

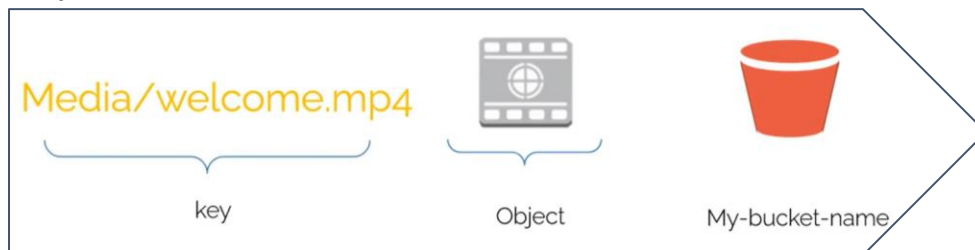
CLOUD CONCEPTS S3

(Simple Storage Service)

- **Amazon S3** is a virtual storage service provided by Amazon, which can be accessed from anywhere at any time through internet access. Common examples of such types of storage are Dropbox, OneDrive and google drive (provide 15GB of free space for storage). Just as in google drive only software is stored or saved but it cannot be installed. Likewise, S3 is also object level storage.
- **Max Storage of a bucket is 5TB.**
- **Types of Storage:**
 - Block Level Storage: Can be used as a storage device and allows the installation of software. E.g., Hard disk drive and in cloud context the default hard drive (30GB) attached to EC2 (windows server) is block level storage.
 - Object Level Storage: Do not provide operation level access (installation) only storage is provided.

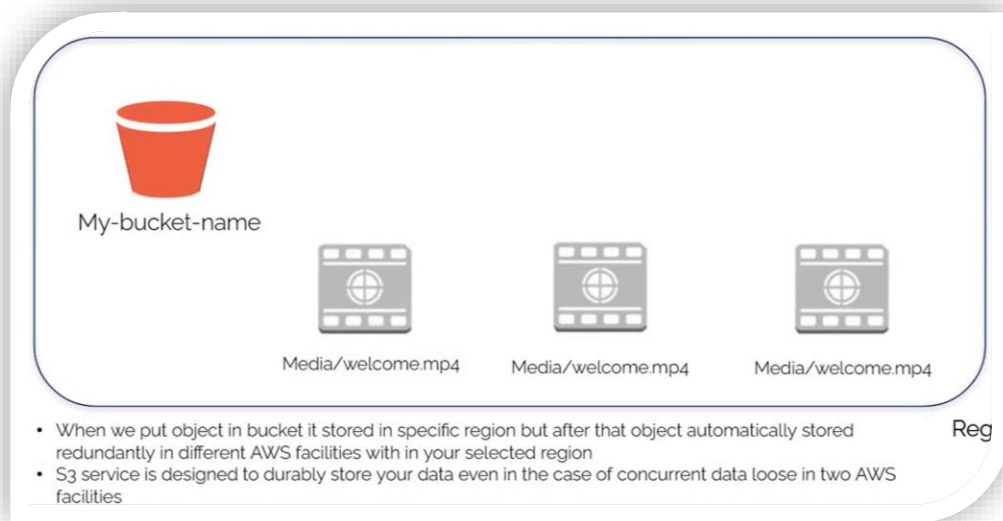


- Objects are stored in buckets. The bucket in cloud is same as folder and object could be any file type say media file, image, or any type of document. Buckets must not have the same names. Multiple objects can be stored in a bucket.



✓ Key has the folder, file name and the extension which is an object to be stored in a bucket.

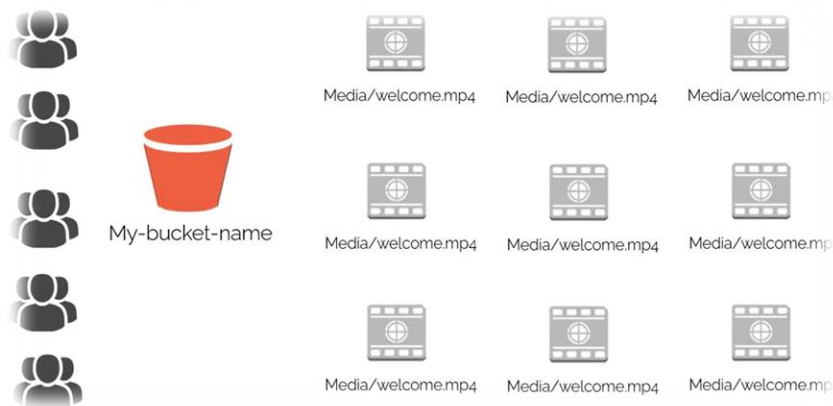
- Say you are in the North Virginia Region with 6 availability zones (physical data centers) and you want to store an object. Whenever you store an object, it is spread amongst multiple AZ. Now, if you want to delete any stored object, it won't be deleted immediately because copies of that object are in various AZs. First all the copies are deleted, and the object is deleted. Same is the case with modification or updating an object. This is **an eventual consistency model** in which any operation performed on an object will take time as all the changes in an object will be replicated amongst all the copies.
 - In context of Data Redundancy Model there are two types of models:
 - Eventual consistency model
 - Read after write model. (Access the data as it is created e.g., renaming of file in a bucket)



- The data is only shared in all the AZs of that region, but it will not be shared outside the region. This can be done and known as CORS (Cross origin resource sharing).

- How cost can be minimized?
- How data can be made secure?

- S3 is designed for seamless scaling it will automatically handle high volume of requests. S3 can manage the data storage behind your bucket if data size exceeds 5TB and allows you to put more data.



- Data can be accessed anywhere through Dashboard, Command Line Interface, or AWS SDK.



- **S3 Storage Classes:** There are 6 types of storage classes.

- **Standard Storage:** This is the default storage in S3. The data kept in this storage can be accessed as soon as it is created. Since it makes data available immediately it costs a lot. Hence it is recommended to use this storage for testing purposes not as permanent storage.

