

# AWS EC2 & EFS Assignment Submission

Name: Vikram

Course: AWS DevOps (Intellipaat)

Assignment: EC2 & EFS Configuration Task

## Problem Statement

You work for XYZ Corporation. Your corporation is working on an application that requires a shared storage solution accessible from multiple servers. The requirement is to configure Amazon Elastic File System (EFS) and mount it on three different EC2 instances running different operating systems: Ubuntu, Red Hat Enterprise Linux, and Amazon Linux 2.

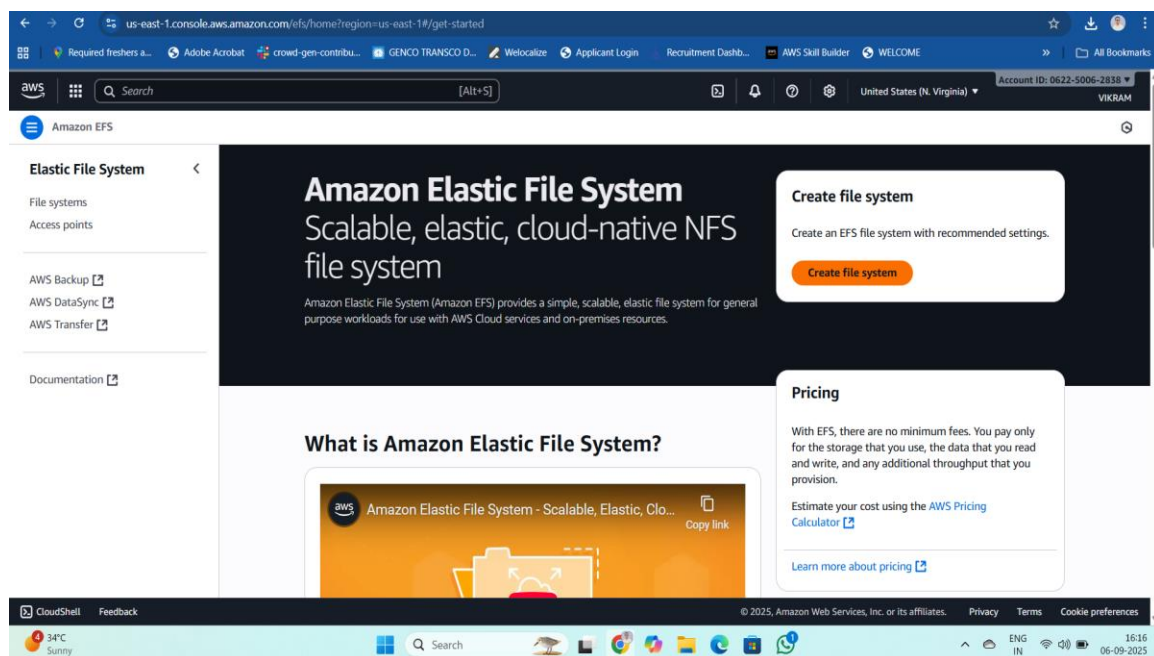
## Tasks Performed

### Task 1: Create an EFS File System

Steps taken:

1. Log in to AWS Console and set region to US-East-1 (N. Virginia).
2. Go to EFS → Create file system.
3. Provide name (e.g., efs multi os).
4. Select VPC and subnets matching the EC2 instances.
5. Ensure security groups default allow NFS (port 2049).
6. Create the file system.

Screenshot:



## General

Name - optional

Name your file system.

efs multi os

Name can include letters, numbers, and +-=,./ symbols, up to 256 characters.

## File system type

Choose to either store data across multiple Availability Zones or within a single Availability Zone. [Learn more](#)

- Regional

Offers the highest levels of availability and durability by storing file system data across multiple Availability Zones within an AWS Region.

☐ One Zone

Provides continuous availability to data within a single Availability Zone within an AWS Region.

### Automatic backups

Automatically backup your file system data with AWS Backup using recommended settings. Additional pricing applies. [Learn more](#)

☒ Enable automatic backups

## Lifecycle management

Automatically save money as access patterns change by moving files into the Infrequent Access (IA) or Archive storage class. [Learn more](#)

### Transition into Infrequent Access

[Transition into Archive](#)

### Transition into Standard

### Performance settings

### Throughput mode

Choose a method for your file system's throughput limits. [Learn more](#)

- Enhanced

Provides more flexibility and higher throughput levels for workloads with a range of performance requirements.

☐ Bursting

Provides throughput that scales with the amount of storage for workloads with basic performance requirements.

- ☒ Elastic (Recommended)

Use this mode for workloads with unpredictable I/O. With Elastic Throughput, performance automatically scales with your workload activity and you only pay for the throughput you use (data transferred for your file systems per month). [Learn more](#)

☐ Provisioned

Use this mode if you can estimate your workload's throughput requirements. With Provisioned mode, you configure your file system's throughput and pay for throughput provisioned.

► Additional settings

► **Tags** optional

Step 1

File system settings

Step 2

**Network access**

Step 3 - optional

File system policy

Step 4

Review and create

## Network access

### Network

Virtual Private Cloud (VPC) [Learn more](#)

Choose the VPC where you want EC2 instances to connect to your file system.

vpc-001f9081dd4304264  
default

### Mount targets

A mount target provides an NFSv4 endpoint at which you can mount an Amazon EFS file system. We recommend creating one mount target per Availability Zone. [Learn more](#)

Availability zone	Subnet ID	IP address type	IPv4 address	IPv6 address	Security groups	
us-east-1a	subnet-0...	IPv4 only	Optional	-	Choose se... sg-02a5bc41d a22b8c9a default	<div>Remove</div>
us-east-1b	subnet-0...	IPv4 only	Optional	-	Choose se... sg-02a5bc41d a22b8c9a	<div>Remove</div>

aws

Search

[Alt+S]

United States (N. Virginia)

VIKRAM

Amazon EFS

File systems

Elastic File System

File systems

Access points

AWS Backup

AWS DataSync

AWS Transfer

Documentation

Success!

File system (fs-045edd7be19d37cbd) is available.

