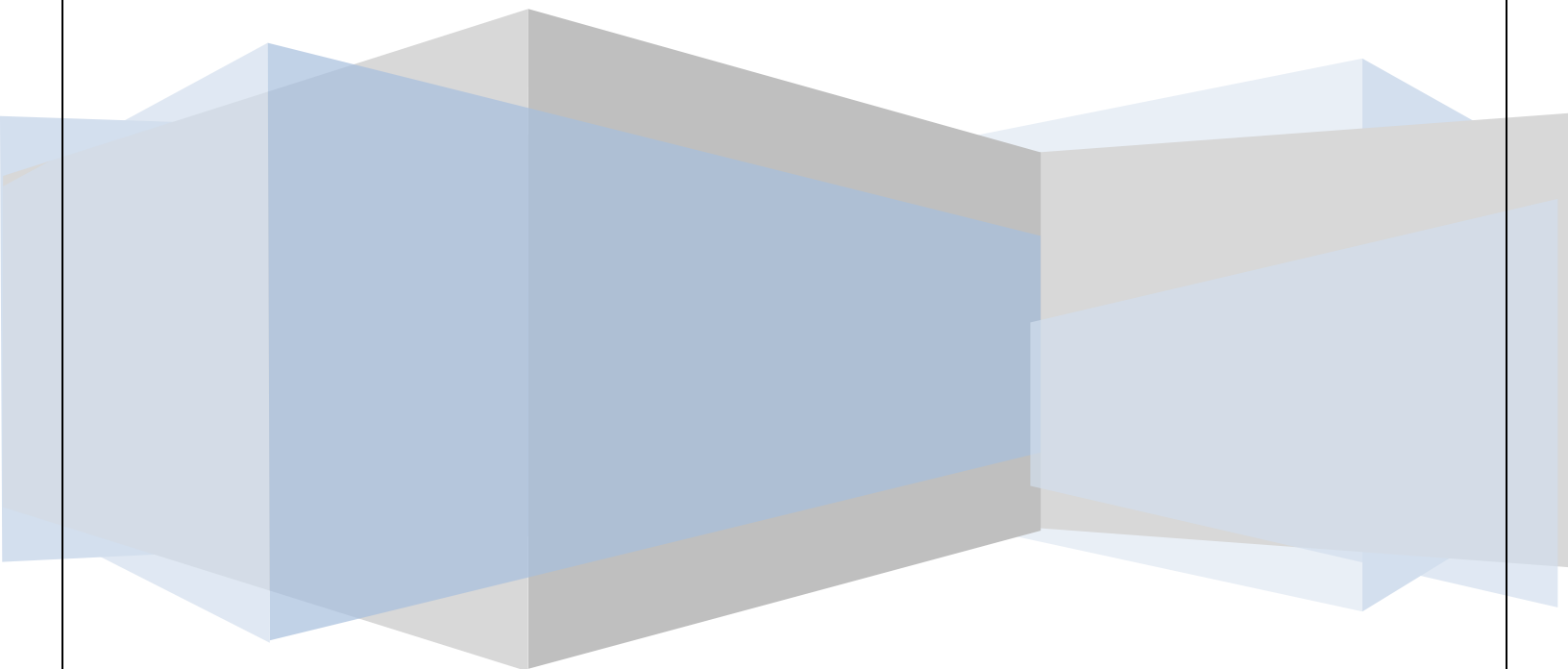


IT TOOK YOU YEARS OF EXPERIENCE  
TO REACH WHERE YOU ARE TODAY.  
ANOTHER **45 HOURS OF TRAINING**  
**WITH TACT WILL TAKE YOU HIGHER.**

TACT is an online technology academy for competency training to train the IT professionals and uplift their career to discover more opportunities in the space of emerging technologies.

**AWS Cloud Training**

**AWS SIMPLE STORAGE SERVICE**



# S3 (SIMPLE STORAGE SERVICE)

## Amazon S3: ESSENTIALS

### ➤ File and Object Storage

- Unlimited Storage
- High availability
- 99.999999999% Durability (Think of Disk set with RAID 1)
- 99.99% Availability (Spread across AZ's)
- Objects size can be from 1byte to 5TB

### ➤ Objects: Are static files

- Contains Key and Value pair (meta data)

### ➤ Buckets: Contain a group of objects/files

- Each bucket name must be unique across AWS S3 region

### ➤ Lifecycle Management

- Archive or Delete Objects

### ➤ Versioning

- preserve, retrieve, and restore every version of every object

### ➤ Encryption

- Server Side Encryption (S3 or KMS) or Client Side Encryption

## STORAGE TYPES

	Standard	Standard - Infrequent Access	Reduced Redundancy Storage
Durability	99.999999999%	99.999999999%	99.99%
Availability	99.99%	99.9%	99.99%

### GLACIER:





- Extremely low cost storage service.
- Archival storage, which takes 3 to 5 hours to for retrieval

# CREATE S3 BUCKET

Once you logged in to AWS console, choose S3 under Storage & Content Delivery From Management Console.

## Amazon Web Services


### Compute

-  **EC2**  
Virtual Servers in the Cloud
-  **EC2 Container Service**  
Run and Manage Docker Containers
-  **Elastic Beanstalk**  
Run and Manage Web Apps
-  **Lambda**  
Run Code in Response to Events




### Storage & Content Delivery

-  **S3**  
Scalable Storage in the Cloud
-  **CloudFront**  
Global Content Delivery Network

### Developer Tools

-  **CodeCommit**  
Store Code in Private Git Repositories
-  **CodeDeploy**  
Automate Code Deployments
-  **CodePipeline**  
Release Software using Continuous Delivery


### Management Tools

-  **CloudWatch**  
Monitor Resources and Applications
-  **CloudFormation**  
Create and Manage Resources with Templates
-  **CloudTrail**  
Track User Activity and API Usage




### Internet of Things

-  **AWS IoT**  
Connect Devices to the Cloud

### Game Development

-  **GameLift**  
Deploy and Scale Session-based Multiplayer Games

### Mobile Services

-  **Mobile Hub**  
Build, Test, and Monitor Mobile Apps
-  **Cognito**  
User Identity and App Data Synchronization
-  **Device Farm**  
Test Android, iOS, and Web Apps on Real Devices in the Cloud

Click on Create Bucket to create a new one.



Specify a name in Bucket Name text field, choose a region for Bucket, then click on Create.

**Create a Bucket - Select a Bucket Name and Region** Cancel

A bucket is a container for objects stored in Amazon S3. When creating a bucket, you can choose a Region to optimize for latency, minimize costs, or address regulatory requirements. For more information regarding bucket naming conventions, please visit the [Amazon S3 documentation](#).

