10/29/2020

TASK 6:

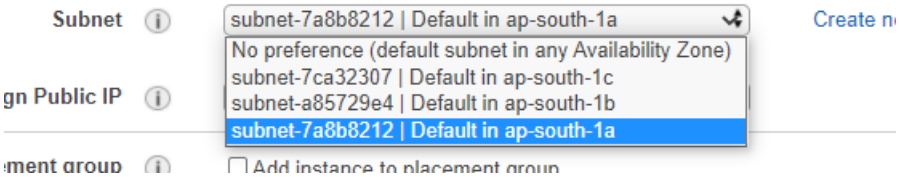
The architecture includes-✅ Webserver configured on EC2 Instance ✅Document Root(/var/www/html) made persistent by mounting on EBS Block Device. ✅Static objects used in code such as pictures stored in S3✅ Setting up Content Delivery Network using CloudFront and using the origin domain as S3 bucket. ✅ Finally place the Cloud Front URL on the webapp code for security and low latency.

Akshay Anil

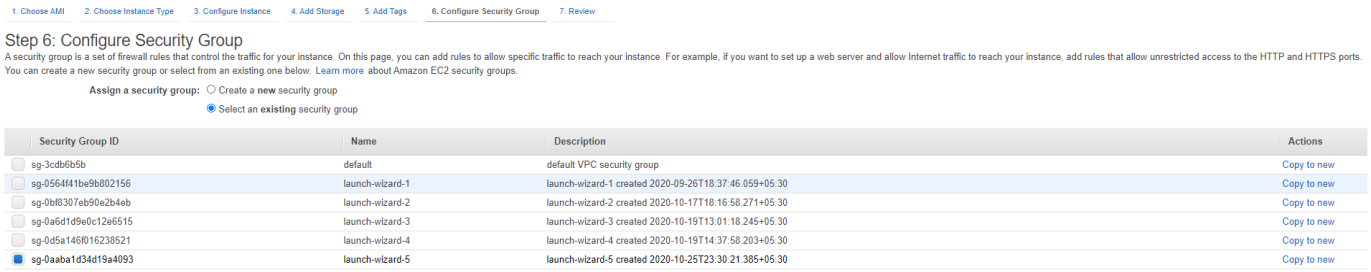
ARTH TASK 6:

SETTING UP AN INSTANCE:

Provide the info of all seven steps as u do in GUI.



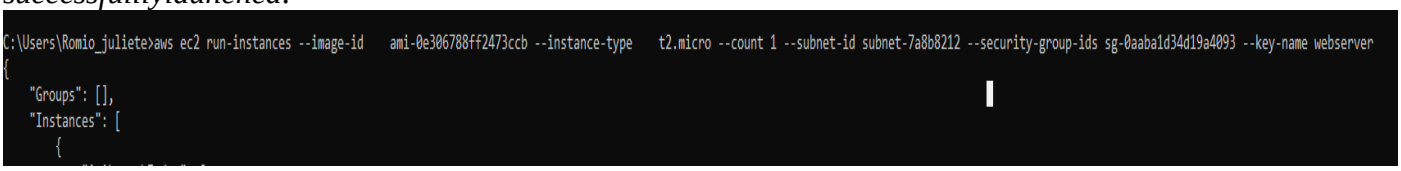
Choose the security grp:



1. *Creating an instance with CLI*

aws ec2 run-instances --image-id ami-0e306788ff2473ccb --instance-type t2.micro --count 1 --subnet-id subnet-7a8b8212 --security-group-ids sg-0aaba1d34d19a4093 --key-name webserver

successfulllylaunched:



ARTH TASK 6:

```
C:\Users\Romio_juliete>aws ec2 run-instances --image-id    ami-0e306788ff2473ccb --instance-type    t2.micro --count 1 --subnet-id subnet-7a8b8212 --security-group-ids sg-0aaba1d34d19a4093 --key-name webserver
{
    "Groups": [],
    "Instances": [
        {
            "AmiLaunchIndex": 0,
            "ImageId": "ami-0e306788ff2473ccb",
            "InstanceId": "i-0a06ad006e50cf731",
            "InstanceType": "t2.micro",
            "KeyName": "webserver",
            "LaunchTime": "2020-10-26T17:15:51+00:00",
            "Monitoring": {
                "State": "disabled"
            },
            "Placement": {
                "AvailabilityZone": "ap-south-1a",
                "GroupName": "",
                "Tenancy": "default"
            },
            "PrivateDnsName": "ip-172-31-42-202.ap-south-1.compute.internal",
            "PrivateIpAddress": "172.31.42.202",
            "ProductCodes": [],
            "PublicDnsName": "",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "StateTransitionReason": "",
            "SubnetId": "subnet-7a8b8212",
            "VpcId": "vpc-77bc5f1c",
            "Architecture": "x86_64",
            "BlockDeviceMappings": [],
            "ClientToken": "e9bb6640-2cec-4d8f-aaaf-c187628b457c",
            "EbsOptimized": false,
            "EnaSupport": true,
            "Hypervisor": "xen",
            "NetworkInterfaces": [
                {
                    "Attachment": {
                        "AttachTime": "2020-10-26T17:15:51+00:00",
                        "AttachmentId": "eni-attach-03c94881c33df1126",
                        "DeleteOnTermination": true,
                        "DeviceIndex": 0,
                        "Status": "attaching"
                    },
                    "Description": "",
                    "Groups": [
                        {
                            "GroupName": "launch-wizard-5",
                            "GroupId": "sg-0aaba1d34d19a4093"
                        }
                    ],
                    "Ipv6Addresses": [],
                    "MacAddress": "02:14:8e:b3:bf:1e",
                    "NetworkInterfaceId": "eni-0b2618dc28c5b8872",
```

Checking on GUI:

Instance ID	Instance state	In...	S..	A..	Ava	Pub	Public IPv4 ...	Elastic Ip	IPv6 IPs	Monitoring	Security Group name	Key na...	Launch time
i-0e6a00e7bced11548	Stopped	t2.mi...	–	+	a...	–	–	–	–	disabled	launch-wizard-4	namenode	2020/10/25 00:01 GMT+5:30
i-0a06ad006e50cf731	Running	t2.mi...		+	a...	e...	13.232.78.32	–	–	disabled	launch-wizard-5	webserver	2020/10/26 22:45 GMT+5:30
i-0d6cd8bf20c41c8d8	Stopped	t2.mi...	–	+	a...	–	–	–	–	disabled	launch-wizard-5	webserver	2020/10/25 23:31 GMT+5:30

1. *Starting instance with CLI*

Name	Instance ID	Instance state	Inst...	Status check	Alarm Status	Availabili...	Public IPv4 DNS	Public IPv4 ...	Elastic Ip	IPv6 IPs
–	i-0e6a00e7bced11548	Stopped	t2.micro	–	No alarms +	ap-south-1a	–	–	–	–
CLI	i-0a06ad006e50cf731	Stopped	t2.micro	–	No alarms +	ap-south-1a	–	–	–	–
webserver	i-0d6cd8bf20c41c8d8	Terminated	t2.micro	–	No alarms +	ap-south-1b	–	–	–	–

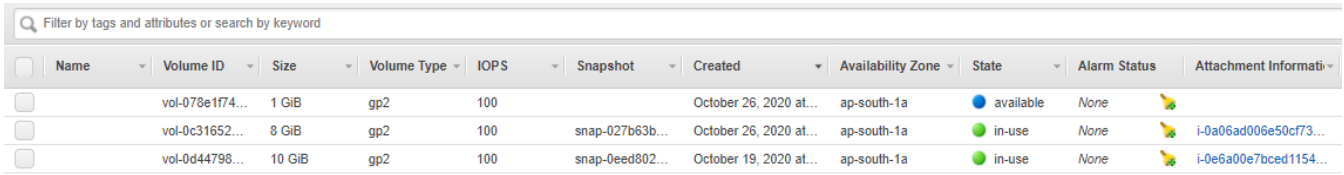
```
C:\Users\Romio_juliete>aws ec2 start-instances --instance-ids i-0a06ad006e50cf731
{
    "StartingInstances": [
        {
            "CurrentState": {
                "Code": 0,
                "Name": "pending"
            },
            "InstanceId": "i-0a06ad006e50cf731",
            "PreviousState": {
                "Code": 80,
                "Name": "stopped"
            }
        }
    ]
}
```