View file system

File systems (1)

Filter by property values

View details

Delete

Create file system

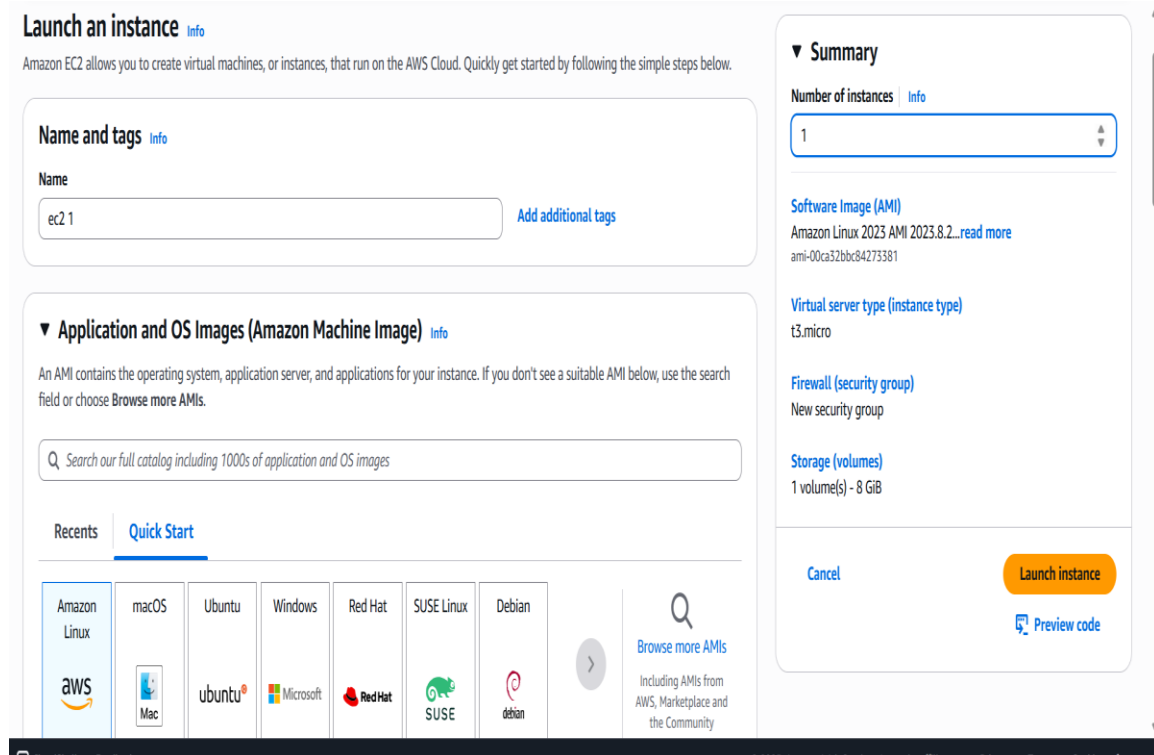
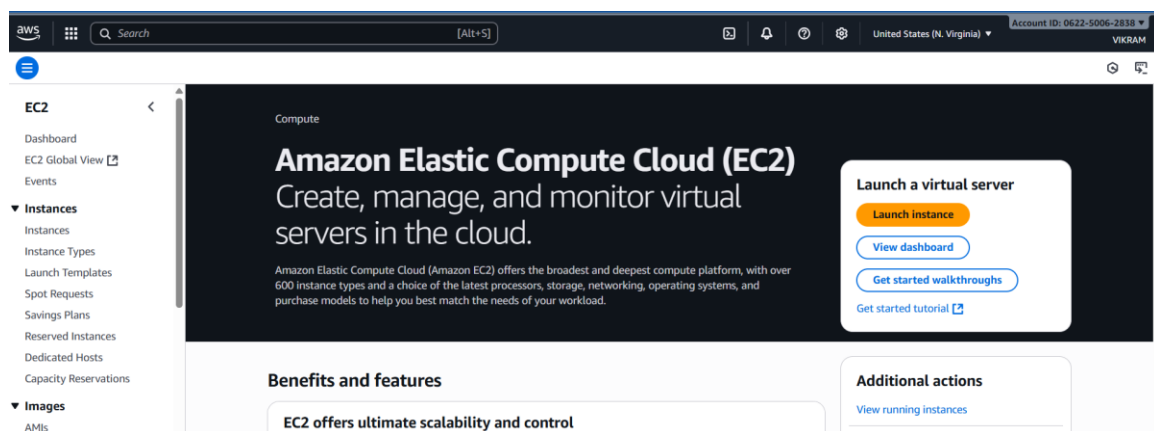
Name	File system ID	Encrypt	Total size	Size in Standard	Size in IA	Size in Archive	Provisioned Throughput (MiB/s)	File system state
<div>efs multi os</div>	<div>fs-045edd7be19d37cbd</div>	<div>Encrypt</div>	6.00 KiB	6.00 KiB	0 Bytes	0 Bytes	-	<div>Avail</div>

## Task 2: Launch 3 EC2 Instances (Ubuntu, RHEL, Amazon Linux 2)

Steps taken:

1. Go to EC2 → Launch instances.
2. Provide names (Ubuntu, RHEL, AmazonLinux2).
3. Select respective AMIs: Ubuntu 20.04 LTS, Red Hat Enterprise Linux, Amazon Linux 2.
4. Choose instance type (t3.micro for free tier).
5. Select existing key pair or create new for SSH access.
6. Ensure default inbound rules allow SSH (22) and NFS (2049).
7. Launch instances.

Screenshot:



▼ Instance type [Info](#) [Get advice](#)

Instance type

t3.micro

Family: t3 2 vCPU 1 GiB Memory Current generation: true

On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour

On-Demand SUSE base pricing: 0.0104 USD per Hour

On-Demand RHEL base pricing: 0.0392 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Proceed without a key pair (Not recommended)

Default value ▼

Create new key pair

▼ Network settings [Info](#)

Network [Info](#)

Edit

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Amazon Linux 2023.8.2...[read more](#)

ami-00ca32bbc84273381

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel

Launch instance

Preview code

Name and tags [Info](#)

Name

ec2 2

Add additional tags

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

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

RedHat

SUSE Linux

SUSE

Debian

debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd6...[read more](#)

ami-0360c520857e3138f

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 8 GiB

Cancel

Launch instance

Preview code

## Launch an instance [Info](#)

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.

