



# **Discipline**

**Elastic File System**

**CS5002**

**Activity-20**

**Sheikh Muhammed Tadeeb (AU19B1014)**

---

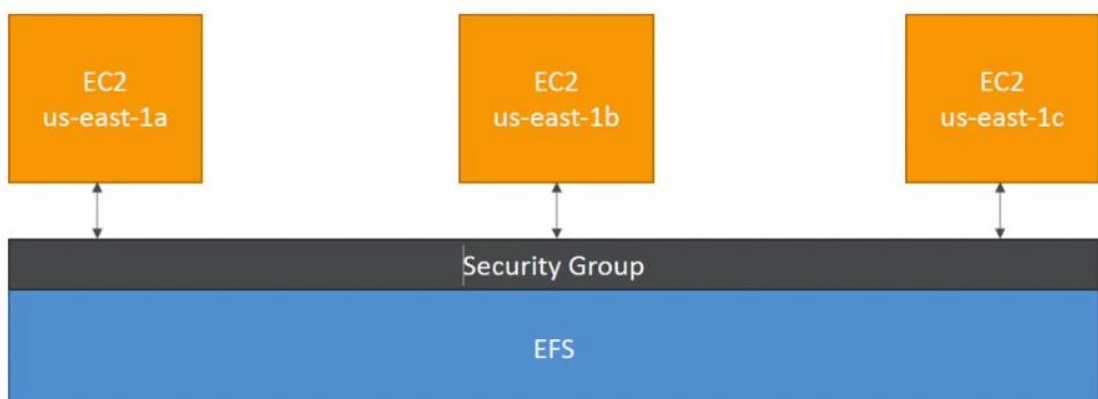
## ❖ Problem Statement:

Create an EFS file system and mount it to 2 Ec2 machines. Use the 2<sup>nd</sup> method shown in the class today. Copy file from one machine to another machine and check whether the EFS worked or not.

## ❖ Theory:

### ➤ Brief:

Amazon Elastic File System (Amazon EFS) provides a simple, serverless, set-and-forget elastic file system for use with AWS Cloud services and on-premises resources. It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth. Amazon EFS has a simple web services interface that allows us to create and configure file systems quickly and easily. The service manages all the file storage infrastructure for us, meaning that we can avoid the complexity of deploying, patching, and maintaining complex file system configurations.



### ➤ Protocol used by EFS:

Amazon EFS supports the Network File System version 4 (NFSv4.1 and NFSv4.0) protocol, so the applications and tools that you use today work seamlessly with Amazon EFS. Multiple compute instances, including Amazon EC2, Amazon ECS, and

AWS Lambda, can access an Amazon EFS file system at the same time, providing a common data source for workloads and applications running on more than one compute instance or server.

➤ **Storage Classes used by EFS:**

With Amazon EFS, we pay only for the storage used by our file system and there is no minimum fee or setup cost. Amazon EFS offers a range of storage classes designed for different use cases. These include:

- **Standard storage classes** – EFS Standard and EFS Standard–Infrequent Access (Standard–IA), which offer multi-AZ resilience and the highest levels of durability and availability.
- **One Zone storage classes** – EFS One Zone and EFS One Zone–Infrequent Access (EFS One Zone–IA), which offer customers the choice of additional savings by choosing to save their data in a single AZ’.

➤ **Security aspects of EFS:**

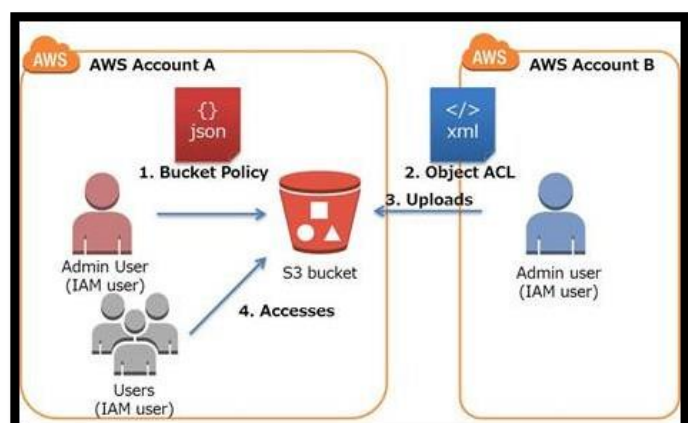
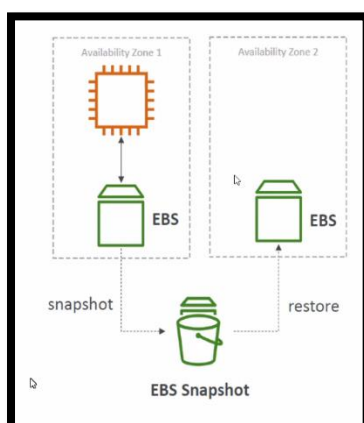
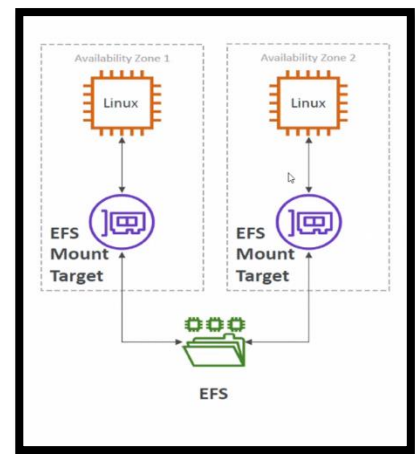
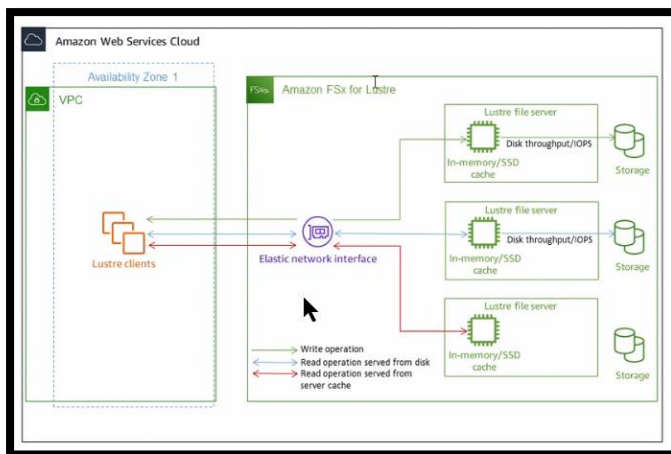
1. Amazon EFS also enables you to control access to your file systems through Portable Operating System Interface (POSIX) permissions.
2. Amazon EFS supports authentication, authorization, and encryption capabilities to help you meet our security and compliance requirements.
3. Amazon EFS supports two forms of encryption for file systems, encryption in transit and encryption at rest.
  - We can enable encryption at rest when creating an Amazon EFS file system. If we do this, all your data and metadata is encrypted.
  - We can also enable encryption in transit when we mount the file system.