Name	Instance ID	Instance state	Inst...	Status check	Alarm Status	Availabili...	Public IPv4 DNS	Public IPv4 ...	Elastic Ip	IPv6 IPs	Monitoring
–	i-0e6a00e7bced11548	Stopped	t2.micro	–	No alarms +	ap-south-1a	–	–	–	–	disabled
CLI	i-0a06ad006e50cf731	Running	t2.micro	Initializing	No alarms +	ap-south-1a	ec2-13-235-24-198....	13.235.24.198	–	–	disabled
webserver	i-0d6cd8bf20c41c8d8	Terminated	t2.micro	–	No alarms +	ap-south-1b	–	–	–	–	disabled

ARTH TASK 6:

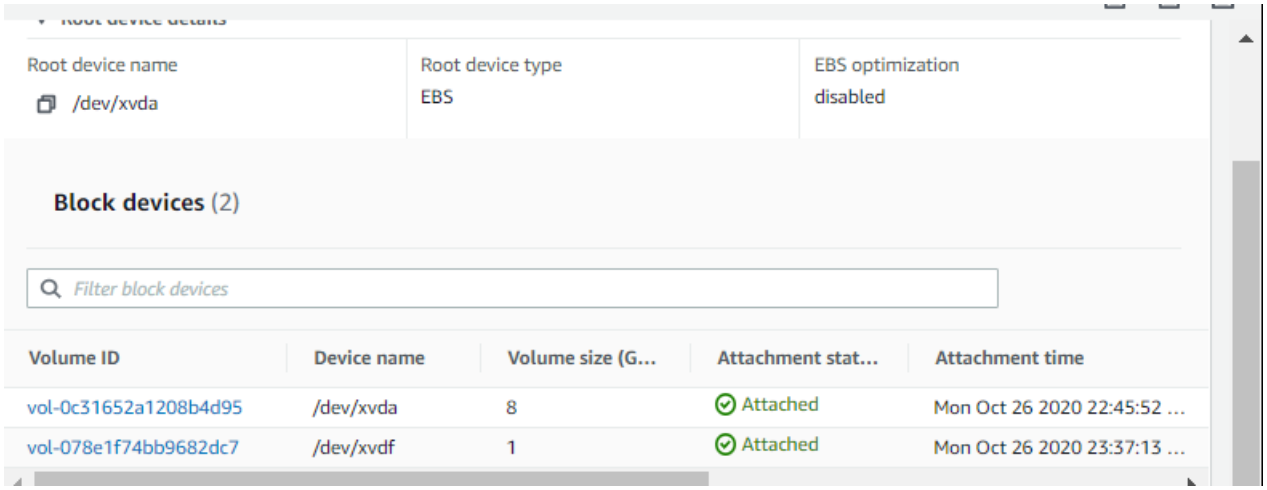
2. *Creating an EBS volume:*
   C:\Users\Romio_juliete> aws ec2 create-volume --volume-type gp2 --size 1 --availability-zone ap-south-1a

```
C:\Users\Romio_juliete>aws ec2 create-volume --volume-type gp2 --size 1 --availability-zone ap-south-1a
{
    "AvailabilityZone": "ap-south-1a",
    "CreateTime": "2020-10-26T18:03:14+00:00",
    "Encrypted": false,
    "Size": 1,
    "SnapshotId": "",
    "State": "creating",
    "VolumeId": "vol-078e1f74bb9682dc7",
    "Iops": 100,
    "Tags": [],
    "VolumeType": "gp2"
}
```



