

Created by:- Abhishek Kumar

## TASK DESCRIPTION:

- ✚ 📖 Create High Availability Architecture with AWS CLI
- ✚ 📖 Webserver configured on EC2 Instance using cli–

- ❖ Document Root(/var/www/html) made persistent by mounting on EBS Block Device.
  - Static objects used in code such as pictures stored in S3 –
- ❖ Setting up Content Delivery Network using CloudFront and using the origin domain as S3 bucket. - Finally place the Cloud Front URL on the webapp code for security and low latency.

At first, I'll create New instance  
then create a storage of 1 GB  
then

Partition , formatting,mounting

Creating S3 bucket via cli and make it public

Putting object in s3-bucket and make it public via cli

Creating Cloud-Front Distribution via cli.

## LAUNCHING AN INSTANCE

```
C:\Users\Abhishek kumar>aws ec2 run-instances --image-id ami-0e306788ff2473ccb --instance-type t2.micro --count 1 --subnet-id subnet-722a231a --security-group-ids sg-02d2c18e59a9aac97 --key-name KeyHadoop
An error occurred (InvalidKeyPair.NotFound) when calling the RunInstances operation: The key pair 'KeyHadoop' does not exist

C:\Users\Abhishek kumar>aws ec2 run-instances --image-id ami-0e306788ff2473ccb --instance-type t2.micro --count 1 --subnet-id subnet-722a231a --security-group-ids sg-02d2c18e59a9aac97 --key-name 23rdsept1
{
  "Groups": [],
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "ami-0e306788ff2473ccb",
      "InstanceId": "i-027e06b1cd5d5b6df",
      "InstanceType": "t2.micro",
      "KeyName": "23rdsept1",
      "LaunchTime": "2020-10-26T17:19:29+00:00",
      "Monitoring": {
        "State": "disabled"
      },
      "Placement": {
        "AvailabilityZone": "ap-south-1a",

```

On aws website you can see, the instance has been created.

|  | Name          | Instance   | Inst... | Avai... | Instance | Status Checks | Alarm Status | Public DNS (IPv4)        | IPv4 Public IP | IPv6 IPs | Key Name  | Monitoring | Launch Time                    | Security Groups  | Owner        |
|--|---------------|------------|---------|---------|----------|---------------|--------------|--------------------------|----------------|----------|-----------|------------|--------------------------------|------------------|--------------|
|  |               | i-027e0... | t2...   | ap...   | run...   | Initializing  | None         | ec2-13-235-254-216.ap... | 13.235.254.216 | -        | 23rdsept1 | disabled   | October 26, 2020 at 10:49:2... | launch-wizard-10 | 946697907226 |
|  | top notch ... | i-0780a... | t2...   | ap...   | sto...   |               | None         |                          | -              | -        | keyHadoop | disabled   | October 25, 2020 at 11:09:4... | launch-wizard-10 | 946697907226 |
|  | 22nd sept-dn  | i-0e183... | t2...   | ap...   | sto...   |               | None         |                          | -              | -        | keyHadoop | disabled   | October 25, 2020 at 11:07:5... | launch-wizard-1  | 946697907226 |

Command:- `aws ec2 run-instances --image-id ami-0e306788ff2473ccb --instance-type t2.micro --count 1 --subnet-id subnet-884872e0 --security-group-ids sg-02fb764381c3dd016 --key-name keyhadoop`

## STARTED THAT INSTANCE

```
C:\Users\Abhishek kumar>aws ec2 start-instances --instance-ids i-027e06b1cd5d5b6df
usage: aws [options] <command> [<subcommand> ...] [parameters]
To see help text, you can run:

aws help
aws <command> help
aws <command> <subcommand> help
aws: error: the following arguments are required: --instance-ids

C:\Users\Abhishek kumar>aws ec2 start-instances --instance-ids i-027e06b1cd5d5b6df
{
  "StartingInstances": [
    {
      "CurrentState": {
        "Code": 0,
        "Name": "pending"
      },
      "InstanceId": "i-027e06b1cd5d5b6df",
      "PreviousState": {
        "Code": 80,
        "Name": "stopped"
      }
    }
  ]
}
```