### ➤ Modes of EFS:

With Amazon EFS, you can choose from two performance modes and two throughput modes:

- The default General Purpose performance mode is ideal for latency-sensitive use cases, like web serving environments, content management systems, home directories, and general file serving. File systems in the Max I/O mode can scale to higher levels of aggregate throughput and operations per second with a trade-off of slightly higher latencies for file metadata operations.
- Using the default Bursting Throughput mode, throughput scales as your file system grows. Using Provisioned Throughput mode, you can specify the throughput of your file system independent of the amount of data stored.

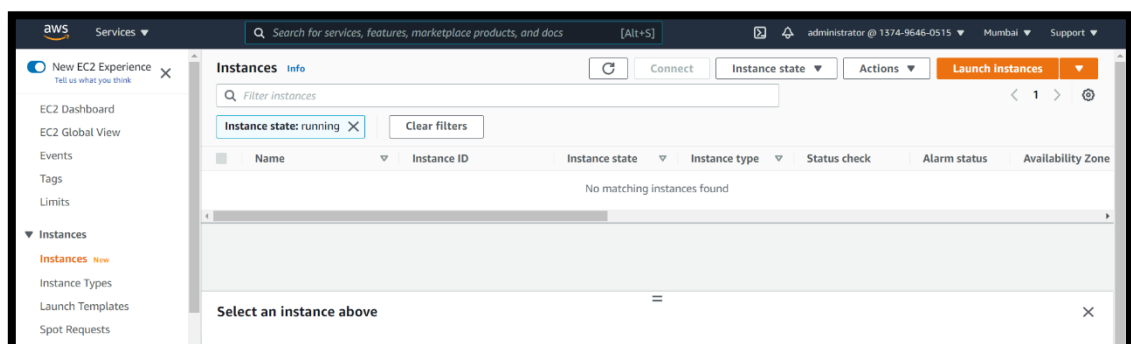
### ➤ Difference between EFS & EBS & S3 & FSx:



- EBS's use case is more easily understood than the other two. It must be paired with an EC2 instance. So, when you need a high-performance storage service for a single instance, use EBS.
- EFS may be used whenever you need a shared file storage option for multiple EC2 instances with automatic, high-performance scaling. This makes it a great candidate for file storage for content management systems, for lift and shift operations, as its autoscaling potential means you do not need to re-architect; for application development, as EFS's shareable file storage is ideal for storing code and media files.
- S3 is an object storage system, designed to provide archiving and data control options and to interface with other services beyond EC2. It's also useful for storing static html pages and shared storage for applications.
- As FSx uses Windows, it's compatible with all Window Server platforms. On the other hand, EFS uses NFS protocol which works on POSIX permissions and hence EFS is only supported by Linux machine. You get to choose from two services:
  1. Amazon FSx for Windows File Server - file storage for business apps.
  2. Amazon FSx for Lustre - shared storage for computing workloads.

## ❖ Solution:

- **Step1)** Create two instances (ex: VM1-AZ-1A & VM2-AZ-1B), make sure the two instances are in different subnets and both are Linux based as NFS is only supported on Linux Machines.



aws

Services

Q

Search for services, features, marketplace products, and docs

[Alt+S]

🔍

🔔

administrator @ 1374-9646-0515

Mumbai

Support

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

## Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Q

Search for an AMI by entering a search term e.g. "Windows"

X

Search by Systems Manager parameter

1 to 41 of 41 AMIs


Quick Start

My AMIs

AWS Marketplace

Community AMIs

☐ Free tier only




Amazon Linux

Free tier eligible

**Amazon Linux 2 AMI (HVM), SSD Volume Type**
ami-0a23ccb2cdd9286bb (64-bit x86) / ami-080c1148a83cea662 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and has been removed from this wizard.

Root device type: ebs    Virtualization type: hvm    ENA Enabled: Yes




Red Hat

Free tier eligible

**Red Hat Enterprise Linux 8 (HVM), SSD Volume Type**
ami-06a0b4e3b7eb7a300 (64-bit x86) / ami-0cbe04a3ce796c98e (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs    Virtualization type: hvm    ENA Enabled: Yes



SUSE Linux

Free tier eligible

**SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type**
ami-0b3ac3edf2397475 (64-bit x86) / ami-0ab71076ab9b53bdb (64-bit Arm)

Select

☒ 64-bit (x86)
☐ 64-bit (Arm)

Select

☒ 64-bit (x86)
☐ 64-bit (Arm)

Select

☒ 64-bit (x86)

Feedback

English (US)

© 2008 - 2021. Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

Cookie preferences

Services

[Alt+S]

administrator@1374-9646-0515
Mumbai
Support

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review

### Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by:

All instance families

Current generation

[Show/Hide Columns](#)

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro <small>Frees for up to 1 year</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes

[Cancel](#)
[Previous](#)
[Review and Launch](#)
[Next: Configure Instance Details](#)

aws

Services ▾

Search for services, features, marketplace products, and docs

[Alt+S]

administrator @ 1374-9646-0515 ▾

Mumbai ▾

Support ▾

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances

1

Launch into Auto Scaling Group ⓘ

Purchasing option

☐ Request Spot instances

Network

vpc-f99b4b92 | default (default) ▾

Create new VPC

Subnet

subnet-b506f8de | Default in ap-south-1a ▾

Create new subnet

4091 IP Addresses available

Auto-assign Public IP

Use subnet setting (Enable) ▾

Placement group

☐ Add instance to placement group

Capacity Reservation

Open ▾

Domain join directory

No directory ▾

Create new directory

IAM role

None ▾

Create new IAM role

Shutdown behavior

Stop ▾

Cancel

Previous

Review and Launch

Next: Add Storage

**Step 5: Add Tags**

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.  
A copy of a tag can be applied to volumes, instances or both.  
Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	Value	Instances	Volumes	Network Interfaces
Name	VM1-AZ-1A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

**Step 3: Configure Instance Details**

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Number of Instances**  [Launch into Auto Scaling Group](#)

