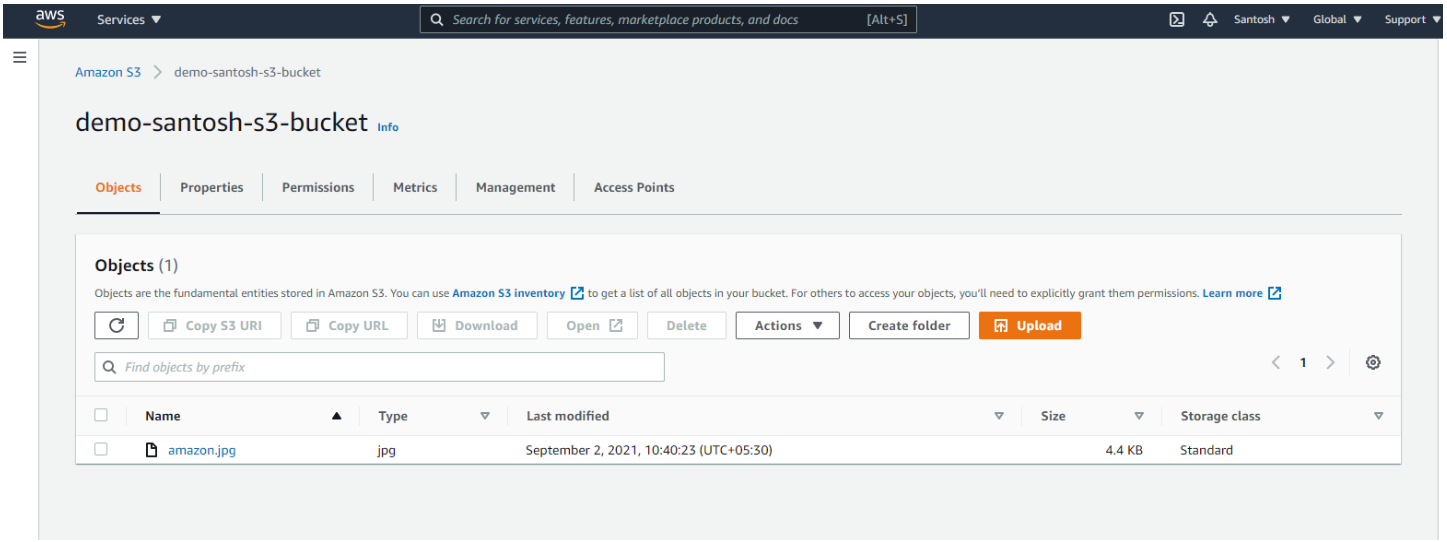
Amazon S3

Amazon S3 Overview:

- Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance.
- Amazon S3 allows people to store objects(file) in buckets.
- Buckets must have a globally unique name.
- Buckets are defined at the region level.
- Objects are files and they must have a key.



demo-santosh-s3-bucket

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	amazon.jpg	jpg	September 2, 2021, 10:40:23 (UTC+05:30)	4.4 KB	Standard

Created a bucket and upload the object(file).

Amazon S3 versioning:

- We can version our files in amazon s3.
- Easy to rollback to previous version.
- It is enabled at the bucket level.
- If you reupload a file then it will create a new version of that file.

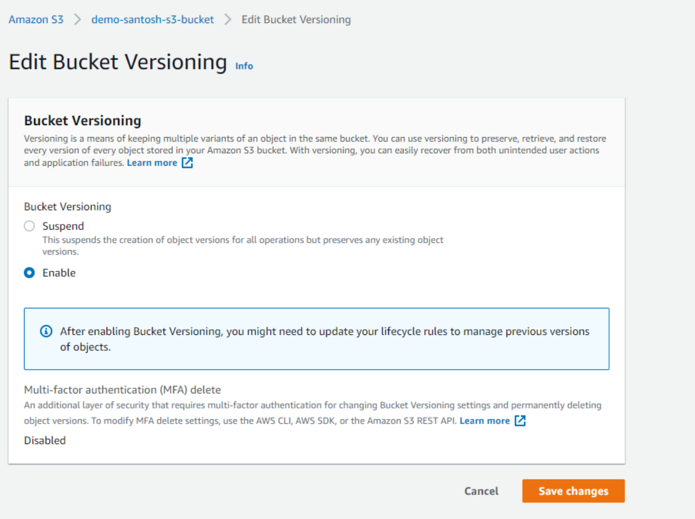
(So instead of overwriting the file that already exists it will create a new file version)

Note:

1. Any file that is not versioned prior to enable versioning will have version "null".
2. Suspending versioning does not delete the previous versions.

Steps:

1. Enable the Bucket Versioning



Amazon S3 > demo-santosh-s3-bucket > Edit Bucket Versioning

Edit Bucket Versioning Info

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

○ Suspend
This suspends the creation of object versions for all operations but preserves any existing object versions.

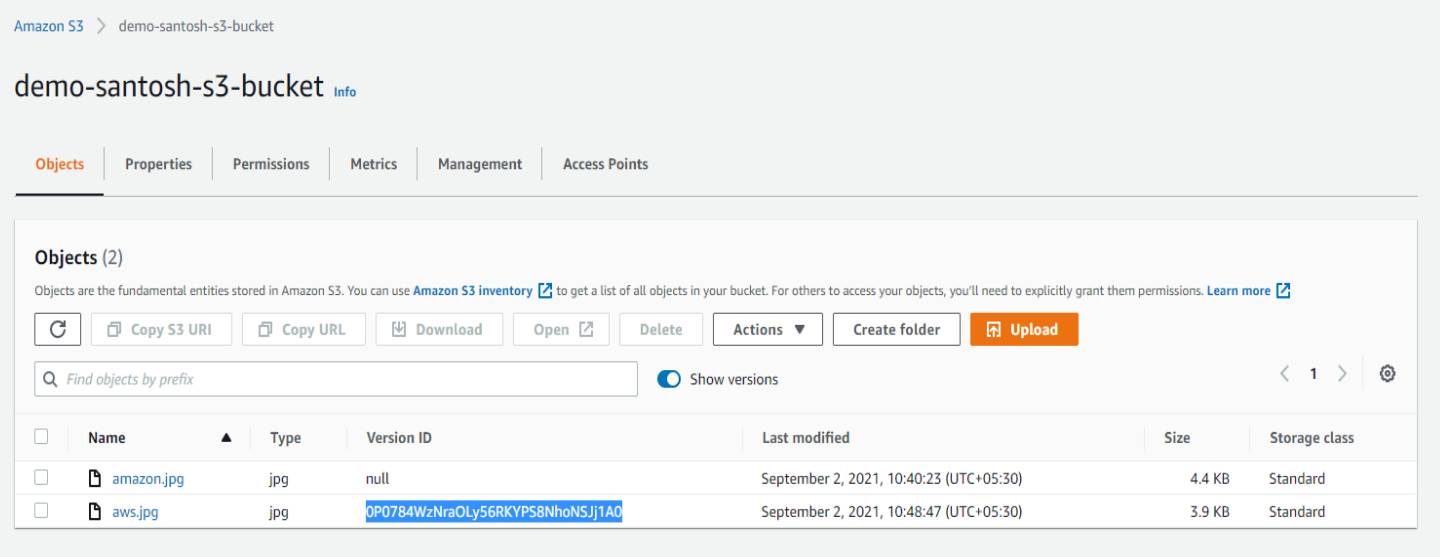
● Enable

After enabling Bucket Versioning, you might need to update your lifecycle rules to manage previous versions of objects.

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. Learn more

Disabled

2. Upload the file



Amazon S3 > demo-santosh-s3-bucket

demo-santosh-s3-bucket Info

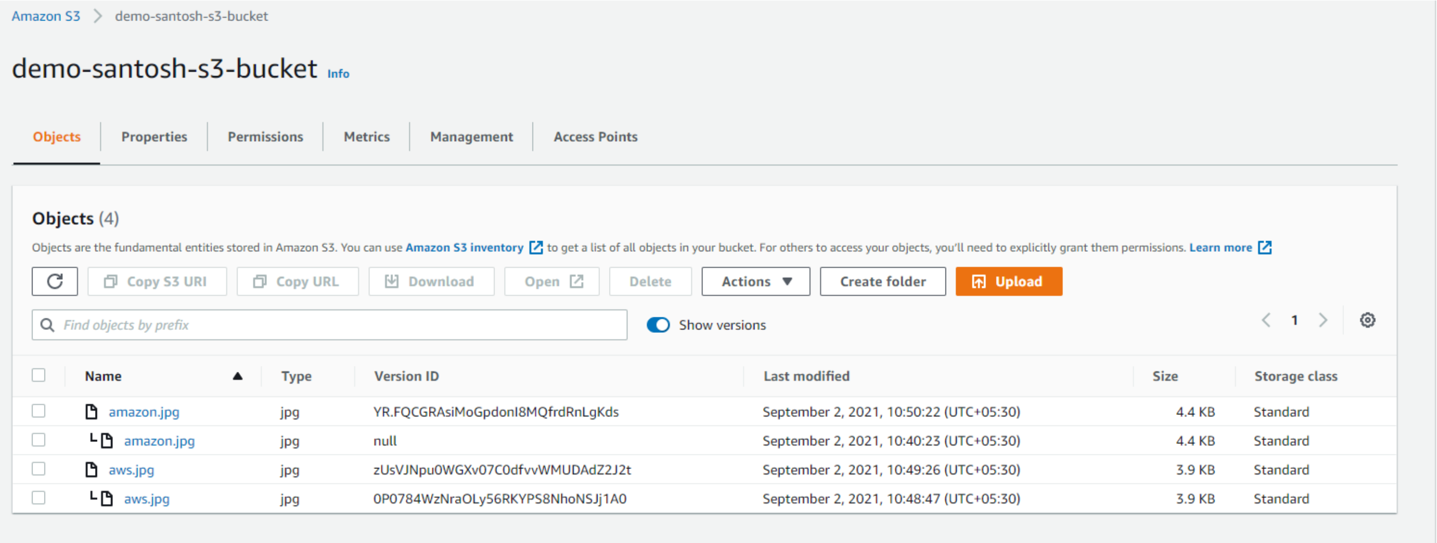
Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Version ID	Last modified	Size	Storage class
□	amazon.jpg	jpg	null	September 2, 2021, 10:40:23 (UTC+05:30)	4.4 KB	Standard
□	aws.jpg	jpg	0P0784WzNraOLy56RKYPS8NhoNSJj1A0	September 2, 2021, 10:48:47 (UTC+05:30)	3.9 KB	Standard

Here you will see the currently uploaded file have Version ID and the old file which are uploaded before the enabling versioning will have Version ID "null".

3. Upload the same file and click on show versions you will see all versions of files.



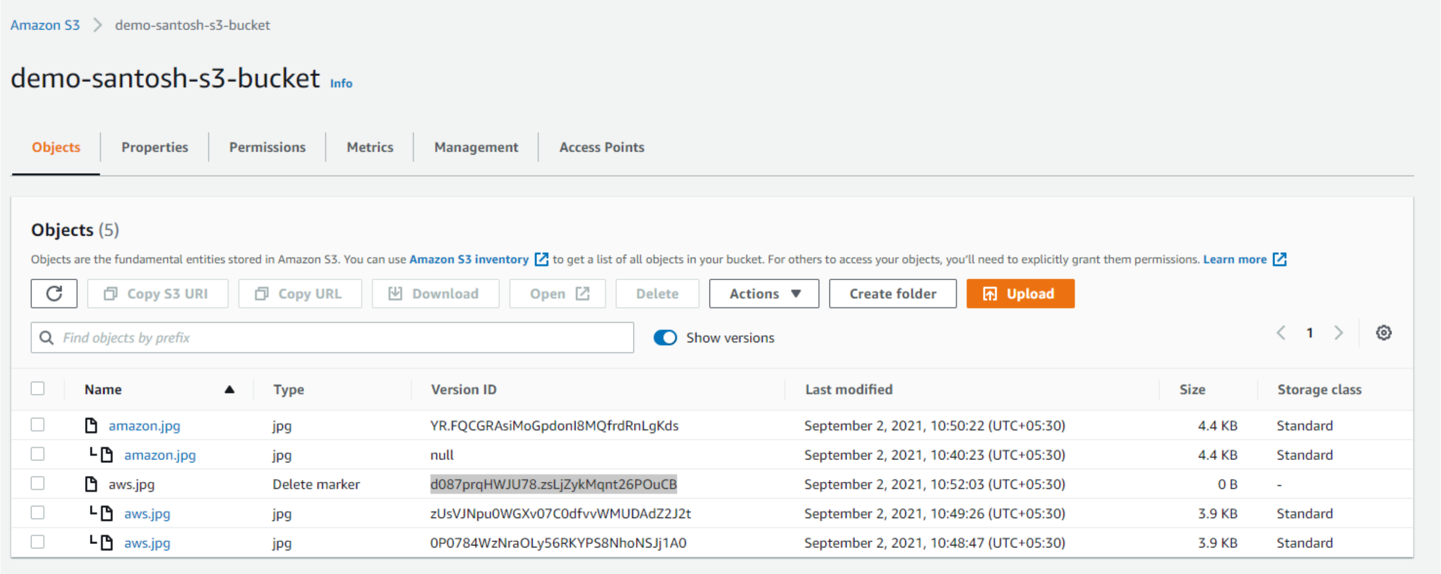
Objects (4)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Version ID	Last modified	Size	Storage class
□	amazon.jpg	jpg	YR.FQCGRAsiMoGpdonl8MQfrdRnLgKds	September 2, 2021, 10:50:22 (UTC+05:30)	4.4 KB	Standard
□	└ amazon.jpg	jpg	null	September 2, 2021, 10:40:23 (UTC+05:30)	4.4 KB	Standard
□	aws.jpg	jpg	zUsVJNpu0WGXv07C0dfvvWMUDAdZ2J2t	September 2, 2021, 10:49:26 (UTC+05:30)	3.9 KB	Standard
□	└ aws.jpg	jpg	0P0784WzNraOLy56RKYPS8NhoNSJj1A0	September 2, 2021, 10:48:47 (UTC+05:30)	3.9 KB	Standard

4. Delete the object(file)

After deleting object(file) from bucket, you will see a delete marker on your deleted object(file). Delete marker has its own version id of size 0 Bytes. This delete marker showing that this file is deleted but it’s not completely deleted. Your old version of file is still there.



Objects (5)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

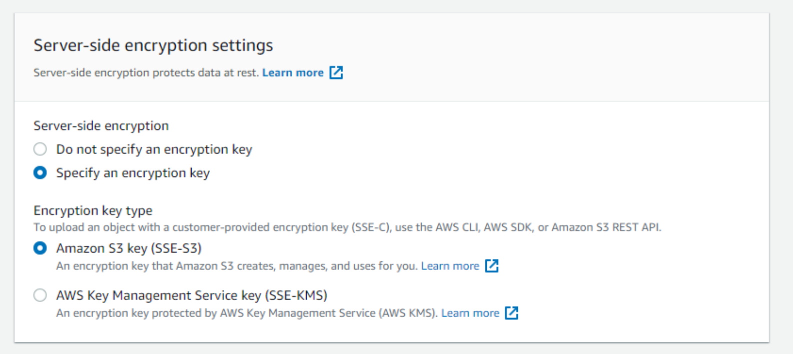
□	Name ▲	Type	Version ID	Last modified	Size	Storage class
□	amazon.jpg	jpg	YR.FQCGRAsiMoGpdonl8MQfrdRnLgKds	September 2, 2021, 10:50:22 (UTC+05:30)	4.4 KB	Standard
□	└ amazon.jpg	jpg	null	September 2, 2021, 10:40:23 (UTC+05:30)	4.4 KB	Standard
□	aws.jpg	Delete marker	d087prqHWJU78.zsLjZykMqnt26POuCB	September 2, 2021, 10:52:03 (UTC+05:30)	0 B	-
□	└ aws.jpg	jpg	zUsVJNpu0WGXv07C0dfvvWMUDAdZ2J2t	September 2, 2021, 10:49:26 (UTC+05:30)	3.9 KB	Standard
□	└ aws.jpg	jpg	0P0784WzNraOLy56RKYPS8NhoNSJj1A0	September 2, 2021, 10:48:47 (UTC+05:30)	3.9 KB	Standard

Note: Deleting delete marker or deleting specific version file it will be permanent delete.

Amazon S3 Encryption:

There are 4 methods of encrypting objects in S3.

1. SSE-S3:
   - Encryption using keys handled and managed by amazon S3.
   - Object is encrypted at server side.
   - AES-256 encryption type (algorithm).
   - To upload an object(file) and sets the SSE-S3 encryption you must set header called "x-amz-server-side-encryption":" AES256".



Server-side encryption settings

Server-side encryption protects data at rest. Learn more

Server-side encryption

○ Do not specify an encryption key

● Specify an encryption key

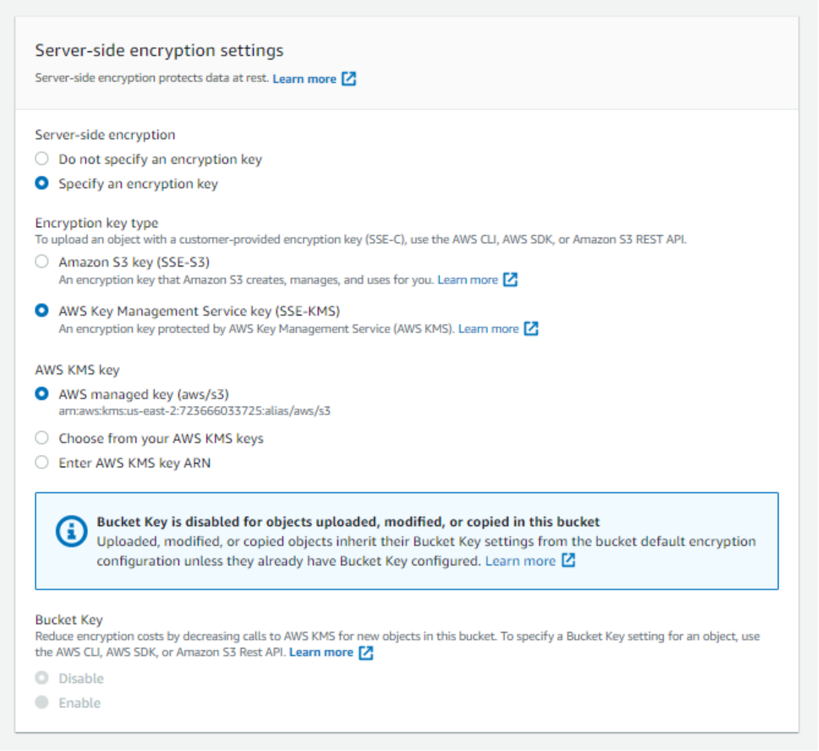
Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

● Amazon S3 key (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. Learn more

○ AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more

2. SSE-KMS:
   - It is a key management service (encryption service).
   - Encryption using keys handled and managed by KMS.
   - KMS advantages: user control + audit trail.
   - Object is encrypted at server side.
   - Must set header: "x-amz-server-side-encryption":"aws:kms"



Server-side encryption settings

Server-side encryption protects data at rest. Learn more

Server-side encryption

○ Do not specify an encryption key

● Specify an encryption key

Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

○ Amazon S3 key (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. Learn more

● AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more

AWS KMS key

● AWS managed key (aws/s3)
arn:aws:kms:us-east-2:723666033725:alias/aws/s3

○ Choose from your AWS KMS keys

○ Enter AWS KMS key ARN

Bucket Key is disabled for objects uploaded, modified, or copied in this bucket
Uploaded, modified, or copied objects inherit their Bucket Key settings from the bucket default encryption configuration unless they already have Bucket Key configured. Learn more

Bucket Key

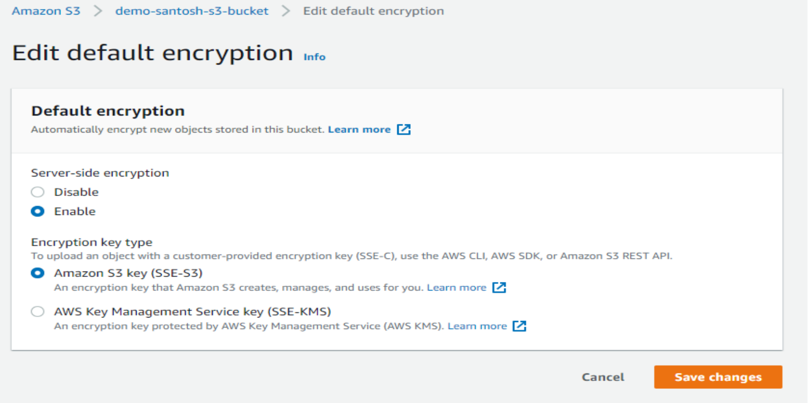
Reduce encryption costs by decreasing calls to AWS KMS for new objects in this bucket. To specify a Bucket Key setting for an object, use the AWS CLI, AWS SDK, or Amazon S3 Rest API. Learn more

○ Disable

● Enable

3. SSE-C:
    - Server-side encryption using data keys fully managed by the customer outside of AWS.
    - Amazon S3 does not store encryption key you provide.
    - HTTPS must be used.
    - Encryption key must provide in HTTP headers for every HTTP request mode.
    - We can only do this encryption on CLI. Because we have to pass encryption key into AWS securely to encrypt the object.
4. Client-side encryption:
    - Client must encrypt the object before uploading it to Amazon S3.
    - Client must decrypt object themselves when retrieving from S3.
    - Client fully manage the keys and encryption.

