**EFS(ELASTIC FILE SYSTEM)**

**Amazon EFS (Elastic File System)** is a fully managed cloud-based file storage service that allows multiple Amazon EC2 instances to **share the same data** at the same time. It automatically scales to handle growing data and is accessible from multiple servers, making it ideal for applications that need shared access to files.

**Examples:**

1. **EFS is Like a Shared Kitchen**

Imagine you live in a **big house with several roommates**.  
Each roommate (like a server in AWS) has their own bedroom, but…

You all use **one shared kitchen** to cook food, store groceries, and wash dishes.

**How is this like EFS?**

| **Real Life (Shared Kitchen)** | **AWS (EFS)** |
| --- | --- |
| Everyone can use the kitchen at the same time. | Multiple servers can use EFS at once. |
| The kitchen has space for all the food you need. | EFS automatically grows as you add more files. |
| If one person leaves, the kitchen still works.  No one needs their own  Fridge or stove. | If one server stops, EFS keeps running.  No needs to copy file to each server. |
|  |  |

**Lab Steps**

Task 1: Sign in to AWS Management Console

1. Click on the Open Console button, and you will get redirected to AWS Console in a new browser tab.

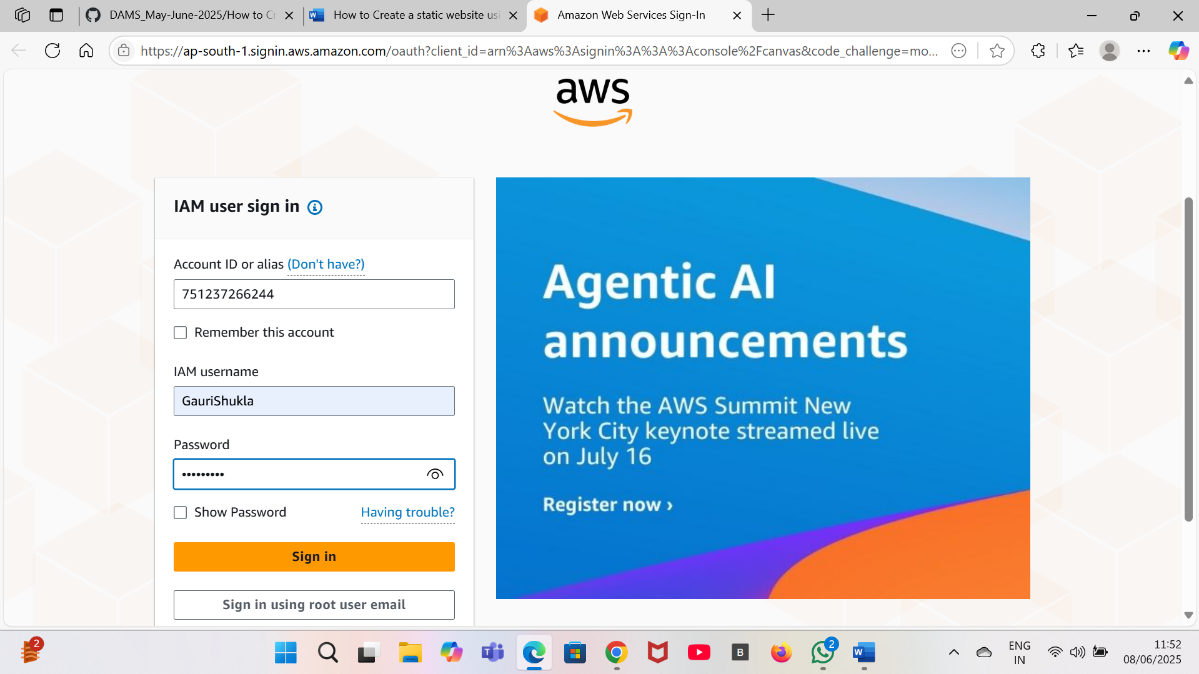
2. On the AWS sign-in page,

· Leave the Account ID as default

·Now enter your username and password.

3. click on sign-in.

