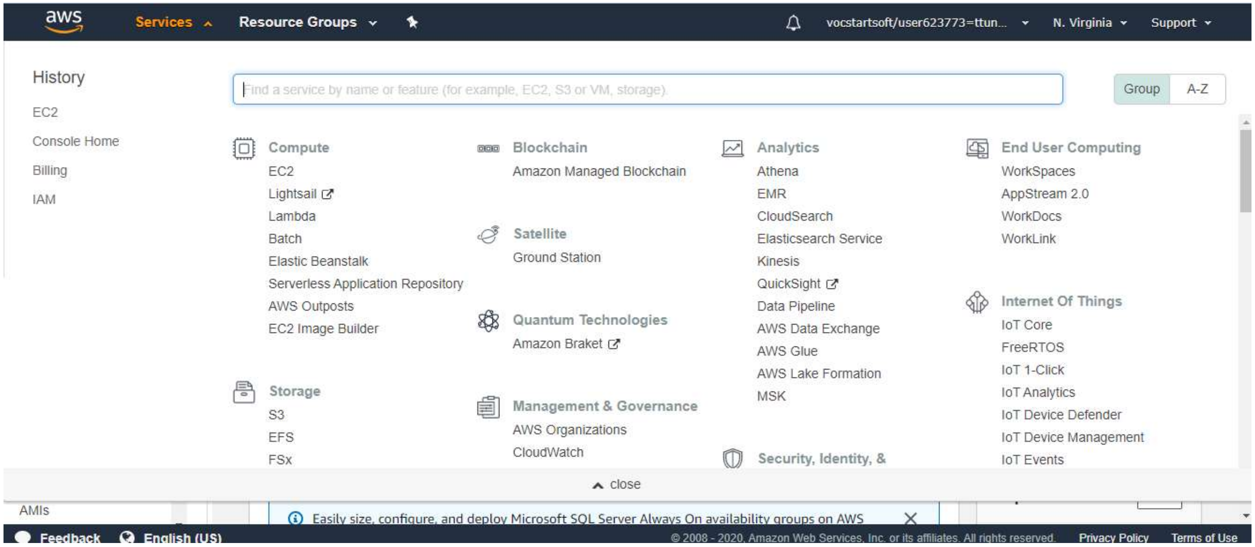
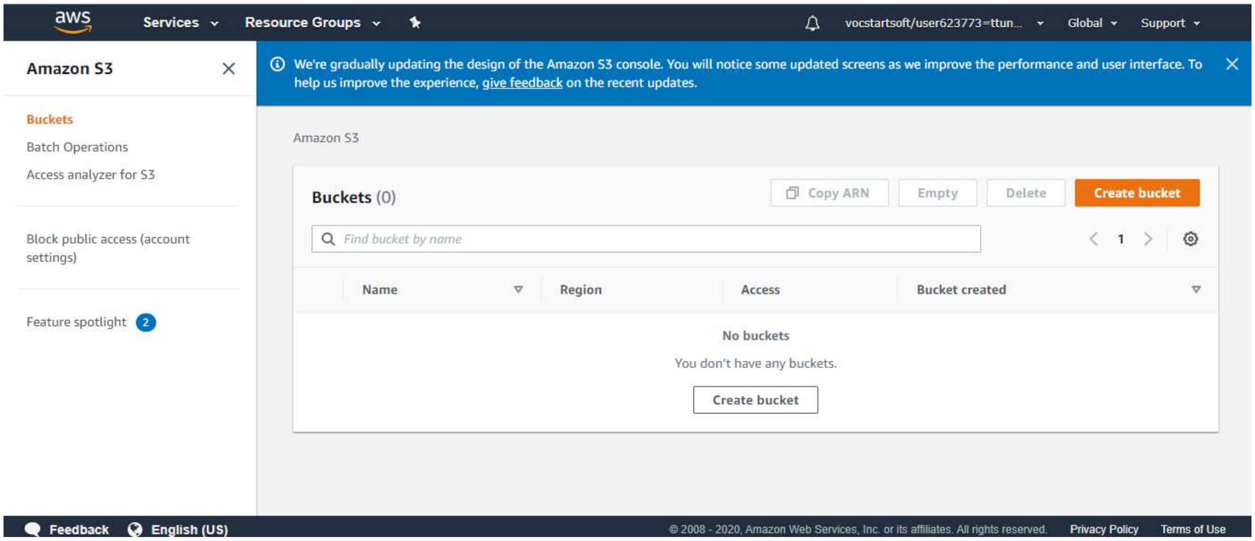
Lab Assignment # 3

1) Create an S3 Bucket:

Step 1: Login to Amazon AWS account and go to EC dashboard. Select S3.



On the next page, Click on Create Bucket:

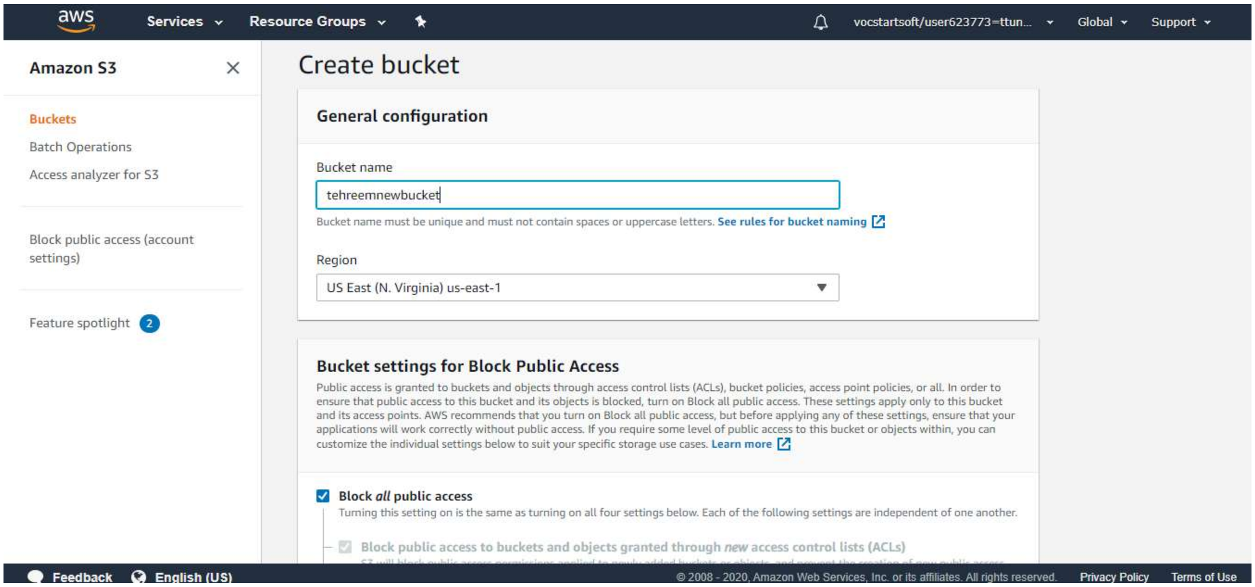


We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.

No buckets

You don't have any buckets.

Give the bucket a unique name:



Create bucket

General configuration

Bucket name: tehreemnewbucket

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

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

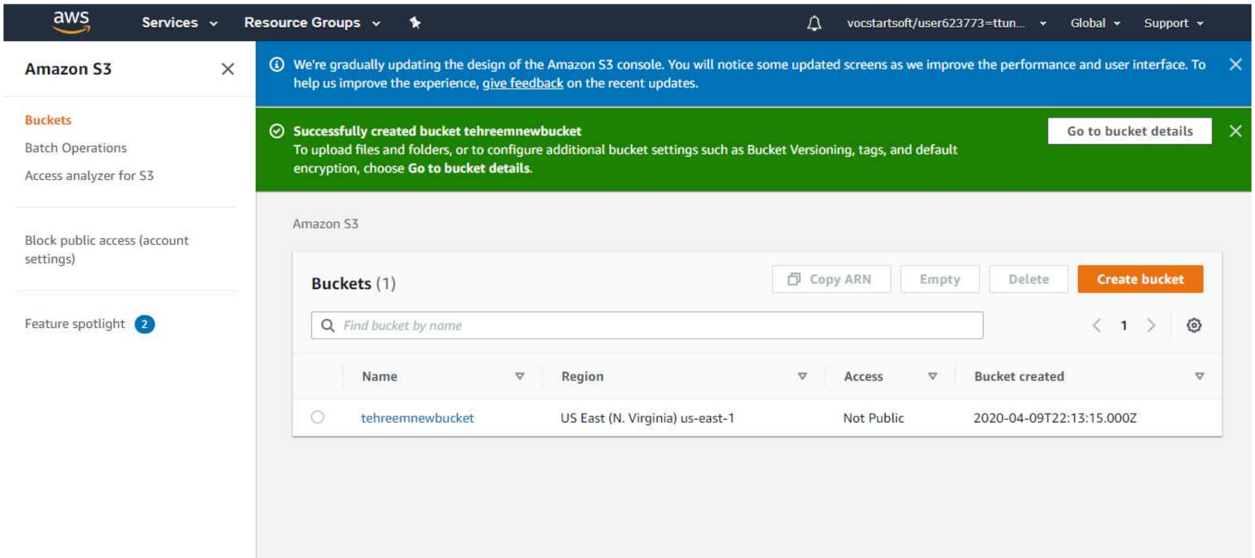
Bucket settings for Block Public Access

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)

By default, all public accesses to this bucket are blocked. Click on Create Bucket.



We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.

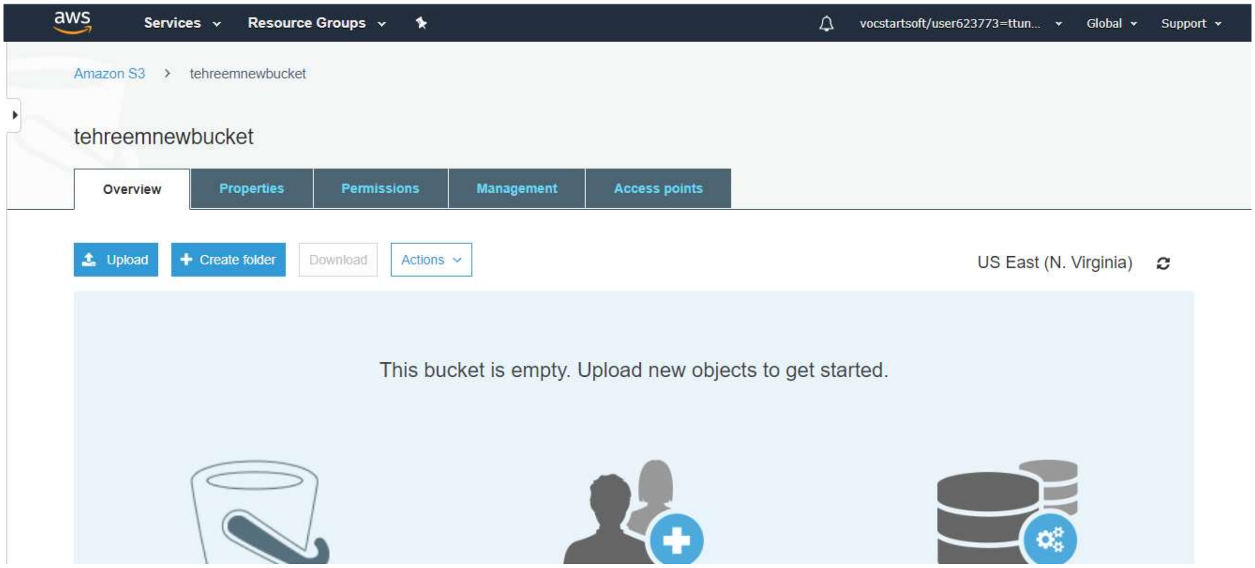
Successfully created bucket tehreemnewbucket
To upload files and folders, or to configure additional bucket settings such as Bucket Versioning, tags, and default encryption, choose **Go to bucket details.**

Amazon S3

Buckets (1)

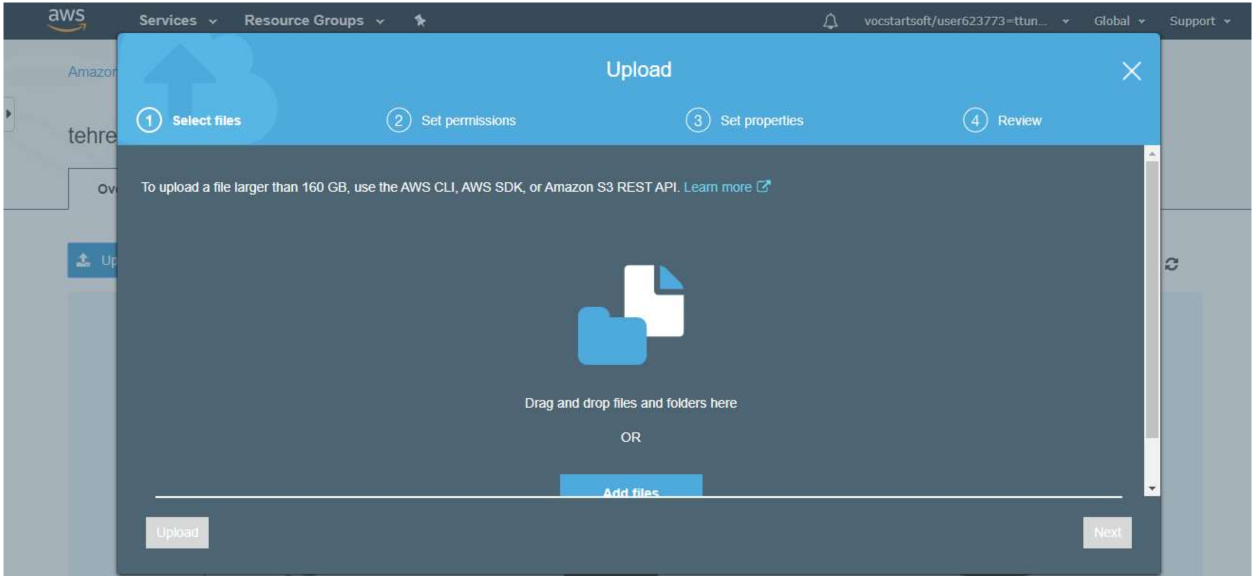
	Name	Region	Access	Bucket created
○	tehreemnewbucket	US East (N. Virginia) us-east-1	Not Public	2020-04-09T22:13:15.000Z

Step 2: Uploading an image to this newly created bucket:
Click on your bucket name as shown on the previous screen.