## CREATED VOLUME OF SIZE 1GB

```
C:\Users\Abhishek kumar>aws ec2 create-volume --volume-type gp2 --size 1 --availability-zone ap-south-1a
{
  "AvailabilityZone": "ap-south-1a",
  "CreateTime": "2020-10-26T18:00:29+00:00",
  "Encrypted": false,
  "Size": 1,
  "SnapshotId": "",
  "State": "creating",
  "VolumeId": "vol-064516ae1fb6adb53",
  "Iops": 100,
  "Tags": [],
  "VolumeType": "gp2"
}

C:\Users\Abhishek kumar>
```

You can check the Volume created:-

| Name | Volume ID        | Size   | Volume Type | IOPS | Snapshot        | Created                | Availability Zone | State     | Alarm Status | Attachment |
|------|------------------|--------|-------------|------|-----------------|------------------------|-------------------|-----------|--------------|------------|
|      | vol-0c6b935...   | 1 GiB  | gp2         | 100  |                 | October 26, 2020 at... | ap-south-1a       | available | None         |            |
|      | vol-06a22c0...   | 8 GiB  | gp2         | 100  | snap-027b63b... | October 25, 2020 at... | ap-south-1a       | in-use    | None         | i-0780a827 |
|      | vol-0c642e9...   | 10 GiB | gp2         | 100  | snap-0eed902... | September 22, 202...   | ap-south-1a       | in-use    | None         | i-0e18362f |
|      | vol-0d4b5e9...   | 8 GiB  | gp2         | 100  | snap-027b63b... | October 26, 2020 at... | ap-south-1a       | in-use    | None         | i-027e00b1 |
|      | vol-0fffe5428... | 1 GiB  | gp2         | 100  |                 | October 26, 2020 at... | ap-south-1a       | in-use    | None         | i-0780a827 |

The Available one

## NOW, WE HAVE TO ATTACH THE VOLUME OF 1 GB

```
Command Prompt
aws <command> <subcommand> help

Unknown options: i-027e06b1cd5d5b6df

C:\Users\Abhishek kumar>aws ec2 attach-volume --instance-id i-027e06b1cd5d5b6df --volume-id vol-0c6b935df280b03c5 --device /dev/xvdf
{
  "AttachTime": "2020-10-26T18:10:09.953000+00:00",
  "Device": "/dev/xvdf",
  "InstanceId": "i-027e06b1cd5d5b6df",
  "State": "attaching",
  "VolumeId": "vol-0c6b935df280b03c5"
}
```

```
aws ec2 attach-volume --instance-id i-0a06ad006e50cf731 --volume-id vol-_____ --device /dev/xvdf
```

Now, to check that volume we first Login to SSh client

```

root@ip-172-31-38-34:~
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Mon Oct 26 17:26:41 2020 from 106.208.193.90

 _ | _ | _ )
 _ | ( _ /   Amazon Linux 2 AMI
 _ | \ _ | _ |

https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 13 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-38-34 ~]$ sudo su - root
[root@ip-172-31-38-34 ~]# fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 66B3909F-969E-4FD1-901C-CEE3A9974A83

Device            Start      End  Sectors Size Type
/dev/xvda1        4096 16777182 16773087   8G Linux filesystem
/dev/xvda128       2048      4095     2048   1M BIOS boot

Partition table entries are not in disk order.

Disk /dev/xvdf: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[root@ip-172-31-38-34 ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda        202:0    0    8G  0 disk
└─xvda1     202:1    0    8G  0 part /
xvdf        202:80   0    1G  0 disk
[root@ip-172-31-38-34 ~]# █

```

Now use command prompt to login to Aws account via IAM user with **Aws configure** command

```

C:\> Command Prompt

Microsoft Windows [Version 10.0.19042.541]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Abhishek kumar>aws configure
AWS Access Key ID [*****IWPY]:
AWS Secret Access Key [*****Vach]:
Default region name [ap-south-1]:
Default output format [None]:

C:\Users\Abhishek kumar>_

```

## SSH LOGIN VIA COMMAND PROMPT

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo fdisk /dev/  
  
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 0xd3deb20d.  
  
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): +500M  
  
Created a new partition 1 of type 'Linux' and of size 500 MiB.  
  
Command (m for help): w  
The partition table has been altered.  
Calling ioctl() to re-read partition table.  
Syncing disks.
```

## PARTITION COMPLETED DOWN

```
C:\Users\Abhishek kumar>cd Desktop  
  
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo 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 0xd3deb20d.  
  
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): +500M  
  
Created a new partition 1 of type 'Linux' and of size 500 MiB.  
  