Designed for general, all-purpose storage
The default storage option
99.999999999% object durability ("11 nines")
99.99% object availability
The most expensive storage class

- **Standard Infrequent Access:** If data is to be accessed frequently this type of class is recommended. Say if you want to access the data once a week or once a month put the data in IA bucket. This way the cost for storage is minimized and you will be charged when you access the data.

Designed for objects that you do not access frequently but must be immediately available when accessed (uses multiple Availability Zones)
99.999999999% object durability
99.90% object availability
Less expensive than the standard storage class

- **One zone Infrequent Access:** It is just like *Standard Infrequent Access*, but the difference is when data is stored in *Standard Storages*, they spread the data in all the zones in a region while in *one zone IA*, data is kept in one Availability Zone only. Problem with one zone IA is that in case of accidental deletion all the data is lost. Since there is no back up it is less recommended and has a very low cost.

Designed for objects that you do not access frequently but must be immediately available when accessed (only uses one Availability Zone)
99.99% object durability
99.00% object availability
20% less expensive than the Standard-IA storage class

- **Intelligent Tiering:** It is an intelligent class which automatically moves the data to low-cost solutions. E.g., if stored data is not accessed for a long period of time, it will automatically move that data to the low-cost storage class. Main limitation for this feature is that file size must be equal to or greater than 120 KB.

Designed to optimize costs by automatically moving data to the most cost-effective tier based on your usage
99.999999999% object durability ("11 nines")
99.90% object availability
Pricing depends on the assigned storage class

- **Glacier:** If the data is to be stored for long term and the access time is very low this class is recommended. In this class stored data cannot be accessed before 90 days (min). In case of an emergency access, a request must be generated before 3 – 5 hours. A new modification in this class allows expedited retrieval access which is charged.
- **Glacier Deep Archive:** It is same as Glacier class, but the access time is 180 days or 6 months (min). Data in this storage can also be accessed through expedited retrieval or access request.

Designed for long-term archival storage

May take several hours for objects stored in Glacier to be retrieved
99.999999999% object durability
The cheapest S3 storage class (very low cost)
Glacier Deep Archive has a longer default minimum storage duration and a longer default retrieval time than Glacier

- ☛ If any class is not selected at the time of storage creation the default storage i.e., standard storage is created which can be changed later.

What Is an S3 Storage Class?

A storage class represents the "classification" assigned to each object in S3.

Available storage classes include:

Standard
Standard-IA (Infrequent Access)
One Zone-IA (Infrequent Access)
Intelligent-Tiering
Glacier
Glacier Deep Archive

Each storage class has varying attributes that dictate things like:

Storage cost
Object availability
Object durability
Frequency of access (to the object)
Each object must be assigned a storage class ("standard" is the default class).

You can change the storage class of an object at any time (for the most part).

Object Durability and Availability

Object Durability:

1. The percent (%) over a one-year time period that a file stored in S3 will not be lost.
For object durability of 99.999999999% (11 nines), that means there is a 0.00000001% chance of a file in S3 being lost in a year.

OR

If you have 10,000 files stored in S3 (at 11 nines durability), then you can expect to lose one file in 10 million years.

Object Availability:

1. The percent (%) over a one-year time period that a file stored in S3 will be accessible

For object availability of 99.99%, that means there is a 0.01% chance you won't be able to access a file stored in S3 in a year.

OR

For every 10,000 hours, you can expect a total of one hour for which a file may not be available to access.

Other S3 Features and Benefits

Features

Object Sharing:

- The ability to make any object publicly available via a URL link

Object Lifecycles:

- Set rules to automatically transfer objects between storage classes at defined time intervals

Object Versioning:

- Automatically keep multiple versions of an object (when enabled).

Additional Benefits

1. Durable, reliable, scalable
 2. Security (offers three different kinds of encryption)
 3. Integrates with almost all other AWS services
 4. Can run big data analytics on objects directly in S3
- Easy to get data in and out of S3
Robust admin and access management options available

- ☛ Versioning is enabled for backup not enabled by default. It is recommended for Disaster Recovery.
- ☛ Other Applications:



Storing Application Assets



Static Web Hosting



Backup & Disaster Recovery



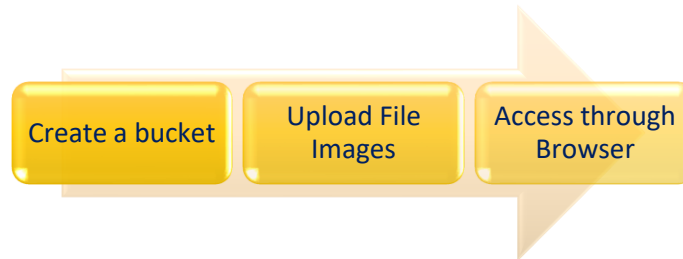
Staging area for Big Data



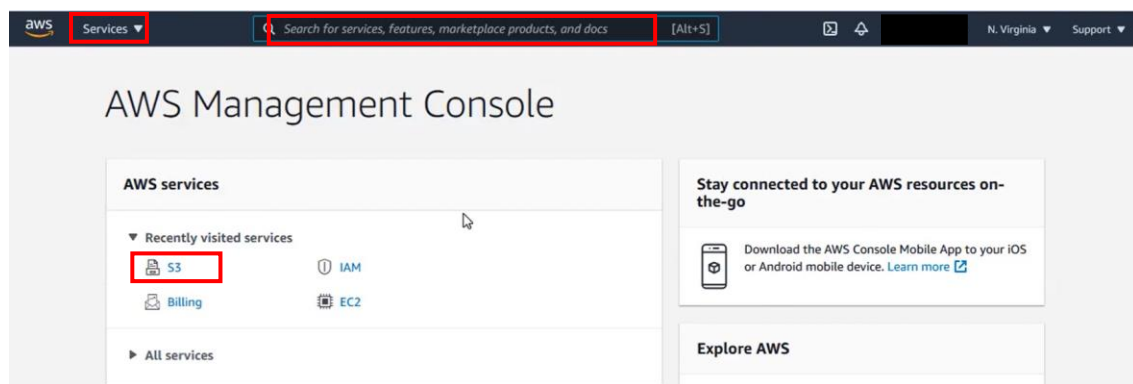
Many more...