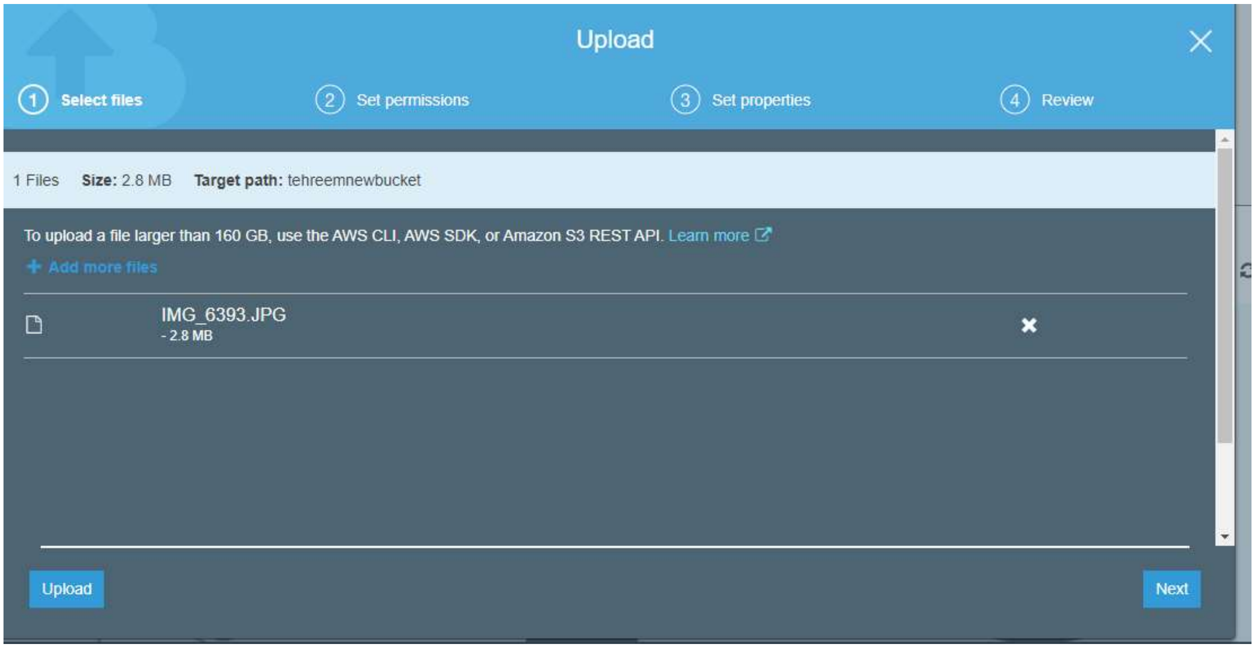
This bucket is empty. Upload new objects to get started.

Click on Upload button:



To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more

Click on Add Files. Select the image file you want to upload.



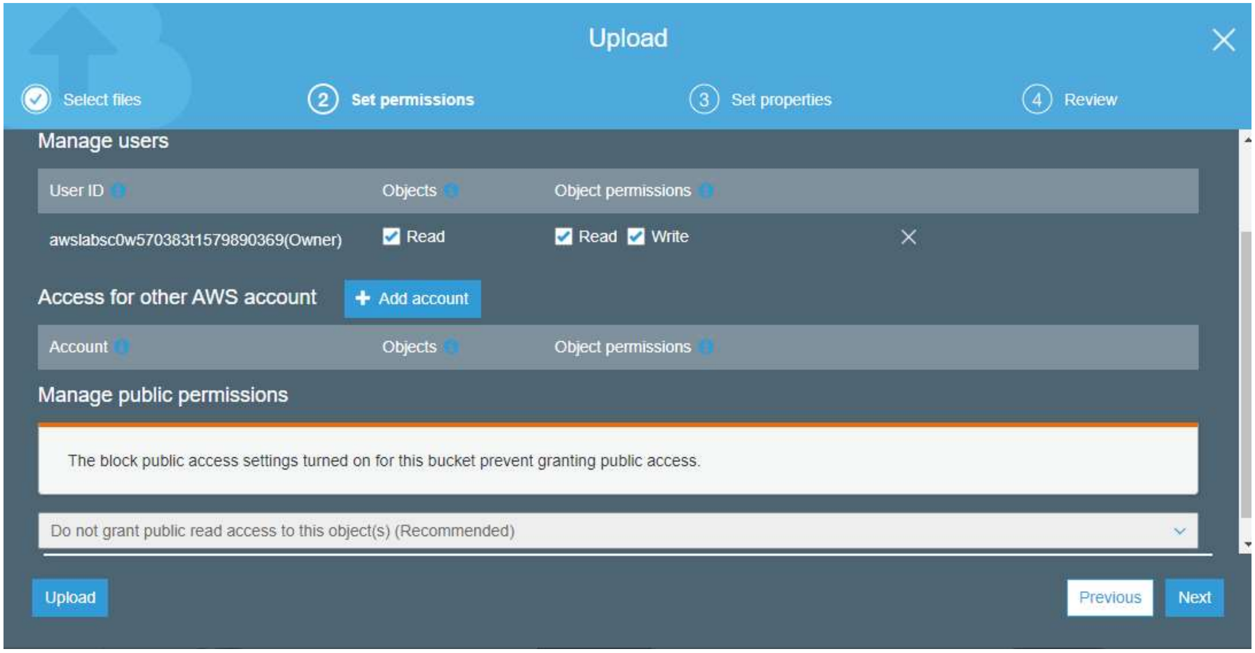
1 Files **Size:** 2.8 MB **Target path:** tehreemnewbucket

To upload a file larger than 160 GB, use the AWS CLI, AWS SDK, or Amazon S3 REST API. Learn more

IMG_6393.JPG
- 2.8 MB

Click on Next.

In order to make it publicly accessible,



Manage users

User ID	Objects	Object permissions
awslabsc0w570383t1579890369(Owner)	☑ Read	☑ Read ☑ Write

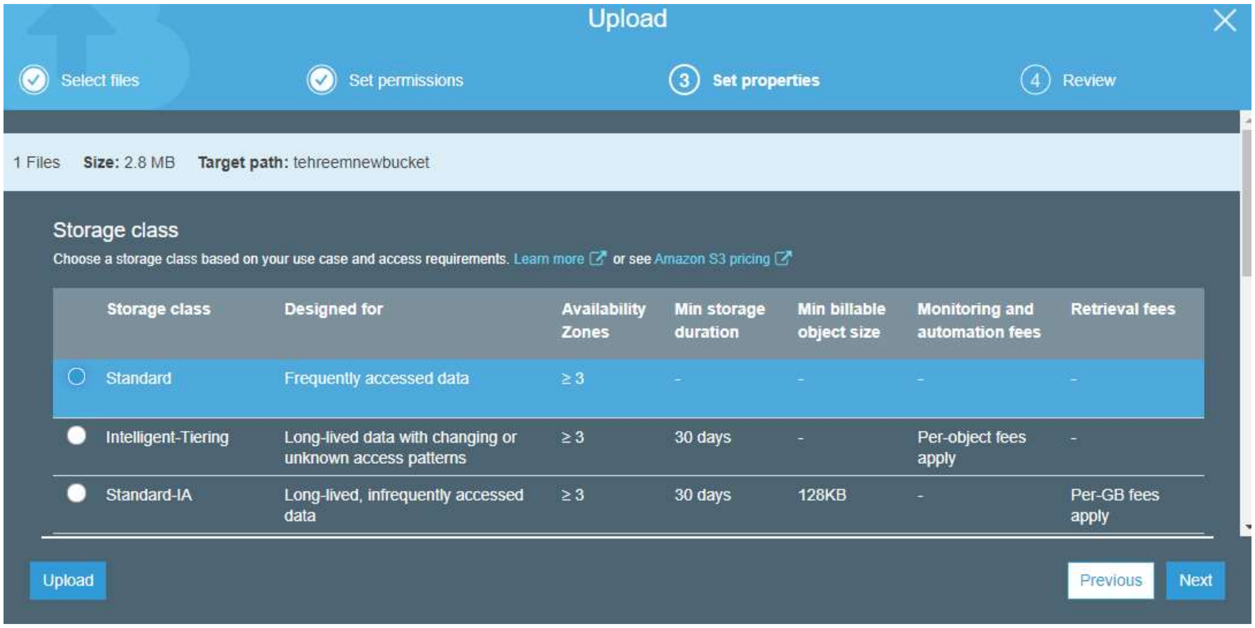
Access for other AWS account

Account	Objects	Object permissions

Manage public permissions

The block public access settings turned on for this bucket prevent granting public access.

Do not grant public read access to this object(s) (Recommended)



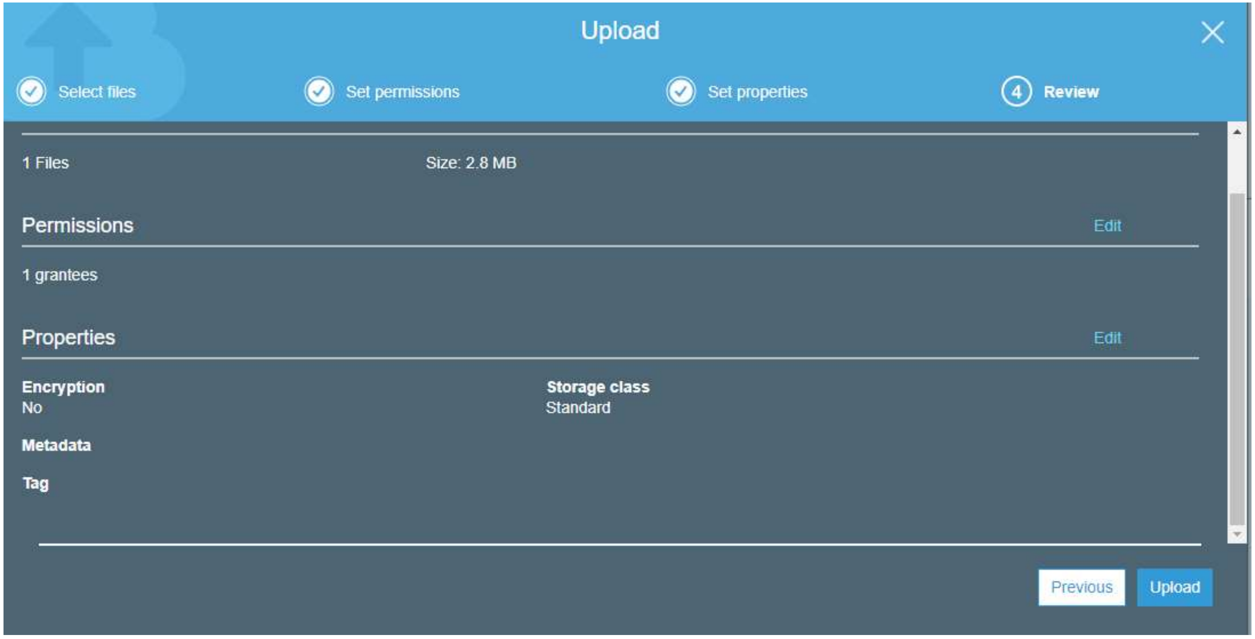
1 Files **Size:** 2.8 MB **Target path:** tehreemnewbucket

Storage class

Choose a storage class based on your use case and access requirements. Learn more or see Amazon S3 pricing

	Storage class	Designed for	Availability Zones	Min storage duration	Min billable object size	Monitoring and automation fees	Retrieval fees
○	Standard	Frequently accessed data	≥ 3	-	-	-	-
○	Intelligent-Tiering	Long-lived data with changing or unknown access patterns	≥ 3	30 days	-	Per-object fees apply	-
○	Standard-IA	Long-lived, infrequently accessed data	≥ 3	30 days	128KB	-	Per-GB fees apply

As of now only I am the one who can see the image.



1 Files Size: 2.8 MB

Permissions

1 grantees

Properties

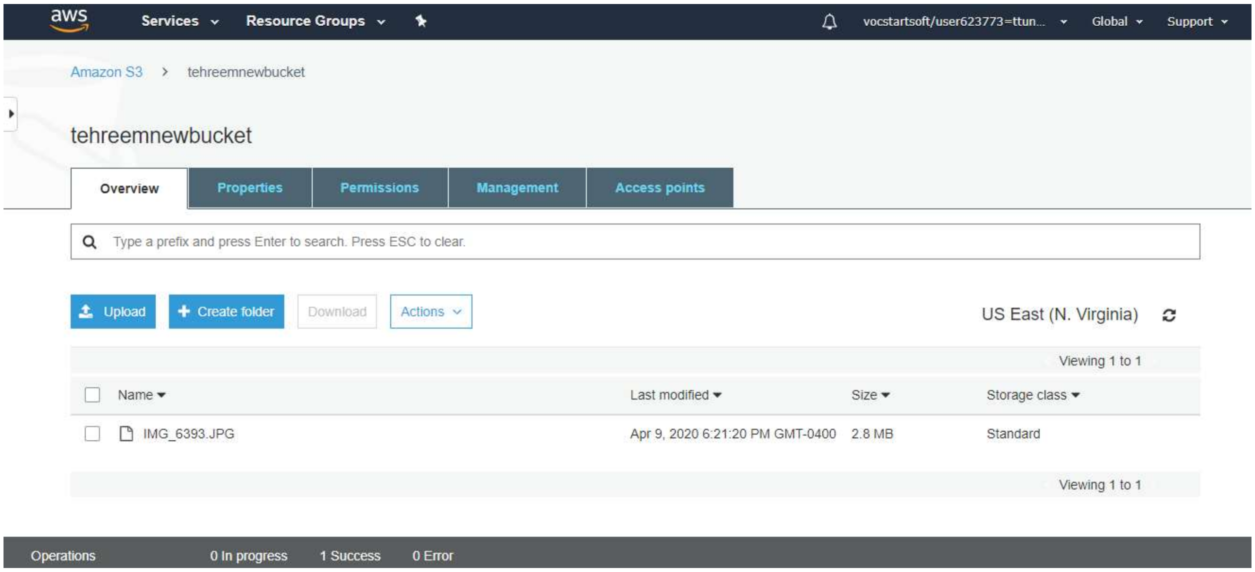
Encryption
No

Storage class
Standard

Metadata

Tag

Click Upload.



