

# Hosting static website in AWS using s3 bucket

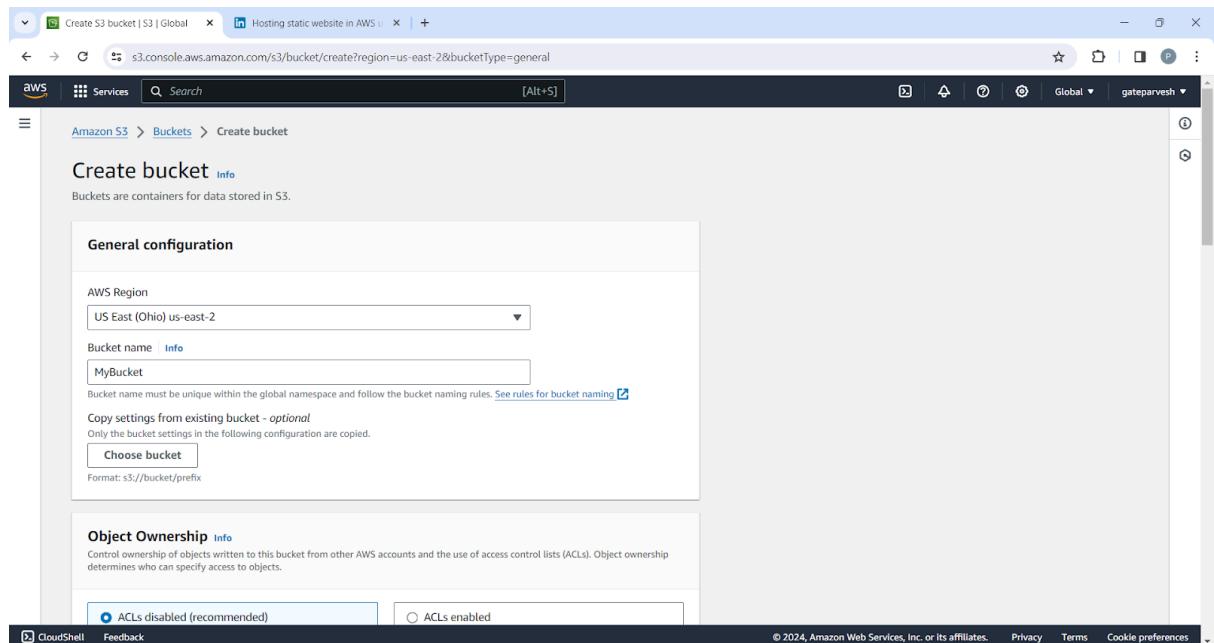
## Step 1: Create an S3 Bucket

Sign in to AWS Console: Go to the AWS Management

Console: <https://console.aws.amazon.com/>

Open S3 Console: Navigate to the S3 service.

### 1) Give name to the bucket.



### 2) Object Ownership -> ACLs enabled

The screenshot shows the 'Object Ownership' section of the AWS S3 Bucket Creation wizard. It includes two radio button options: 'ACLs disabled (recommended)' and 'ACLs enabled'. The 'ACLs enabled' option is selected, indicating that objects in the bucket can be owned by other AWS accounts. A note below recommends disabling ACLs unless specific access control is required. The 'Bucket owner preferred' radio button is selected under 'Object Ownership', which means new objects will have the bucket-owner-full-control canned ACL. A note states that if object ownership enforcement is needed, a bucket policy must specify the bucket-owner-full-control canned ACL.

### 3) Block all public access.

The screenshot shows the 'Block Public Access settings for this bucket' section of the AWS S3 Bucket Creation wizard. It lists four checkboxes for blocking public access: 'Block all public access', 'Block public access to buckets and objects granted through new access control lists (ACLS)', 'Block public access to buckets and objects granted through any access control lists (ACLS)', and 'Block public access to buckets and objects granted through new public bucket or access point policies'. The 'Block all public access' checkbox is checked. A note below explains that turning this setting on is equivalent to enabling all four individual settings. A warning message states that turning off this setting might result in the bucket becoming public, and a checkbox allows acknowledging this risk.

### 4) Enable Bucket version and remain all settings same(default)

The screenshot shows the 'Bucket Versioning' configuration page. It includes a description of what bucket versioning is, a section to enable or disable it (with 'Enable' selected), and a 'Tags - optional (0)' section where users can add tags to track storage costs and organize buckets. Below that is a 'Default encryption' section indicating server-side encryption is automatically applied to new objects. At the bottom, there are links for CloudShell and Feedback, and a copyright notice for 2024.

