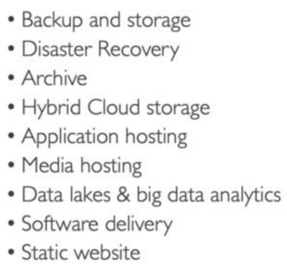
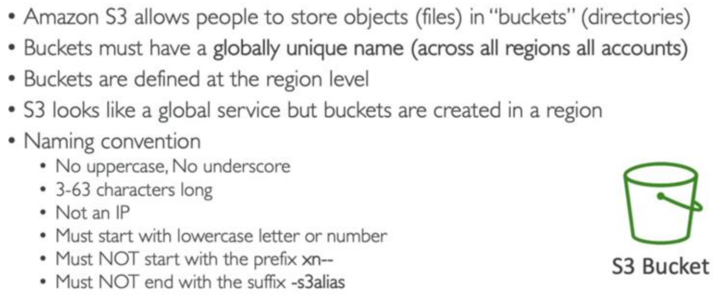
Simple Storage Service -S3

S3 use cases



- Backup and storage
- Disaster Recovery
- Archive
- Hybrid Cloud storage
- Application hosting
- Media hosting
- Data lakes & big data analytics
- Software delivery
- Static website

S3 -Buckets

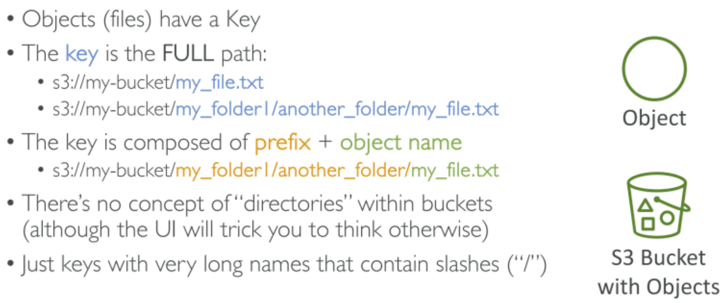


- Amazon S3 allows people to store objects (files) in "buckets" (directories)
- Buckets must have a **globally unique name (across all regions all accounts)**
- Buckets are defined at the region level
- S3 looks like a global service but buckets are created in a region
- Naming convention
  - No uppercase, No underscore
  - 3-63 characters long
  - Not an IP
  - Must start with lowercase letter or number
  - Must NOT start with the prefix **xn--**
  - Must NOT end with the suffix **-s3alias**

Availability: up time 90% means in 10 days it will be down 1 day. 100% availability means it will never down.

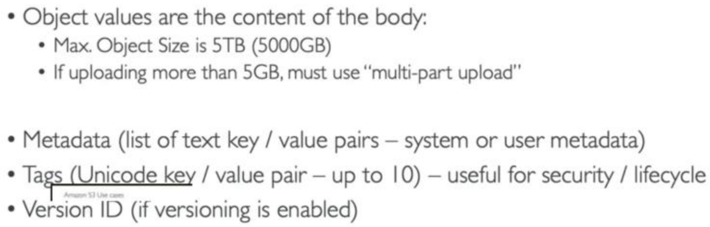
Durability: data persistent 90% durability means if we upload 10 files 9 files will be keep safe 1 file will be damage by any reason.

S3 -Objects



- Objects (files) have a Key
- The key is the FULL path:
  - s3://my-bucket/my_file.txt
  - s3://my-bucket/my_folder1/another_folder/my_file.txt
- The key is composed of prefix + object name
  - s3://my-bucket/my_folder1/another_folder/my_file.txt
- There's no concept of "directories" within buckets (although the UI will trick you to think otherwise)
- Just keys with very long names that contain slashes ("/")

S3 – Objects (cont.)



- Object values are the content of the body:
  - Max. Object Size is 5TB (5000GB)
  - If uploading more than 5GB, must use "multi-part upload"

- Metadata (list of text key / value pairs – system or user metadata)
- Tags (Unicode key / value pair – up to 10) – useful for security / lifecycle
- Version ID (if versioning is enabled)

S3- Security

- User-Based
  - IAM Policies – which API calls should be allowed for a specific user from IAM

- Resource-Based
  - Bucket Policies – bucket wide rules from the S3 console - allows cross account
  - Object Access Control List (ACL) – finer grain (can be disabled)
  - Bucket Access Control List (ACL) – less common (can be disabled)

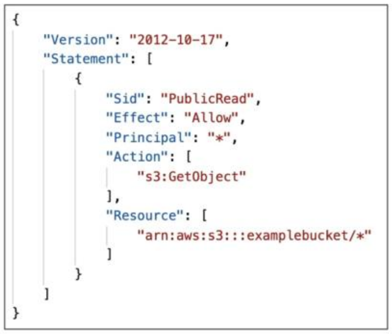
- Note: an IAM principal can access an S3 object if
  - The user IAM permissions ALLOW it OR the resource policy ALLOWS it
  - AND there's no explicit DENY

- Encryption: encrypt objects in Amazon S3 using encryption keys

S3 Bucket Policies

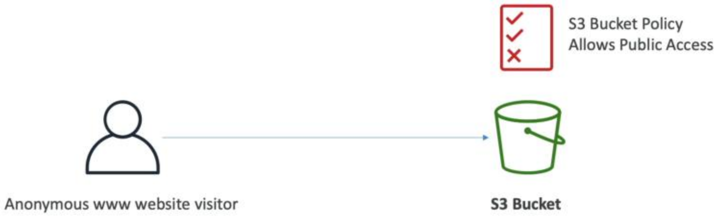
- JSON based policies
  - Resources: buckets and objects
  - Effect: Allow / Deny
  - Actions: Set of API to Allow or Deny
  - Principal: The account or user to apply the policy to

- Use S3 bucket for policy to:
  - Grant public access to the bucket
  - Force objects to be encrypted at upload