DEMO

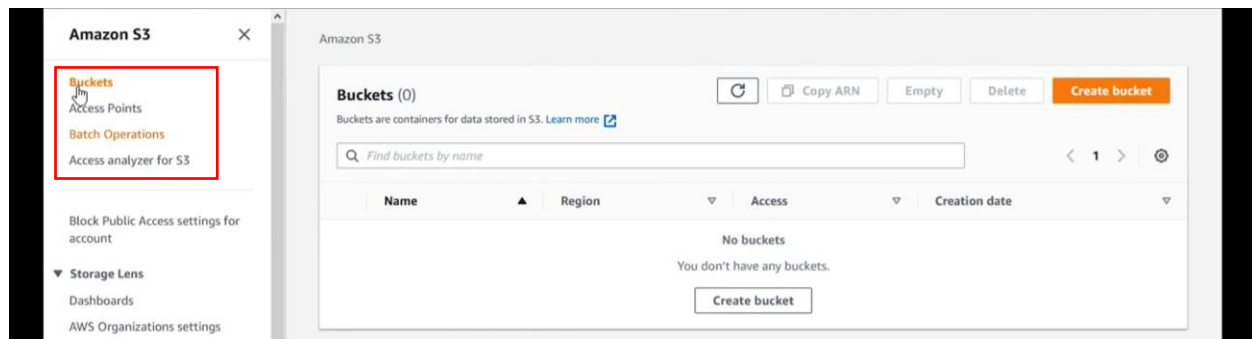
(Lab creating a bucket with object and access it through browser.)



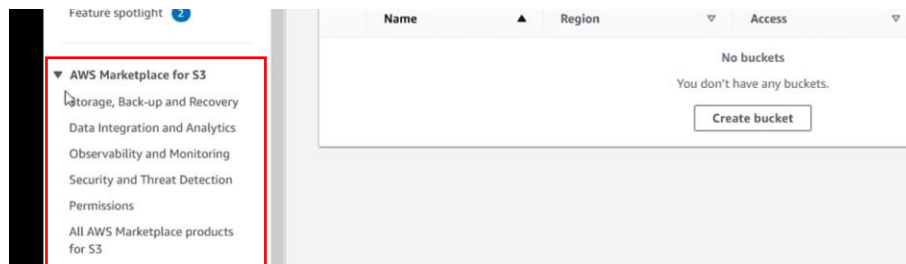
- Search for S3 or go through Services.



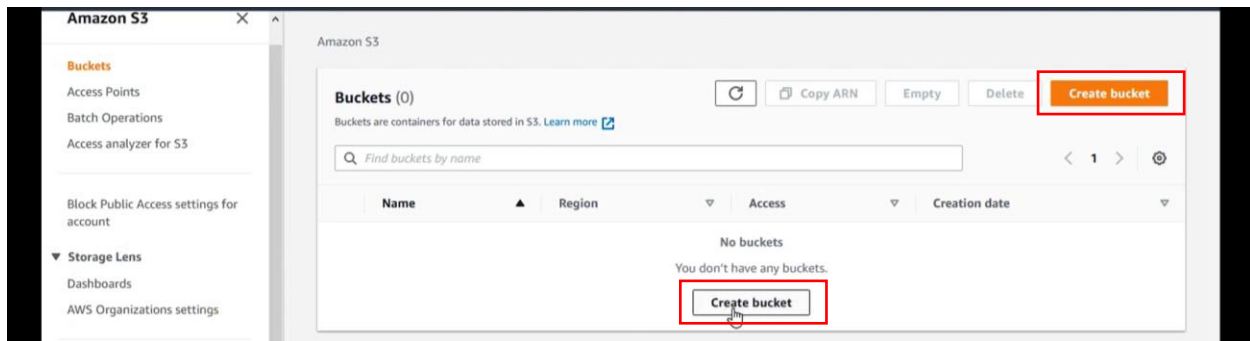
- Dashboard for storage S3



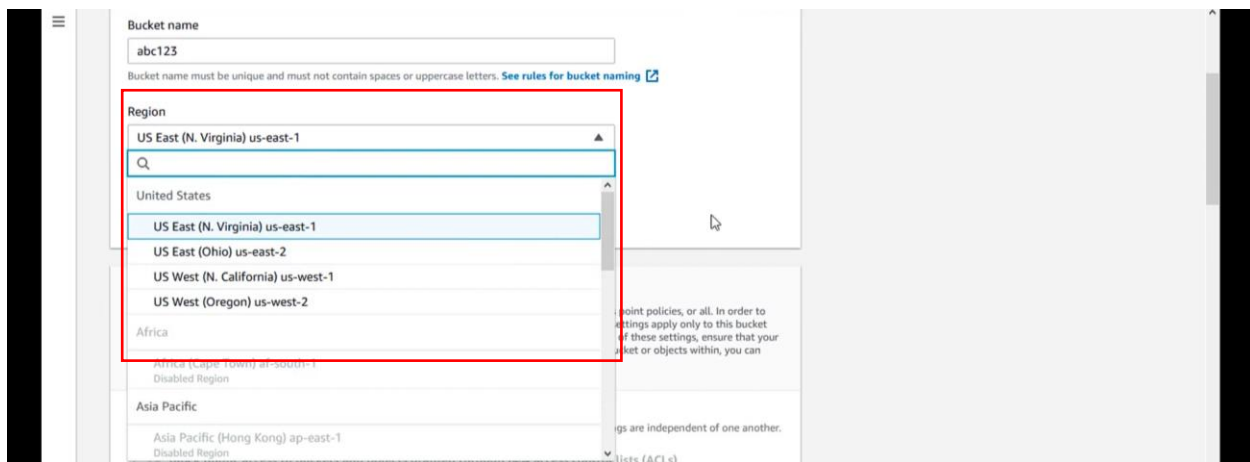
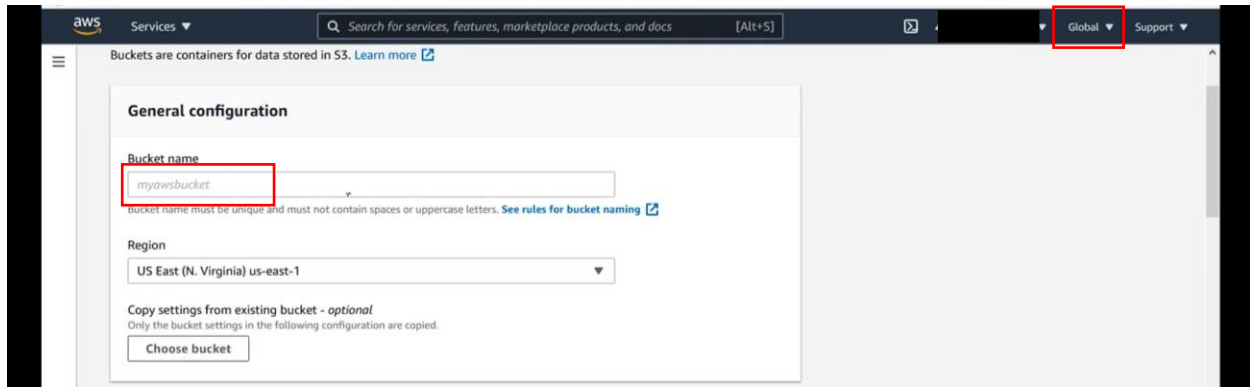
- AWS Market place