Amazon S3 > tehreemnewbucket

tehreemnewbucket

Type a prefix and press Enter to search. Press ESC to clear.

US East (N. Virginia)

Viewing 1 to 1

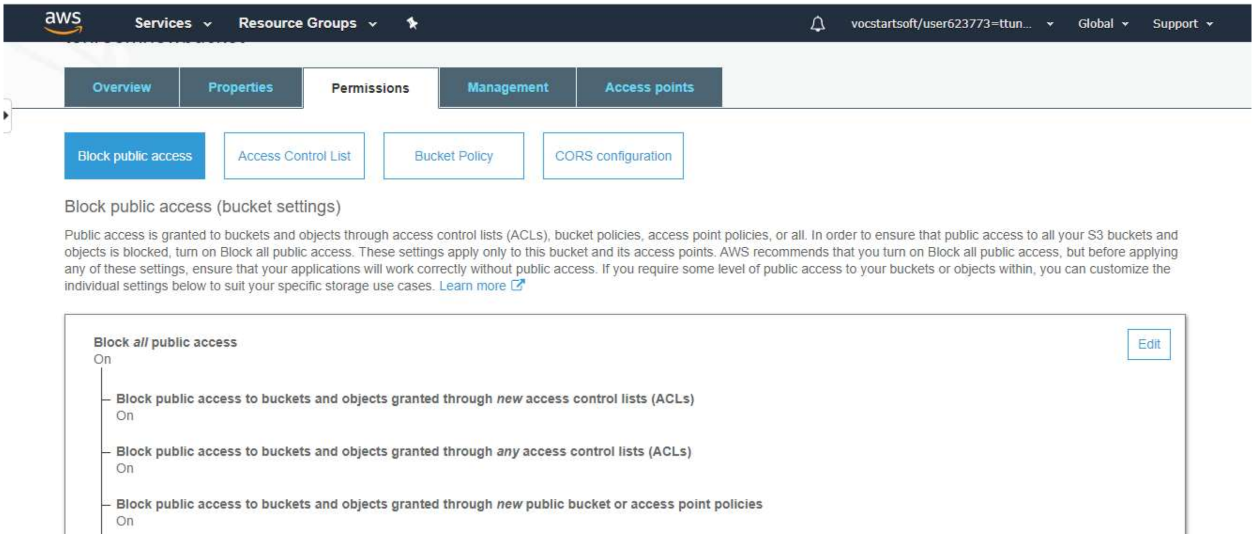
Name	Last modified	Size	Storage class
IMG_6393.JPG	Apr 9, 2020 6:21:20 PM GMT-0400	2.8 MB	Standard

Viewing 1 to 1

Operations 0 In progress 1 Success 0 Error

Image has been uploaded successfully.

Click on Permissions tab.



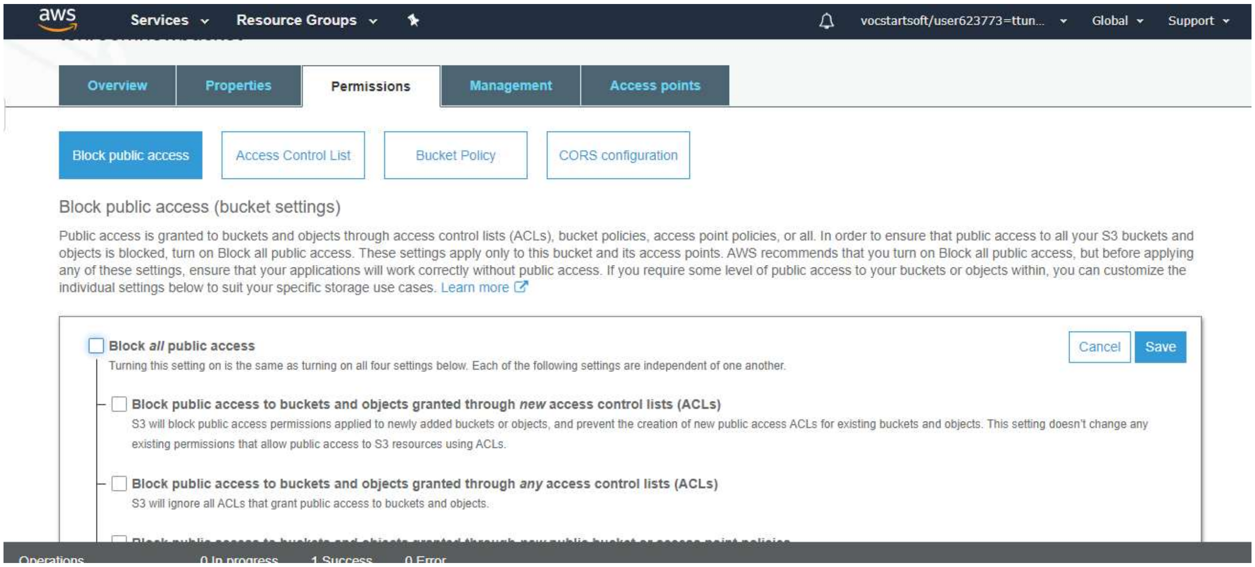
Block public access (bucket settings)

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
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

Click Edit and uncheck the Block all public access checkbox:



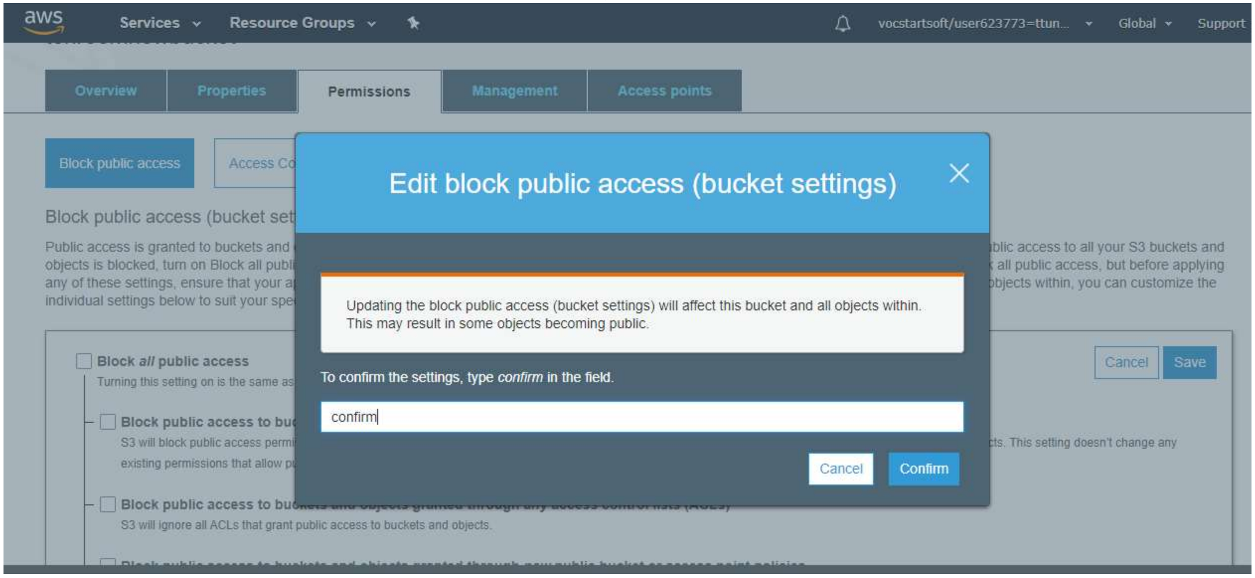
Block public access (bucket settings)

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**

You will be asked to confirm this action when you click on Save.

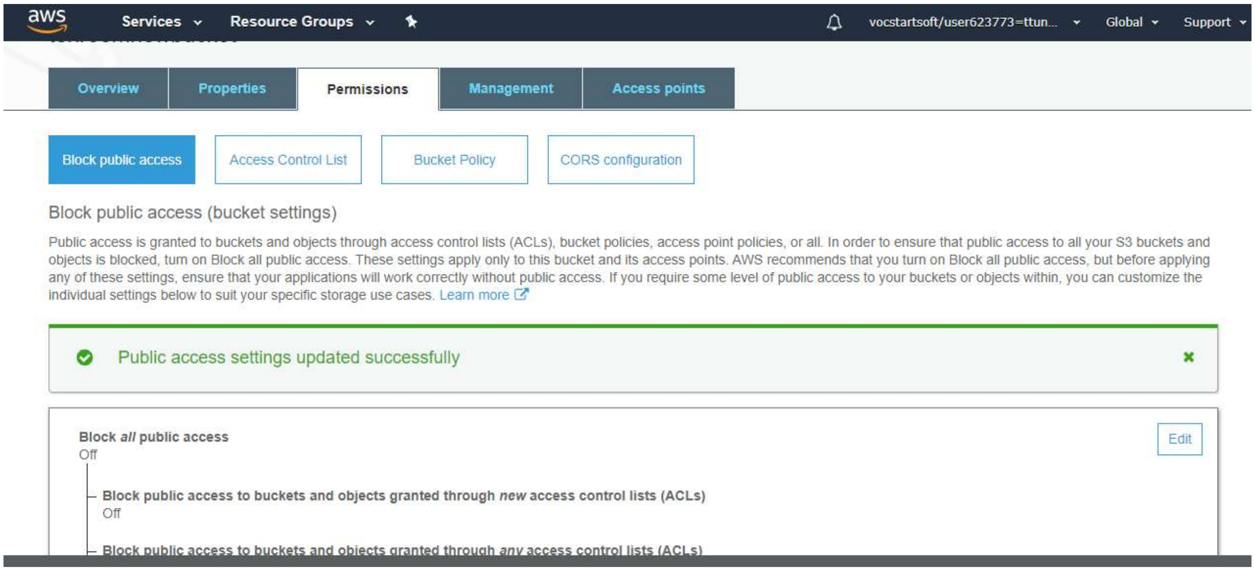


Edit block public access (bucket settings)

Updating the block public access (bucket settings) will affect this bucket and all objects within. This may result in some objects becoming public.

To confirm the settings, type *confirm* in the field.

confirm



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

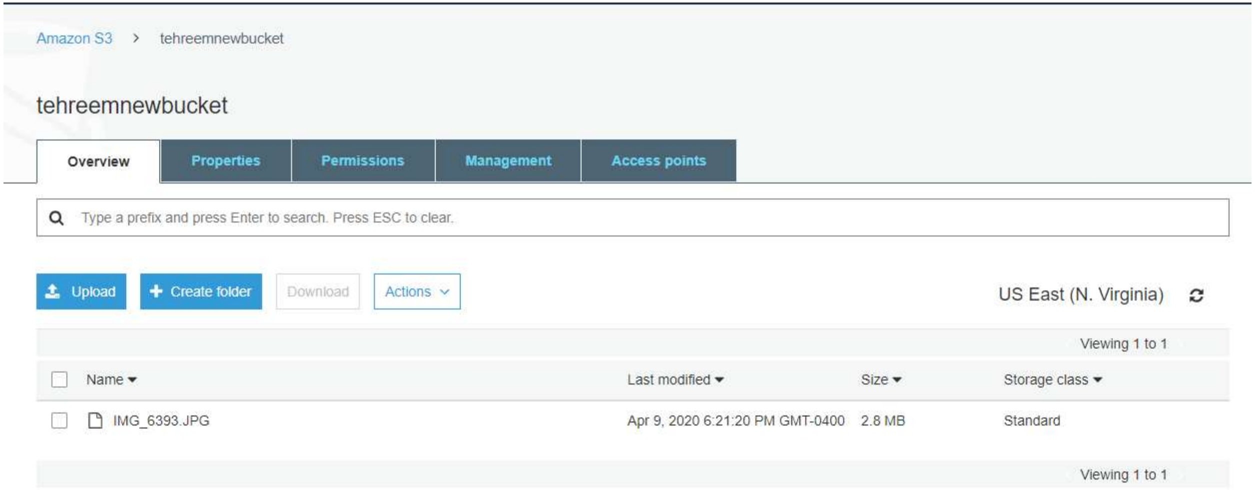
Public access settings updated successfully

Block *all* public access
Off

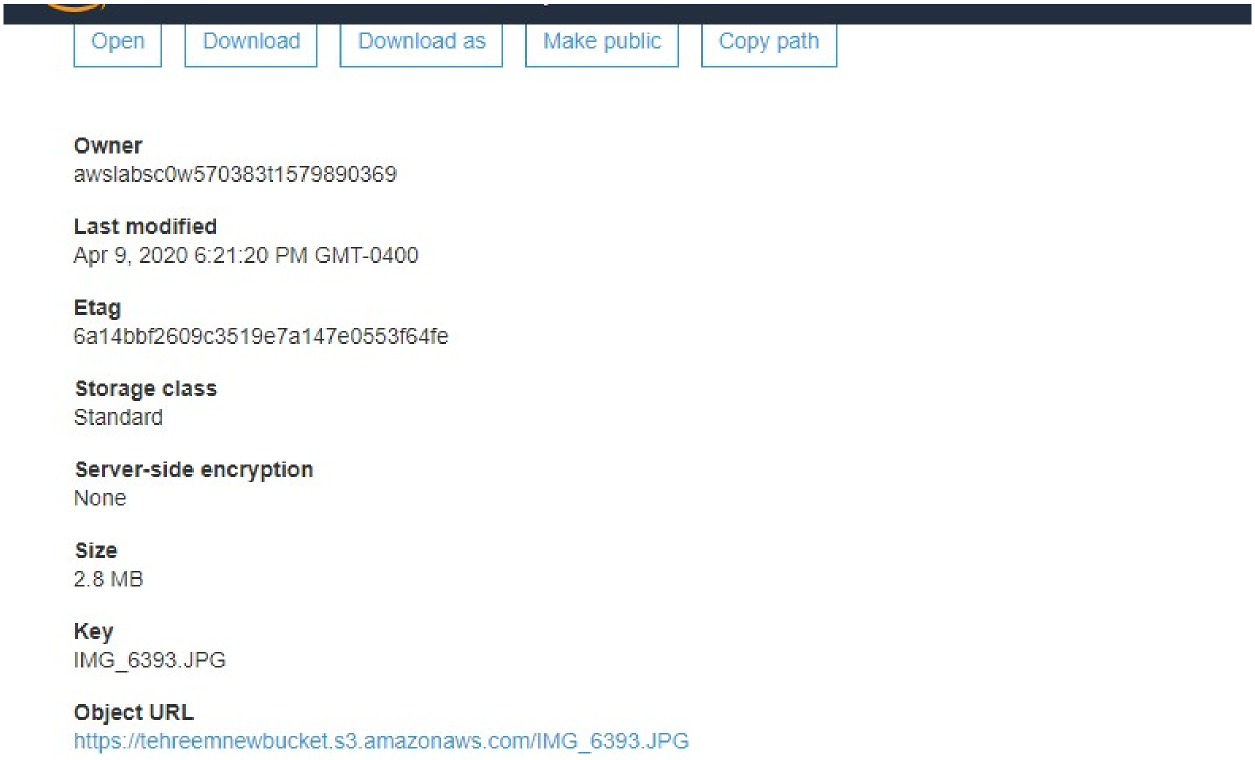
Block public access to buckets and objects granted through *new* access control lists (ACLs)
Off

Block public access to buckets and objects granted through *any* access control lists (ACLs)

Go to Overview tab and click on the image uploded.