4.After signing in select US East (N. Virginia) us-east- as AWS region.



Task 2: Launch two EC2 Instances

1. Click on the search bar and enter EC2 and then click on EC2.

2. Click on Instances on the right side and then click on Launch instances.

3. Enter details:

**Region**:Select United States (N. Virginia).

**Number of Instances** : Enter 2 on the right side.

**Instance Name** : Enter your instance name.

**Amazon Machine Image (AMI):** Search Amazon Linux 2 AMI in search box and select.

4. **Instance Type** : Select t2.micro

5. **Key pair**: Select Create a new key pair

**Key pair name:** Enter key pair name.

· **Key pair type**: RSA

· **Private key file format**: .pem

. If using **putty** then select **.ppk** in private key file format.

6. Select **Create key pair**.

7. Network Settings Click on Edit:

· **Auto-assign public IP**: Enable

· Select **Create new Security group**

· **Security Group Name**: Enter EFS Security Group

· **To add SSH**:

· **Choose Type**: SSH

· **Source**: Anywhere

· Click on **Add security group rule**

· **Choose Type**: NFS

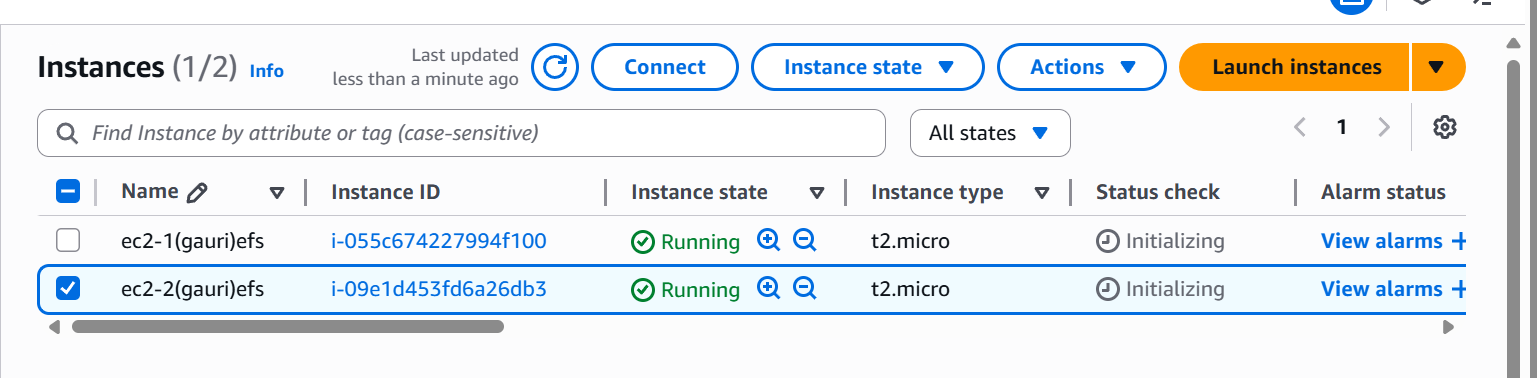
· **Source**: Anywhere

8. Keep rest thing default and Click on Launch Instance.

9. Select View all Instances to View Instance you created.

10. Click on each instance and enter a names as ec2-1 and ec2-2.

11. Copy **Public IPv4 Addresses** of the EC2 instances and save them for later.



Task 3: Creating an Elastic File System

1. Click on search bar. Enter EFS and click on EfS.

2. Click on Create file system

3. Click on Customize button.

4. Enter the details :

**Name** : Enter the file name .( I have given gauriefs as file name )

Select **default VPC**.

In **File System type** select **Regional**.

5. Uncheck the option of Enable automated backups

6. Leave everything as default and click on the Next button present below.

7. Network Access:

· VPC

· An Amazon EFS file system is accessed by EC2 instances running inside one of your VPCs.

· Choose the same VPC you selected while launching the EC2 instance (leave as default).

**Mount Targets**

· Instances connect to a file system by using a network interface called a mount target. Each mount target has an IP address, which we assign automatically or you can specify.

