

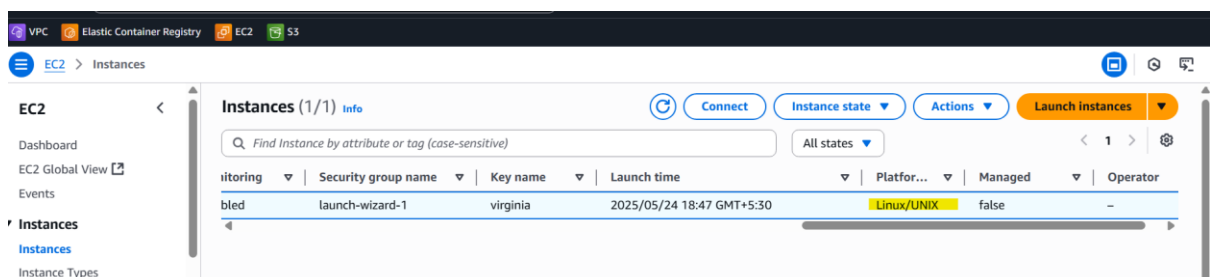
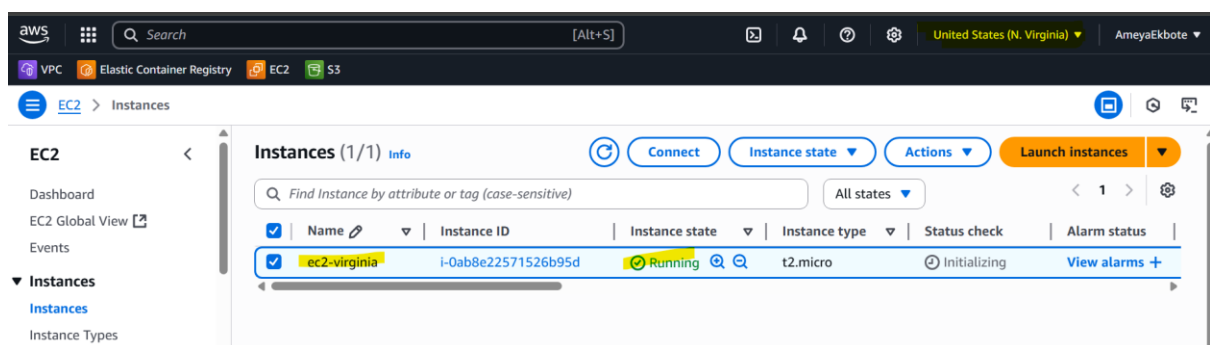
**1 ] Problem Statement: You work for XYZ Corporation. Your corporation is working on an application and they require secured web servers on Linux to launch the application.**

### Tasks To Be Performed:

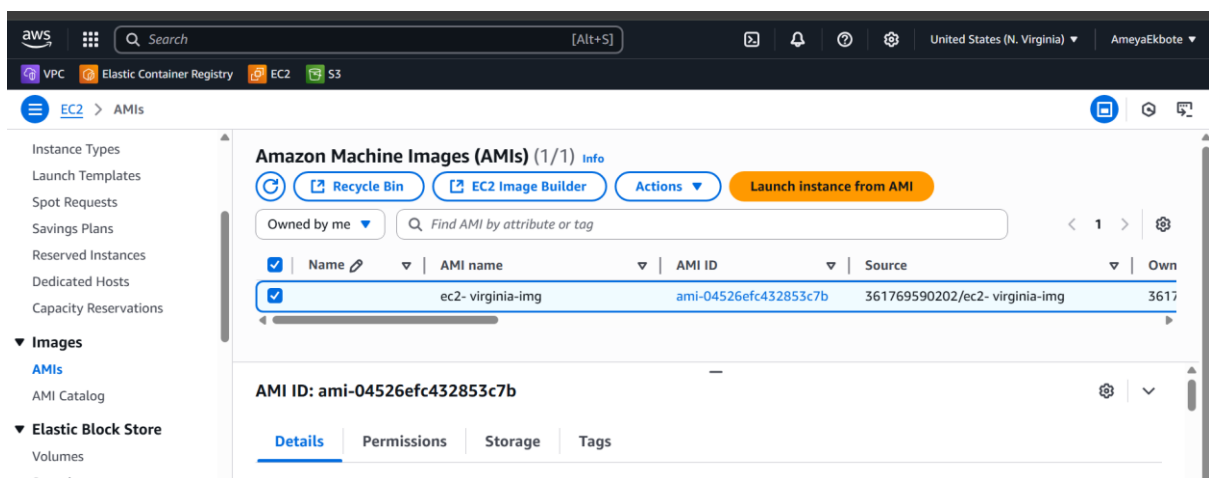
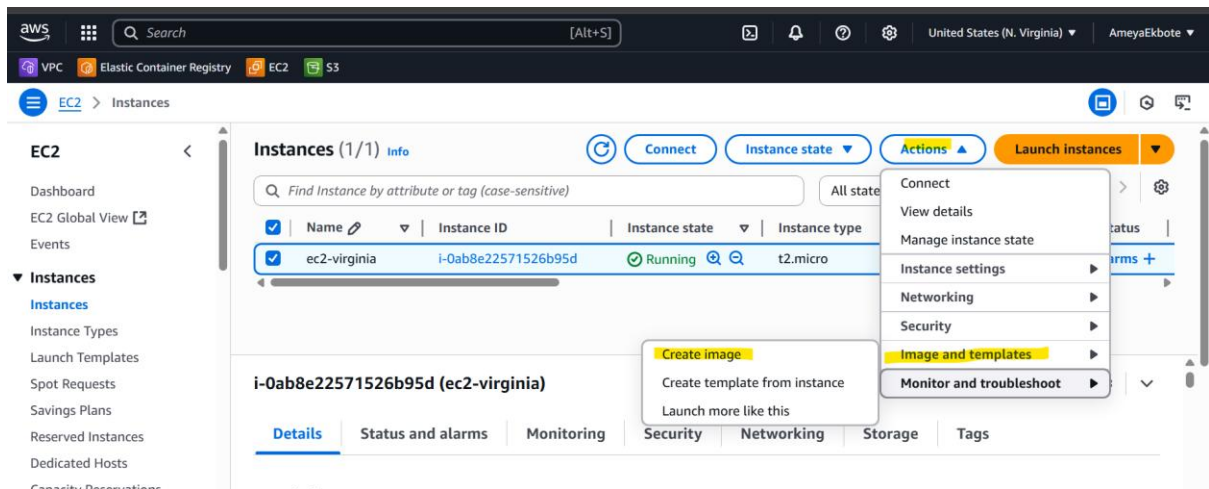
**1. Create an instance in the US-East-1 (N. Virginia) region with Linux OS and manage the requirement of web servers of your company using AMI. 2. Replicate the instance in the US-West-2 (Oregon) region. 3. Build two EBS volumes and attach them to the instance in the US-East-1 (N. Virginia) region. 4. Delete one volume after detaching it and extend the size of the other volume. 5. Take backup of this EBS volume**