**Purchasing option** ☐ Request Spot instances

**Network**  [Create new VPC](#)

**Subnet**  [Create new subnet](#)  
4091 IP Addresses available

**Auto-assign Public IP**

**Placement group** ☐ Add instance to placement group

**Capacity Reservation**

**Domain join directory**  [Create new directory](#)

**IAM role**  [Create new IAM role](#)

**Shutdown behavior**

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

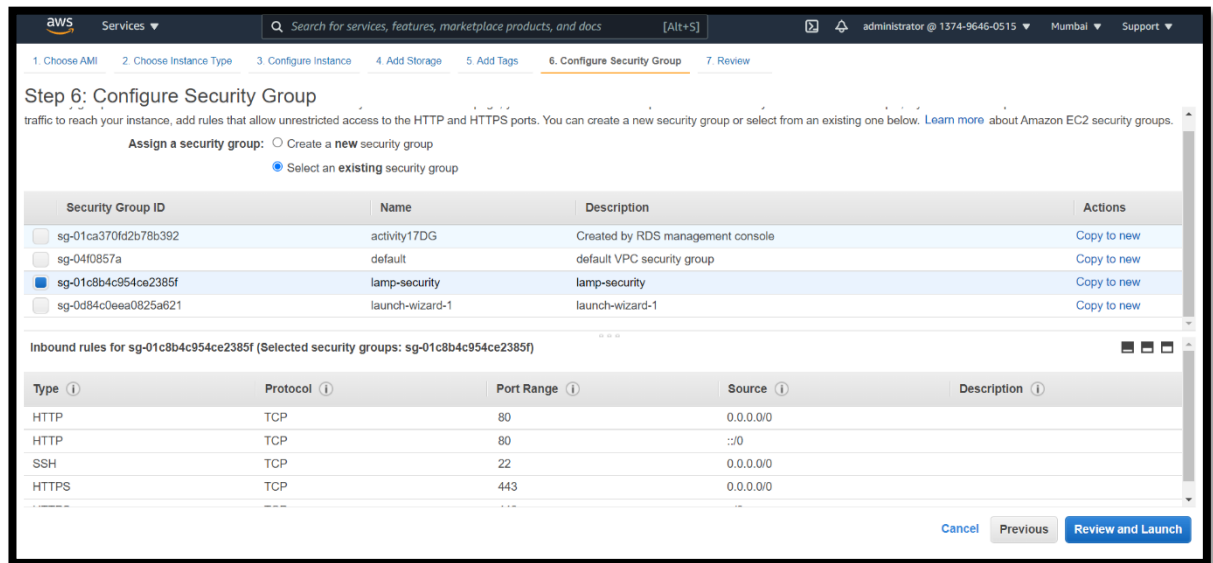
**Instances (2)** [Info](#) [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	VM1-AZ-1A	i-00da42ecb88bbd6f3	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a
<input type="checkbox"/>	VM2-AZ-1B	i-07e7b3c1e611314c7	Running	t2.micro	-	No alarms	ap-south-1b

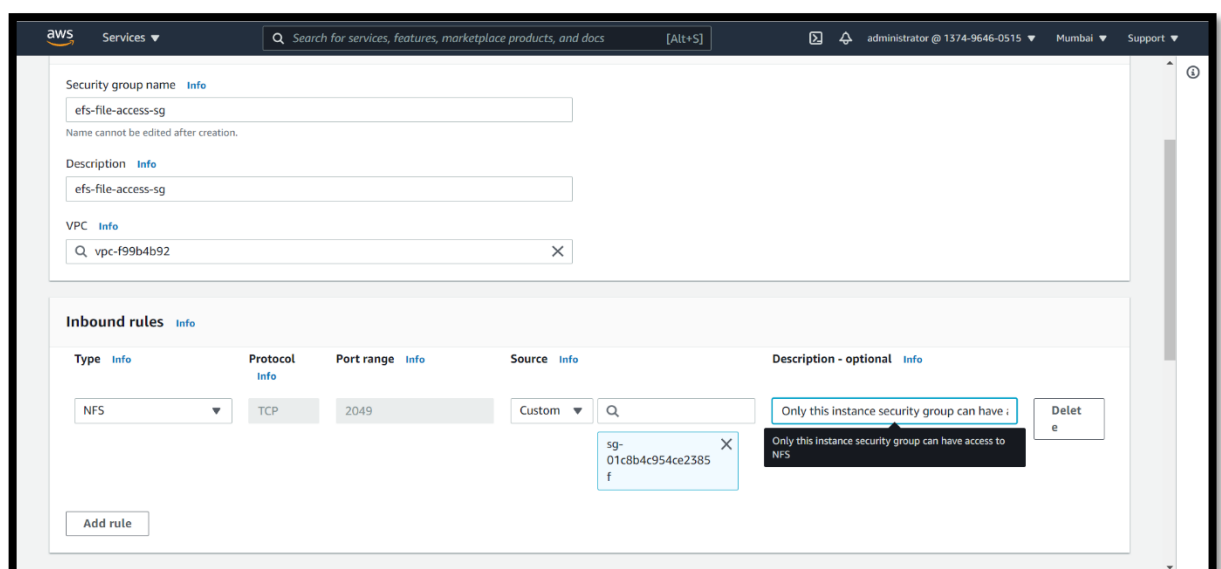
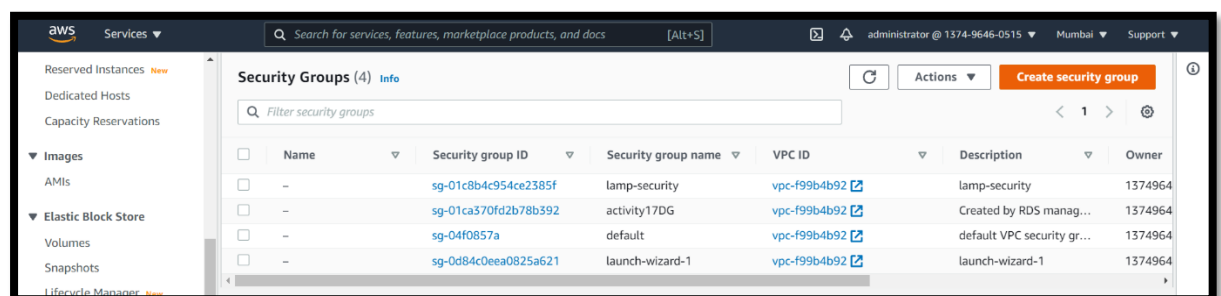
Select an instance above

© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#) [Cookie preferences](#)