**Bucket Name:**

**Region:** Singapore

- US Standard
- Oregon
- Northern California
- Ireland
- Singapore**
- Tokyo
- Sydney
- Sao Paulo
- Frankfurt
- Seoul

Set Up Logging > **Create** Cancel

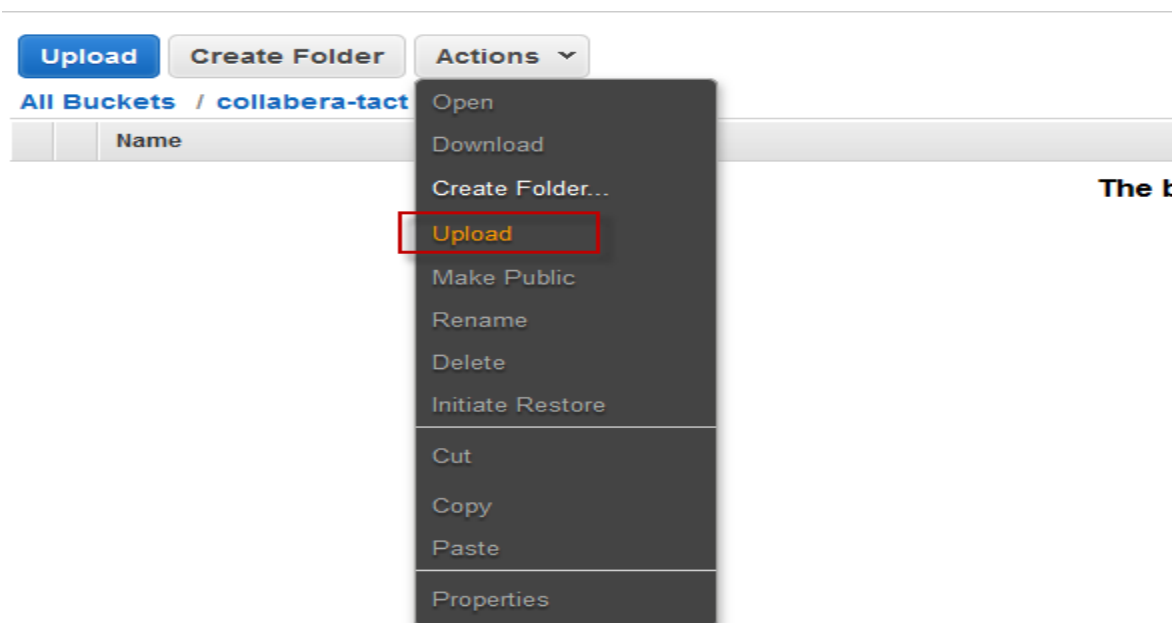
Choose the newly created Bucket to upload data.

Create Bucket Actions

**All Buckets (1)**

	Name
	<b>collabera-tact</b>
	elasticbeanstalk-ap-southeast-1-168600309204

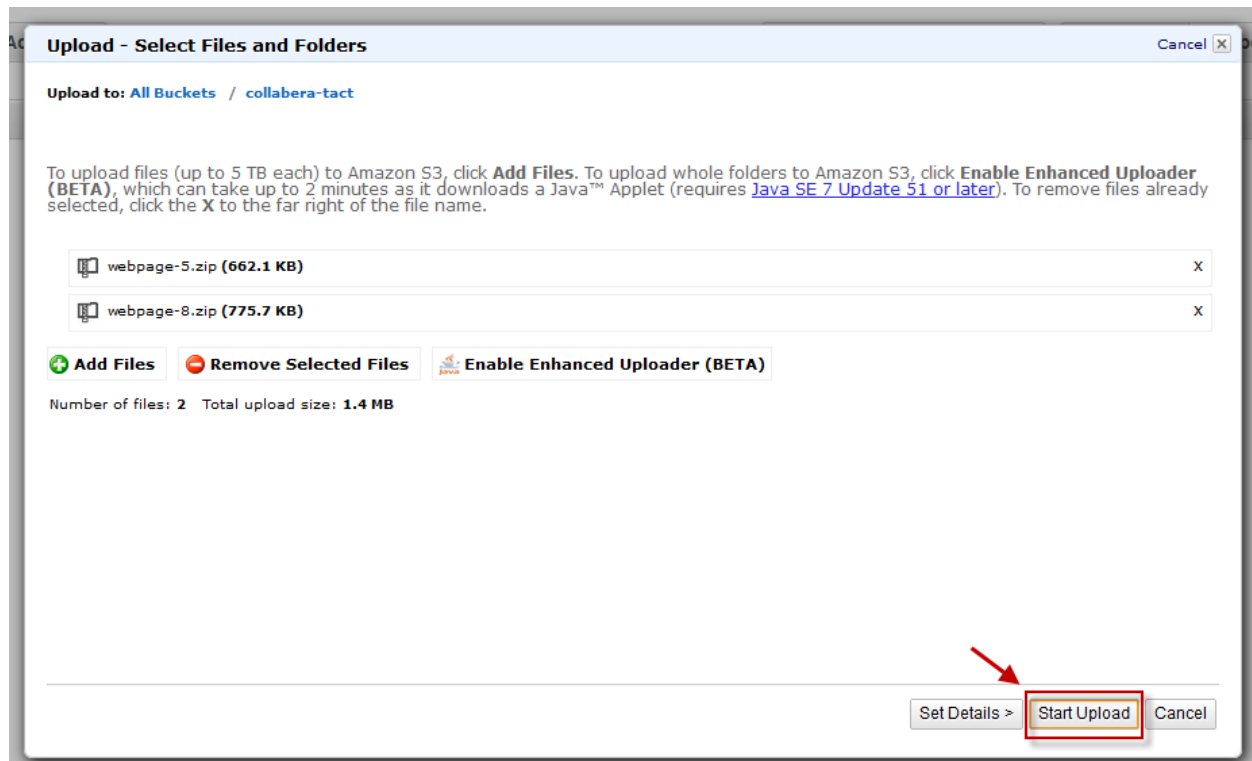
Once you are under the Bucket, choose Upload from the Actions drop down list.



