Amazon S3 notes

Amazon Web Services (Chhatrapati Shivaji Maharaj University)

Amazon S3: Amazon Simple Storage Service

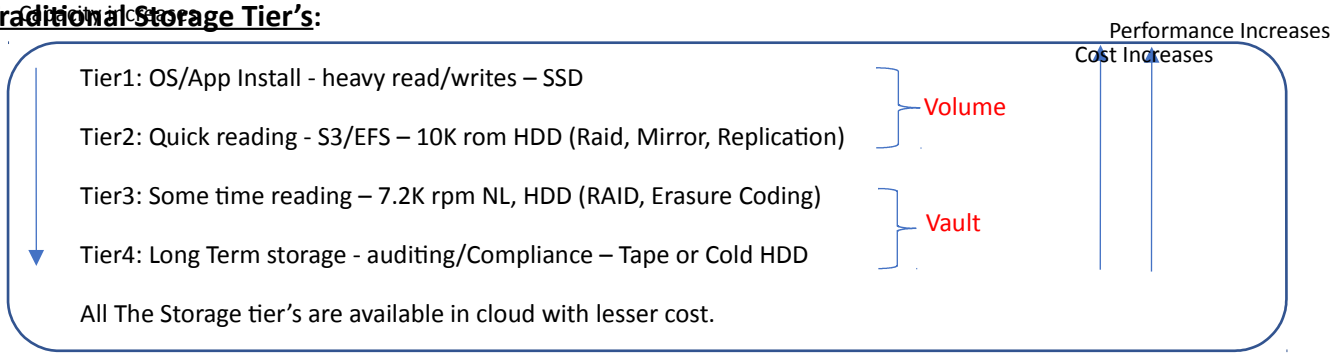
Volume v/s vault Storage:

Volume: It's a storage device, that is formatted to store directories and file for frequent use.

e.g., fixed disk, CD-ROM etc.

Vault: A storage box or a container which store the archive data for longer period of time.

Traditional Storage Tier's:



Disadvantage of Traditional Storage to Cloud:

Sr. No.	Drawbacks of Traditional Storage	Solution provided by Cloud
1.	Storage is sitting idle in the Data Centre. - On Average 40% storage purchased is not used.	Pay for Infrastructure as you need it, and No upfront payment.
2.	Inactive Data is sitting on costly storage. - Up to 99% of is cold	Data reduction technique and archiving to store inactive cold data.
3.	Data silos duplicate management, hardware. - Doc., images & other files are growing rapidly & multiplying data silos	Easy cloud based backup and archiving solution
4.	Backup process slow storage during day - Nearly 50% of organizations need to reduce backup times	Fast service with low cost and low risk.
5.	Data protection strategies are incomplete - Almost 40% of respondents have only one backup method	Fast service with low cost and low risk.
6.	Migrations are frequent, costly and lengthy - Plan for storage migration every 3 years	Easy Migration of data.

Cloud Storage:

Cloud Storage is a service model in which data is maintained, managed, backed up remotely and made available to users over a network.
Users pay for their cloud data storage on their pay per use or monthly rate.

Types of storage in Cloud:

1. Object storage: All types of files can be stored- txt, binary, audio, video, image, .exe, backup etc.
All these files are available over the Internet.
To relate it is like the Google Drive - No OS expansion, No App installation.
e.g., Amazon S3, Glacier
Best Fit for: 1. Picture, videos, highly durable media storage.
2.Cold storage for long-term archive

2. Block storage: All types of files can be stored- txt, binary, audio, video, image, .exe, backup etc.
All these files are not available over the Internet --
Just like the DISK in your VM- will OS expansion, will App installation.
e.g., EBS (Elastic Block Service)
Best Fit for: 1. Access to raw unformatted block level storage.
2. Persistent Storage.

- Block storage can be accessed by only one machine/compute instance at a time whereas Object storage can be accessed directly by multiple machines.