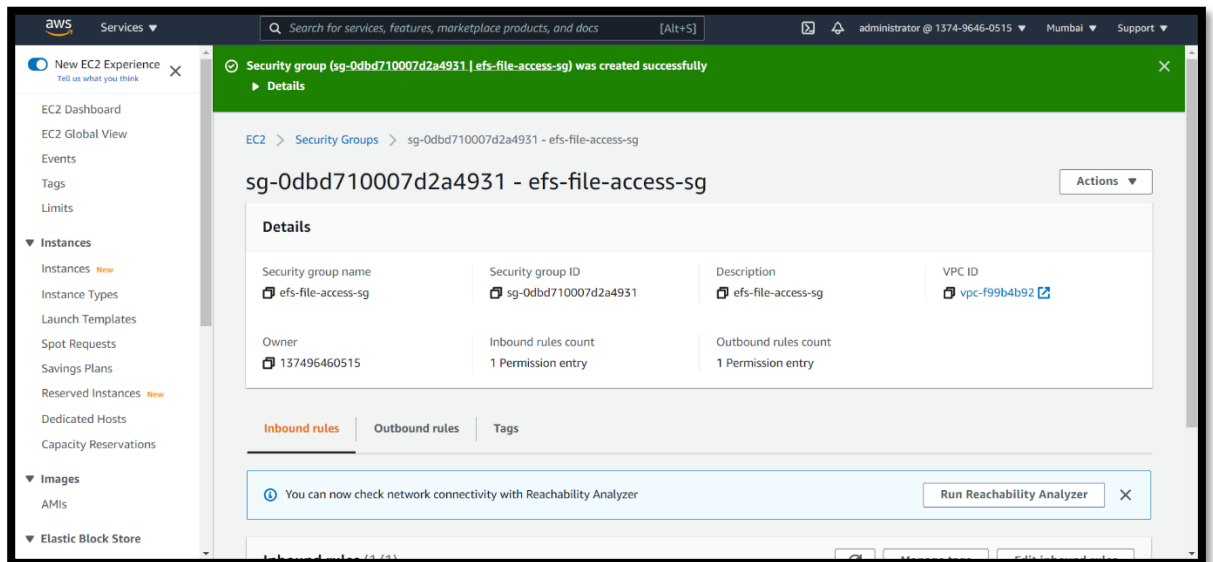
- **Step2)** Add Security group having ports 22 i.e., SSH, port 80 i.e., HTTP and port 443 i.e., HTTPS enabled to both the machines.



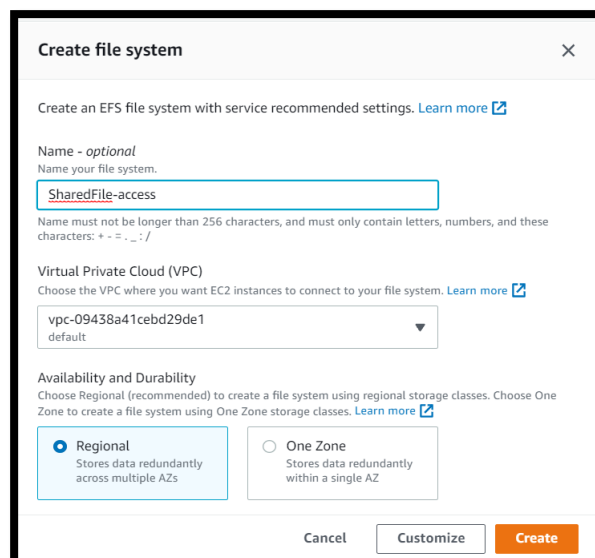
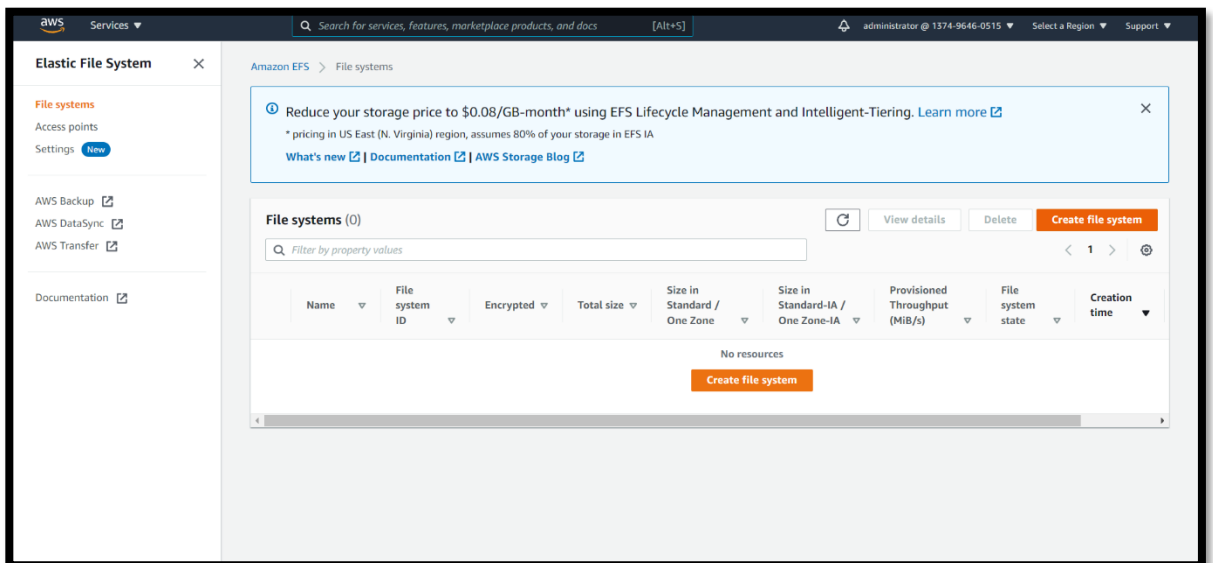
- **Step3)** Create a security group for EFS (example: efs-file-access-sg) with inbound rule as NFS and source is security group of ec2-instance.







- **Step4)** Create an EFS (example: SharedFile-access) under default VPC and regional AZ.

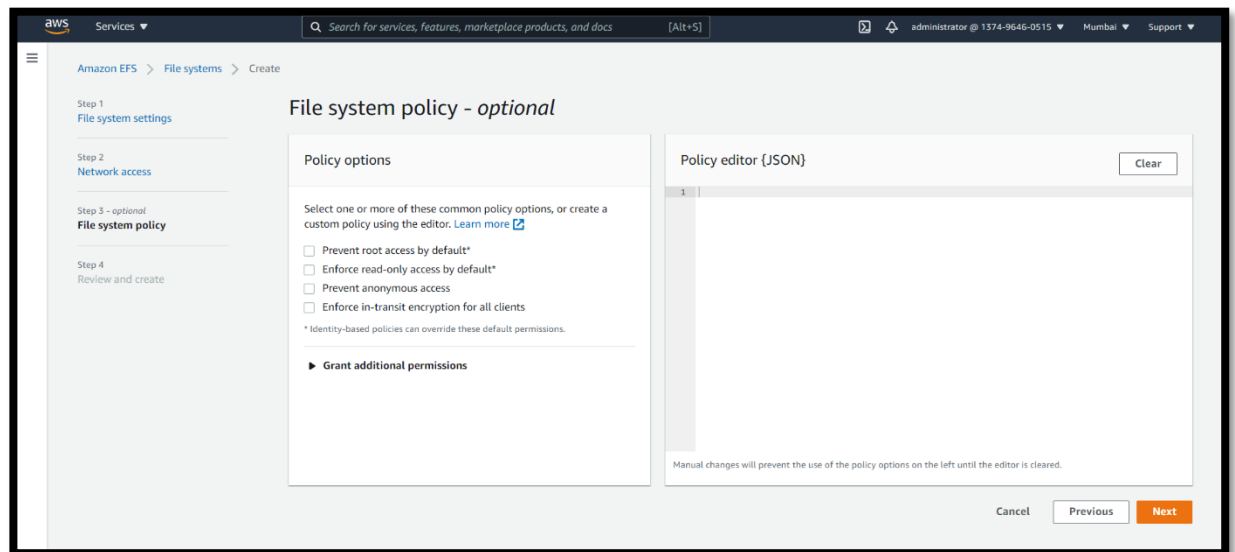


- **Step5)** Click on Customise in above step and *uncheck automatic backups* and *disable encryption* as we are making this EFS for testing purpose but when doing it for enterprises the scenario will be different. The click *Next*.

