# AWS WITH GUI

Previously, I have shown configuration of webserver on AWS using CLI. This repo deals with the same project, but now, with the help of GUI.

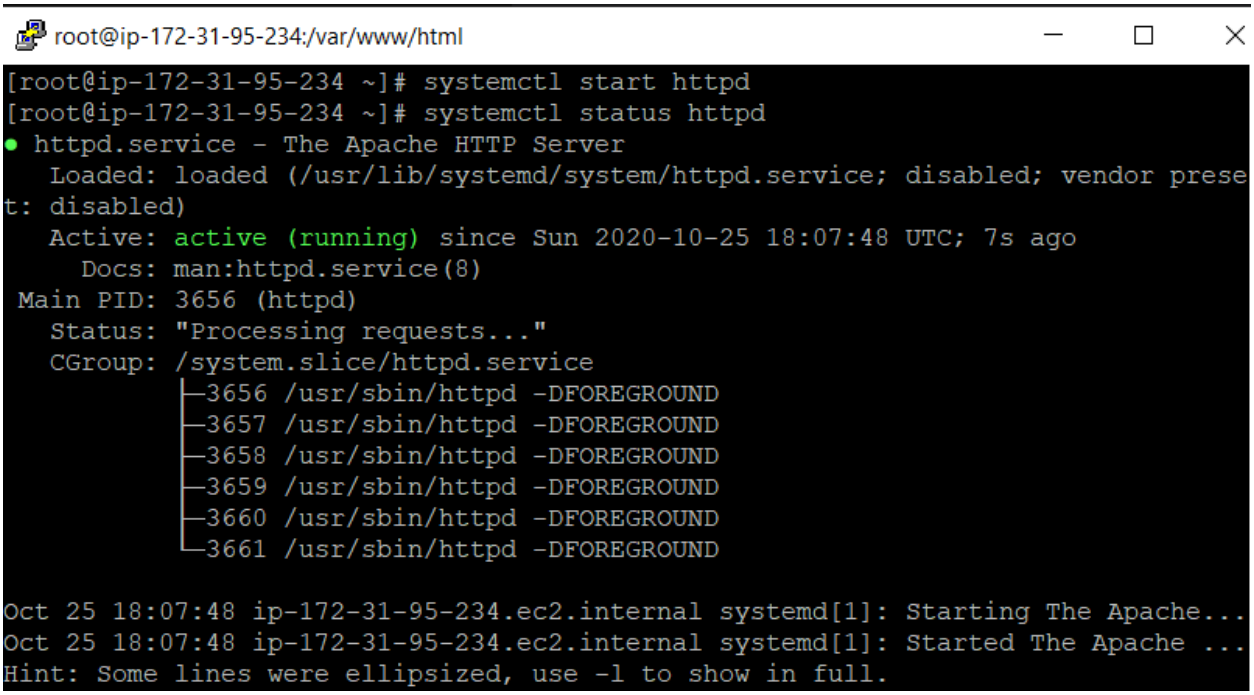
For this, we need to launch an ec-2 instance on aws and get the public IP. Later, we can connect to that instance using putty. We need the public ip and the key in .ppm format to login via putty.

When you login to putty, you land on the ec2-user directory. To perform our function, we need to move to the root directory. We can do that by using the following command sudo su – root

Using this command, you land to the root directory. If httpd is not installed in your system, first install it using yum install httpd

Now, we need to start httpd and check its status. This can be done using the following command simultaneously.

systemctl start httpd
systemctl staus httpd



## 1. Creating partition

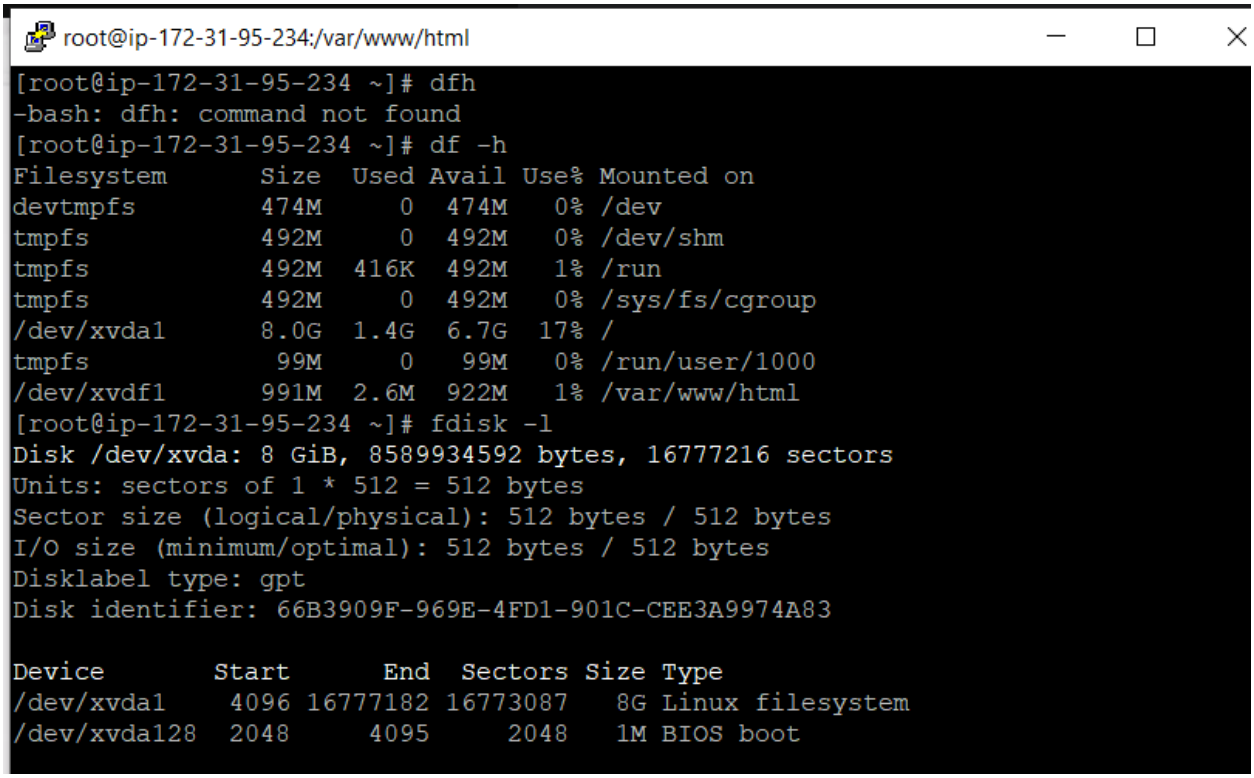
root@ip-172-31-95-234:/var/www/html

```
[root@ip-172-31-95-234 html]# fdisk /dev/xvdf

Welcome to fdisk (util-linux 2.30.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x436f9319.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-2097151, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-2097151, default 2097151):

Created a new partition 1 of type 'Linux' and of size 1023 MiB.