### Name and tags [Info](#)

Name

ec2.3

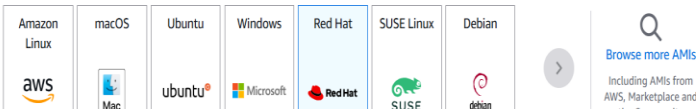
[Add additional tags](#)

### ▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Recents

**Quick Start**



### ▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Provided by Red Hat, Inc.  
ami-0fd3ac4abb734302a

Virtual server type (instance type)

t3.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 10 GiB

[Cancel](#)

[Launch instance](#)

[Preview code](#)

aws

Search

[Alt+S]

United States (N. Virginia)

Account ID: 0622-5006-2838

VIKRAM

EC2 > Instances

EC2

Dashboard

EC2 Global View

Events

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

Images

AMIs

AMI Catalog

Instances (3/3) [Info](#)

Last updated 1 minute ago

[Connect](#)

[Instance state](#)

[Actions](#)

[Launch instances](#)

All states

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
<input checked="" type="checkbox"/>	ec2.1	i-04c91efd6e3bf42c1	Running	t3.micro	3/3 checks passed	<a href="#">View alarms</a>	us-east-1a	ec2-54-54-90-...
<input checked="" type="checkbox"/>	ec2.2	i-0358cabb47f920bc7	Running	t3.micro	Initializing	<a href="#">View alarms</a>	us-east-1a	ec2-98-81-...
<input checked="" type="checkbox"/>	ec2.3	i-0b5862dd3cdeef766	Running	t3.micro	Initializing	<a href="#">View alarms</a>	us-east-1a	ec2-52-90-...

3 instances selected

[Monitoring](#)

[Investigate with AI - new](#)

1h 3h 12h 1d 3d 1w Custom

UTC timezone

[Configure CloudWatch agent](#)

[Explore related](#)

## Security Groups (1/4) [Info](#)

[Actions](#)

[Export security groups to CSV](#)

[Create security group](#)

<input type="checkbox"/>	Name	Security group ID	Security group name	VPC ID	Description
<input type="checkbox"/>	-	sg-077ccc428aa3839e4	launch-wizard-2	vpc-001f9081dd4304264	launch-wizard-2 c
<input type="checkbox"/>	-	sg-03f16cc74592e63f3	launch-wizard-1	vpc-001f9081dd4304264	launch-wizard-1 c
<input type="checkbox"/>	-	sg-050b16da73b372728	launch-wizard-3	vpc-001f9081dd4304264	launch-wizard-3 c
<input checked="" type="checkbox"/>	-	sg-02a5bc41da22b8c9a	default	vpc-001f9081dd4304264	default VPC secur

### sg-02a5bc41da22b8c9a - default

#### Inbound rules (2)

<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range
<input type="checkbox"/>	-	sgr-058e2b529e052d65a	IPv4	NFS	TCP	2049
<input type="checkbox"/>	-	sgr-0a5a0cc71c7e07588	-	All traffic	All	All

## Task 3: Connect each instance through EC2 Instance Connect & Install NFS Utilities

Steps taken:

1. Select the instance you want to connect and click on connect and connect it using EC2 instance connect.
2. Repeat the above step for other instances.

### Installing NFS utilities

- Ubuntu:

```
sudo apt-get update  
sudo apt-get install nfs-common
```

- Amazon Linux 2:

```
sudo yum update  
sudo yum install nfs-utils
```

- RHEL:

```
sudo yum update  
sudo yum install nfs-utils
```

Screenshot:

### 1. UBUNTU

The screenshot displays the AWS Management Console interface. At the top, the 'Instances' page is shown with a search bar and filters. A table lists three EC2 instances, all in a 'Running' state. The second instance, 'ec2 2' with ID 'i-0358cabb47f920bc7', is selected. Below the table, the details for this instance are expanded, showing its configuration, including the public IPv4 address '98.81.202.185' and private IPv4 address '172.31.20.14'.

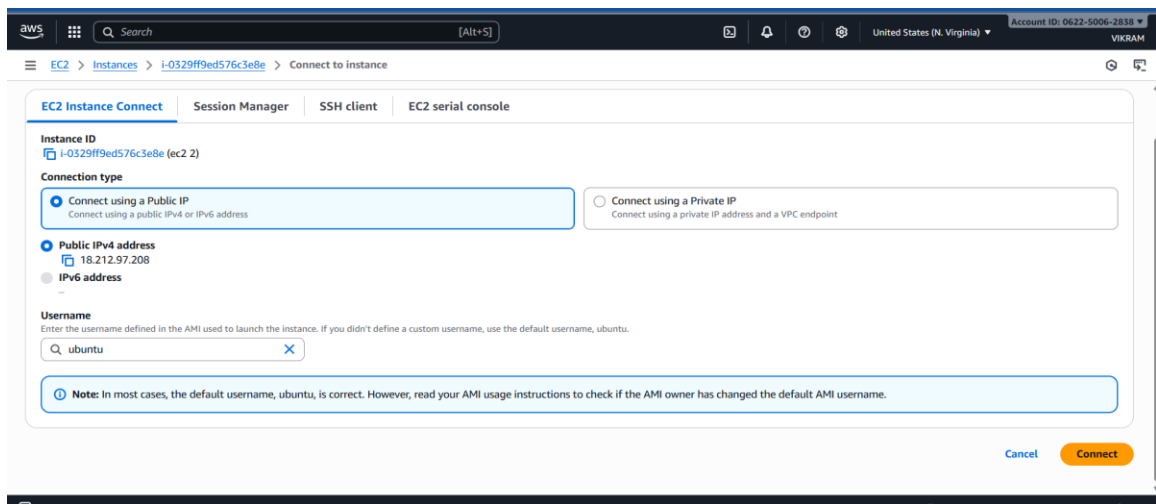
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
ec2 1	i-04c91efd6e3bf42c1	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-54-90-7
<b>ec2 2</b>	<b>i-0358cabb47f920bc7</b>	<b>Running</b>	<b>t3.micro</b>	<b>3/3 checks passed</b>	<b>View alarms +</b>	<b>us-east-1a</b>	<b>ec2-98-81-2</b>
ec2 3	i-0b5862dd3cdeef766	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-52-90-5