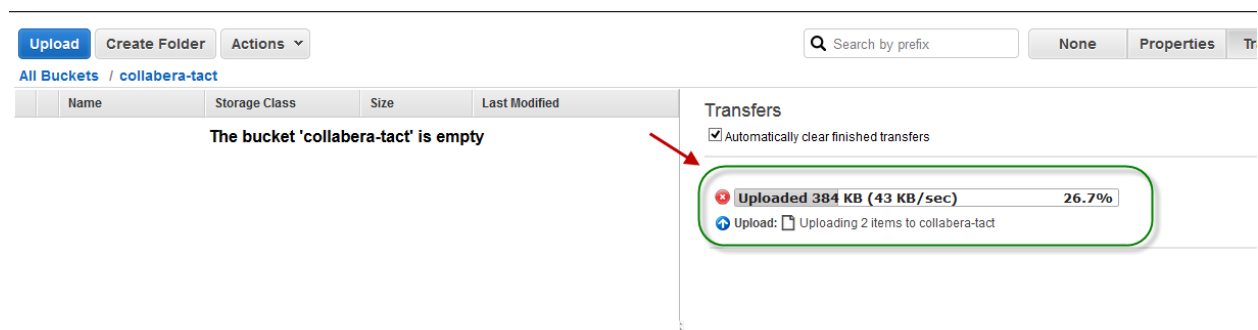
Click on Add files to upload files to S3 Bucket.



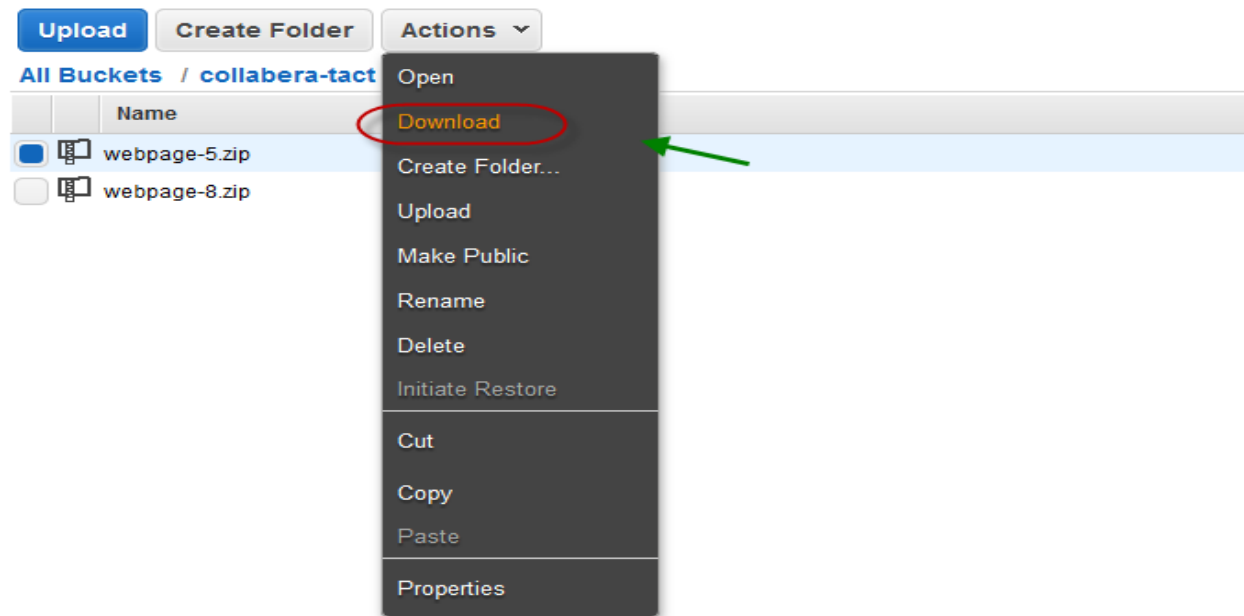
Select the files to upload and then select Start Upload button.



Files will start upload; you can see the upload status from the right pane.



To download the file, choose file and click Download from actions tab.



# HOSTING A WEBSITE ON S3

Once you created a bucket, create a html file with following code and name that file as index.html and upload to created bucket.

```
<html>
<body>
<h1 style="color:blue;">This is for testing S3</h1>
</body>
</html>
```

Once you uploaded, under All Buckets, click Properties for the bucket. Under Properties, expand Static Website Hosting, choose Enable website hosting and specify Index Document as index.html which was created and uploaded. Then choose save button.

The screenshot shows the AWS Management Console interface for a bucket named 'collabera-tact'. The 'Properties' tab is selected, and the 'Static Website Hosting' section is expanded. The 'Enable website hosting' radio button is selected, and the 'Index Document' is set to 'index.html'. The 'Save' button is highlighted.

**Create Bucket** Actions

All Buckets (2)

Name
collabera-tact
elasticbeanstalk-ap-southeast-1-168600309204

None **Properties** Transfers

Static Website Hosting

You can host your static website entirely on Amazon S3. Once you enable your bucket for static website hosting, all your content is accessible to web browsers via the Amazon S3 website endpoint for your bucket.

**Endpoint:** collabera-tact.s3-website-ap-southeast-1.amazonaws.com

Each bucket serves a website namespace (e.g. "www.example.com"). Requests for your host name (e.g. "example.com" or "www.example.com") can be routed to the contents in your bucket. You can also redirect requests to another host name (e.g. redirect "example.com" to "www.example.com"). See our [walkthrough](#) for how to set up an Amazon S3 static website with your host name.

☐ Do not enable website hosting

☒ **Enable website hosting**

**Index Document:** index.html

**Error Document:**

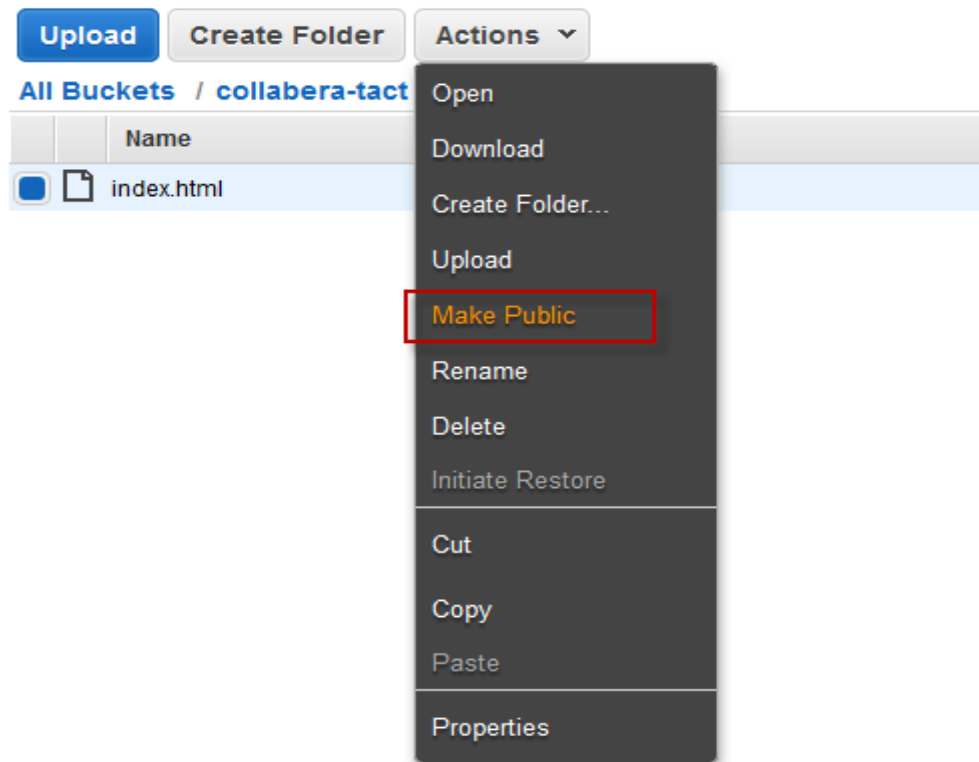
► **Edit Redirection Rules:** You can set custom rules to automatically redirect web page requests for specific content.