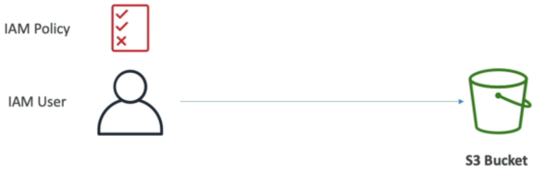


```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicRead",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::examplebucket/*"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

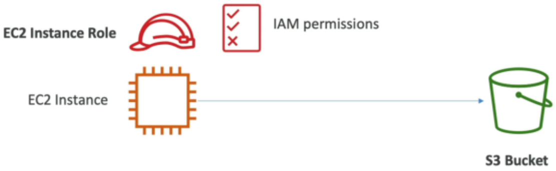
Example: Public Access – Use Bucket Policy



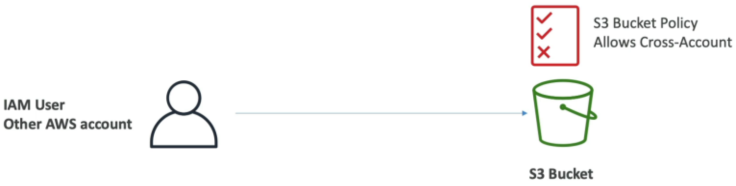
Example: User Access to S3 – IAM permission



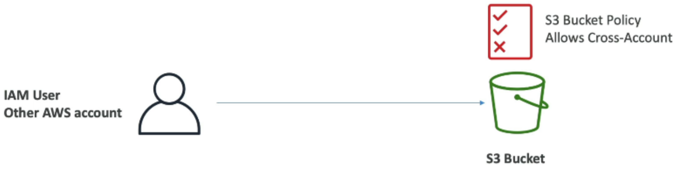
Example: Ec2 instance access – Use IAM role



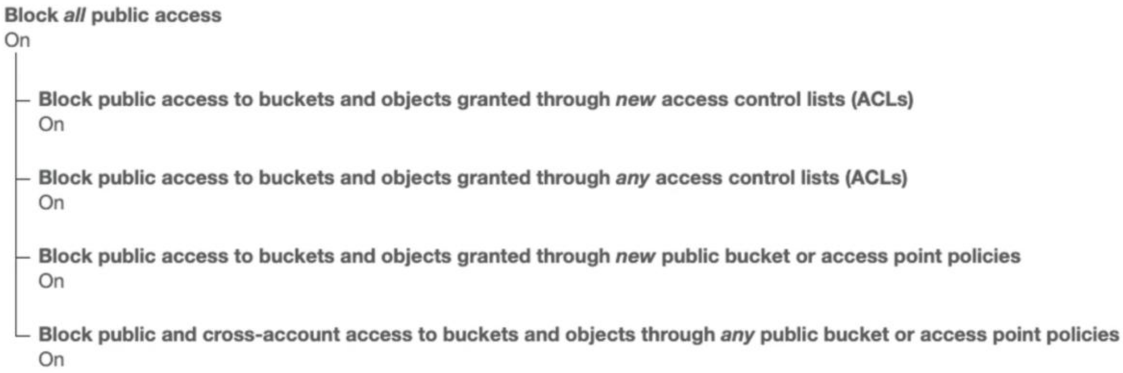
Cross Account Access – Use bucket policy



Cross Account access Using bucket policy

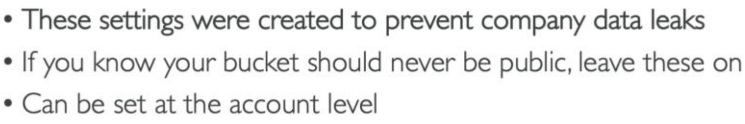


Bucket setting for Block public Access



Block *all* public access
On

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

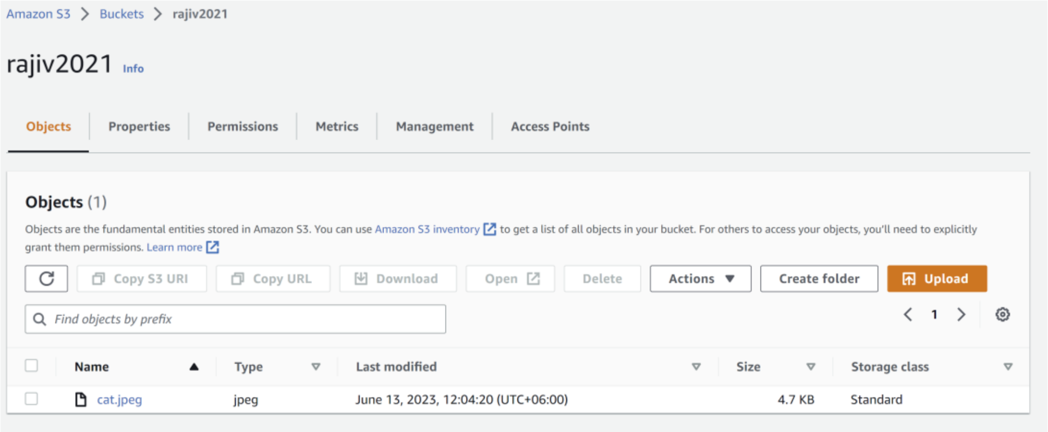


- These settings were created to prevent company data leaks
- If you know your bucket should never be public, leave these on
- Can be set at the account level

Lab:1

Create a bucket
upload a image to the bucket
check

Create a bucket
Upload a image to the bucket
Change the permissions

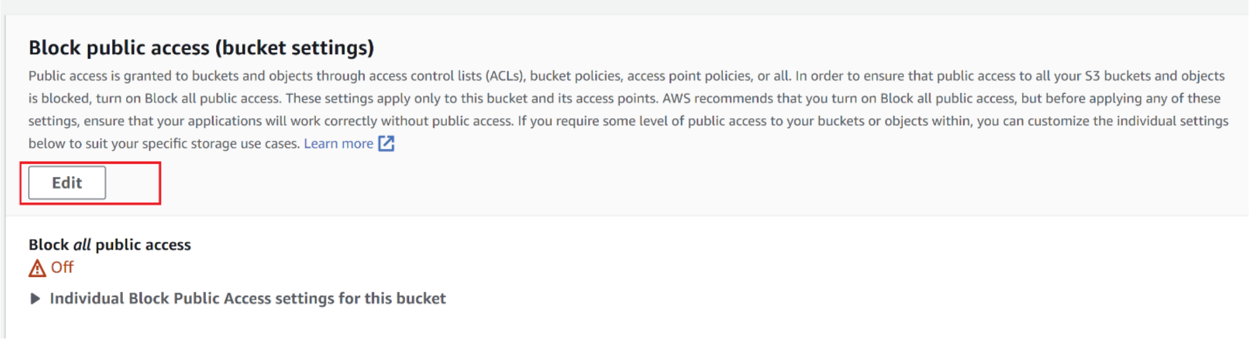


rajiv2021

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
□	cat.jpeg	jpeg	June 13, 2023, 12:04:20 (UTC+06:00)	4.7 KB	Standard

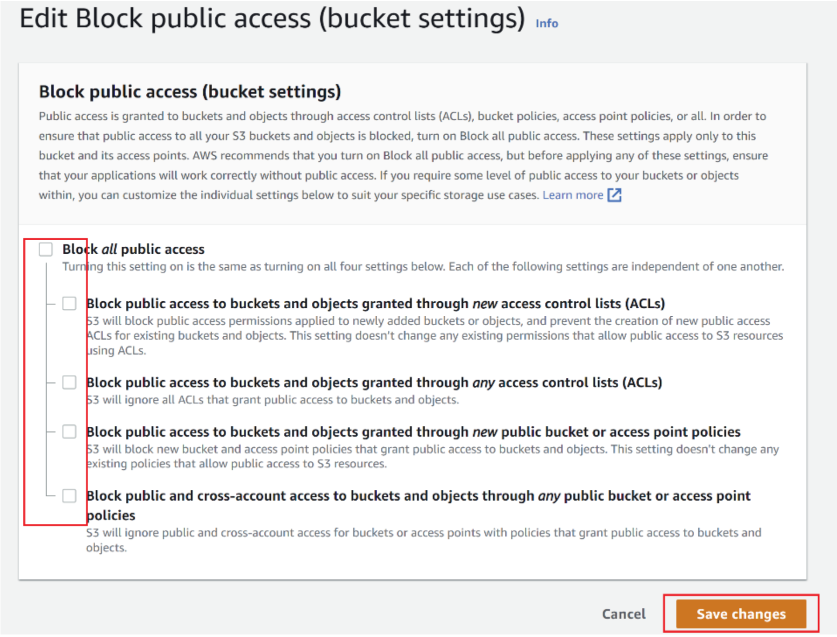