- **Step6)** In Network Access we'll make our NFS available to different Az's. Make sure to remove default security group from source and give security group of EFS in source. Click *Next*.

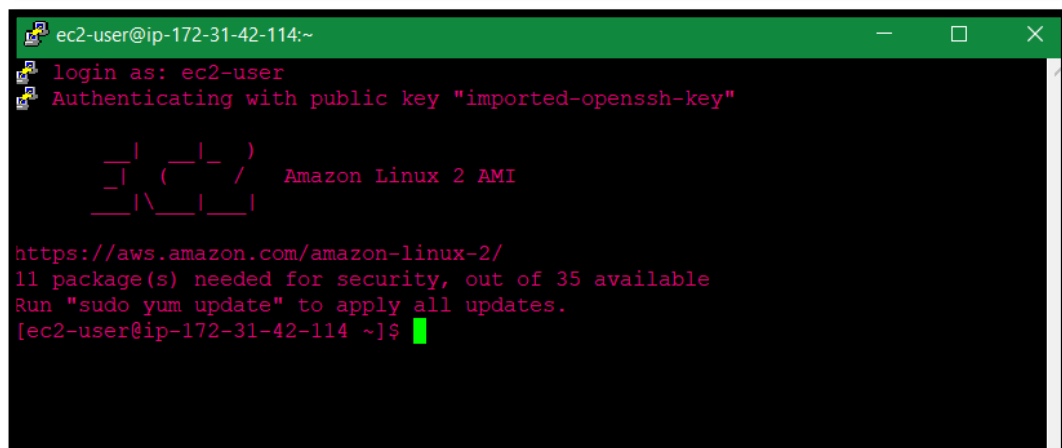
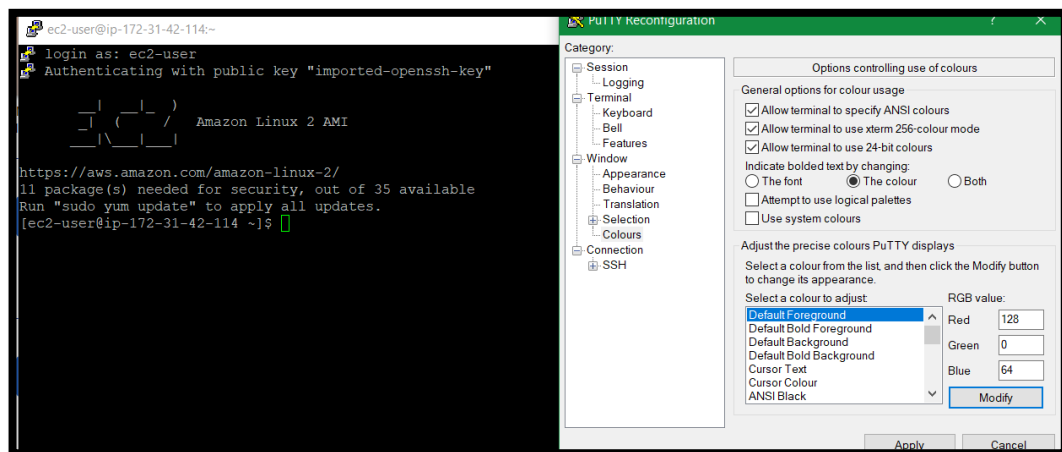
Availability zone	Subnet ID	IP address	Security groups
ap-south-1a	subnet-b506f8de	172.31.45.183	sg-0dbd710007d2a4931 (efs-file-access-sg)
ap-south-1b	subnet-17d1fe5b	172.31.5.237	sg-0dbd710007d2a4931 (efs-file-access-sg)
ap-south-1c	subnet-b45021cf	172.31.20.147	sg-0dbd710007d2a4931 (efs-file-access-sg)

- **Step7)** As this is a test EFS we won't mention any policies but when working in enterprises it is crucial to set policies. Click *Next*.

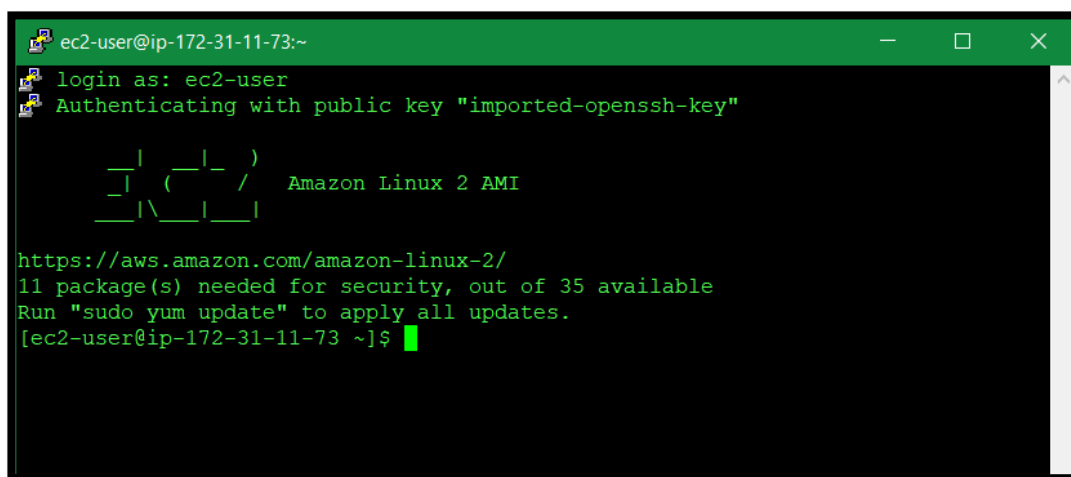
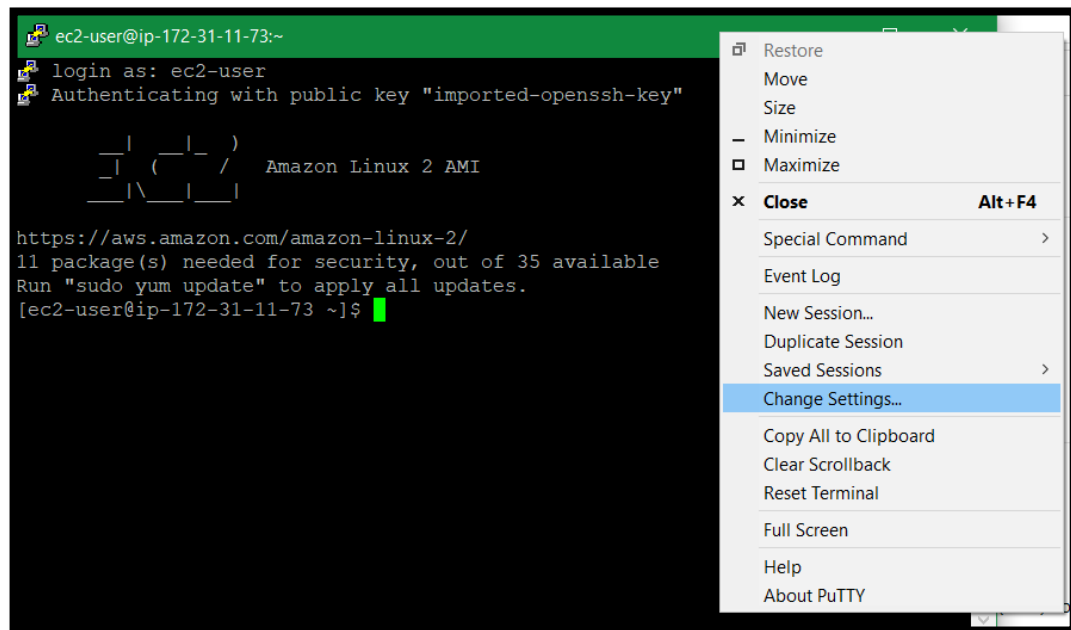