AWS Connecting Storage:
The connecting Storage in AWS are the storage system which connects an on-premises software appliance with cloud-based storage.
Data can be transferred through internet also but it will be slow and costly.
AWS connecting storages:
1.Storage Gateway: Integrates on-premises IT environments with AWS storage (Storage transfer limit: 32 TB)
2. Snowball: A service that enables large volume data transfer. (Storage limit: 80 TB/Snowball)

Requirement: There is a matrimony portal, having a dataset (image/video/audio/profiles/location) of 200 TB. All data in on-prem setup. Now they want to move into AWS- S3.
Solution- Upload over Internet is not feasible option. Use AWS Snowball

Objects:

- Objects are fundamental entities stored in Amazon S3. Generally, objects are files only but here we shall call then Object.
- Each Amazon S3 object has data, a key, and metadata.
- Objects are uniquely identified within a bucket by a Key(name) and a Version ID. An object key is the unique identifier for an object in a bucket.
- Each object can contain up to 5 TB of data.
- AWS Recommendation- any object bigger than 100mb should be uploaded using MULTIPART UPLOAD

Multipart Upload:

Multipart upload allows you to upload a single object as a set of parts. Each part is a contiguous portion of the object's data. You can upload these object parts independently and in any order. If transmission of any part fails, you can retransmit that part without affecting

other parts. After all parts of your object are uploaded, Amazon S3 assembles these parts and creates the object. In general, when your object size reaches 100 MB, you should consider using multipart uploads instead of uploading the object in a single operation.

Bucket:

- Bucket is used to store the objects, which consists of data and metadata that describes the data.
- The bucket can be configured and created in any specific region. Buckets are containers for data stored in S3.
- When an object is added to the bucket, Amazon S3 generates a unique version ID and assigns it to the object.
- By default, only 100 buckets can be created in each AWS account.
- Bucket name must be unique and must not contain spaces or uppercase letters.
- Bucket name must be globally DNS complaint.

Requirement- How to make an object Public?
Solution- Objects can be made public only when the bucket is Public. Make you Bucket public and then make the Object Public.

Que- What can be the maximum size of a bucket?
Ans- no limit
Que- What is the maximum size of an individual object in a bucket?
Ans- max size is 5 TB.
Individual Amazon S3 objects can range in size from a minimum of 0 bytes to a maximum of 5 terabytes.
Que- if i have a object of 5 TB, can I upload it, the way we have been uploading the object till now?
Ans- Not possible
Que- limit on the size of object which can be uploaded in single PUT operation?
Ans- 5 GB

Events:

- Events are notification when objects are created via PUT, POST, Copy, Multipart Upload, or Delete
- Filter on prefixes and suffixes
- Trigger workflow with Amazon SNS, Amazon SQS, and AWS Lambda functions

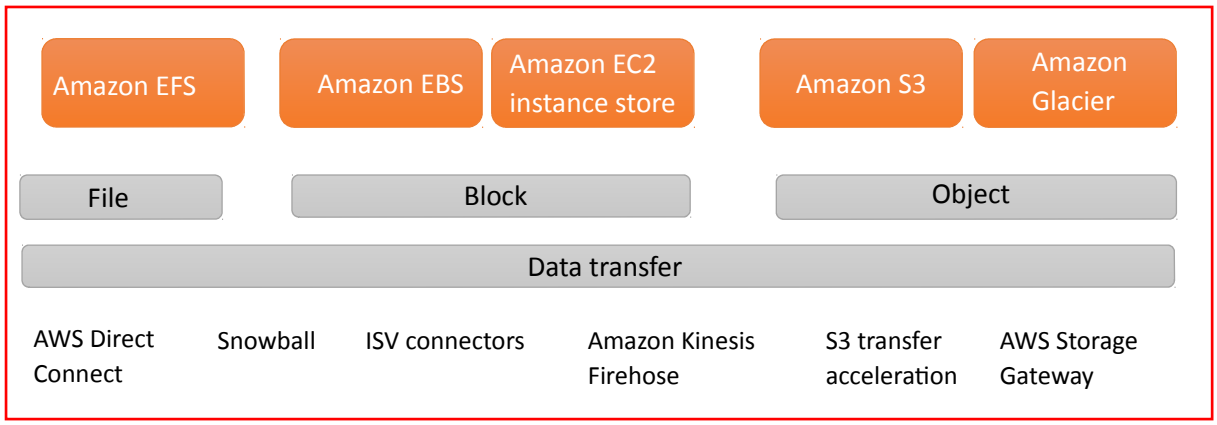
Amazon S3:

- Amazon Simple Storage Service (S3) is a storage designed to make web-scale computing easier for developers.
- Simple interface that helps to store and retrieve any amount of data, at any point of time, from anywhere on the web.
- Amazon S3 is an "object store" in AWS & developed in 2006.
- Amazon S3 is a Global Service with Regional storage. S3 is a global service, however, buckets are created within a region specified during the creation of the bucket.