**i-0358cabb47f920bc7 (ec2 2)**

**Details** | Status and alarms | Monitoring | Security | Networking | Storage | Tags

**Instance summary**

Instance ID i-0358cabb47f920bc7	Public IPv4 address 98.81.202.185   open address	Private IPv4 addresses 172.31.20.14
IPv6 address	Instance state	Public DNS



```
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1011-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Sun Sep  7 04:53:04 UTC 2025

System load:  0.03           Temperature:   -273.1 C
Usage of /:   29.3% of 6.71GB Processes:    118
Memory usage: 24%           Users logged in: 0
Swap usage:   0%            IPv4 address for ens5: 172.31.16.121

Expanded Security Maintenance for Applications is not enabled.

19 updates can be applied immediately.
17 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Sun Sep  7 04:51:37 2025 from 18.206.107.28
ubuntu@ip-172-31-16-121:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
```

```
ubuntu@ip-172-31-16-121:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
ubuntu@ip-172-31-16-121:~$ sudo apt-get install nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nfs-common is already the newest version (1:2.6.4-3ubuntu5.1).
0 upgraded, 0 newly installed, 0 to remove and 13 not upgraded.
```



## 2. AMAZON LINUX

EC2 > Instances > i-04c91efd6e3bf42c1 > Connect to instance

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-04c91efd6e3bf42c1 (ec2 1)

Connection type

☒ Connect using a Public IP

Connect using a public IPv4 or IPv6 address

☐ Connect using a Private IP

Connect using a private IP address and a VPC endpoint

☒ Public IPv4 address

54.90.77.115

☐ IPv6 address

-

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.

ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Connect

```
'_#_
~\_####_      Amazon Linux 2023
~~\_#####\
~~\_###|
~~\_#/_____ https://aws.amazon.com/linux/amazon-linux-2023
~~_V~' '->
~~~~_/
~~._.-/_/
___/_/_/
__/_m/'
Last login: Sat Sep 6 10:58:25 2025 from 18.206.107.28
[ec2-user@ip-172-31-29-111 ~]$ sudo yum install nfs-utils
Last metadata expiration check: 0:01:09 ago on Sat Sep 6 10:58:57 2025.
Package nfs-utils-1:2.5.4-2.rc3.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
```

### 3. RHEL

**NOTE:** I tried to connect to this RHEL instance using EC2 Instance Connect, but it always showed an error. I checked all the required settings many times, but it still didn't work. Then I tried connecting through **Windows PowerShell**, and it worked.

The screenshot shows the AWS Management Console 'Instances' page. At the top, there's a search bar and buttons for 'Connect', 'Instance state', 'Actions', and 'Launch instances'. Below the search bar is a table of instances. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. Three instances are listed: ec2 1, ec2 2, and ec2 3. Instance ec2 3 is selected. Below the table, the details for instance i-07cbd3947ae131a7a (ec2 3) are shown. The 'Details' tab is active, displaying the instance summary with fields for Instance ID, Public IPv4 address, Private IPv4 addresses, IPv6 address, Instance state, and Public DNS.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
ec2 1	i-06ed998938ff6ca49	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-3-90-2f
ec2 2	i-0329ff9ed576c3e8e	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-18-212
ec2 3	i-07cbd3947ae131a7a	Running	t3.micro	3/3 checks passed	View alarms +	us-east-1a	ec2-3-85-9f

**i-07cbd3947ae131a7a (ec2 3)**

**Details** | Status and alarms | Monitoring | Security | Networking | Storage | Tags

**▼ Instance summary** Info

Instance ID i-07cbd3947ae131a7a	Public IPv4 address 3.85.93.254   open address	Private IPv4 addresses 172.31.26.133
IPv6 address -	Instance state Running	Public DNS ec2-3-85-93-254.compute-1.amazonaws.com   open

The screenshot shows the 'Connect to instance' dialog in the AWS Management Console. The 'EC2 Instance Connect' tab is selected. The dialog displays the instance ID i-07cbd3947ae131a7a (ec2 3). Under 'Connection type', there are two options: 'Connect using a Public IP' (selected) and 'Connect using a Private IP'. The 'Public IP' option is further divided into 'Public IPv4 address' (selected) and 'IPv6 address'. The 'Public IPv4 address' field shows 3.85.93.254. The 'Username' field shows ec2-user. A note at the bottom states: 'Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.' The 'Connect' button is highlighted in orange.

**EC2 Instance Connect** | Session Manager | SSH client | EC2 serial console

Instance ID  
i-07cbd3947ae131a7a (ec2 3)

Connection type

☒ Connect using a Public IP  
Connect using a public IPv4 or IPv6 address

☐ Connect using a Private IP  
Connect using a private IP address and a VPC endpoint

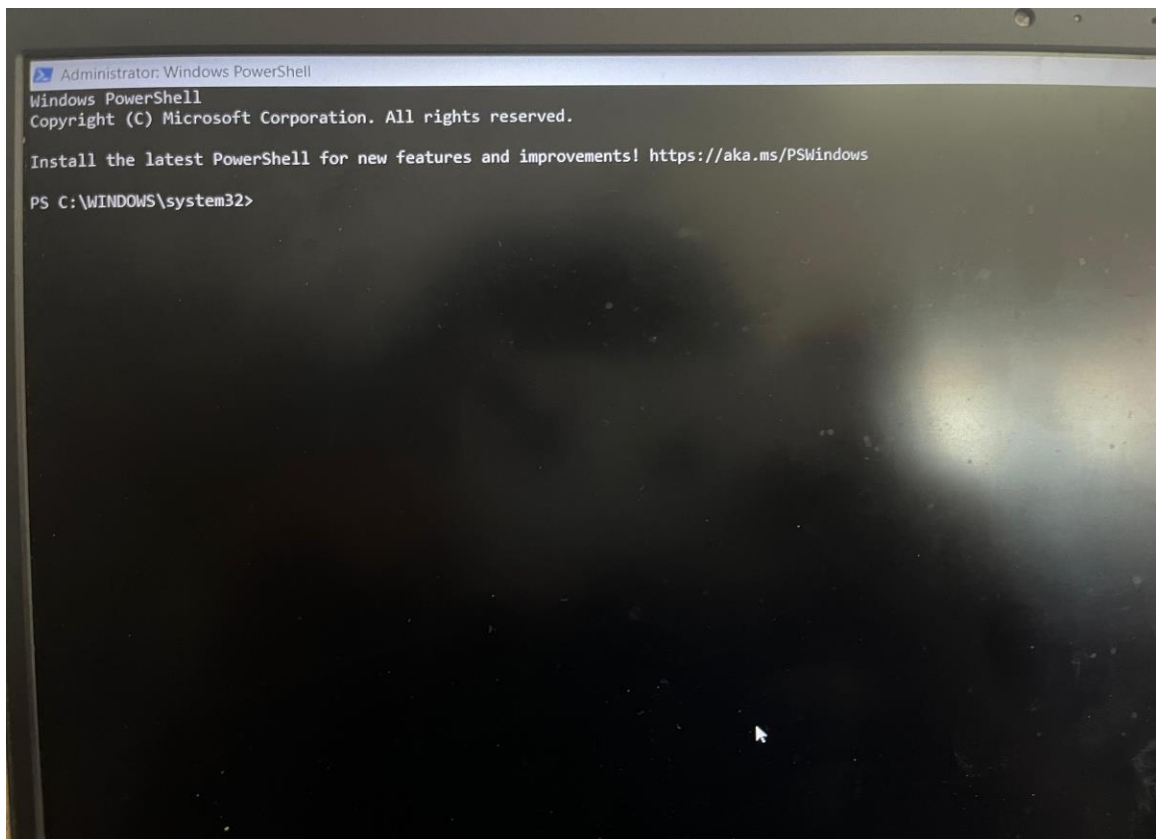
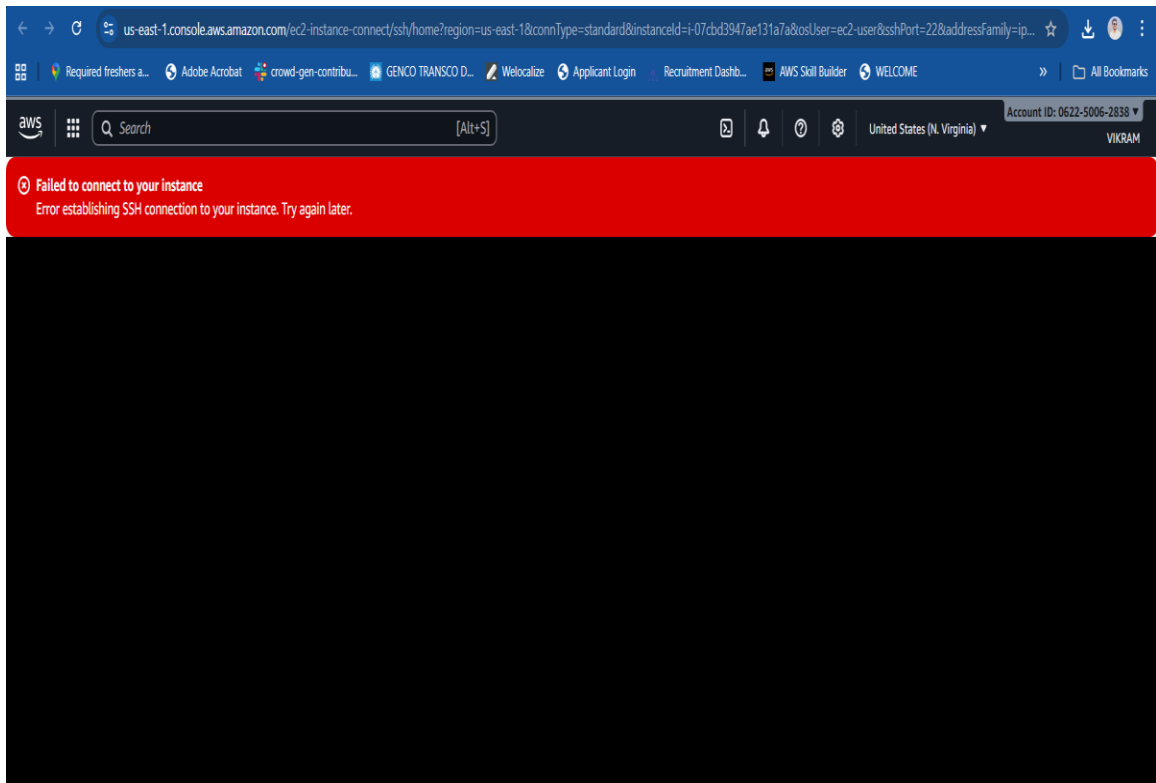
☒ Public IPv4 address  
3.85.93.254

☐ IPv6 address  
-

Username  
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ec2-user.  
ec2-user

Note: In most cases, the default username, ec2-user, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel | **Connect**



ec2-user@ip-172-31-16-58:~

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\WINDOWS\system32> ssh -i C:\Users\Vikram\Downloads\ec2-vs-85.pem ec2-user@52.91.25.152
The authenticity of host '52.91.25.152 (52.91.25.152)' can't be established.
ED25519 key fingerprint is SHA256:+gKfdQ1fhxiMOViyMlV1KLp6BxRf64FoyjJVUSyB3U4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '52.91.25.152' (ED25519) to the list of known hosts.
Register this system with Red Hat Insights: rhc connect
```

Example:

```
# rhc connect --activation-key <key> --organization <org>
```

The rhc client and Red Hat Insights will enable analytics and additional management capabilities on your system.