- Create a bucket.



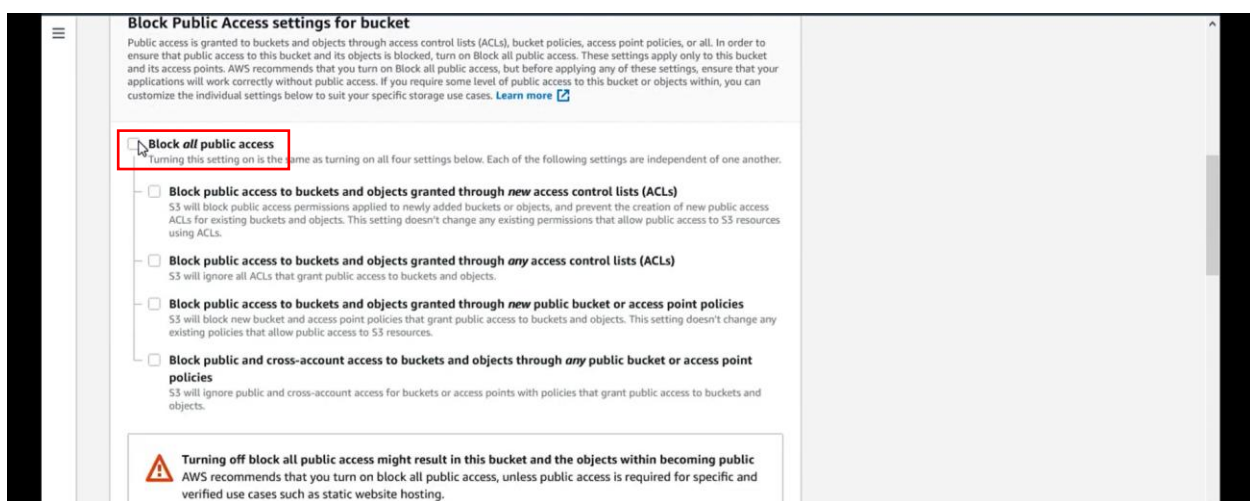
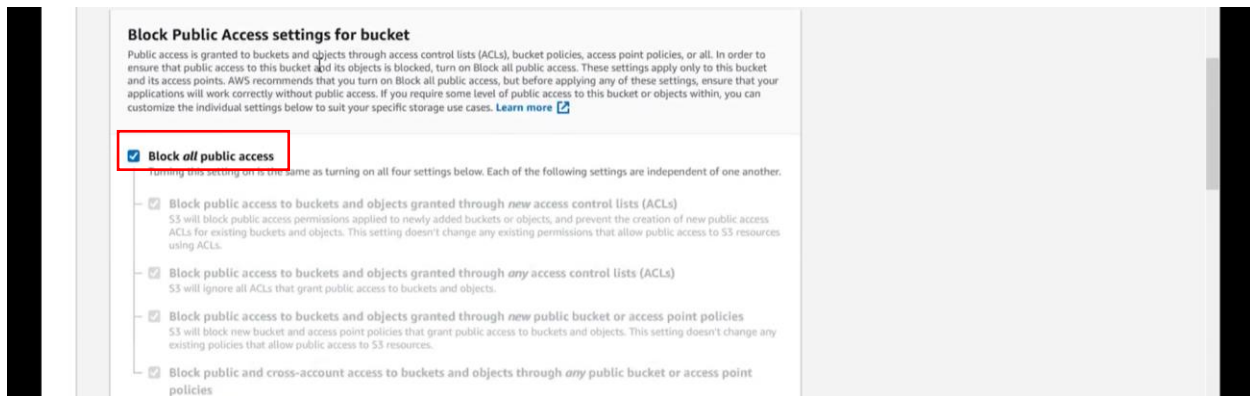
- Enter the bucket name. Bucket name must be unique and can be as long as 63 characters (digit only, alphabets or alphanumeric can be used). There must not be a capital letter. As each object is accessed through a specific URL.
- It is a global service that can be accessed from any region. *Region selection option is optional.*