□	Name	Last modified	Size	Storage class
□	IMG_6393.JPG	Apr 9, 2020 6:21:20 PM GMT-0400	2.8 MB	Standard



Owner
awslabsc0w570383t1579890369

Last modified
Apr 9, 2020 6:21:20 PM GMT-0400

Etag
6a14bbf2609c3519e7a147e0553f64fe

Storage class
Standard

Server-side encryption
None

Size
2.8 MB

Key
IMG_6393.JPG

Object URL
https://tehreemnewbucket.s3.amazonaws.com/IMG_6393.JPG

Click on Make Public and then Click on Object URL.

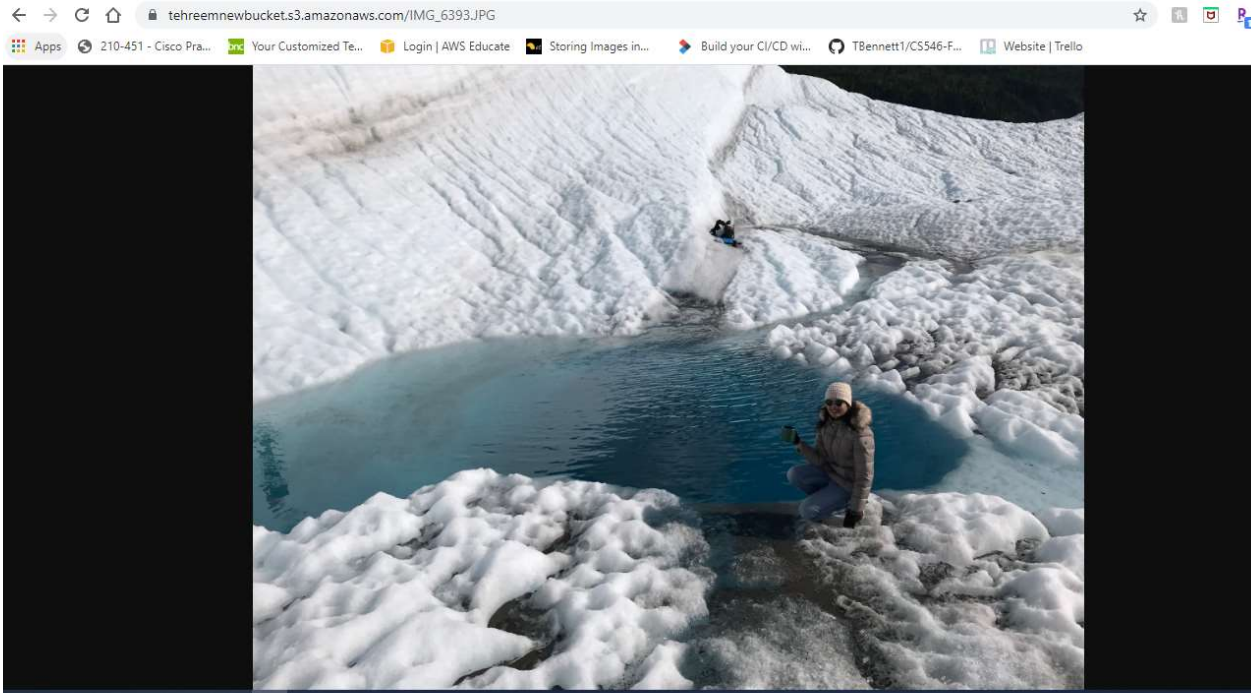
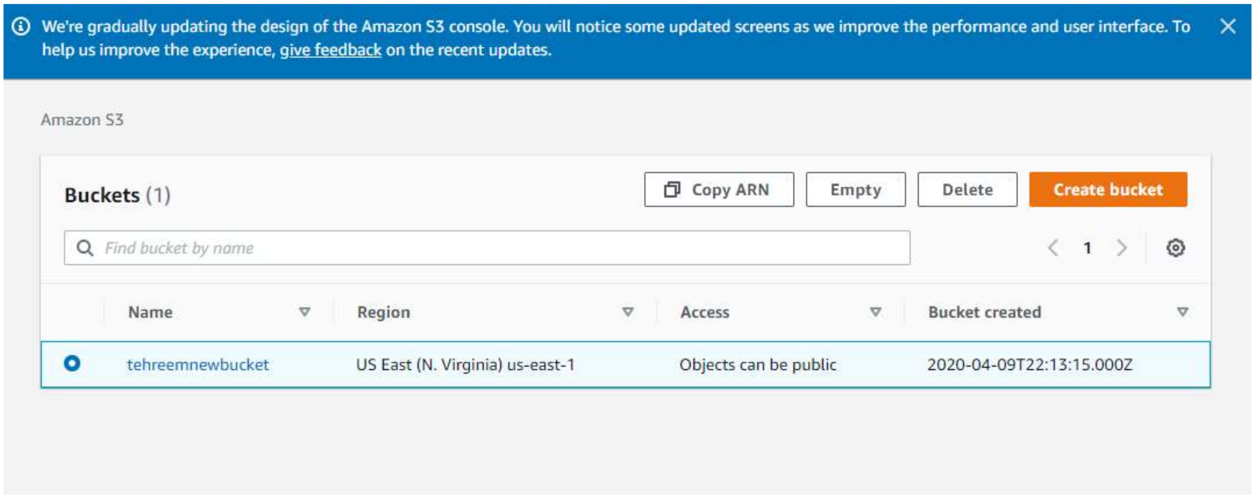


Image is displayed successfully!



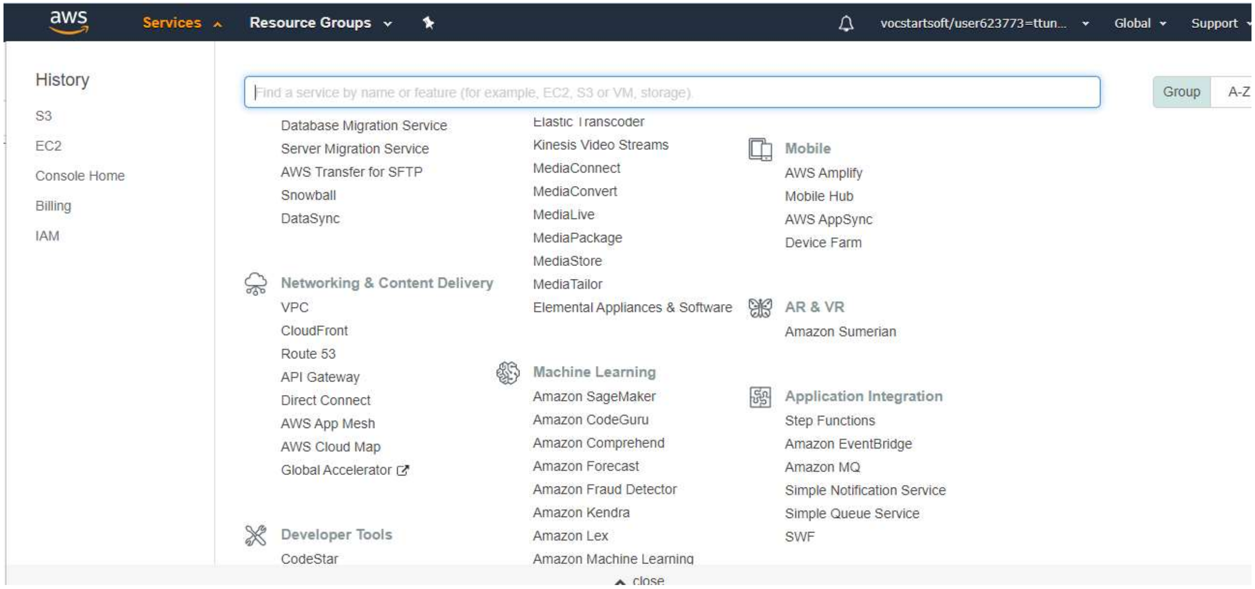
We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.

	Name	Region	Access	Bucket created
●	tehreemnewbucket	US East (N. Virginia) us-east-1	Objects can be public	2020-04-09T22:13:15.000Z

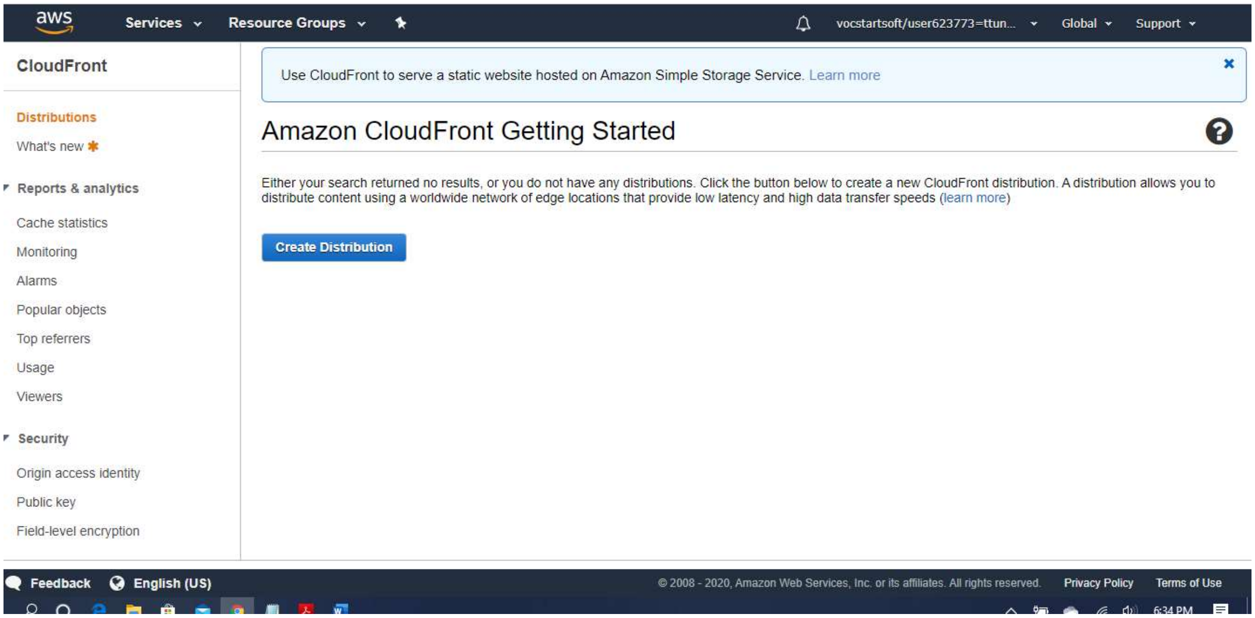
Click on Copy ARN:
arn:aws:s3:::tehreemnewbucket
is the arn for my bucket.

2) Create a Web Distribution in Cloud Front:

Go to EC2 dashboard and under Services>Networking & Content Delivery>CloudFront:



Click on Create Distribution on the next page:

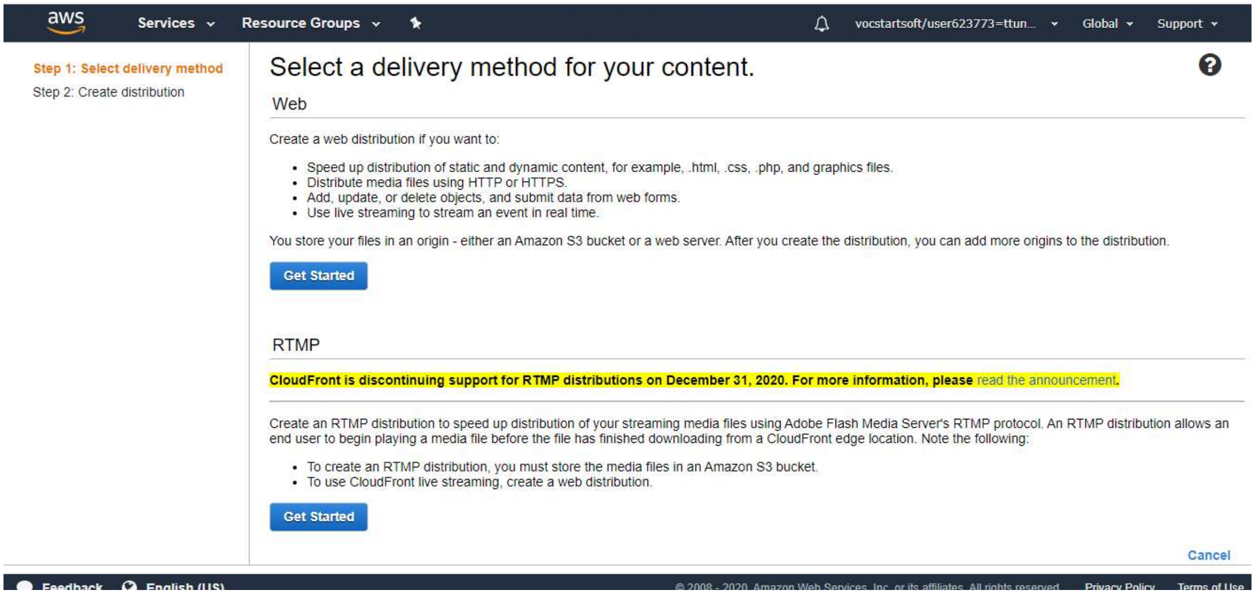


Use CloudFront to serve a static website hosted on Amazon Simple Storage Service. Learn more

Amazon CloudFront Getting Started

Either your search returned no results, or you do not have any distributions. Click the button below to create a new CloudFront distribution. A distribution allows you to distribute content using a worldwide network of edge locations that provide low latency and high data transfer speeds (learn more)

Click on Get Started:


aws
Services
Resource Groups
vocstartsoft/user623773=ttun...
Global
Support
Step 1: Select delivery method
Step 2: Create distribution
Get Started
Get Started
Cancel
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Select a delivery method for your content.