Block public access (bucket settings)

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
⚠ Off

▶ Individual Block Public Access settings for this bucket



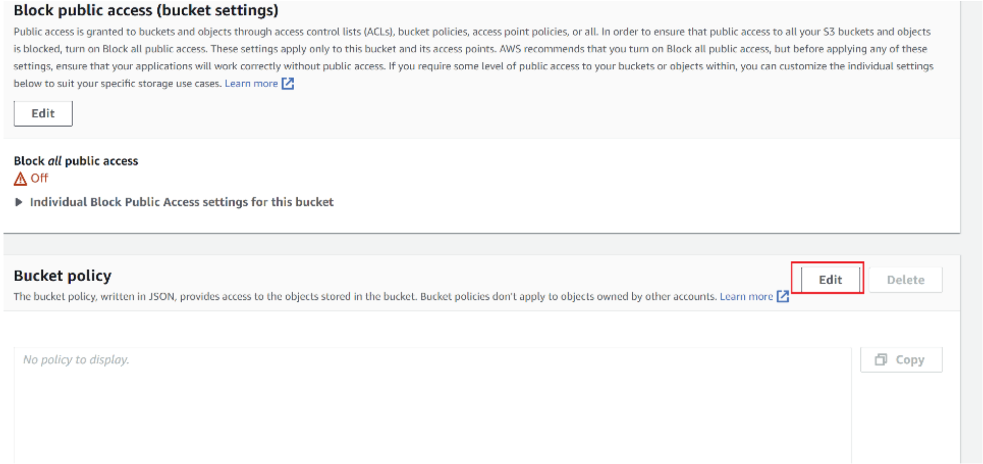
Edit Block public access (bucket settings)

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.



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

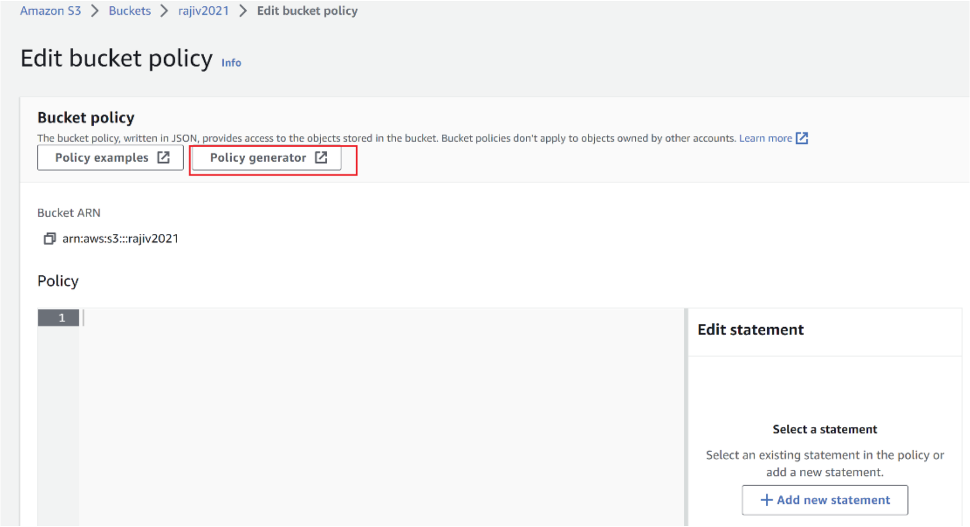
⚠ Off

▶ Individual Block Public Access settings for this bucket

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

No policy to display.



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

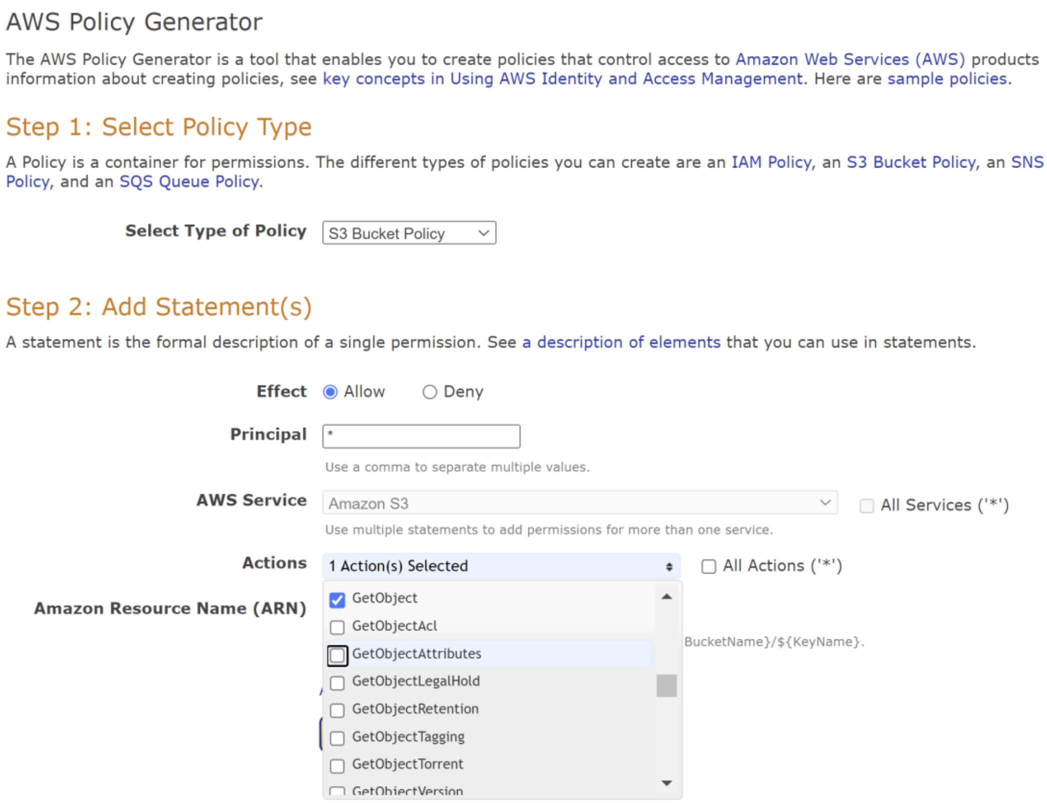
arn:aws:s3:::rajiv2021

Policy

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.



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
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 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 ('*')

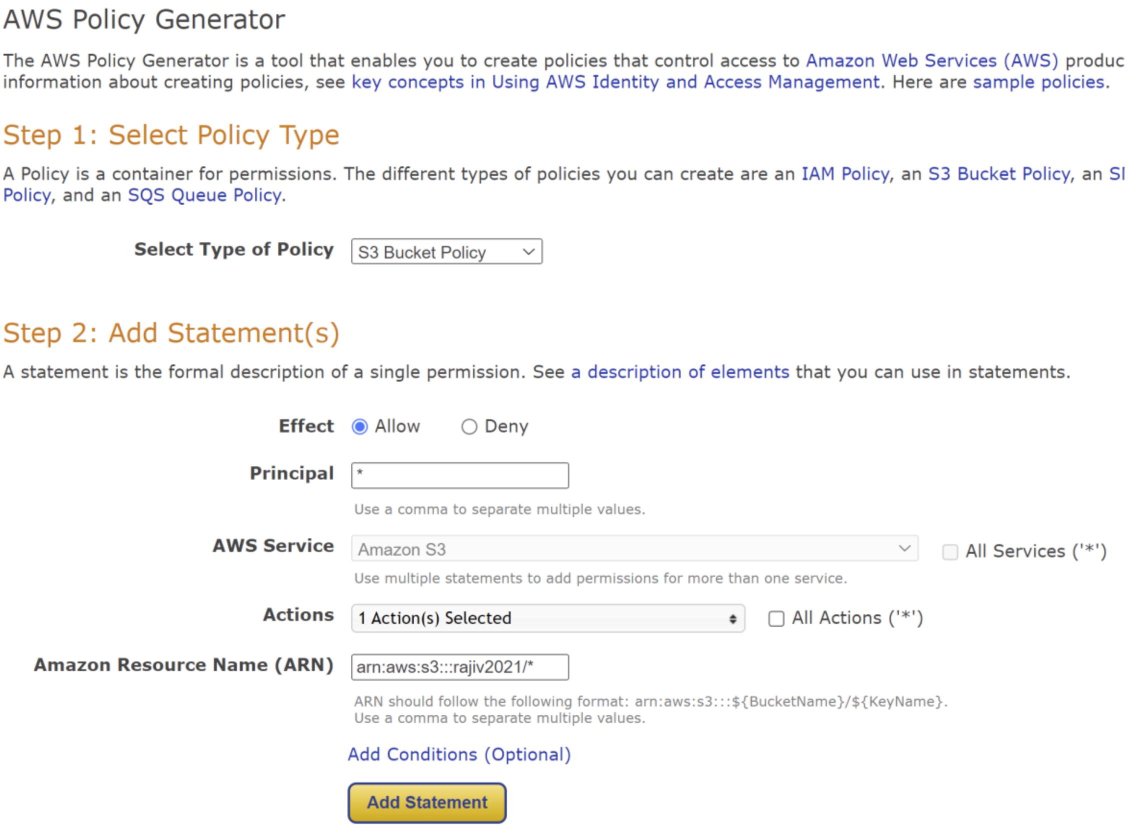
Use multiple statements to add permissions for more than one service.

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

Amazon Resource Name (ARN)

☑ GetObject
□ GetObjectAcl
□ GetObjectAttributes
□ GetObjectLegalHold
□ GetObjectRetention
□ GetObjectTagging
□ GetObjectTorrent
□ GetObjectVersion

BucketName}/${KeyName}.



AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) produc