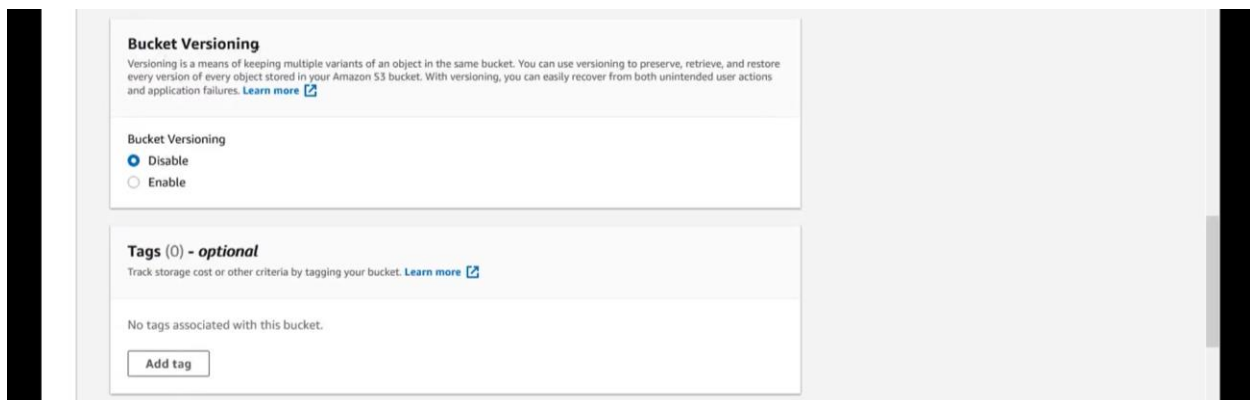
- Existing buckets can be used (*optional*).



- Public access option can be checked or unchecked.



- Bucket Versioning is disabled by default and works as a backup. It is used to create different versions of a file. In case of accidental deletion, different versions of a file will be available.
- Tagging (naming) can be done.



- Default encryption is disabled. Can be enabled.

Default encryption
Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

☒ Disable
☐ Enable

[Advanced settings](#)

- In advanced settings objects can be locked which disables the access of the object by any other. After that click on create bucket.

Advanced settings

Object Lock
Store objects using a write-once-read-many (WORM) model to help you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely. [Learn more](#)

☒ Disable
☐ Enable

Permanently allows objects in this bucket to be locked. Additional Object Lock configuration is required in bucket details after bucket creation to protect objects in this bucket from being deleted or overwritten.

Object Lock works only in versioned buckets. Enabling Object Lock automatically enables Bucket Versioning.

After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel **Create bucket**

- **Name error:** Name already exists it means that there exists a bucket around somewhere with same name though we do not have any bucket in our bucket list.

Create bucket
Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name
abc123

Bucket with the same name already exists
Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

Region
US East (N. Virginia) us-east-1

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.
[Choose bucket](#)

Amazon S3

Buckets
Access Points
Batch Operations
Access analyzer for S3

Block Public Access settings for account

Storage Lens
Dashboards
AWS Organizations settings

Buckets (0)
Buckets are containers for data stored in S3. [Learn more](#)

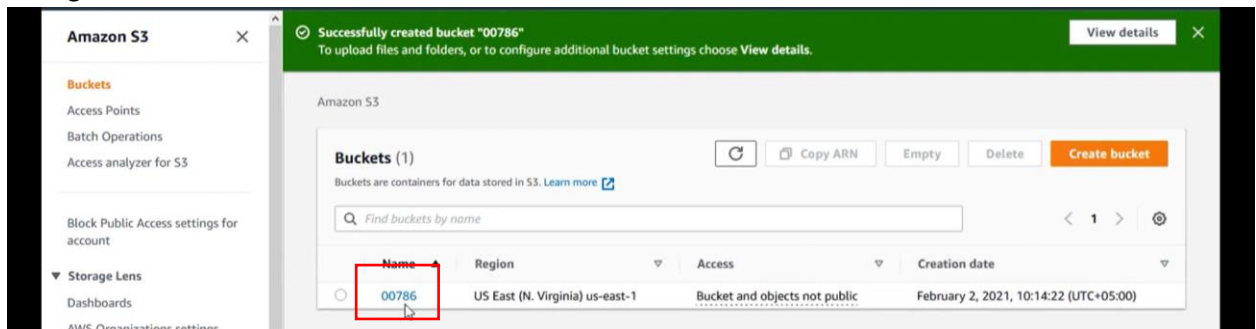
Find buckets by name

Name	Region	Access	Creation date
No buckets			

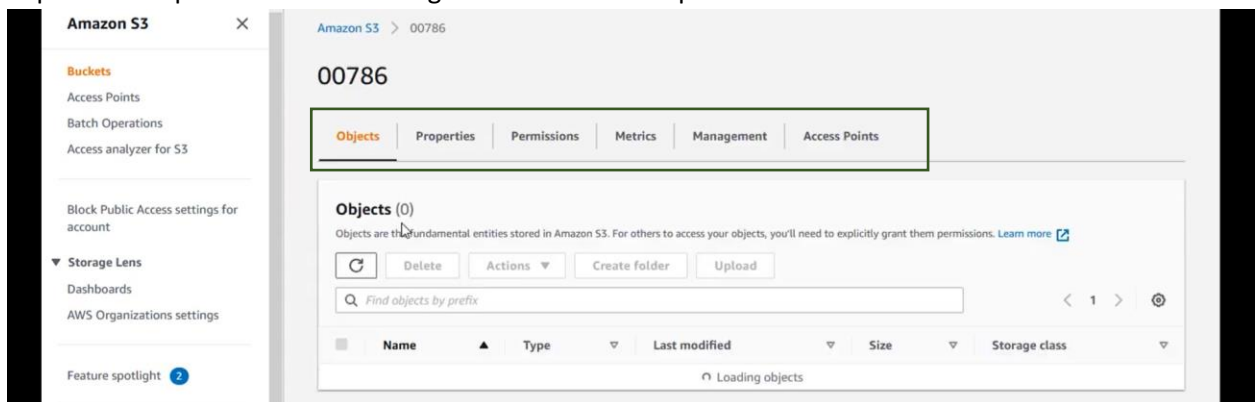
You don't have any buckets.