3. *Attach EBS to EC2 instance*
   C:\Users\Romio_juliete> aws ec2 attach-volume --instance-id i-0a06ad006e50cf731 --volume-id vol-078e1f74bb9682dc7 --device /dev/xvdf

```
C:\Users\Romio_juliete>aws ec2 attach-volume --instance-id i-0a06ad006e50cf731 --volume-id vol-078e1f74bb9682dc7 --device /dev/xvdf
{
    "AttachTime": "2020-10-26T18:07:13.441000+00:00",
    "Device": "/dev/xvdf",
    "InstanceId": "i-0a06ad006e50cf731",
    "State": "attaching",
    "VolumeId": "vol-078e1f74bb9682dc7"
}
```



ARTH TASK 6:

```
[root@ip-172-31-42-202 ~]# 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-172-31-42-202 ~]# 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-172-31-42-202 ~]#
```

4. *Creating a partition:*
   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo fdisk /dev/xvdf

```
C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo fdisk /dev/xvdf
The authenticity of host '13.235.24.198 (13.235.24.198)' can't be established.
ECDSA key fingerprint is SHA256:Wma37eZg9cqPoUR74E1kYBkbn4CAHvPSPRrETI6hQao.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '13.235.24.198' (ECDSA) to the list of known hosts.

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 0x9449b0f5.

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.
```

5. *Formatting the partition created.*
   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo mkfs.ext4 /dev/xvdf

```
C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.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
```

ARTH TASK 6:

Before mounting, install httpd else the directory /var/www/html will not exist

6. *Installing Apache Web server*
   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo yum install httpd -y

```
C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.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

Dependencies Resolved

================================================================================
 Package                 Arch       Version                 Repository      Size
================================================================================
Installing:
 httpd                   x86_64     2.4.46-1.amzn2          amzn2-core     1.3 M
Installing for dependencies:
 apr                     x86_64     1.6.3-5.amzn2.0.2       amzn2-core     118 k
 apr-util                x86_64     1.6.1-5.amzn2.0.2       amzn2-core      99 k
 apr-util-bdb            x86_64     1.6.1-5.amzn2.0.2       amzn2-core      19 k
 generic-logos-httpd     noarch     18.0.0-4.amzn2          amzn2-core      19 k
 httpd-filesystem        noarch     2.4.46-1.amzn2          amzn2-core      23 k
 httpd-tools             x86_64     2.4.46-1.amzn2          amzn2-core      87 k
 mailcap                 noarch     2.1.41-2.amzn2          amzn2-core      31 k
 mod_http2               x86_64     1.15.14-2.amzn2         amzn2-core     147 k

Transaction Summary
================================================================================
Install  1 Package (+8 Dependent packages)

Total download size: 1.8 M
Installed size: 5.1 M
```

7. *Mounting the Partition.*
   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo mount /dev/xvdf1 /var/www/html

8. *Start the apache web server and check the status*

   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo systemctl start httpd

   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo systemctl status httpd

ARTH TASK 6:

```
C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo mount /dev/xvdf1 /var/www/html

C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem sudo systemctl start httpd

C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.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 07:55:56 UTC; 6s ago
     Docs: man:httpd.service(8)
 Main PID: 16491 (httpd)
   Status: "Processing requests..."
   CGroup: /system.slice/httpd.service
           ├─16491 /usr/sbin/httpd -DFOREGROUND
           ├─16492 /usr/sbin/httpd -DFOREGROUND
           ├─16493 /usr/sbin/httpd -DFOREGROUND
           ├─16494 /usr/sbin/httpd -DFOREGROUND
           ├─16495 /usr/sbin/httpd -DFOREGROUND
           └─16496 /usr/sbin/httpd -DFOREGROUND

Oct 29 07:55:56 ip-172-31-42-202.ap-south-1.compute.internal systemd[1]: Starting The Apache HTTP Server...
Oct 29 07:55:56 ip-172-31-42-202.ap-south-1.compute.internal systemd[1]: Started The Apache HTTP Server.
```