Command (m for help): w  
The partition table has been altered.  
Calling ioctl() to re-read partition table.  
Syncing disks.  
  
C:\Users\Abhishek kumar\Desktop>
```

## FORMAT DONE DOWN

```
ec2-user 35.154.172.164 -i 23rdsept1.pem sudo mkfs.ext4 /dev/xvdf
```

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo mkfs.ext4 /dev/xvdf
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, 262144 blocks
13107 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 (8192 blocks): done
Writing superblocks and filesystem accounting information: done

C:\Users\Abhishek kumar\Desktop>
```

## INSTALL APACHE HTTPD

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.46-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.46-1.amzn2 for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.46-1.amzn2 for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.46-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.46-1.amzn2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.6.3-5.amzn2.0.2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.46-1.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.46-1.amzn2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.14-2.amzn2 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution
```

# MOUNT THAT VOLUME

```
ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo mount /dev/xvdf /var/www/html
```

```
mailcap.noarch 0:2.1.41-2.amzn2
mod_http2.x86_64 0:1.15.14-2.amzn2

Complete!

C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo mount /dev/xvdf1 /var/www/html

C:\Users\Abhishek kumar\Desktop>
```

Systemctl start httpd

Systemctl status httpd

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo mount /dev/xvdf1 /var/www/html

C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo systemctl start httpd

C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Thu 2020-10-29 07:55:46 UTC; 8s ago
     Docs: man:httpd.service(8)
  Main PID: 14571 (httpd)
    Status: "Processing requests..."
    CGroup: /system.slice/httpd.service
            └─14571 /usr/sbin/httpd -DFOREGROUND
              └─14572 /usr/sbin/httpd -DFOREGROUND
                └─14573 /usr/sbin/httpd -DFOREGROUND
                  └─14574 /usr/sbin/httpd -DFOREGROUND
                    └─14575 /usr/sbin/httpd -DFOREGROUND
                      └─14576 /usr/sbin/httpd -DFOREGROUND

Oct 29 07:55:46 ip-172-31-38-34.ap-south-1.compute.internal systemd[1]: Starting The Apache HTTP Server...
Oct 29 07:55:46 ip-172-31-38-34.ap-south-1.compute.internal systemd[1]: Started The Apache HTTP Server.

C:\Users\Abhishek kumar\Desktop>
```

## Login in instance via cli :

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem
```

```
C:\Users\Abhishek kumar\Desktop>ssh -l ec2-user 35.154.172.164 -i 23rdsept1.pem
Last login: Thu Oct 29 07:46:17 2020 from ec2-13-233-177-0.ap-south-1.compute.amazonaws.com
Last login: Thu Oct 29 07:46:17 2020 from ec2-13-233-177-0.ap-south-1.compute.amazonaws.com

  _| _|_ )
  _| ( /   Amazon Linux 2 AMI
  _|\_|_|_|

https://aws.amazon.com/amazon-linux-2/
26 package(s) needed for security, out of 40 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-38-34 ~]$
```

*Sudo su – root*

*Go inside html*

*Created file of html*

*Viewed by cat command*

*To run that webpage in web browser*

<http://35.154.172.164/task.html>

<ip>/filename

```
[root@ip-172-31-38-34 ~]# cd /var/www/html
[root@ip-172-31-38-34 html]# ls
lost+found
[root@ip-172-31-38-34 html]# vi task.html
[root@ip-172-31-38-34 html]# cat task.html
<html>
<body>
<h1>PAge of AWS via CLI</h1>
<h2>by Abhishek</h2>
</body>
</html>

[root@ip-172-31-38-34 html]#
```

Browser view

← → ↻ 🏠 ⚠ Not secure | 35.154.172.164/task.html

# PAge of AWS via CLI

## by Abhishek

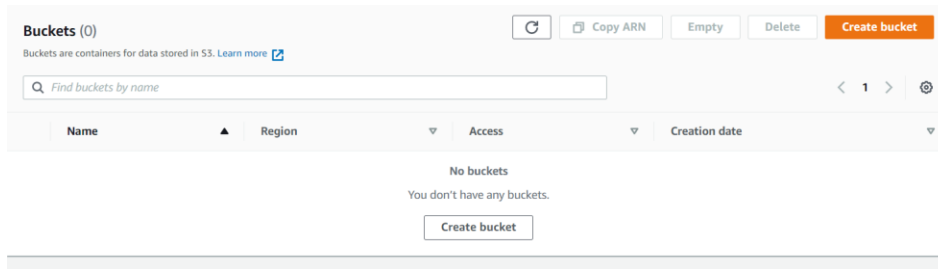