[Create bucket](#)

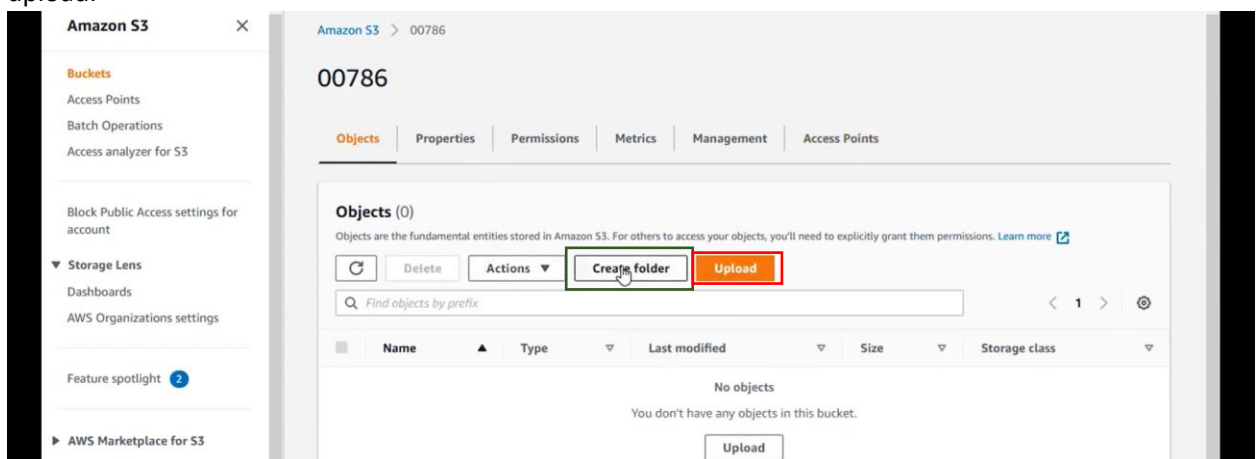
- Change the name and create the new bucket.

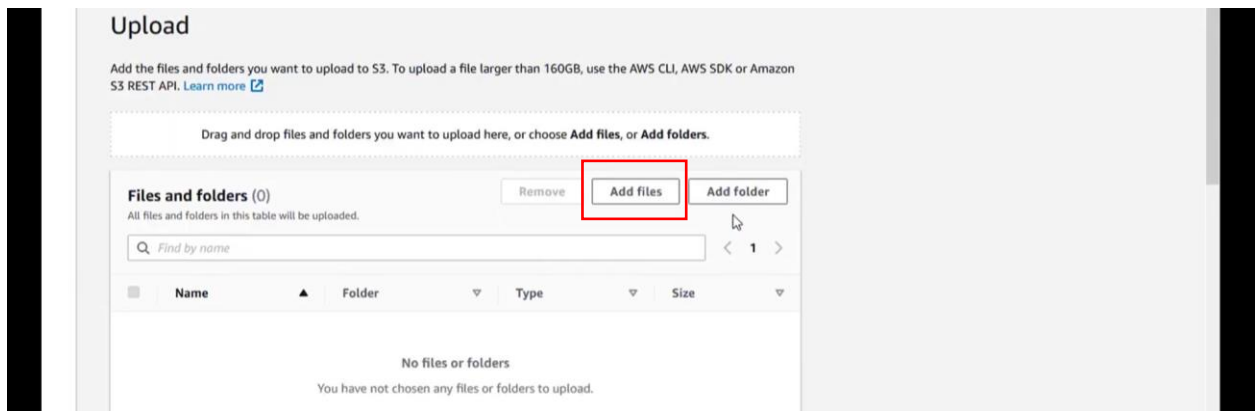


- To store the object, click on the bucket name.
- See the properties tab to explore different configurations like tagging, intelligent tiering, static web hosting, encryption, bucket versioning, object locking etc.
- In permission public access is managed and bucket level permissions.

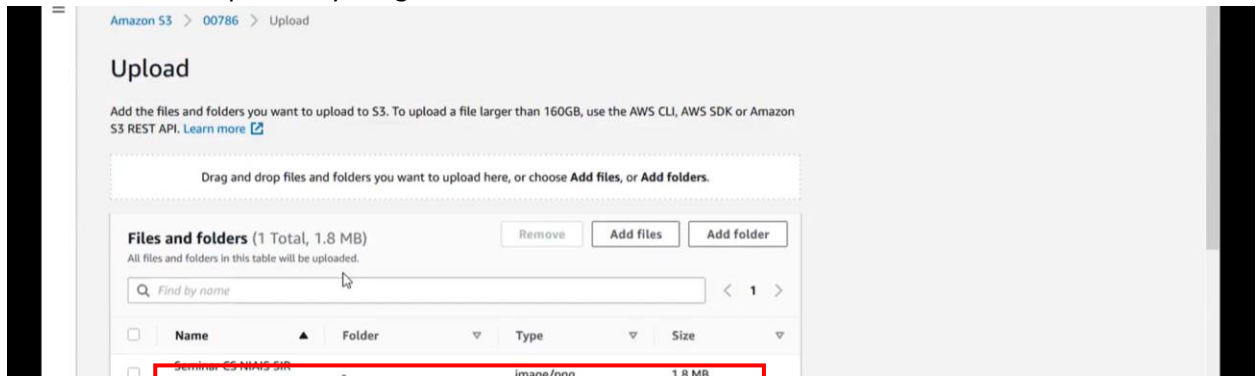


- Now click on object. You can add object directly or can place an object in a folder. Click on upload.