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 SI
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.

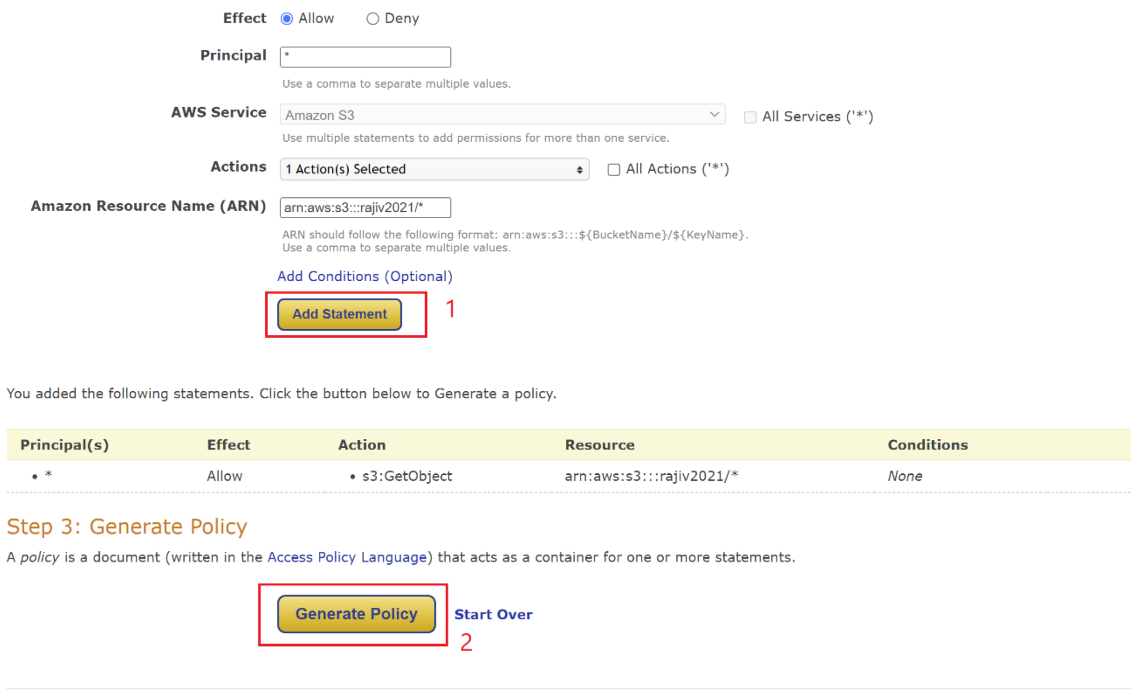
AWS Service Amazon S3 □ All Services ('*')

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:::rajiv2021/*

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



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:::rajiv2021/*`

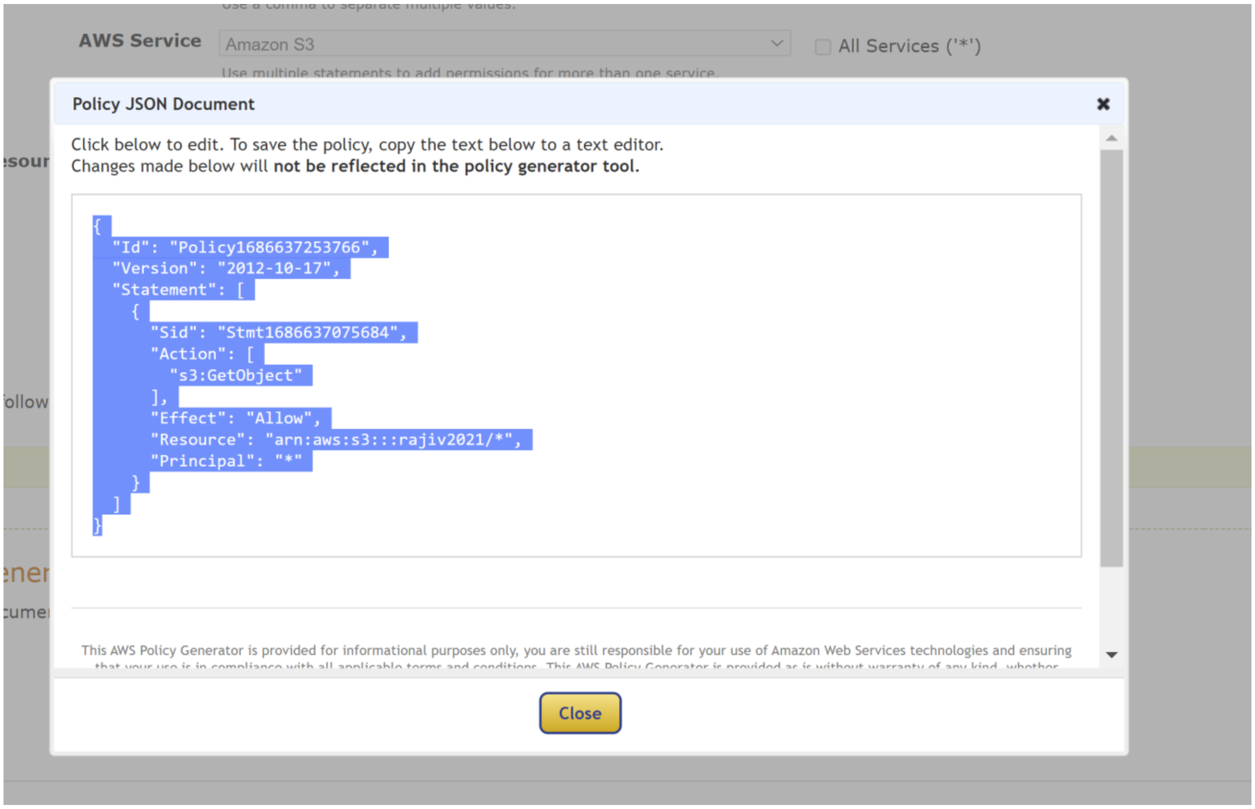
ARN should follow the following format: arn:aws:s3:::${BucketName}/${KeyName}.
Use a comma to separate multiple values.

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:::rajiv2021/*	*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.



Use a comma to separate multiple values.

AWS Service `Amazon S3` □ All Services ('*')

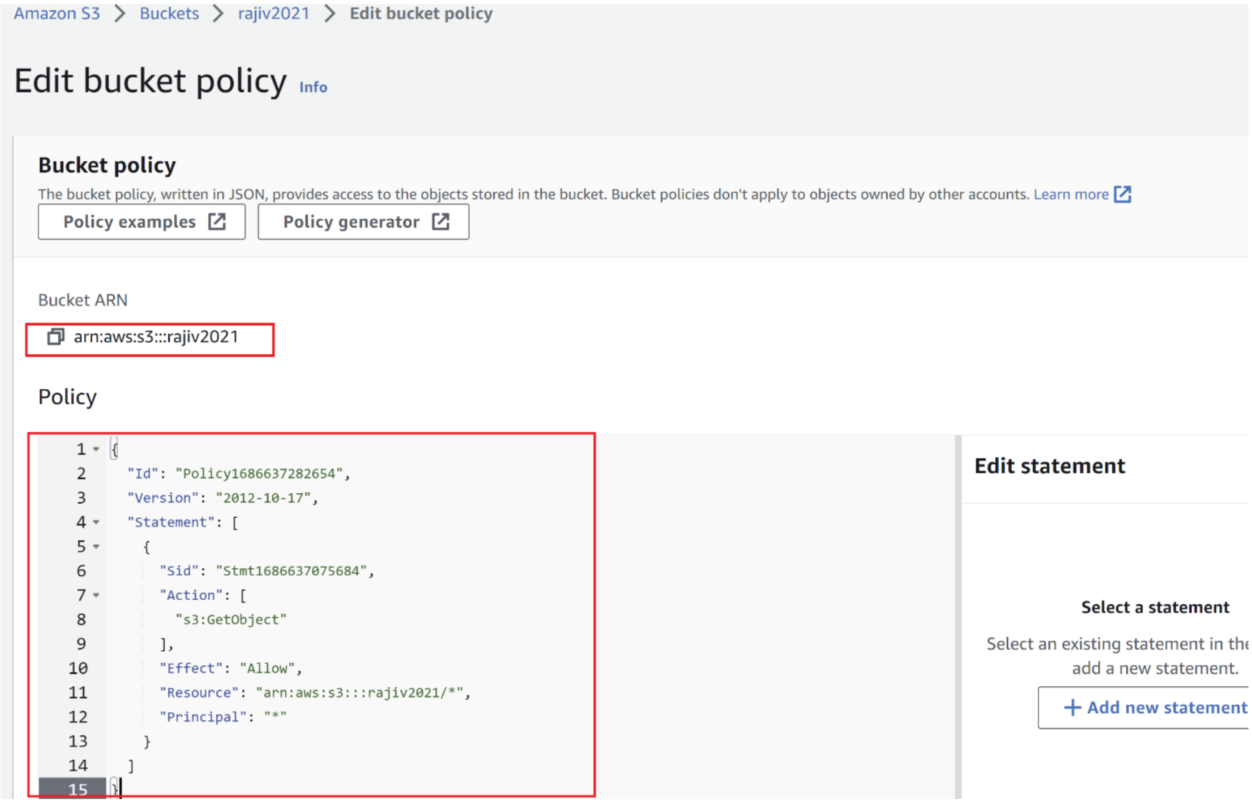
Use multiple statements to add permissions for more than one service.

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": "Policy1686637253766",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1686637075684",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::rajiv2021/*",
      "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



Edit bucket policy

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:::rajiv2021

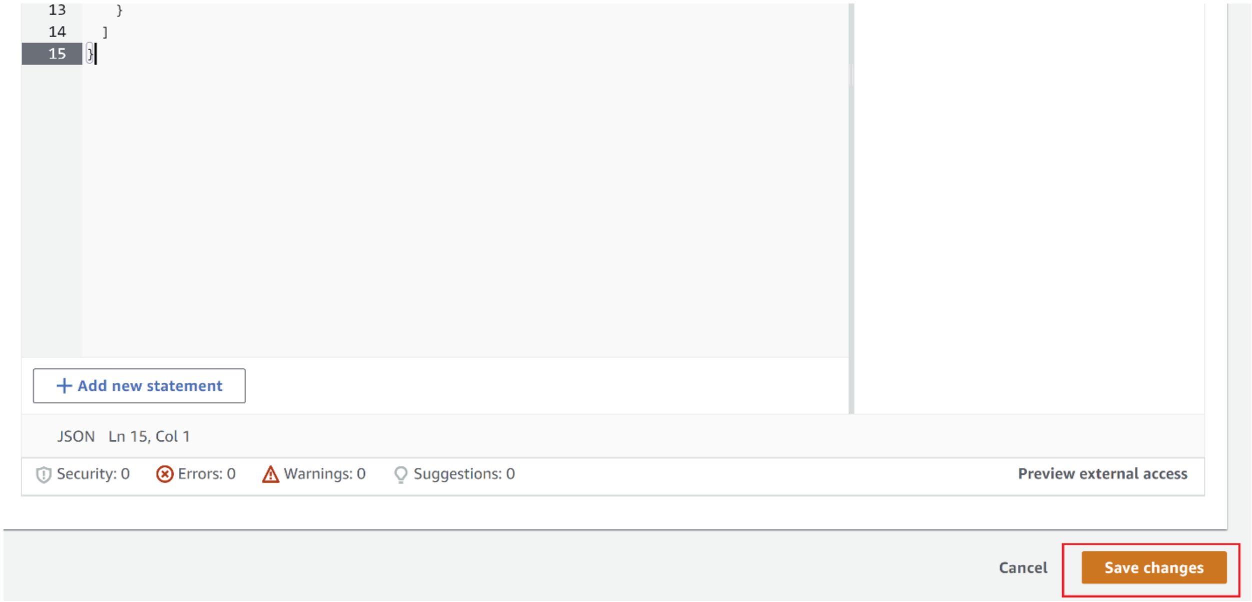
Policy