☐ Redirect all requests to another host name

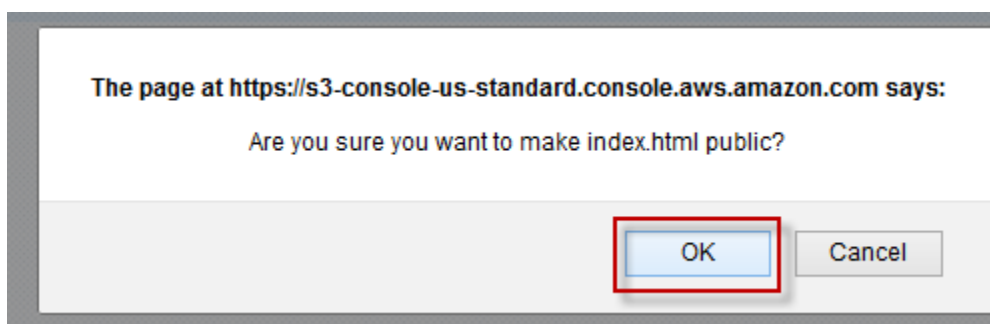
**Save** Cancel



Then go inside of the bucket, choose index.html file select Make Public under Actions tab.



A pop up window will open to confirm, choose OK to confirm.

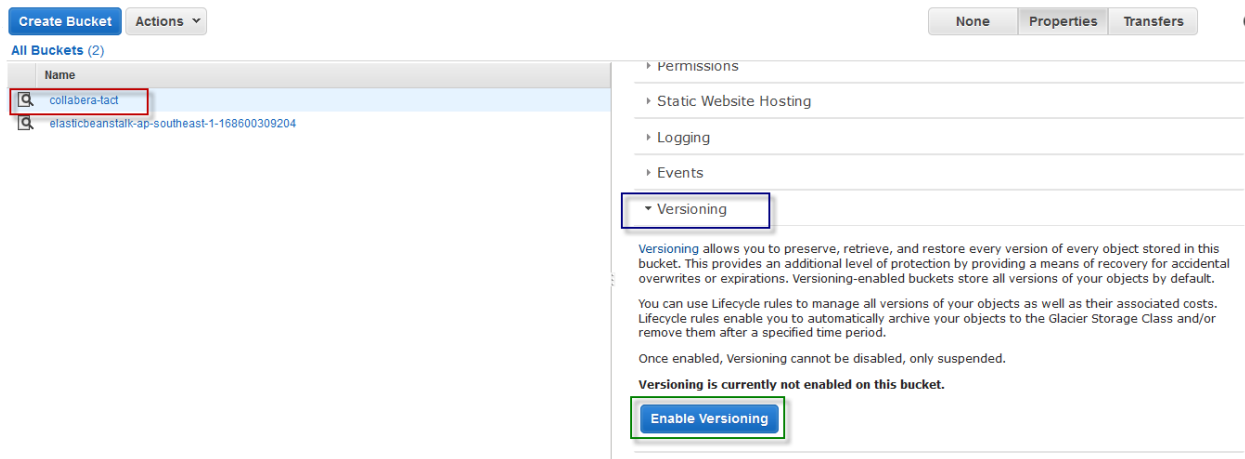


Once done, again go to the All buckets section, select your bucket, go to properties of your bucket, expand Static Website Hosting.

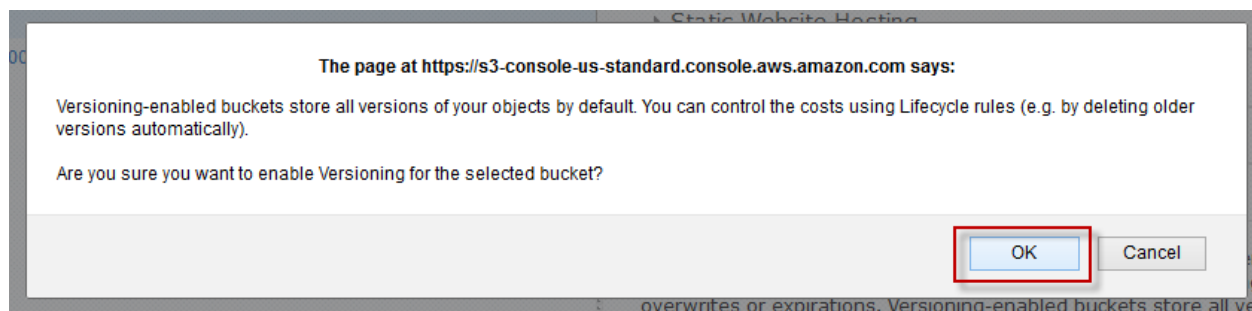
Copy or click on the link to open your static website hosted on S3.

# S3 OBJECT VERSIONING

Under S3, go to all Buckets section, choose Properties of the Bucket.  
Under Properties, expand Versioning, then choose Enable versioning to enable.



A pop up window will open to confirm, choose OK to confirm.