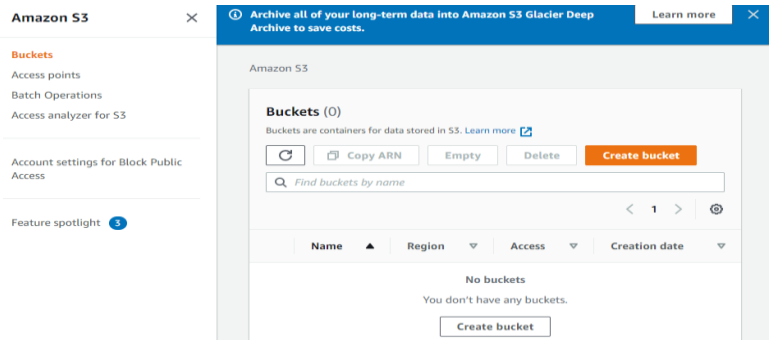
9. *Login in instance via CLI.*
   C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem

```
C:\Users\Romio_juliete\Desktop\pem>ssh -l ec2-user 13.235.24.198 -i webserver.pem
Last login: Thu Oct 29 07:46:15 2020 from ec2-13-233-177-1.ap-south-1.compute.amazonaws.com

Last login: Thu Oct 29 07:46:15 2020 from ec2-13-233-177-1.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.
```

10. *Creating the webpage: aks.html*

```
[ec2-user@ip-172-31-42-202 ~]$ sudo su - root
Last login: Thu Oct 29 07:46:26 UTC 2020 on pts/4
[root@ip-172-31-42-202 ~]# cd var/www/html
-bash: cd: var/www/html: No such file or directory
[root@ip-172-31-42-202 ~]# cd /var/www/html
[root@ip-172-31-42-202 html]# ls
lost+found
[root@ip-172-31-42-202 html]# vi aks.html
[root@ip-172-31-42-202 html]# cd /home/ec2-user
[root@ip-172-31-42-202 ec2-user]# ls
profile.jpg
[root@ip-172-31-42-202 ec2-user]# cp profile.jpg /var/www/html
[root@ip-172-31-42-202 ec2-user]# cd /var/www/html
[root@ip-172-31-42-202 html]# ls
aks.html  lost+found  profile.jpg
```