```

## 2. Format

root@ip-172-31-95-234:/var/www/html

```
[root@ip-172-31-95-234 html]# mkfs.ext4 /dev/xvdf1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
65536 inodes, 261888 blocks
13094 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=268435456
8 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```




### 3. Mount

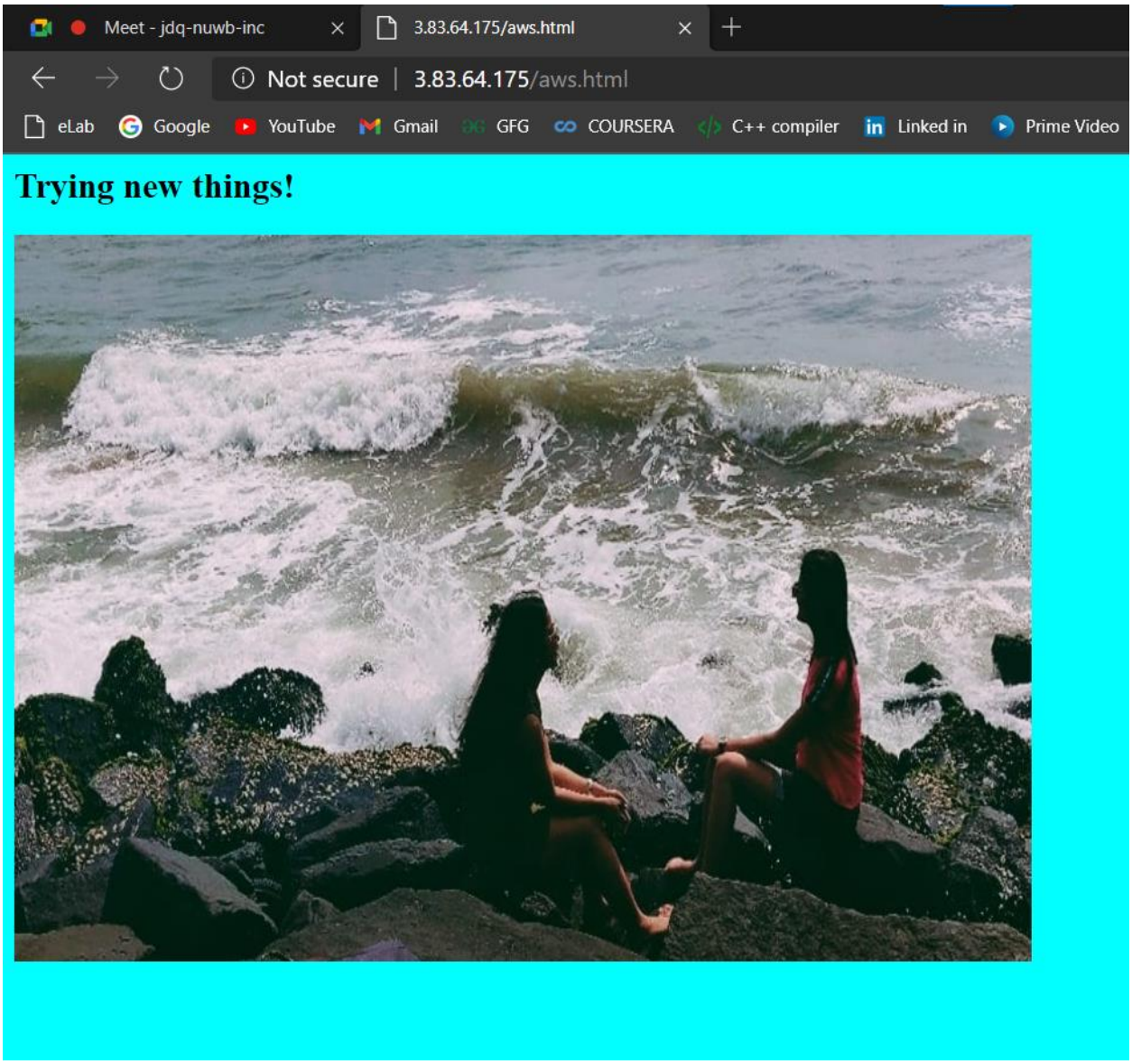
```
[root@ip-172-31-95-234 html]# mount /dev/xvdf1 /var/www/html
```

```
[root@ip-172-31-95-234 html]# ls
lost+found
```

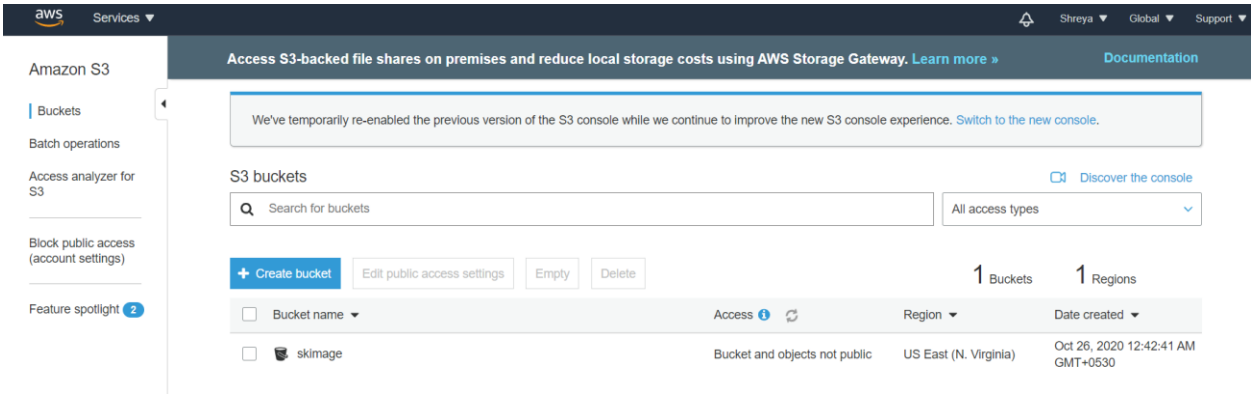