Add below code to existing index.html file and upload it to same bucket.

```
<html>
<body>
<h1 style="color:blue;">This is for testing S3 Versioning</h1>
<h2 style="color:green;">Hi Welcome to S3 Versioning</h2>
</body>
</html>
```

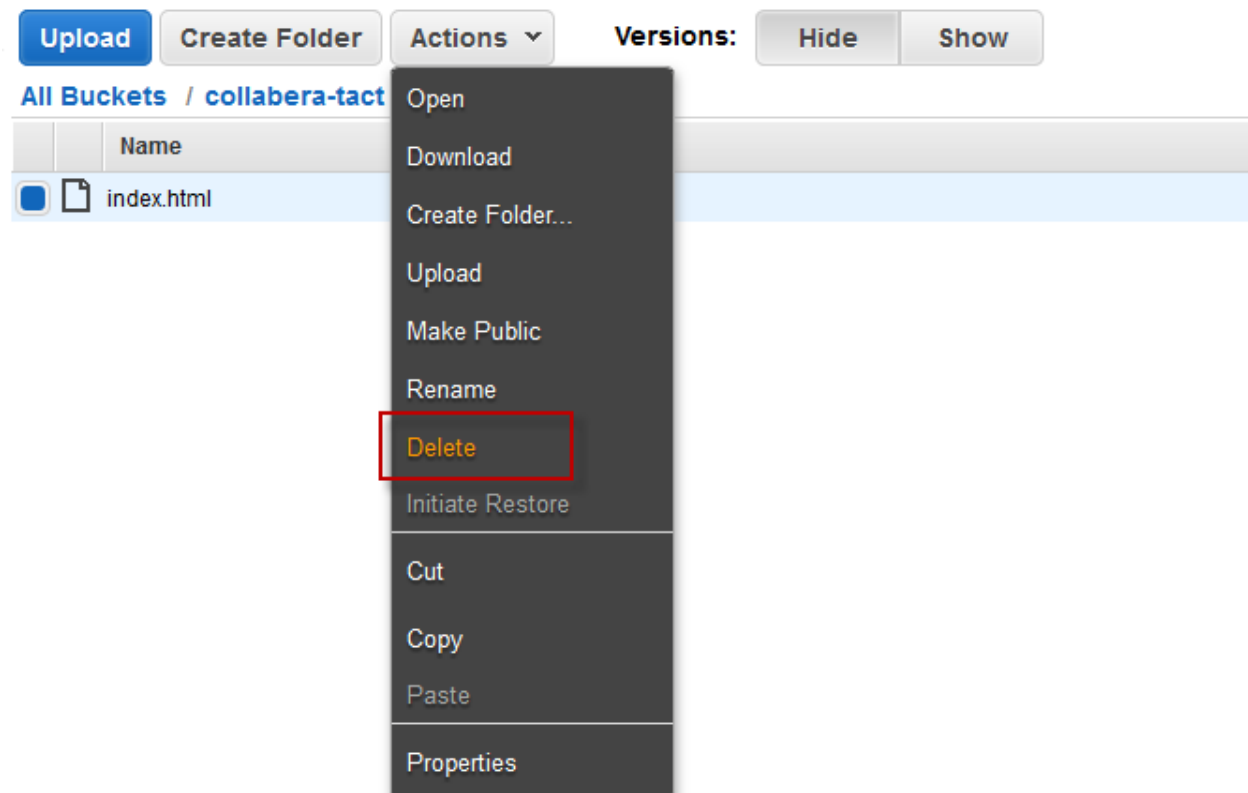
You can see those changes were reflected by opening the static website which we configured.

Once enabled versioning, you can see a Versions tab will display beside of the Actions tab.

Choose Show to see the versions of current file.



Choose index.html under the Bucket, click on Delete under Actions tab to remove the current version of the object.



Click on Show option for Versions to see the available versions.  
You can be able to see file has been deleted and added Delete marker to the file.

All Buckets / collabera-tact

	Name / Version Create Date	Storage Class
	index.html	—
<input type="checkbox"/>	Fri May 06 01:24:16 GMT+530 2016 (Delete Marker)	—
<input type="checkbox"/>	Fri May 06 01:15:35 GMT+530 2016	Standard
<input type="checkbox"/>	Fri May 06 01:09:15 GMT+530 2016	Standard

You choose the file which has delete marker and choose delete from the Actions tab.

All Buckets / collabera-tact

	Name / Version Create Date
	index.html
<input checked="" type="checkbox"/>	Fri May 06 01:24:16 GMT+530 2016 (Delete Marker)
<input type="checkbox"/>	Fri May 06 01:15:35 GMT+530 2016
<input type="checkbox"/>	Fri May 06 01:09:15 GMT+530 2016

Open

Create Folder...

Upload

**Delete**

Initiate Restore

Properties

Confirm deletion of delete marker.

The page at <https://s3-console-us-standard.console.aws.amazon.com> says:

Are you sure you want to permanently delete version 8q.cKPdV9Ta0yhduwiiME6rR1hbqQfCw of index.html?



You can be able to see the file has been restored back.

To suspend versioning on the Bucket, go to properties of the Bucket, expand Versioning, under versioning choose Suspend Versioning.

Create BucketActions ▾

NonePropertiesTransfers

All Buckets (2)

Name
 collabera-tact
 elasticbeanstalk-ap-southeast-1-168600309204

▸ Events

▾ Versioning

Versioning allows you to preserve, retrieve, and restore every version of every object stored in the bucket. This provides an additional level of protection by providing a means of recovery for accidental overwrites or deletions. Versioning-enabled buckets store all versions of your objects by default.

You can use Lifecycle rules to manage all versions of your objects as well as their associated costs. Lifecycle rules enable you to automatically archive your objects to the Glacier Storage Class and/or remove them after a specified time period.

Once enabled, Versioning cannot be disabled, only suspended.

**Versioning is currently enabled on this bucket.**

Suspend Versioning

# S3 LIFECYCLE POLICIES

Expand Lifecycle under bucket properties, then click on Add rule.

The screenshot shows the AWS S3 console interface. At the top, there are buttons for 'Create Bucket' and 'Actions'. Below these, a list of buckets is shown, with 'collabera-tact' selected. To the right, the 'Properties' tab is active, showing various bucket settings like 'Static Website Hosting', 'Logging', 'Events', 'Versioning', and 'Lifecycle'. The 'Lifecycle' option is expanded, revealing a description of lifecycle rules and a status message: 'Versioning is currently suspended on this bucket.' Below this, the 'Add rule' button is highlighted with an orange box. At the bottom right of the properties panel, there are 'Save' and 'Cancel' buttons.

Specify you want to create lifecycle for whole Bucket or for specific folder, then choose Configure Rule.

**Apply the Rule to:**

☒ Whole Bucket: collabera-tact

☐ A Prefix

Cancel

**Configure Rule >**

Now choose the options to how you want your objects to be moved over to glacier or Standard Infrequent Access storage or permanently deleted.  
Select the lifecycle policy and specify the days where object needs to be moved over or deleted.  
Then choose Review.

### Action on Objects

☐ Transition to the Standard - Infrequent Access Storage Class
 

Days after the object's creation date

Standard - Infrequent Access has a 30-day minimum retention period and a 128KB minimum object size. Lifecycle policy will not transition objects that are less than 128KB. [Refer here to learn more about Standard - Infrequent Access.](#)

☒ Archive to the Glacier Storage Class
 

Days after the object's creation date

This rule could reduce your storage costs. [Refer here to learn more about Glacier pricing.](#) Note that objects archived to the Glacier Storage Class are **not immediately accessible**.