## 5) Now create bucket.

The screenshot shows the 'Buckets' list in the AWS S3 console. A green success message at the top states 'Successfully created bucket "mybuckets3bucket"'. Below it, the 'General purpose buckets' section lists one item: 'mybuckets3bucket' (Name), 'US East (Ohio) us-east-2' (AWS Region), 'Objects can be public' (Access), and 'March 4, 2024, 09:42:11 (UTC+05:30)' (Creation date). There are buttons for Copy ARN, Empty, Delete, and Create bucket.

## 6)

The screenshot shows the 'Objects' list for the 'mybuckets3bucket'. The top navigation bar shows 'Amazon S3 > Buckets > mybuckets3bucket'. The 'Objects' tab is selected. The main area displays a table with columns: Name, Type, Last modified, Size, and Storage class. A message at the bottom says 'No objects' and 'You don't have any objects in this bucket.' There is a 'Upload' button at the bottom right.

## 7) Set Bucket Permissions. Go to permission → Bucket Policy

The screenshot shows the AWS S3 console for a bucket named 'mybuckets3bucket'. The 'Permissions' tab is selected. Under 'Access', it says 'Objects can be public'. In the 'Block public access (bucket settings)' section, 'Block all public access' is set to 'Off'. Below that, there's a link to 'Individual Block Public Access settings for this bucket'. The 'Bucket policy' section shows a JSON policy with a note that it applies to objects in the bucket. There are 'Edit' and 'Delete' buttons for the policy.

## 8) Bucket Policy → Edit policy → policy generator

The screenshot shows the AWS Policy Generator page for the same bucket. A message at the top says 'Bucket ARN copied' with a green checkmark icon. Below it, the ARN 'arn:aws:s3:::mybuckets3bucket' is displayed. The main area is titled 'Policy' and contains a single statement: '1 | { "Effect": "Allow", "Principal": "\*", "Action": "s3:GetObject", "Resource": "arn:aws:s3:::mybuckets3bucket/\*" }'. To the right, there's a 'Select a statement' dropdown with options to 'Edit statement', 'Add new statement', or 'Delete statement'. At the bottom, there are links for 'CloudShell', 'Feedback', and copyright information.

## 9) Policy Generator →

**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 [S3 Bucket 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 [Amazon S3](#)  All Services (\*)

Actions [1 Action\(s\) Selected](#)  All Actions (\*)

Amazon Resource Name (ARN)  arn:s3:::mybuckets3bucket

ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${Keyname}.  
Use a comma to separate multiple values.

Add Conditions (Optional)

[Add Statement](#)

**Step 3: Generate Policy**

Use a comma to separate multiple values.

AWS Service [Amazon S3](#)  All Services (\*)

Actions [-- Select Actions --](#)  All Actions (\*)

Amazon Resource Name (ARN)

ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${Keyname}.  
Use a comma to separate multiple values.

Add Conditions (Optional)

[Add Statement](#)

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
* * *	Allow	s3:GetObject	arn:aws:s3:::mybuckets3bucket\*	None

**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.

[Generate Policy](#) [Start Over](#)

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable terms and conditions. This AWS Policy Generator is provided **as is** without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies.

©2010, Amazon Web Services LLC or its affiliates. All rights reserved.  
An [amazon.com](#) company

## 10) Generate Policy

The screenshot shows the AWS Policy Generator interface. A modal window displays a JSON policy document:

```
{
  "Id": "Policy1709526069023",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stat1709526062255",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::mybuckets3bucket/*",
      "Principal": "*"
    }
  ]
}
```

Below the modal, a note states: "This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable laws and regulations. Any use of this generator is provided as is without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies." A "Close" button is visible at the bottom right of the modal.

## 11) Copy Policy Json Document → Paste that in Policy

The screenshot shows the AWS S3 console with the policy editor open for a bucket named "mybuckets3bucket". The policy document is identical to the one shown in the previous screenshot:

```
{
  "Id": "Policy1709526069023",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stat1709526062255",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::mybuckets3bucket/*",
      "Principal": "*"
    }
  ]
}
```

To the right of the policy editor, there is an "Edit statement" panel with a "Select a statement" dropdown and a "+ Add new statement" button.

## 12) Click on save change your Policy added.

## 13) Click on Upload files (Buckets→ Objects)

The screenshot shows the AWS S3 console with the objects page for a bucket named "buckets123s3". The page displays a table with no objects:

Name	Type	Last modified	Size	Storage class
No objects You don't have any objects in this bucket.				
<a href="#">Upload</a>				

At the top of the page, there are tabs for "Objects", "Properties", "Permissions", "Metrics", "Management", and "Access Points". The "Objects" tab is selected. There are also buttons for "Copy S3 URI", "Copy URL", "Download", "Open", "Delete", "Actions", "Create folder", and "Upload".