- S3 is an Object level storage (not a Block level) and cannot be used to host OS or dynamic websites.
- S3 bucket names are globally unique, regardless of the AWS region in which you create the bucket.
- No limit to the number of objects that can be stored in a bucket.
- Buckets cannot be nested and cannot have bucket within another bucket. Can have folders inside a bucket. They are path variables and not the container.

AWS Storage choices:

1. **Amazon S3**: Durable object storage for all types of data.
   Economic, pay as you go
   No upfront investment, No commitment
2. **Amazon Glacier**: Archival storage for infrequently accessed data
   Easy to Use; Self-service administration
   SDKs for simple integration
3. **Amazon EBS**: Block storage for use with Amazon EC2
   Reduce risk; Durable and Secure
   Avoid risks of physical media handling
4. **Amazon EFS**: File storage for use with Amazon EC2
   Agility, Scale; Reduce time to Market
   Focus on your business, not your infrastructure



AWS S3 Storage Classes:

Because S3 One Zone-IA stores data in a single AWS Availability Zone, data stored in this storage class will be lost in the event of Availability Zone destruction.

Que- Can I have a bucket that has different objects in different storage classes? – Yes

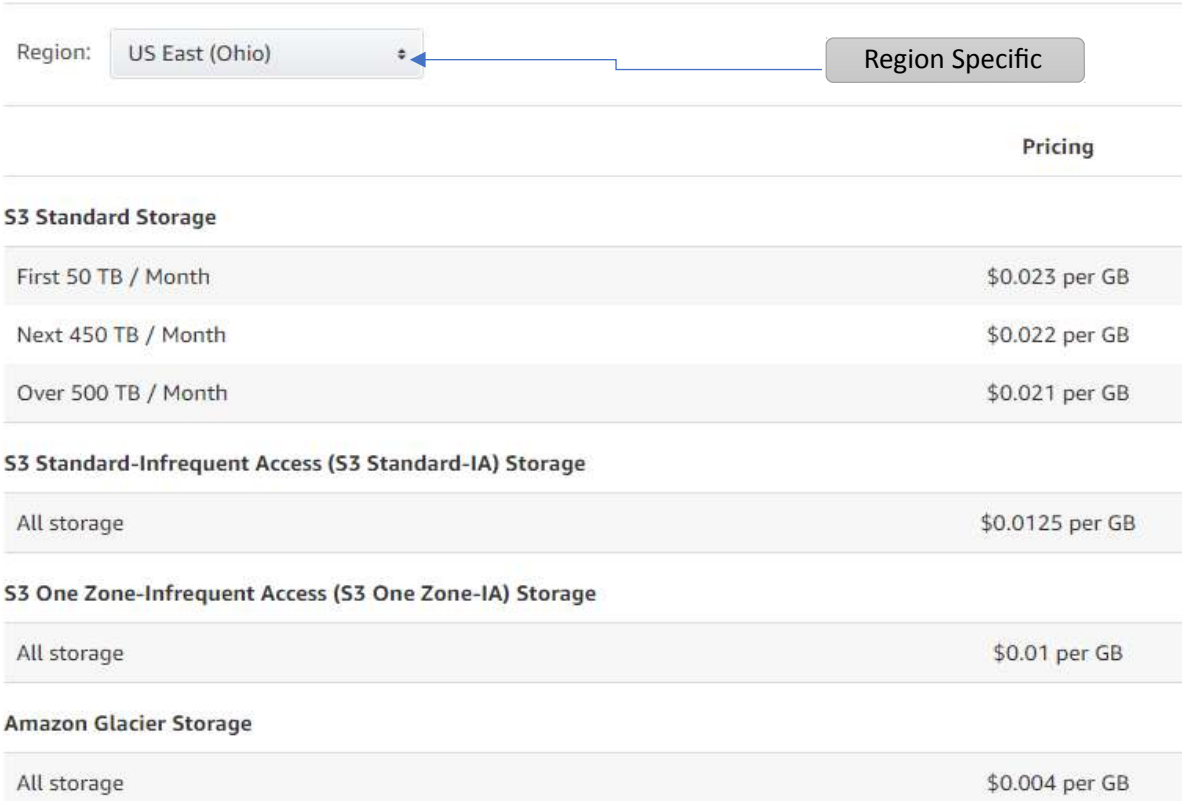
Performance across the S3 Storage Classes:

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days

Que- What are the factors/variables on which cost of S3 will depend?

