

# HOSTING A STATIC WEBSITE BY USING S3.

- Amazon simple storage services S3 provides the object based storage where data is stored in the form of s3 bucket in district unit which is called object instead of files. Amazon S3 buckets are similar to file folders and can be used to store, retrieve, backup and access objects. First we will start creating a s3 bucket by clicking create bucket and after that we will be able to see some general configuration in which we will select our region in this I have selected my nearest region Mumbai. We will give name to our s3 bucket which should be unique because it works globally.

The screenshot displays the AWS Management Console interface for creating a new S3 bucket. The breadcrumb navigation at the top indicates the path: Amazon S3 > Buckets > Create bucket. The main heading is 'Create bucket' with an 'Info' link. A sub-header states, 'Buckets are containers for data stored in S3.'

The 'General configuration' section contains the following fields:

- AWS Region:** A dropdown menu showing 'Asia Pacific (Mumbai) ap-south-1'.
- Bucket name:** A text input field containing 'myawsbucket'. Below the field, a note states: 'Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)'.
- Copy settings from existing bucket - optional:** A section with a note: 'Only the bucket settings in the following configuration are copied.' Below this is a 'Choose bucket' button and the format 'Format: s3://bucket/prefix'.


The 'Object Ownership' section is below the general configuration. It includes a note: 'Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.'

There are two radio button options for Object Ownership:

- ACLs disabled (recommended):** This option is selected. The description states: 'All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.'
- ACLs enabled:** This option is unselected. The description states: 'Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.'

The footer of the console shows 'CloudShell', 'Feedback', 'Language', and the copyright notice '© 2024, Amazon Web Services, Inc. or its affiliates.' along with links for 'Privacy', 'Terms', and 'Cookie preferences'.

## 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.

- In block public access settings for this bucket we will unselect the defaults and select the acknowledge point for public access. In bucket versioning we will keep it disable as we don't want it right now to prevent any datas and click to create bucket.

## 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 both unintended user actions and application failures. [Learn more](#) 

### Bucket Versioning

- ☒ Disable  
☐ Enable

- Now we will make a simple index.html file and save it to our computer and we will upload it. We will try to open it and see the object by clicking in url provided there as we can see it is saying access denied. So, we need to generate a S3 policy for this.

## Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

### Encryption type [Info](#)

- ☒ Server-side encryption with Amazon S3 managed keys (SSE-S3)
- ☐ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
- ☐ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)  
Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the [Amazon S3 pricing page](#). [↗](#)

### Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#) [↗](#)

- ☐ Disable
- ☒ Enable

### ► Advanced settings

[i](#) After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel

Create bucket

# 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 SDK 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 (0)

Remove

Add files

Add folder

All files and folders in this table will be uploaded.

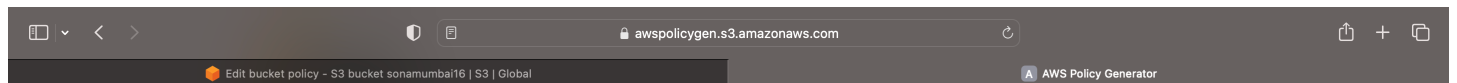
< 1 >

<input type="checkbox"/>	Name	Folder	Type
--------------------------	------	--------	------

No files or folders

You have not chosen any files or folders to upload.

- We will go to Amazon policy generator to generate a policy to see our object. In actions we will select get object and copy the ARN (Amazon resource name) from the object page and paste it to the amazon policy generator page add statement and generate policy.



## AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to [Amazon Web Services \(AWS\)](#) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#).

### Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an [IAM Policy](#), an [S3 Bucket Policy](#), an [SNS Topic Policy](#), a [VPC Endpoint Policy](#), and an [SQS Queue Policy](#).

Select Type of Policy

### Step 2: Add Statement(s)

A statement is the formal description of a single permission. See a [description of elements](#) that you can use in statements.

Effect ☒ Allow ☐ Deny

Principal

Use a comma to separate multiple values.

AWS Service  ☐ All Services ("\*")

Use multiple statements to add permissions for more than one service.