```
{
  "Id": "Policy1686637282654",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1686637075684",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::rajiv2021/*",
      "Principal": "*"
    }
  ]
}
```

Edit statement

Select a statement

Select an existing statement in the add a new statement.


```
    }
  ]
}
```

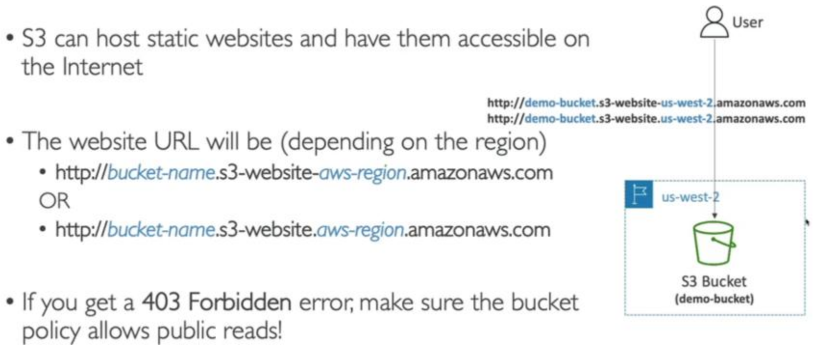
JSON Ln 15, Col 1

Security: 0 Errors: 0 Warnings: 0 Suggestions: 0

Now we can browse the object as a public from the following URL

https://rajiv2021.s3.amazonaws.com/cat.jpeg

Amazon S3 – Static Website Hosting

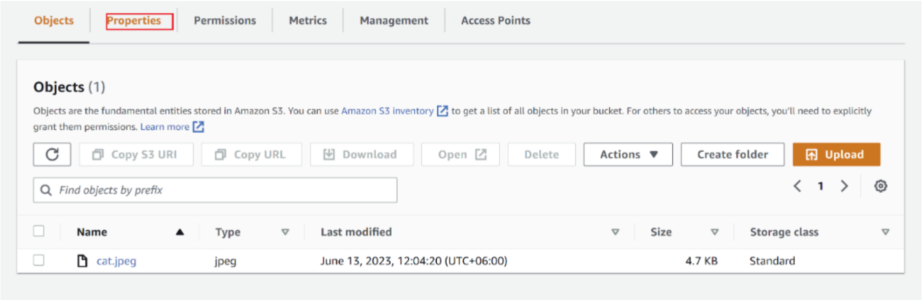


- S3 can host static websites and have them accessible on the Internet
- The website URL will be (depending on the region)
    - http://*bucket-name*.s3-website-*aws-region*.amazonaws.com

  OR
    - http://*bucket-name*.s3-website.*aws-region*.amazonaws.com
- If you get a **403 Forbidden** error, make sure the bucket policy allows public reads!

User
http://demo-bucket.s3-website-us-west-2.amazonaws.com
http://demo-bucket.s3-website.us-west-2.amazonaws.com
us-west-2
S3 Bucket
(demo-bucket)

Lab: Create a static website

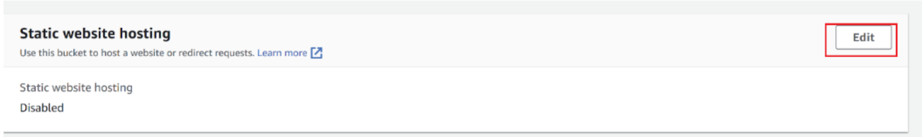


Objects (1)

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
□	cat.jpeg	jpeg	June 13, 2023, 12:04:20 (UTC+06:00)	4.7 KB	Standard

Now scroll down the page and you get the option

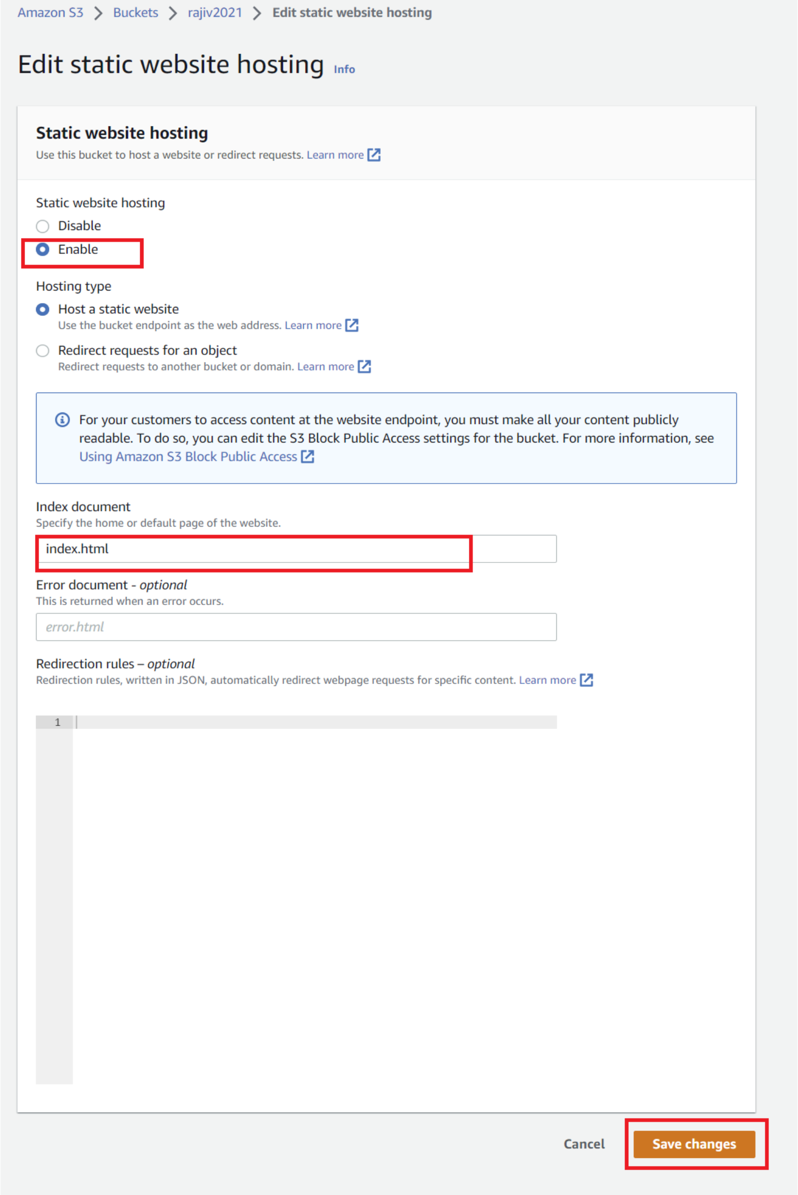


Static website hosting

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

Static website hosting

Disabled



Amazon S3 > Buckets > rajiv2021 > 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*

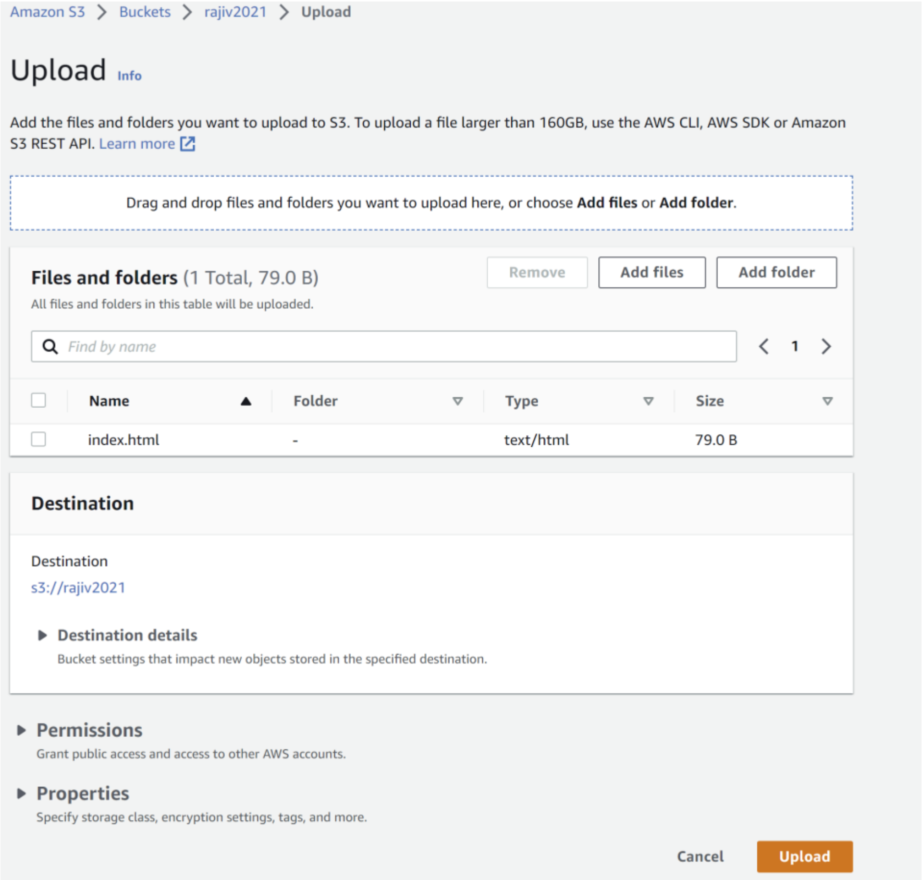
This is returned when an error occurs.

error.html

Redirection rules – *optional*

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

Now go to the object and upload the index.html file



Upload

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

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

Files and folders (1 Total, 79.0 B)

All files and folders in this table will be uploaded.

□	Name	Folder	Type	Size
□	index.html	-	text/html	79.0 B

Destination

Destination

s3://rajiv2021

Destination details

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

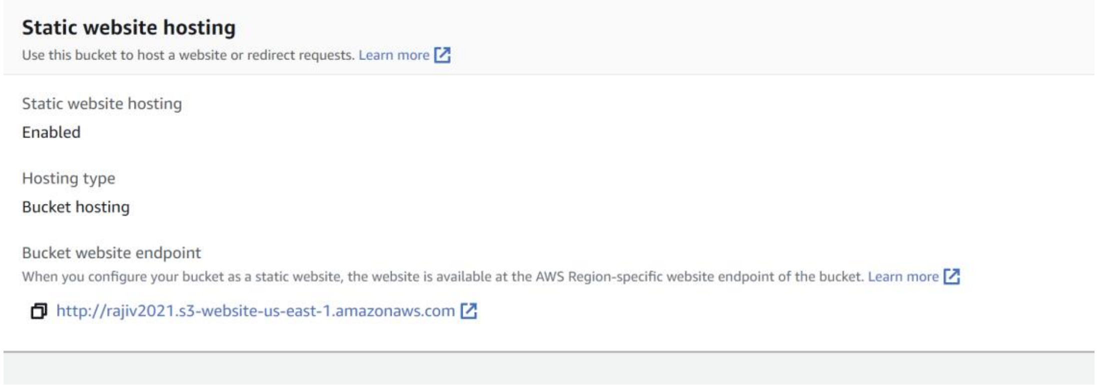
Permissions

Grant public access and access to other AWS accounts.

Properties

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

now go to the properties and scroll down and you will get the s3 website url



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

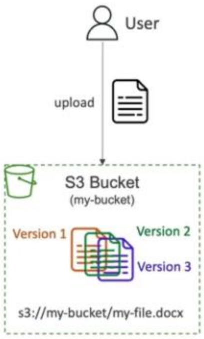
When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. Learn more

http://rajiv2021.s3-website-us-east-1.amazonaws.com

now go to the browser and pest the URL and you will get the website.

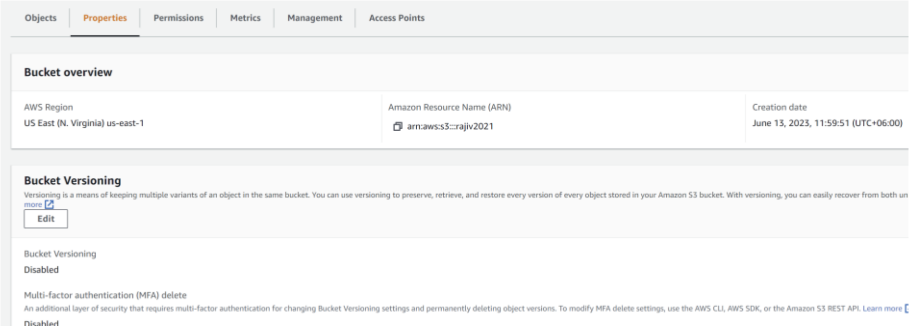
Amazon S3 - Versioning