```
cp: cannot stat '0pic.jpg': No such file or directory
[root@ip-172-31-38-34 html]# cd /home/ec2-user
[root@ip-172-31-38-34 ec2-user]# ls
0pic.jpg
[root@ip-172-31-38-34 ec2-user]# cp 0pic.jpg /var/www/html
[root@ip-172-31-38-34 ec2-user]# cd /var/www/html
[root@ip-172-31-38-34 html]# ls
0pic.jpg  lost+found  task.html
[root@ip-172-31-38-34 html]# vi task.html
[root@ip-172-31-38-34 html]# cat task..html
cat: task..html: No such file or directory
[root@ip-172-31-38-34 html]# cat task.html
<html>
<body>
<h1>PAge of AWS via CLI</h1>
<h2>by Abhishek </h2>
<h3>here is my picture</h3>
<img src=0pic.jpg height=500 width 300>
</body>
</html>
```

```
[root@ip-172-31-38-34 html]# vi task.html
[root@ip-172-31-38-34 html]# cat task.html
<html>
<body>
<h1>PAge of AWS via CLI</h1>
<h2>by Abhishek </h2>
<h3>here is my picture</h3>
<img src=0pic.jpg height=500 width 300>
</body>
</html>

[root@ip-172-31-38-34 html]#
```

# CREATE BUCKET



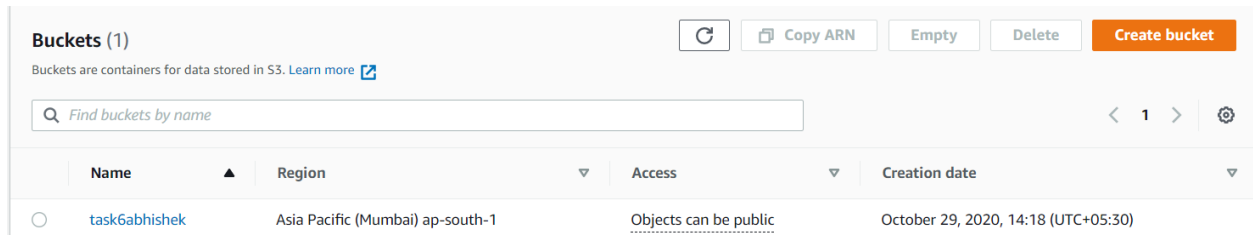
## CREATING BUCKET

```
C:\Users\Abhishek kumar\Desktop>aws s3api create-bucket --bucket task6abhishek --region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1
```

```
C:\Users\Abhishek kumar\Desktop>aws s3api create-bucket --bucket task6abhishek --region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1
{
  "Location": "http://task6abhishek.s3.amazonaws.com/"
}

C:\Users\Abhishek kumar\Desktop>
```

## SEE



## MAKING BUCKET PUBLICALLY ACCESSIBLE

```
C:\Users\Abhishek kumar\Desktop>aws s3api put-bucket-acl --acl public-read --bucket task6abhishek
```

```
C:\Users\Abhishek kumar\Desktop>aws s3api put-bucket-acl --acl public-read --bucket task6abhishek
```



## UPLAODING FILES IN BUCKET

```
aws s3api put-object --bucket task6abhishek --key 0pic.jpg --body "C:\Users\Abhishek kumar\Desktop\0pic.jpg"
```

```
C:\Users\Abhishek kumar\Desktop>aws s3api put-object --bucket task6abhishek --key 0pic.jpg --body "C:\Users\Abhishek kumar\Desktop\0pic.jpg"
{
  "ETag": "\"15d11d9d50b9f5f5b9abce09008927b0\""
}
```

## task6abhishek

### Bucket overview

|  |  |  |   |
|--|--|--|---|
| Region<br>Asia Pacific (Mumbai) ap-south-1 | Amazon resource name (ARN)<br> arn:aws:s3:::task6abhishek | Creation date<br>October 29, 2020, 14:18 (UTC+05:30) | Access<br> <b>Public</b> |
|--|--|--|---|


**Objects** | Properties | Permissions | Metrics | Management | Access points

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


### Objects (1)

Objects are the fundamental entities stored in Amazon S3. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

  **Actions** ▼  **Upload**

 Find objects by prefix

< 1 >

| <input type="checkbox"/> | Name ▲   | Type ▼ | Last modified ▼                     | Size ▼   | Storage class |
|--------------------------|--|--------|-------------------------------------|----------|---------------|
| <input type="checkbox"/> |  <a href="#">Opic.jpg</a> | jpg    | October 29, 2020, 17:26 (UTC+05:30) | 552.2 KB | Standard      |

## WE HAVE TO MAKE THE OBJECT PUBLIC

