

DAY 20

DATE:20/05/2025

NAME: ANNIE JOHN

USER ID:27739

Batch: 25VID0885_DC_Batch4

TITLE: DEPLOYING STATIC WEBSITE IN AWS S3 WITH CLOUD BASED SCALABILITY

➤ Objective

The primary objective of this project is to successfully deploy a static website using **Amazon Web Services (AWS) S3**, demonstrating a scalable and cost-effective method for hosting web content. This project aims to configure S3 buckets for static website hosting, ensuring proper access policies and permissions are applied for public access. Additionally, it seeks to enhance website performance and availability by integrating **Amazon CloudFront**, which provides global content delivery and caching to reduce latency.

➤ Step By Step Process

1. Open Amazon S3 console->Click on create bucket->choose bucket type as General purpose->Give a bucket name.

The screenshot displays the 'Create bucket' interface in the AWS S3 console. The 'General configuration' section is active, showing the 'AWS Region' as 'Asia Pacific (Mumbai) ap-south-1'. Under 'Bucket type', 'General purpose' is selected, with a note recommending it for most use cases. The 'Bucket name' field is populated with 'annie.com'. Below the name field, there's a section for 'Copy settings from existing bucket - optional' with a 'Choose bucket' button. The 'Object Ownership' section shows 'ACLs disabled (recommended)' as the selected option, with a note stating that objects will be owned by the account and access is specified using only policies. A yellow warning box at the bottom advises disabling ACLs to simplify permissions management and auditing, suggesting the use of bucket policies instead.

2. Choose object ownership as bucket owner preferred->uncheck the radio button of block all public access->accept the Acknowledgement and create bucket.

ap-south-1.console.aws.amazon.com/s3/bucket/create?region=ap-south-1&bucketType=general

Amazon S3 > Buckets > Create bucket

Object Ownership

☒ **Bucket owner preferred**
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

☐ **Object writer**
The object writer remains the object owner.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☒ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Turning off block all public access might result in this bucket and the objects within becoming public.
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from

3. Open the bucket->Upload html document(eg:index.html) for static web hosting.

ap-south-1.console.aws.amazon.com/s3/upload/annie.com?region=ap-south-1&bucketType=general

Amazon S3 > Buckets > annie.com > Upload

Upload info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 total, 1.5 KB)

All files and folders in this table will be uploaded.

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	index.html	-	text/html	1.5 KB

Destination info

Destination
[s3://annie.com](#)

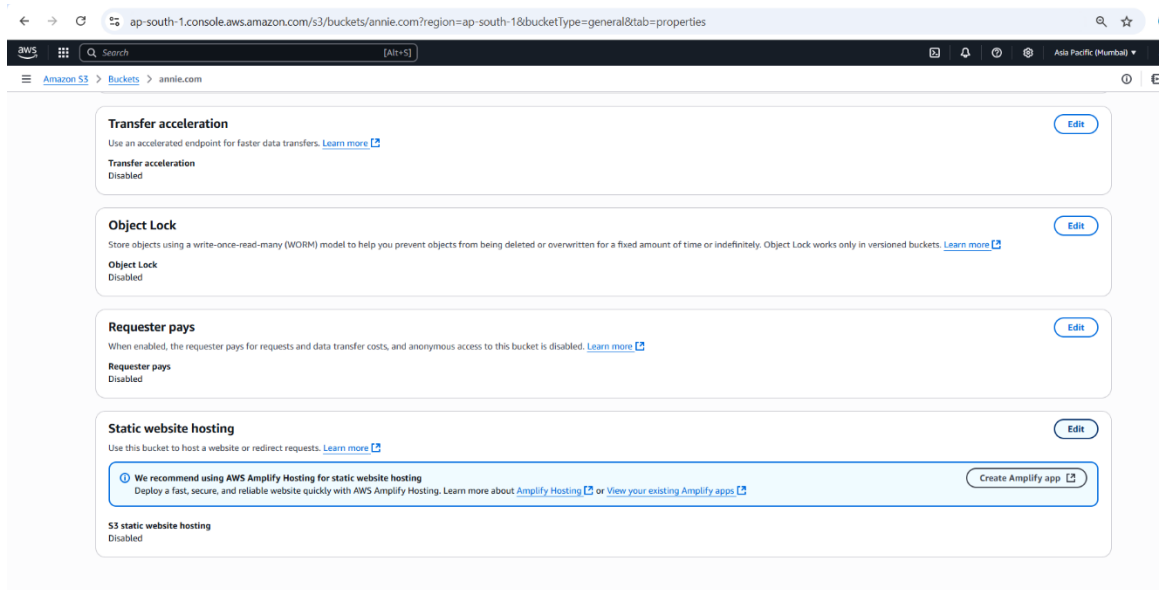
Destination details
Bucket settings that impact new objects stored in the specified destination.