Actions  ☐ All Actions ("\*")

Amazon Resource Name (ARN)

☒ GetObject ☐ GetObjectAcl ☐ GetObjectAttributes

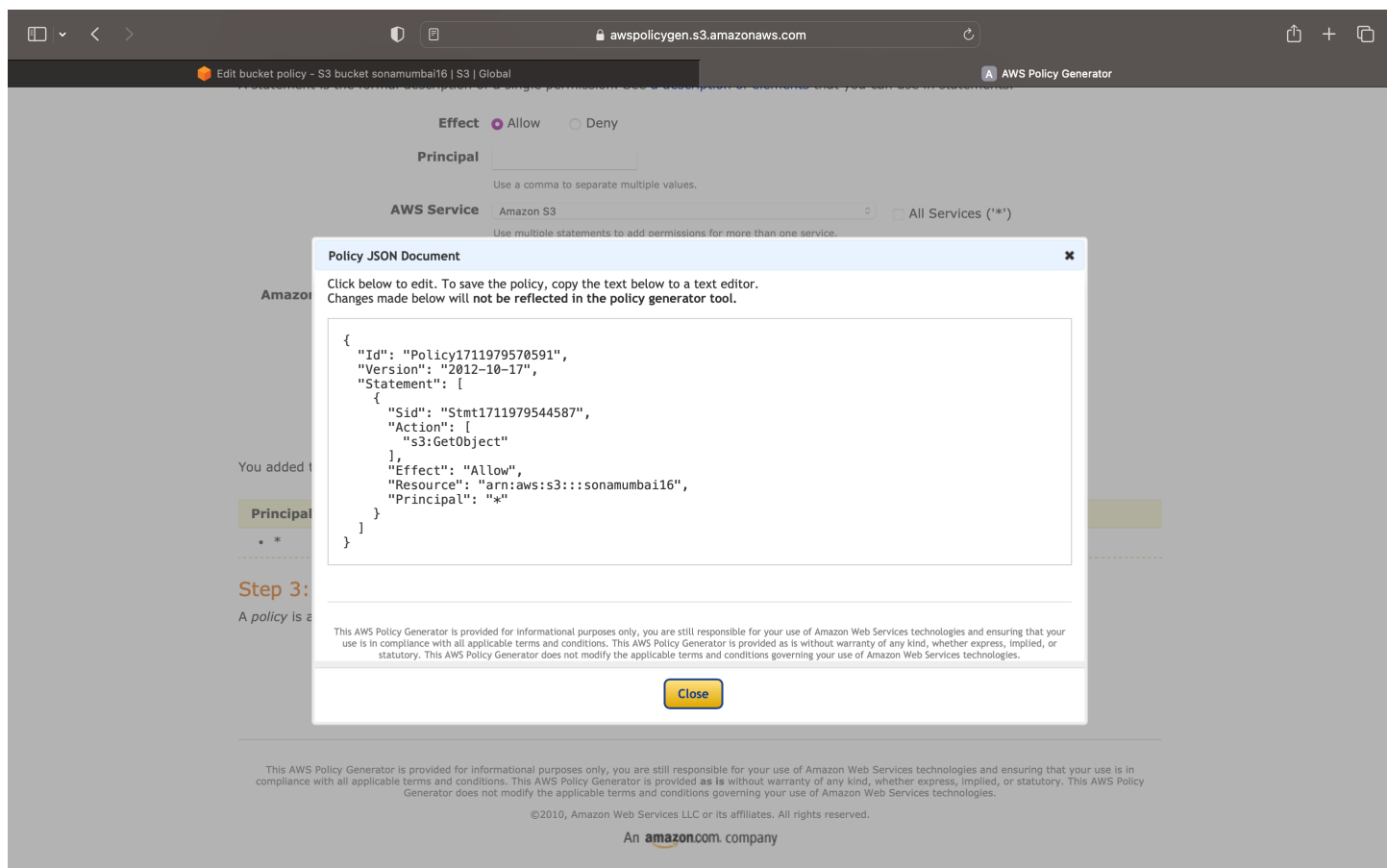
☐ GetMetricsConfiguration ☐ GetMultiRegionAccessPoint ☐ GetMultiRegionAccessPointPolicy ☐ GetMultiRegionAccessPointPolicyStatus ☐ GetMultiRegionAccessPointRoutes

**Id. You must enter a valid ARN.**

### Step 3: Generate Policy

A [policy](#) is a document (written in the [Access Policy Language](#)) that acts as a container for one or more statements.

**Add one or more statements above to generate a policy.**



- We will copy the generated JSON policy and paste it in our object permission page at bucket policy and edit it by adding /\* after bucket name and save changes. Now we will open url and we will be able to see the object.
- In properties we will change the static website hosting to enable and fill the essentials and save changes. Now we are ready to host static website by using s3.

# Edit static website hosting

**Static website hosting**  
Use this bucket to host a website or redirect requests. [Learn more](#)

Static website hosting

- ☐ Disable
- ☒ Enable

Hosting type

- ☒ Host a static website  
Use the bucket endpoint as the web address. [Learn more](#)
- ☐ Redirect requests for an object  
Redirect requests to another bucket or domain. [Learn more](#)

For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see [Using Amazon S3 Block Public Access](#)

**Index document**  
Specify the home or default page of the website.

index.html

**Error document - optional**  
This is returned when an error occurs.

error.html

**Redirection rules - optional**  
Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more](#)