```
C:\Users\Abhishek kumar\Desktop>aws s3api put-object-acl --bucket task6abhishek --key Opic.jpg --grant-read uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Here I am making bucket object public but here access policy is only download and read instead of read on webserver. Here this policy download object first and after download we can read it. You can change this policy by changing -acl rule like private,public-read,public-read-write etc.

```
<html>
<body>
<h1>Page of AWS via CLI</h1>
<h2>by Abhishek </h2>
<h3>here is my picture</h3>
<img src=https://task6abhishek.s3.ap-south-1.amazonaws.com/opic.jpg height=500 width 300>
</body>
</html>
```

# CREATING CLOUDFRONT

```
C:\Users\Abhishek kumar\Desktop>aws cloudfront create-distribution --origin-domain-name task6abhishek.s3.amazonaws.com --default-root-object 0pic.jpg
```

```
C:\Users\Abhishek kumar\Desktop>aws cloudfront create-distribution --origin-domain-name task6abhishek.s3.amazonaws.com --default-root-object 0pic.jpg
{
  "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/E34I40GBA0AIPN",
  "ETag": "E2YU44PG1J7KIJ",
  "Distribution": {
    "Id": "E34I40GBA0AIPN",
    "ARN": "arn:aws:cloudfront::946697907226:distribution/E34I40GBA0AIPN",
    "Status": "InProgress",
    "LastModifiedTime": "2020-10-29T12:33:13.584000+00:00",
    "InProgressInvalidationBatches": 0,
    "DomainName": "d3ctnh42i18csc.cloudfront.net",
    "ActiveTrustedSigners": {
      "Enabled": false,
      "Quantity": 0
    },
    "DistributionConfig": {
      "CallerReference": "cli-1603974791-144290",
      "Aliases": {
        "Quantity": 0
      },
      "DefaultRootObject": "0pic.jpg",
      "Origins": {
        "Quantity": 1,
        "Items": [
          {
            "Id": "task6abhishek.s3.amazonaws.com-1603974791-505365",
            "DomainName": "task6abhishek.s3.amazonaws.com",
            "OriginPath": "",
            "CustomHeaders": {
              "Quantity": 0
            },
            "S3OriginConfig": {
              "OriginAccessIdentity": ""
            }
          }
        ]
      }
    }
  }
}
```

Now you can see here.

CloudFront Is successfully Deployed

## CloudFront Distributions

| Create Distribution   Distribution Settings   Delete   Enable   Disable |                 |                |                               |         |                                |                    |
|---|-----------------|----------------|-------------------------------|---------|--------------------------------|--------------------|
| Viewing: Any Delivery Method   Any State <input type="text"/>           |                 |                |                               |         |                                |                    |
|   | Delivery Method | ID             | Domain Name                   | Comment | Origin                         | Price Class        |
| <input type="checkbox"/>  | Web             | E34I40GBA0AIPN | d3ctnh42i18csc.cloudfront.net | -       | task6abhishek.s3.amazonaws.com | Use All Edge Locat |

| Viewing: Any Delivery Method   Any State <input type="text"/> |                 |                |                               |         |                                |  |
|---|-----------------|----------------|-------------------------------|---------|--------------------------------|--|
|   | Delivery Method | ID             | Domain Name                   | Comment | Origin                         |  |
| <input checked="" type="checkbox"/>                           | Web             | E34I40GBA0AIPN | d3ctnh42i18csc.cloudfront.net | -       | task6abhishek.s3.amazonaws.com |  |

**D3CTNH42I18CSC.CLOUDFRONT.NET** << link

the image source file

Now changed to that link

```
[root@ip-172-31-38-34 html]# cat task.html
<html>
<body>
<h1>PAge of AWS via CLI</h1>
<h2>by Abhishek </h2>
<h3>here is my picture</h3>
<img src=http://d3ctnh42i18csc.cloudfront.net height=500 width 300>
</body>
</html>
```