☒ Permanently Delete
 

Days after the object's creation date

**EXAMPLE:**

[Cancel](#)
[< Set Target](#)
[Review >](#)

Then specify a rule name under Rule Name text field and choose Create and Activate Rule.

### Rule Name

Choose a descriptive name for your rule so you can easily identify it in the future. If you do not want to enter a name now, we will generate one for you.

Rule Name:  (Optional)

### Rule Target

This rule will apply to the whole bucket: **collaberatact** [Edit](#)

### Rule Configuration

[Edit](#)

#### Action on Objects

Archive to the Glacier Storage Class **30** days after the object's creation date.

This rule could reduce your storage costs. [Refer here to learn more about Glacier pricing.](#) Note that objects archived to the Glacier Storage Class are **not immediately accessible**.

#### Permanently Delete



**365** days after the object's creation date

As versioning is not enabled, lifecycle delete rule will permanently delete the objects with no recovery.

[Cancel](#)
[< Configure Rule](#)
[Create and Activate Rule](#)

Under lifecycle policies choose cross mark against the rule and click save to delete the rule which you created.

**All Buckets (2)**


Name
 collaberaact
 elasticbeanstalk-ap-southeast-1-168600309204


▼ Lifecycle



You can manage the lifecycle of objects by using [Lifecycle rules](#). Lifecycle rules enable you to automatically transition objects to the [Standard - Infrequent Access](#) Storage Class, and/or archive objects to the [Glacier](#) Storage Class, and/or remove objects after a specified time period. Rules are applied to all the objects that share the specified prefix.

**Versioning is not currently enabled on this bucket.**

You can use Lifecycle rules to manage all versions of your objects. This includes both the Current version and Previous versions.

Enabled	Name	Rule Target	
<input checked="" type="checkbox"/>	collaber-deletion	Whole Bucket	

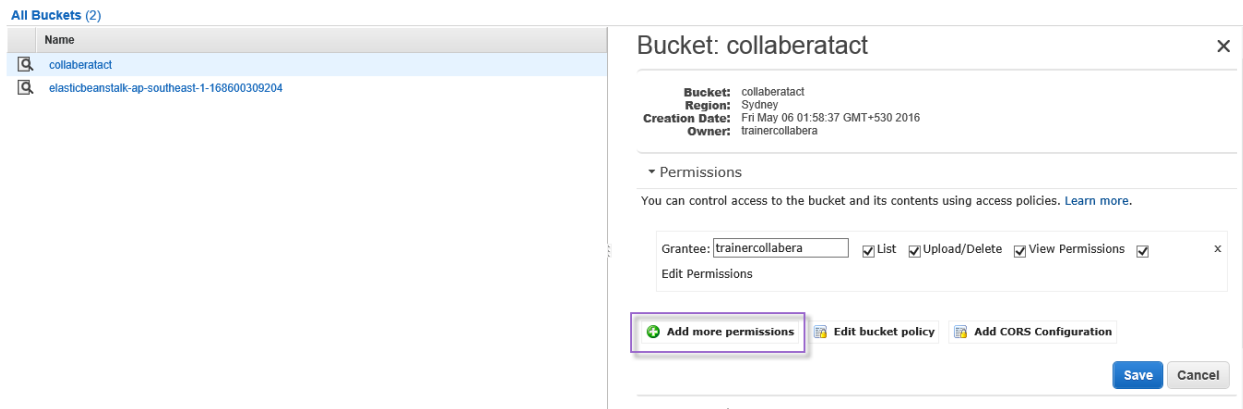
 Add rule

 Save  Cancel



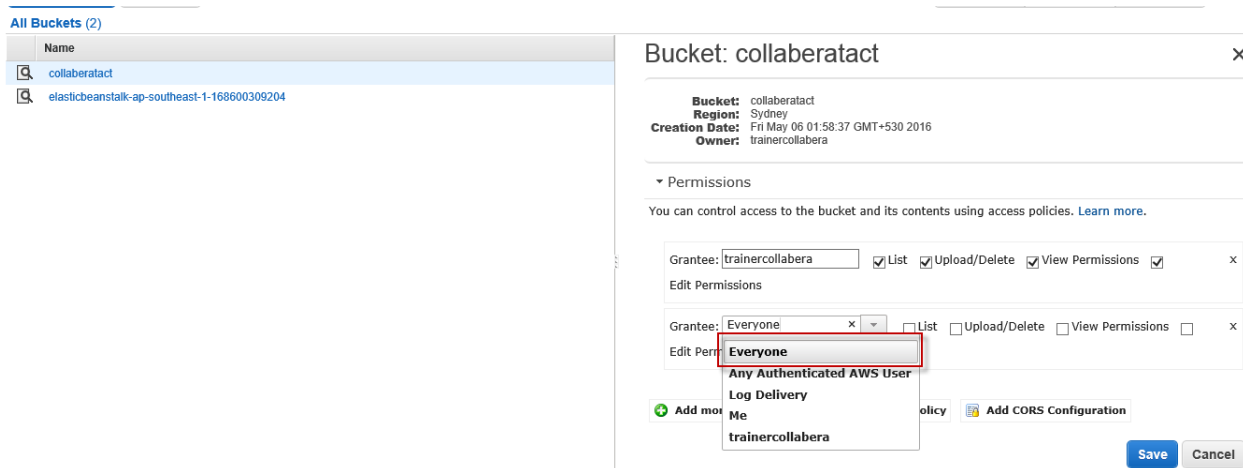
## S3 PERMISSIONS AND BUCKET POLICIES

Go to properties of your bucket and under properties expand permissions. Then choose Add more permissions.



The screenshot shows the AWS S3 console. On the left, under 'All Buckets (2)', the bucket 'collabertact' is selected. The main panel shows the bucket's properties: Bucket: collabertact, Region: Sydney, Creation Date: Fri May 06 01:58:37 GMT+530 2016, and Owner: trainercollabera. The 'Permissions' section is expanded, showing a table with one entry for 'trainercollabera' with permissions for List, Upload/Delete, and View Permissions. Below the table, the 'Add more permissions' button is highlighted with a red box. Other buttons visible are 'Edit bucket policy' and 'Add CORS Configuration'. At the bottom right are 'Save' and 'Cancel' buttons.

Under Grantee choose option for your permission.



This screenshot shows the 'Add more permissions' dropdown menu that appears after clicking the 'Add more permissions' button. The dropdown lists several options: 'Everyone' (highlighted with a red box), 'Any Authenticated AWS User', 'Log Delivery', 'Me', and 'trainercollabera'. The 'Everyone' option is selected, and its permissions (List, Upload/Delete, View Permissions) are shown as unchecked in the table below. The 'Save' and 'Cancel' buttons remain at the bottom right.