· Select all the **Availability Zones (AZ’s)**

So that the EC2 instances across your VPC can access the file system.

· In the Security Groups, select EFS Security Group instead of the default .

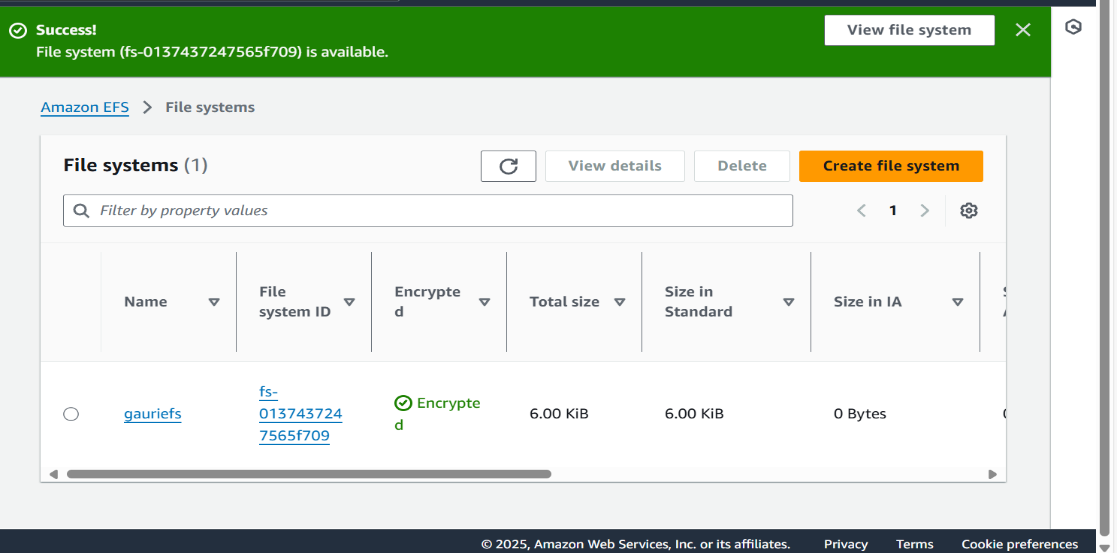
· Make sure you remove the default security group and select the EFS Security Group.

· Click on Next button

8. **File system policy** Click on Next button

9. **Review and Create**: Review the configuration below before proceeding to create your file system then Click on **Create.**

10. EFS file system is created.



Task 4: Mount the File System to ec2-1 Instance

1. Select the first Instance which you’ve created and copy the IPv4 Public IP.

2. Under connect choose the EC2 instance connect tab .Click connect. A browser based terminal will open.

3. Switch to root user using command:

**sudo -s**

4. Run the updates using command:

**yum -y update**

5. Install the NFS client as amazon-efs-utils.

**yum install -y amazon-efs-utils**

6. Create a directory by the name **efs**

**mkdir efs**

7. We have to mount the file system in this directory.

8. To do so, go to the AWS console and click on the file system which you have created earlier. On the top-right corner, click on Attach.

· Copy the command the EFS mount helper.

**sudo mount -t efs -o tls fs-2ad0a9a8:/ efs**

· Note: fs-2ad0a9a8 is file system id in my case, it could be different in your case, make sure to replace it.