- Instead of enabling encryption when uploading the objects, we could set default encryption on S3 bucket.



Amazon S3 > demo-santosh-s3-bucket > Edit default encryption

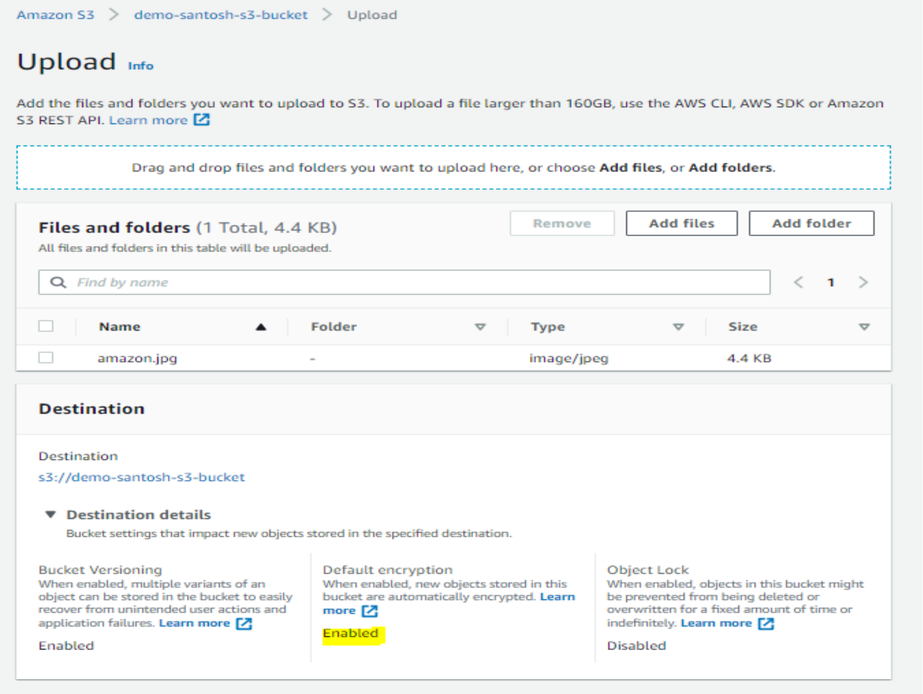
Edit default encryption Info

Default encryption
Automatically encrypt new objects stored in this bucket. Learn more

Server-side encryption
○ Disable
● Enable

Encryption key type
To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.
● Amazon S3 key (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. Learn more
○ AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more

When you upload the object, you will see that the default encryption is enabled.



Amazon S3 > demo-santosh-s3-bucket > 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 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 folders**.

Files and folders (1 Total, 4.4 KB)
All files and folders in this table will be uploaded.

□	Name	Folder	Type	Size
□	amazon.jpg	-	image/jpeg	4.4 KB

Destination

Destination
s3://demo-santosh-s3-bucket

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

Bucket Versioning
When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended user actions and application failures. Learn more
Enabled

Default encryption
When enabled, new objects stored in this bucket are automatically encrypted. Learn more
Enabled

Object Lock
When enabled, objects in this bucket might be prevented from being deleted or overwritten for a fixed amount of time or indefinitely. Learn more
Disabled

Note: - You can edit default encryption type while uploading the file

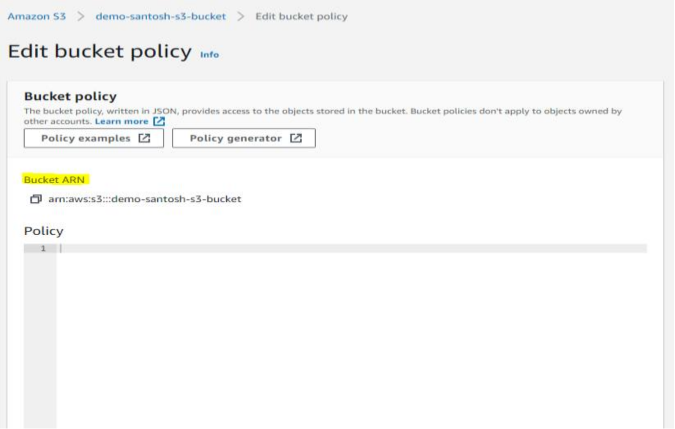
S3 Bucket policies:

- **JSON based policies:**
    1) Resources: Bucket and objects.
    2) Actions set of API to allow or Deny.
    3) Effect: Allow/Deny
    4) Principle: The account or user to apply the policy to
- **Use of S3 bucket policies:**
    1) Grant public access to the bucket.
    2) Force objects to be encrypted at upload.
    3) Grant access to another account (cross account).

Creating policies to upload only SSE-S3 encryption objects.

Steps:

1. Edit bucket policy and generate the policy.



Amazon S3 > demo-santosh-s3-bucket > Edit bucket policy

Edit bucket policy Info

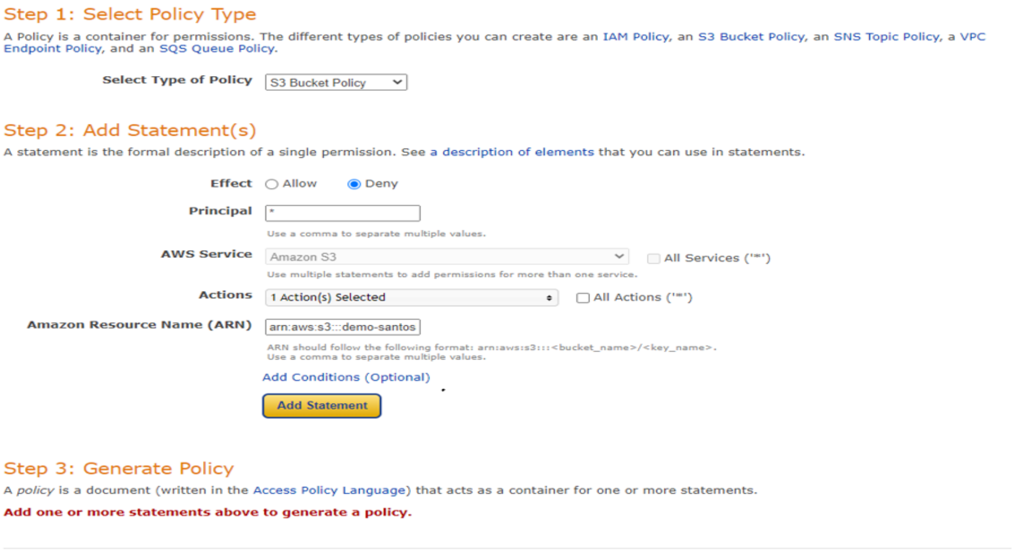
Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

Bucket ARN

arn:aws:s3:::demo-santosh-s3-bucket

Policy



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 ('*')

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

Actions: 1 Action(s) Selected □ All Actions ('*')

Amazon Resource Name (ARN): arn:aws:s3:::demo-santos

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

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.

a. Select S3 bucket policy
b. Adding 1st statement
c. Select effect deny -> Principle is * (All)
d. Select action PutObject

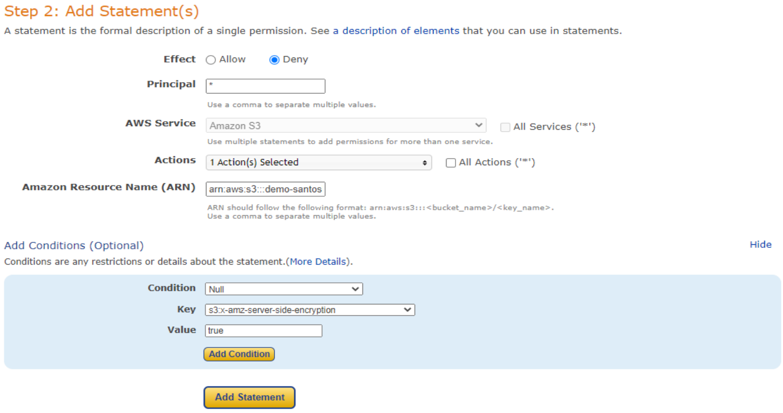
e. Set ARN copy from edit bucket policy and paste it and end of the ARN add /* . This * indicates any object within that bucket. Example: arn:aws:s3:::demo-santosh-s3-bucket/*

f. Add condition

   i. Condition = null

   ii. Key = S3:x-amz-server-side-encryption

   iii. Value = true



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 ('*')

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

Actions: 1 Action(s) Selected □ All Actions ('*')

Amazon Resource Name (ARN): arn:aws:s3:::demo-santos

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

Add Conditions (Optional) Hide

Conditions are any restrictions or details about the statement.(More Details).

Condition: Null

Key: s3:x-amz-server-side-encryption

Value: true

In this condition we set if header(key) is null then deny. If header(key) is null we are sending the file and we don't ask for any kind of encryption.

Add condition -> Add statement.

g. Adding 2nd statement

h. Select effect deny -> Principle is * (All)

i. Select action PutObject

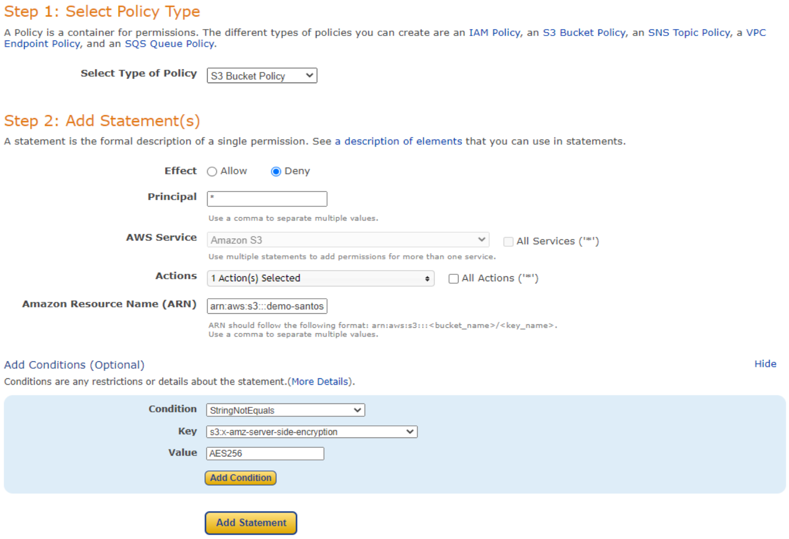
j. Set ARN copy from edit bucket policy and paste it and end of the ARN add /*. This * indicates any object within that bucket. Example: arn:aws:s3:::demo-santosh-s3-bucket/*

k. Add condition

   iv. Condition = StringNotEqual

   v. Key = S3:x-amz-server-side-encryption

   vi. Value = AES256



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 ('*')

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

Actions: 1 Action(s) Selected □ All Actions ('*')

Amazon Resource Name (ARN): arn:aws:s3:::demo-santos

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

Add Conditions (Optional) Hide

Conditions are any restrictions or details about the statement.(More Details).

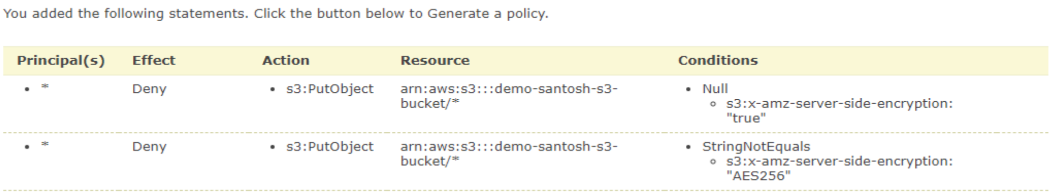
Condition: StringNotEquals

Key: s3:x-amz-server-side-encryption

Value: AES256

In this condition If the file is uploaded with the header. But the header value is not equal to AES256(SSE-S3). Means if the object is not encrypted with SSE-S3 then Deny.

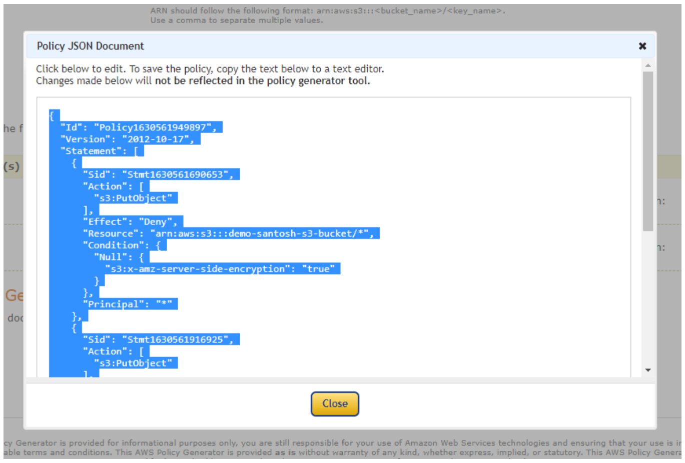
Add condition -> Add the statement



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

Principal(s)	Effect	Action	Resource	Conditions
• *	Deny	• s3:PutObject	arn:aws:s3:::demo-santosh-s3-bucket/*	• Null ∘ s3:x-amz-server-side-encryption: "true"
• *	Deny	• s3:PutObject	arn:aws:s3:::demo-santosh-s3-bucket/*	• StringNotEquals ∘ s3:x-amz-server-side-encryption: "AES256"

I. Generate the policy -> copy that JSON code and paste it in S3 bucket policy -> Add Save the policy.



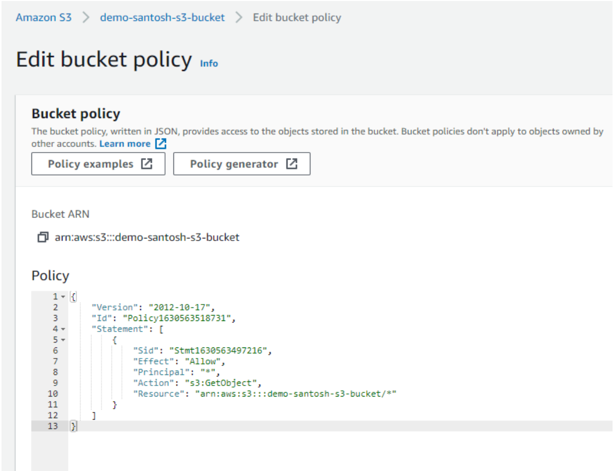
ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor.
Changes made below will **not be reflected in the policy generator tool.**

```
{
  "Id": "Policy1630561949897",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1630561690653",
      "Action": [
        "s3:PutObject"
      ],
      "Effect": "Deny",
      "Resource": "arn:aws:s3:::demo-santosh-s3-bucket/*",
      "Condition": {
        "Null": {
          "s3:x-amz-server-side-encryption": "true"
        }
      },
      "Principal": "*"
    },
    {
      "Sid": "Stmt1630561916925",
      "Action": [
        "s3:PutObject"
      ],
```

cy 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 i
able terms and conditions. This AWS Policy Generator is provided **as is** without warranty of any kind, whether express, implied, or statutory. This AWS Policy Genera



Edit bucket policy Info

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

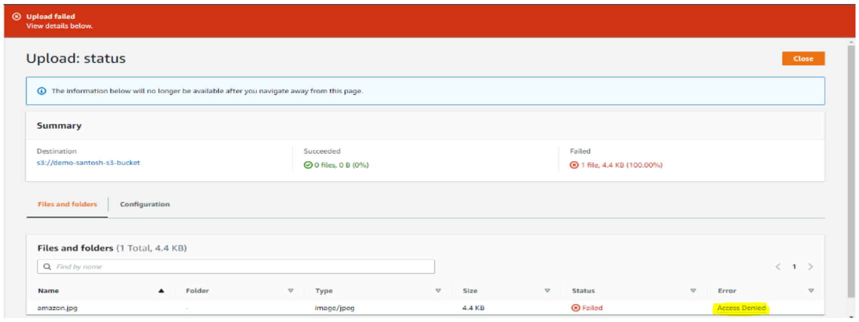
Bucket ARN

arn:aws:s3:::demo-santosh-s3-bucket

Policy

```
{
    "Version": "2012-10-17",
    "Id": "Policy1630563518731",
    "Statement": [
        {
            "Sid": "Stmt1630563497216",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::demo-santosh-s3-bucket/*"
        }
    ]
}
```

2. If you try to upload the object(file) without encryption then it will be Access denied because of the policies which we are created.



Upload failed
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

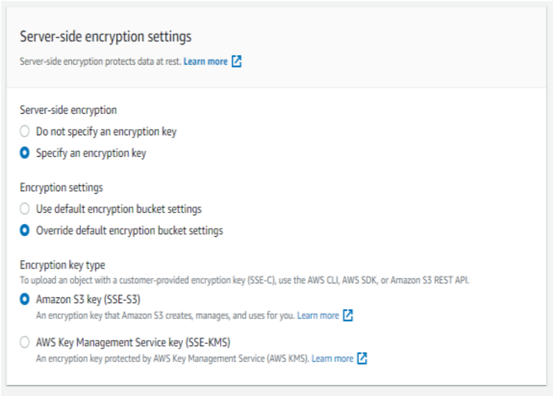
Summary

Destination	Succeeded	Failed
s3://demo-santosh-s3-bucket	0 files, 0 B (0%)	1 file, 4.4 KB (100.00%)

Files and folders (1 Total, 4.4 KB)

Name	Folder	Type	Size	Status	Error
amazon.jpg	-	image/jpeg	4.4 KB	Failed	Access Denied

3. If you try to upload object(file) but this time specify the encryption (SSE-S3) then this should be uploaded.



Server-side encryption settings

Server-side encryption protects data at rest. Learn more

Server-side encryption

○ Do not specify an encryption key
● Specify an encryption key

Encryption settings

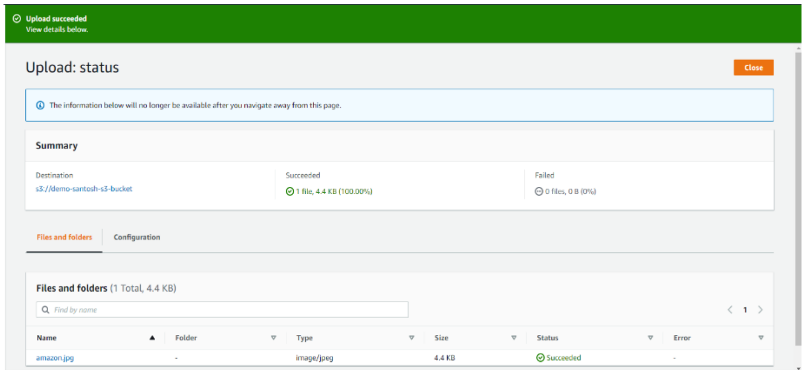
○ Use default encryption bucket settings
● Override default encryption bucket settings

Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

● Amazon S3 key (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. Learn more

○ AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more



Upload succeeded
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

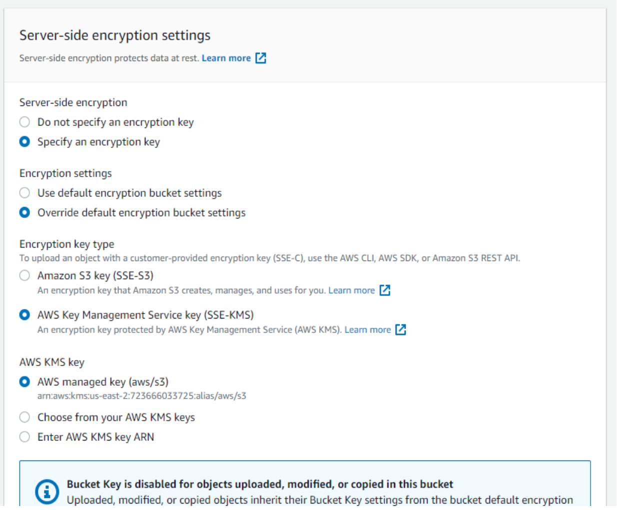
Summary

Destination	Succeeded	Failed
s3://demo-santosh-s3-bucket	1 file, 4.4 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 Total, 4.4 KB)

Name	Folder	Type	Size	Status	Error
amazon.jpg	-	image/jpeg	4.4 KB	Succeeded	-

4. If you try to upload object(file) but this time specify the other encryption(SSE-KMS) not SSE-S3 encryption then this should be access denied.