View your connected systems at <https://console.redhat.com/insights>

You can learn more about how to register your system using rhc at <https://red.ht/registration>

```
[ec2-user@ip-172-31-16-58 ~]$
```

Running transaction

```
Preparing : 1/10
Installing : libtirpc-1.3.5-1.el10.x86_64 1/10
Installing : libnfsidmap-1:2.8.2-3.el10.x86_64 2/10
Running scriptlet: rpcbind-1.2.7-3.el10.x86_64 3/10
Installing : rpcbind-1.2.7-3.el10.x86_64 3/10
Running scriptlet: rpcbind-1.2.7-3.el10.x86_64 3/10
Created symlink '/etc/systemd/system/multi-user.target.wants/rpcbind.service' + '/usr/lib/systemd/system/rpcbind.service'.
Created symlink '/etc/systemd/system/sockets.target.wants/rpcbind.socket' + '/usr/lib/systemd/system/rpcbind.socket'.

Installing : quota-nls-1:4.09-9.el10.noarch 4/10
Installing : quota-1:4.09-9.el10.x86_64 5/10
Installing : libev-4.33-14.el10.x86_64 6/10
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 reload units.
Warning: The unit file, source configuration file or drop-ins of gssproxy.service changed on disk. Run 'systemctl daemon-reload' to reload 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

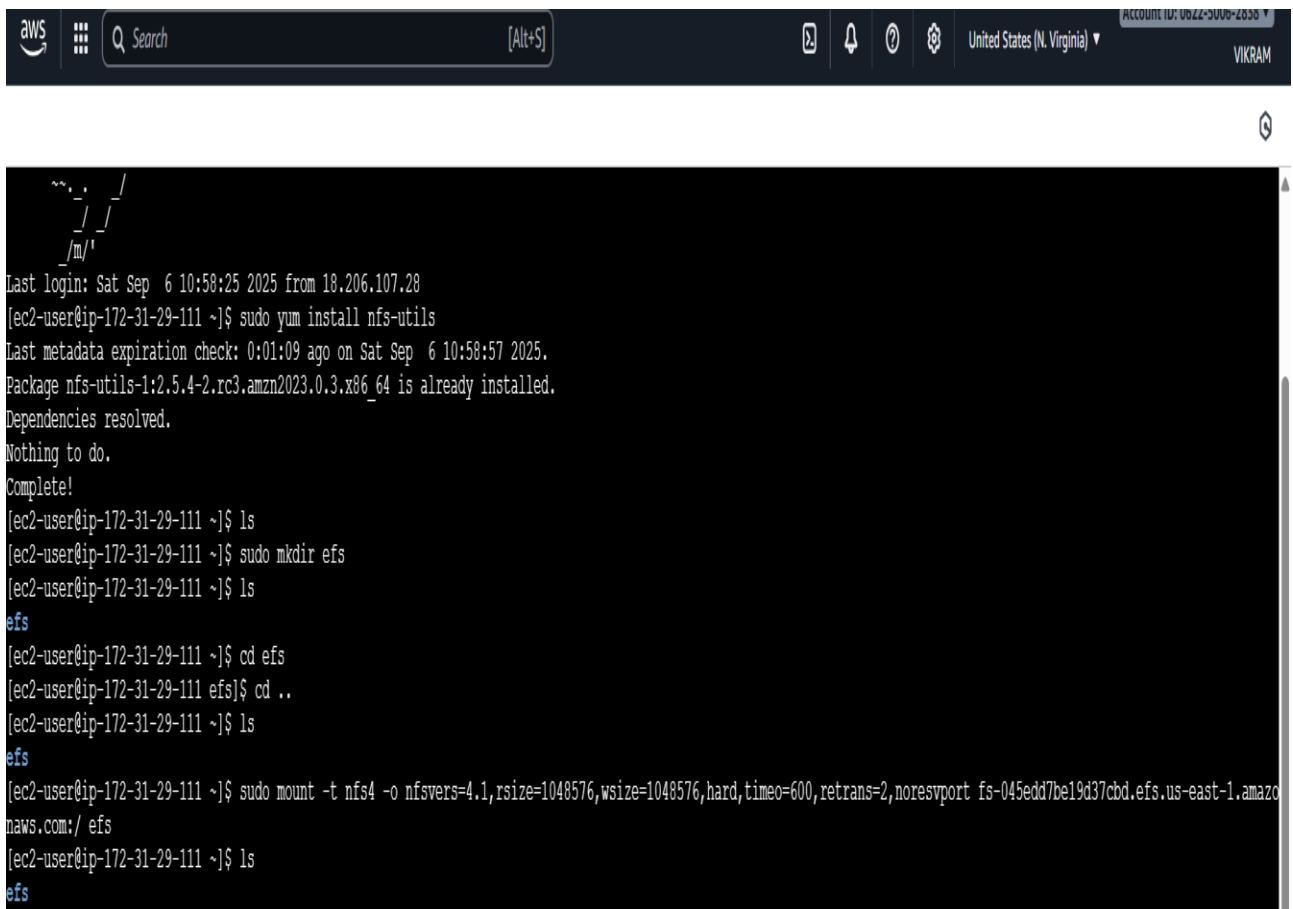
Complete!
```

Steps taken:

1. Create mount directory:  
`sudo mkdir efs`
2. Mount EFS file system
3. Verify mount with `df -h`.

Screenshot:

## 1. AMAZON LINUX



## 2. UBUNTU

```
ubuntu@ip-172-31-16-121:~$ ls
efs
ubuntu@ip-172-31-16-121:~$ cd efs
ubuntu@ip-172-31-16-121:~/efs$ cd ..
ubuntu@ip-172-31-16-121:~$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
ubuntu@ip-172-31-16-121:~$ ls
efs
```