- **Step8)** Open putty and change two machines foreground colour so that we can rectify them easily.

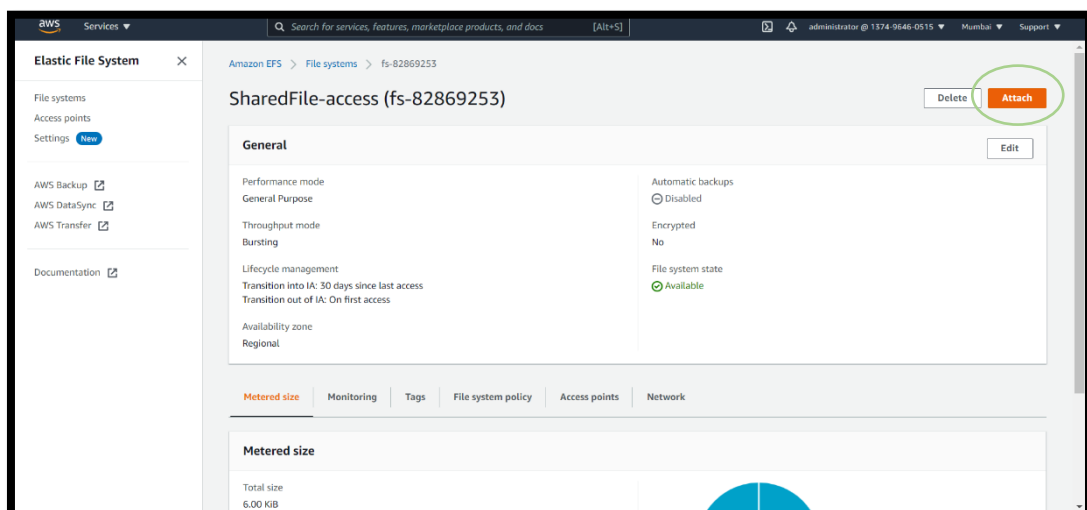
- **Machine 1 (VM1-AZ-1A):**

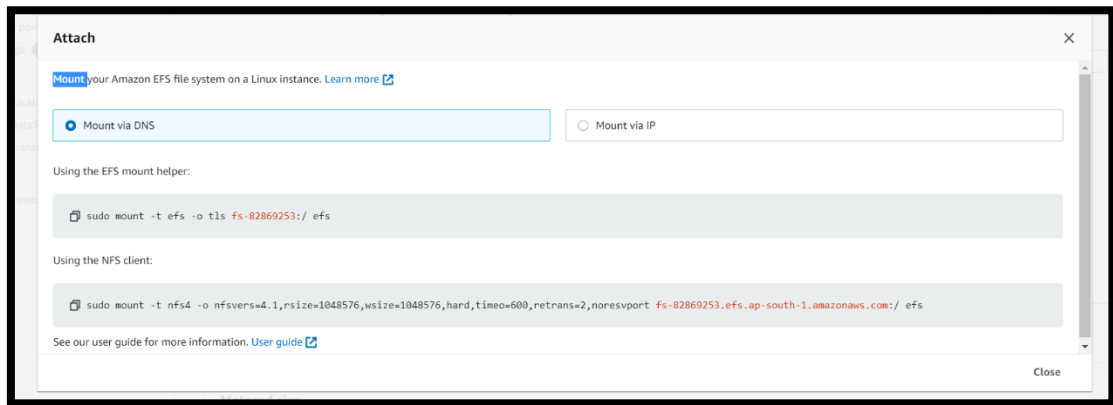


- **Machine 2 (VM2-AZ-1B):**



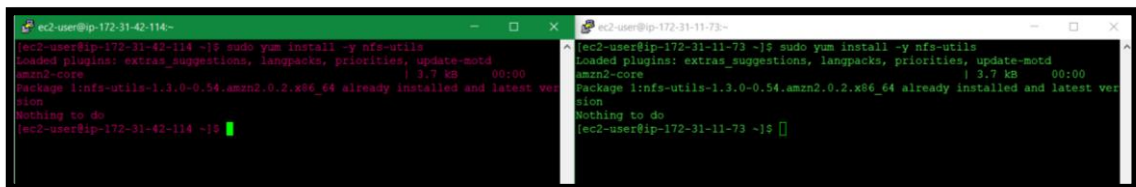
- **Step9)** Now we need to attach EFS to our both machines, Go to EFS and Click *Attach*.



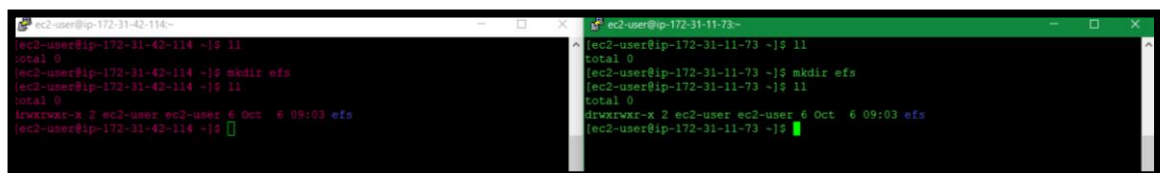


- **Step10)** Install NFS utility in both the machines with the command:

*sudo yum install -y nfs-utils*



- **Step11)** Create *efs* folder in both the machines.