## Solution →

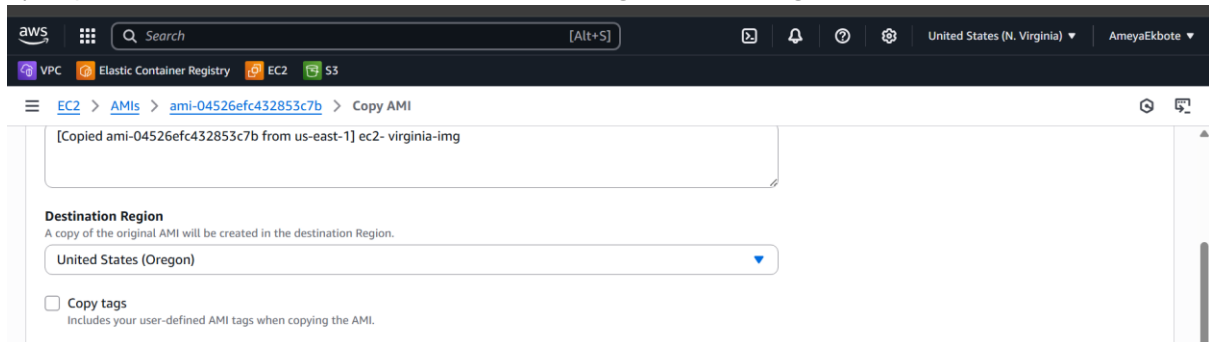
1) Created an instance in Virginia Region used Linux OS



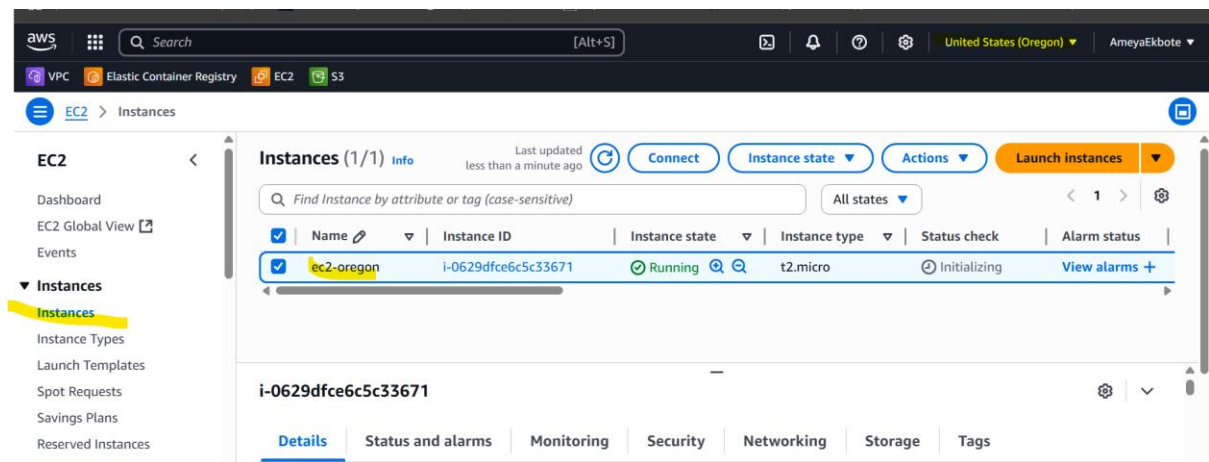
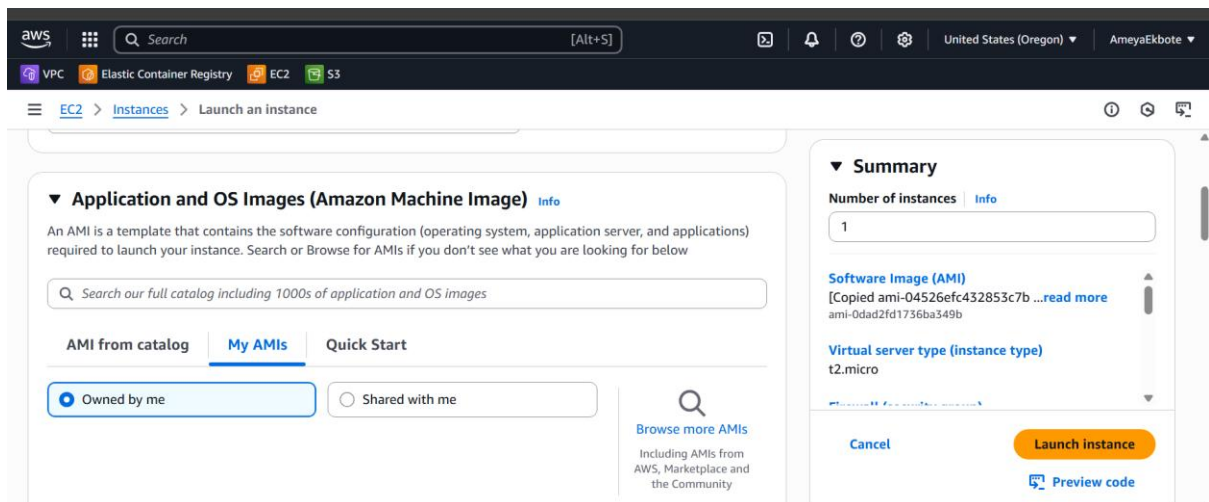
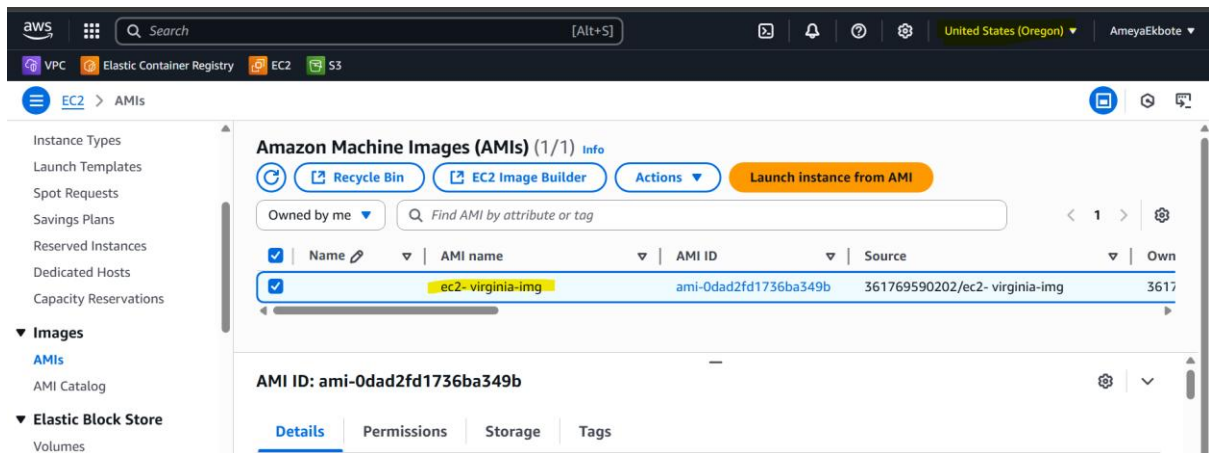
2) Created Image of Instance