Server-side encryption settings

Server-side encryption protects data at rest. Learn more

Server-side encryption

○ Do not specify an encryption key

● Specify an encryption key

Encryption settings

○ Use default encryption bucket settings

● Override default encryption bucket settings

Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

○ Amazon S3 key (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. Learn more

● AWS Key Management Service key (SSE-KMS)
An encryption key protected by AWS Key Management Service (AWS KMS). Learn more

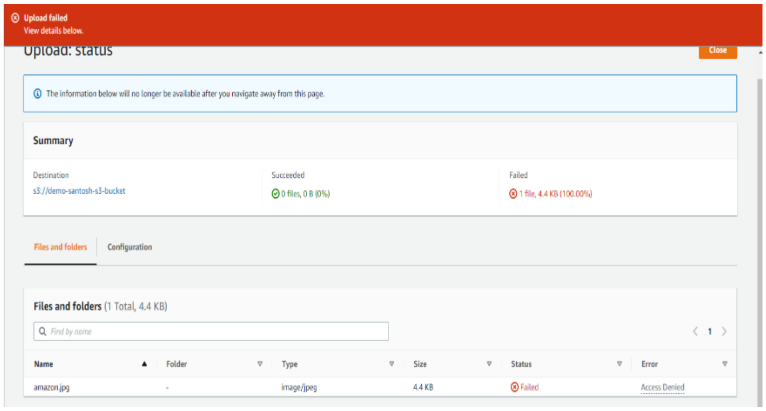
AWS KMS key

● AWS managed key (aws/s3)
arn:aws:kms:us-east-2:723666033725:alias/aws/s3

○ Choose from your AWS KMS keys

○ Enter AWS KMS key ARN

Bucket Key is disabled for objects uploaded, modified, or copied in this bucket
Uploaded, modified, or copied objects inherit their Bucket Key settings from the bucket default encryption



Upload failed
View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

Summary

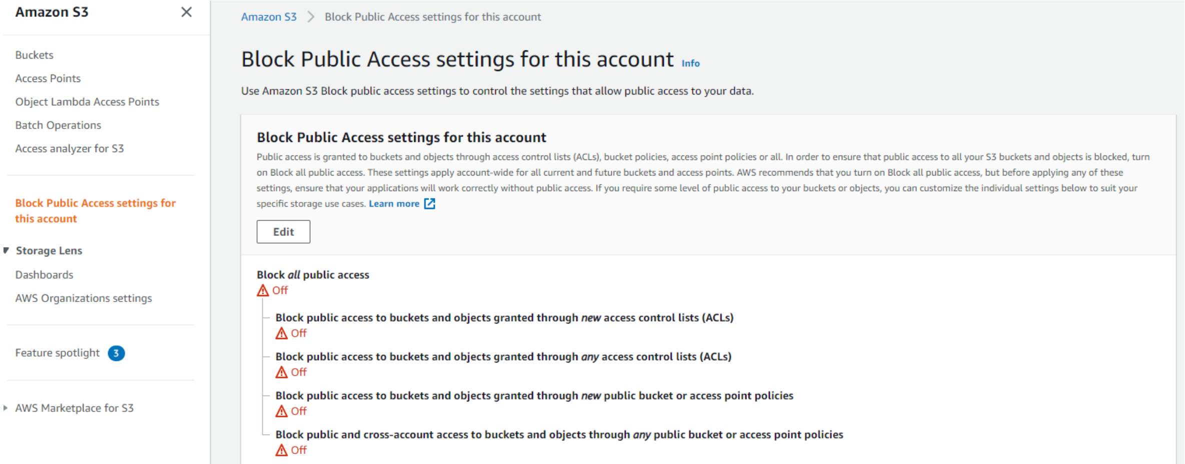
Destination	Succeeded	Failed
s3://demo-santosh-s3-bucket	0 files, 0 B (0%)	1 file, 4.4 KB (100.00%)

Files and folders (1 Total, 4.4 KB)

Name	Folder	Type	Size	Status	Error
amazon.jpg	-	image/jpeg	4.4 KB	Failed	Access Denied

Other type of security settings:

You can set public access settings at account level.



Amazon S3
Buckets
Access Points
Object Lambda Access Points
Batch Operations
Access analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
AWS Organizations settings
Feature spotlight 3
AWS Marketplace for S3
Amazon S3 > Block Public Access settings for this account

Block Public Access settings for this account Info

Use Amazon S3 Block public access settings to control the settings that allow public access to your data.

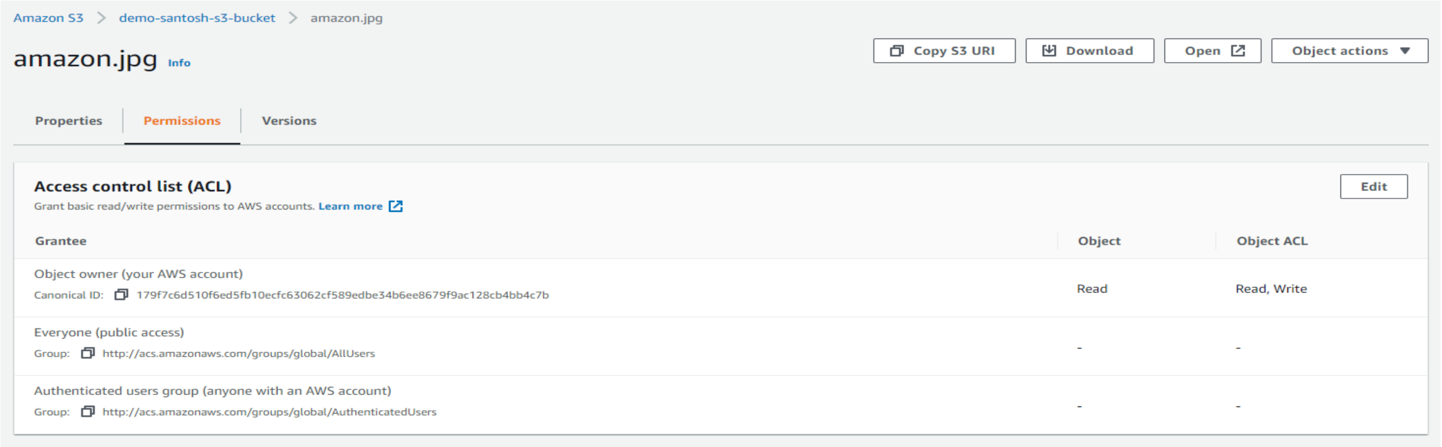
Block Public Access settings for this account

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 all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply account-wide for all current and future buckets and 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 your buckets or objects, you can customize the individual settings below to suit your specific storage use cases. Learn more

Block *all* public access
⚠ Off

- **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  ⚠ Off
- **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  ⚠ Off
- **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  ⚠ Off
- **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  ⚠ Off

You can set ACL at objects level



Access control list (ACL)

Grant basic read/write permissions to AWS accounts. Learn more

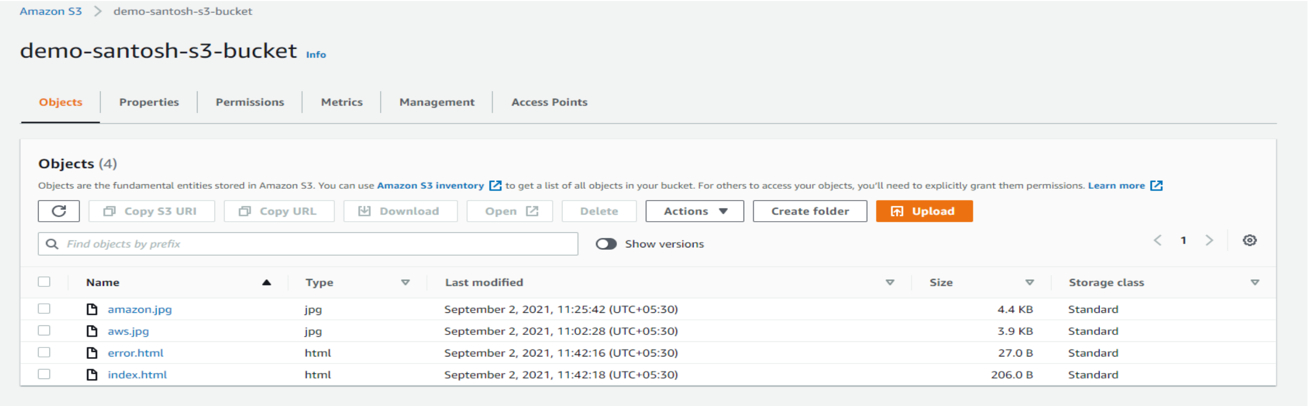
Grantee	Object	Object ACL
Object owner (your AWS account) Canonical ID: 179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	Read	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-

Amazon S3 website:

S3 can host static websites

Steps:

1. Upload html files



Amazon S3 > demo-santosh-s3-bucket

demo-santosh-s3-bucket Info

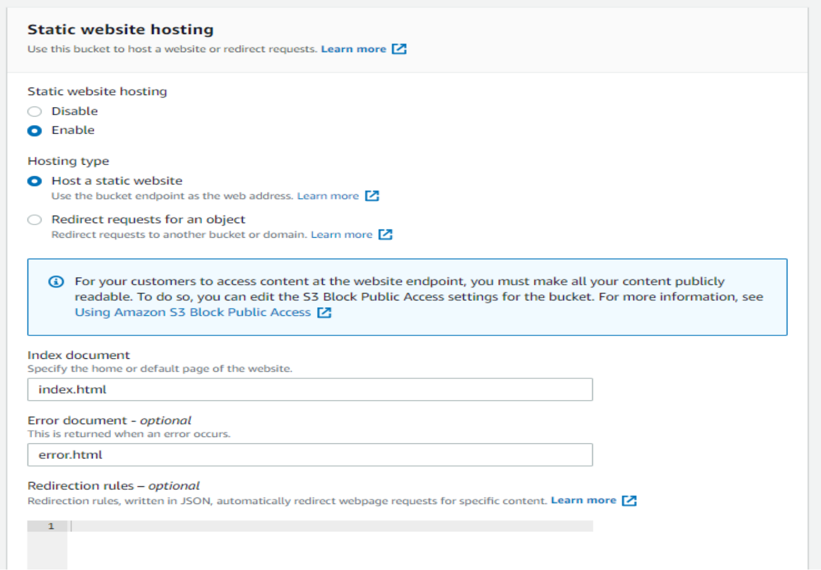
Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (4)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	amazon.jpg	jpg	September 2, 2021, 11:25:42 (UTC+05:30)	4.4 KB	Standard
□	aws.jpg	jpg	September 2, 2021, 11:02:28 (UTC+05:30)	3.9 KB	Standard
□	error.html	html	September 2, 2021, 11:42:16 (UTC+05:30)	27.0 B	Standard
□	index.html	html	September 2, 2021, 11:42:18 (UTC+05:30)	206.0 B	Standard

2. Enable 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.

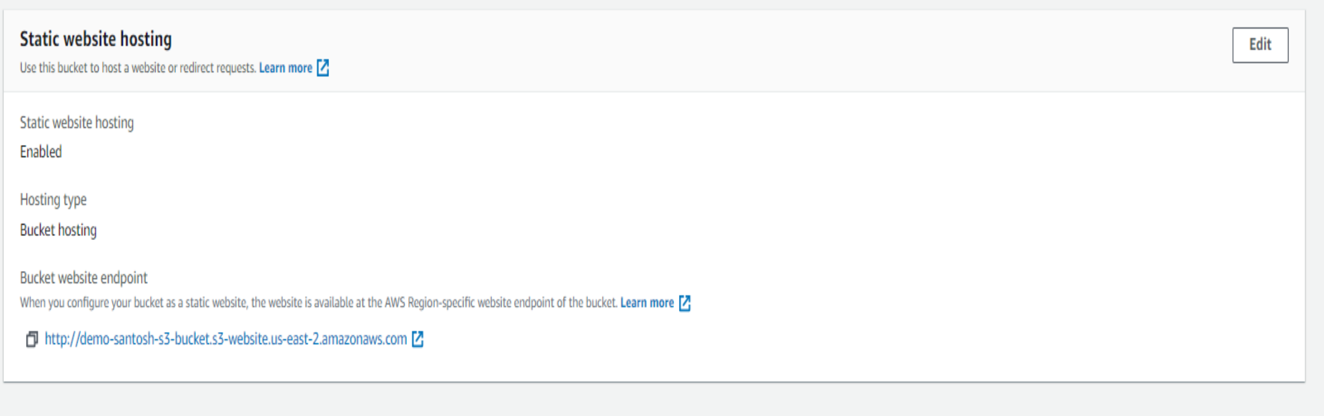
error.html

Redirection rules – *optional*

Redirection rules, written in JSON, automatically redirect webpage requests for specific content. Learn more

```
1
```

Save it and you will get bucket website endpoints



Static website hosting

Use this bucket to host a website or redirect requests. Learn more

Static website hosting

Enabled

Hosting type

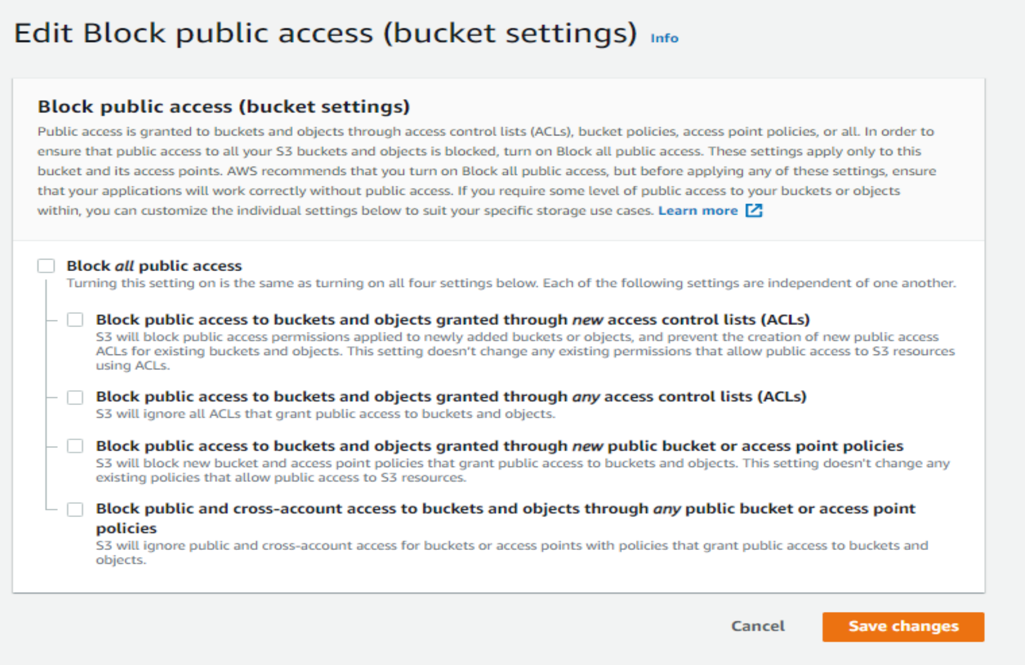
Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. Learn more

http://demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com

3. Disabled the block public access.



Edit Block public access (bucket settings) Info

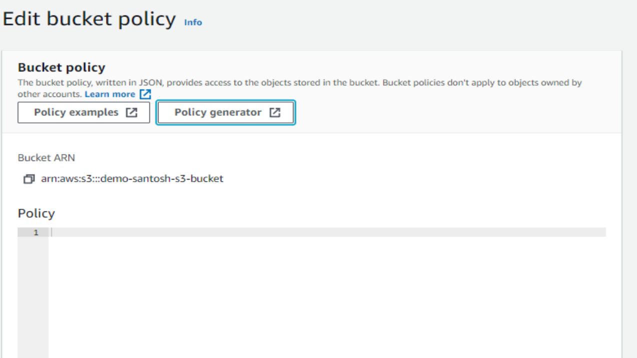
Block public access (bucket settings)

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 all your S3 buckets and 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 your buckets 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.

4. Write a bucket policy to allow public access.



Edit bucket policy Info

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

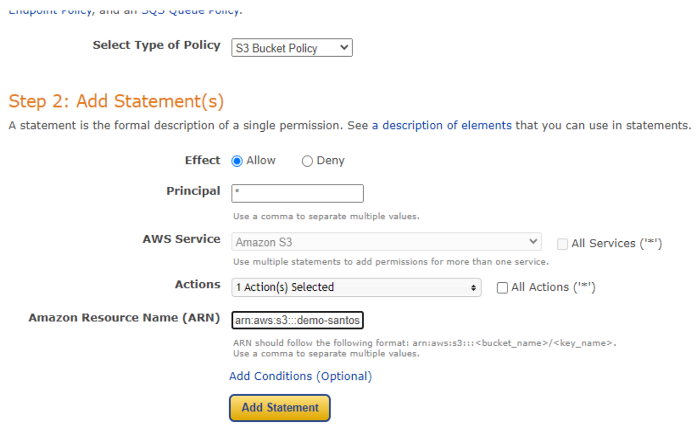
Bucket ARN

arn:aws:s3:::demo-santosh-s3-bucket

Policy

```
1
```

Generate policy



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 ('*')

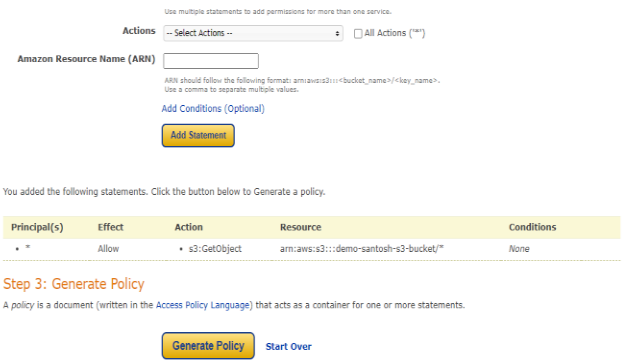
Use multiple statements to add permissions for more than one service.

Actions: 1 Action(s) Selected □ All Actions ('*')

Amazon Resource Name (ARN): arn:aws:s3:::demo-santos

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

(action = GetObject)



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

Actions -- Select Actions -- □ All Actions ('*')

Amazon Resource Name (ARN)

ARN should follow the following format: arn:aws:s3:::<bucket_name>/<key_name>.
Use a comma to separate multiple values.

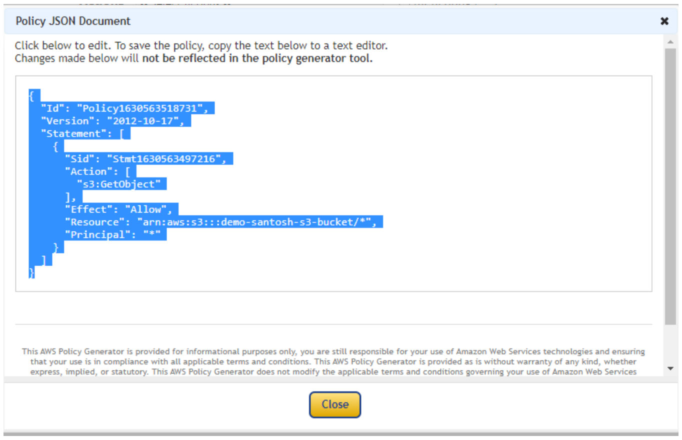
You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
• *	Allow	• s3:GetObject	arn:aws:s3:::demo-santosh-s3-bucket/*	*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.