## 3. RHEL

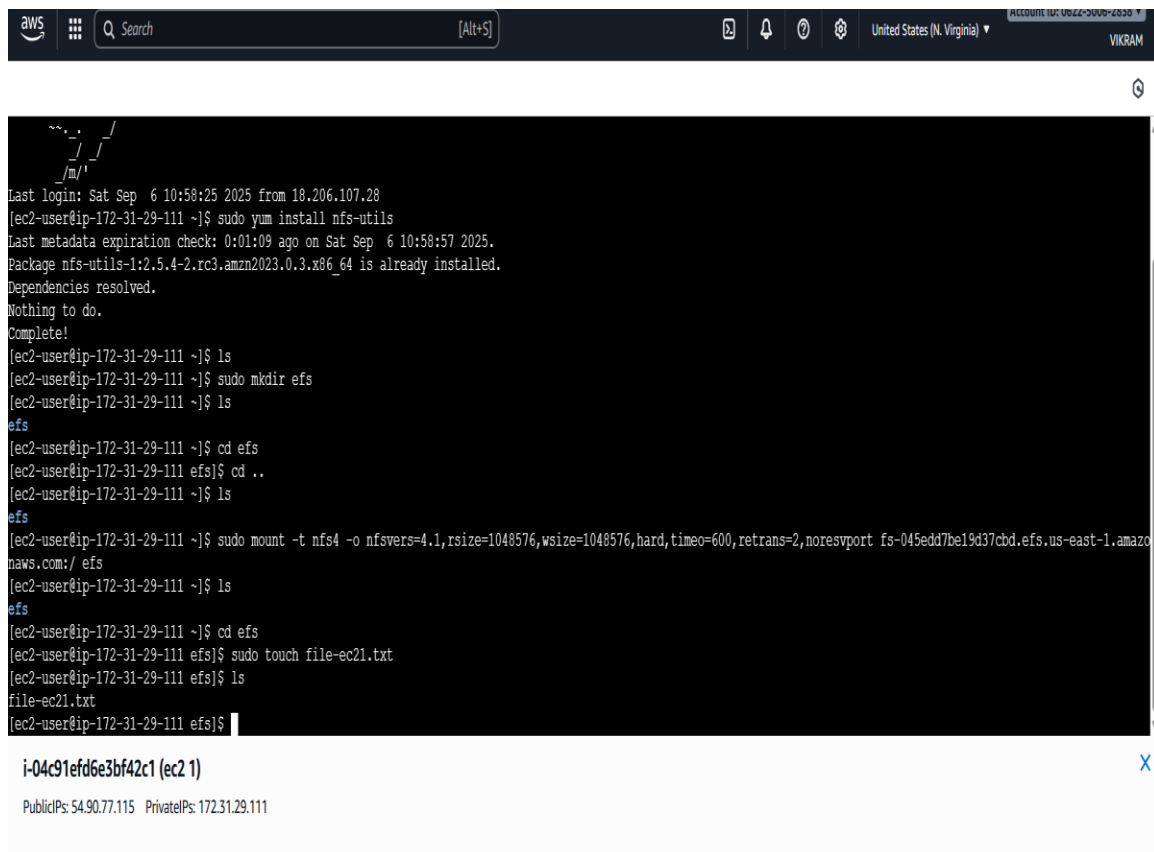
```
[ec2-user@ip-172-31-16-58 ~]$ ls
[ec2-user@ip-172-31-16-58 ~]$ sudo mkdir efs
[ec2-user@ip-172-31-16-58 ~]$ ls
efs
[ec2-user@ip-172-31-16-58 ~]$ cd efs
[ec2-user@ip-172-31-16-58 efs]$ cd ..
[ec2-user@ip-172-31-16-58 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-16-58 ~]$ ls
efs
```

## Task 5: Verify Shared Storage

Steps taken:

1. On Amazon Linux 2 instance:

- After mounting EFS, ran: `cd efs`
- Ran: `sudo touch file-ec1.txt`
- Checked with `ls`
- **Output:** `file-ec1.txt` (only this file visible initially on Amazon Linux)



```
aws
[Alt+S]
United States (N. Virginia)
VIKRAM

Last login: Sat Sep  6 10:58:25 2025 from 18.206.107.28
[ec2-user@ip-172-31-29-111 ~]$ sudo yum install nfs-utils
Last metadata expiration check: 0:01:09 ago on Sat Sep  6 10:58:57 2025.
Package nfs-utils-1:2.5.4-2.rc3.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-29-111 ~]$ ls
[ec2-user@ip-172-31-29-111 ~]$ sudo mkdir efs
[ec2-user@ip-172-31-29-111 ~]$ ls
efs
[ec2-user@ip-172-31-29-111 ~]$ cd efs
[ec2-user@ip-172-31-29-111 efs]$ cd ..
[ec2-user@ip-172-31-29-111 ~]$ ls
efs
[ec2-user@ip-172-31-29-111 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport fs-045edd7be19d37cbd.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-29-111 ~]$ ls
efs
[ec2-user@ip-172-31-29-111 ~]$ cd efs
[ec2-user@ip-172-31-29-111 efs]$ sudo touch file-ec1.txt
[ec2-user@ip-172-31-29-111 efs]$ ls
file-ec1.txt
[ec2-user@ip-172-31-29-111 efs]$
```

i-04c91efd6e3bf42c1 (ec2 1)

PublicIPs: 54.90.77.115 PrivateIPs: 172.31.29.111



## 2. On Ubuntu instance:

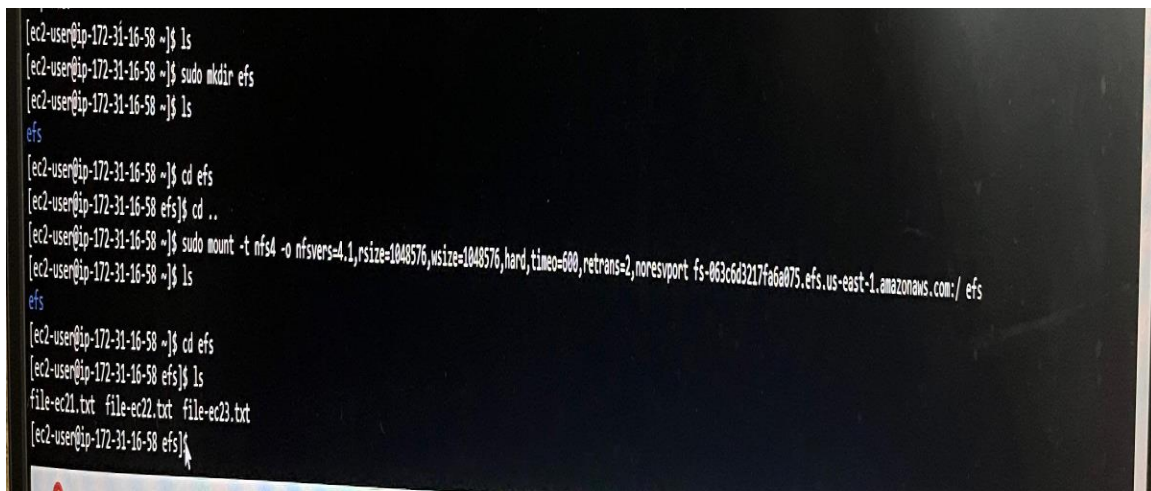
- After mounting EFS, ran: `cd efs`
- Ran: `sudo touch file-ec2.txt`
- Checked with `ls`
- **Output:** `file-ec1.txt file-ec2.txt` (Amazon Linux file + Ubuntu visible here)

```
ubuntu@ip-172-31-16-121:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
ubuntu@ip-172-31-16-121:~$ sudo apt-get install nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nfs-common is already the newest version (1:2.6.4-3ubuntu5.1).
0 upgraded, 0 newly installed, 0 to remove and 13 not upgraded.
ubuntu@ip-172-31-16-121:~$ ls
efs
ubuntu@ip-172-31-16-121:~$ cd efs
ubuntu@ip-172-31-16-121:~/efs$ cd ..
ubuntu@ip-172-31-16-121:~$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
ubuntu@ip-172-31-16-121:~$ ls
efs
ubuntu@ip-172-31-16-121:~$ cd efs
ubuntu@ip-172-31-16-121:~/efs$ ls
file-ec21.txt
ubuntu@ip-172-31-16-121:~/efs$ sudo touch file-ec22.txt
ubuntu@ip-172-31-16-121:~/efs$ ls
file-ec21.txt file-ec22.txt
ubuntu@ip-172-31-16-121:~/efs$
```