3) Copied the AMI and choose the destination region to be Oregon



4. Switched the region to US West Oregon and launched instance using the AMI.



5) Created two volumes and attached it to the instance. I have created two volumes of 10GB and 12GB respectively.

Successfully created volume vol-07212e0320511f7e7.

**Volumes (3)** Info

Choose filter set  Search

<input type="checkbox"/>	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Created
<input type="checkbox"/>		vol-02ab1e4431dff483f	gp3	8 GiB	3000	125	snap-0066a3d...	2025/05/24 18:47
<input type="checkbox"/>		vol-07212e0320511f7e7	gp3	12 GiB	3000	125	-	2025/05/24 19:26
<input type="checkbox"/>		vol-0a948ba80e8856f7a	gp3	10 GiB	3000	125	-	2025/05/24 19:26

Fault tolerance for all volumes in this Region

6. Attached both the volumes to the instance in Virginia region.

**EC2** > Instances

**Instances (1/1)** Info

Find Instance by attribute or tag (case-sensitive)  All states

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	ec2-virginia	i-0ab8e22571526b95d	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

**i-0ab8e22571526b95d (ec2-virginia)**

<input checked="" type="checkbox"/>	Volume ID	Device name	Volume size (GiB)	Volume State	Attachment status	Attachment time
<input checked="" type="checkbox"/>	vol-00ccd0a3c8694d492	/dev/sdf	12	In-use	Attached	2025/05/24 19:30
<input checked="" type="checkbox"/>	vol-03845e488dba3e3e3	/dev/sdg	10	In-use	Attached	2025/05/24 19:30

**EC2** > Instances

**Instances (1/1)** Info

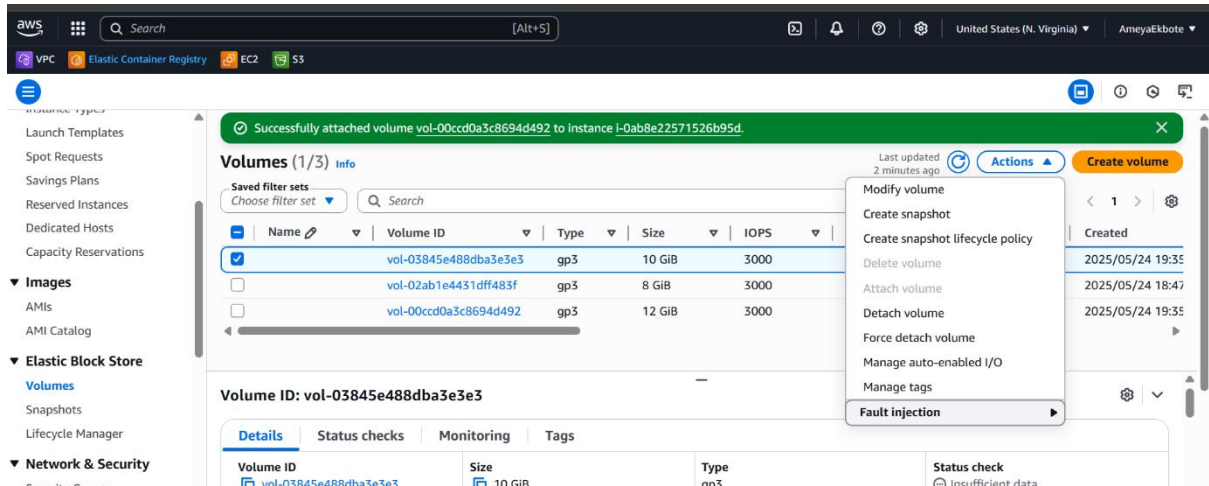
Find Instance by attribute or tag (case-sensitive)  All states

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	ec2-virginia	i-0ab8e22571526b95d	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

**i-0ab8e22571526b95d (ec2-virginia)**

<input checked="" type="checkbox"/>	Volume ID	Device name	Volume size (GiB)	Volume State	Attachment status	Attachment time
<input checked="" type="checkbox"/>	vol-02ab1e4431dff483f	/dev/xvda	8	In-use	Attached	2025/05/24 18:47
<input checked="" type="checkbox"/>	vol-00ccd0a3c8694d492	/dev/sdf	12	In-use	Attached	2025/05/24 19:30

7. After attaching both the volumes detach one of them and extended the size of other by modifying the volume.



Successfully attached volume vol-00ccd0a3c8694d492 to instance i-0ab8e22571526b95d.