Add statement and generate policy



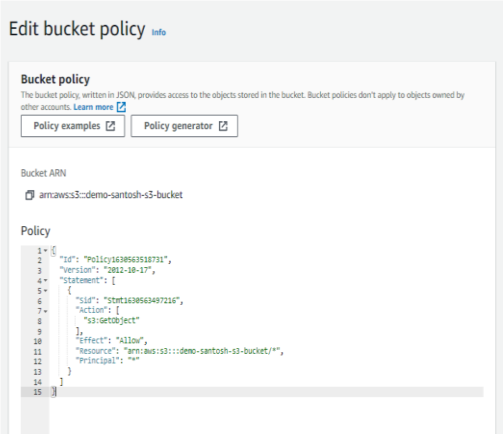
Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor.
Changes made below will **not be reflected in the policy generator tool.**

```
{
  "Id": "Policy1630563518731",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1630563497216",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::demo-santosh-s3-bucket/*",
      "Principal": "*"
    }
  ]
}
```

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

And paste in edit bucket policy and save it



Edit bucket policy Info

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

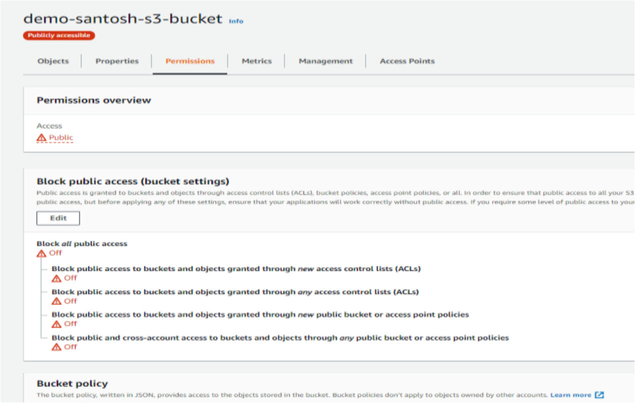
Bucket ARN

arn:aws:s3:::demo-santosh-s3-bucket

Policy

```
{
  "Id": "Policy1630563518731",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1630563497216",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::demo-santosh-s3-bucket/*",
      "Principal": "*"
    }
  ]
}
```

After saving the policy you will see access is public now.



demo-santosh-s3-bucket Info

Permissions overview

Access
⚠ Public

Block public access (bucket settings)

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 all your S3
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 your

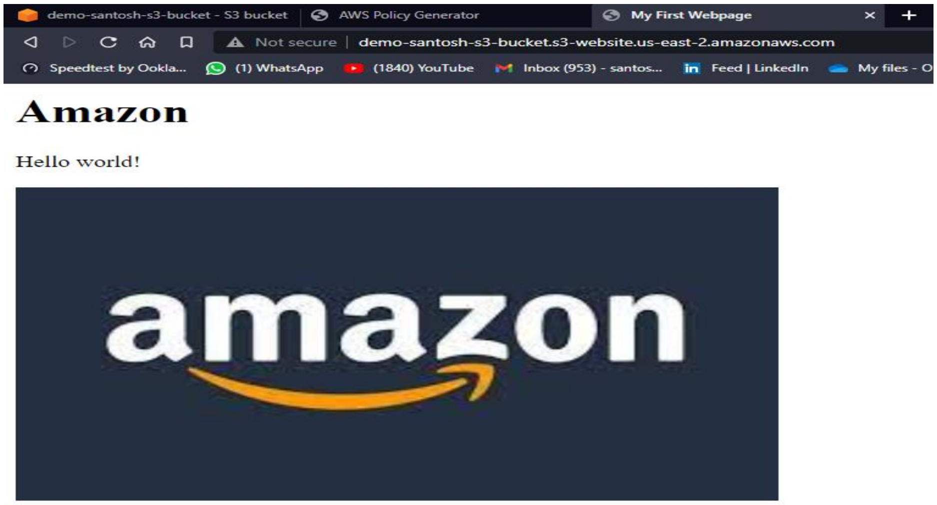
Block *all* public access
⚠ Off

- **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  ⚠ Off
- **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  ⚠ Off
- **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  ⚠ Off
- **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  ⚠ Off

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

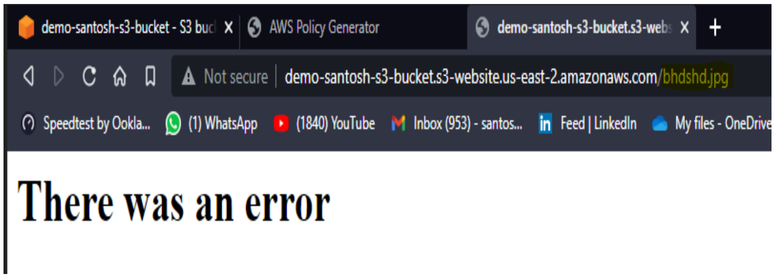
5. Go to the URLs



Not secure | demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com

Amazon

Hello world!



Not secure | demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com/bhdshd.jpg

There was an error

Amazon S3 CORS:

- CORS stands for Cross-Origin Resource Sharing. (Getting resources from a different region)
- Origin is a scheme (protocol), host(domain) and port.

  E.g. https://www.example.com (implied port is 443 for https and 80 for http).
- Web browser-based mechanism to allow requests to other origin while visiting the main origin.

  (Basically, means that when you visit a website. You can make request to other origins only if the other origin allow you to make these request.)
- Example:-

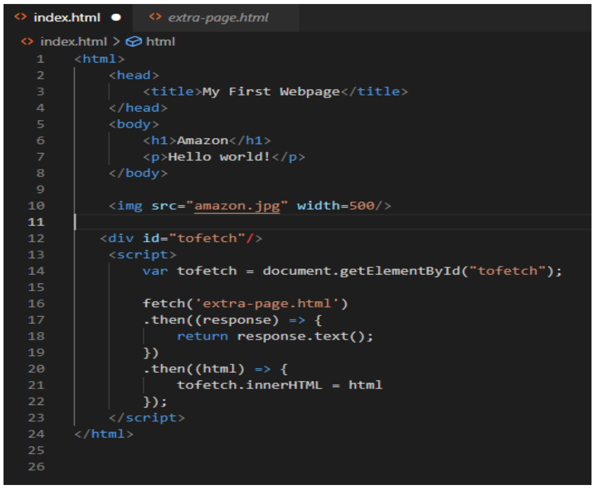
  https://www.example.com and https://other.example.com

  When you visit https://www.example.com then you asking your web browser to make a request to other origin website https://other.example.com this is called CROSS-ORIGIN request.

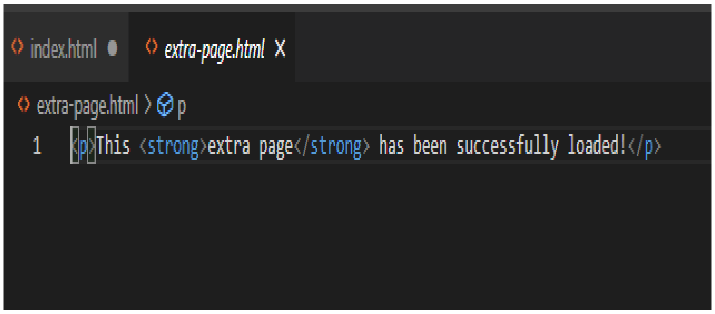
  If you have not correct headers the web browser will block it. The request won't be fulfilled unless the other origin allows for the requests using CORS headers. (e.g. Access-Control-Allow-Origin).
- If a client does a cross-origin request on our S3 bucket we need to enable the correct CORS headers.

Steps:

1) Upload these 2 files to S3 buckets

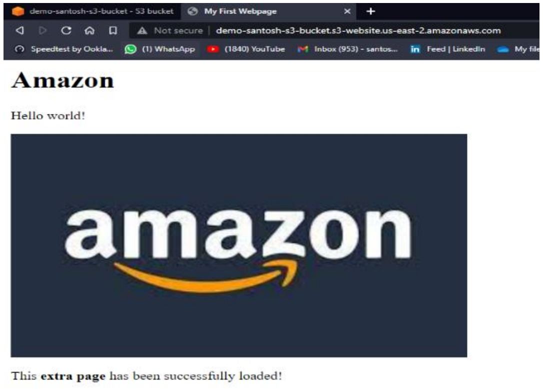


```html
<html>
    <head>
        <title>My First Webpage</title>
    </head>
    <body>
        <h1>Amazon</h1>
        <p>Hello world!</p>
    </body>

    <img src="amazon.jpg" width=500/>

   <div id="tofetch"/>
    <script>
        var tofetch = document.getElementById("tofetch");

        fetch('extra-page.html')
        .then((response) => {
            return response.text();
        })
        .then((html) => {
            tofetch.innerHTML = html
        });
    </script>
</html>
```



```html
<p>This <strong>extra page</strong> has been successfully loaded!</p>
```

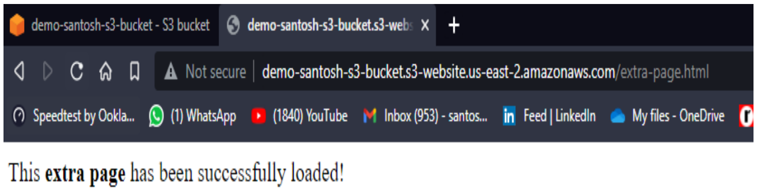
2) Open this html website in web browser



Amazon

Hello world!

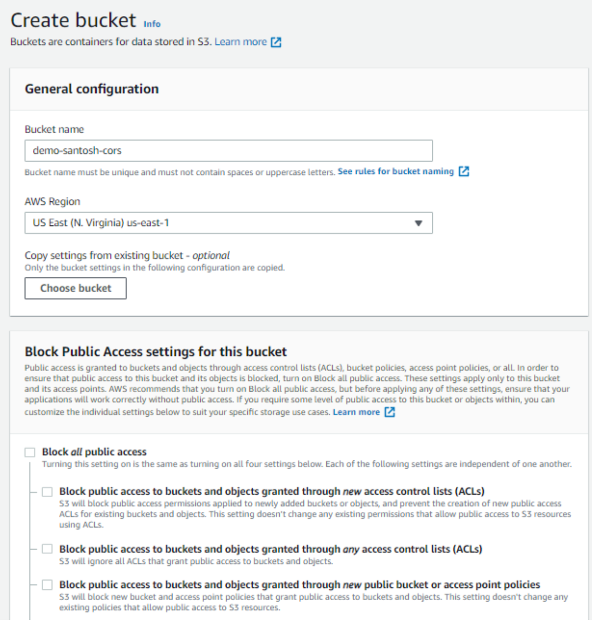
This **extra page** has been successfully loaded!


demo-santosh-s3-bucket - S3 bucket
demo-santosh-s3-bucket.s3-web
Not secure
demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com/extra-page.html
Speedtest by Ookla...
(1) WhatsApp
(1840) YouTube
Inbox (953) - santos...
Feed | LinkedIn
My files - OneDrive

This **extra page** has been successfully loaded!

These are on the same bucket website. But we want to do CORS for that we need another bucket on different region.

3) Create 2nd bucket on different region



Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name

demo-santosh-cors

Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming

AWS 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

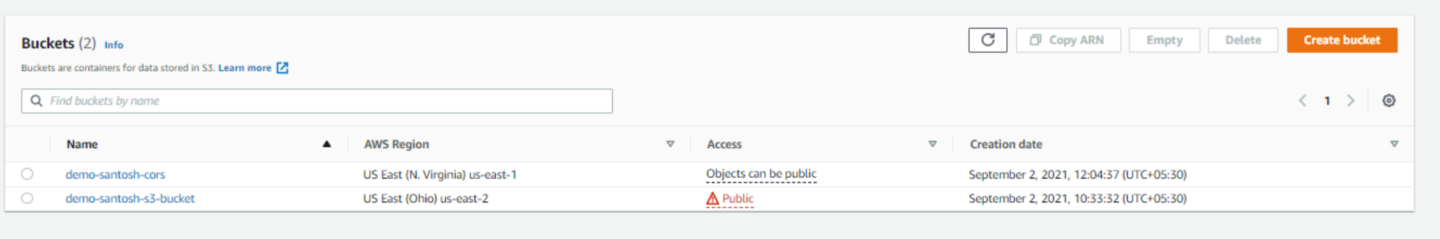
Block Public Access settings for this bucket

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.

Now we have 2 buckets



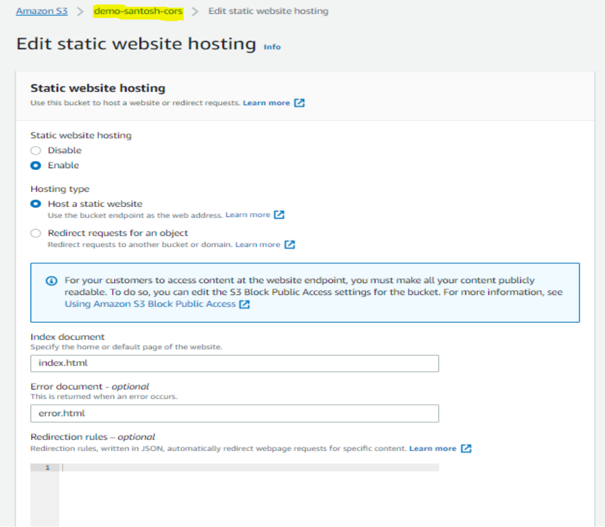
Buckets (2) Info

Buckets are containers for data stored in S3. Learn more

Find buckets by name

	Name	AWS Region	Access	Creation date
○	demo-santosh-cors	US East (N. Virginia) us-east-1	Objects can be public	September 2, 2021, 12:04:37 (UTC+05:30)
○	demo-santosh-s3-bucket	US East (Ohio) us-east-2	⚠ Public	September 2, 2021, 10:33:32 (UTC+05:30)

4) Enable static website hosting on new bucket



Amazon S3 > demo-santosh-cors > Edit static website hosting

Edit static website hosting Info

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.

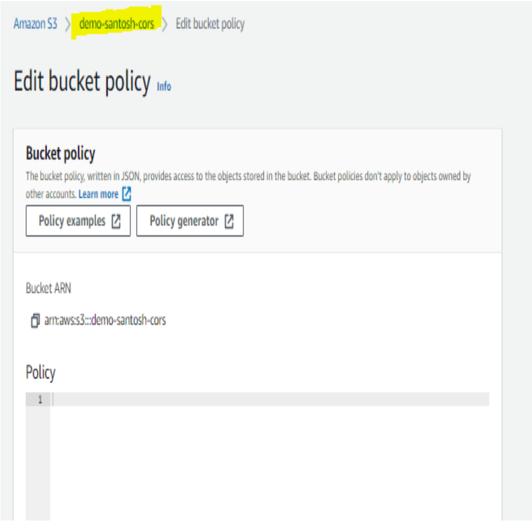
index.html

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

5) Create bucket policy



Amazon S3 > demo-santosh-cors > Edit bucket policy

Edit bucket policy Info

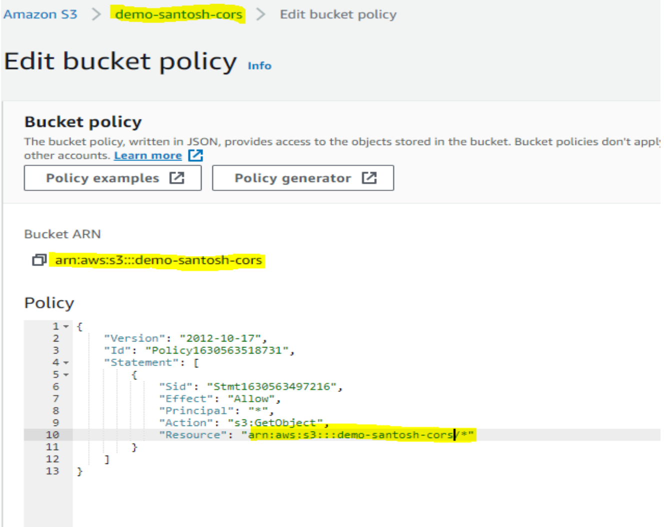
Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

Bucket ARN

arn:aws:s3:::demo-santosh-cors

Policy



Amazon S3 > demo-santosh-cors > Edit bucket policy

Edit bucket policy Info

Bucket policy

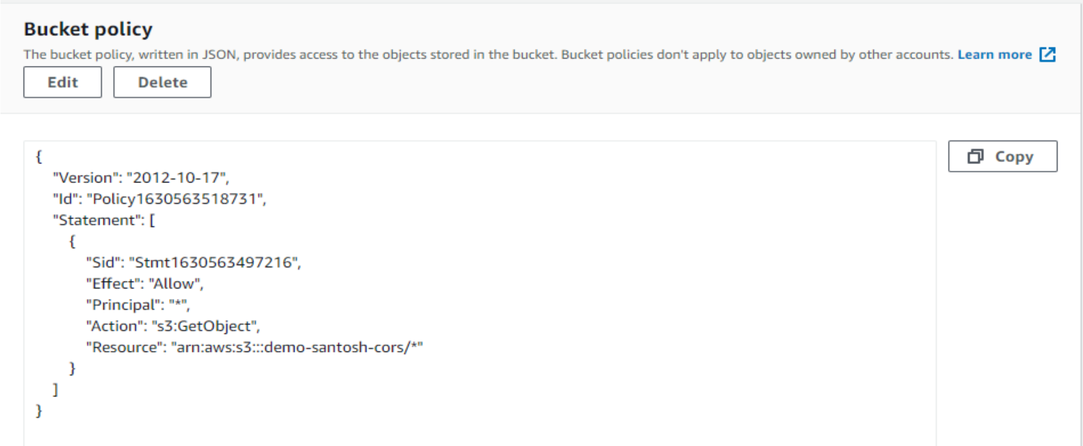
The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't appl
other accounts. Learn more

Bucket ARN

arn:aws:s3:::demo-santosh-cors

Policy

```
{
    "Version": "2012-10-17",
    "Id": "Policy1630563518731",
    "Statement": [
        {
            "Sid": "Stmt1630563497216",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::demo-santosh-cors/*"
        }
    ]
}
```

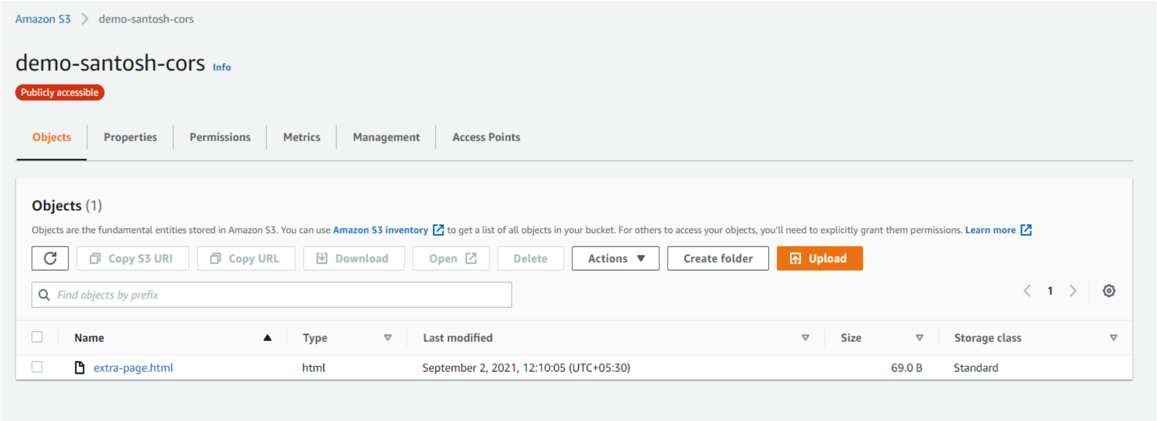


Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

```
{
  "Version": "2012-10-17",
  "Id": "Policy1630563518731",
  "Statement": [
    {
      "Sid": "Stmt1630563497216",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::demo-santosh-cors/*"
    }
  ]
}
```

6) Upload the html file



Amazon S3 > demo-santosh-cors

demo-santosh-cors Info

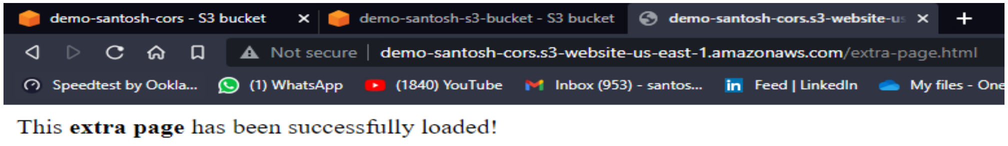
Publicly accessible

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

	Name	Type	Last modified	Size	Storage class
□	extra-page.html	html	September 2, 2021, 12:10:05 (UTC+05:30)	69.0 B	Standard

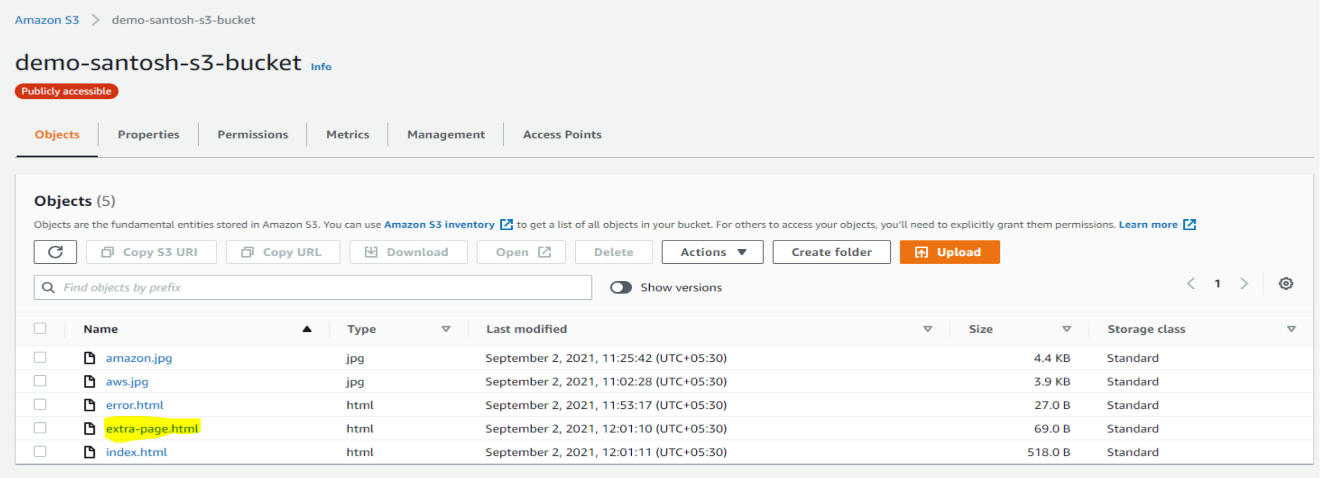
7) Get access this bucket object



demo-santosh-cors.s3-website-us-east-1.amazonaws.com/extra-page.html

This **extra page** has been successfully loaded!