4. **Transfer req image and write html code**

```
[root@ip-172-31-95-234 html]# vi aws.html
[root@ip-172-31-95-234 html]# ls
aws.html  lost+found  sk.jpeg
[root@ip-172-31-95-234 html]# ls
```

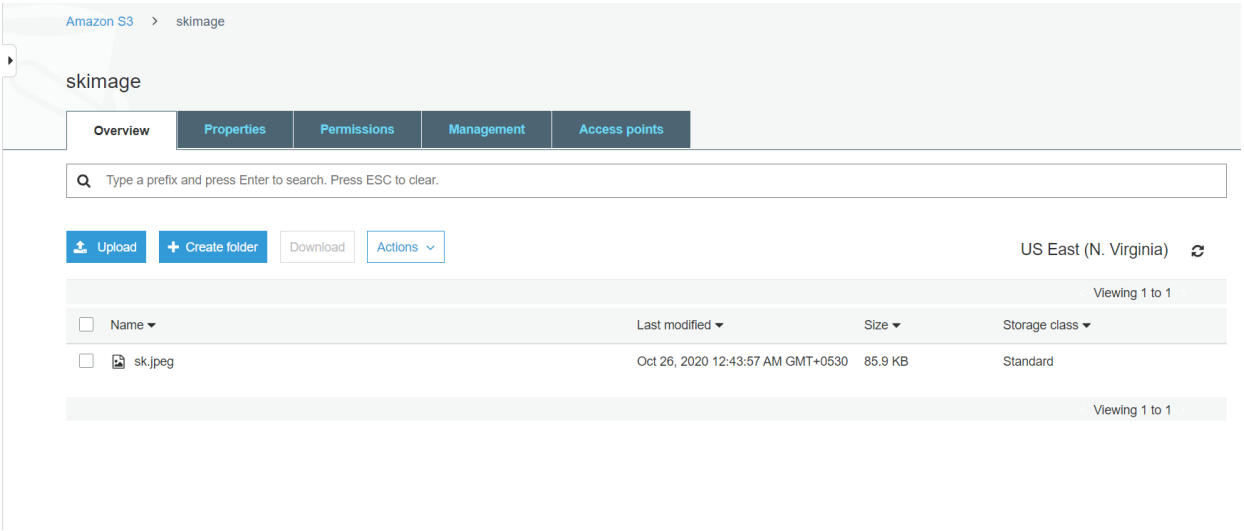