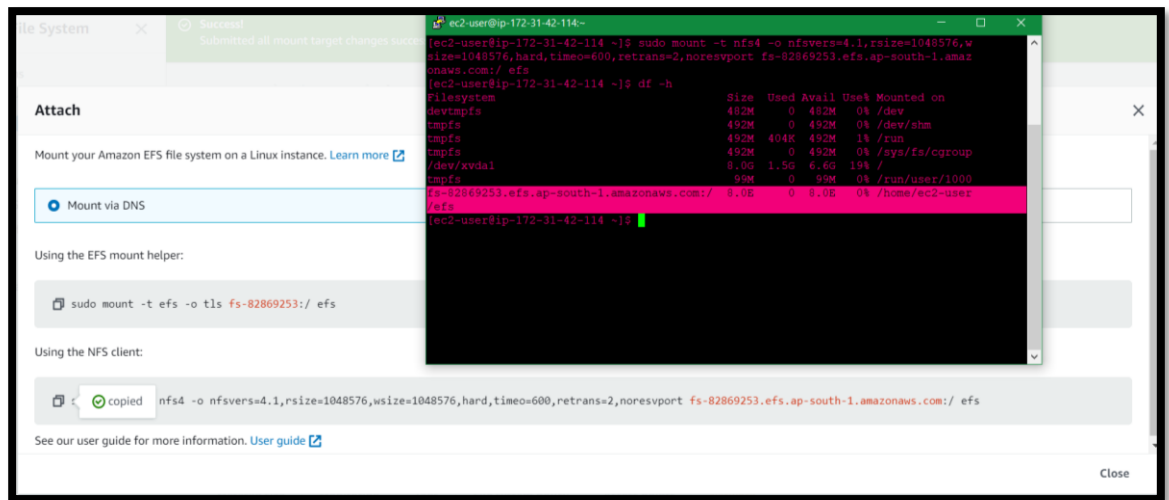


- **Step12)**

- Brief:

Now there exist two ways to mount our *efs* via DNS, so we will do with both ways. IN machine 1, I will do using *NFS client command* and in machine 2, I will do with EFS mount helper.

- **Step12.1)** Copy *NFS client command* and paste it in the putty machine-1 console, if no error comes that means our mount is done successfully and we can check it using the command *df -h*



- **Step12.2)** Go to Machine-2 and Mount EFS using a *EFS Mount Helper*. To use EFS Mount Helper we need to install another helper utility with the command:

`sudo yum install amazon-efs-utils -y`

After this copy Mount helper command and paste it in putty console of Machine-2.

If no error comes means our EFS is mounted properly and we can check mount points using the command:

`df -h`

```
ec2-user@ip-172-31-11-73:~$ sudo yum install amazon-efs-utils -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                1.3.7 kB  00:00
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.31.2-1.amzn2 will be installed
--> Processing Dependency: stunnel >= 4.56 for package: amazon-efs-utils-1.31.2-1.amzn2.noarch
--> Running transaction check
--> Package stunnel.x86_64 0:4.56-6.amzn2.0.3 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch          Version           Repository        Size
=====
Installing:
amazon-efs-utils        noarch        1.31.2-1.amzn2    amzn2-core        46 k
Installing for dependencies:
stunnel                 x86_64        4.56-6.amzn2.0.3  amzn2-core        149 k
=====

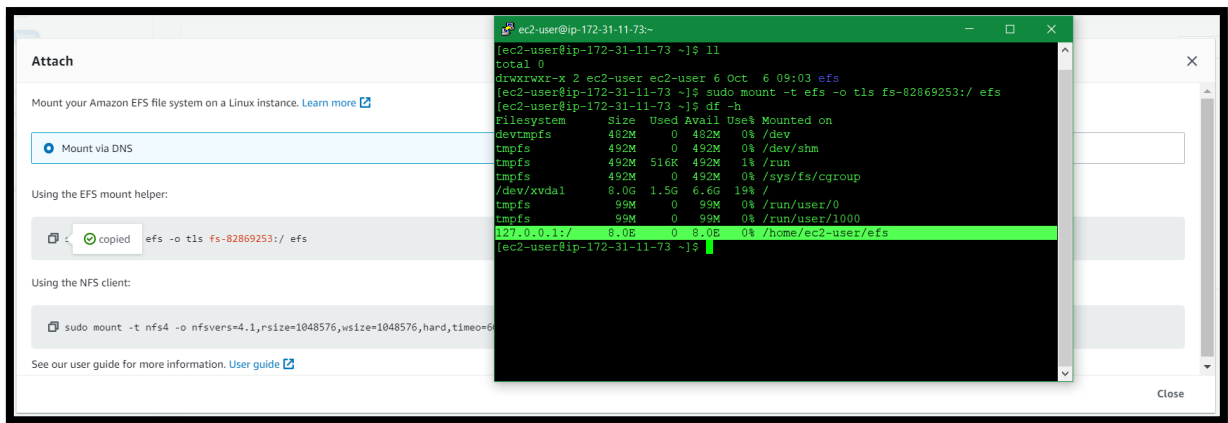
Transaction Summary
=====
Install 1 Package (+1 Dependent package)

Total download size: 195 k
Installed size: 479 k
Downloading packages:
(1/2): amazon-efs-utils-1.31.2-1.amzn2.noarch.rpm | 46 kB  00:00
(2/2): stunnel-4.56-6.amzn2.0.3.x86_64.rpm      | 149 kB  00:00
-----
Total                                     1.5 MB/s | 195 kB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : stunnel-4.56-6.amzn2.0.3.x86_64                1/2
  Installing : amazon-efs-utils-1.31.2-1.amzn2.noarch         2/2
  Verifying   : stunnel-4.56-6.amzn2.0.3.x86_64              1/2
  Verifying   : amazon-efs-utils-1.31.2-1.amzn2.noarch        2/2

Installed:
  amazon-efs-utils.noarch 0:1.31.2-1.amzn2

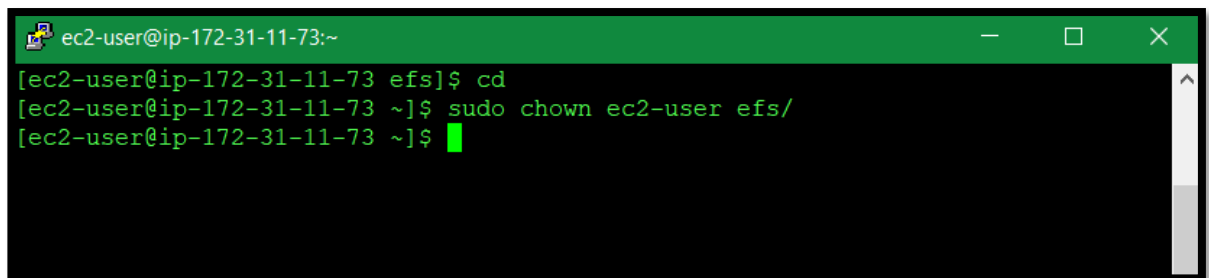
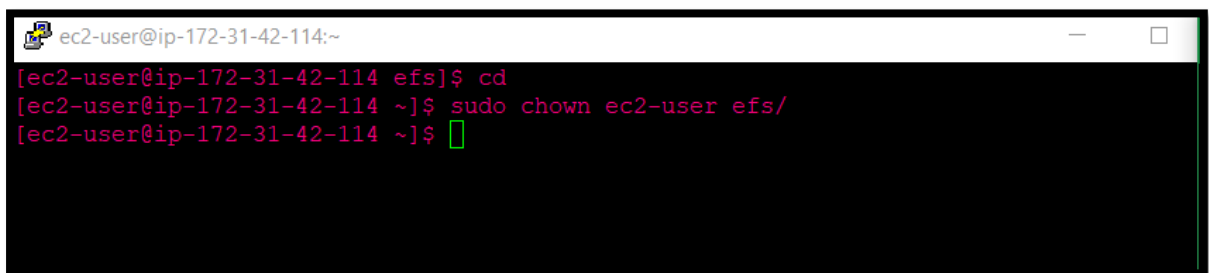
Dependency Installed:
  stunnel.x86_64 0:4.56-6.amzn2.0.3

Complete!
ec2-user@ip-172-31-11-73:~$
```