Ans- 1. Region; 2. amount of data; 3. storage class
4. access (no of reads/writes/modifications etc.) pattern
5. data transfer charges (in same region very less, cross region high charges)

S3 Pricing:



	Pricing
S3 Standard Storage	
First 50 TB / Month	$0.023 per GB
Next 450 TB / Month	$0.022 per GB
Over 500 TB / Month	$0.021 per GB
S3 Standard-Infrequent Access (S3 Standard-IA) Storage	
All storage	$0.0125 per GB
S3 One Zone-Infrequent Access (S3 One Zone-IA) Storage	
All storage	$0.01 per GB
Amazon Glacier Storage	
All storage	$0.004 per GB

S3 Intelligent-Tiering:

An S3 storage class for data with unknown access patterns or changing access patterns that are difficult to learn. It is the first cloud storage class that delivers automatic cost savings by moving objects between two access tiers when access patterns change. One tier is optimized for frequent access and the other lower-cost tier is designed for infrequent access.

S3 Intelligent-Tiering works by monitoring access patterns and then moving the objects that have not been accessed in 30 consecutive days to the infrequent access tier.

- Objects of size 5GB can be uploaded in a single PUT operation
- Multipart upload – can be used for objects of size > 5GB and supports max size of 5TB can is recommended for objects above size 100MB
- Amazon S3 costs vary by region
- Charges in S3 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

S3: Standard-Infrequent Access Storage

Integrated: Lifecycle Management

- Transition Standard to Standard-IA
- Transition Standard-IA to Amazon Glacier storage
- Expiration lifecycle policy
- Versioning support
- Directly PUT to Standard-IA

Standard – infrequent Access

AWS S3 Data Consistency Model:

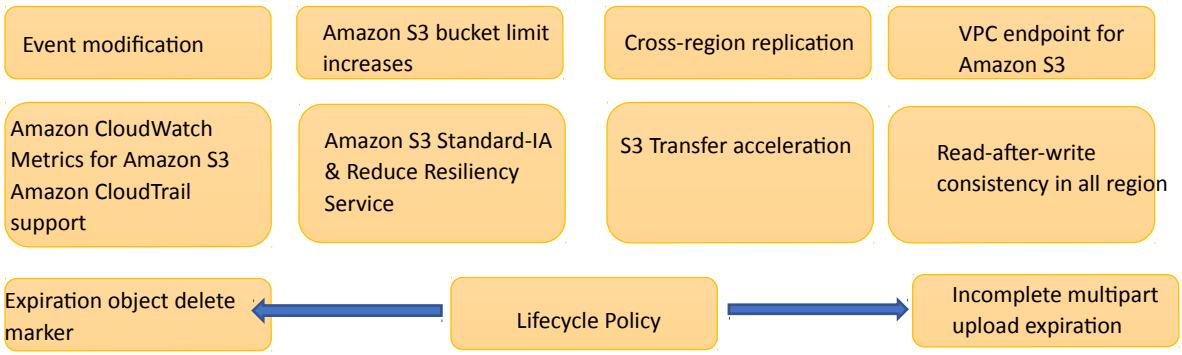
- S3 achieves high availability by replicating data across multiple servers within Amazon's data centers.
- S3 provides read-after-write consistency for PUTS of new objects.
- S3 provides eventual consistency for overwrite PUTS and DELETES.
- Updates to a single key are atomic. *for e.g., if you PUT to an existing key, a subsequent read might return the old data or the updated data, but it will never write corrupted or partial data.*

Amazon Commitment:

Highly durable object storage for all types of data

- Internet-scale storage: Grow without limits
- Built-in redundancy: Designed for 99.999999999% durability
- Low price per GB per month: No commitment, No up-front cost
- Benefit from AWS's massive security investment

S3 Features:



Versioning:

- Versioning is means of keeping multiple variants of an object in the same bucket.
- Every version of every object stored in Amazon S3 bucket is Reserved, Retrieved, Restored.
- It automatically adds new version with every upload. And easily control No. of Versions.
- Versioning-enabled buckets enable you to recover objects from accidental deletion or overwrite.
- Three States of an Amazon S3 bucket: (1). Default-Un versioned
  (2). Versioning-enabled (3). Versioning-suspended

Que.- Can my computer perform versioning? No

Requirement- I am into a Credit Card division of HDFC bank. CC statements get generated 4 times/cycles in a month (4th, 13th,21st,29th). Let the Statement be in "Frequently accessed" class for 1st 3 months. After that, move statements into "in Frequently accessed" class. After 1 year, move it into archive. After 5 years, then delete it.
Solution- Lifecycle Management Rule

Lifecycle Policies: Lifecycle Management Rule

Use lifecycle rules to define actions you want Amazon S3 to take during an object's lifetime such as transitioning objects to another storage class, archiving them, or deleting them after a specified period of time.