Then check permissions which you want to give for particular grantee, and choose save to apply the policy permission.

Bucket: collaberatact

**Bucket:** collaberatact  
**Region:** Sydney  
**Creation Date:** Fri May 06 01:58:37 GMT+530 2016  
**Owner:** trainercollabera

▼ Permissions

You can control access to the bucket and its contents using access policies. [Learn more.](#)

Grantee: trainercollabera ☒ List ☒ Upload/Delete ☒ View Permissions ☒ x

Edit Permissions

Grantee: Any Authenticated AWS ☒ List ☒ Upload/Delete ☐ View Permissions ☐ x

Edit Permissions

[Add more permissions](#) [Edit bucket policy](#) [Add CORS Configuration](#)

**Save** Cancel

## S3 BUCKET POLICIES:

Go to properties of your bucket and under properties expand permissions, then choose **Add bucket policy** to add a new policy to bucket.

Bucket: collaberatact

**Bucket:** collaberatact  
**Region:** Sydney  
**Creation Date:** Fri May 06 01:58:37 GMT+530 2016  
**Owner:** trainercollabera

▼ Permissions

You can control access to the bucket and its contents using access policies. [Learn more.](#)

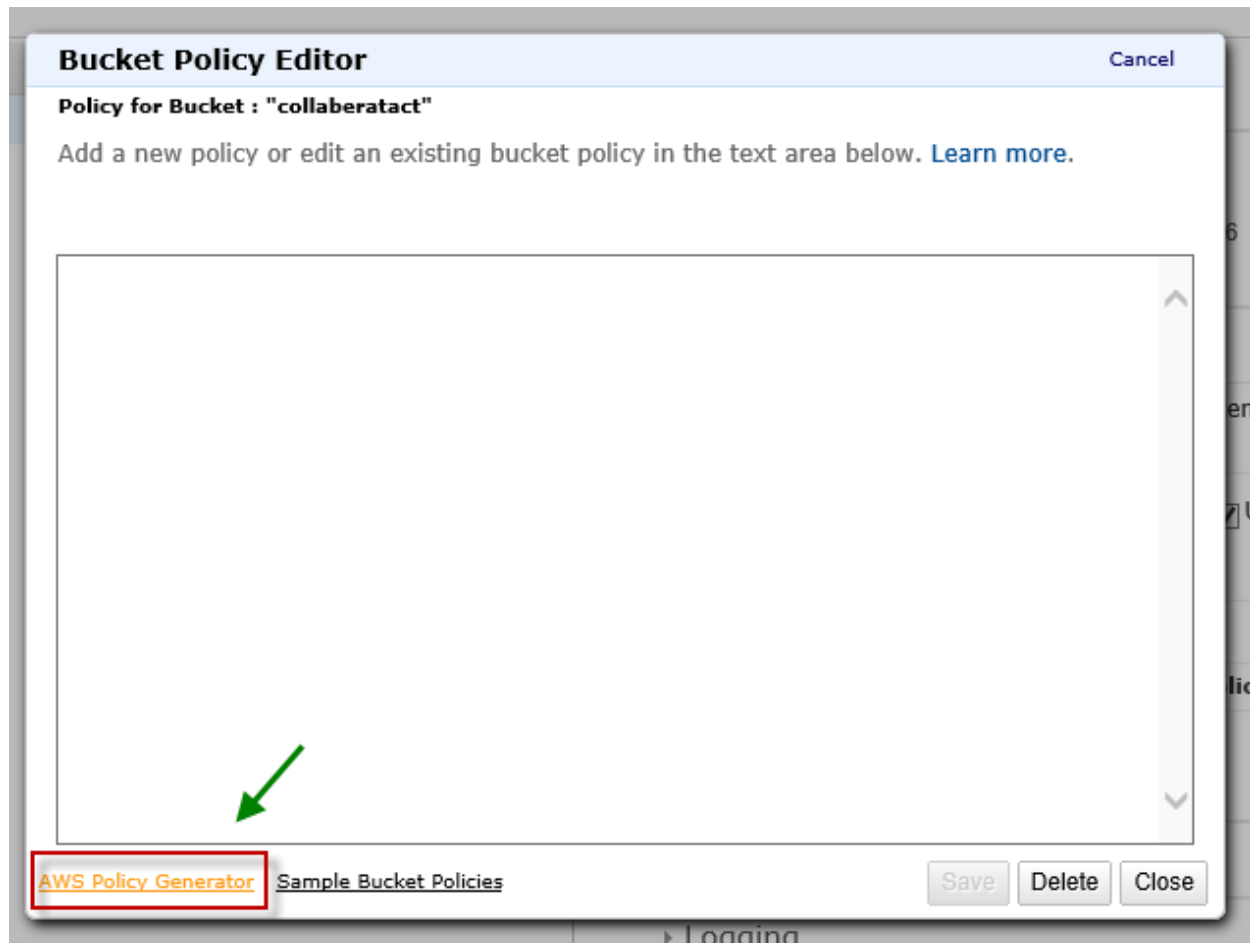
Grantee: trainercollabera ☒ List ☒ Upload/Delete ☒ View Permissions ☒ x

Edit Permissions

[Add more permissions](#) [Add bucket policy](#) [Add CORS Configuration](#)

**Save** Cancel

Bucket policy Editor text will be opened, then choose AWS Policy Generator.



From the AWS policy generator page, choose various options to generate policy.

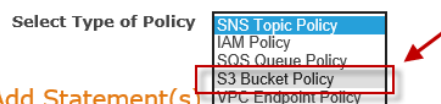
Choose S3 Bucket Policy from Select type of policy.

### AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to [Amazon Web Services \(AWS\)](#) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#). You can [submit your samples](#) (Enter 'AWS Policy Examples' in the Library Title field).

#### 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](#) and an [SQS Queue 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

Choose Allow or Deny from Effect, then specify the AWS IAM user ARN in Principal text box.

### Step 2: Add Statement(s)

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

Effect ☒ Allow ☐ Deny

Principal

Use a comma to separate multiple values.

AWS Service  ☐ All Services (\*\*)

From the Actions drop down list either choose All Actions or choose only specific options to give.

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

Actions  ☐ All Actions (\*)

Amazon Resource Name (ARN)

arn:aws:s3:::<bucket\_name>/<key\_name>.

valid. You must enter a valid ARN.

☐ AbortMultipartUpload  
☐ CreateBucket  
☐ DeleteBucket  
☐ DeleteBucketPolicy  
☐ DeleteBucketWebsite  
☒ DeleteObject  
☐ DeleteObjectVersion  
☐ GetAcceleratedConfiguration

### Step 3: Generate Policy

Under Amazon Resource Name (ARN) specify the S3 bucket ARN in the following format.

arn:aws:s3:::<bucket\_name>/<key\_name>

Once you specified choose Add statement button.