9. To display information for all currently-mounted file systems, we'll use the command bellow:

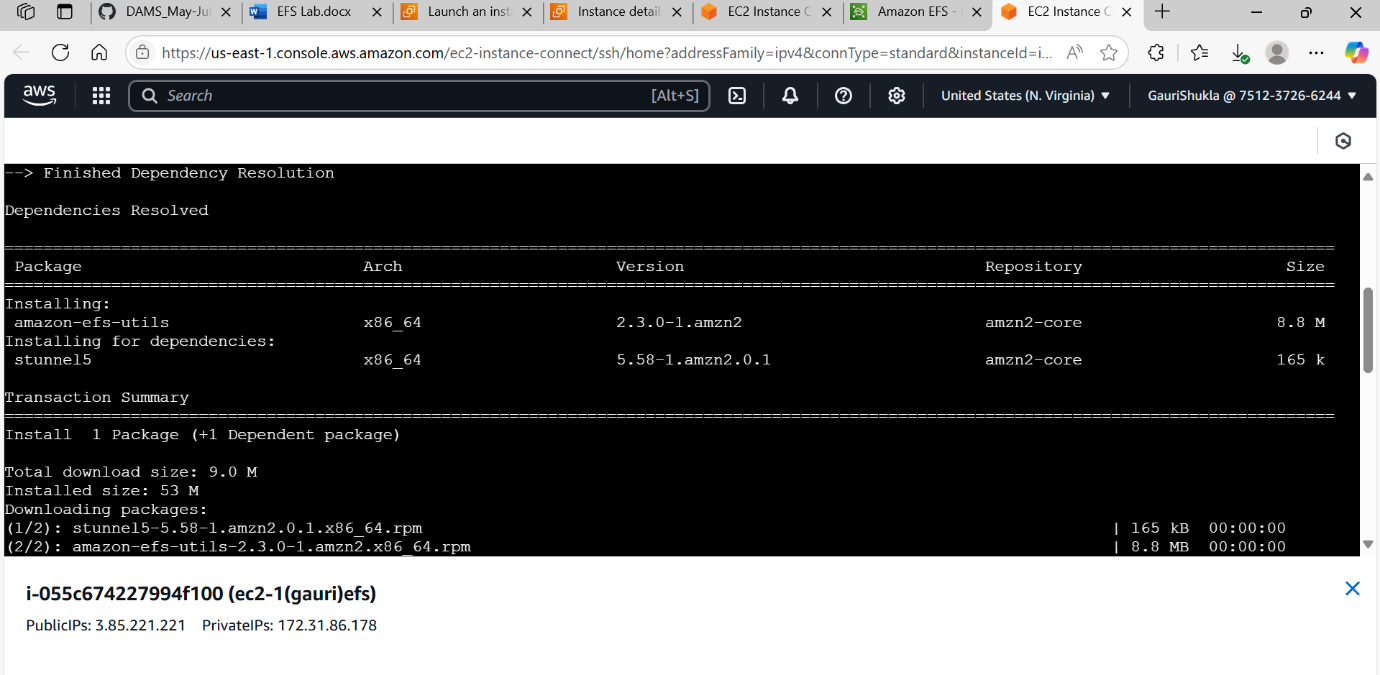