- **Step13)** We need to change the ownership of efs folder in both the ec2 machines so that our ec2-user could access it. 1<sup>st</sup> exit the efs folder using `cd` which will take us to root directory and then use command:

`sudo chown ec2-user efs/`



- **Step14)**

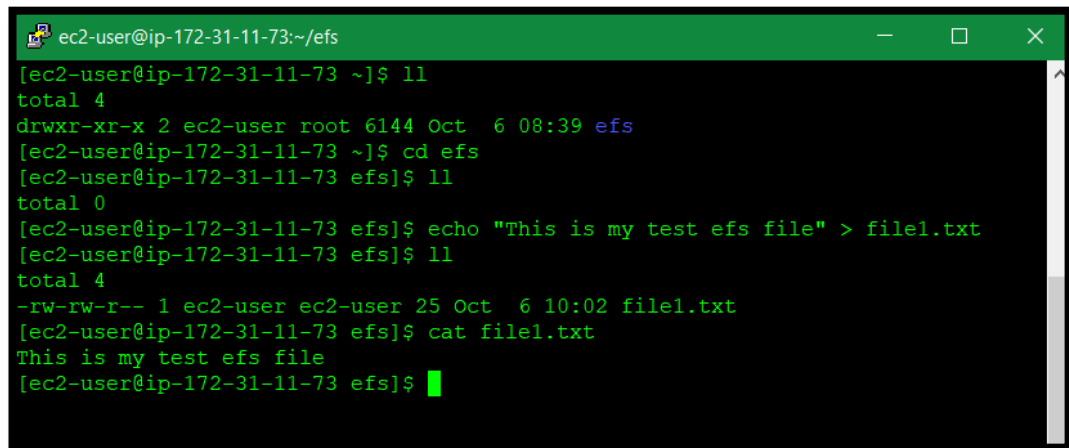
- **Brief:**

Now anything we create in EFS folder using one machine could easily be accessed by another machine as efs is a shared folder between them.

- **Step14.1)** Now we'll check whether our efs folder is shared between the two machines in different AZ's or not. For this we'll enter in the efs folder through

machine-2 and create text file and then we will check whether Machine-1 have access to this text file created by machine or not. If accessible, then we'll make some updates in the same text file using machine-2 and reconfirm the changes from machine-1.

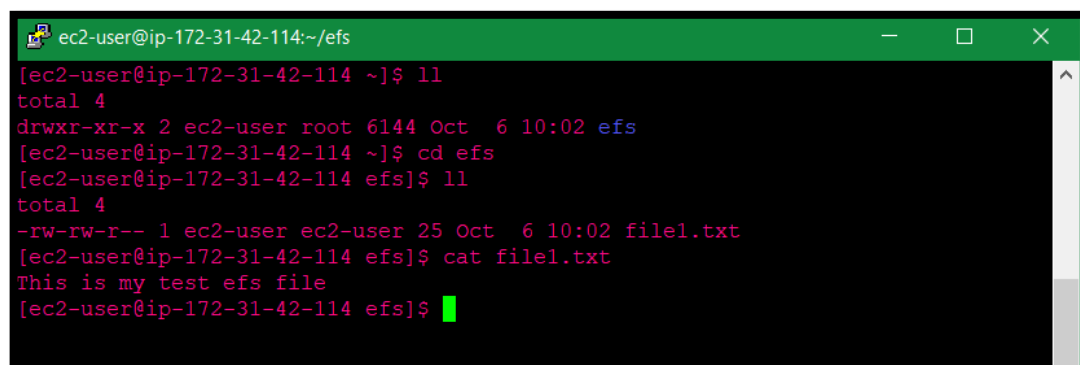
- **Machine-2:**



```
ec2-user@ip-172-31-11-73:~/efs
[ec2-user@ip-172-31-11-73 ~]$ ll
total 4
drwxr-xr-x 2 ec2-user root 6144 Oct  6 08:39 efs
[ec2-user@ip-172-31-11-73 ~]$ cd efs
[ec2-user@ip-172-31-11-73 efs]$ ll
total 0
[ec2-user@ip-172-31-11-73 efs]$ echo "This is my test efs file" > file1.txt
[ec2-user@ip-172-31-11-73 efs]$ ll
total 4
-rw-rw-r-- 1 ec2-user ec2-user 25 Oct  6 10:02 file1.txt
[ec2-user@ip-172-31-11-73 efs]$ cat file1.txt
This is my test efs file
[ec2-user@ip-172-31-11-73 efs]$
```

**Note:** In the Above Image we created a file1.txt from Machine-2 inside efs folder.

- **Machine-1:**



```
ec2-user@ip-172-31-42-114:~/efs
[ec2-user@ip-172-31-42-114 ~]$ ll
total 4
drwxr-xr-x 2 ec2-user root 6144 Oct  6 10:02 efs
[ec2-user@ip-172-31-42-114 ~]$ cd efs
[ec2-user@ip-172-31-42-114 efs]$ ll
total 4
-rw-rw-r-- 1 ec2-user ec2-user 25 Oct  6 10:02 file1.txt
[ec2-user@ip-172-31-42-114 efs]$ cat file1.txt
This is my test efs file
[ec2-user@ip-172-31-42-114 efs]$
```

- ✓ **Note:** In the Above Image we found the same file1.txt which we created in Machine-2 inside efs folder.
- ✓ Now we'll make some updates in the same text file using machine-2 and reconfirm the changes from machine-1.