5. **To view the file, type your public ip and file name on browser.**



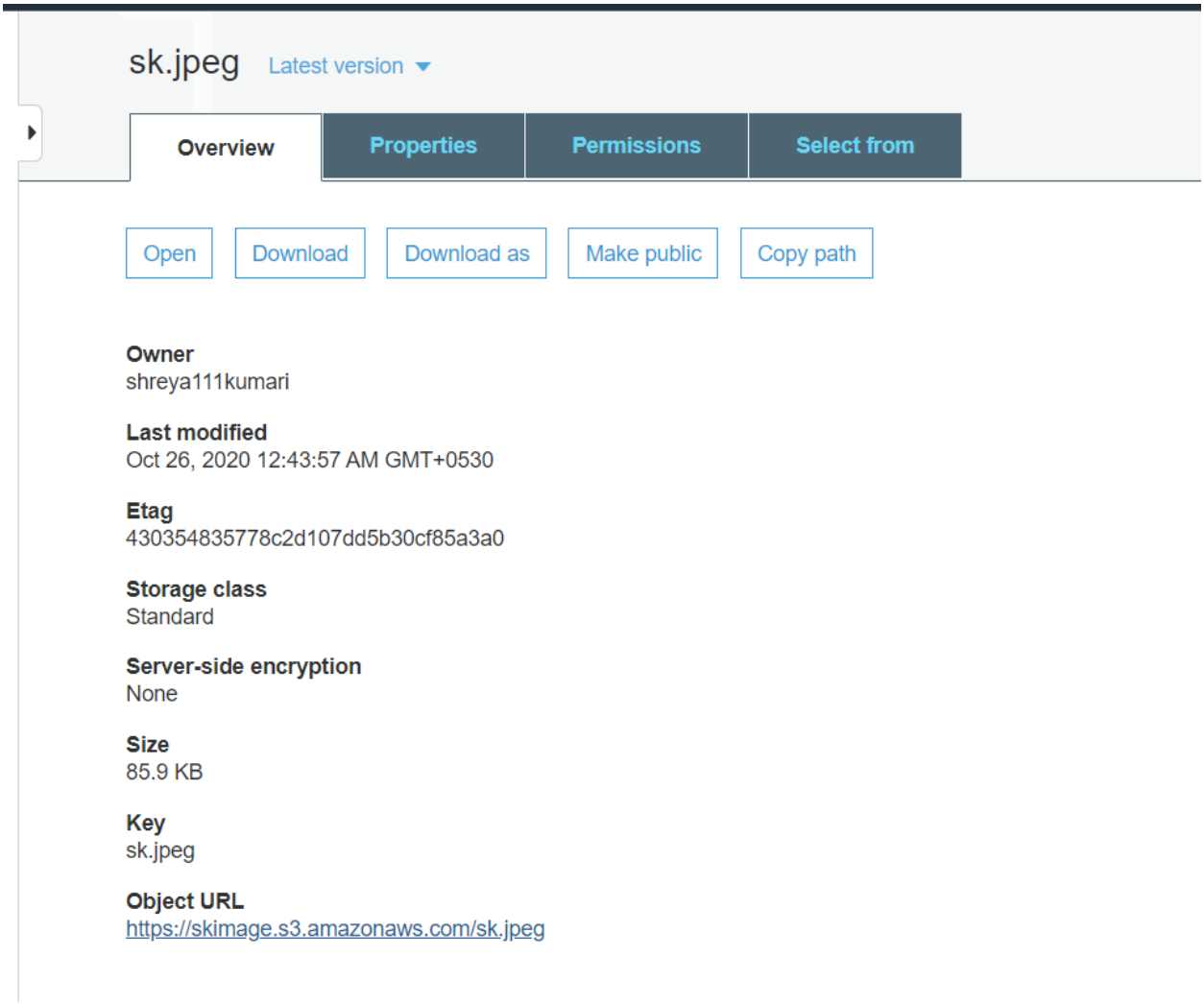
## 6. Create S3 bucket




## 7. Add object to the bucket

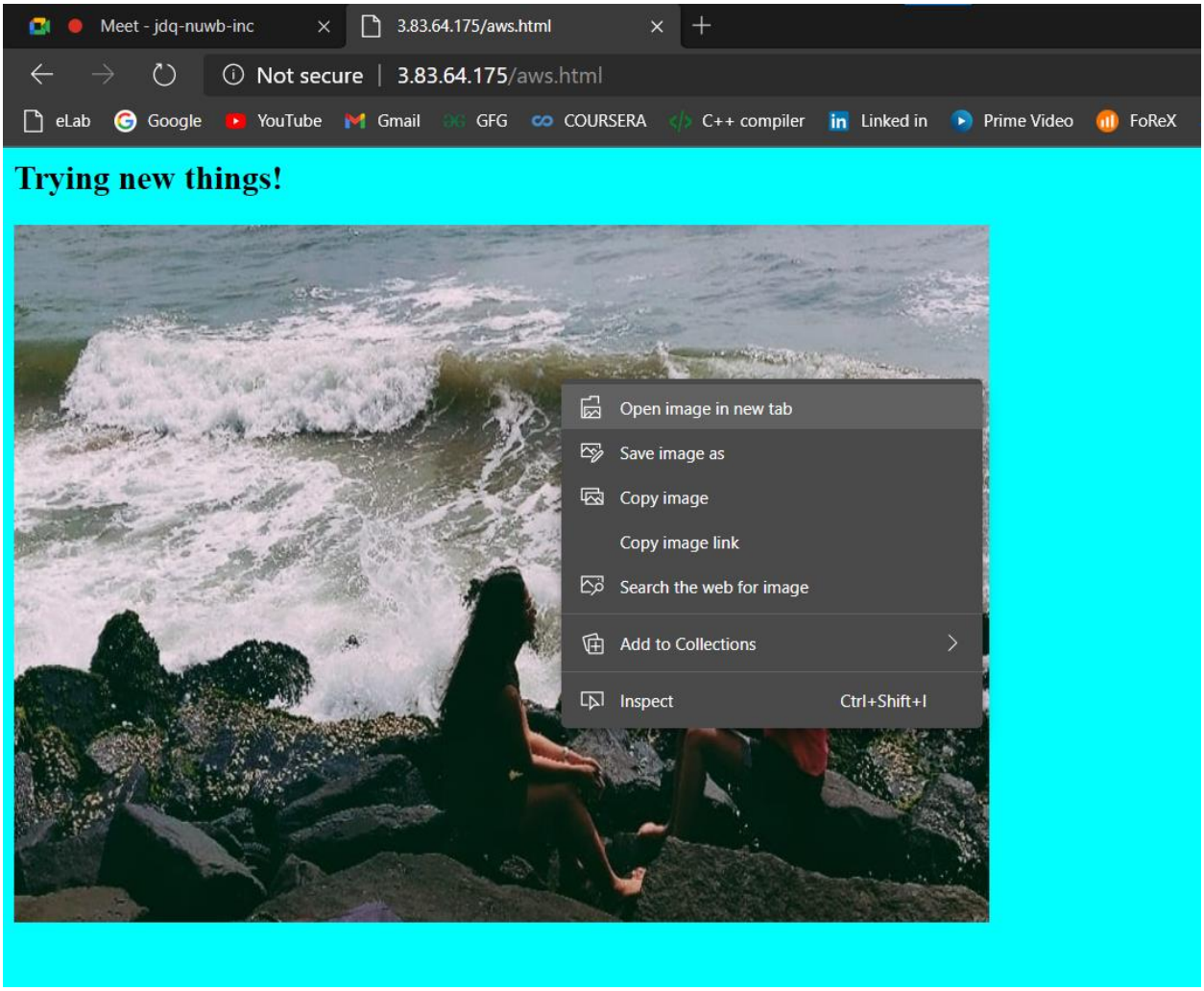


## 8. Get the url of the image

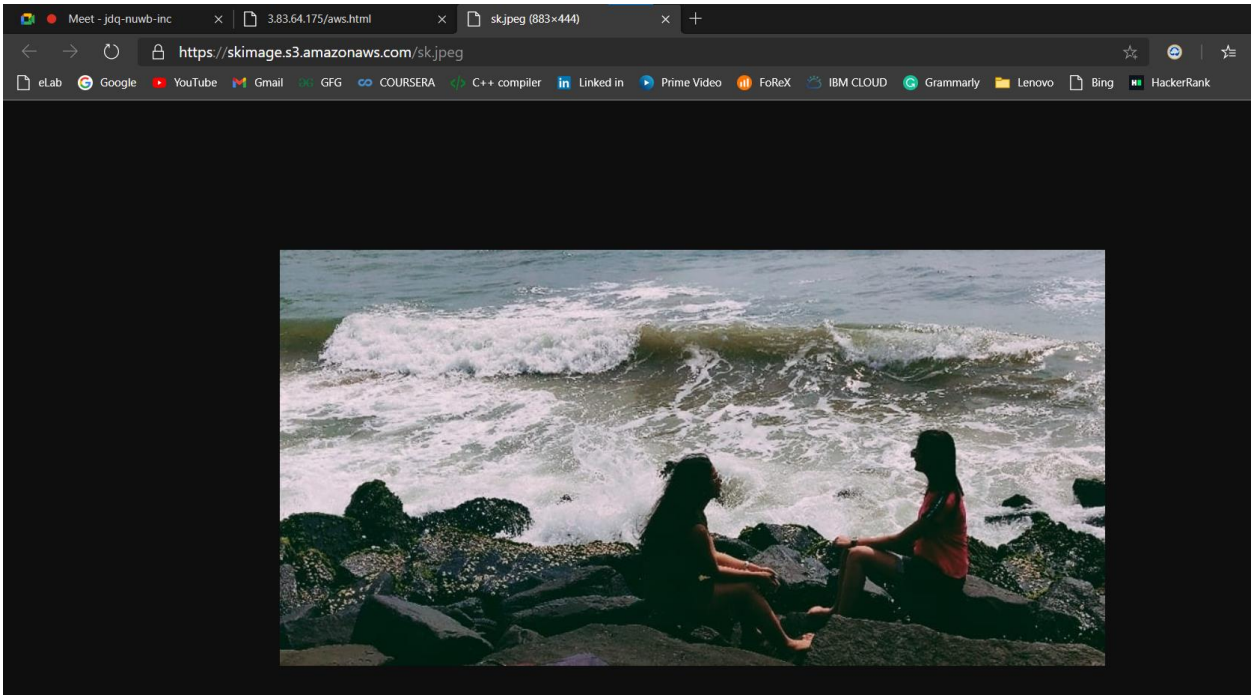