**Volumes (1/3)** Info

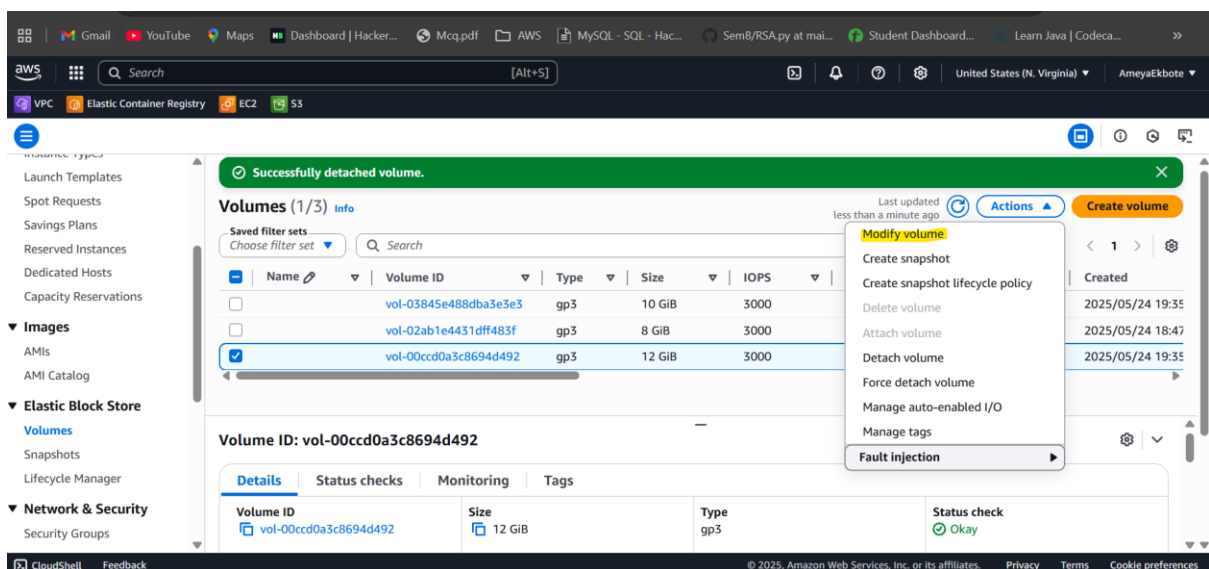
Saved filter sets: Choose filter set

Name	Volume ID	Type	Size	IOPS
<input checked="" type="checkbox"/>	vol-03845e488dba3e3e3	gp3	10 GiB	3000
<input type="checkbox"/>	vol-02ab1e4431dff483f	gp3	8 GiB	3000
<input type="checkbox"/>	vol-00ccd0a3c8694d492	gp3	12 GiB	3000

Volume ID: vol-03845e488dba3e3e3

Details | Status checks | Monitoring | Tags

Volume ID: vol-03845e488dba3e3e3 | Size: 10 GiB | Type: gp3 | Status check: Insufficient data



Successfully detached volume.

**Volumes (1/3)** Info

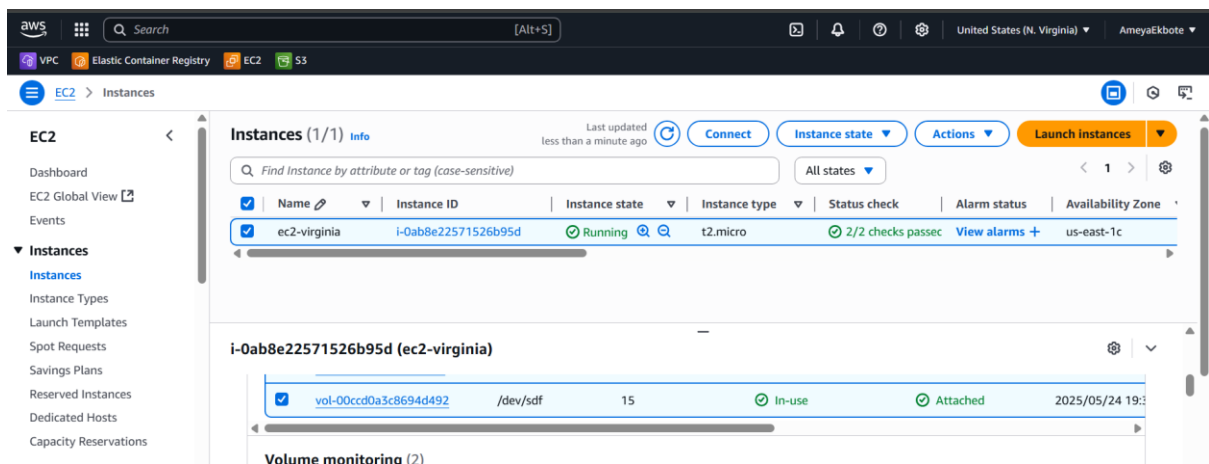
Saved filter sets: Choose filter set

Name	Volume ID	Type	Size	IOPS
<input type="checkbox"/>	vol-03845e488dba3e3e3	gp3	10 GiB	3000
<input type="checkbox"/>	vol-02ab1e4431dff483f	gp3	8 GiB	3000
<input checked="" type="checkbox"/>	vol-00ccd0a3c8694d492	gp3	12 GiB	3000

Volume ID: vol-00ccd0a3c8694d492

Details | Status checks | Monitoring | Tags

Volume ID: vol-00ccd0a3c8694d492 | Size: 12 GiB | Type: gp3 | Status check: Okay



**Instances (1/1)** Info

Find Instance by attribute or tag (case-sensitive)