- Automatic tiering and cost controls
- Includes two possible actions:
  - Transition: Archives to Standard-IA or Amazon Glacier after specified time
  - Expiration: Deletes objects after specified time
- Allows for actions to be combined
- Set policies at the prefix level

Que- Can a single bucket has multiple Lifecycle rules? Ans- Yes you can

Versioning +Lifecycle Policies

Requirement- I have a source bucket (in Mumbai region). If I upload a new file in my source bucket, then it should be uploaded in my target bucket (in Singapore region).
Solution- Object Replication

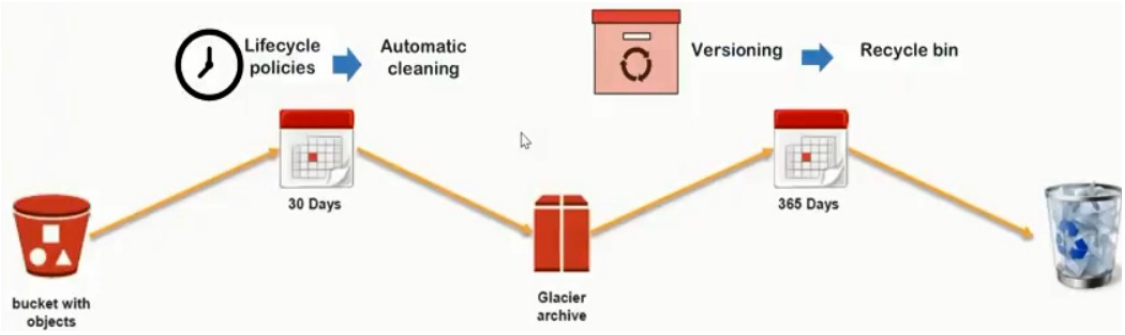
Object Replication:

1. CRR (cross region replication) -- source and target buckets will be in diff regions
2. SRR (Same region replication) -- source and target buckets will be in same region

** both the source and target buckets must be versioned

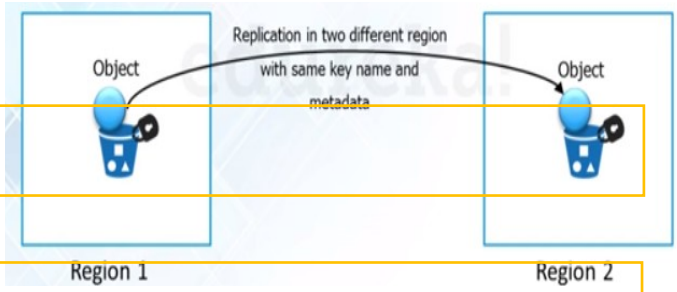
** can be done in Your own AWS's account buckets or diff AWS's account bucket.

** we need an IAM Role



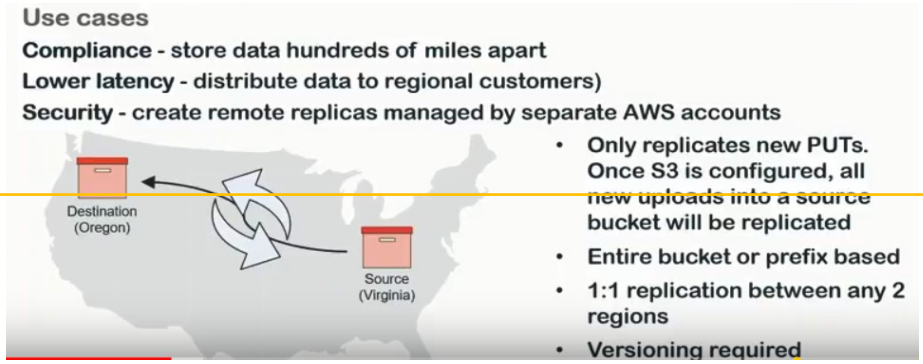
Cross-Region Replication:

- It is a bucket level feature that enables automatic copying of objects across buckets in different AWS regions.
- The Object replicas in destination bucket = replicas of the objects in the source bucket