**df -h**

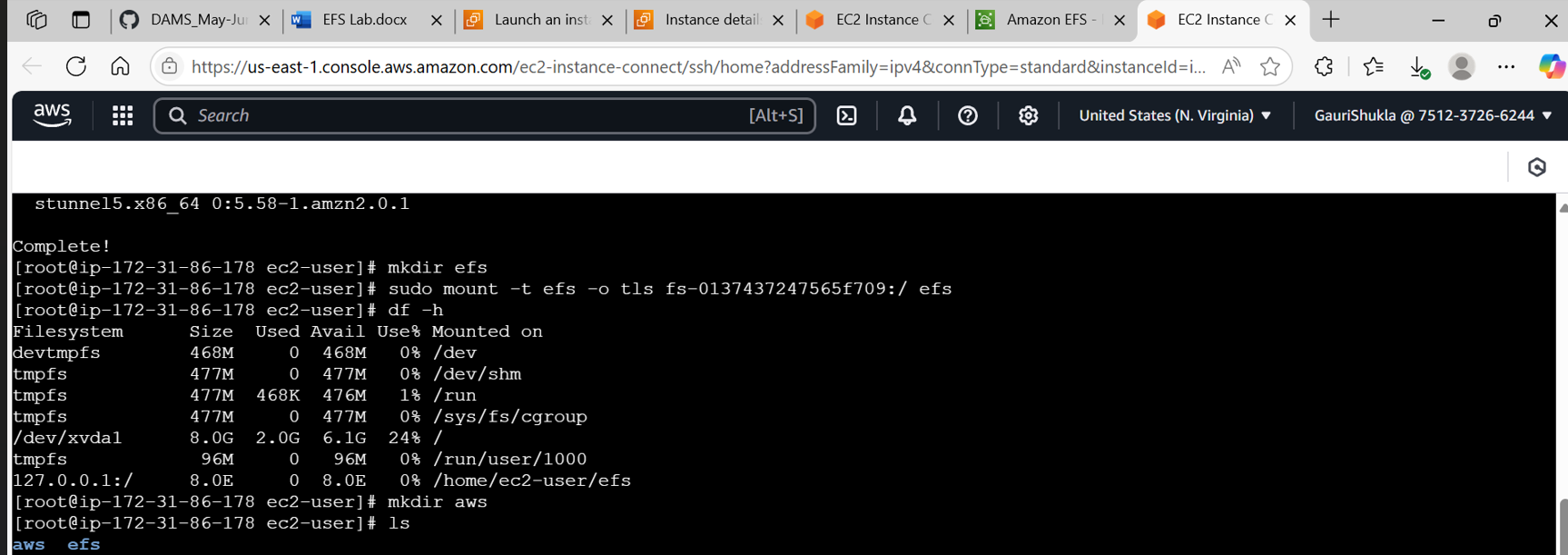
10. Create a directory in our current location:

**mkdir aws**

11.Now use ls command to see list of the directories

**ls**





Task 5: Mount the File System to ec2-2 Instance

1. Select the MyEC2-2 Instance and copy the IPv4 Public IP.

2. Under connect choose the EC2 instance connect tab .Click connect. A browser based terminal will open.

3. Switch to root user

**sudo -s**

4. Run the updates using the command:

**yum -y update**

5. Install the NFS client as amazon-efs-utils.

**yum -y install amazon-efs-utils**

6. Create a directory with the name efs

**mkdir efs**

7. We have to mount the file system in this directory.

8. To do so, navigate to the AWS console and click on the created file system. On the top-right corner, click on Attach

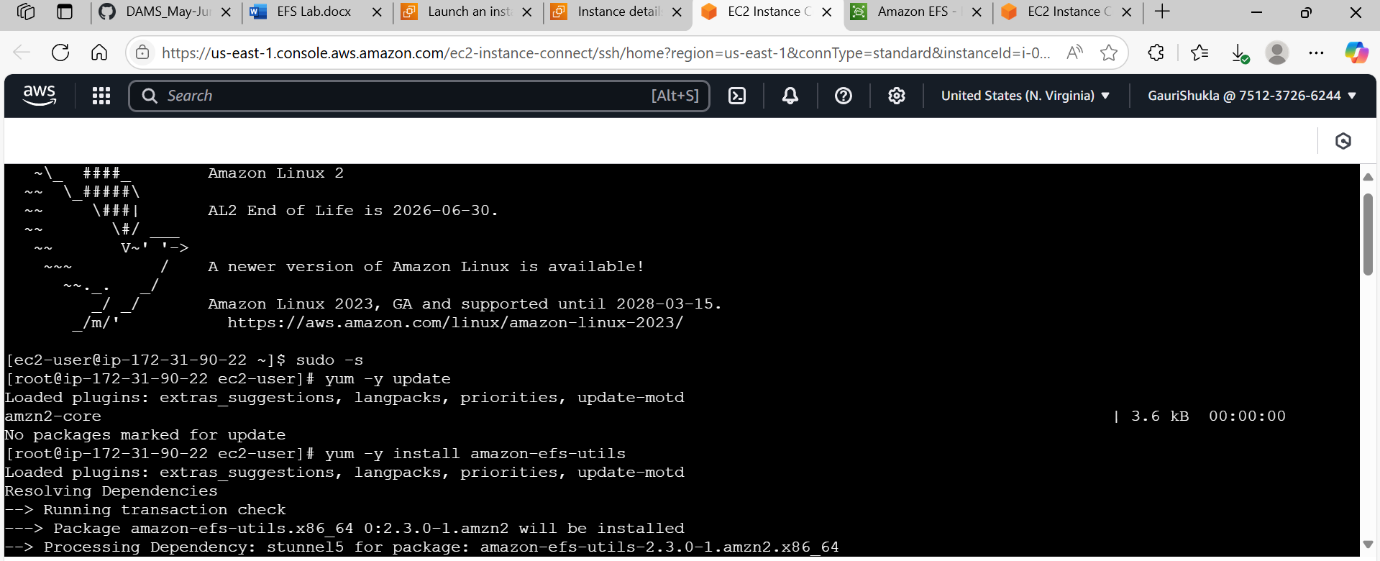
· Copy the command of Using the EFS mount helper.