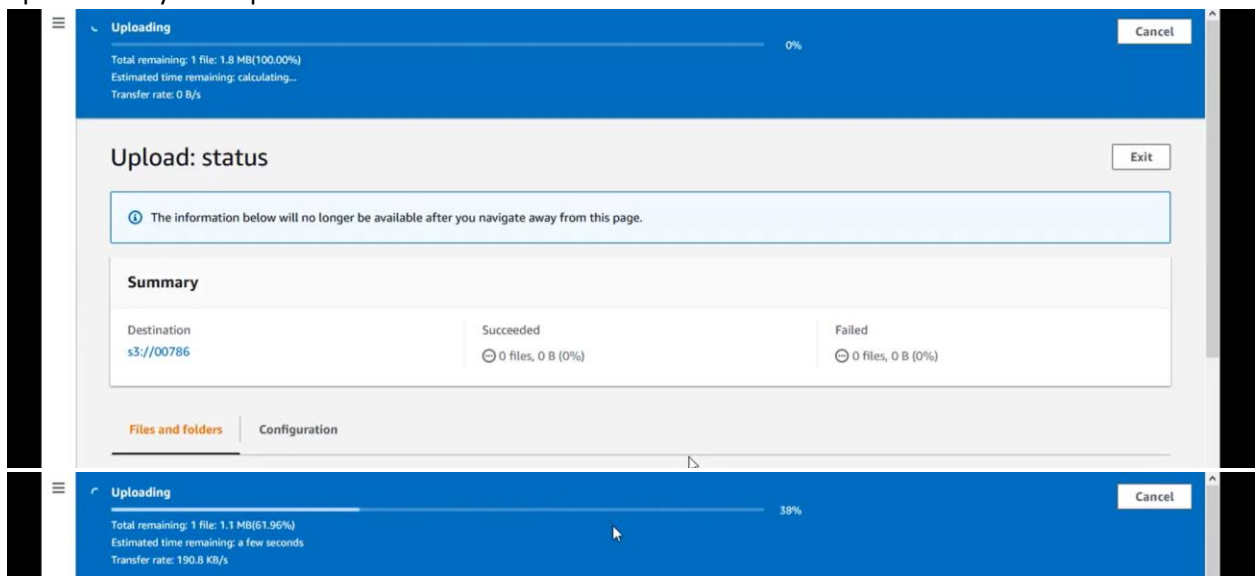
- Click on files and upload any image file and file is added.



- Destination is the bucket name.

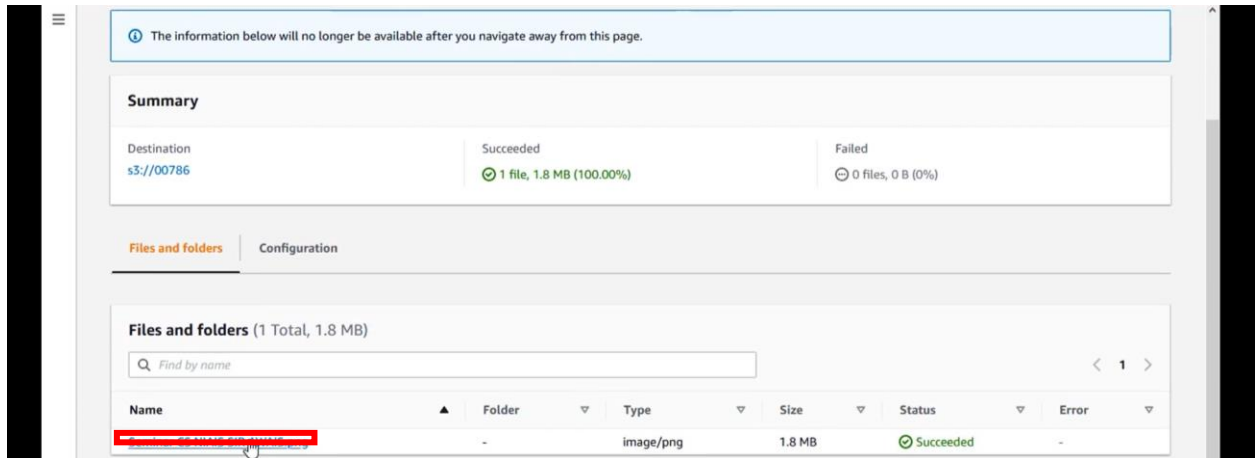


- Click on upload. File will be uploaded. Disturbing internet connection will result in unsuccessful upload or any other problem.

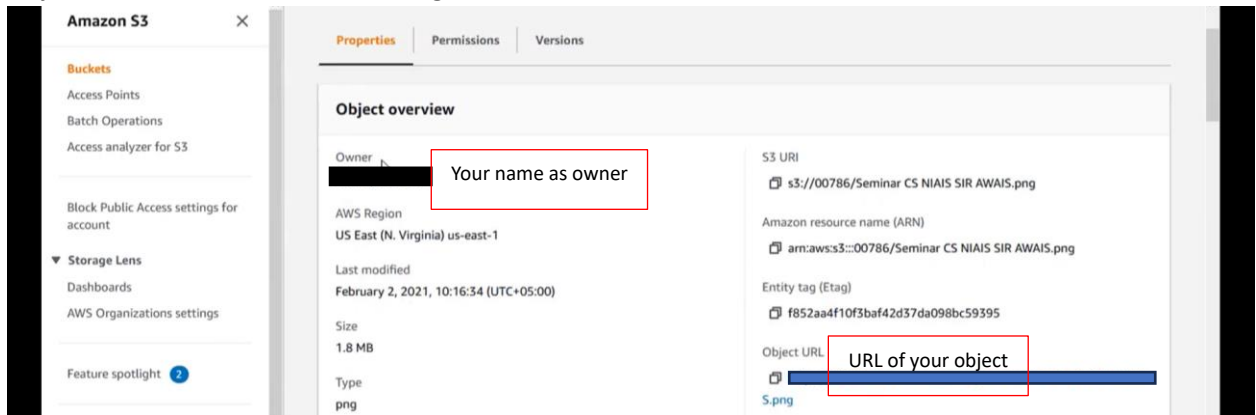




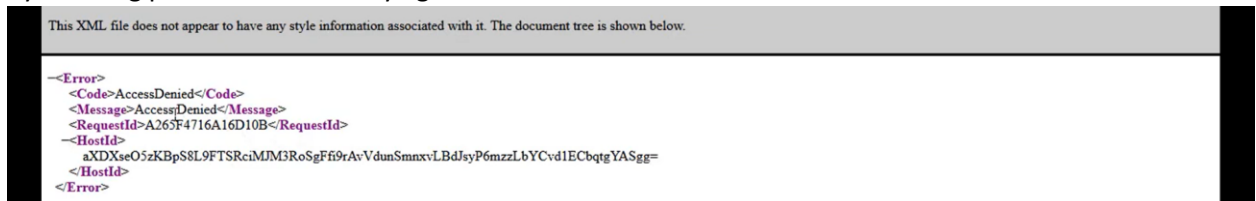
- Click on file name.