## 14) Upload files (image): -

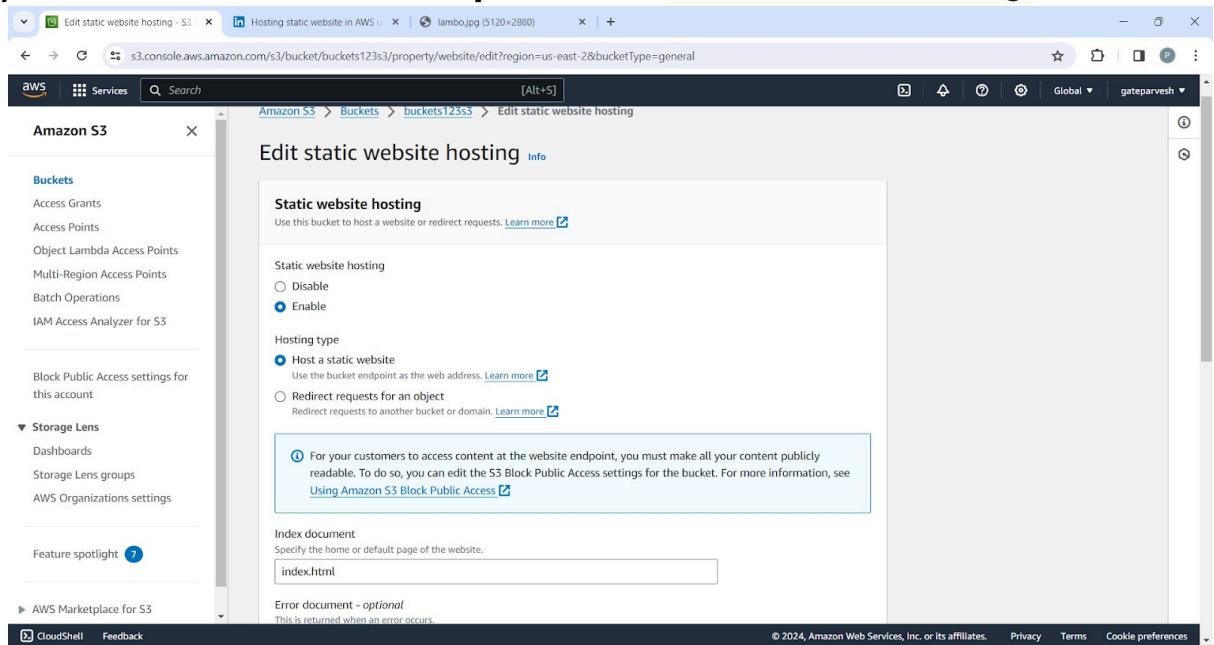
The screenshot shows the AWS S3 console interface for uploading files. The top navigation bar includes tabs for 'Upload objects - S3 bucket buck' and 'Hosting static website in AWS'. The main page title is 'Upload' under 'Amazon S3 > Buckets > buckets123s3 > Upload'. A large central area is titled 'Upload' with a sub-section 'Info'. Below this is a text box with the placeholder 'Drag and drop files and folders you want to upload here, or choose Add files or Add folder.' A table titled 'Files and folders (1 Total, 1.9 MB)' lists one file: 'lambo.jpg' (image/jpeg). There are buttons for 'Remove', 'Add files', and 'Add folder'. To the right of the file list is a search bar labeled 'Find by name' and a pagination indicator '1'. Below the file list is a section titled 'Destination' with the URL 's3://buckets123s3'. At the bottom of the page are links for 'CloudShell', 'Feedback', and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates.' and 'Privacy Terms Cookie preferences'.