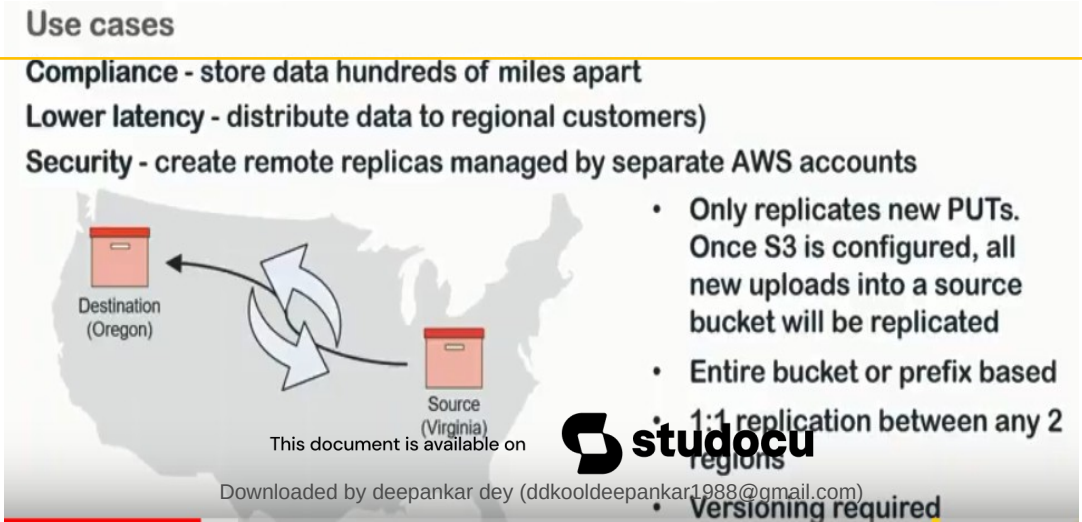


Durability: related to file loss (11 9s)
Availability: when you demanded the file, at that time how quickly you got access

Que- How to share Bucket with other AWS account & also with other AWS account IAM USER?
Ans- ACL (Access Control List) (can only grant basic read/write permission)
Two types of permission strategy



1. Fine grained access control- minute level of access can be controlled- list the files but not read the contents of the file
2. coarse grained access control- basic read/write/execute – ACL



Que- How to share Bucket with other AWS account & also with other AWS account IAM USER. But grant only LIST permission.
Ans- ACL can't work here. BUCKET POLICY will work here.

Bucket Policy and ACL:

- S3 bucket policies, are attached only to S3 buckets. S3 bucket policies specify what actions are allowed or denied for which principals on the bucket that the bucket policy is attached to (e.g., allow user Alice to PUT but not DELETE objects in the bucket).
- You attach S3 bucket policies at the bucket level (i.e., you can't attach a bucket policy to an S3 object), but the permissions specified in the bucket policy apply to all the objects in the bucket.
- Bucket Policy is very handy in giving bucket access to a different AWS account users without creating roles.
- The bucket policy, written in JSON, provides access to the objects stored in the bucket.
- IAM Policy will be attached to USER/Groups/Role.
- BUCKET POLICY will be created on an individual bucket.

Bucket policies are recommended over ACLs.

- ACLs is a legacy access control mechanism that predates IAM and Bucket Policy.
- With help of ACLs, we can only give simple permissions.
- Complex permissions can only be given via Bucket Policy (e.g., allow user Alice to PUT but not DELETE objects in the bucket).
- In ACL, you can't DENY any permission, it can only ALLOW READ/WRITE.
- ACL is coarse-grained access control but Bucket Policy is fine-grained access control.

Cross-Origin Resource Sharing (CORS):

- Defines a way for client web applications that are loaded in one domain to interact with resources in a different domain.
- With CORS support, you can build rich client-side web applications with Amazon S3 and selectively allow cross-origin access to your Amazon S3 resources.

Use-case Scenarios: This is practically beyond the domain of this training. It involves knowledge of JavaScript to perform proper scripting.

Encryption Options:

1. Client-side encryption use AWS SDKs:
   You manage the encryption keys and never send them to AWS
2. Server-side encryption (SSE):
   i. SSE with Amazon S3 managed keys:
      - "Check-the-box" to encrypt your data at rest. Keys managed by S3
      - Keys managed centrally in AWS KMS with permissions and auding of usage Encryption Options using KMS

   ii. SSE with customer provided keys:
      - You manage your encryption keys and provide them for PUTs and GETS

HTTP Methods	Original activity
POST	Create
GET	Read
PUT	Update/Replace
PATCH	Update/Modify
DELETE	Delete