- You can version your files in Amazon S3
- It is enabled at the **bucket level**
- Same key overwrite will change the "version": 1, 2, 3….
- It is best practice to version your buckets
  - Protect against unintended deletes (ability to restore a version)
  - Easy roll back to previous version
- Notes:
  - Any file that is not versioned prior to enabling versioning will have version "null"
  - Suspending versioning does not delete the previous versions



Lab: how to enable versioning

First enable the versioning



Bucket overview

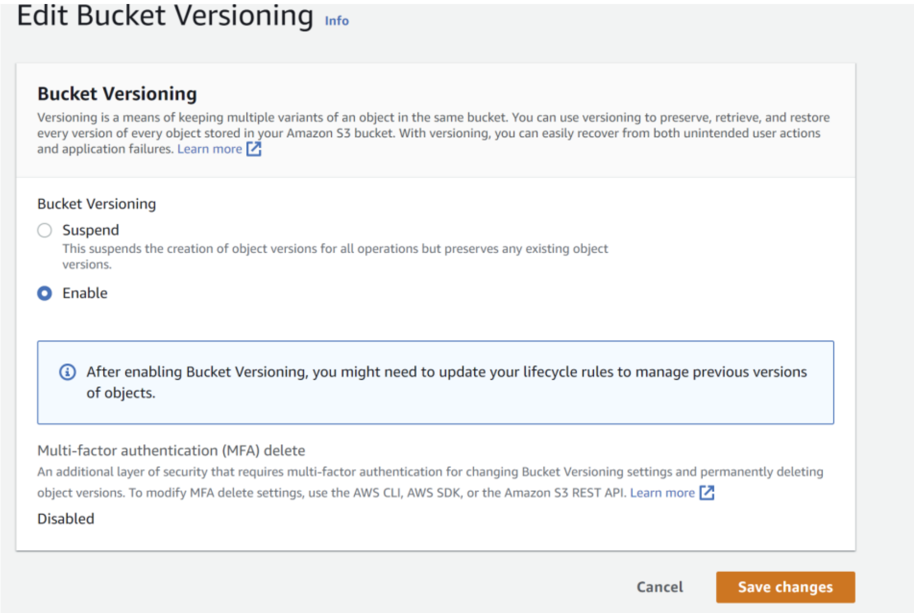
AWS Region	Amazon Resource Name (ARN)	Creation date
US East (N. Virginia) us-east-1	arn:aws:s3:::rajiv2021	June 13, 2023, 11:59:51 (UTC+06:00)

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 un
more

Bucket Versioning
Disabled

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



Edit Bucket Versioning

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.

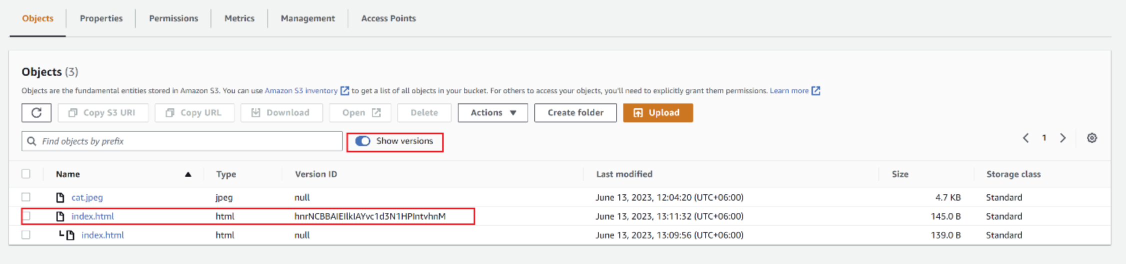
● Enable

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

Now edit the index file and upload it and then check.



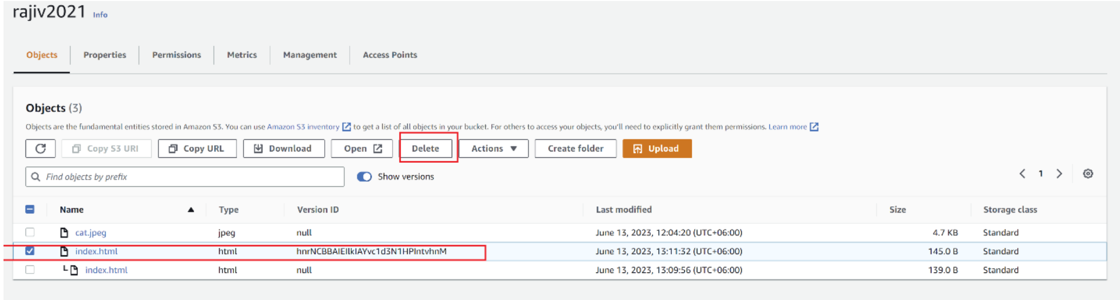
Objects (3)

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
□	cat.jpeg	jpeg	null	June 13, 2023, 12:04:20 (UTC+06:00)	4.7 KB	Standard
□	index.html	html	hnrNCBBAIEIlkIAYvc1d3N1HPIntvhnM	June 13, 2023, 13:11:32 (UTC+06:00)	145.0 B	Standard
□	└ index.html	html	null	June 13, 2023, 13:09:56 (UTC+06:00)	139.0 B	Standard

NB: Version ID null means its uploaded before enable versioning.

Now we see the current updated page now it we want back to previous one page then delete the current one version and then brows it



rajiv2021

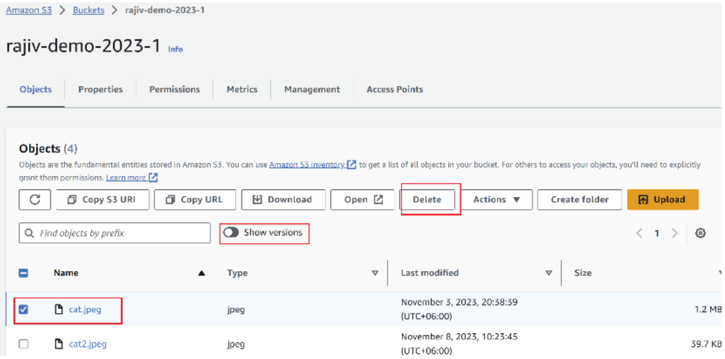
Objects (3)

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
□	cat.jpeg	jpeg	null	June 13, 2023, 12:04:20 (UTC+06:00)	4.7 KB	Standard
☑	index.html	html	hnrNCBBAIEIlkIAYvc1d3N1HPIntvhnM	June 13, 2023, 13:11:32 (UTC+06:00)	145.0 B	Standard
□	└ index.html	html	null	June 13, 2023, 13:09:56 (UTC+06:00)	139.0 B	Standard

Now if we browse then we get the previous version file

Now we delete any image example cat.jpeg



rajiv-demo-2023-1

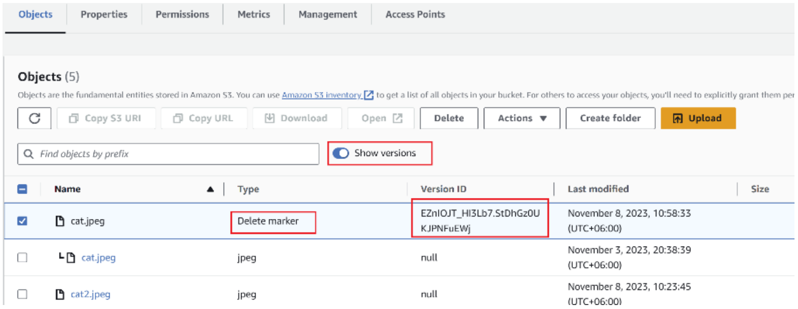
Objects (4)

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
☑	cat.jpeg	jpeg	November 3, 2023, 20:38:39 (UTC+06:00)	1.2 MB
□	cat2.jpeg	jpeg	November 8, 2023, 10:23:45 (UTC+06:00)	39.7 KB

after delete the cat.jpeg image then we can see the page is not showing.

now back to image again we need to delete the delete marker then we can see the image is showing again



Objects (5)

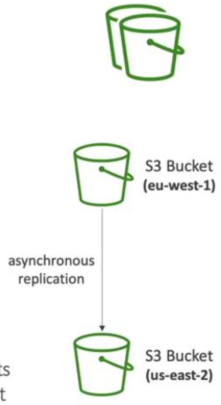
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them per

	Name	Type	Version ID	Last modified	Size
☑	cat.jpeg	Delete marker	EZnIOJT_HI3Lb7.StDhGz0UKJPNFuEWj	November 8, 2023, 10:58:33 (UTC+06:00)	