Web

Create a web distribution if you want to:

- Speed up distribution of static and dynamic content, for example, .html, .css, .php, and graphics files.
- Distribute media files using HTTP or HTTPS.
- Add, update, or delete objects, and submit data from web forms.
- Use live streaming to stream an event in real time.

You store your files in an origin - either an Amazon S3 bucket or a web server. After you create the distribution, you can add more origins to the distribution.

RTMP

CloudFront is discontinuing support for RTMP distributions on December 31, 2020. For more information, please read the announcement.

Create an RTMP distribution to speed up distribution of your streaming media files using Adobe Flash Media Server's RTMP protocol. An RTMP distribution allows an end user to begin playing a media file before the file has finished downloading from a CloudFront edge location. Note the following:

- To create an RTMP distribution, you must store the media files in an Amazon S3 bucket.
- To use CloudFront live streaming, create a web distribution.

https://tehreemnewbucket.s3.amazonaws.com/IMG_6393.JPG is the Origin name of my bucket.

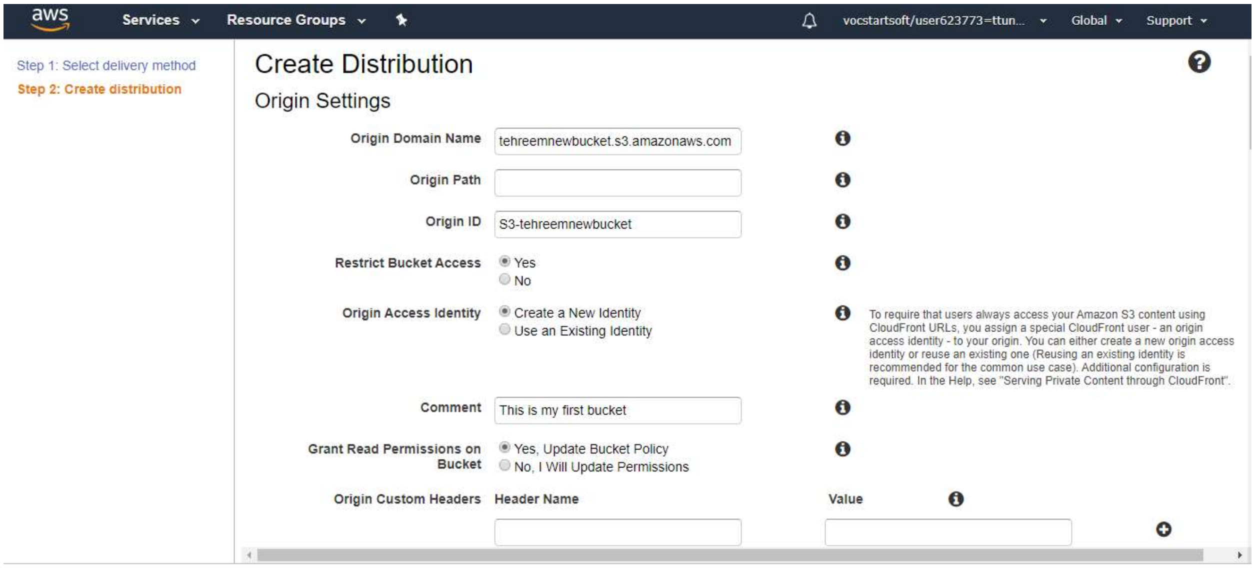
The first parameter it asks for is Origin domain Name.

Origin Domain Name is the DNS domain name of the Amazon S3 bucket from which you want CloudFront to get objects for this origin. The file must be publicly readable unless you secure your content in Amazon S3 by using a CloudFront origin access identity.

Origin Path: If you want CloudFront to request your content from a directory in your Amazon S3 bucket or your custom origin, enter the directory name here

Origin ID: This value lets you distinguish multiple origins in the same distribution from one another.

Many other fields are available like whether you want to have viewer policy as HTTP or HTTPS and whether it should be GET,POST,PUT,PATCH,DELETE. Since it is just an image that I am accessing, I will stick to GET request.
And GET is also the default!
In order to restrict access to content that you serve from Amazon S3 buckets, you create CloudFront signed URLs or signed cookies. Then you create a special CloudFront user called an origin access identity (OAI) and associate it with your distribution. Then you configure permissions so that CloudFront can use the OAI to access and serve files to your users, but users can't use a direct URL to the S3 bucket to access a file there. Taking these steps help you maintain secure access to the files that you serve through CloudFront.


aws
Services
Resource Groups
vocstartsoft/user623773=ttun...
Global
Support
Step 1: Select delivery method
Step 2: Create distribution

Create Distribution

Origin Settings

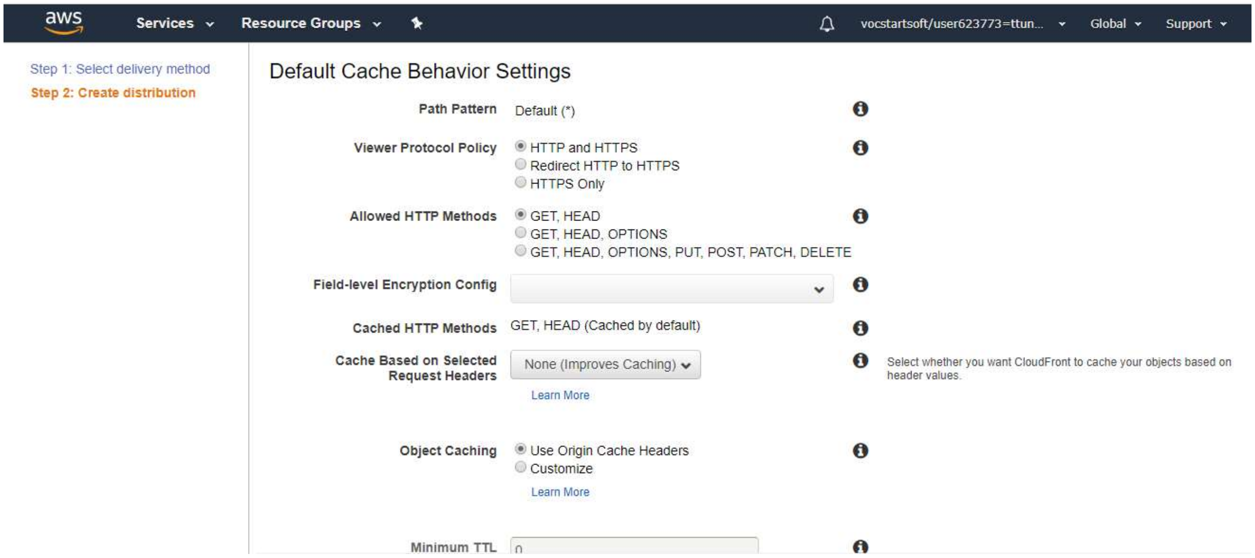
Origin Domain Name	tehreemnewbucket.s3.amazonaws.com
Origin Path	
Origin ID	S3-tehreemnewbucket
Restrict Bucket Access	● Yes ○ No
Origin Access Identity	● Create a New Identity ○ Use an Existing Identity
Comment	This is my first bucket
Grant Read Permissions on Bucket	● Yes, Update Bucket Policy ○ No, I Will Update Permissions
Origin Custom Headers	Header Name / Value

To require that users always access your Amazon S3 content using CloudFront URLs, you assign a special CloudFront user - an origin access identity - to your origin. You can either create a new origin access identity or reuse an existing one (Reusing an existing identity is recommended for the common use case). Additional configuration is required. In the Help, see "Serving Private Content through CloudFront".

So, in the above screen, the fields can be explained as:

- Origin Domain Name, I selected my bucket from the dropdown(When you click on the text box, you get a dropdown list automatically).
- Origin ID: It lets you distinguish your bucket from other buckets on Amazon. This value was automatically filled when I selected my bucket's origin domain name.
- Restrict Bucket Access: It is important to set to parameter to "Yes" if you want the bucket's access be restricted to allow only users with CloudFront signed URL's to access it. This is important in terms of security.
- Origin Access Identity: Since we want to create a new OAI, which can access our bucket, select "Create a new Identity".
- Comment: For a new OAI, this field can be replaced by a short description about the bucket.
- Grant read Permissions on Bucket: I selected "Yes, update Bucket Policy". CloudFront updates bucket permissions to grant the specified OAI permission to read files in your bucket. However, CloudFront does not remove existing permissions. If users currently have permission to access the files in your bucket

using Amazon S3 URLs, they will still have that permission after CloudFront updates your bucket permissions. To view or remove existing bucket permissions, use a method provided by Amazon S3. If the other option, “No, I Will Update Permissions” is selected, then you need to manually update permissions on your S3 bucket.


aws
Services
Resource Groups
vocstartsoft/user623773=ttun...
Global
Support
Step 1: Select delivery method
Step 2: Create distribution

Default Cache Behavior Settings

Path Pattern: Default (*)

Viewer Protocol Policy: ● HTTP and HTTPS ○ Redirect HTTP to HTTPS ○ HTTPS Only

Allowed HTTP Methods: ● GET, HEAD ○ GET, HEAD, OPTIONS ○ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

Field-level Encryption Config:

Cached HTTP Methods: GET, HEAD (Cached by default)

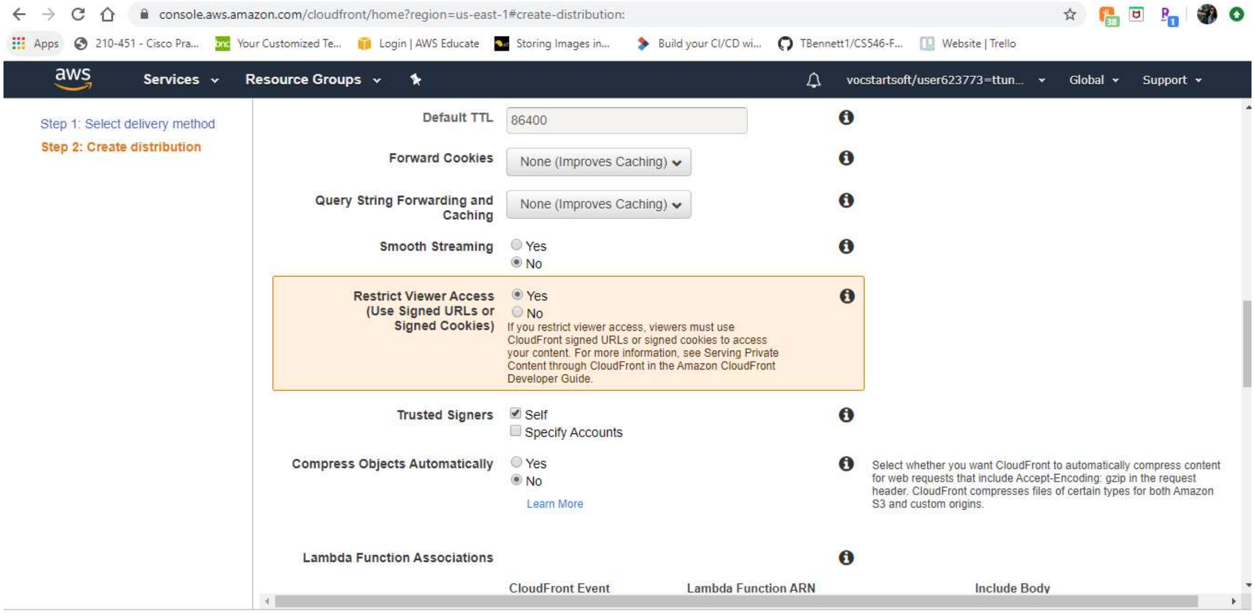
Cache Based on Selected Request Headers: None (Improves Caching) — Learn More

Select whether you want CloudFront to cache your objects based on header values.

Object Caching: ● Use Origin Cache Headers ○ Customize — Learn More

Minimum TTL: 0

I have kept all the other values to default.



Default TTL: 86400

Forward Cookies: None (Improves Caching)

Query String Forwarding and Caching: None (Improves Caching)

Smooth Streaming: ○ Yes ● No

Restrict Viewer Access (Use Signed URLs or Signed Cookies): ● Yes ○ No

If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content. For more information, see Serving Private Content through CloudFront in the Amazon CloudFront Developer Guide.

Trusted Signers: ☑ Self ☐ Specify Accounts

Compress Objects Automatically: ○ Yes ● No — Learn More

Select whether you want CloudFront to automatically compress content for web requests that include Accept-Encoding: gzip in the request header. CloudFront compresses files of certain types for both Amazon S3 and custom origins.

