

1) Creating S3 bucket:

To create S3 bucket log into your AWS account and from the services dropdown click into the S3 as seen below is in the storage section.

The screenshot shows the AWS Management Console home page. The 'Services' menu is expanded, and the 'Storage' section is selected. Under 'Storage', 'S3' is listed. To the right, there's a sidebar titled 'Explore AWS' with sections like 'S3 Intelligent-Tiering', 'AMD Powered EC2 Instances', 'Debug ML Models', and 'EMR Migration Guide'. At the bottom right of the sidebar is a 'Have feedback?' section with a 'Submit feedback' button.

- Click on the create bucket well to create a S3 bucket.

The screenshot shows the Amazon S3 service console. On the left, there's a sidebar with 'Buckets', 'Batch Operations', 'Access analyzer for S3', 'Block public access (account settings)', and a 'Feature spotlight' section. The main area is titled 'Amazon S3' and shows a message about console updates. Below that is a table titled 'Buckets (0)' with columns for 'Name', 'Region', 'Access', and 'Bucket created'. A large orange 'Create bucket' button is prominently displayed at the bottom of the table area.

- Bucket name must be unique, and you cannot create it unless it is and meets all the requirements which is mentioned. Also make sure to note region as buckets made in specific region and are accessible globally.

The screenshot shows the 'Create bucket' page in the AWS S3 console. In the 'General configuration' section, the 'Bucket name' is set to 'myawsbucket' and the 'Region' is set to 'US East (N. Virginia) us-east-1'. Below this, the 'Bucket settings for Block Public Access' section is expanded, showing the 'Block all public access' setting is checked. A note explains that this setting applies to all four settings below and is independent of one another. Under this setting, the 'Block public access to buckets and objects granted through new access control lists (ACLs)' option is also checked. At the bottom of the page, there are links for 'Feedback', 'English (US)', and copyright information: '© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.' followed by 'Privacy Policy' and 'Terms of Use'.

-We can change the bucket setting to public or private, it is suggested to make the bucket not public but for the sake of this experiment, I have changed the permission to public in permission tab.

-With that we have successfully created an S3 bucket.

s3.console.aws.amazon.com/s3/home?region=us-east-1

Amazon S3

Buckets

Batch Operations
Access analyzer for S3

Block public access (account settings)

Feature spotlight

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 labcloudcompute
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
labcloudcompute	US East (N. Virginia) us-east-1	Objects can be public	2020-04-18T22:29:37.000Z

Feedback English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

s3.console.aws.amazon.com/s3/buckets/labcloudcompute/?region=us-east-1

labcloudcompute

Overview Properties Permissions Management Access points

Upload Create folder Download Actions US East (N. Virginia)

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

Upload an object
Buckets are globally unique containers for everything that you store in Amazon S3.
[Learn more](#)