□	└ cat.jpeg	jpeg	null	November 3, 2023, 20:38:39 (UTC+06:00)	
□	cat2.jpeg	jpeg	null	November 8, 2023, 10:23:45 (UTC+06:00)	

NB: Force refresh sfift+R+reload or refresh the page

Amazon S3 - Replication (CRR and SRR)

- Must enable **Versioning** in source and destination buckets
- **Cross-Region Replication (CRR)**
- **Same-Region Replication (SRR)**
- Buckets can be in different AWS accounts
- Copying is asynchronous
- Must give proper IAM permissions to S3



- Use cases:
  - **CRR** – compliance, lower latency access, replication across accounts
  - **SRR** – log aggregation, live replication between production and test accounts

Amazon S3 – Replication (Notes)

- After you enable Replication, only new objects are replicated
- Optionally, you can replicate existing objects using **S3 Batch Replication**
  - Replicates existing objects and objects that failed replication

- For DELETE operations
  - **Can replicate delete markers** from source to target (optional setting)
  - Deletions with a version ID are not replicated (to avoid malicious deletes)

- **There is no "chaining" of replication**
  - If bucket 1 has replication into bucket 2, which has replication into bucket 3
  - Then objects created in bucket 1 are not replicated to bucket 3

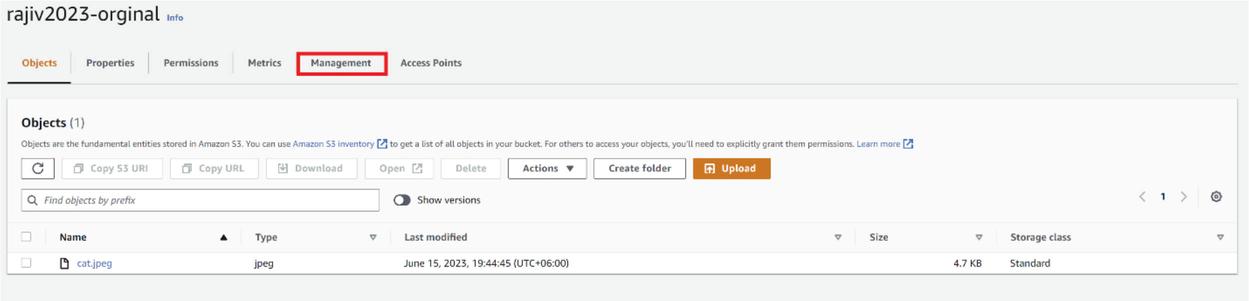
Lab- Replication

Create 2 buckets with versioning.

1.rajiv2023-original
2.rajiv-2023-replica

Now upload a image to rajiv-2023-orinal

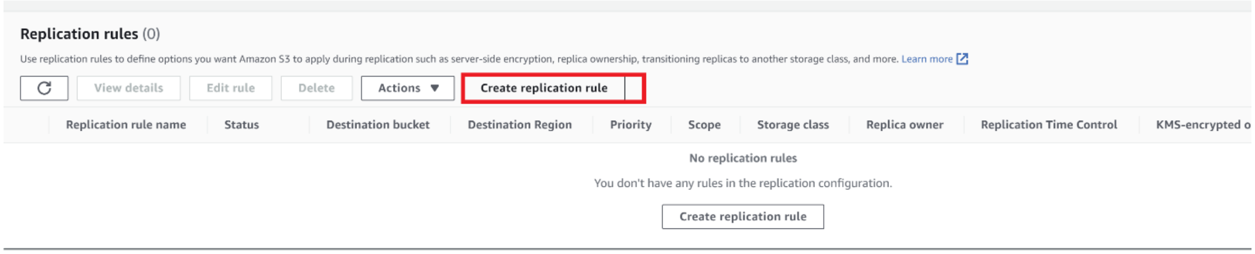
Now enable the replication in rajiv2023-original bucket



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
□	cat.jpeg	jpeg	June 15, 2023, 19:44:45 (UTC+06:00)	4.7 KB	Standard



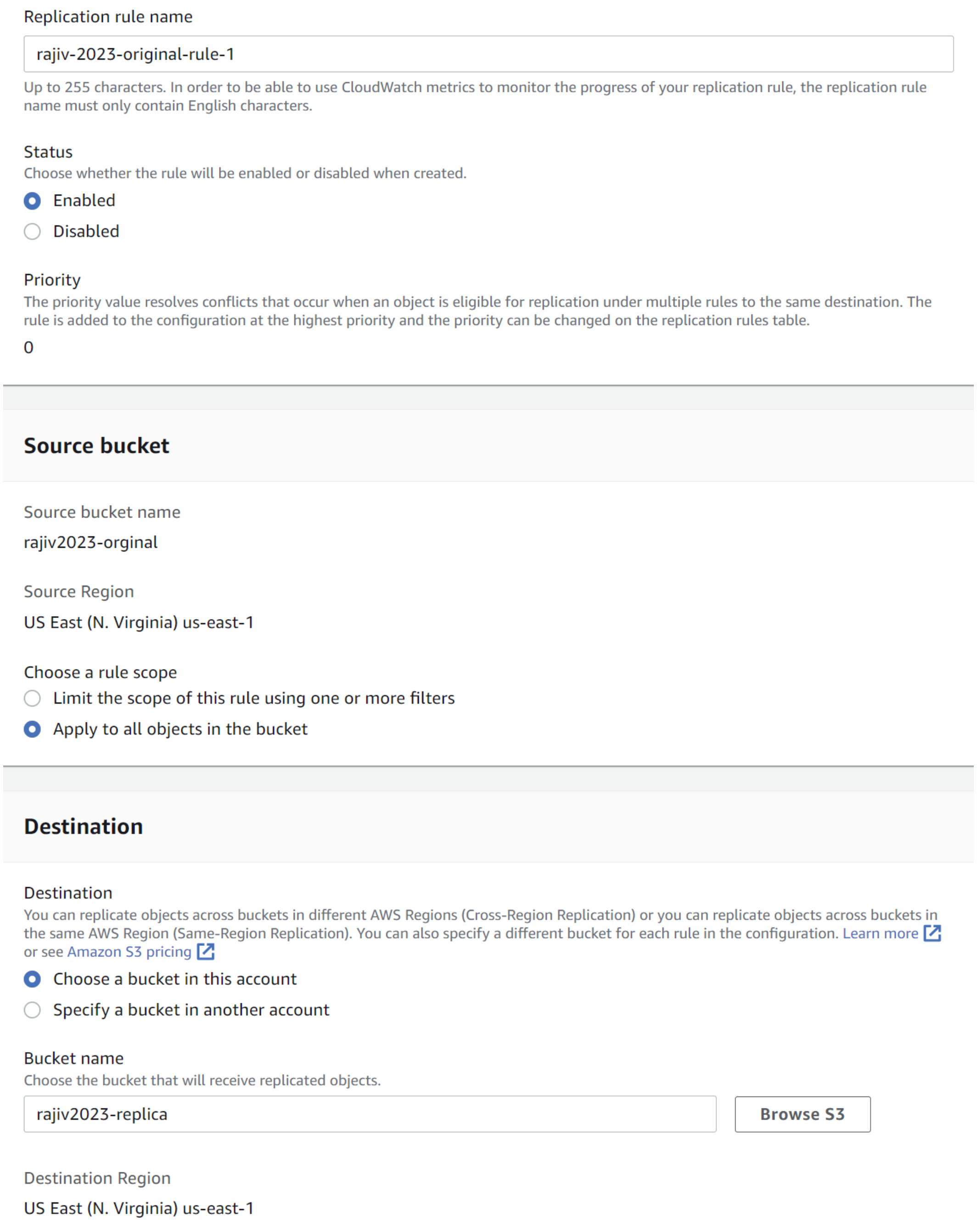
Replication rules (0)

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 o

No replication rules

You don't have any rules in the replication configuration.



Replication rule name

rajiv-2023-original-rule-1

Up to 255 characters. In order to be able to use CloudWatch metrics to monitor the progress of your replication rule, the replication rule name must only contain English 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

rajiv2023-orginal

Source Region

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

Choose a rule scope

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

● Apply to all objects in the bucket

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

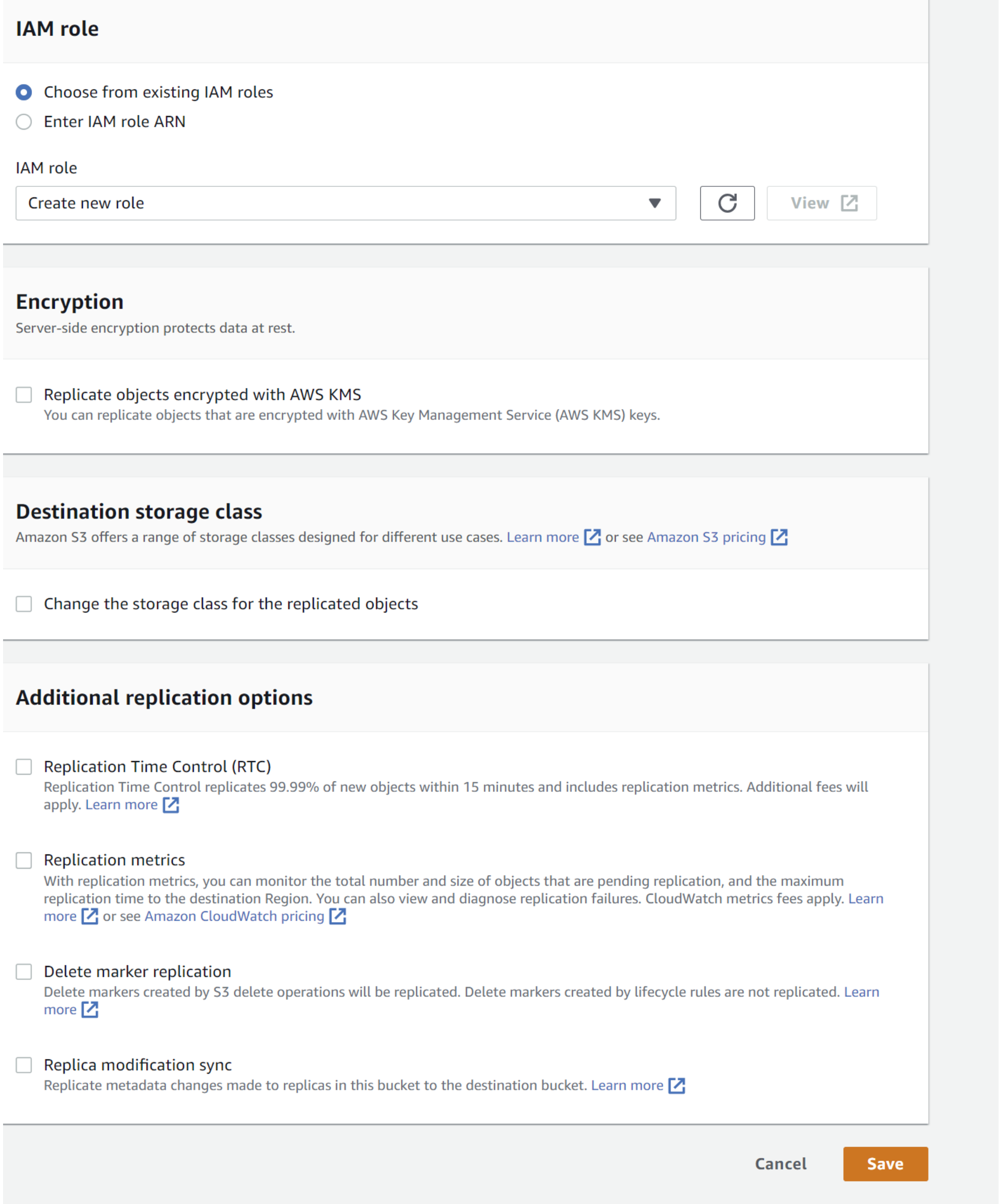
Bucket name

Choose the bucket that will receive replicated objects.

rajiv2023-replica

Destination Region

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



IAM role

● Choose from existing IAM roles
○ Enter IAM role ARN

IAM role

Create new role ▼

Encryption

Server-side encryption protects data at rest.

□ Replicate objects encrypted with AWS KMS
You can replicate objects that are encrypted with AWS Key Management Service (AWS KMS) keys.