Permissions
Grant public access and access to other AWS accounts.

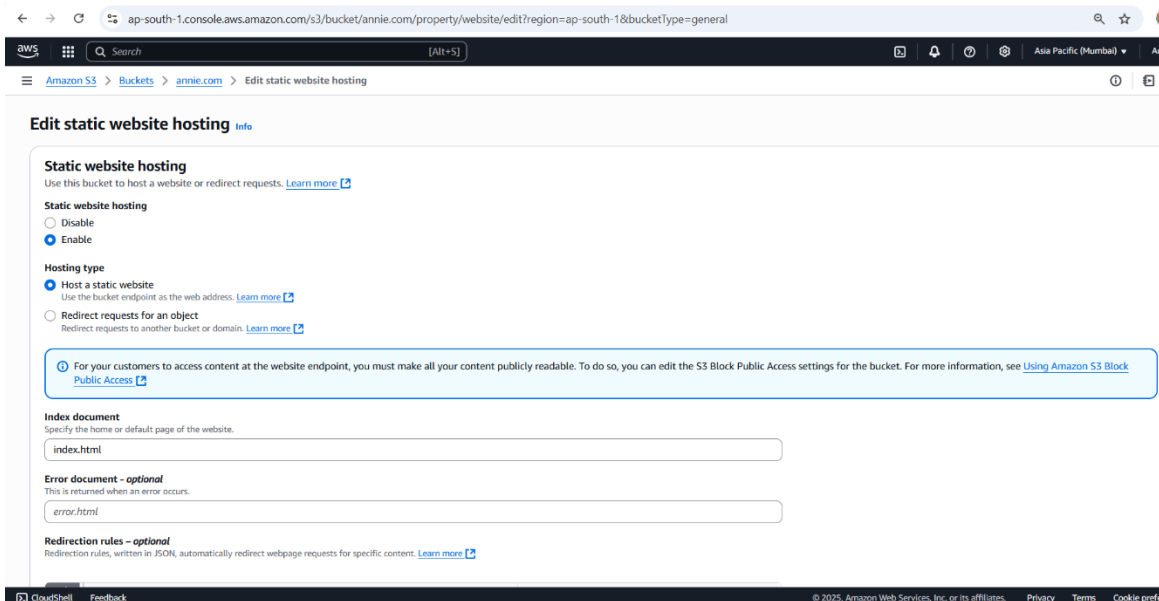
Properties
Specify storage class, encryption settings, tags, and more.

[Cancel](#) [Upload](#)

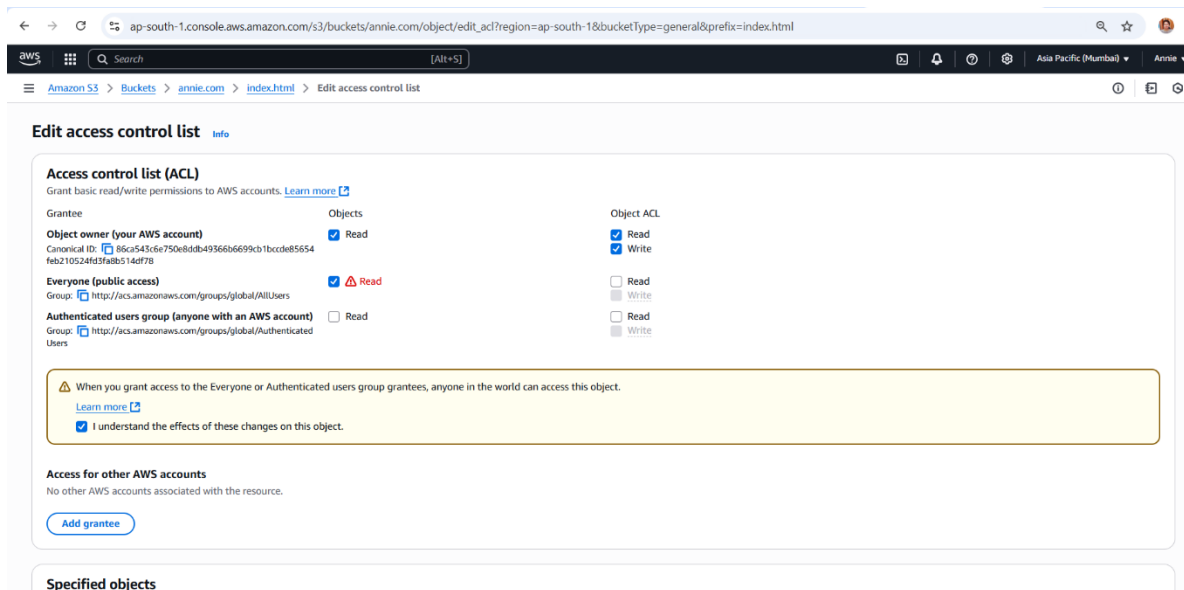
4. Open your created bucket (eg: wipro.com) go to **Properties**-> Static web Hosting->click on edit.