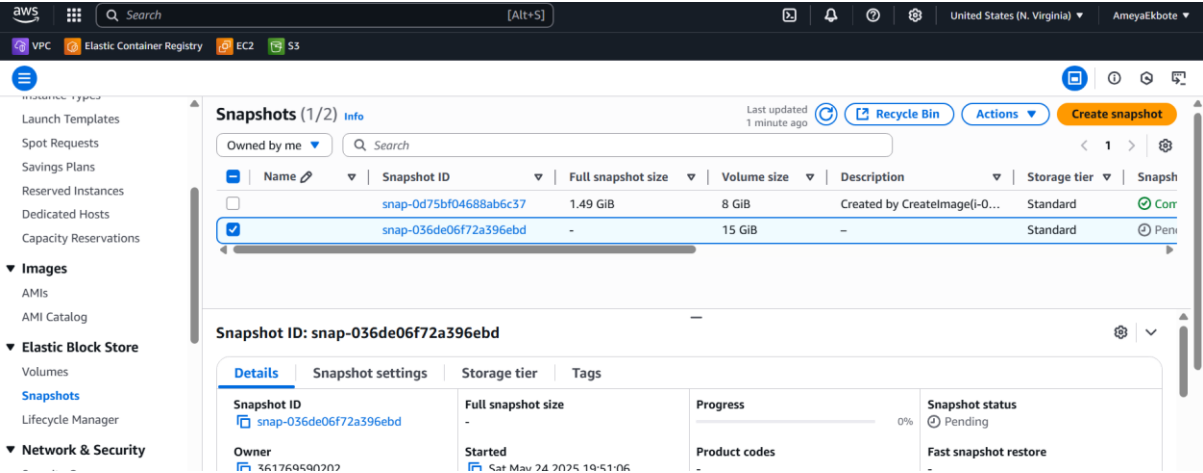
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	ec2-virginia	i-0ab8e22571526b95d	Running	t2.micro	2/2 checks passed	us-east-1c

**i-0ab8e22571526b95d (ec2-virginia)**

Volume ID	Mount point	Size	State	Attachment ID	Created
<input checked="" type="checkbox"/>	/dev/sdf	15	In-use	vol-00ccd0a3c8694d492	2025/05/24 19:35
<input type="checkbox"/>	/dev/sdb	8	Attached	vol-02ab1e4431dff483f	2025/05/24 18:47

Volume monitorina (2)

8. To take the backup of the Volume, we need to create the Snapshot of the Volume -



**2] Tasks To Be Performed: 1. Create an EFS and connect it to 3 different EC2 instances. Make sure that all instances have different operating systems. For instance, Ubuntu, Red Hat Linux and Amazon Linux 2.**

**Solution →**

Creating 3 instances

## 1) Linux Instance creation =>

The screenshot shows the AWS Management Console interface for creating a new EC2 instance. The top navigation bar includes the AWS logo, a search bar, and various service icons (VPC, Elastic Container Registry, EC2, S3). The breadcrumb trail indicates the path: EC2 > Instances > Launch an instance. The main heading is 'Launch an instance' with an 'Info' link. Below this, a brief description states: 'Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.'

The 'Launch an instance' page is divided into several sections:

- Name and tags:** A text input field contains 'ec2-linux'. To the right is a link 'Add additional tags'.
- Application and OS Images (Amazon Machine Image):** This section includes a search bar with the placeholder text 'Search our full catalog including 1000s of application and OS images'. Below the search bar are two tabs: 'Recents' and 'Quick Start'.
- Summary:** A sidebar on the right contains a 'Summary' section with the following details:
  - Number of instances:** A dropdown menu set to '1'.
  - Software Image (AMI):** 'Amazon Linux 2023 AMI 2023.7.2...read more' with the ID 'ami-0953476d60561c955'.
  - Virtual server type (instance type):** 't2.micro'.
  - Firewall (security group):** 'New security group'.

At the bottom right of the summary section are two buttons: 'Cancel' and 'Launch instance'.

EC2 > Instances > Launch an instance

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0953476d60561c955 (64-bit (x86), uefi-preferred) / ami-05a3e0187917e3e24 (64-bit (Arm), uefi)

Virtualization: hvm    ENA enabled: true    Root device type: ebs

Free tier eligible

Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

## 2) Ubuntu Instance creation =>

aws

Search

[Alt+S]

VPC

Elastic Container Registry

EC2

S3

United States (N. Virginia)

AmeyaEkbote

EC2 > Instances > Launch an instance

Name

ec2-ubuntu

Add additional tags

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...read more

ami-084568db4383264d4

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

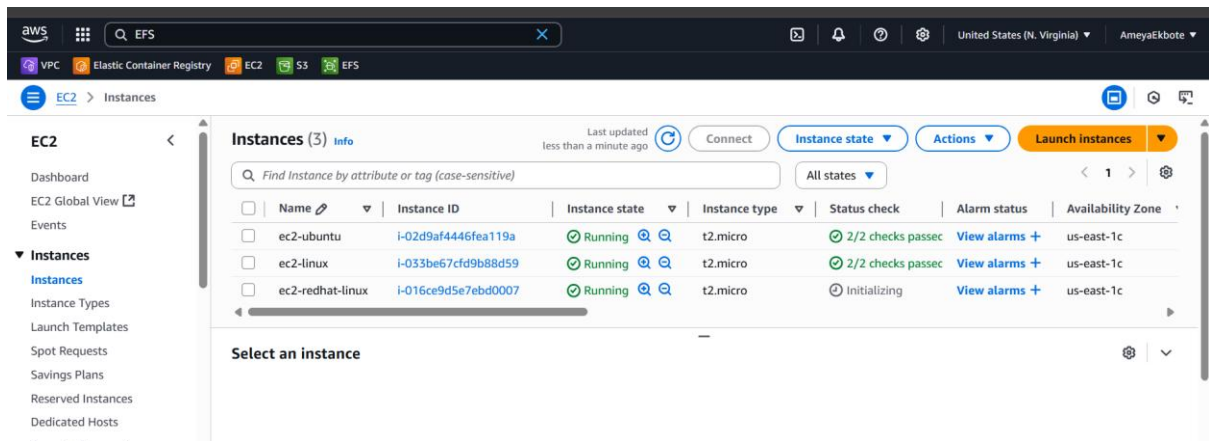
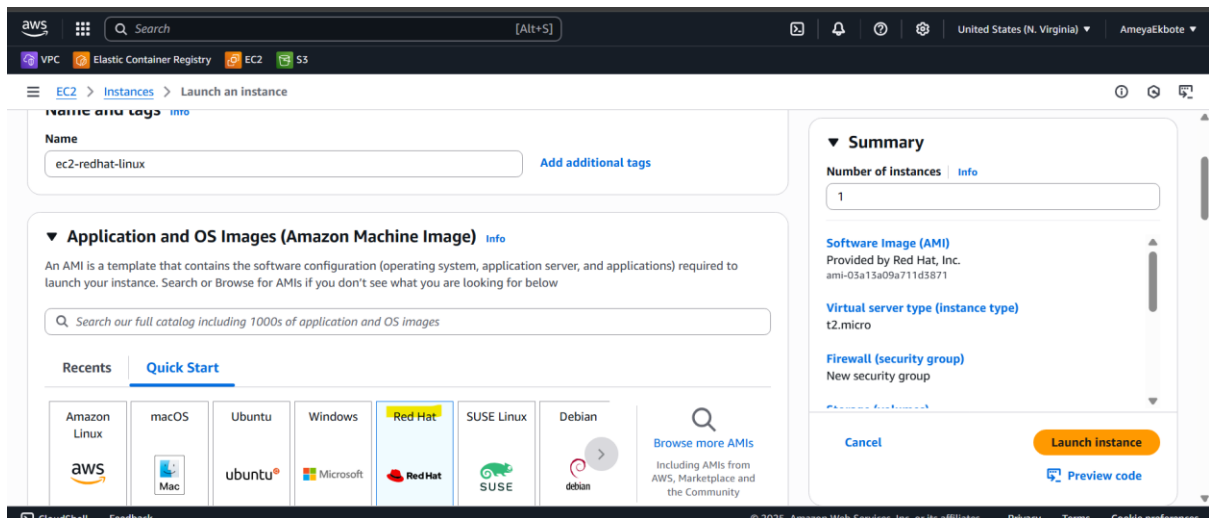
Cancel