**sudo mount -t efs -o tls fs-2ad0a9a8:/ efs**

· Note1: fs-2ad0a9a8 is file system id in my case, it could be different in your case, make sure to replace it.

· To display information for all currently mounted file systems, we'll use the command:

**df -h**





Task 6: Testing the File System

1. SSH into both instances in a side-by-side view on your machine, if possible.

2. Switch to root user

**sudo -s**

3. Navigate to the efs directory in both the servers using the command

**cd efs**

4. Create a file in any one server.

**touch hello.txt**

5. Check the file using the command

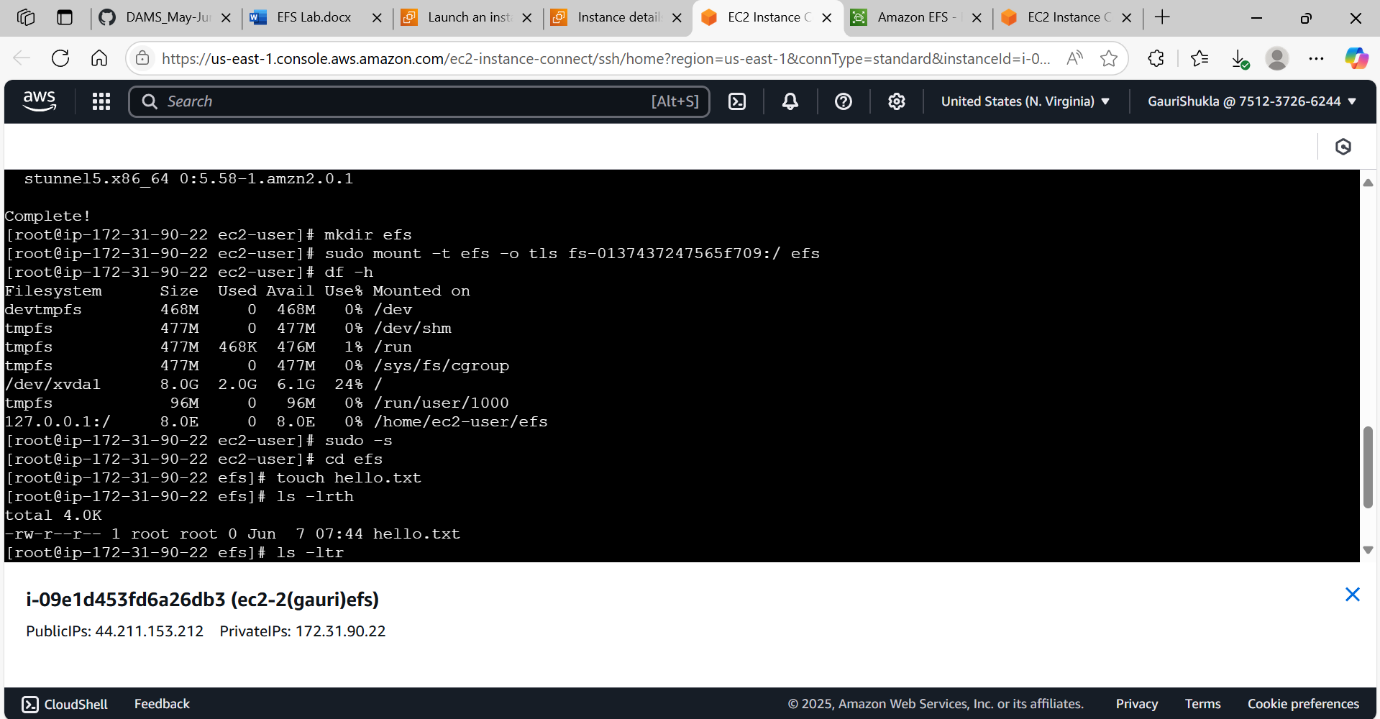
**ls -ltr**

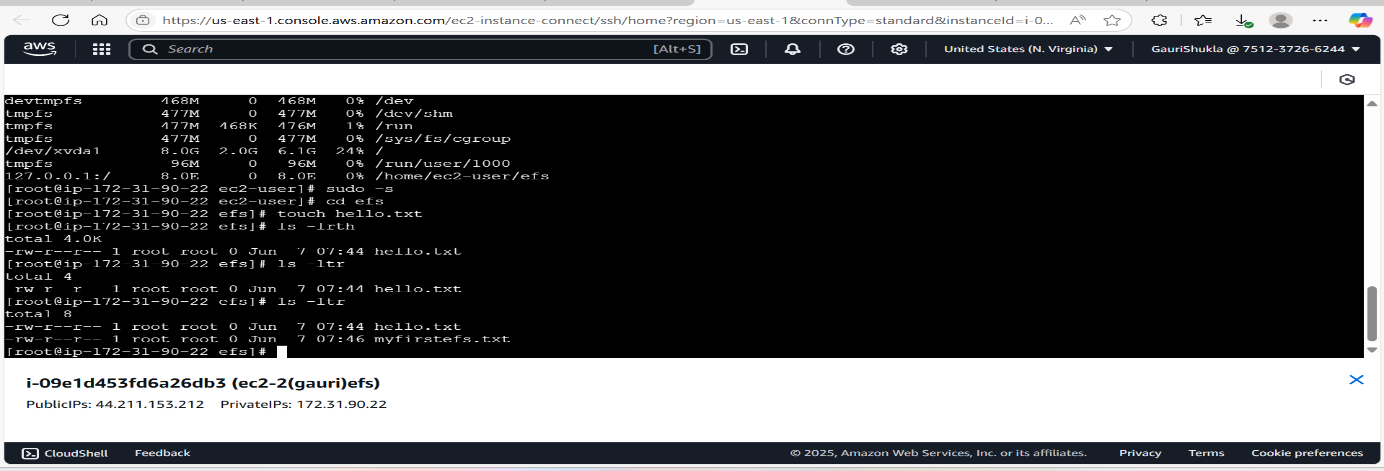
6. Now go to the other server and give the command