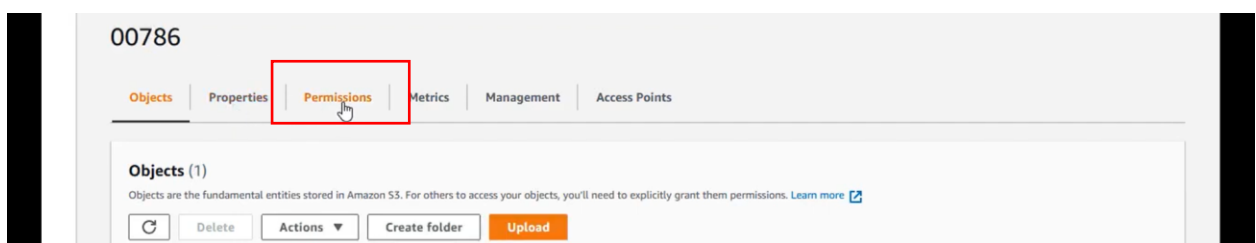
- Object is created with default settings.

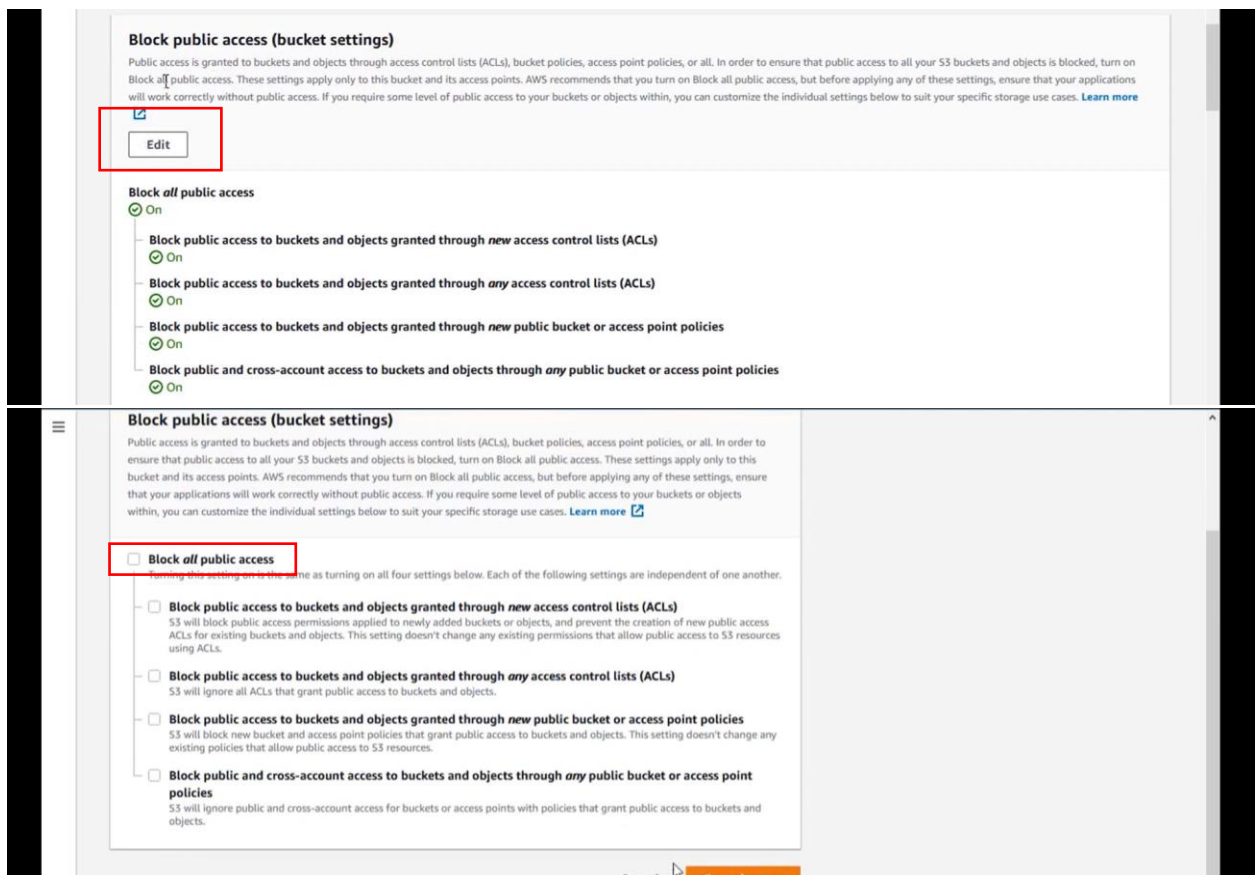


- Click on URL and open it in browser.
- Access denied error is due to settings that public access of bucket is blocked. Change the settings by allowing public access and try again.



Go-to permissions click on edit uncheck the option and click save changes.





- And type confirm.



- Next step is to make object publicly accessible. Click on object then click on Permissions and click on edit.



- Check the options and save the changes and refresh the browser page to view the file.

Access control list (ACL)

Grant basic read/write permissions to AWS accounts. [Learn more](#)

Grantee	Objects	Object ACL
Object owner Canonical ID: d96fd2683f	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write
1d5057aa65b9fde86ed470b29e93 Everyone (public access)	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Read <input type="checkbox"/> Write
Group: http://acs.amazonaws.com/groups/global/AllUsers		
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	<input type="checkbox"/> Read	<input type="checkbox"/> Read <input type="checkbox"/> Write

When you grant access to the Everyone or Authenticated users group grantees, anyone in the world can access this object.
[Learn more](#)
☒ I understand the effects of these changes on this object.

Access for other AWS accounts

No other AWS accounts associated with the resource.

[Add grantee](#)