Launch instance

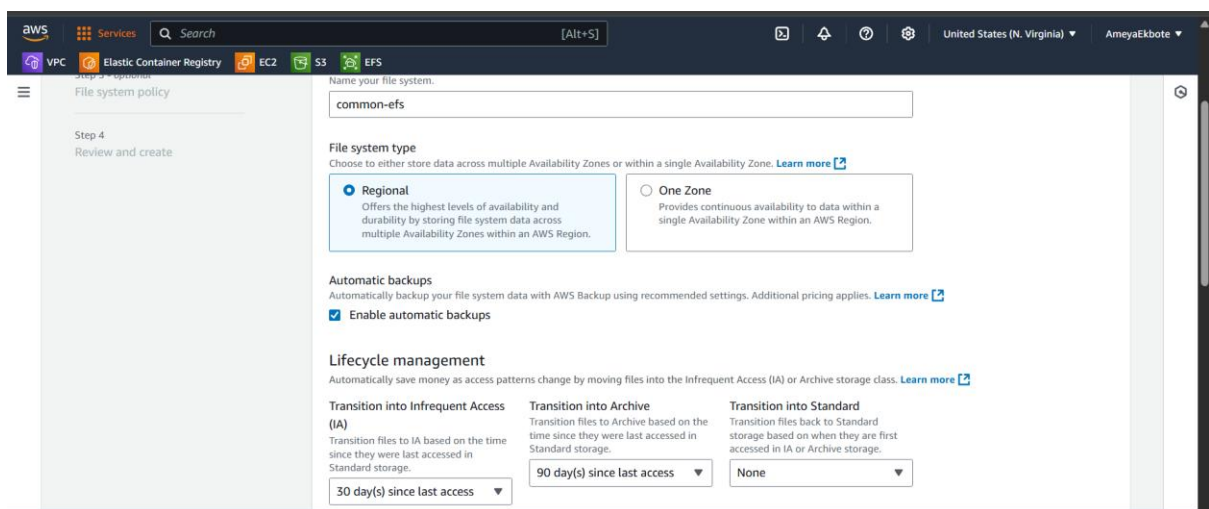
Preview code

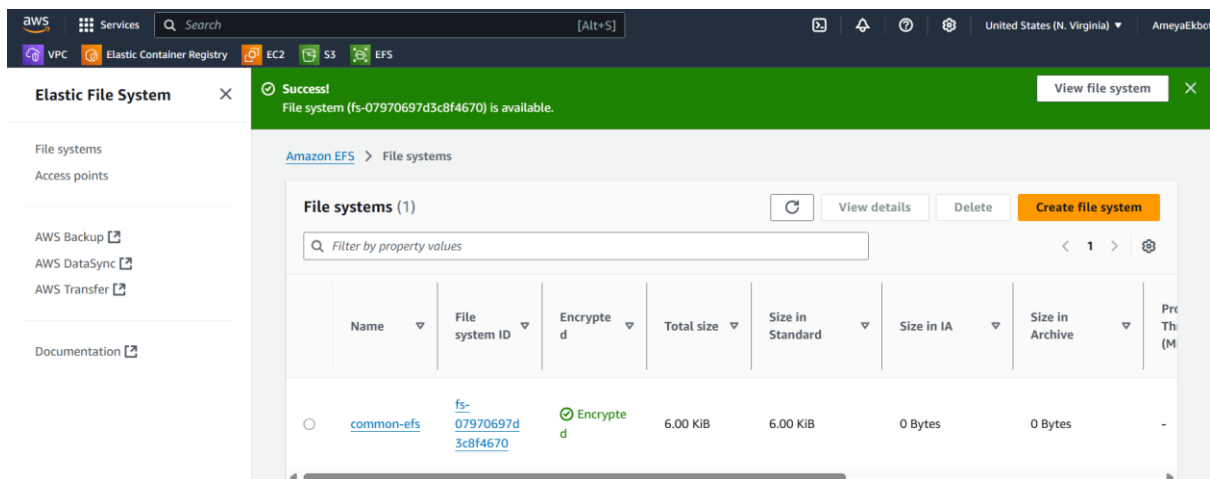
## 3) Red Hat Linux Instance creation =>



