucket level operations**. With the help of bucket policies, you can grant permissions to the users who can access the objects present in the bucket. If you grant permissions to any user he can download, and upload the objects to the bucket. You can create the bucket policy by using Python.

**2. Access Control Lists (ACLs):** ACLs are legacy access control mechanisms for S3 buckets **instead of ACLs we are using the bucket policies to control the permissions of the S3 bucket.** By using ACL you can grant the read, and access to the S3 bucket or you can make the objects public based on the requirements.

**3. IAM Policies:** IAM policies are mostly used to manage the permissions to the users and groups and resources available in the AWS by using the **IAM roles options.** You can attach an IAM policy to an IAM entity (user, group, or role) granting them access to specific S3 buckets and operations.  
The **most effective way** to control the permissions to the S3 buckets is **by using bucket policies.**

## Features of Amazon S3

- **Durability:** AWS claims Amazon S3 to have a **99.999999999%** of durability (11 9's). This means the possibility of losing your data stored on S3 is **one in a billion.**
- **Availability:** AWS ensures that the **up-time of AWS S3 is 99.99% for standard access.**
  - Note that availability is related to being able to access data and durability is related to losing data altogether.
- **Server-Side-Encryption (SSE):** AWS S3 supports three types of SSE models:
  - **SSE-S3:** AWS S3 manages encryption keys.
  - **SSE-C:** The customer manages encryption keys.
  - **SSE-KMS:** The AWS Key Management Service (KMS) manages the encryption keys.
- **File Size support:** AWS S3 can **hold files of size ranging from 0 bytes to 5 terabytes.** A 5TB limit on file size should not be a blocker for most of the applications in the world.

- **Infinite storage space**: Theoretically AWS S3 is supposed to have **infinite storage space**. This makes S3 infinitely scalable for all kinds of use cases.
- **Pay as you use**: The users are **charged according to the S3 storage they hold**.

## **Advantages of Amazon S3**

1. **Scalability**: Amazon S3 can be **scalable horizontally which makes it handle a large amount of data**. It can be **scaled automatically without human intervention**.
2. **High availability**: Amazon S3 bucket is famous for its **high availability nature you can access the data whenever you required it from any region**. It offers a **Service Level Agreement (SLA)** guaranteeing **99.9% uptime**.
3. **Data Lifecycle Management**: You can **manage the data which is stored in the S3 bucket by automating the transition and expiration of objects based on predefined rules**. You can **automatically move the data to the Standard-IA or Glacier, after a specified period**.
4. **Integration with Other AWS Services**: You can **integrate the S3 bucket service with different services** in the AWS like you can integrate with the **AWS lambda function** where the **lambda will be triggered based upon the files or objects added to the S3 bucket**.

## **Some References:-**

### **AWS Playlist:-**

**[https://youtube.com/playlist?list=PL6XT0grm\\_TfgtwtwUit305qS-HhDvb4du&si=G9KdQl2wc2-XPwR9](https://youtube.com/playlist?list=PL6XT0grm_TfgtwtwUit305qS-HhDvb4du&si=G9KdQl2wc2-XPwR9)**