## 15) Click on the file you upload → open that → copy Object URL

The screenshot shows the AWS S3 console interface for viewing an uploaded file. The top navigation bar includes tabs for 'Hosting static website in AWS' and 's3.console.aws.amazon.com'. The main page title is 'lambo.jpg - Object in S3 bucket' under 'Amazon S3 > Buckets > buckets123s3 > lambo.jpg'. On the left is a sidebar with sections like 'Buckets', 'Storage Lens', and 'Feature spotlight'. The main content area shows the file 'lambo.jpg' with the 'Properties' tab selected. The 'Object overview' section displays details such as Owner (3c79a92bc9ba0b5100987b62b54dc684fa091dcba1d78d905ba440182dbbc2), AWS Region (US East (Ohio) us-east-2), Last modified (March 4, 2024, 10:05:26 (UTC+0:30)), Size (1.9 MB), Type (jpg), and Key (lambo.jpg). To the right, there are buttons for 'Copy S3 URI', 'Download', 'Open', and 'Object actions'. A green callout bubble with the text 'Object URL Copied' points to the 'Copy S3 URI' button. Below the button is the copied URL: 'https://buckets123s3.s3.us-east-2.amazonaws.com/lambo.jpg'. The bottom of the page includes links for 'CloudShell', 'Feedback', and copyright information: '© 2024, Amazon Web Services, Inc. or its affiliates.' and 'Privacy Terms Cookie preferences'.

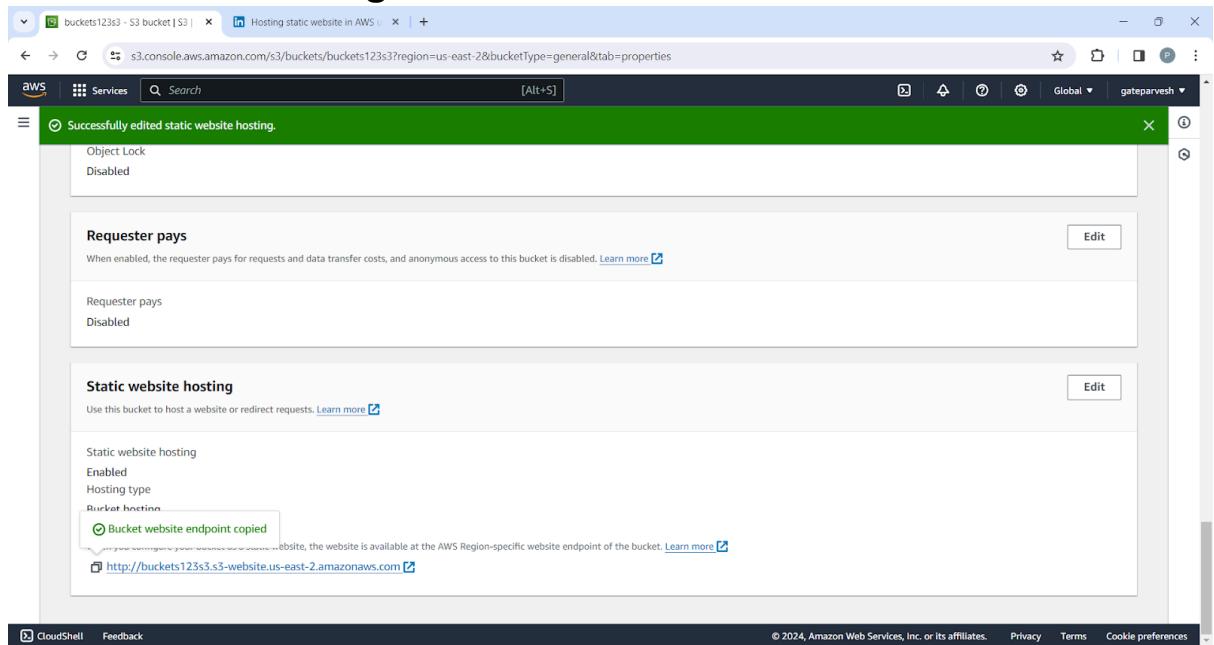
# Static website hosting

16) Goes to Bucket → Properties → Static website hosting.



The screenshot shows the 'Edit static website hosting' configuration for the 'buckets123s3' bucket. The 'Static website hosting' section has the 'Enable' radio button selected. Under 'Hosting type', the 'Host a static website' radio button is selected. A note states: '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.' The 'Index document' field is set to 'index.html'.

Click on save change.



The screenshot shows a confirmation message: 'Successfully edited static website hosting.' It lists the following configurations: Object Lock (Disabled), Requester pays (Disabled), and Static website hosting (Enabled). A note says: 'When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled.' Below this, it says: 'Bucket website endpoint copied' and provides the URL 'http://buckets123s3.s3-website.us-east-2.amazonaws.com'.

17) Copy URL in from the static website hosting and paste in new tab your website is hosted.

## 18) Goes to Bucket → Properties → Bucket Version

The screenshot shows the 'Edit Bucket Versioning' page in the AWS Management Console. The left sidebar has 'Buckets' selected. The main content area is titled 'Edit Bucket Versioning'. It contains a section titled 'Bucket Versioning' with a description: '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.' Below this is a 'Bucket Versioning' section with two options: 'Suspend' (radio button) and 'Enable' (radio button, which is selected). A note about 'Multi-factor authentication (MFA) delete' follows, mentioning it's disabled. At the bottom are 'Cancel' and 'Save changes' buttons.