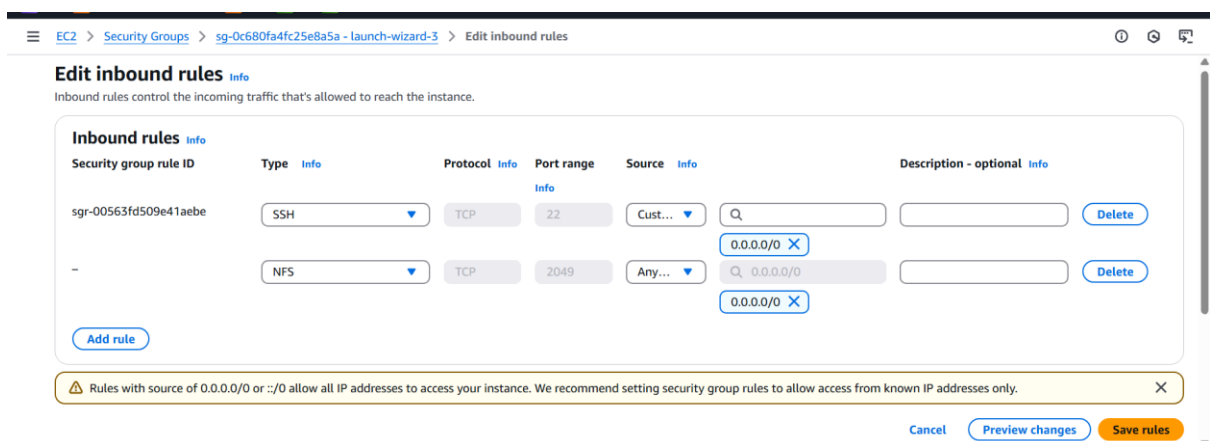
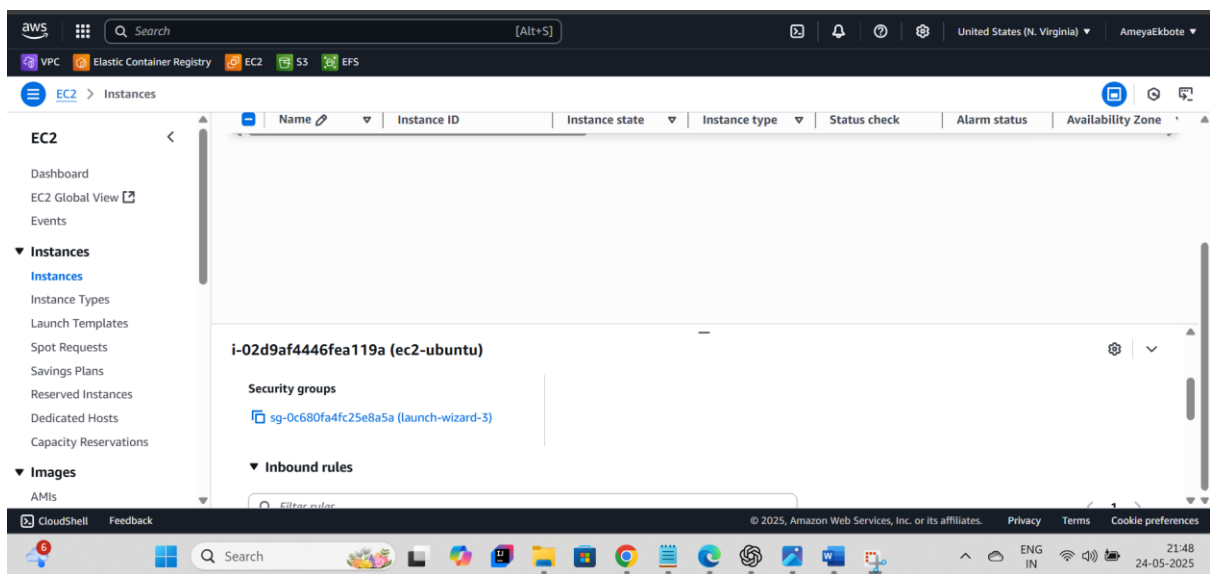


#### 4) Creating common EFS =>





5) Attaching EFS to ubuntu instance , updating security group and inbound connectivity opened to NFS services .



6) Attached EFS to ubuntu instance and created Ameya\_ubuntu.txt file , will see this file after attaching EFS to linux instance =>

```
aws
Search [Alt+S]
VPC Elastic Container Registry EC2 S3 EFS

ubuntu@ip-172-31-23-8:~$ df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/root                  6.8G    2.2G    4.6G   33% /
tmpfs                      479M         0   479M    0% /dev/shm
tmpfs                      192M    896K   191M    1% /run
tmpfs                      5.0M         0    5.0M    0% /run/lock
/dev/xvda16                881M     79M    741M   10% /boot
/dev/xvda15               105M     6.1M     95M    6% /boot/efi
fs-07970697d3c8f4670.efs.us-east-1.amazonaws.com:/ 8.0E         0    8.0E    0% /home/ubuntu/efs
tmpfs                      96M     12K    96M    1% /run/user/1000
ubuntu@ip-172-31-23-8:~$ cd /efs/
ubuntu@ip-172-31-23-8:~/efs$ ls
ubuntu@ip-172-31-23-8:~/efs$ sudo touch ameya_ubuntu.txt
ubuntu@ip-172-31-23-8:~/efs$ ls
ameya_ubuntu.txt
ubuntu@ip-172-31-23-8:~/efs$
```

7 ) Linux – Attached Common EFS to linux instance and able to see ubuntu.txt file , created one more txt file Ameya\_linux.txt which will be check in redhat linux instance.

```
aws
Search [Alt+S]
VPC Elastic Container Registry EC2 S3 EFS

[ec2-user@ip-172-31-27-34 efs]$ ls
ameya_ubuntu.txt
[ec2-user@ip-172-31-27-34 efs]$ df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                  4.0M         0   4.0M    0% /dev
tmpfs                      475M         0   475M    0% /dev/shm
tmpfs                      190M    440K   190M    1% /run
/dev/xvda1                 8.0G    1.6G    6.4G   20% /
tmpfs                      475M         0   475M    0% /tmp
/dev/xvda128              10M     1.3M    8.7M   13% /boot/efi
tmpfs                      95M         0    95M    0% /run/user/1000
fs-07970697d3c8f4670.efs.us-east-1.amazonaws.com:/ 8.0E         0    8.0E    0% /home/ec2-user/efs
[ec2-user@ip-172-31-27-34 efs]$
```

**i-033be67cfd9b88d59 (ec2-linux)**

Public IPs: 54.173.17.81 Private IPs: 172.31.27.34

```
aws
Search [Alt+S]
VPC Elastic Container Registry EC2 S3 EFS

[ec2-user@ip-172-31-27-34 efs]$ ls
ameya_ubuntu.txt
[ec2-user@ip-172-31-27-34 efs]$
```