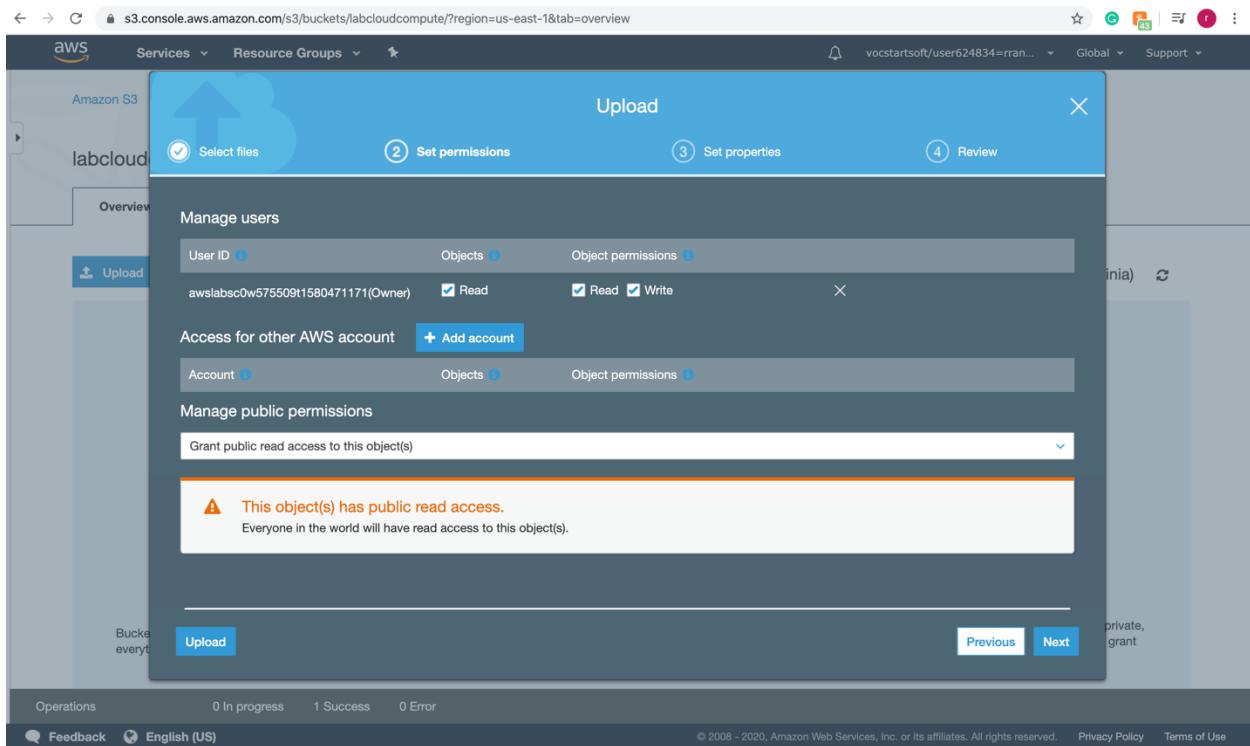
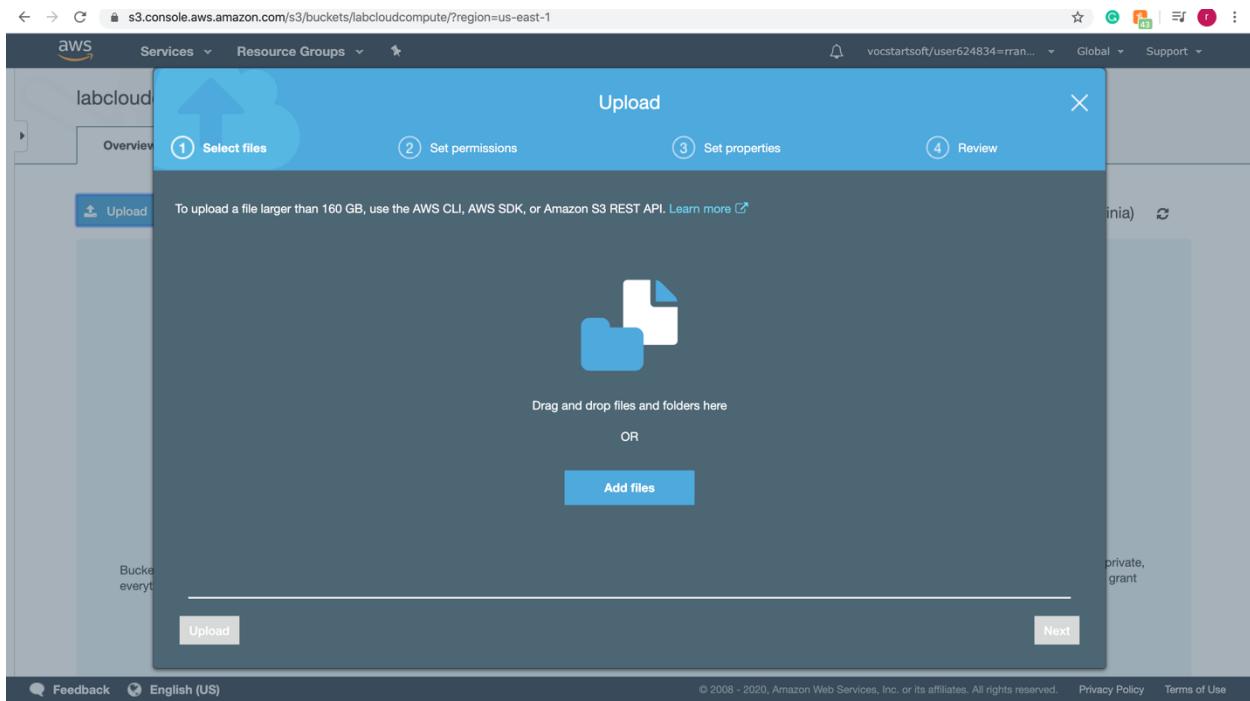
Set object properties
After you create a bucket, you can upload your objects (for example, your photo or video files).
[Learn more](#)