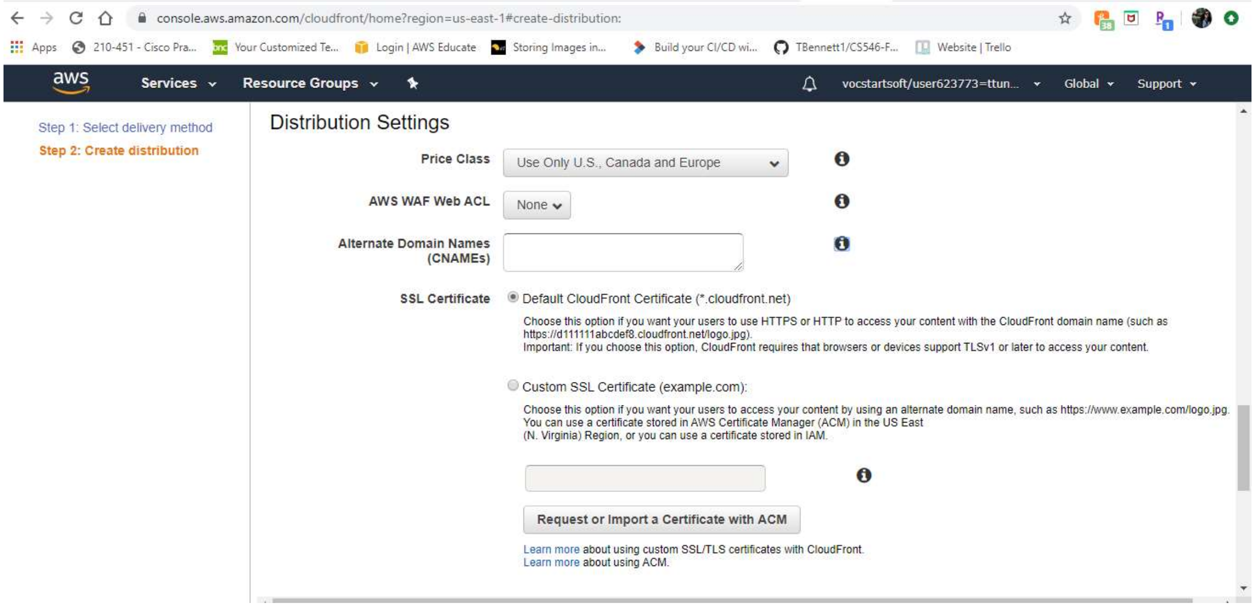
Lambda Function Associations

CloudFront Event	Lambda Function ARN	Include Body

You can also restrict viewer access. If this is chose, only users with signed URLs or Signed Cookies will be able to view the content.

You can also specify trusted accounts i.e. Choose whether you want to use the current AWS account and/or other AWS accounts to create signed URLs or signed cookies. If you choose to Specify Accounts, it asks for Account numbers so that CloudFront can create Signed URLs or Signed Cookies for those particular accounts. Since I do not want to add any other account, I selected "Self".

Lambda function associations can also be specified i.e. Specify Lambda function ARNs to associate with specific event types, up to one ARN per event type. By selecting Include Body, you can also choose to read the request body for viewer request and origin request events.



console.aws.amazon.com/cloudfront/home?region=us-east-1#create-distribution:

Step 1: Select delivery method

Step 2: Create distribution

Distribution Settings

Price Class: Use Only U.S., Canada and Europe

AWS WAF Web ACL: None

Alternate Domain Names (CNAMEs):

SSL Certificate:

● Default CloudFront Certificate (*.cloudfront.net)

Choose this option if you want your users to use HTTPS or HTTP to access your content with the CloudFront domain name (such as https://d111111abcdef8.cloudfront.net/logo.jpg).
Important: If you choose this option, CloudFront requires that browsers or devices support TLSv1 or later to access your content.

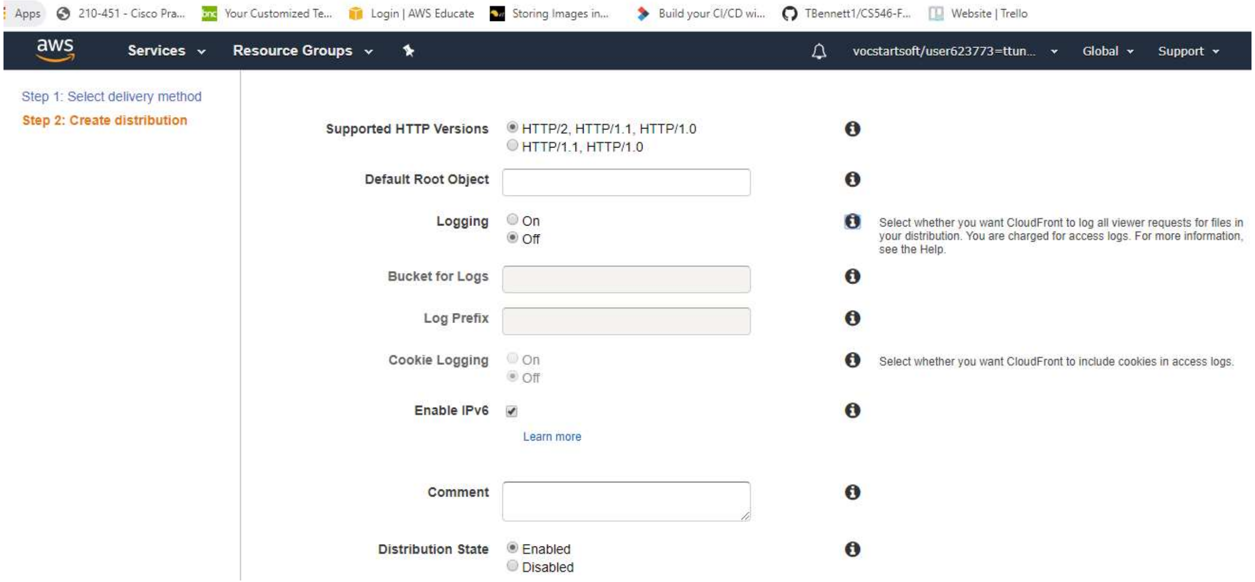
○ Custom SSL Certificate (example.com):

Choose this option if you want your users to access your content by using an alternate domain name, such as https://www.example.com/logo.jpg.
You can use a certificate stored in AWS Certificate Manager (ACM) in the US East
(N. Virginia) Region, or you can use a certificate stored in IAM.

Request or Import a Certificate with ACM

Learn more about using custom SSL/TLS certificates with CloudFront.
Learn more about using ACM.

Since I am not concerned about someone other than US, Canada and Europe accessing, I do not want to pay more to include everyone in the world, so selected this option. So, users apart from these three regions may experience higher latency.



Step 1: Select delivery method

Step 2: Create distribution

Supported HTTP Versions: ● HTTP/2, HTTP/1.1, HTTP/1.0 ○ HTTP/1.1, HTTP/1.0

Default Root Object:

Logging: ○ On ● Off

Select whether you want CloudFront to log all viewer requests for files in your distribution. You are charged for access logs. For more information, see the Help.

Bucket for Logs:

Log Prefix:

Cookie Logging: ○ On ● Off

Select whether you want CloudFront to include cookies in access logs.

Enable IPv6: ☑

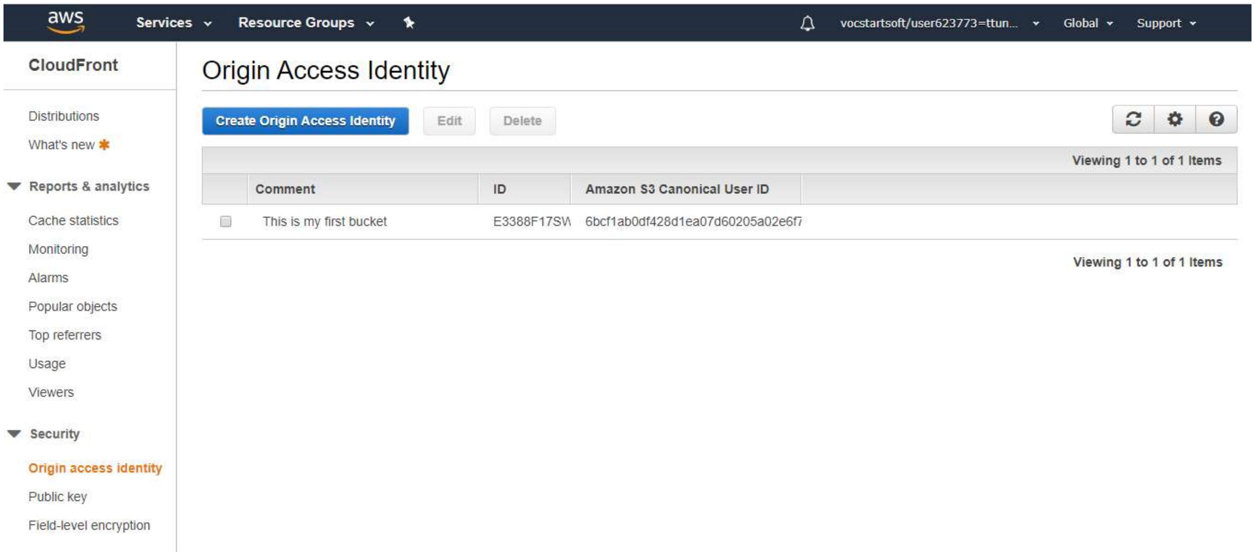
Learn more

Comment:

Distribution State: ● Enabled ○ Disabled

Amazon will charge extra for Logging, so I turned it off. Click on Create Distribution.

As you can see, a new Origin Access Identity has been created.



Origin Access Identity

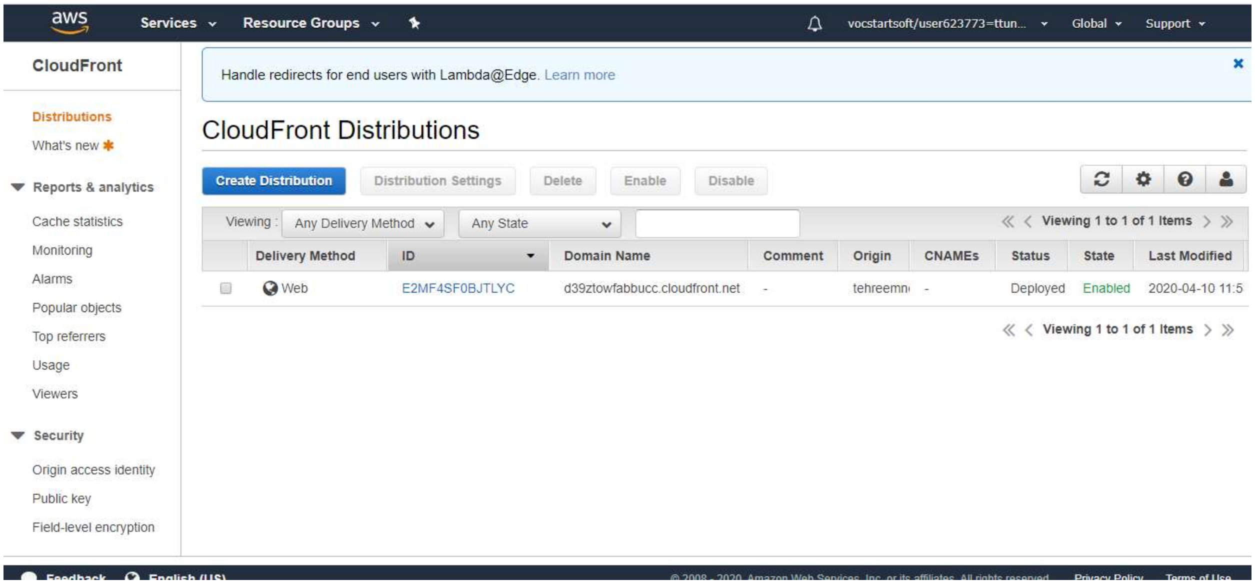
Viewing 1 to 1 of 1 Items

	Comment	ID	Amazon S3 Canonical User ID
□	This is my first bucket	E3388F17SW	6bcf1ab0df428d1ea07d60205a02e6f7

Viewing 1 to 1 of 1 Items

3) Note the domain name of your distribution:

Click on Distributions tab on the left side of the screen:



Handle redirects for end users with Lambda@Edge. Learn more

CloudFront Distributions

Viewing : Any Delivery Method · Any State

Viewing 1 to 1 of 1 Items

	Delivery Method	ID	Domain Name	Comment	Origin	CNAMEs	Status	State	Last Modified
□	Web	E2MF4SF0BJTLYC	d39ztowfabbucc.cloudfront.net	-	tehreemn	-	Deployed	Enabled	2020-04-10 11:5

Viewing 1 to 1 of 1 Items

The distribution has successfully been deployed!

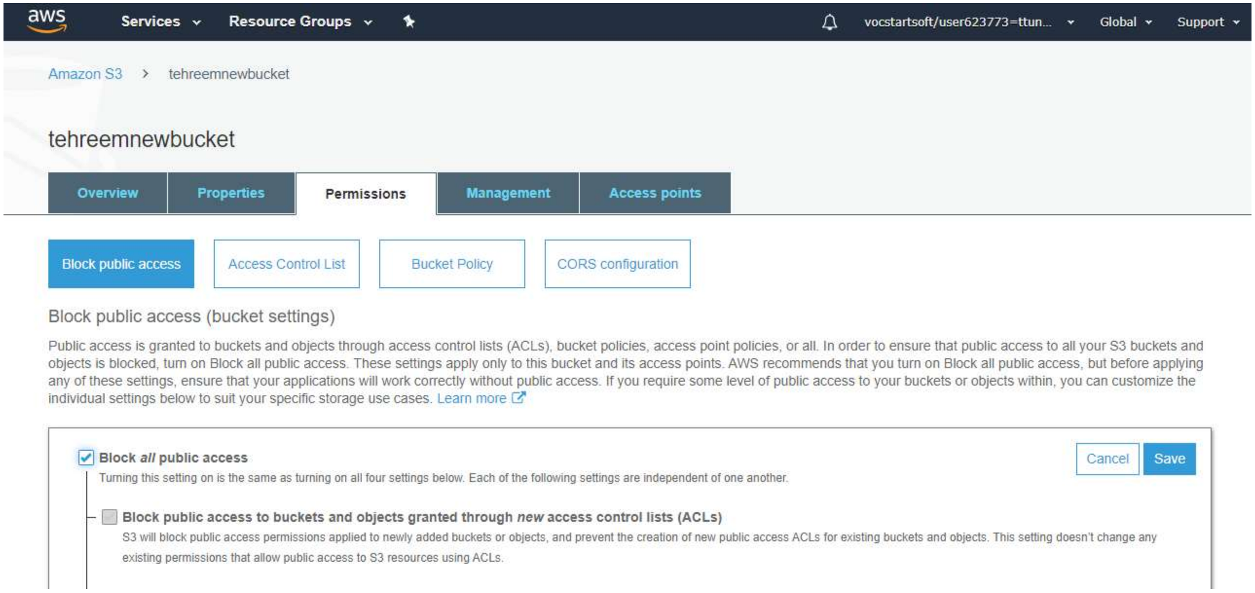
The domain name of the distribution is:

d39ztowfabbucc.cloudfront.net

4) **Going back to the bucket and disabling public read access:**

Select S3 from Services on EC2 dashboard and go to your bucket. When I clicked on the bucket, Object URL, I was able to see the image.

Now go to Permissions tab and select to block public access and click on Save.