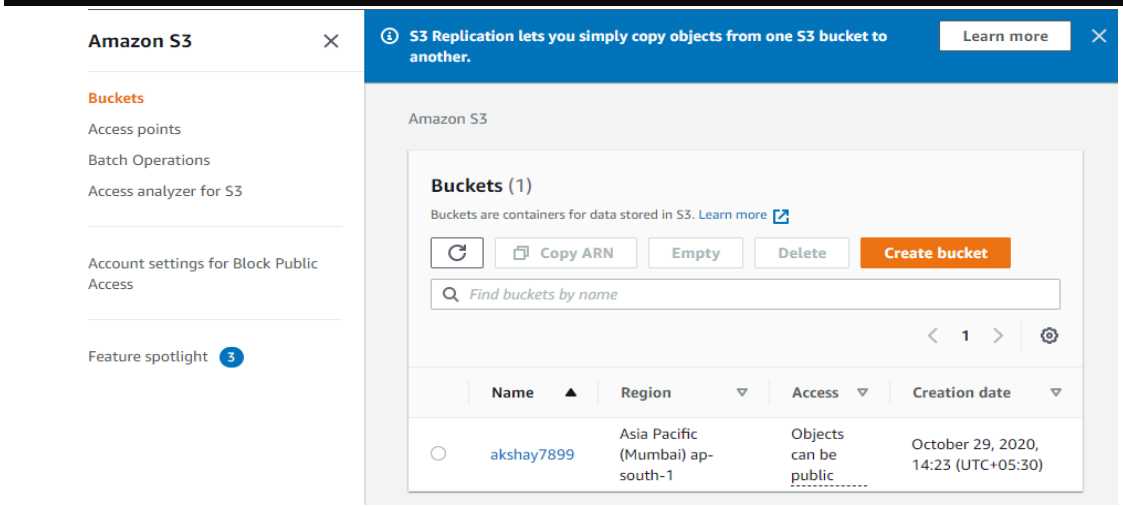
11. *Creating a bucket:*



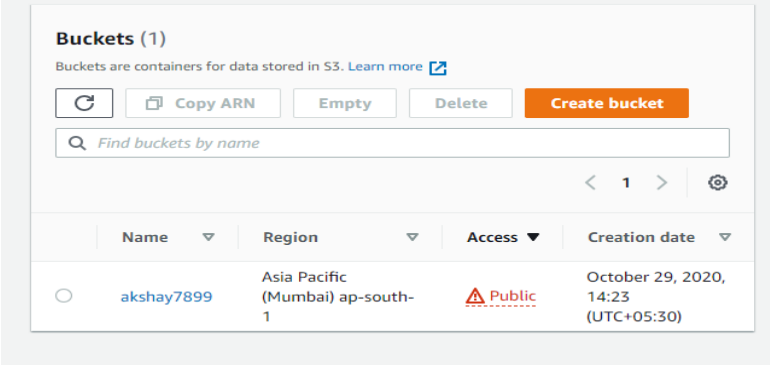
ARTH TASK 6:

C:\Users\Romio_juliete\Desktop\pem>aws s3api create-bucket --bucket akshay7899 --region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1

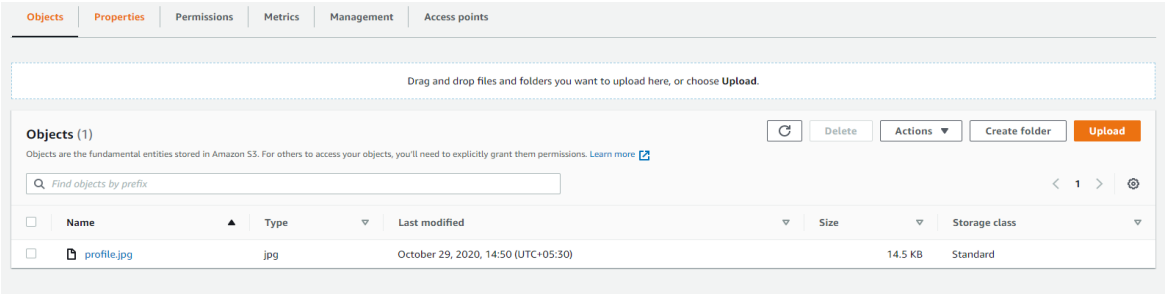
```
C:\Users\Romio_juliete\Desktop\pem>aws s3api create-bucket --bucket akshay7899 --region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1
{
    "Location": "http://akshay7899.s3.amazonaws.com/"
}
```



- ✓ *Making the bucket publicly accessible:*
  C:\Users\Romio_juliete\Desktop\pem>aws s3api put-bucket-acl --acl public-read --bucket akshay7899