Set object permissions
By default, the permissions on an object are private, but you can set up access control policies to grant permissions to others.
[Learn more](#)

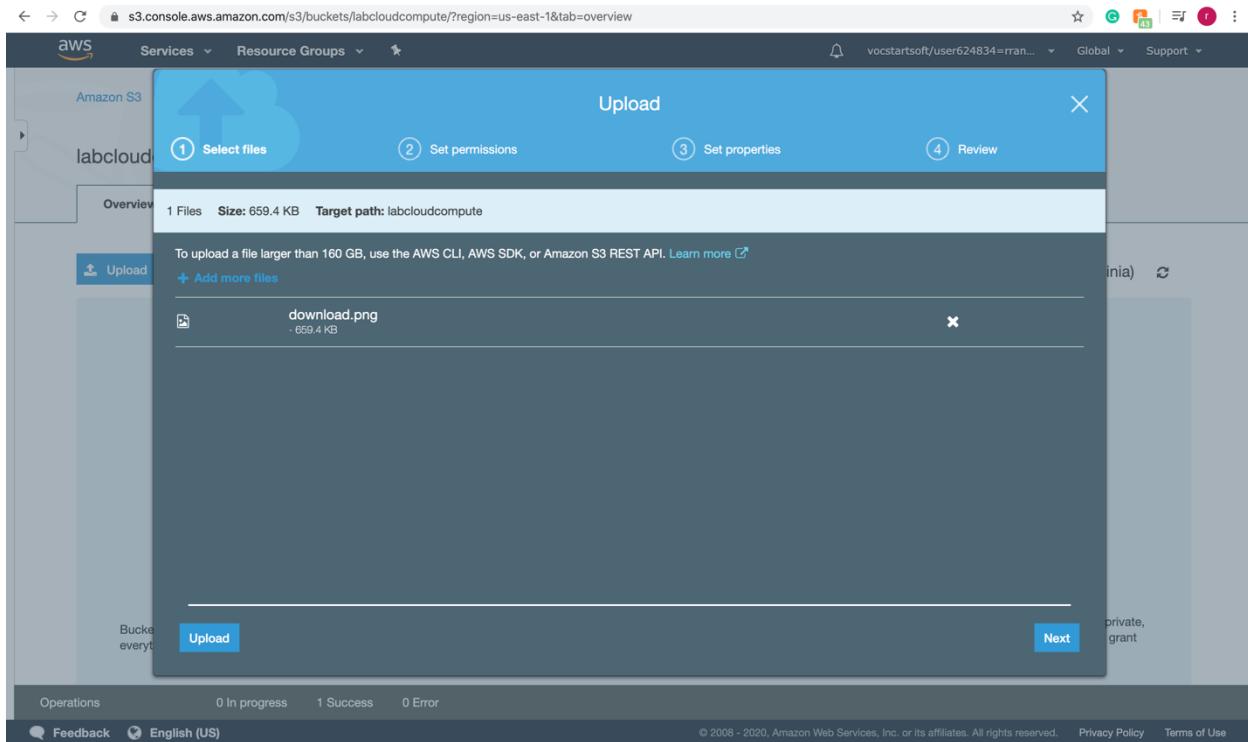
Get started

Feedback English (US)