## 9. Update the image url in img src

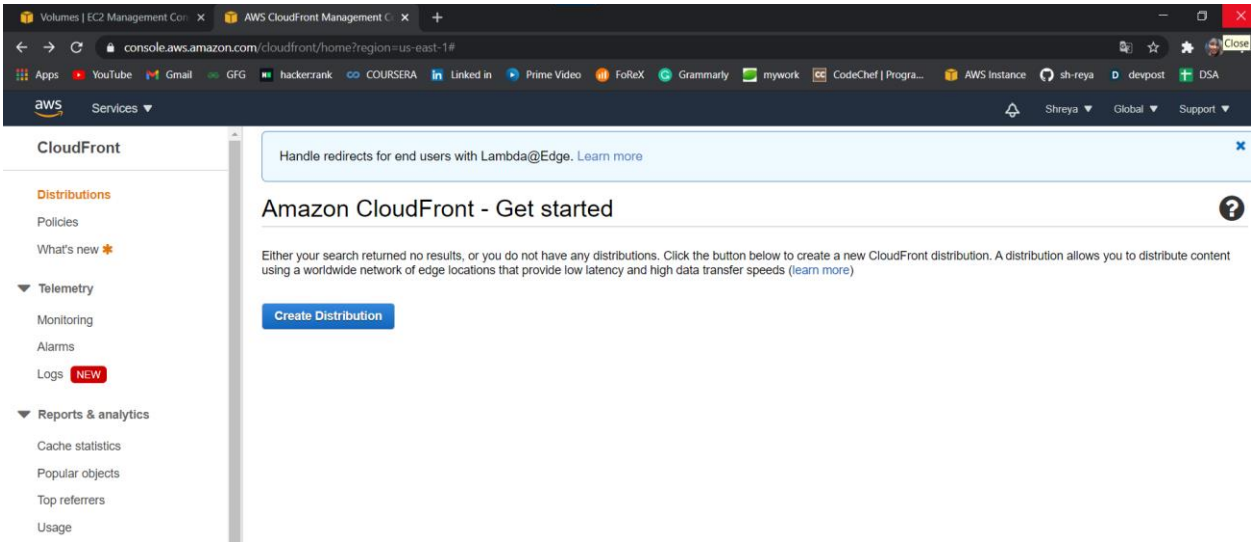
## 10. Check on browser

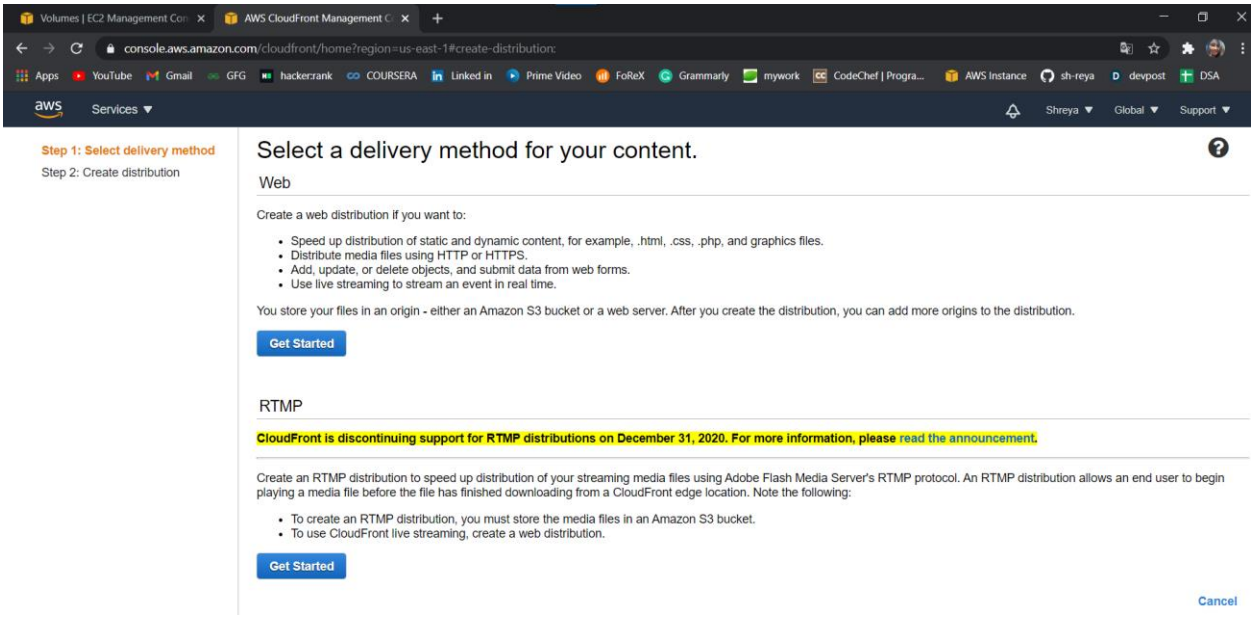


## 11. Check img url to verify the S3 url

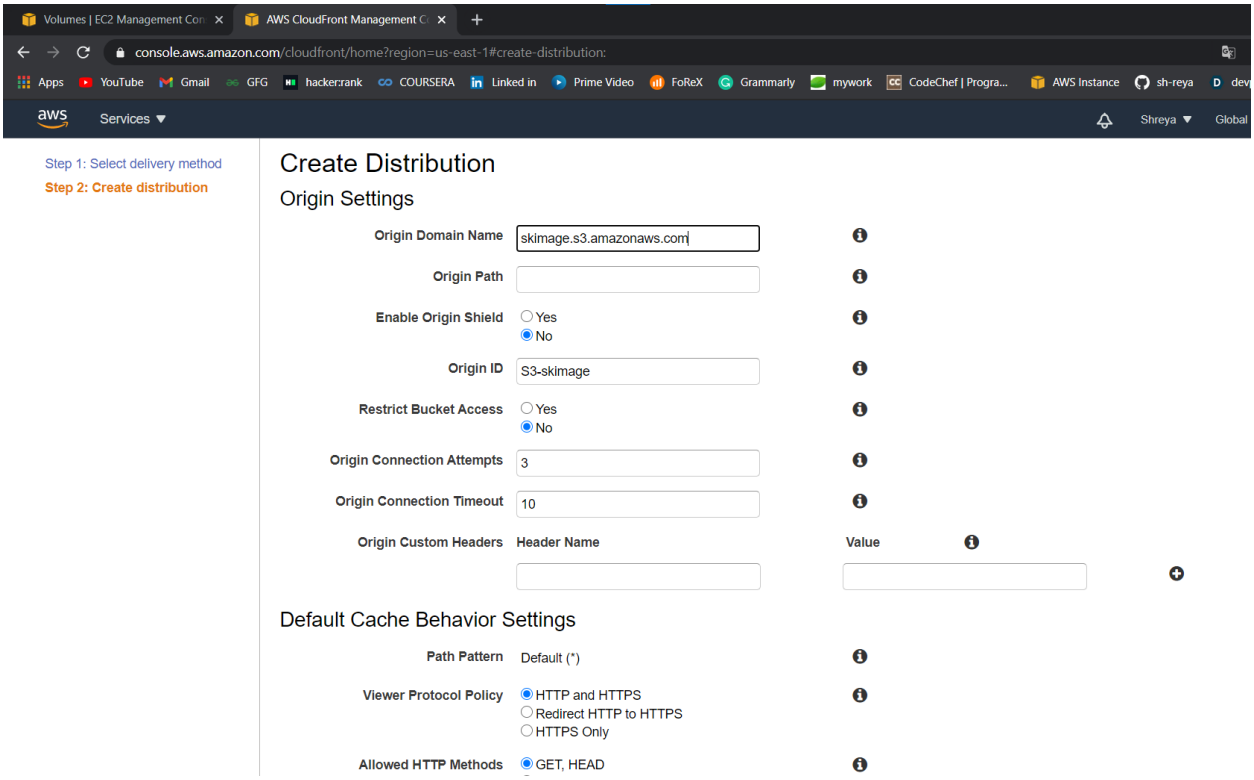