8) Remove this file from 1st main bucket

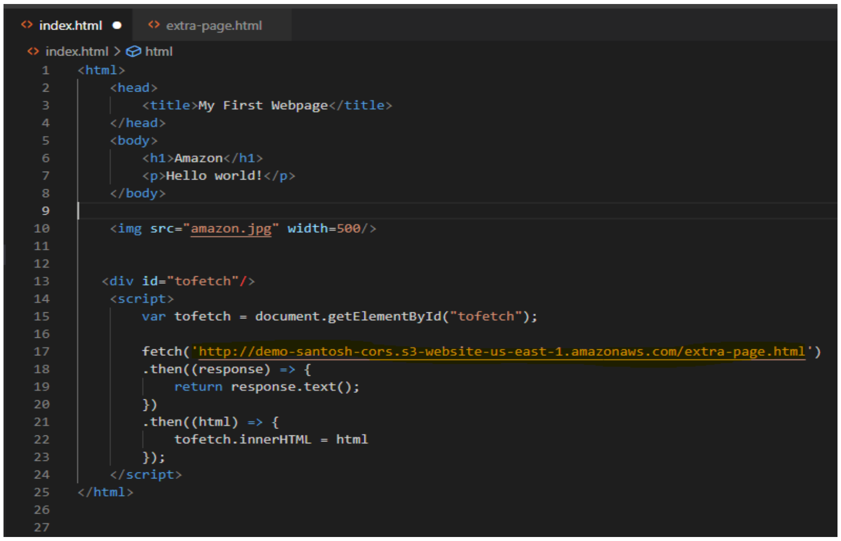


Objects (5)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	amazon.jpg	jpg	September 2, 2021, 11:25:42 (UTC+05:30)	4.4 KB	Standard
□	aws.jpg	jpg	September 2, 2021, 11:02:28 (UTC+05:30)	3.9 KB	Standard
□	error.html	html	September 2, 2021, 11:53:17 (UTC+05:30)	27.0 B	Standard
□	extra-page.html	html	September 2, 2021, 12:01:10 (UTC+05:30)	69.0 B	Standard
□	index.html	html	September 2, 2021, 12:01:11 (UTC+05:30)	518.0 B	Standard

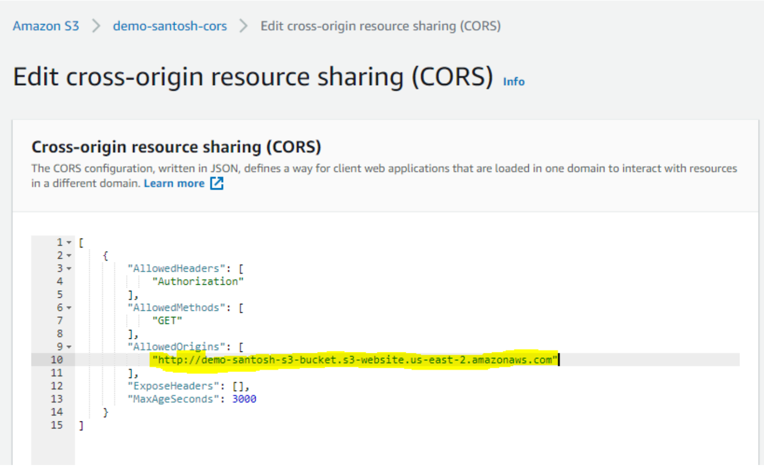
9) Edit 1st index.html and copy the 2nd bucket website URL and paste in this fetch and reupload to 1st bucket.



```
<html>
    <head>
        <title>My First Webpage</title>
    </head>
    <body>
        <h1>Amazon</h1>
        <p>Hello world!</p>
    </body>

    <img src="amazon.jpg" width=500/>


   <div id="tofetch"/>
    <script>
        var tofetch = document.getElementById("tofetch");

        fetch('http://demo-santosh-cors.s3-website-us-east-1.amazonaws.com/extra-page.html')
        .then((response) => {
            return response.text();
        })
        .then((html) => {
            tofetch.innerHTML = html
        });
    </script>
</html>
```

10) Edit the 2nd bucket CORS to allow 1st bucket to make the request.

Go to 2nd bucket CORS


Amazon S3 > demo-santosh-cors > Edit cross-origin resource sharing (CORS)
Edit cross-origin resource sharing (CORS) Info

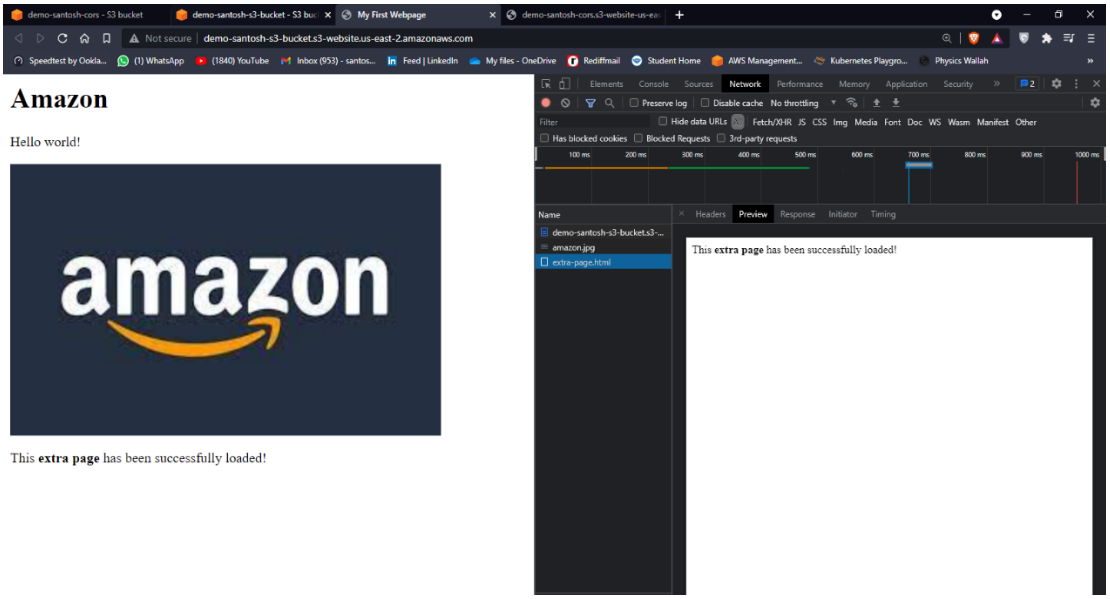
Cross-origin resource sharing (CORS)

The CORS configuration, written in JSON, defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. Learn more

```
[
    {
        "AllowedHeaders": [
            "Authorization"
        ],
        "AllowedMethods": [
            "GET"
        ],
        "AllowedOrigins": [
            "http://demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com"
        ],
        "ExposeHeaders": [],
        "MaxAgeSeconds": 3000
    }
]
```

(Put the first bucket URL with http://... Without slash at the end)

11) Go to 1st bucket web page. Now 1st web page gets the access of 2nd bucket (different origin web page access to 1st bucket web page.)



Address bar: demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com

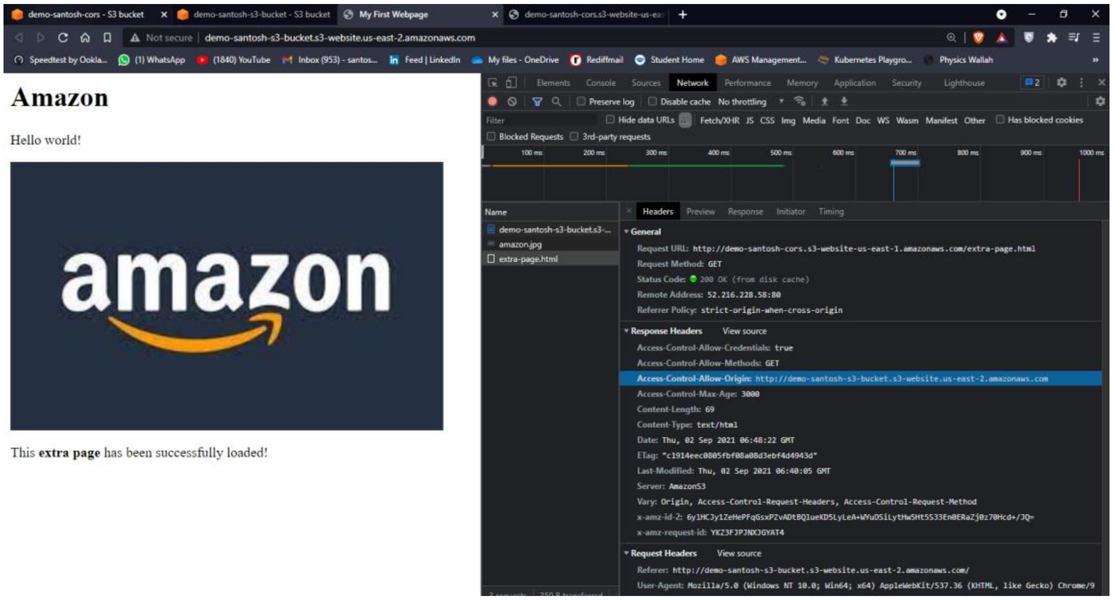
Amazon

Hello world!

This **extra page** has been successfully loaded!

Preview (extra-page.html): This **extra page** has been successfully loaded!

You can see headers in this (Access-Control-Allow-Origin).



Amazon

Hello world!

This **extra page** has been successfully loaded!

```
▾ General
  Request URL: http://demo-santosh-cors.s3-website-us-east-1.amazonaws.com/extra-page.html
  Request Method: GET
  Status Code: 200 OK (from disk cache)
  Remote Address: 52.216.228.58:80
  Referrer Policy: strict-origin-when-cross-origin
▾ Response Headers    View source
  Access-Control-Allow-Credentials: true
  Access-Control-Allow-Methods: GET
  Access-Control-Allow-Origin: http://demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com
  Access-Control-Max-Age: 3000
  Content-Length: 69
  Content-Type: text/html
  Date: Thu, 02 Sep 2021 06:48:22 GMT
  ETag: "c1914eec0805fbf08a08d3ebf4d4943d"
  Last-Modified: Thu, 02 Sep 2021 06:40:05 GMT
  Server: AmazonS3
  Vary: Origin, Access-Control-Request-Headers, Access-Control-Request-Method
  x-amz-id-2: 6ylHCJy1ZeHePFqGsxPZvADtBQlueKD5LyLeA+WYuDSiLytHw5Ht5S33En0ERaZj0z70Hcd+/JQ=
  x-amz-request-id: YKZ3FJPJNXJGYAT4
▾ Request Headers    View source
  Referer: http://demo-santosh-s3-bucket.s3-website.us-east-2.amazonaws.com/
  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/9
```

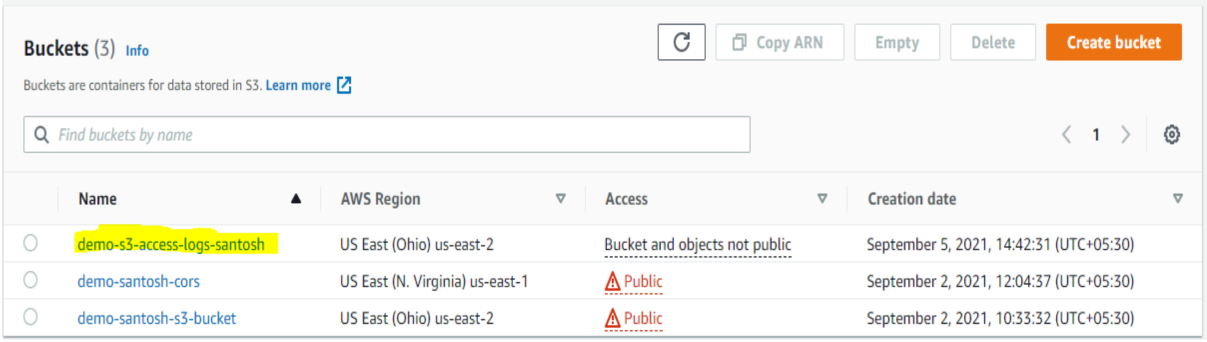
Note:

If website 1 needs to access resources from website 2 through a web browser, then website 2 needs to have CORS setting to allow a request from 1st website. Otherwise, web browser will block it.

S3 Access Logs

You want to log all the access into your S3 buckets. So that means any request that is done to S3 from any account you want to be logged into another S3 bucket. So, you can analyze it later using analysis tool (Athena).

Create s3 bucket



Buckets (3) Info

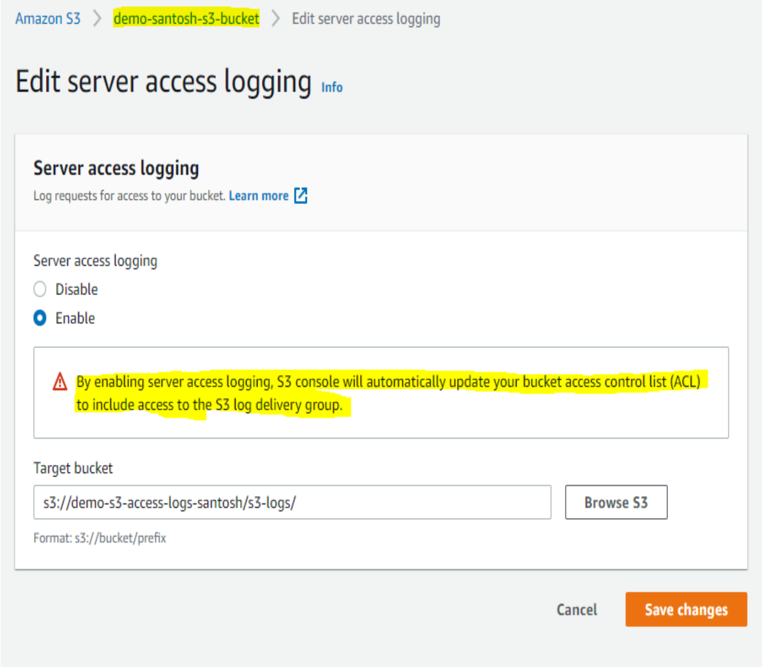
Buckets are containers for data stored in S3. Learn more

Find buckets by name

	Name	AWS Region	Access	Creation date
○	demo-s3-access-logs-santosh	US East (Ohio) us-east-2	Bucket and objects not public	September 5, 2021, 14:42:31 (UTC+05:30)
○	demo-santosh-cors	US East (N. Virginia) us-east-1	⚠ Public	September 2, 2021, 12:04:37 (UTC+05:30)
○	demo-santosh-s3-bucket	US East (Ohio) us-east-2	⚠ Public	September 2, 2021, 10:33:32 (UTC+05:30)

To get the access logs of bucket (demo-santosh-s3-bucket) into demo-s3-access-logs-santosh this bucket

- Go to other bucket(demo-santosh-s3-bucket) and enable the server access logging
- Select the Target bucket (demo-s3-access-logs-santosh) and set the path.



Amazon S3 > demo-santosh-s3-bucket > Edit server access logging

Edit server access logging Info

Server access logging

Log requests for access to your bucket. Learn more

Server access logging

○ Disable

● Enable

⚠ By enabling server access logging, S3 console will automatically update your bucket access control list (ACL) to include access to the S3 log delivery group.

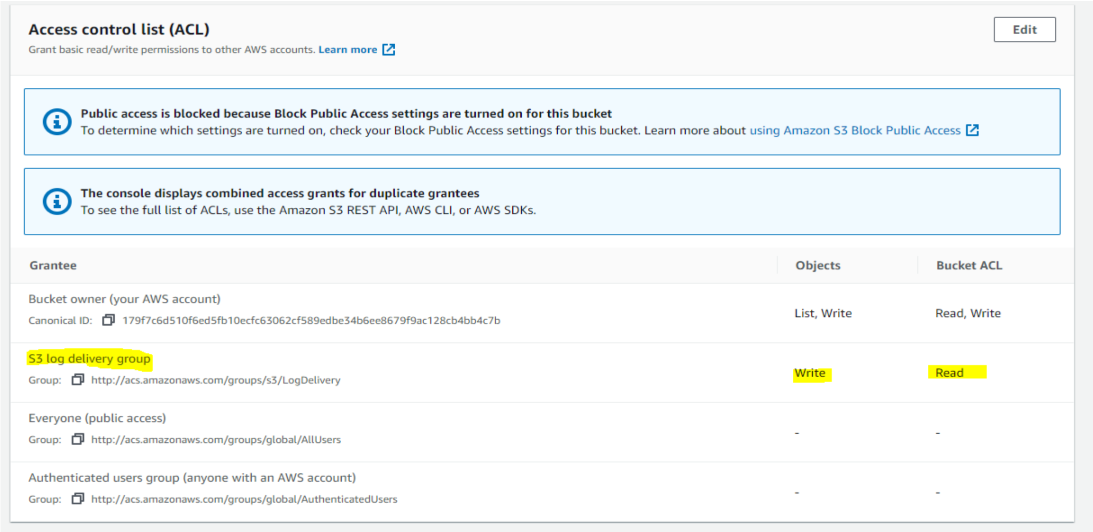
Target bucket

s3://demo-s3-access-logs-santosh/s3-logs/

Format: s3://bucket/prefix

By enabling server access logging, S3 console will automatically update your bucket access control list (ACL) to include access to the S3 log delivery group.

demo-s3-access-logs-santosh



Access control list (ACL)

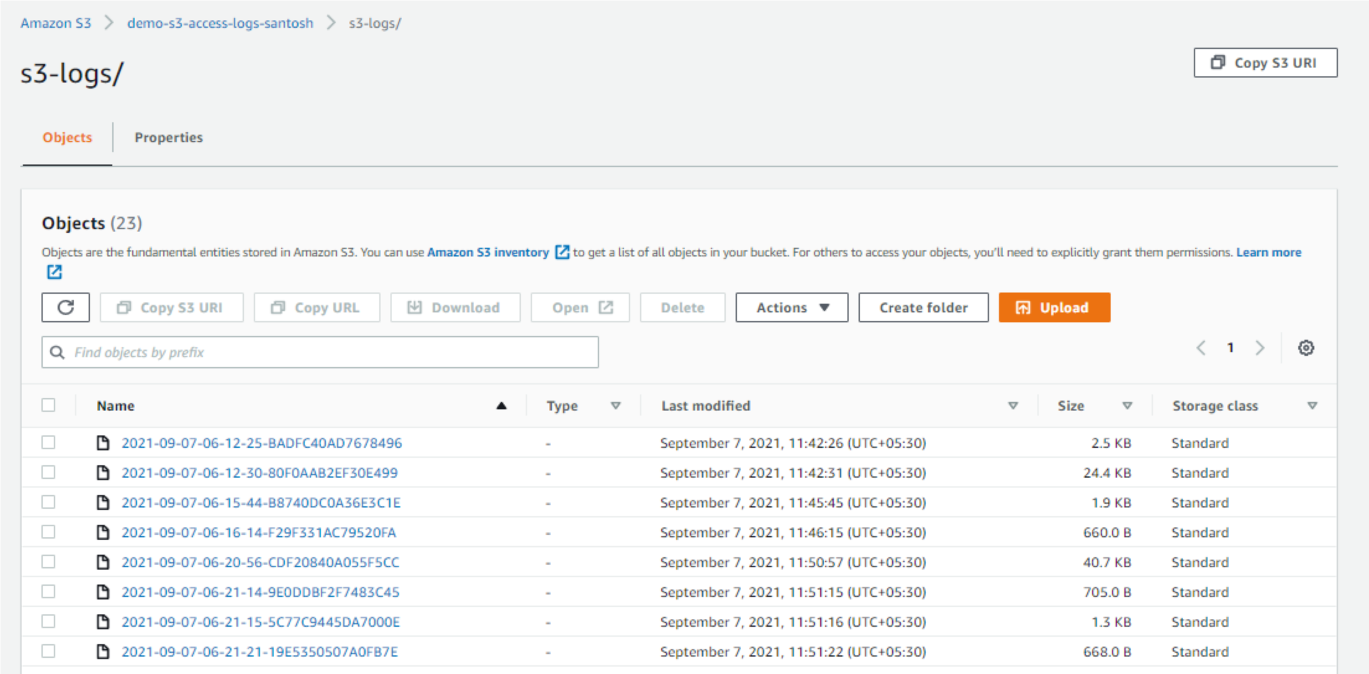
Grant basic read/write permissions to other AWS accounts. Learn more

Public access is blocked because Block Public Access settings are turned on for this bucket
To determine which settings are turned on, check your Block Public Access settings for this bucket. Learn more about using Amazon S3 Block Public Access

The console displays combined access grants for duplicate grantees
To see the full list of ACLs, use the Amazon S3 REST API, AWS CLI, or AWS SDKs.

Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: 179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	List, Write	Read, Write
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	Write	Read
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	-	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-

Now open any file from bucket(demo-santosh-s3-bucket) an it will generate some traffic on to my bucket and this log goes to logs bucket. It takes few hours to update in your logs bucket.



s3-logs/

Objects (23)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	2021-09-07-06-12-25-BADFC40AD7678496	-	September 7, 2021, 11:42:26 (UTC+05:30)	2.5 KB	Standard
□	2021-09-07-06-12-30-80F0AAB2EF30E499	-	September 7, 2021, 11:42:31 (UTC+05:30)	24.4 KB	Standard
□	2021-09-07-06-15-44-B8740DC0A36E3C1E	-	September 7, 2021, 11:45:45 (UTC+05:30)	1.9 KB	Standard
□	2021-09-07-06-16-14-F29F331AC79520FA	-	September 7, 2021, 11:46:15 (UTC+05:30)	660.0 B	Standard
□	2021-09-07-06-20-56-CDF20840A055F5CC	-	September 7, 2021, 11:50:57 (UTC+05:30)	40.7 KB	Standard
□	2021-09-07-06-21-14-9E0DDBF2F7483C45	-	September 7, 2021, 11:51:15 (UTC+05:30)	705.0 B	Standard
□	2021-09-07-06-21-15-5C77C9445DA7000E	-	September 7, 2021, 11:51:16 (UTC+05:30)	1.3 KB	Standard
□	2021-09-07-06-21-21-19E5350507A0FB7E	-	September 7, 2021, 11:51:22 (UTC+05:30)	668.0 B	Standard

S3 Replication

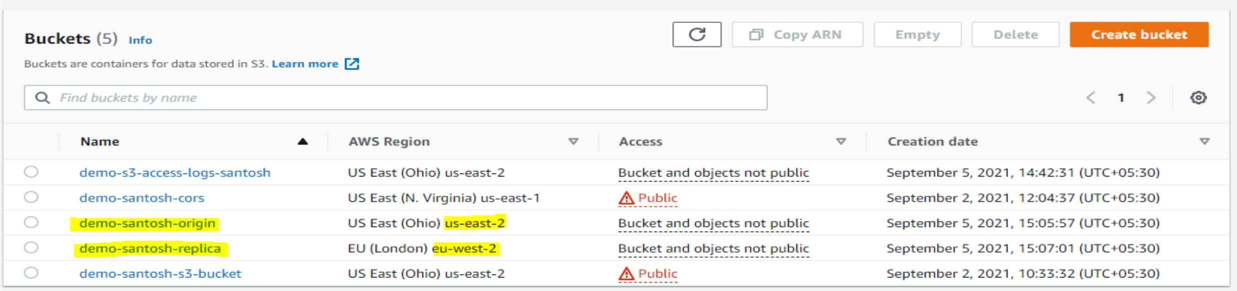
There are 2 replication types: -

1. CRR (Cross region replication)
2. SRR (Same region replication)

Cross region replication

Steps:

1. Create 2 buckets in different region and enable the versioning on both.

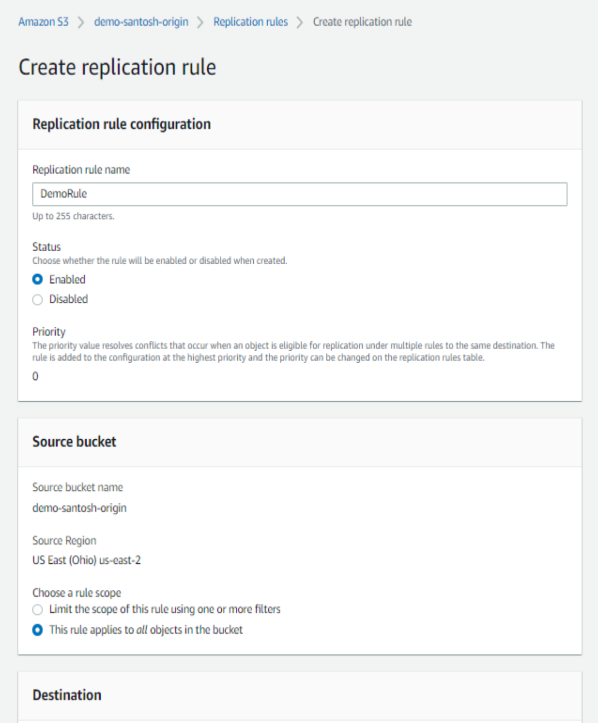


Buckets (5) Info

Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
○	demo-s3-access-logs-santosh	US East (Ohio) us-east-2	Bucket and objects not public	September 5, 2021, 14:42:31 (UTC+05:30)
○	demo-santosh-cors	US East (N. Virginia) us-east-1	Public	September 2, 2021, 12:04:37 (UTC+05:30)
○	demo-santosh-origin	US East (Ohio) us-east-2	Bucket and objects not public	September 5, 2021, 15:05:57 (UTC+05:30)
○	demo-santosh-replica	EU (London) eu-west-2	Bucket and objects not public	September 5, 2021, 15:07:01 (UTC+05:30)
○	demo-santosh-s3-bucket	US East (Ohio) us-east-2	Public	September 2, 2021, 10:33:32 (UTC+05:30)

2. Create replication rule on demo-santosh-origin bucket.


Amazon S3 > demo-santosh-origin > Replication rules > Create replication rule

Create replication rule

Replication rule configuration

Replication rule name: DemoRule

Up to 255 characters.

Status

Choose whether the rule will be enabled or disabled when created.

● Enabled

○ Disabled

Priority

The priority value resolves conflicts that occur when an object is eligible for replication under multiple rules to the same destination. The rule is added to the configuration at the highest priority and the priority can be changed on the replication rules table.

0

Source bucket

Source bucket name: demo-santosh-origin

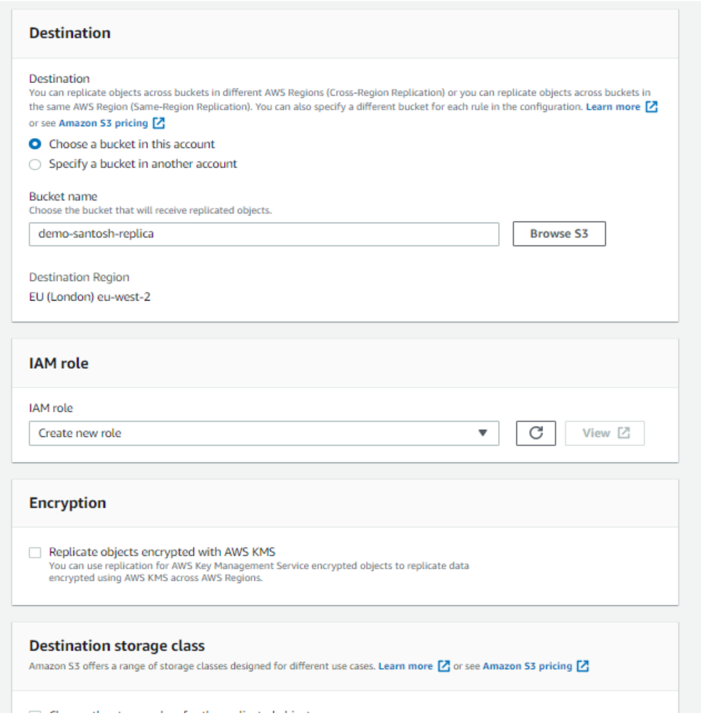
Source Region: US East (Ohio) us-east-2

Choose a rule scope

○ Limit the scope of this rule using one or more filters

● This rule applies to *all* objects in the bucket

Destination



Destination

Destination

You can replicate objects across buckets in different AWS Regions (Cross-Region Replication) or you can replicate objects across buckets in the same AWS Region (Same-Region Replication). You can also specify a different bucket for each rule in the configuration. Learn more or see Amazon S3 pricing

● Choose a bucket in this account

○ Specify a bucket in another account

Bucket name

Choose the bucket that will receive replicated objects.

demo-santosh-replica

Destination Region: EU (London) eu-west-2

IAM role

IAM role: Create new role

Encryption

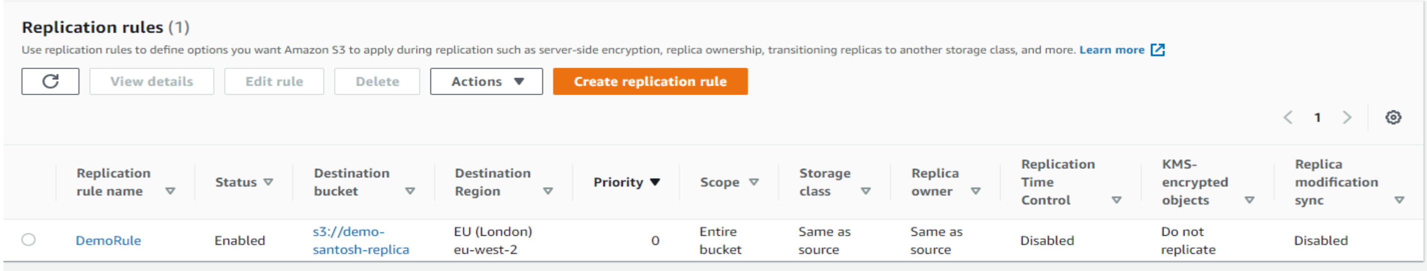
□ Replicate objects encrypted with AWS KMS

You can use replication for AWS Key Management Service encrypted objects to replicate data encrypted using AWS KMS across AWS Regions.

Destination storage class

Amazon S3 offers a range of storage classes designed for different use cases. Learn more or see Amazon S3 pricing

Save it.

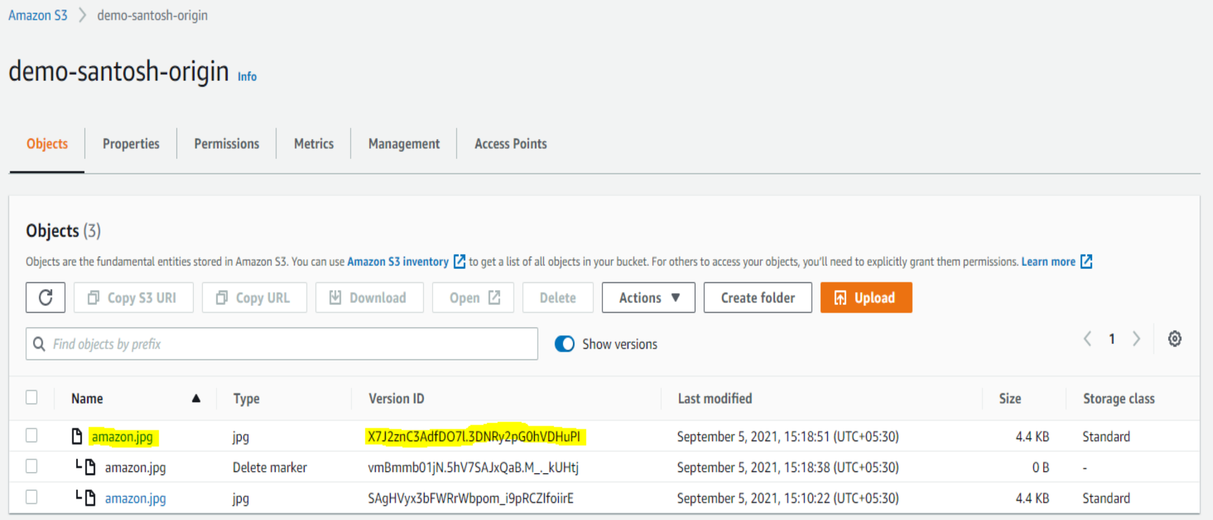


Replication rules (1)

Use replication rules to define options you want Amazon S3 to apply during replication such as server-side encryption, replica ownership, transitioning replicas to another storage class, and more. Learn more

	Replication rule name	Status	Destination bucket	Destination Region	Priority	Scope	Storage class	Replica owner	Replication Time Control	KMS-encrypted objects	Replica modification sync
○	DemoRule	Enabled	s3://demo-santosh-replica	EU (London) eu-west-2	0	Entire bucket	Same as source	Same as source	Disabled	Do not replicate	Disabled

3. Upload some objects on origin bucket



Amazon S3 > demo-santosh-origin

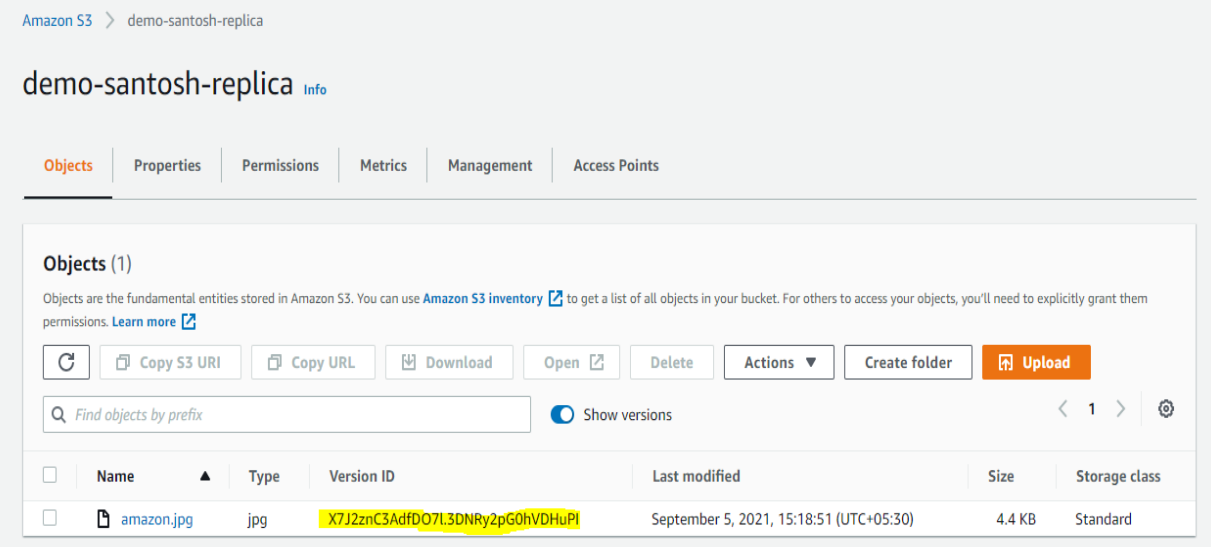
demo-santosh-origin Info

Objects (3)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Version ID	Last modified	Size	Storage class
□	amazon.jpg	jpg	X7J2znC3AdfDO7l.3DNRy2pG0hVDHuPI	September 5, 2021, 15:18:51 (UTC+05:30)	4.4 KB	Standard
□	amazon.jpg	Delete marker	vmBmmb01jN.5hV7SAJxQaB.M_._kUHtj	September 5, 2021, 15:18:38 (UTC+05:30)	0 B	-
□	amazon.jpg	jpg	SAgHVyx3bFWRrWbpom_i9pRCZIfoiirE	September 5, 2021, 15:10:22 (UTC+05:30)	4.4 KB	Standard

And now go to replica bucket you will see same object here.



Amazon S3 > demo-santosh-replica

demo-santosh-replica Info

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Version ID	Last modified	Size	Storage class
□	amazon.jpg	jpg	X7J2znC3AdfDO7l.3DNRy2pG0hVDHuPI	September 5, 2021, 15:18:51 (UTC+05:30)	4.4 KB	Standard

You will see the object version ID is also same. The object is replicated including version ID.

Note: After activating replication rule, only the new objects are replicated.

4. Deletion

- After deleting the object, the delete marker is not going to replicated by default. If you want to then there are settings in replication rule.
- After deletion with a version ID are note replicated to avoid malicious deletes.

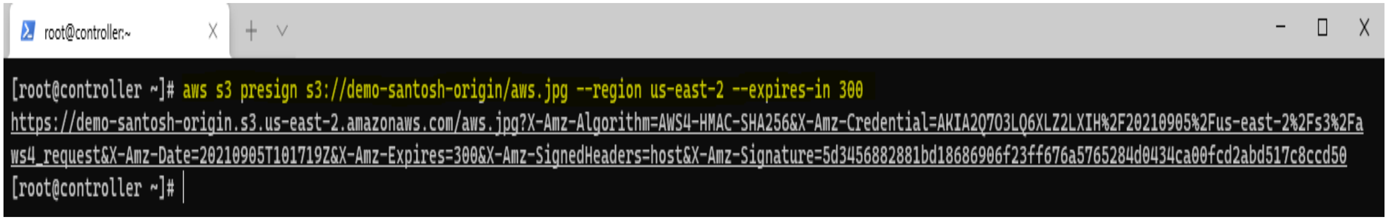
S3 Pre-signed URLs

- User given a pre-signed URL inherit the permissions of the person who generated the URL for GET/PUT.
- Valid for default of 3600 Seconds. Can change timeout with expires in [TIME-BY-SECOND] argument.

Creating pre-signed URL for object.

Steps:

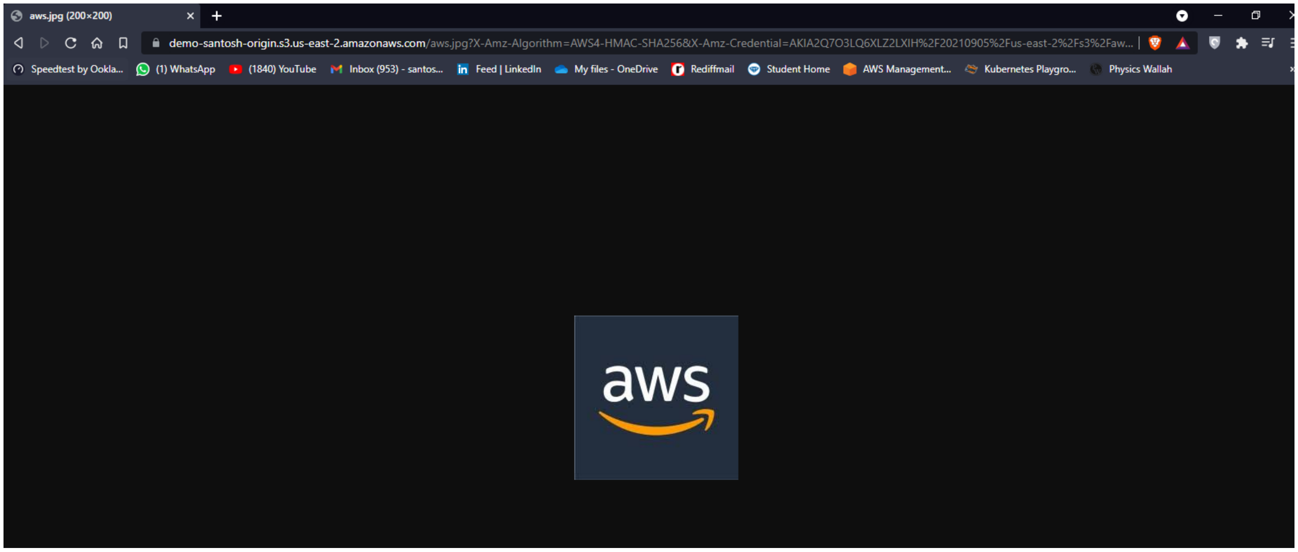
1. Create pre-signed URL from CLI



```
[root@controller ~]# aws s3 presign s3://demo-santosh-origin/aws.jpg --region us-east-2 --expires-in 300
https://demo-santosh-origin.s3.us-east-2.amazonaws.com/aws.jpg?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIA2Q7O3LQ6XLZ2LXIH%2F20210905%2Fus-east-2%2Fs3%2Fa
ws4_request&X-Amz-Date=20210905T101719Z&X-Amz-Expires=300&X-Amz-SignedHeaders=host&X-Amz-Signature=5d3456882881bd18686906f23ff676a5765284d0434ca00fcd2abd517c8ccd50
[root@controller ~]#
```

This URL is expired after 300 seconds.

2. Copy that URL and access on browser.



```
demo-santosh-origin.s3.us-east-2.amazonaws.com/aws.jpg?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIA2Q7O3LQ6XLZ2LXIH%2F20210905%2Fus-east-2%2Fs3%2Faw...
```

Now you have access of the object using pre-signed URL.