Amazon S3 > tehreemnewbucket

tehreemnewbucket

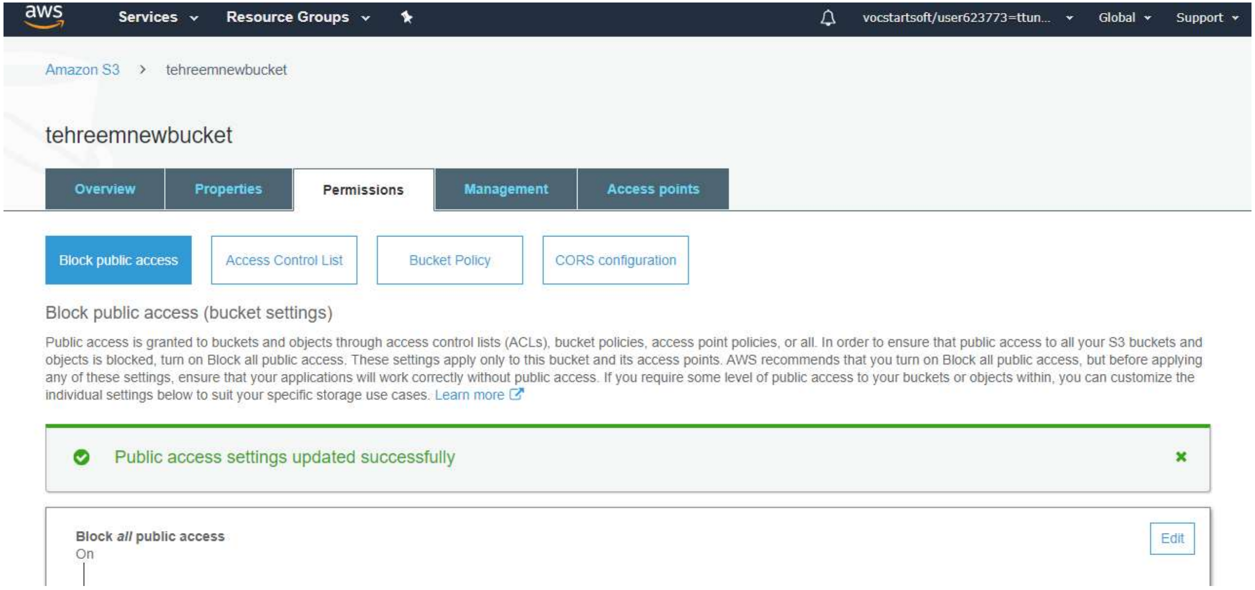
Block public access (bucket settings)

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.

After confirming,



Amazon S3 > tehreemnewbucket

tehreemnewbucket

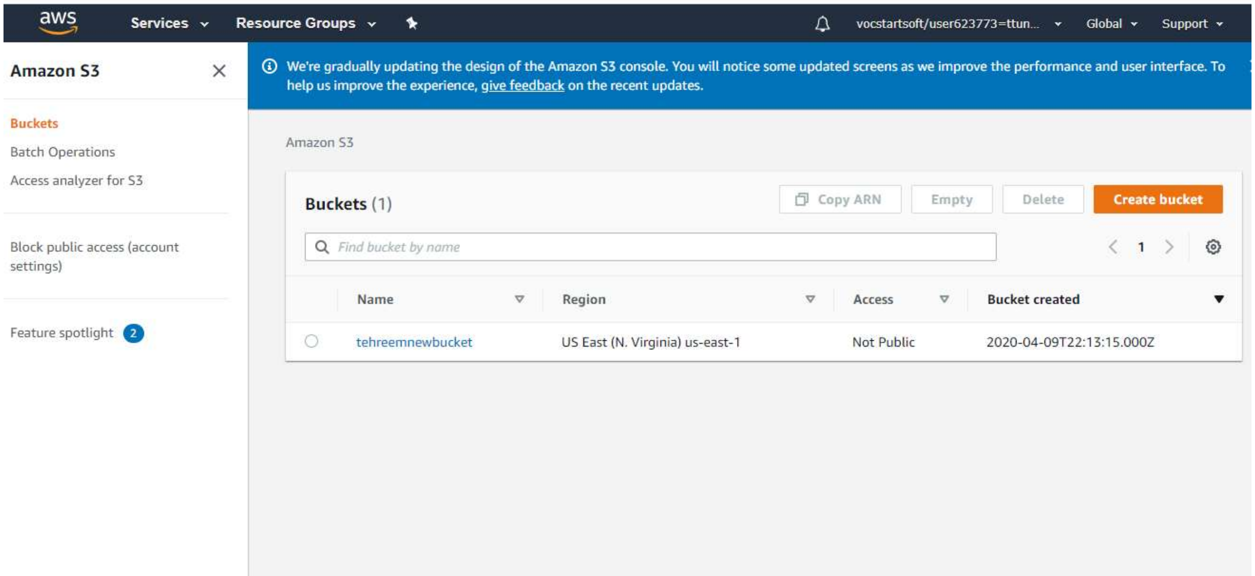
Block public access (bucket settings)

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

✓ Public access settings updated successfully

Block *all* public access
On

5) **Click again on Object URL of the image:**



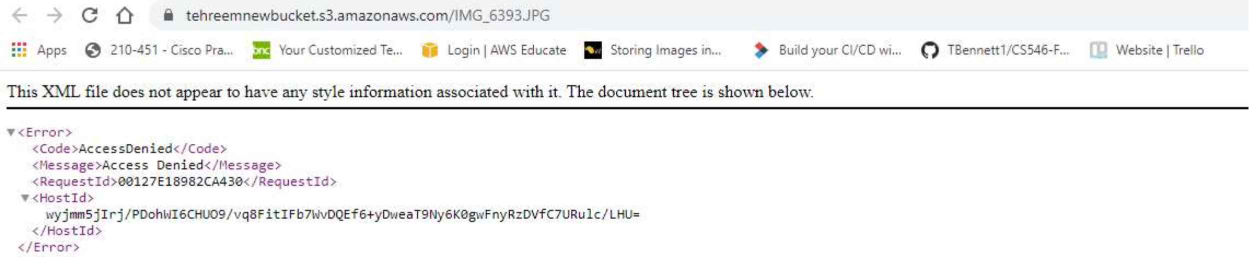
We're gradually updating the design of the Amazon S3 console. You will notice some updated screens as we improve the performance and user interface. To help us improve the experience, give feedback on the recent updates.

Amazon S3

Buckets (1)

	Name	Region	Access	Bucket created
○	tehreemnewbucket	US East (N. Virginia) us-east-1	Not Public	2020-04-09T22:13:15.000Z

Click on the name of bucket, Go to Overview tab and Click on the Image, Object URL:
After some 2 minutes, click on Object URL:

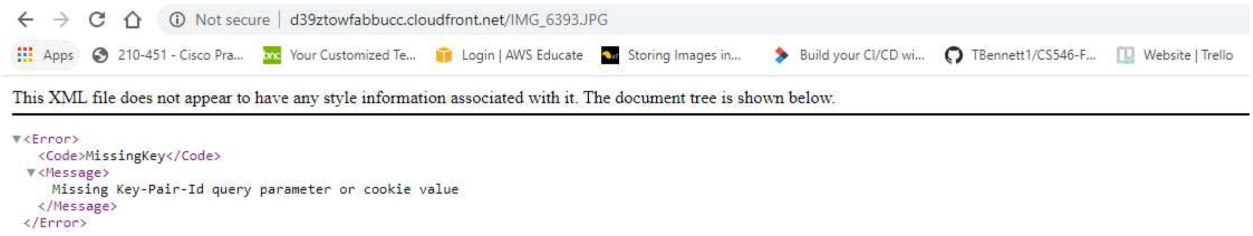


This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>AccessDenied</Code>
   <Message>Access Denied</Message>
   <RequestId>00127E18982CA430</RequestId>
 ▼<HostId>
     wyjmm5jIrj/PDohWI6CHUO9/vq8FitIFb7WvDQEf6+yDweaT9Ny6K0gwFnyRzDVfC7URulc/LHU=
   </HostId>
 </Error>
```

Since I disabled the Public Access parameter, I am unable to access the image through the Object URL. So earlier, when the bucket was publicly accessible, anyone who had my bucket's object URL could go and see it using it. Now, since the access is not public anymore, the bucket cannot be accessed by anyone publicly. It will securely be accessed only through the CloudFront's secure URL. This is a security feature which is very essential as you do not want anyone to access your data.

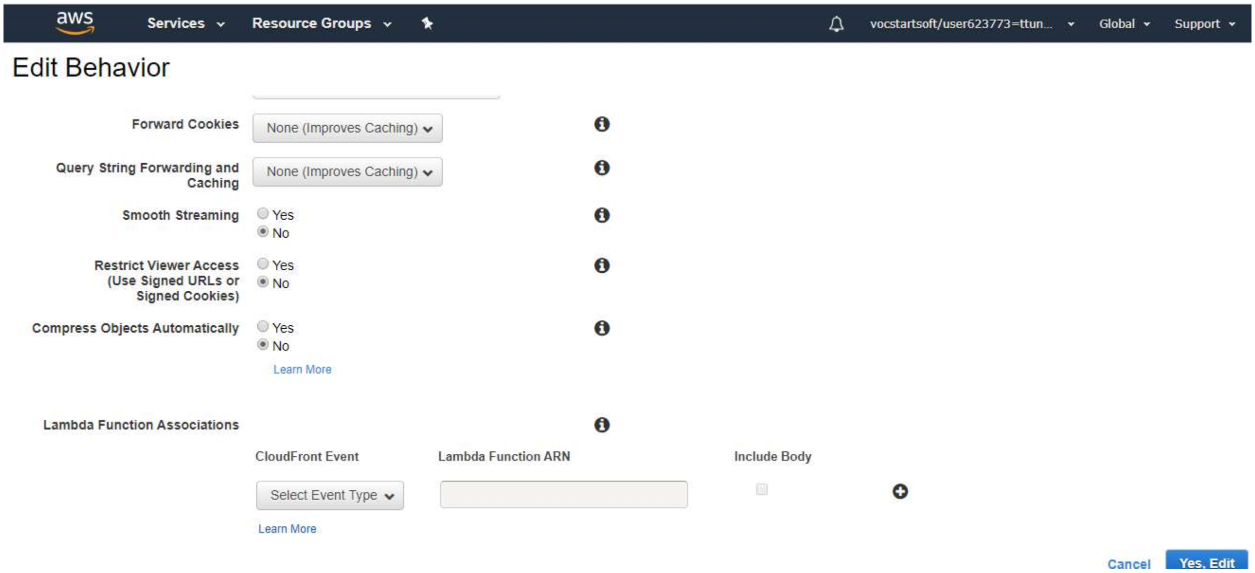
6) **Now change the Object URL to replace some part of it by your distribution's domain name so that that it displays the image.**



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
   <Code>MissingKey</Code>
  ▼<Message>
     Missing Key-Pair-Id query parameter or cookie value
   </Message>
 </Error>
```

This error is because I restricted viewer access earlier to signed URLs or Signed Cookies only.
Go to Distributions and Edit Restrict Viewer access by Selecting No:



Edit Behavior

Forward Cookies: None (Improves Caching)

Query String Forwarding and Caching: None (Improves Caching)

Smooth Streaming: ○ Yes ● No

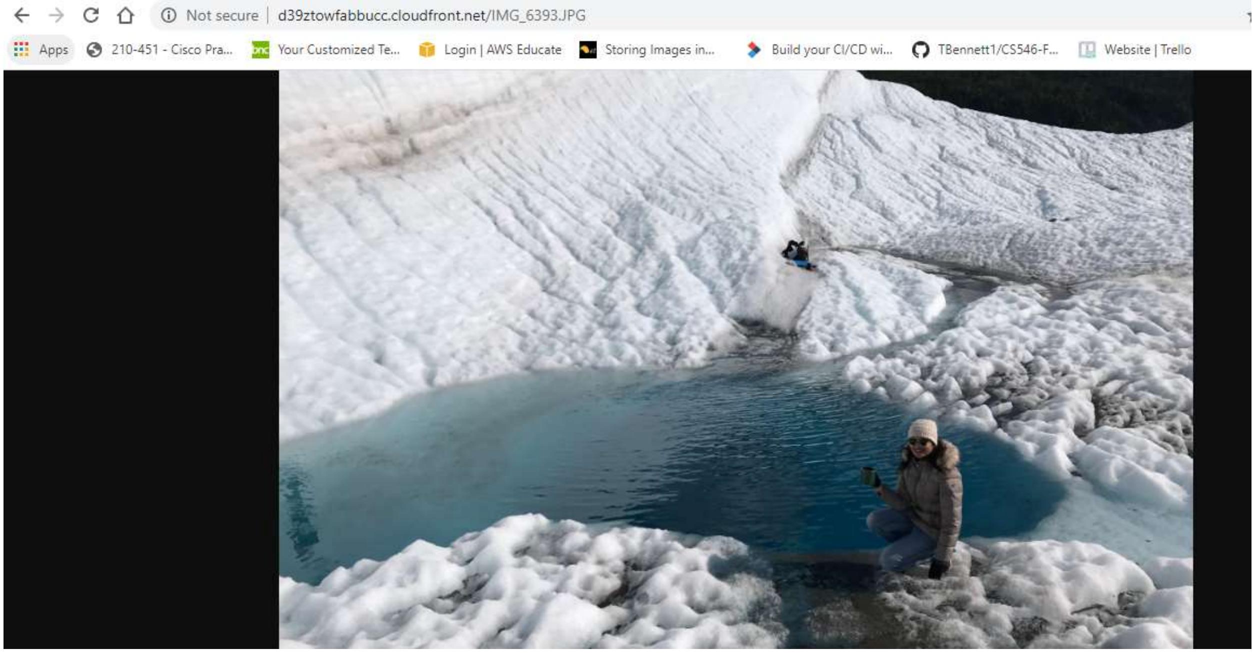
Restrict Viewer Access (Use Signed URLs or Signed Cookies): ○ Yes ● No

Compress Objects Automatically: ○ Yes ● No
Learn More

Lambda Function Associations

CloudFront Event: Select Event Type | Lambda Function ARN: | Include Body: □
Learn More