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

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.

[Add Conditions \(Optional\)](#)

Then choose Generate Policy to generate S3 bucket policy.

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

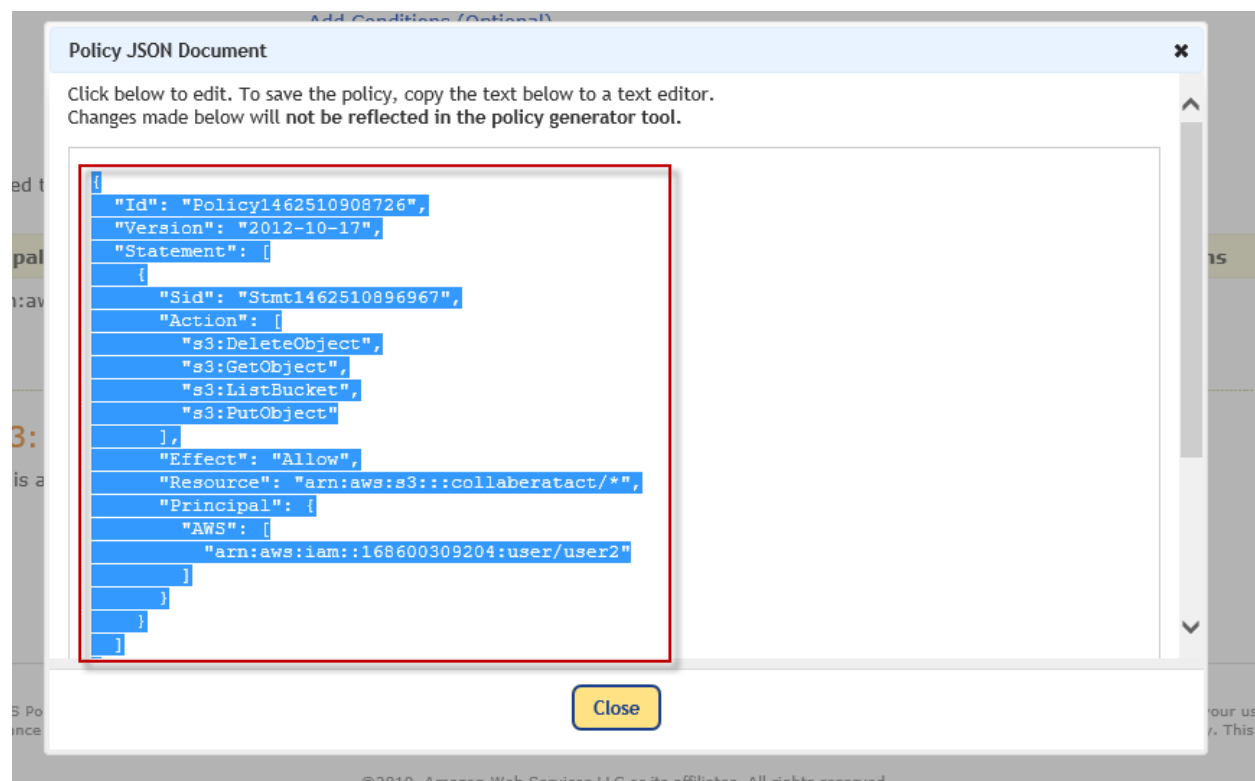
Principal(s)	Effect	Action	Resource	Conditions
• arn:aws:iam::168600309204:user/user2	Allow	<ul style="list-style-type: none"> <li>s3:DeleteObject</li> <li>s3:GetObject</li> <li>s3:ListBucket</li> <li>s3:PutObject</li> </ul>	arn:aws:s3:::collabertact/*	None

### 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.



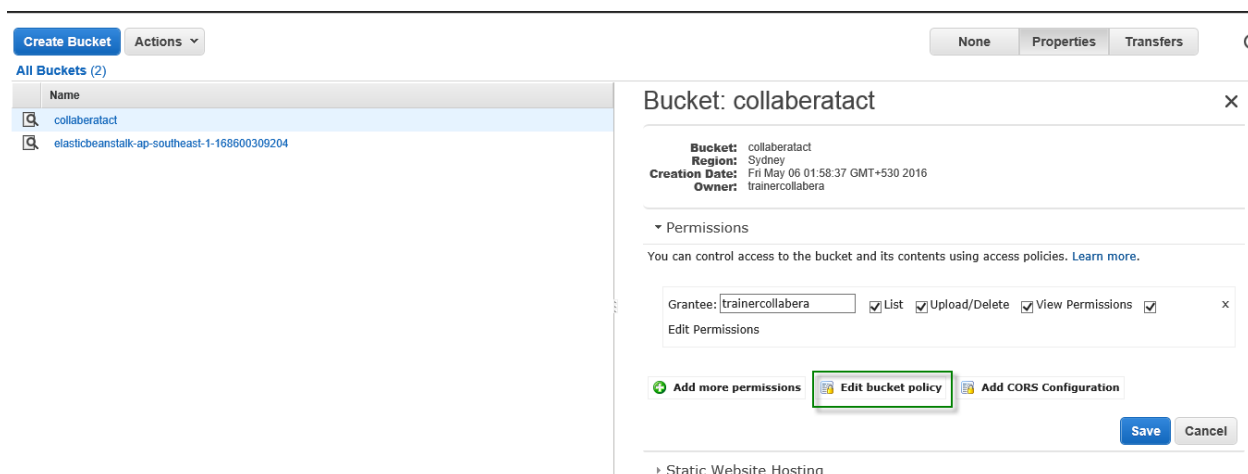
Then a pop up window will open with JSON format code, copy the code and paste it in the AWS bucket policy page where we choose AWS policy generator option.



Once pasted, select Save to save and apply the policy.



To edit the policy, choose edit bucket policy option under Permissions of the bucket properties.



Existing policy will open, either you can modify code or choose AWS policy generator to create a new policy then choose save to apply policies.

Create BucketActions

All Buckets (2)

Name
collabertact
elasticbeanstalk-ap-southeast-1-168600309204

Bucket Policy Editor

Cancel

Policy for Bucket : "collabertact"

Add a new policy or edit an existing bucket policy in the text area below. [Learn more.](#)

```
{
  "Id": "Policy1462510908726",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1462510896967",
      "Action": [
        "s3:DeleteObject",
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3::collabertact/*",
      "Principal": {
        "AWS": [
          "arn:aws:iam::168600309204:user/user2"
        ]
      }
    }
  ]
}
```

[AWS Policy Generator](#) | [Sample Bucket Policies](#)

SaveDeleteClose