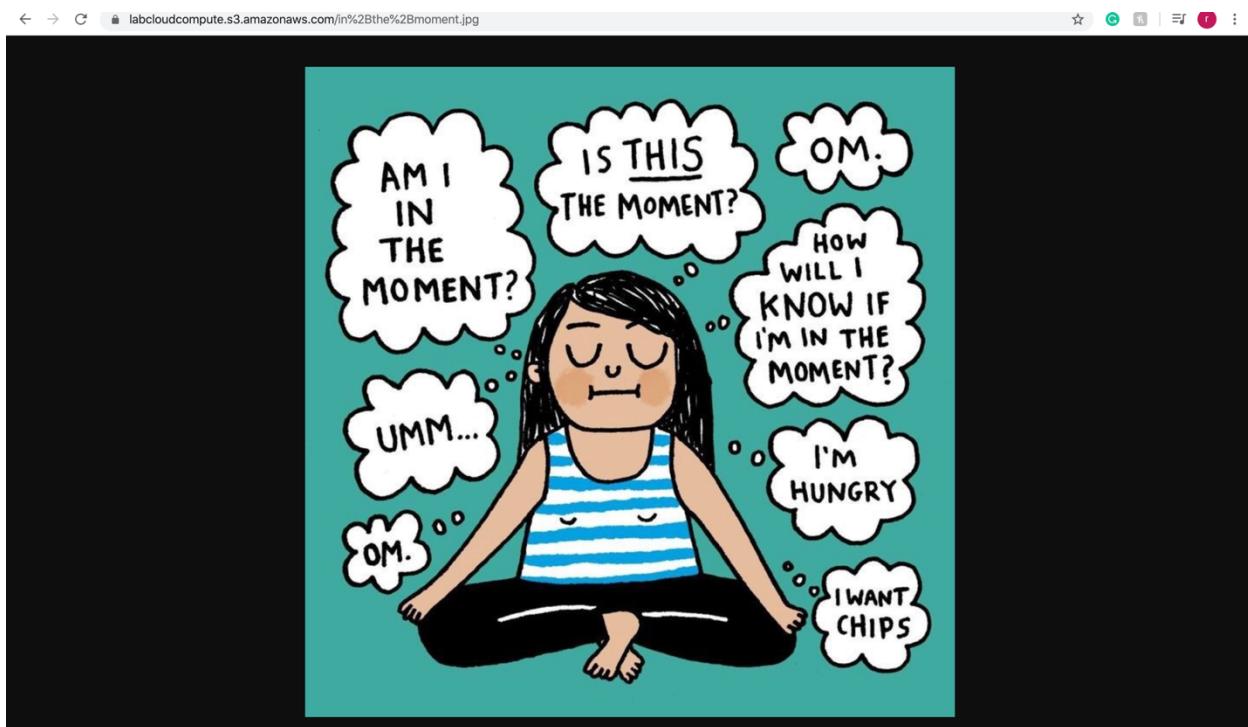
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



- In the created S3 uploading image and making it public.



- Clicking the object URL. We have successfully accessed.



2) Creating Web Distribution in Cloud Front:

- To create click on CloudFront as shown below.