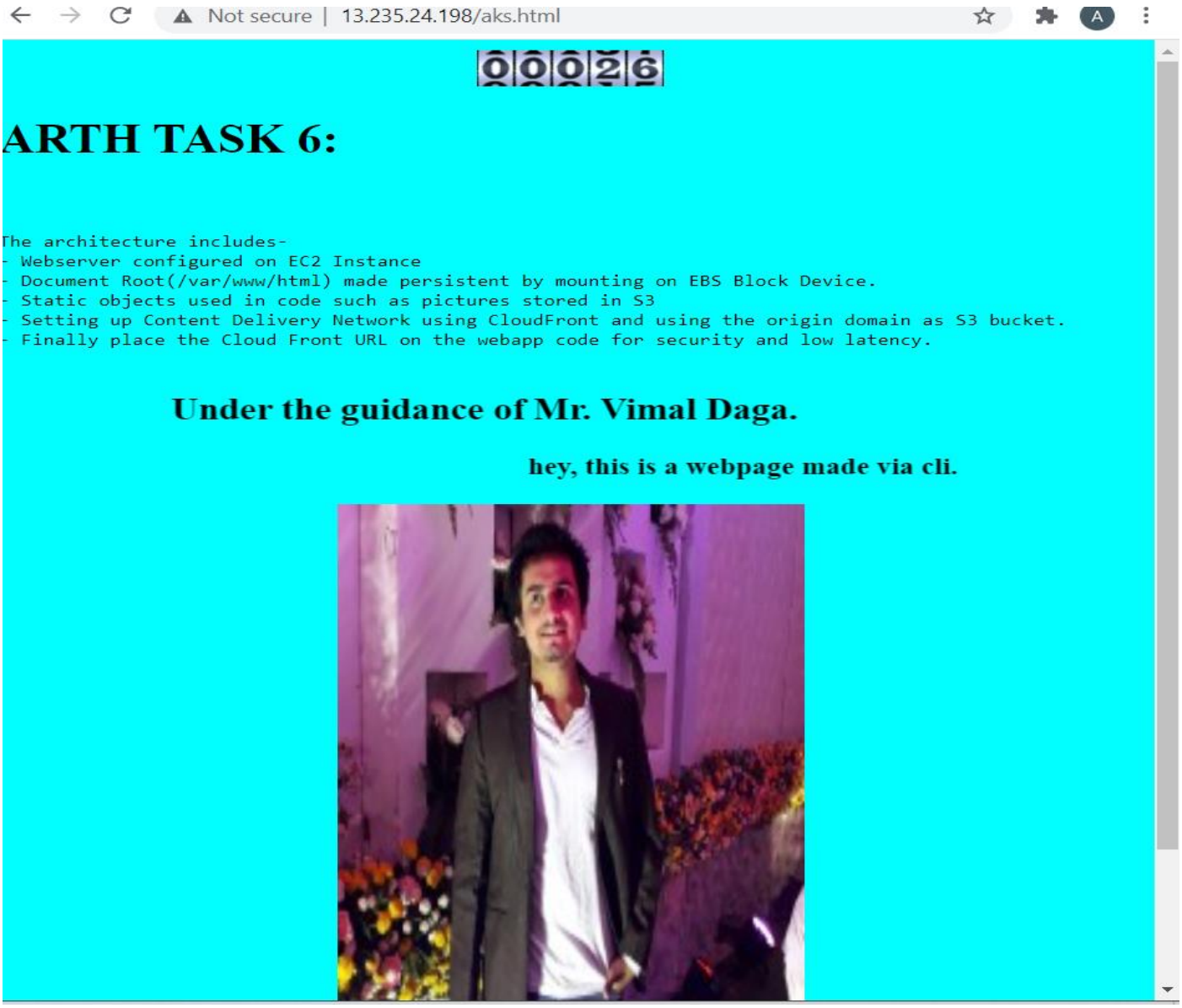
- ✓ *Putting the object in bucket:*
  C:\Users\Romio_juliete\Desktop\pem>aws s3api put-object --bucket akshay7899 --key profile.jpg --body C:\Users\Romio_juliete\Desktop\images\profile.jpg



- ✓ *Making the object file public.*

ARTH TASK 6:

Adding the link of object url in the html file, and the interface looks like:



ARTH TASK 6:

12. *Creating the CloudFront Distribution: linking object in S3 to every edge location to decrease the latency.*
    C:\Users\Romio_juliete\Desktop\pem>aws cloudfront create-distribution --origin-domain-name akshay7899.s3.amazonaws.com --default-root-object profile.jpg