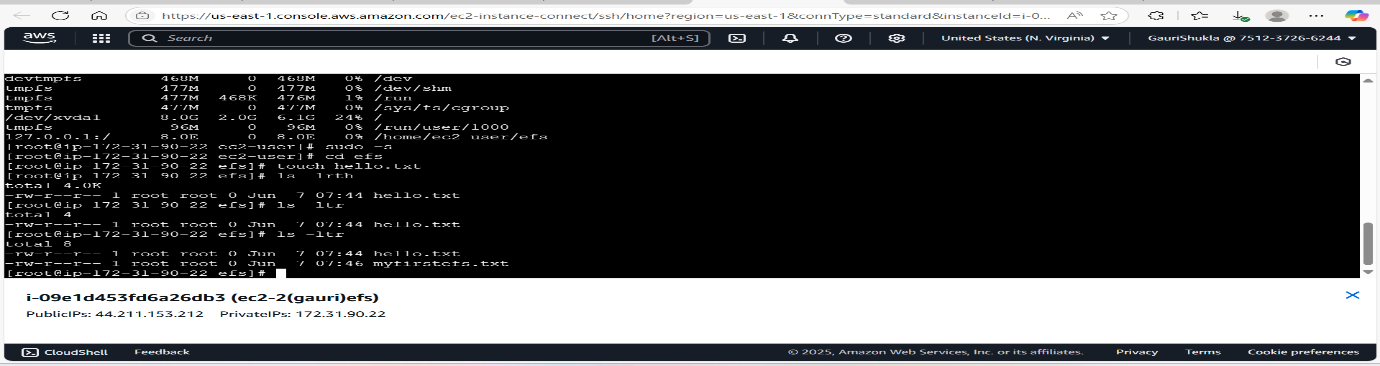
**cd efs**

7. You can see the file created on this server as well. This proves that our EFS is working.

8. You can try creating files (touch command) or directories (mkdir command) on other servers to continue to grow the EFS implementation.







***EFS uses the Network File System (NFS) protocol, which enables EC2 instances to mount EFS as a network file system, providing a common data source for applications running on multiple instances.***

**INTERVIEW QUESTIONS RELATED TO EFS(ELASTIC FILE SYSTEM)**

1. **What is Amazon EFS?**  
   a) A scalable, managed file storage service for EC2 instances  
   b) A block storage system for databases  
   c) A local file system for AWS Lambda  
   d) A caching service for DynamoDB  
   **Answer:** a) A scalable, managed file storage service for EC2 instances
2. **Which protocol does EFS use?**  
   a) SMB  
   b) NFS  
   c) iSCSI  
   d) S3 API  
   **Answer:** b) NFS
3. **Which AWS service is EFS most similar to?**  
   a) Amazon S3  
   b) Amazon EBS  
   c) Amazon RDS  
   d) AWS Lambda  
   **Answer:** b) Amazon EBS
4. **What is the primary advantage of EFS over EBS?**  
   a) EFS can be shared across multiple instances  
   b) EFS is cheaper than EBS  
   c) EFS does not require IAM permissions  
   d) EFS offers unlimited object storage  
   **Answer:** a) EFS can be shared across multiple instances
5. **Which file system standard does EFS use?**  
   a) NFSv3 and NFSv4.1  
   b) NTFS  
   c) FAT32  
   d) XFS  
   **Answer:** a) NFSv3 and NFSv4.1
6. **Which AWS region feature affects EFS performance?**  
   a) Availability Zones  
   b) Edge locations  
   c) AWS Outposts  
   d) Glacier storage class  
   **Answer:** a) Availability Zones
7. **Which EC2 instance attribute affects EFS performance?**  
   a) CPU type  
   b) Storage size  
   c) Network bandwidth  
   d) Root volume type  
   **Answer:** c) Network bandwidth
8. **Which feature allows EFS to serve multiple instances concurrently?**  
   a) Multi-AZ replication  
   b) NFS protocol support  
   c) CloudFront integration  
   d) IAM role configuration  
   **Answer:** b) NFS protocol support
9. **Which AWS feature restricts access to EFS?**  
   a) Security Groups  
   b) IAM Policies  
   c) VPC Peering  
   d) AWS Auto Scaling  
   **Answer:** b) IAM Policies
10. **Which factor has the greatest impact on EFS pricing?**  
    a) Number of connected EC2 instances  
    b) Amount of data stored  
    c) Data transfer rate  
    d) Number of file operations  
    **Answer:** b) Amount of data stored
11. **Which command is used to mount an EFS file system on Linux?**  
    a) mount -t nfs  
    b) mount -t efs  
    c) sudo efs mount  
    d) aws efs attach  
    **Answer:** a) mount -t nfs
12. **If an EC2 instance cannot access an EFS file system, what should you check first?**  
    a) Security Group and NFS port access (port 2049)  
    b) The size of the root volume  
    c) The number of EBS volumes attached  
    d) The EC2 instance type  
    **Answer:** a) Security Group and NFS port access (port 2049)
13. **What happens to an EFS file system if all EC2 instances accessing it are terminated?**  
    a) Data remains stored and accessible  
    b) Data is automatically deleted  
    c) The EFS file system is archived  
    d) It switches to read-only mode  
    **Answer:** a) Data remains stored and accessible