The screenshot shows the AWS CloudFront service page. On the left, there's a sidebar with navigation links like History, CloudFront, S3, Console Home, EC2, VPC, and Systems Manager. The main content area has a search bar at the top. Below it, there are several sections: 'Migration & Transfer' (AWS Migration Hub, Application Discovery Service, Database Migration Service, Server Migration Service, AWS Transfer for SFTP, Snowball, DataSync), 'Networking & Content Delivery' (VPC, CloudFront, Route 53, API Gateway, Direct Connect, AWS App Mesh, AWS Cloud Map, Global Accelerator), 'Developer Tools' (CodeStar), and various other services like AWS Well-Architected Tool, Personal Health Dashboard, AWS Chatbot, Launch Wizard, AWS Compute Optimizer, Media Services (Elastic Transcoder, Kinesis Video Streams, MediaConnect, MediaConvert, MediaLive, MediaPackage, MediaStore, MediaTailor, Elemental Appliances & Software), Mobile (AWS Amplify, Mobile Hub, AWS AppSync, Device Farm), AR & VR (Amazon Sumerian), Machine Learning (Amazon SageMaker, Amazon CodeGuru, Amazon Comprehend, Amazon Forecast, Amazon Fraud Detector, Amazon Kendra, Amazon Lex, Amazon Machine Learning), Application Integration (Step Functions, Amazon EventBridge, Amazon MQ, Simple Notification Service, Simple Queue Service, SWF), and more. A message at the bottom says "Best practices to optimize Lambda@Edge with CloudFront. Learn more". Below that is the "Amazon CloudFront Getting Started" section with a "Create Distribution" button. The footer includes links for Feedback, English (US), Privacy Policy, and Terms of Use.

-When **Creating Distribution** there are various settings. The origin domain name we specified the S3 domain that was created, and it should show in the dropdown if we created the S3 from the same account. There is option allowing CloudFront to update the bucket policy so that CloudFront can read the objects from our bucket. There is an option to specify origin path if we put objects in a subfolder, but for this experiment leaving it blank since the object is in the root bucket. For the origin custom headers, we could provide a header name and origin or specified value that CloudFront will forward to the origin on each request, useful for custom origins that would like to know which origins came from where.

- **Default Cache Behavior Settings**, we have the option of selecting **HTTP and HTTPS**, **Redirect HTTP to HTTPS**, we could support requests using either protocol or use **HTTPS only** for which the HTTP traffic would get dropped. CloudFront allows us to specify which HTTP methods we want to by default it is **GET HEAD**, allowing users to read, write or both to the user. And since we are only doing read only operations for this lab, we can leave it as default. In the **Cache** based on selected we can specify which HTTP header we'd want to forge to the origin, recommended is Whitelist just the headers that our origin will care about, like when the origin is looking for specific HTTP headers to decide which object to return. If we forward all **HTTP headers** CloudFront will actually bypass the caching layers and not attempt to cache the object at all. But since S3 does not look at headers we can just let it be none. The **TTL** (time to leave) that we want the object to stay in cache, we can customize as well, for this lab leaving at default relying on the cache control headers returned from the origin. specifies the cached object lifespan, specified in seconds and the default Max value is for one-year minimum cache life is 0 and default time period is 24 hours. **Lambda** function can call a function on particular action for post process, to manipulate the request or the response.

-**Distribution Settings**, have the ability to specify which parts of network we'd want to use, we could specify a lower price class which would reduce CloudFront costs and restrict viewers to using just edge locations within the regions. This does not mean viewers outside could not access the content it just means they get routed to the nearest edge location within the specified regions. Default root object – URL for default root object. HTTP version Aws also supports HTTP2.0, we can specify a couple of accessing protocols for our objects. IPv6 enable or disable access via IPv6

The screenshot shows the 'Create Distribution' wizard on the AWS CloudFront console. The current step is 'Step 2: Create distribution'. The 'Origin Settings' section is active, showing fields for Origin Domain Name (abcloudcompute.s3.amazonaws.com), Origin Path (empty), Origin ID (S3-labcloudcompute), and Restrict Bucket Access (Yes). It also includes sections for Origin Access Identity (Create a New Identity selected), Comment (access-identity-labcloudcompute), and Grant Read Permissions on Bucket (Yes, Update Bucket Policy selected). The 'Default Cache Behavior Settings' section is partially visible below.