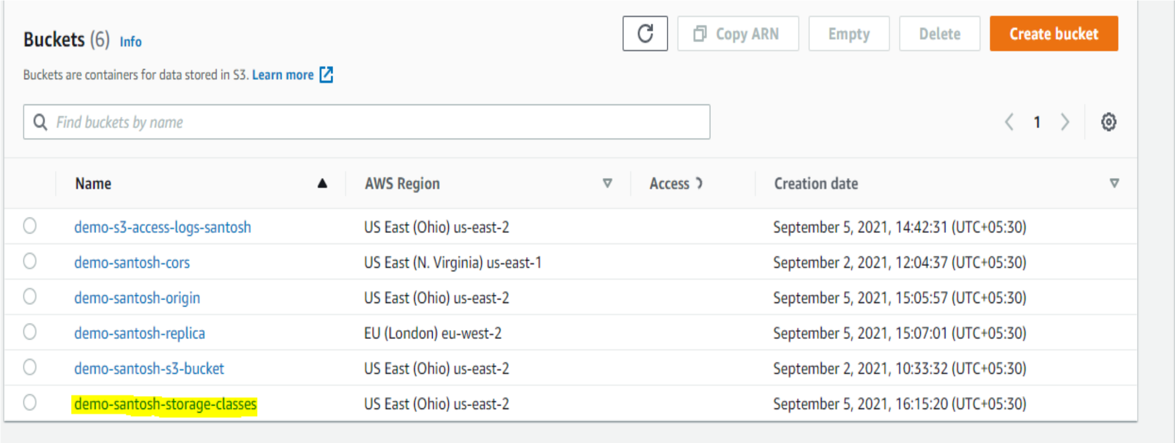
S3 Storage Classes

Types of storage classes: -

a. Amazon S3 standard – General purpose
b. Amazon S3 standard – Infrequent Access (IA)
c. Amazon S3 one zone – Infrequent Access (IA)
d. Amazon S3 Intelligent Tiering
e. Amazon Glacier
f. Amazon Glacier Deep Archive.

Steps:

1. Create a bucket.

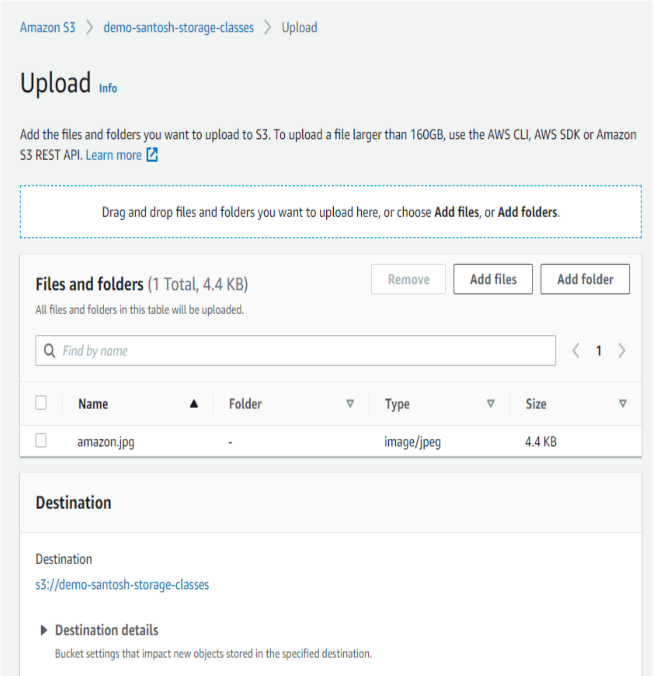


Buckets (6) Info

Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
○	demo-s3-access-logs-santosh	US East (Ohio) us-east-2		September 5, 2021, 14:42:31 (UTC+05:30)
○	demo-santosh-cors	US East (N. Virginia) us-east-1		September 2, 2021, 12:04:37 (UTC+05:30)
○	demo-santosh-origin	US East (Ohio) us-east-2		September 5, 2021, 15:05:57 (UTC+05:30)
○	demo-santosh-replica	EU (London) eu-west-2		September 5, 2021, 15:07:01 (UTC+05:30)
○	demo-santosh-s3-bucket	US East (Ohio) us-east-2		September 2, 2021, 10:33:32 (UTC+05:30)
○	demo-santosh-storage-classes	US East (Ohio) us-east-2		September 5, 2021, 16:15:20 (UTC+05:30)

2. While uploading the object on bucket choose the storage class.



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 folders**.

Files and folders (1 Total, 4.4 KB)

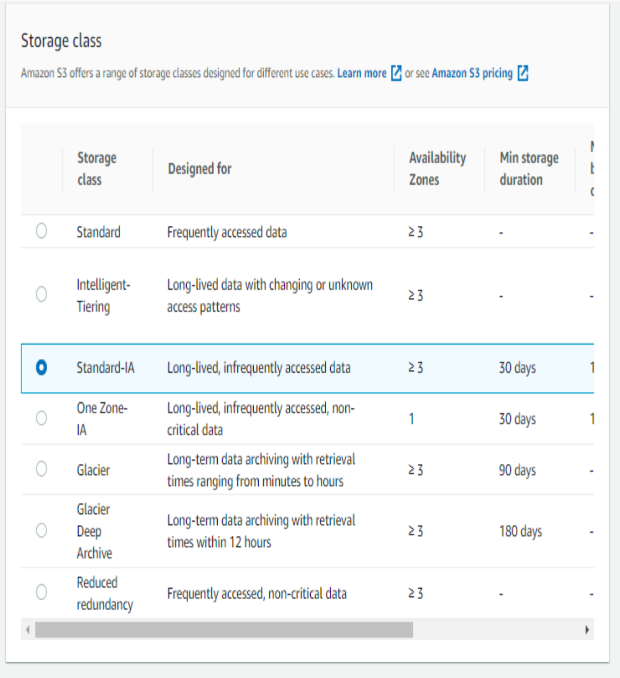
All files and folders in this table will be uploaded.

□	Name	Folder	Type	Size
□	amazon.jpg	-	image/jpeg	4.4 KB

Destination

Destination
s3://demo-santosh-storage-classes

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

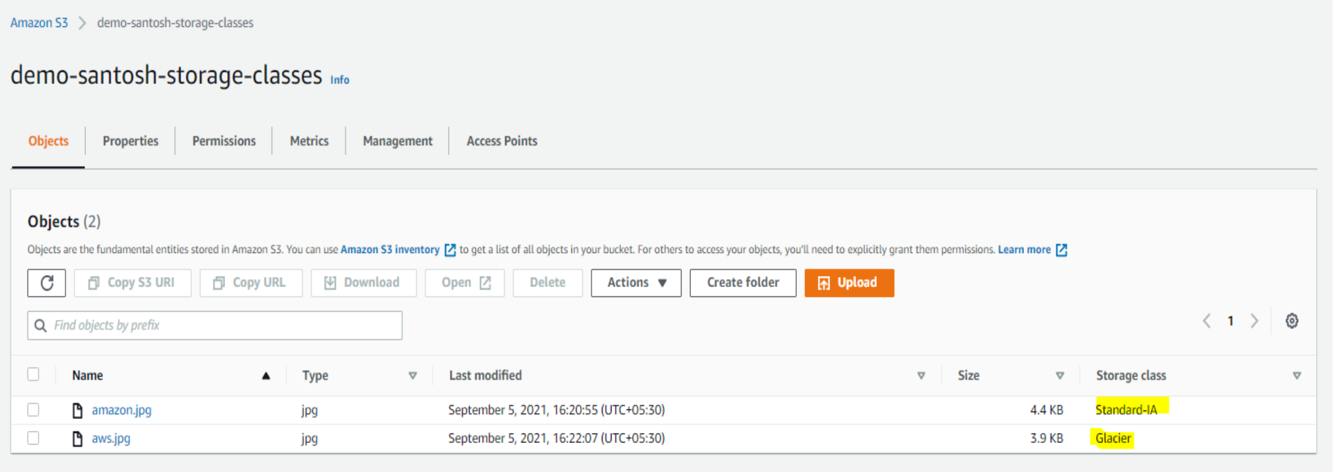


Storage class

Amazon S3 offers a range of storage classes designed for different use cases. Learn more or see Amazon S3 pricing

	Storage class	Designed for	Availability Zones	Min storage duration	
○	Standard	Frequently accessed data	≥ 3	-	-
○	Intelligent-Tiering	Long-lived data with changing or unknown access patterns	≥ 3	-	-
●	Standard-IA	Long-lived, infrequently accessed data	≥ 3	30 days	1
○	One Zone-IA	Long-lived, infrequently accessed, non-critical data	1	30 days	1
○	Glacier	Long-term data archiving with retrieval times ranging from minutes to hours	≥ 3	90 days	-
○	Glacier Deep Archive	Long-term data archiving with retrieval times within 12 hours	≥ 3	180 days	-
○	Reduced redundancy	Frequently accessed, non-critical data	≥ 3	-	-

Upload another object and select glacier storage class.



Amazon S3 > demo-santosh-storage-classes

demo-santosh-storage-classes Info

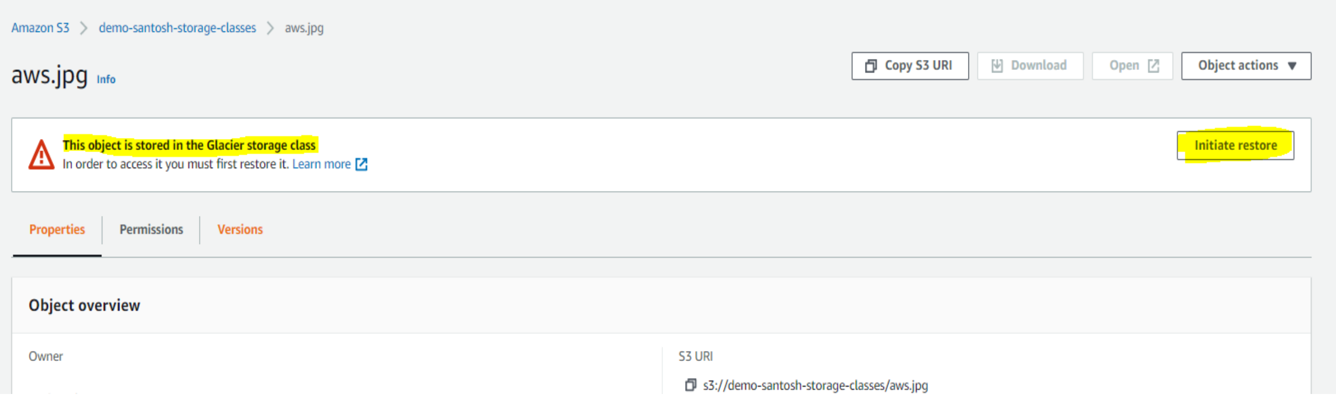
Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	amazon.jpg	jpg	September 5, 2021, 16:20:55 (UTC+05:30)	4.4 KB	Standard-IA
□	aws.jpg	jpg	September 5, 2021, 16:22:07 (UTC+05:30)	3.9 KB	Glacier

After uploading the objects, you can edit the storage class

3. After uploading the object with storage class try to access it.



Amazon S3 > demo-santosh-storage-classes > aws.jpg

aws.jpg Info

This object is stored in the Glacier storage class
In order to access it you must first restore it. Learn more

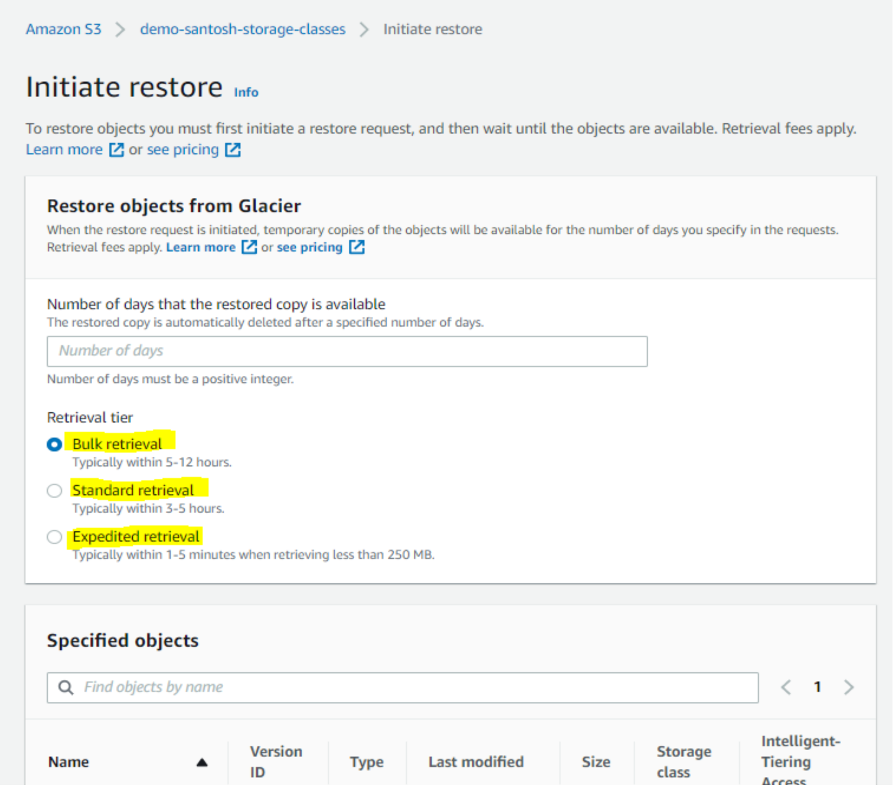
Object overview

Owner

S3 URI
s3://demo-santosh-storage-classes/aws.jpg

You will see that the glacier class object is not accessible until we restore it.

First, we need to initiate restore it.



Amazon S3 > demo-santosh-storage-classes > Initiate restore

Initiate restore Info

To restore objects you must first initiate a restore request, and then wait until the objects are available. Retrieval fees apply. Learn more or see pricing

Restore objects from Glacier

When the restore request is initiated, temporary copies of the objects will be available for the number of days you specify in the requests. Retrieval fees apply. Learn more or see pricing

Number of days that the restored copy is available
The restored copy is automatically deleted after a specified number of days.

Number of days

Number of days must be a positive integer.

Retrieval tier

● Bulk retrieval
Typically within 5-12 hours.

○ Standard retrieval
Typically within 3-5 hours.

○ Expedited retrieval
Typically within 1-5 minutes when retrieving less than 250 MB.

Specified objects

Find objects by name

Name ▲	Version ID	Type	Last modified	Size	Storage class	Intelligent-Tiering Access

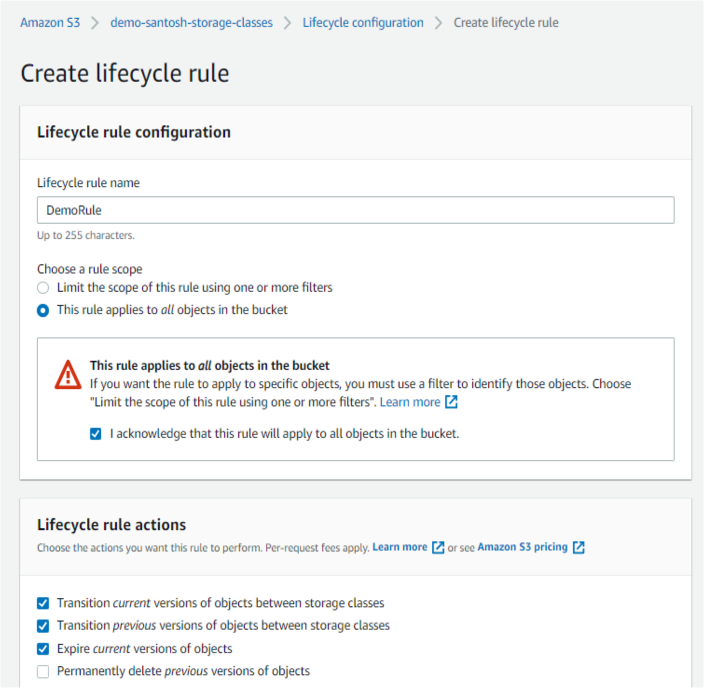
It takes too much time and if we choose Expedited retrieval its more expensive.

S3 Lifecycle Rules

You can define transaction actions when you want to transition your objects from one storage to another.

Examples: -

1. Move objects to Standard IA class 60 days after creation.
2. Move to glacier for archiving after 6 months later, etc.



Amazon S3 > demo-santosh-storage-classes > Lifecycle configuration > Create lifecycle rule

Create lifecycle rule

Lifecycle rule configuration

Lifecycle rule name

DemoRule

Up to 255 characters.

Choose a rule scope

○ Limit the scope of this rule using one or more filters

● This rule applies to *all* objects in the bucket

This rule applies to *all* objects in the bucket
If you want the rule to apply to specific objects, you must use a filter to identify those objects. Choose "Limit the scope of this rule using one or more filters". Learn more

☑ I acknowledge that this rule will apply to all objects in the bucket.

Lifecycle rule actions

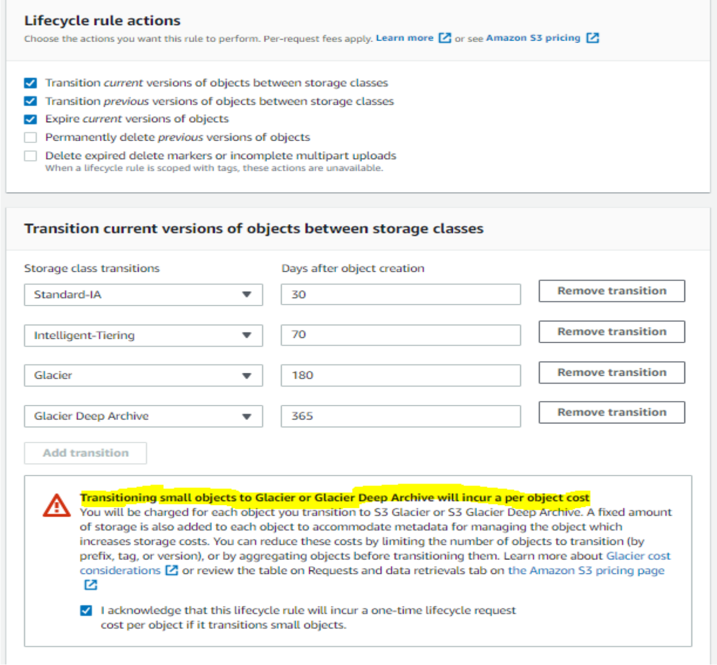
Choose the actions you want this rule to perform. Per-request fees apply. Learn more or see Amazon S3 pricing

☑ Transition *current* versions of objects between storage classes

☑ Transition *previous* versions of objects between storage classes

☑ Expire *current* versions of objects

☐ Permanently delete *previous* versions of objects



Lifecycle rule actions

Choose the actions you want this rule to perform. Per-request fees apply. Learn more or see Amazon S3 pricing

☑ Transition *current* versions of objects between storage classes

☑ Transition *previous* versions of objects between storage classes

☑ Expire *current* versions of objects

☐ Permanently delete *previous* versions of objects

☐ Delete expired delete markers or incomplete multipart uploads
When a lifecycle rule is scoped with tags, these actions are unavailable.

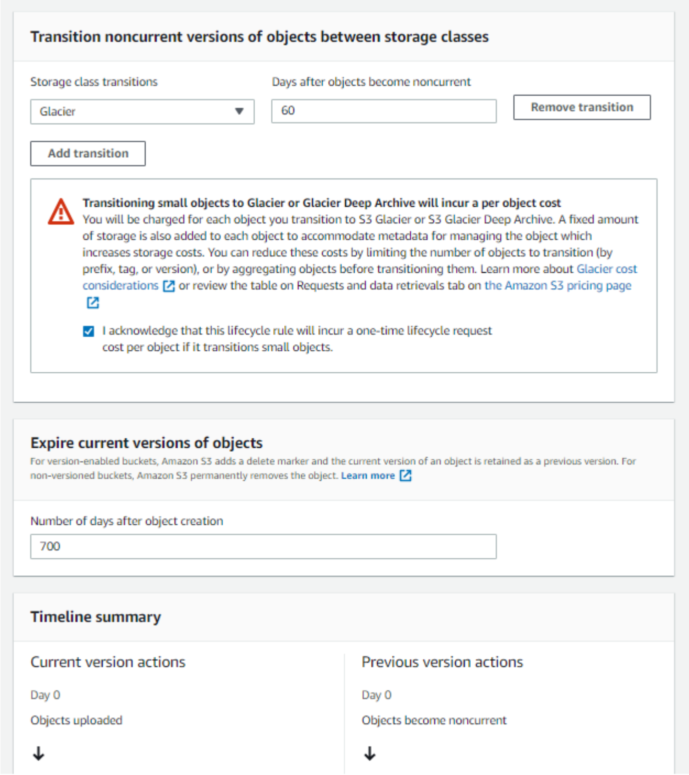
Transition current versions of objects between storage classes

Storage class transitions	Days after object creation	
Standard-IA	30	Remove transition
Intelligent-Tiering	70	Remove transition
Glacier	180	Remove transition
Glacier Deep Archive	365	Remove transition

Add transition

Transitioning small objects to Glacier or Glacier Deep Archive will incur a per object cost
You will be charged for each object you transition to S3 Glacier or S3 Glacier Deep Archive. A fixed amount of storage is also added to each object to accommodate metadata for managing the object which increases storage costs. You can reduce these costs by limiting the number of objects to transition (by prefix, tag, or version), or by aggregating objects before transitioning them. Learn more about Glacier cost considerations or review the table on Requests and data retrievals tab on the Amazon S3 pricing page

☑ I acknowledge that this lifecycle rule will incur a one-time lifecycle request cost per object if it transitions small objects.