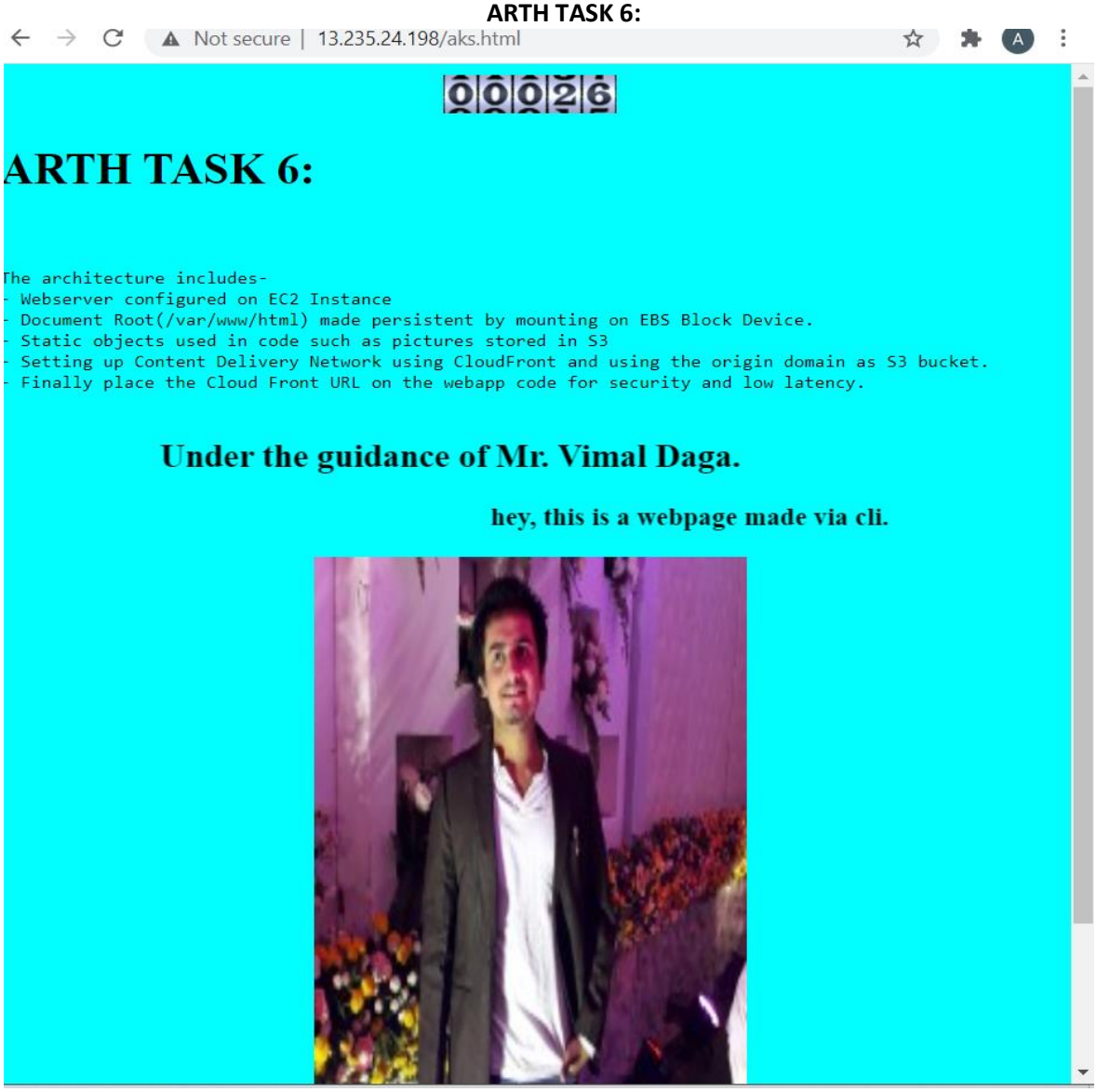
```
C:\Users\Romio_juliete>aws cloudfront create-distribution --origin-domain-name akshay7899.s3.amazonaws.com --default-root-object profile.jpg
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/E2Z4X7STOJJGIT",
    "ETag": "EL35R2KAYEBJR",
    "Distribution": {
        "Id": "E2Z4X7STOJJGIT",
        "ARN": "arn:aws:cloudfront::675678579729:distribution/E2Z4X7STOJJGIT",
        "Status": "InProgress",
        "LastModifiedTime": "2020-10-29T12:41:26.980000+00:00",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d3mc1se2fykfun.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-1603975283-242430",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "profile.jpg",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "akshay7899.s3.amazonaws.com-1603975283-196795",
                        "DomainName": "akshay7899.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
```

13. *Updating the link in html file with cloud-front domain name.*

```
</marquee>
<img title='yay !!' src='http://d3mc1se2fykfun.cloudfront.net' height='500' width='300' />
```

And it's working.

ARTH TASK 6:



CONCLUSION:
Successful completed Arth Task 6
Making an instance, Attaching EBS where code is written, Attaching S3 for static data.And finally, creating the cloud front distribution to create a local cache to all Edge locations to decrease the latency. Deploying the websrver
All done via awscli.

Thank you.