← → C https://console.aws.amazon.com/cloudfront/home?region=us-east-1#create-distribution: ☆ G F E ⌂ ⓘ

AWS Services Resource Groups Global Support

Step 1: Select delivery method Step 2: Create distribution

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.](#)

Supported HTTP Versions HTTP/2, HTTP/1.1, HTTP/1.0 HTTP/1.1, HTTP/1.0

Default Root Object ⓘ

Logging On Off

Bucket for Logs ⓘ

Log Prefix ⓘ

Cookie Logging On Off

Enable IPv6 [Learn more](#)

Comment ⓘ

Distribution State Enabled Disabled

Cancel Back Create Distribution

3) The distribution is deployed its domain name is given.

← → C https://console.aws.amazon.com/cloudfront/home?region=us-east-1#distributions: ☆ G F E ⓘ

AWS Services Resource Groups Global Support

CloudFront

Distributions What's new *

Reports & analytics Cache statistics Monitoring Alarms Popular objects Top referrers Usage Viewers

Security Origin access identity Public key Field-level encryption

Enable new real-time metrics for better visibility of your traffic. [Learn more](#)

CloudFront Distributions

Create Distribution Distribution Settings Delete Enable Disable

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	ETQOSA5W408UY	d1afb1a3jpft9.clo	-	labcloudco	-	In Progress	Enabled	2020-04-18 19:00

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

4) Going back to S3 bucket and disable the public read access.

The screenshot shows the AWS S3 console with the 'labcloudcompute' bucket selected. The 'Permissions' tab is active. Under 'Block public access (bucket settings)', the 'Block all public access' checkbox is checked. Below it, four sub-options are listed, each with a descriptive tooltip:

- 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.

At the bottom right of the dialog are 'Cancel' and 'Save' buttons. The footer of the page includes links for Feedback, English (US), Copyright notice (© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.), Privacy Policy, and Terms of Use.

5) Once done that we cannot access the uploaded image in the S3 bucket as previously through object URL.

The screenshot shows a browser window displaying a 403 Access Denied error for the URL `labcloudcompute.s3.amazonaws.com/in%2Bthe%2Bmoment.jpg`. The error message is: "This XML file does not appear to have any style information associated with it. The document tree is shown below." Below the message is the XML error response:

```
<Error>
<Code>AccessDenied</Code>
<Message>Access Denied</Message>
<RequestId>7841CE1425A6992D</RequestId>
<HostId>
    YON/65uP41G8UrIkHRNX+juxabt5kg4f3WodeJXue0NSFG+cSlyobNTtCaoa+c7b6hLGSpEox+g=
</HostId>
</Error>
```

- We have blocked all the public access to this S3 bucket and removed the read permission from public access suggesting that all the public cannot access the objects in the bucket as of, now this is the reason originally it is not recommended to make the objects public and a notification also

pops up when we tried to make it public. The reason is that anyone would be able to view files also some sensitive datas by anyone.