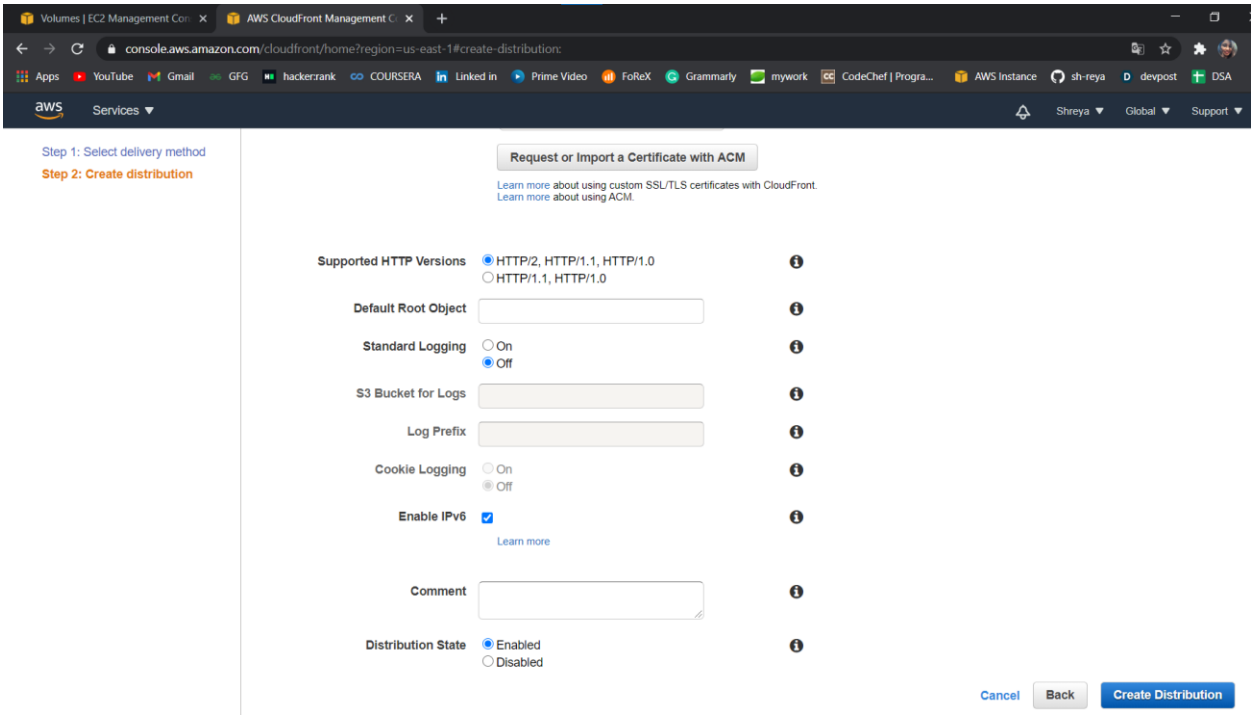

## 12. Moving towards cloudfront



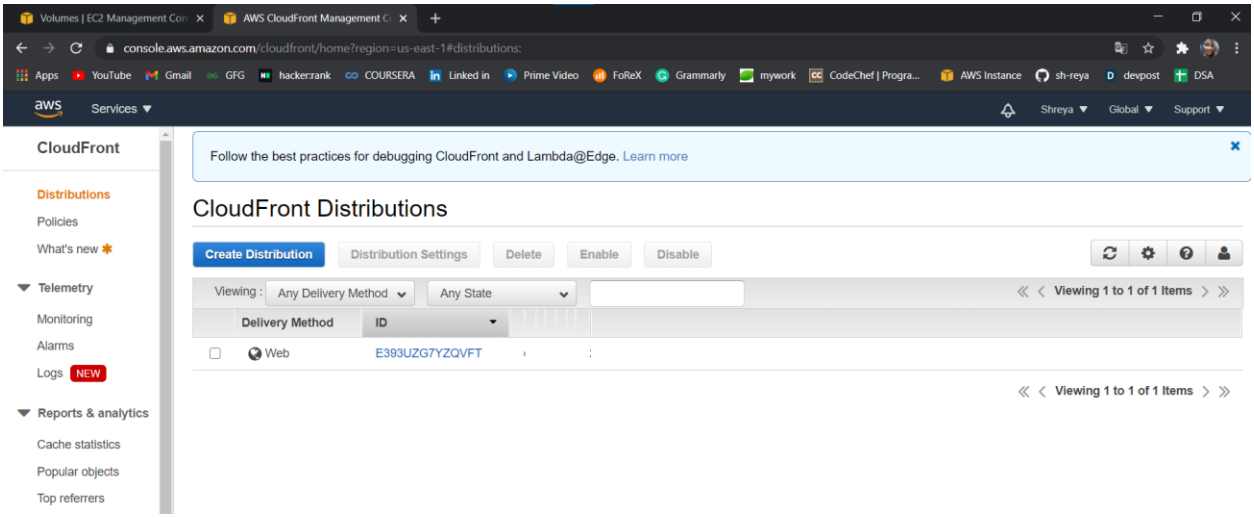



## 13. Provide the url of the S3 bucket

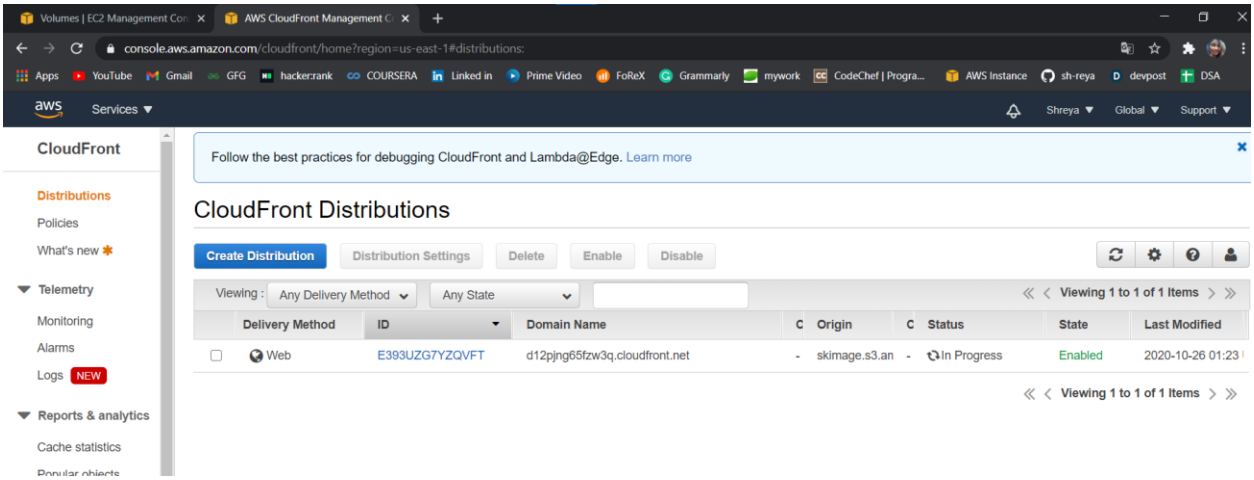




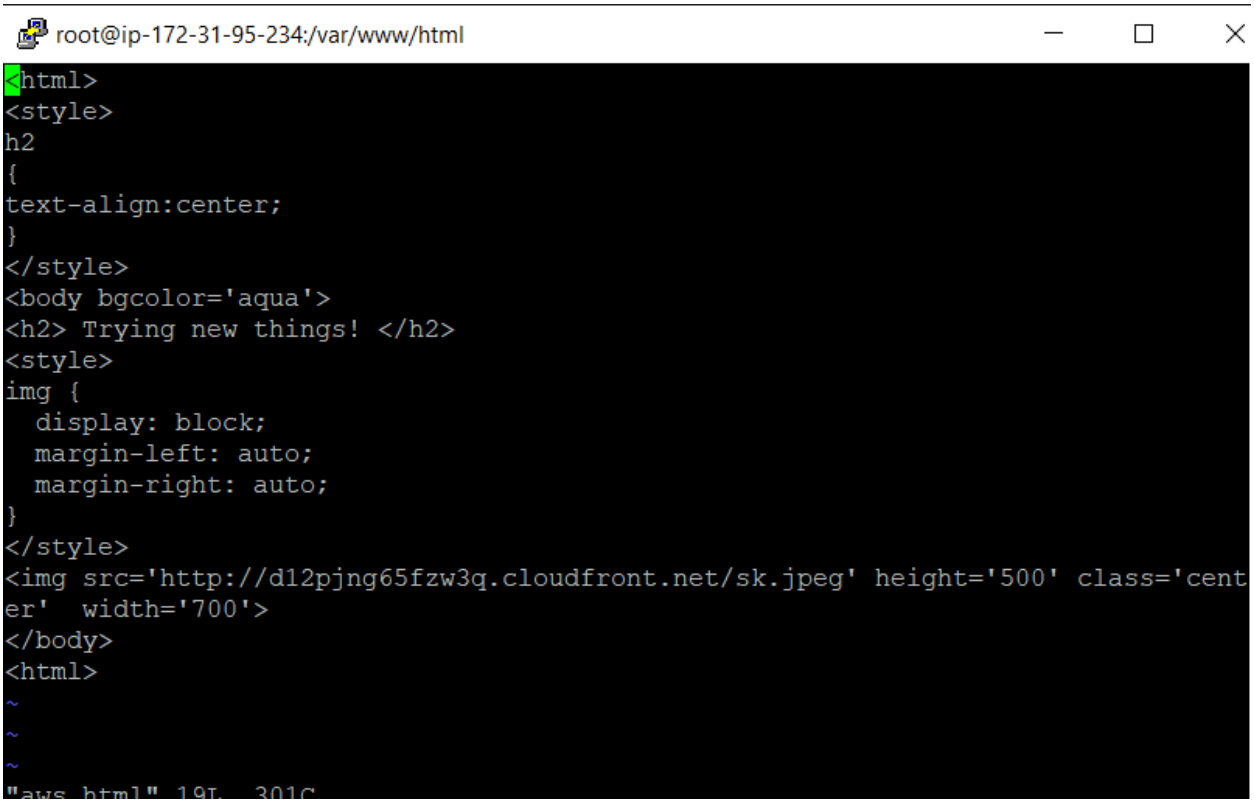