3. On RHEL instance:

- After mounting EFS, ran: `cd efs`
- Ran: `sudo touch file-ec23.txt`
- Checked with `ls`
- **Output:** `file-ec1.txt file-ec2.txt file-ec23.txt` (all three now visible here)

A terminal window screenshot showing the process of mounting an Amazon EFS file system on a RHEL instance. The user navigates to the /efs directory, mounts the file system using 'sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsz=1048576,hard,timeo=600,retr=2,noresuport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs', and then lists the contents of the directory. The output shows three files: file-ec1.txt, file-ec2.txt, and file-ec23.txt.

```
[ec2-user@ip-172-31-16-58 ~]$ ls
[ec2-user@ip-172-31-16-58 ~]$ sudo mkdir efs
[ec2-user@ip-172-31-16-58 ~]$ ls
efs
[ec2-user@ip-172-31-16-58 ~]$ cd efs
[ec2-user@ip-172-31-16-58 efs]$ cd ..
[ec2-user@ip-172-31-16-58 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsz=1048576,hard,timeo=600,retr=2,noresuport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-16-58 ~]$ ls
efs
[ec2-user@ip-172-31-16-58 ~]$ cd efs
[ec2-user@ip-172-31-16-58 efs]$ ls
file-ec1.txt file-ec2.txt file-ec23.txt
[ec2-user@ip-172-31-16-58 efs]$
```

4. Finally, checking again on **all three instances** (`ls` in `/efs`):

- **Output on every instance:** `file-ec1.txt file-ec2.txt file-ec23.txt`
- This confirms that the files created from any instance are automatically available across all instances — proving successful shared storage through EFS.

```
~/.ssh/
./

[ec2-user@ip-172-31-19-248 ~]$ sudo yum update
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-19-248 ~]$ sudo yum install nfs-utils
Last metadata expiration check: 0:00:21 ago on Sun Sep 7 08:18:14 2025.
Package nfs-utils-1:2.5.4-2.rc3.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-19-248 ~]$ sudo mkdir efs
[ec2-user@ip-172-31-19-248 ~]$ ls
efs
[ec2-user@ip-172-31-19-248 ~]$ cd efs
[ec2-user@ip-172-31-19-248 efs]$ cd ..
[ec2-user@ip-172-31-19-248 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
[ec2-user@ip-172-31-19-248 ~]$ ls
efs
[ec2-user@ip-172-31-19-248 ~]$ cd efs
[ec2-user@ip-172-31-19-248 efs]$ ls
file-ec21.txt file-ec22.txt file-ec23.txt
[ec2-user@ip-172-31-19-248 efs]$
```

i-06d010ddbcb8cd2c0 (ec2 1)

PublicIPs: 54.82.75.202 PrivateIPs: 172.31.19.248

```
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.5) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-31-180:~$ sudo mkdir efs
ubuntu@ip-172-31-31-180:~$ ls
efs
ubuntu@ip-172-31-31-180:~$ cd efs
ubuntu@ip-172-31-31-180:~/efs$ cd ..
ubuntu@ip-172-31-31-180:~$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport fs-063c6d3217fa6a075.efs.us-east-1.amazonaws.com:/ efs
ubuntu@ip-172-31-31-180:~$ ls
efs
ubuntu@ip-172-31-31-180:~$ cd efs
ubuntu@ip-172-31-31-180:~/efs$ ls
file-ec21.txt file-ec22.txt file-ec23.txt
ubuntu@ip-172-31-31-180:~/efs$
```

i-05fba25ac60dff81 (ec2 2)

PublicIPs: 44.220.137.50 PrivateIPs: 172.31.31.180

```
Complete!
[ec2-user@ip-172-31-16-58 ~]$ ls
[ec2-user@ip-172-31-16-58 ~]$ sudo mkdir efs
[ec2-user@ip-172-31-16-58 ~]$ ls
efs
[ec2-user@ip-172-31-16-58 ~]$ cd efs
[ec2-user@ip-172-31-16-58 efs]$ cd ..
[ec2-user@ip-172-31-16-58 ~]$ sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576
efs
[ec2-user@ip-172-31-16-58 ~]$ cd efs
[ec2-user@ip-172-31-16-58 efs]$ ls
file-ec21.txt file-ec22.txt file-ec23.txt
[ec2-user@ip-172-31-16-58 efs]$
```

## Conclusion

Successfully created an Amazon EFS file system, launched three EC2 instances with different operating systems (Ubuntu, RHEL, Amazon Linux 2), installed NFS utilities, mounted the EFS on all instances, and verified shared file access. This demonstrates the use of Amazon EFS for scalable, shared storage across multiple servers.