## 19) Click on show version in bucket it shows all file version.

The screenshot shows the 'Objects (10)' page in the AWS Management Console. The top navigation bar includes 'CloudShell' and 'Feedback'. The main content area displays a table of objects. A 'Show versions' button is visible above the table. The table columns are: Name, Type, Version ID, Last modified, Size, and Storage class. The data in the table is as follows:

Name	Type	Version ID	Last modified	Size	Storage class
app.js	Delete marker	yHJEXKGca... 44MSkzgUB... 8_HuPgePX... HP6B	March 4, 2024, 10:39:16 (UTC+05:30)	0 B	-
app.js	js	P5zJaR7D1p... yJMMRQ5Gs... hs5ZJMMkD... oHO2	March 4, 2024, 10:34:48 (UTC+05:30)	842.0 B	Standard
img/	Folder	-	-	-	-
img1.jpg	jpg	At9OITdxgF... N66y1qwJct... D5WAVQe6... 9bBS	March 4, 2024, 10:40:59 (UTC+05:30)	2.8 MB	Standard
index.html	Delete marker	ErGqEOoan... 3Q3PlkbBy... HynVNzqGu... 3516	March 4, 2024, 10:39:16 (UTC+05:30)	0 B	-
		5nMA5XQh... nLe2WsoSE	March 4, 2024, 10:35:00		

## 20) Then Off show version and delete a file.

Here the 2-image upload in bucket.

The screenshot shows the AWS S3 console interface. At the top, there are tabs for 'Objects', 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. Below this, a table lists objects in the 'buckets123s3' bucket. The table has columns for Name, Type, Last modified, Size, and Storage class. Two files are listed: 'img1.jpg' and 'img2.jpg', both of which are jpg files. The last modified date for both files is March 4, 2024, at 10:40:59 (UTC+05:30). The size of 'img1.jpg' is 2.8 MB and 'img2.jpg' is 4.8 MB, both in the Standard storage class.

## Delete file.

The screenshot shows a confirmation dialog for deleting an object. It includes a warning message about folder deletion, a note about delete markers, and a list of specified objects. The 'Delete objects?' section contains a text input field with the word 'delete' and a 'Delete objects' button. The 'Specified objects' table shows one item: 'img1.jpg' (jpg type, 2.8 MB).

21) Now again click on show version where you can find the delete file with TYPE “Delete Marker”.

The screenshot shows the AWS S3 console interface. The top navigation bar includes tabs for Services, Search, and [Alt+S]. Below the navigation is a breadcrumb trail: Amazon S3 > Buckets > buckets123s3. The main content area is titled "buckets123s3" and shows the "Objects" tab selected. A red box highlights the "Publicly accessible" status. The object list table has columns for Name, Type, Version ID, Last modified, Size, and Storage class. The objects listed are:

Name	Type	Version ID	Last modified	Size	Storage class
img1.jpg	Delete marker	WJPRyw8RoOr8cj1a4OEnCaDjFafhYm4	March 4, 2024, 11:08:19 (UTC+05:30)	0 B	-
img1.jpg	jpg	_g4Tlvh.2Ry8xzEjj2sEjzR6o3nTqBE	March 4, 2024, 11:05:53 (UTC+05:30)	2.8 MB	Standard
img2.jpg	jpg	DjtEajZq2q3EwwojDanp92YnrkahkU	March 4, 2024, 11:06:00 (UTC+05:30)	4.8 MB	Standard

## 22) Now delete the Delete marker (file) permanently delete.

The screenshot shows the AWS S3 console interface for deleting objects. The top navigation bar includes tabs for Services, Search, and [Alt+S]. Below the navigation is a breadcrumb trail: Amazon S3 > Buckets > buckets123s3 > object/delete?<object>. The main content area displays a warning message about the permanence of deletions and a table of specified objects. The table shows one object: img1.jpg (Delete marker). Below the table is a section for confirming the deletion, with a text input field containing "permanently delete".

**Specified objects**

Name	Version ID	Type	Last modified
img1.jpg	WJPRyw8RoOr8cj1a4OEnCaDjFafhYm4	Delete marker	March 4, 2024, 11:08:19 (UTC+05:30)

**Permanently delete objects?**

To confirm deletion, type *permanently delete* in the text input field.

permanently delete

Cancel   **Delete objects**

## 23) Go to bucket your file will restore automatically.