## 14. Click on create distribution button to create distribution



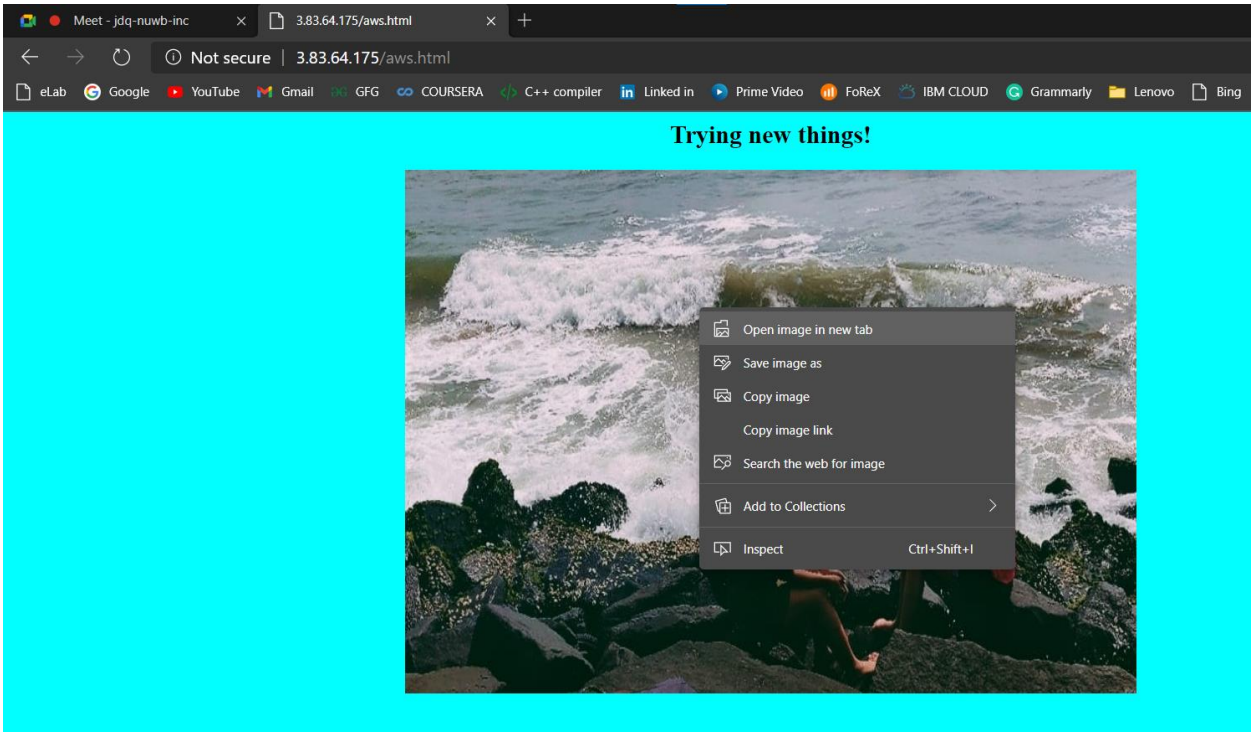
## 15. Get the CloudFront Distribution ID




## 16. Update the id in img src



## 17. Open the webapp in the browser




## 18. Open image in new tab to confirm the url of the image