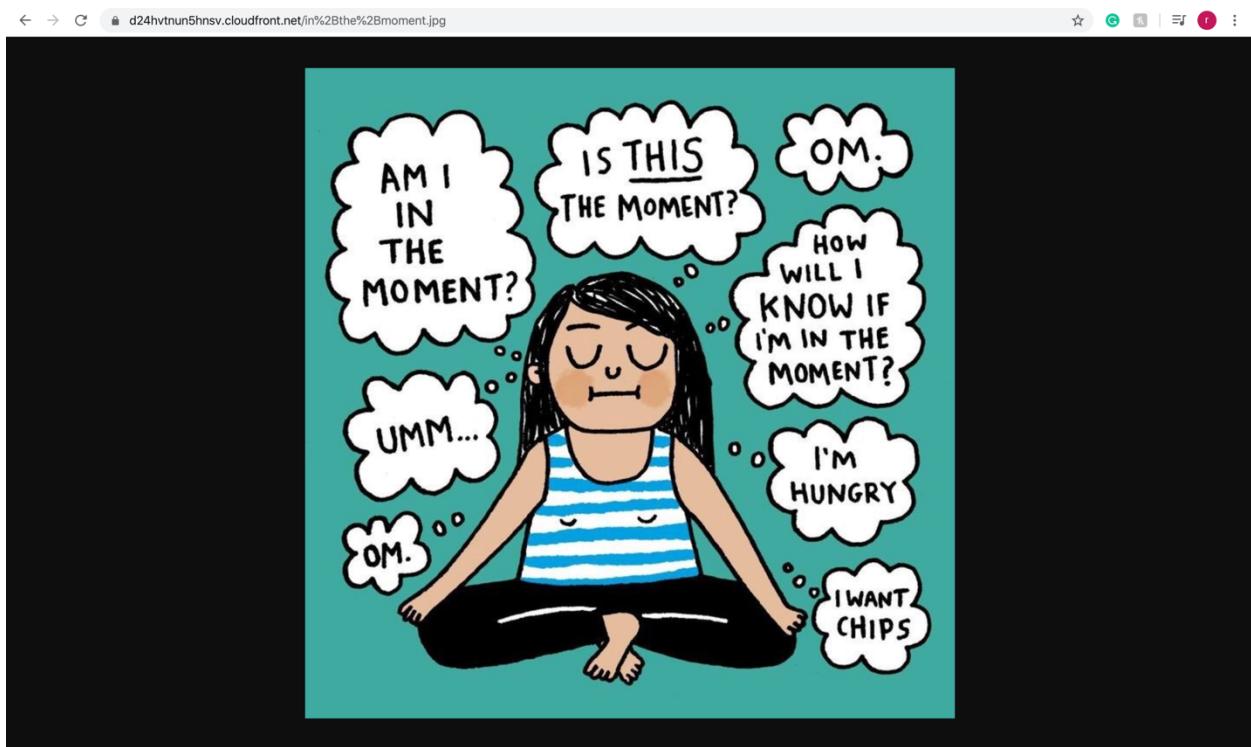
6) Change the object URL:

<https://labcloudcompute.s3.amazonaws.com/in%2Bthe%2Bmoment.jpg>

with distribution domain such as;

<https://d24hvtnun5hnsv.cloudfront.net?in%2Bthe%2Bmoment.jpg>

-We can now access the image through the created distribution network and ensure that this service can read objects in S3 bucket, we did it to see that we can access the objects using CloudFront distribution using the distribution URL.



Amazon S3 (Simple Storage Service) is a better and safe way for storing, managing and distributing data and object storage, from Amazon EC2 or anywhere on the web. Choosing the region where we want the data to be stored and Amazon S3 automatically makes copies of objects on multiple devices across multiple facilities, paying for the storage we actually use. There is no monthly charge and you only pay for what you use, so you don't have to worry about any long-term commitment. S3 does the storage and one of the things that we would want to do is to migrate our images or videos out of our server and serve them from a Content Delivery Network (CDN). The major benefit of a CDN is that it can serve our web content from the location closest to your reader, and hence speeding up the loading of site while reducing sever bandwidth usage and also rank well in the search engine.

Amazon CloudFront is the CDN providers that is well integrated with Amazon S3 and best suited. For it to be used as a CDN have to activate CloudFront and configure Amazon S3 with it. When we are using Cloud front distribution, the case remains true when we are accessing the object for the first time I.e. case when the image is not cached in the edge location. However, the load speed difference changes drastically once any of the user has requested the resource, the reason because now that object has been cached on the edge location closer to the user. The image or object can be accessed from anywhere, if the object available in US and if the request is coming from part of Asia then the request goes halfway around the world making the use of bucket URL slow and lagging can be seen. But that is not case as explained above when using the CloudFront Distribution