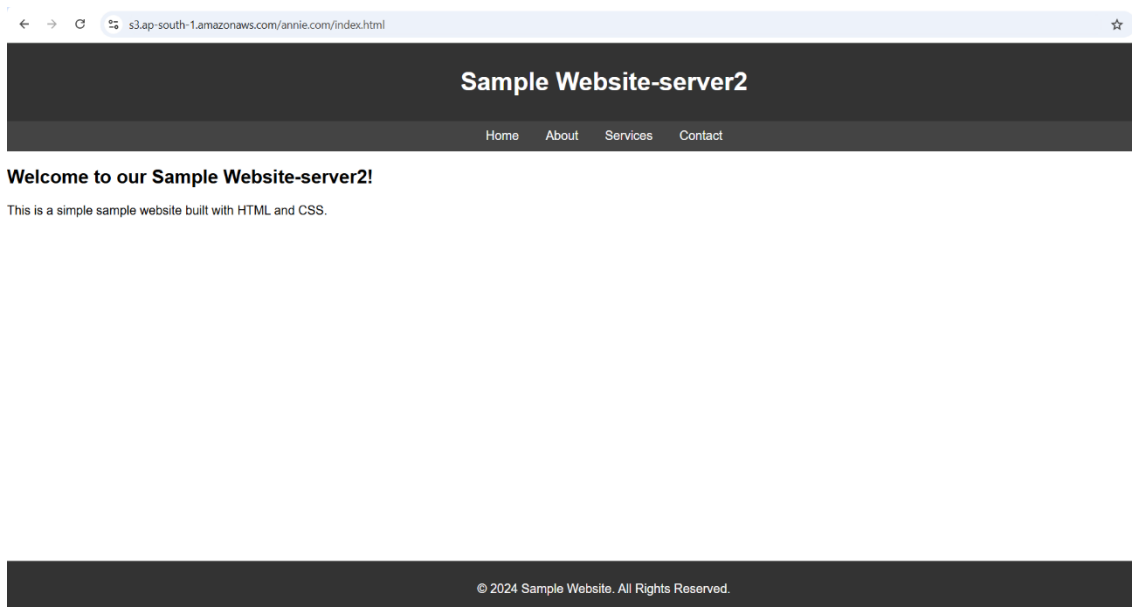
5. Enable the static web hosting->give the name of the html document that has uploaded in the bucket (eg: index.html)->save changes.



6. Open the object->go to permission->edit access control list for object give permission->save changes.



7. Reload the browser.



➤ Conclusion

Deploying a static website using Amazon S3 provides a cost-effective, scalable, and highly available solution for hosting web content. By leveraging AWS S3 for storage and integrating with services such as CloudFront for content delivery, Route 53 for DNS management, and IAM for secure access, organizations can ensure their websites are both fast and secure.

Cloud-based scalability is achieved seamlessly through S3's inherent design, allowing for automatic handling of traffic spikes without the need for manual

intervention. This approach reduces operational complexity, lowers maintenance costs, and delivers a reliable user experience at scale.

Overall, hosting static websites on AWS S3 is an ideal solution for businesses seeking to combine simplicity, performance, and scalability in their web architecture.