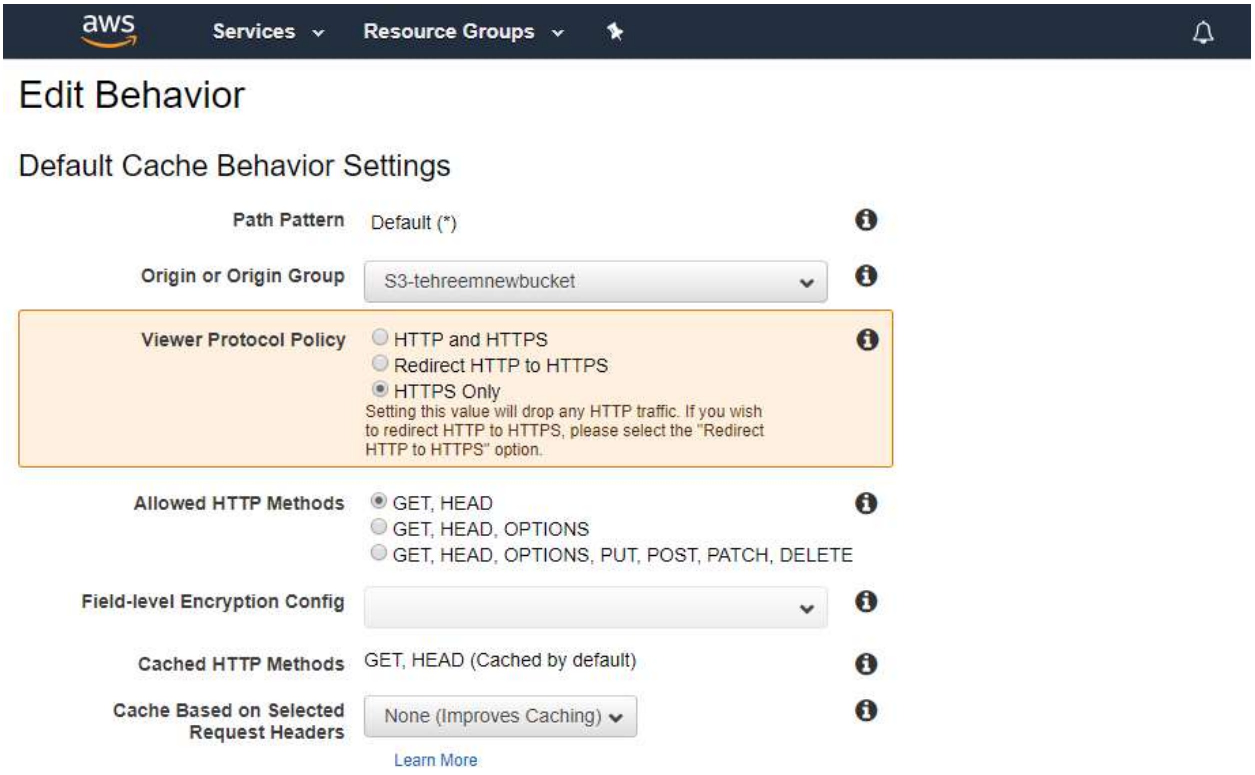
Now edit the object URL and check again:



I did not experience any latency when the image was loading (This may be due to the fact that I included only US, Euro and Canada). In fact, the **speed increased** after creating the CDN.

Extra Steps:

Experimenting more with other fields:



Edit Behavior

Default Cache Behavior Settings

Path Pattern: Default (*)

Origin or Origin Group: S3-tehreemnewbucket

Viewer Protocol Policy:
○ HTTP and HTTPS
○ Redirect HTTP to HTTPS
● HTTPS Only

Setting this value will drop any HTTP traffic. If you wish to redirect HTTP to HTTPS, please select the "Redirect HTTP to HTTPS" option.

Allowed HTTP Methods:
● GET, HEAD
○ GET, HEAD, OPTIONS
○ GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

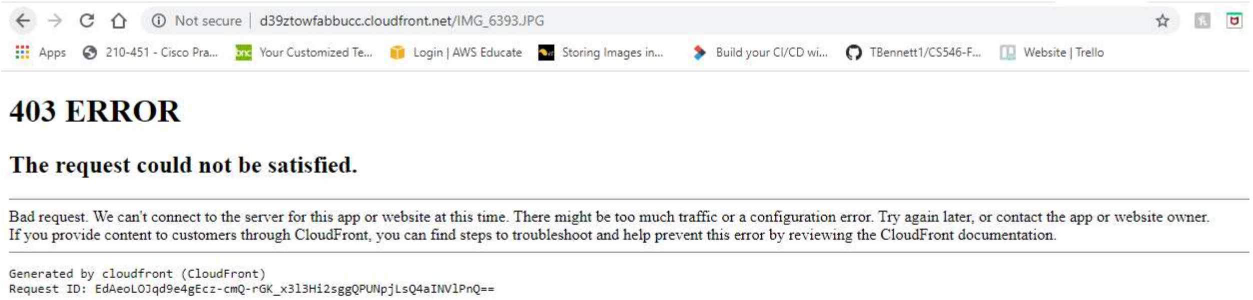
Field-level Encryption Config:

Cached HTTP Methods: GET, HEAD (Cached by default)

Cache Based on Selected Request Headers: None (Improves Caching)

Learn More

Now my site should not be accessible if I put http://d39ztowfabbucc.cloudfront.net/IMG_6393.JPG



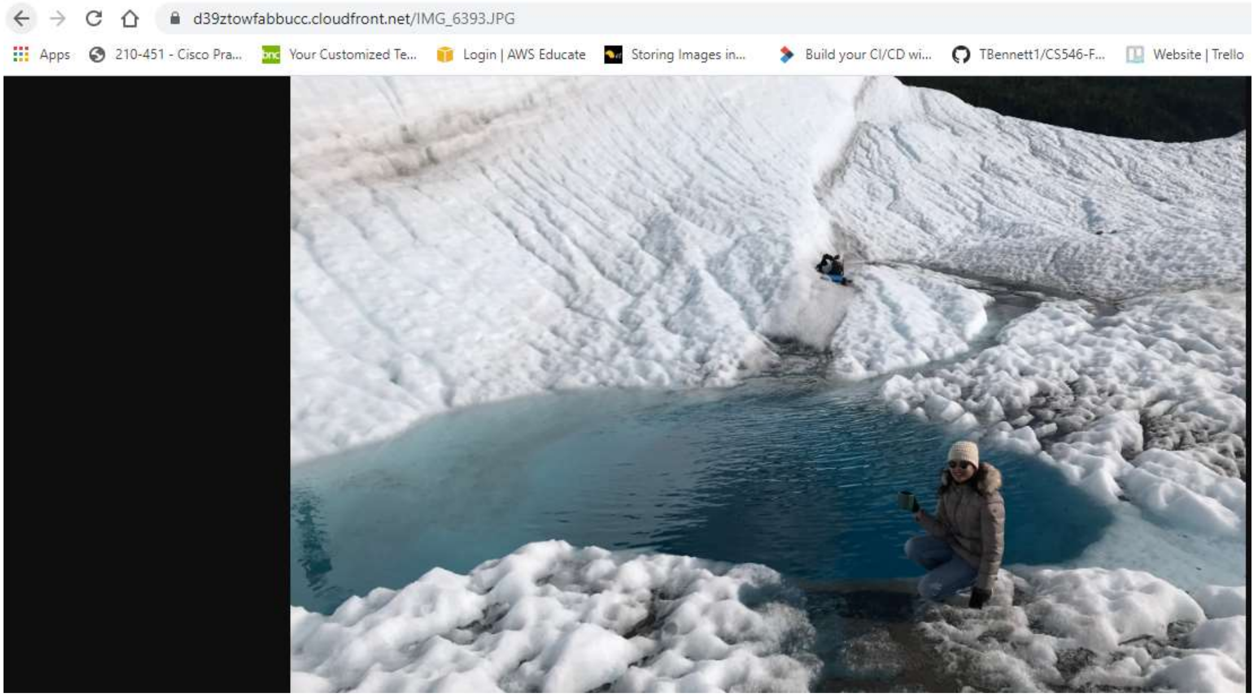
403 ERROR

The request could not be satisfied.

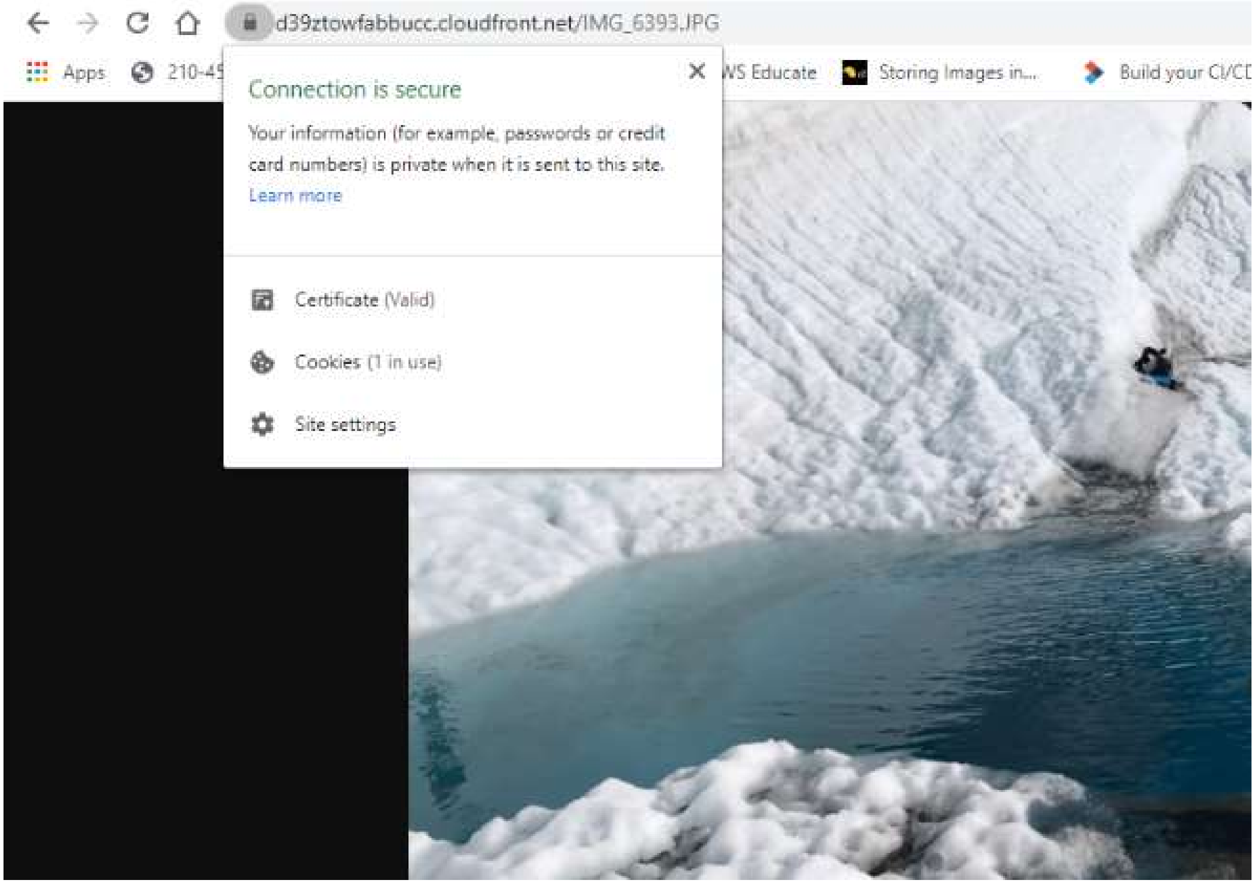
Bad request. We can't connect to the server for this app or website at this time. There might be too much traffic or a configuration error. Try again later, or contact the app or website owner.
If you provide content to customers through CloudFront, you can find steps to troubleshoot and help prevent this error by reviewing the CloudFront documentation.

```
Generated by cloudfront (CloudFront)
Request ID: EdAeoLOJqd9e4gEcz-cmQ-rGK_x3l3Hi2sggQPUNpjLsQ4aINVlPnQ==
```

Using https:// d39ztowfabbucc.cloudfront.net/IMG_6393.JPG
Also, notice that the “Not Secure” besides the address bar has now turned into a lock/secure symbol:



This shows that the connection is secure through HTTP and TLS. If you click on the lock symbol, it shows:



Connection is secure

Your information (for example, passwords or credit card numbers) is private when it is sent to this site.

Companies which are in media, entertainment, gaming, software, online retail and many more which have digital rich content on their website and want to deliver the same to their audience quickly and reliably can use CDN. Consumers want a high-quality online experience whether they are watching a movie, streaming an event, playing a game or shopping online. Using CDNs results in an increase of performance, thus giving the end users an enhanced consumer experience.

Here are few of the benefits of using a CDN for your website:

1. Your Server Load Will Decrease:

As a result of, strategically placed servers which form the backbone of the network the companies can have an increase in capacity and number of concurrent users that they can handle. Essentially, the content is spread out across several servers, as opposed to offloading them onto one large server.

2. Content Delivery Will Become Faster:

Due to higher reliability, operators can deliver high-quality content with a high level of service, low network server loads, and thus, lower costs. Moreover, jQuery is ubiquitous on the web. There's a high probability that someone visiting a particular page has already done that in the past using the Google CDN. Therefore, the file has already been cached by the browser and the user won't need to download again.

3. Segmenting Your Audience Becomes Easy:

CDNs can deliver different content to different users depending on the kind of device requesting the content. They are capable of detecting the type of mobile devices and can deliver a device-specific version of the content.

4. Lower Network Latency And Packet Loss:

End users experience less jitter and improved stream quality. CDN users can, therefore, deliver high definition content with high Quality of Service, low costs, and low network load.

5. Higher Availability And Better Usage Analytics:

CDNs dynamically distribute assets to the strategically placed core, fallback, and edge servers. CDNs can give more control of asset delivery and network load. They can optimize capacity per customer, provide views of real-time load and statistics, reveal which assets are popular, show active regions and report exact viewing details to customers. CDNs can thus offer 100% availability, even with large power, network or hardware outages.

6. Storage And Security:

CDNs offer secure storage capacity for content such as videos for enterprises that need it, as well as archiving and enhanced data backup services. CDNs can secure content through Digital Rights Management and limit access through user authentication.

Source: [https://www.bluepiit.com/blog/6-advantages-of-using-a-content-delivery-network-cdn/]