Transition noncurrent versions of objects between storage classes

Storage class transitions	Days after objects become noncurrent	
Glacier	60	Remove transition

Add transition

Transitioning small objects to Glacier or Glacier Deep Archive will incur a per object cost
You will be charged for each object you transition to S3 Glacier or S3 Glacier Deep Archive. A fixed amount of storage is also added to each object to accommodate metadata for managing the object which increases storage costs. You can reduce these costs by limiting the number of objects to transition (by prefix, tag, or version), or by aggregating objects before transitioning them. Learn more about Glacier cost considerations or review the table on Requests and data retrievals tab on the Amazon S3 pricing page

☑ I acknowledge that this lifecycle rule will incur a one-time lifecycle request cost per object if it transitions small objects.

Expire current versions of objects

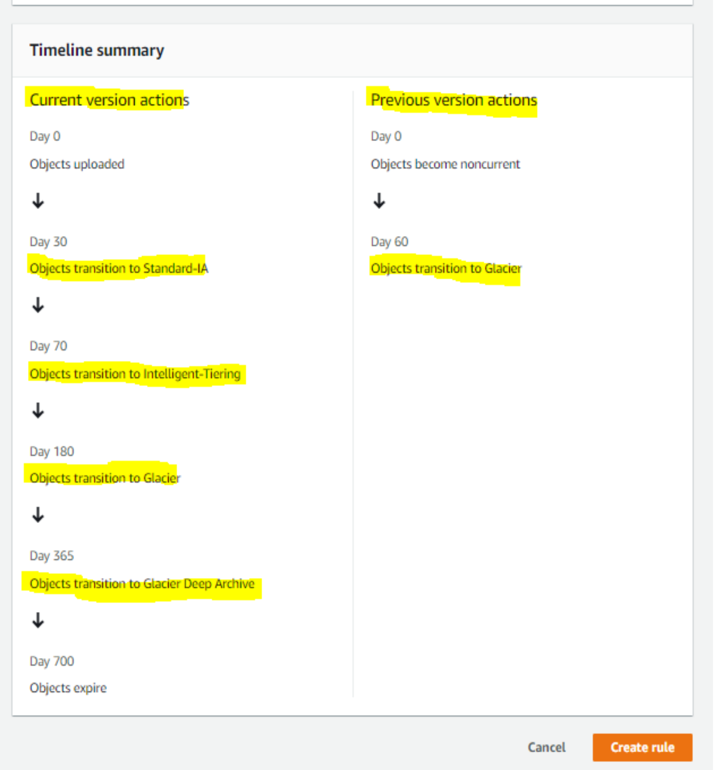
For version-enabled buckets, Amazon S3 adds a delete marker and the current version of an object is retained as a previous version. For non-versioned buckets, Amazon S3 permanently removes the object. Learn more

Number of days after object creation

700

Timeline summary

Current version actions	Previous version actions
Day 0 Objects uploaded ↓	Day 0 Objects become noncurrent ↓



Timeline summary

Current version actions	Previous version actions
Day 0 Objects uploaded ↓	Day 0 Objects become noncurrent ↓
Day 30 Objects transition to Standard-IA ↓	Day 60 Objects transition to Glacier
Day 70 Objects transition to Intelligent-Tiering ↓	
Day 180 Objects transition to Glacier ↓	
Day 365 Objects transition to Glacier Deep Archive ↓	
Day 700 Objects expire	

Athena

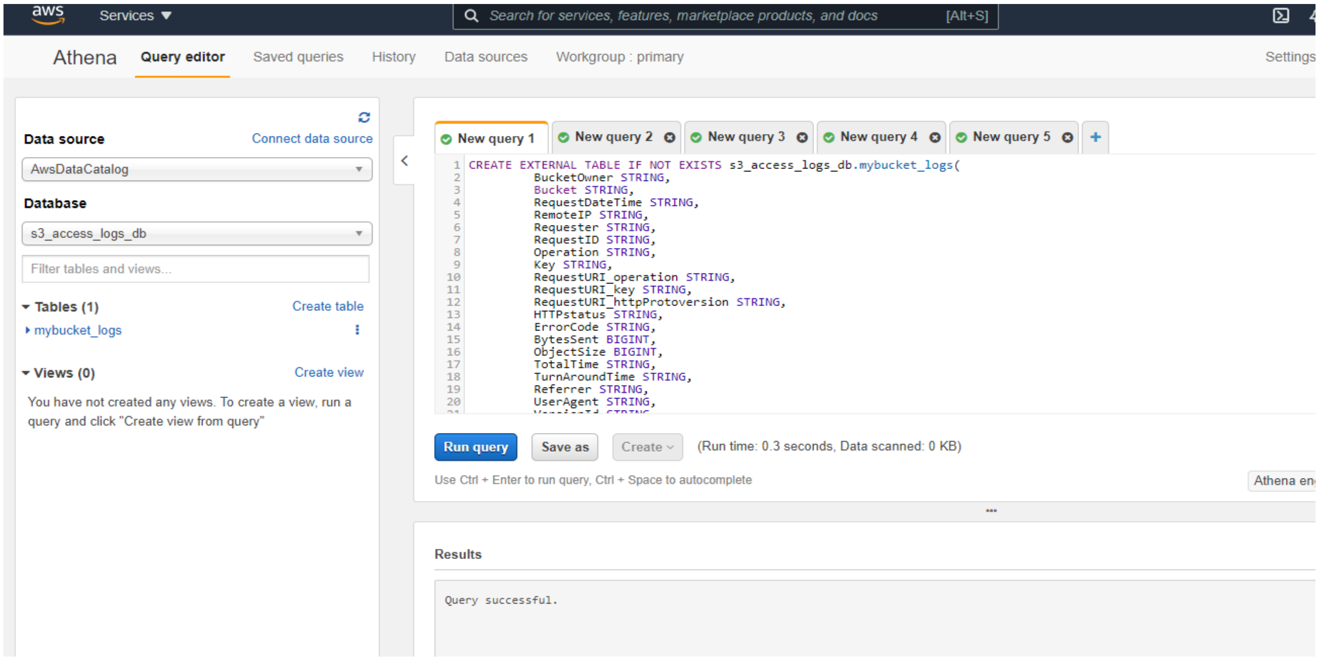
- Serverless service to perform analytics directly against S3 files. Usually, you have to load your files from S3 into a database and do queries.
- But in Athena you can do queries directly you don't need to download or load your files
- Use case:
  1. Business Intelligence
  2. Analytics
  3. Reporting
  4. ELB logs
  5. Analyze and query vpc flow logs

Steps:

1. Create the database.

   create database s3_access_logs_db;

2. Create a table in the database. For LOCATION, enter the S3 bucket and prefix path from step 1. Be sure to include a forward slash (/) at the end of the prefix (for example, s3://doc-example-bucket/prefix/).



You have not created any views. To create a view, run a query and click "Create view from query"

```
CREATE EXTERNAL TABLE IF NOT EXISTS s3_access_logs_db.mybucket_logs(
         BucketOwner STRING,
         Bucket STRING,
         RequestDateTime STRING,
         RemoteIP STRING,
         Requester STRING,
         RequestID STRING,
         Operation STRING,
         Key STRING,
         RequestURI_operation STRING,
         RequestURI_key STRING,
         RequestURI_httpProtoversion STRING,
         HTTPstatus STRING,
         ErrorCode STRING,
         BytesSent BIGINT,
         ObjectSize BIGINT,
         TotalTime STRING,
         TurnAroundTime STRING,
         Referrer STRING,
         UserAgent STRING,
```

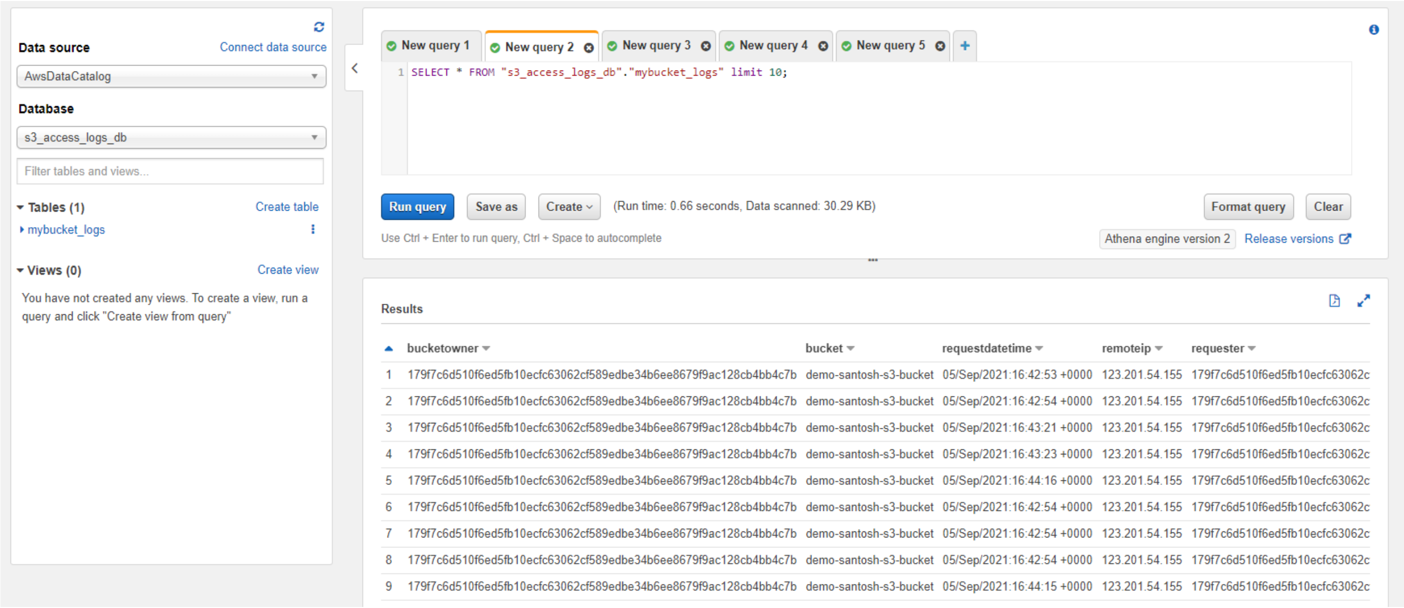
(Run time: 0.3 seconds, Data scanned: 0 KB)

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Results

```
Query successful.
```

3. Display the first 10 rows from the S3 bucket (demo-s3-access-logs-santosh).



Data source: AwsDataCatalog

Database: s3_access_logs_db

Tables (1)

- mybucket_logs

Views (0)

You have not created any views. To create a view, run a query and click "Create view from query"

```
SELECT * FROM "s3_access_logs_db"."mybucket_logs" limit 10;
```

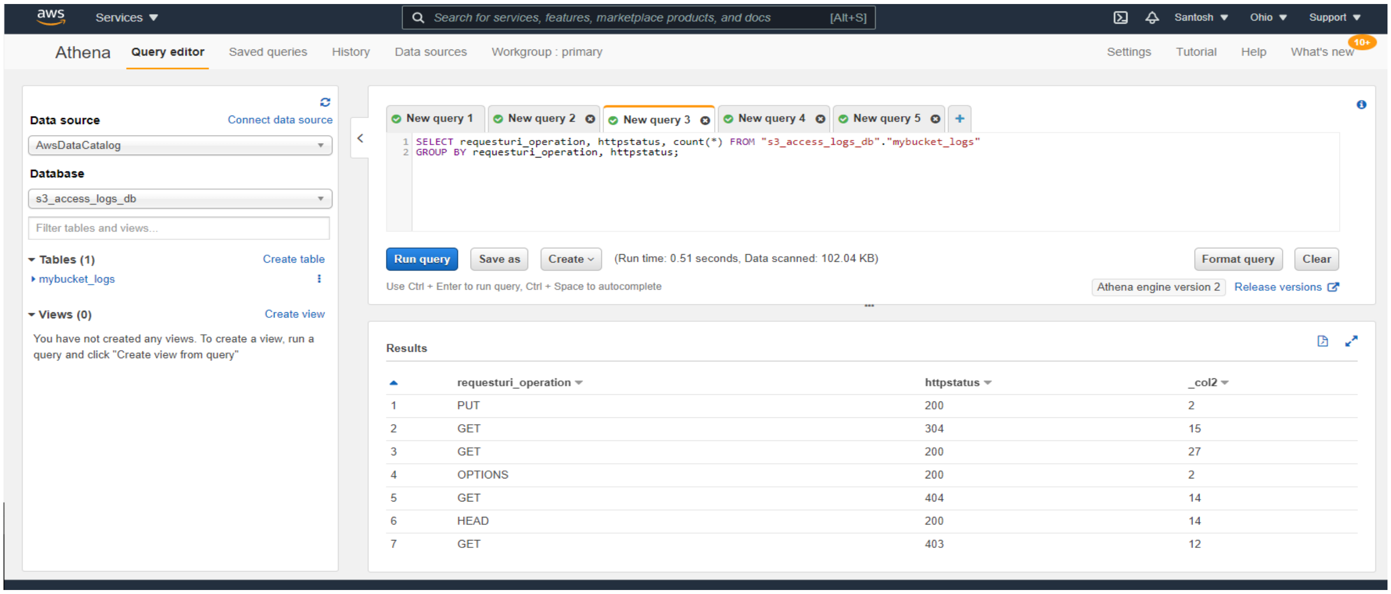
(Run time: 0.66 seconds, Data scanned: 30.29 KB)

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Athena engine version 2

	bucketowner	bucket	requestdatetime	remoteip	requester
1	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:42:53 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
2	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:42:54 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
3	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:43:21 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
4	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:43:23 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
5	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:16 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
6	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:42:54 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
7	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:42:54 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
8	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:42:54 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c
9	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:15 +0000	123.201.54.155	179f7c6d510f6ed5fb10ecfc63062c

4. Display the shows httpstatus, requesturi_operation, and count of hits.



Data source: AwsDataCatalog

Database: s3_access_logs_db

Tables (1)

- mybucket_logs

Views (0)

You have not created any views. To create a view, run a query and click "Create view from query"

```
SELECT requesturi_operation, httpstatus, count(*) FROM "s3_access_logs_db"."mybucket_logs"
GROUP BY requesturi_operation, httpstatus;
```

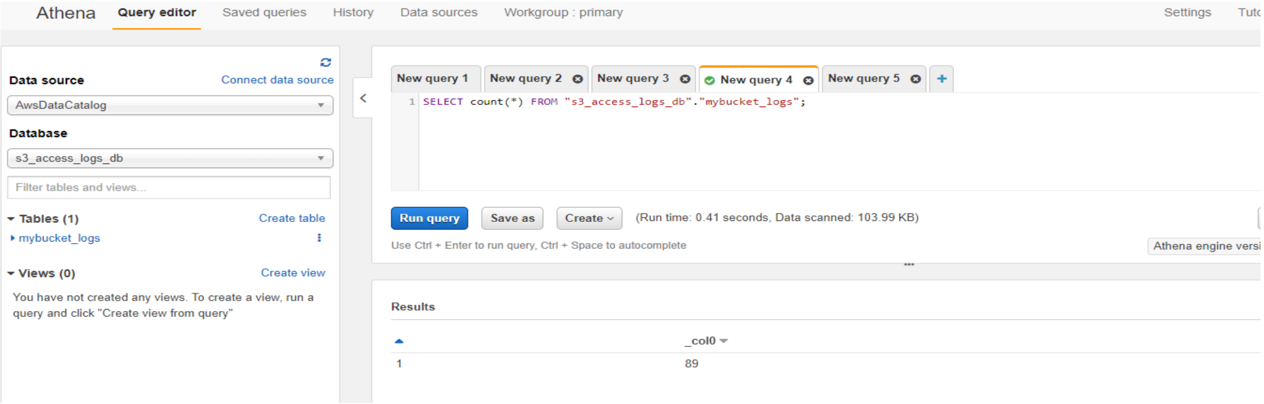
(Run time: 0.51 seconds, Data scanned: 102.04 KB)

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Athena engine version 2

	requesturi_operation	httpstatus	_col2
1	PUT	200	2
2	GET	304	15
3	GET	200	27
4	OPTIONS	200	2
5	GET	404	14
6	HEAD	200	14
7	GET	403	12

5. Shows how many access logs rows.



Data source: AwsDataCatalog

Database: s3_access_logs_db

You have not created any views. To create a view, run a query and click "Create view from query"

```
SELECT count(*) FROM "s3_access_logs_db"."mybucket_logs";
```

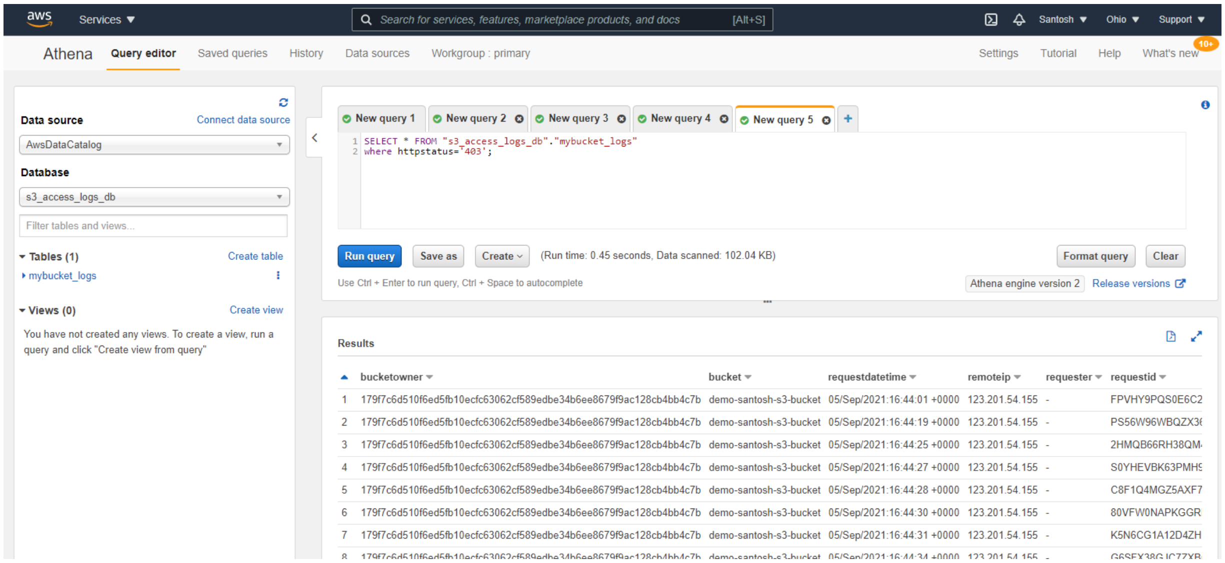
(Run time: 0.41 seconds, Data scanned: 103.99 KB)

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Results

	_col0
1	89

6. Display 403 httpstatus data.



Data source: AwsDataCatalog

Database: s3_access_logs_db

You have not created any views. To create a view, run a query and click "Create view from query"

```
SELECT * FROM "s3_access_logs_db"."mybucket_logs"
where httpstatus='403';
```

(Run time: 0.45 seconds, Data scanned: 102.04 KB)

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Results

	bucketowner	bucket	requestdatetime	remoteip	requester	requestid
1	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:01 +0000	123.201.54.155	-	FPVHY9PQS0E6C2
2	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:19 +0000	123.201.54.155	-	PS56W96WBQZX3
3	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:25 +0000	123.201.54.155	-	2HMQB66RH38QM
4	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:27 +0000	123.201.54.155	-	S0YHEVBK63PMH
5	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:28 +0000	123.201.54.155	-	C8F1Q4MGZ5AXF7
6	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:30 +0000	123.201.54.155	-	80VFW0NAPKGGR
7	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:31 +0000	123.201.54.155	-	K5N6CG1A12D4ZH
8	179f7c6d510f6ed5fb10ecfc63062cf589edbe34b6ee8679f9ac128cb4bb4c7b	demo-santosh-s3-bucket	05/Sep/2021:16:44:34 +0000	123.201.54.155	-	G6SFX38GJC7ZXB

Reference: https://aws.amazon.com/premiumsupport/knowledge-center/analyze-logs-athena/