**i-033be67cfd9b88d59 (ec2-linux)**

Public IPs: 54.173.17.81 Private IPs: 172.31.27.34

```
aws
Search [Alt+S]
VPC Elastic Container Registry EC2 S3 EFS
United States (N. Virginia) AmeyaEkbote

[ec2-user@ip-172-31-27-34 efs]$ df -h
Filesystem              Size  Used Avail Use% Mounted on
devtmpfs                4.0M   0  4.0M   0% /dev
tmpfs                   475M   0  475M   0% /dev/shm
tmpfs                   190M  440K  190M   1% /run
/dev/xvda1              8.0G  1.6G  6.4G  20% /
tmpfs                   475M   0  475M   0% /tmp
/dev/xvda128            10M  1.3M  8.7M  13% /boot/efi
tmpfs                   95M   0   95M   0% /run/user/1000
fs-07970697d3c8f4670.efs.us-east-1.amazonaws.com:/ 8.0E   0  8.0E   0% /home/ec2-user/efs
[ec2-user@ip-172-31-27-34 efs]$ ls
ameya_ubuntu.txt
[ec2-user@ip-172-31-27-34 efs]$ sudo nano ameya_linux.txt
[ec2-user@ip-172-31-27-34 efs]$ ls
ameya_linux.txt  ameya_ubuntu.txt
[ec2-user@ip-172-31-27-34 efs]$ cat ameya_linux.txt
"Hello Linux !!"
[ec2-user@ip-172-31-27-34 efs]$
```

8) Redhat – Connected redhat instance using SSH connection with key pair and attached common efs to this instance as well . Able to see both Ameya\_ubuntu.txt and Ameya\_linux.txt files .

aws Search [Alt+S] VPC Elastic Container Registry EC2 S3 EFS United States (N. Virginia) AmeyaEkbote

EC2 > Instances > i-016ce9d5e7ebd0007 > Connect to instance

Instance ID  
i-016ce9d5e7ebd0007 (ec2-redhat-linux)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is `virginia.pem`
3. Run this command, if necessary, to ensure your key is not publicly viewable.  
`chmod 400 "virginia.pem"`
4. Connect to your instance using its Public DNS:  
`ec2-54-226-10-52.compute-1.amazonaws.com`

Example:  
`ssh -i "virginia.pem" ec2-user@ec2-54-226-10-52.compute-1.amazonaws.com`

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

```
ec2-user@ip-172-31-26-2:~/efs$ sudo yum install nfs-utils gssproxy-0.9.2-10.el10.x86_64 libnfsidmap-1.2.8.2-3.el10.x86_64
Installing      : libverto-libev-0.3.2-10.el10.x86_64                                7/10
Running scriptlet: gssproxy-0.9.2-10.el10.x86_64                                8/10
Installing      : gssproxy-0.9.2-10.el10.x86_64                                8/10
Running scriptlet: gssproxy-0.9.2-10.el10.x86_64                                8/10
Running scriptlet: nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Installing      : nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Running scriptlet: nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Created symlink '/etc/systemd/system/multi-user.target.wants/nfs-client.target' -> '/usr/lib/systemd/system/nfs-client.target'.
Created symlink '/etc/systemd/system/remote-fs.target.wants/nfs-client.target' -> '/usr/lib/systemd/system/nfs-client.target'.

Warning: The unit file, source configuration file or drop-ins of gssproxy.service changed on disk. Run 'systemctl daemon-reload' to r
eload units.

Warning: The unit file, source configuration file or drop-ins of gssproxy.service changed on disk. Run 'systemctl daemon-reload' to r
eload units.

Installing      : sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64                            10/10
Running scriptlet: sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64                            10/10
Installed products updated.

Installed:
gssproxy-0.9.2-10.el10.x86_64                libev-4.33-14.el10.x86_64                libnfsidmap-1:2.8.2-3.el10.x86_64
libtirpc-1.3.5-1.el10.x86_64                libverto-libev-0.3.2-10.el10.x86_64      nfs-utils-1:2.8.2-3.el10.x86_64
quota-1:4.09-9.el10.x86_64                  quota-nls-1:4.09-9.el10.noarch           rpcbind-1.2.7-3.el10.x86_64
sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64

Complete!
[ec2-user@ip-172-31-26-2 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-079
70697d3c8f4670.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-26-2 ~]$ cd efs/
[ec2-user@ip-172-31-26-2 efs]$ ls
ameya_linux.txt  ameya_ubuntu.txt
[ec2-user@ip-172-31-26-2 efs]$
```

```
ec2-user@ip-172-31-26-2:~/efs$ sudo yum install nfs-utils gssproxy-0.9.2-10.el10.x86_64 libnfsidmap-1.2.8.2-3.el10.x86_64
Running scriptlet: gssproxy-0.9.2-10.el10.x86_64                                8/10
Running scriptlet: nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Installing      : nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Running scriptlet: nfs-utils-1:2.8.2-3.el10.x86_64                                9/10
Created symlink '/etc/systemd/system/multi-user.target.wants/nfs-client.target' -> '/usr/lib/systemd/system/nfs-client.target'.
Created symlink '/etc/systemd/system/remote-fs.target.wants/nfs-client.target' -> '/usr/lib/systemd/system/nfs-client.target'.

Warning: The unit file, source configuration file or drop-ins of gssproxy.service changed on disk. Run 'systemctl daemon-reload' to r
eload units.

Warning: The unit file, source configuration file or drop-ins of gssproxy.service changed on disk. Run 'systemctl daemon-reload' to r
eload units.

Installing      : sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64                            10/10
Running scriptlet: sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64                            10/10
Installed products updated.

Installed:
gssproxy-0.9.2-10.el10.x86_64                libev-4.33-14.el10.x86_64                libnfsidmap-1:2.8.2-3.el10.x86_64
libtirpc-1.3.5-1.el10.x86_64                libverto-libev-0.3.2-10.el10.x86_64      nfs-utils-1:2.8.2-3.el10.x86_64
quota-1:4.09-9.el10.x86_64                  quota-nls-1:4.09-9.el10.noarch           rpcbind-1.2.7-3.el10.x86_64
sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64

Complete!
[ec2-user@ip-172-31-26-2 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-079
70697d3c8f4670.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-26-2 ~]$ cd efs/
[ec2-user@ip-172-31-26-2 efs]$ ls
ameya_linux.txt  ameya_ubuntu.txt
[ec2-user@ip-172-31-26-2 efs]$ cat ameya_linux.txt
"Hello Linux !!"
[ec2-user@ip-172-31-26-2 efs]$
```

Assignment is completed !!