Destination storage class

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

□ Change the storage class for the replicated objects

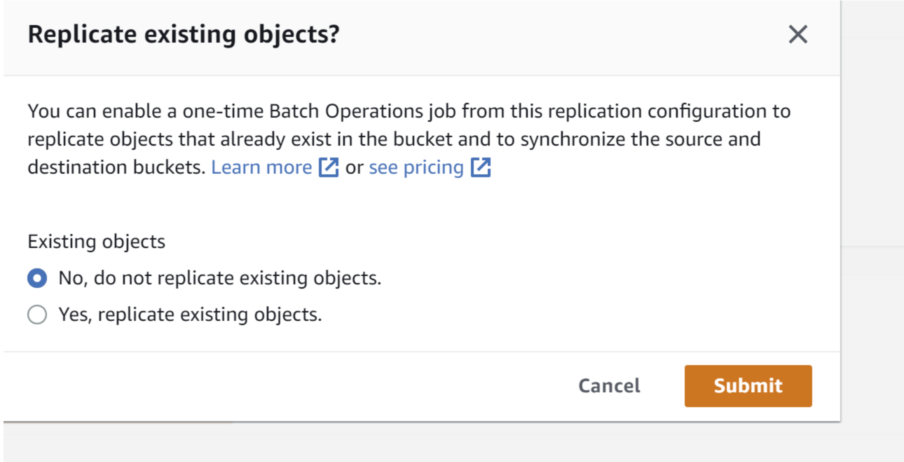
Additional replication options

□ Replication Time Control (RTC)
Replication Time Control replicates 99.99% of new objects within 15 minutes and includes replication metrics. Additional fees will apply. Learn more

□ Replication metrics
With replication metrics, you can monitor the total number and size of objects that are pending replication, and the maximum replication time to the destination Region. You can also view and diagnose replication failures. CloudWatch metrics fees apply. Learn more or see Amazon CloudWatch pricing

□ Delete marker replication
Delete markers created by S3 delete operations will be replicated. Delete markers created by lifecycle rules are not replicated. Learn more

□ Replica modification sync
Replicate metadata changes made to replicas in this bucket to the destination bucket. Learn more



Replicate existing objects?

You can enable a one-time Batch Operations job from this replication configuration to replicate objects that already exist in the bucket and to synchronize the source and destination buckets. Learn more or see pricing

Existing objects

● No, do not replicate existing objects.

○ Yes, replicate existing objects.

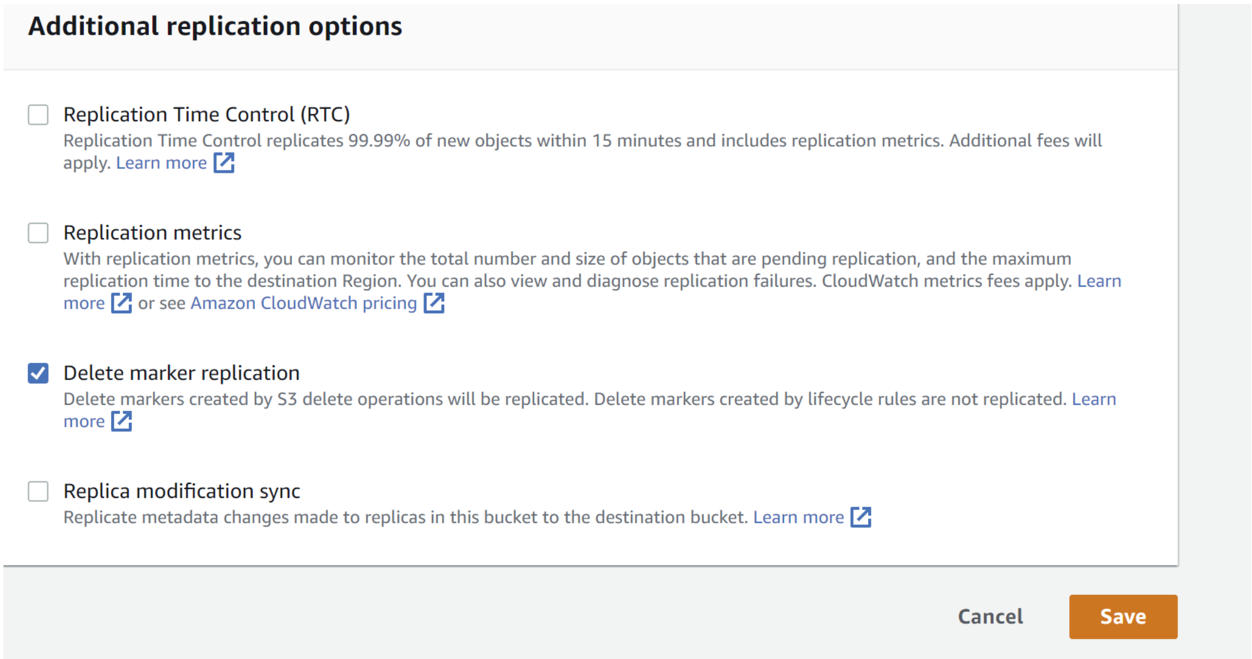
Now upload a picture in rajiv-2023-original

And now go to the rajiv-2023-replica bucket and wait for few seconds and refresh the page and we can see the image is now sowing here.

Enable the delete market

select the bucket >Management>select the rule>click edit rule>scroll down >and select the Delete marker replication and save it

Now it we delete any image from the original file the delete maker will replica to the replica bucket



Additional replication options

☐ Replication Time Control (RTC)
Replication Time Control replicates 99.99% of new objects within 15 minutes and includes replication metrics. Additional fees will apply. Learn more

☐ Replication metrics
With replication metrics, you can monitor the total number and size of objects that are pending replication, and the maximum replication time to the destination Region. You can also view and diagnose replication failures. CloudWatch metrics fees apply. Learn more or see Amazon CloudWatch pricing

☑ Delete marker replication
Delete markers created by S3 delete operations will be replicated. Delete markers created by lifecycle rules are not replicated. Learn more

☐ Replica modification sync
Replicate metadata changes made to replicas in this bucket to the destination bucket. Learn more

NB: When we delete any image, it will not copy to the replica bucket it only copy delete marker.
if we delete the image from original file permanently it will not delete the file from replica