

LAB: Create an Elastic File System and connect it to EC2 Instances

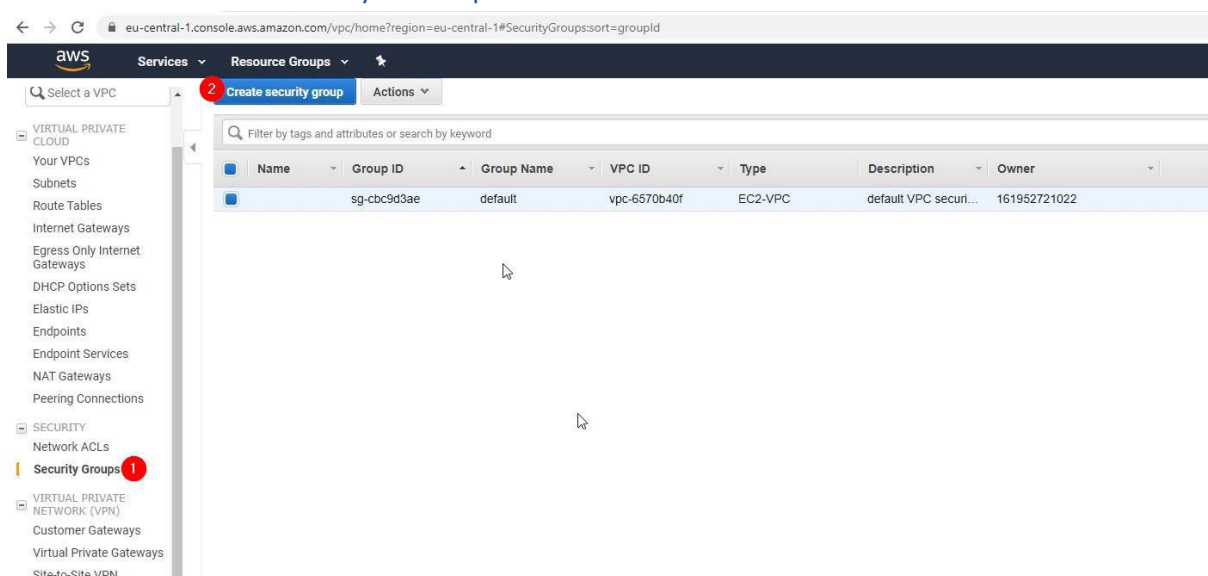
You need:

- An AWS Account

Duration of the Lab: 30 Minutes.

Difficulty: medium

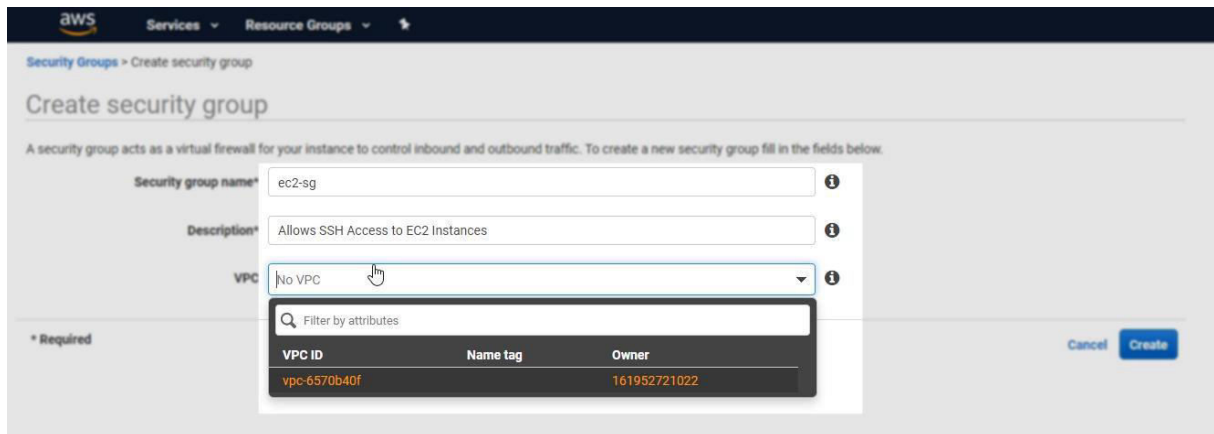
Create a new Security Group for the EFS and the EC2 instances



First the EFS:

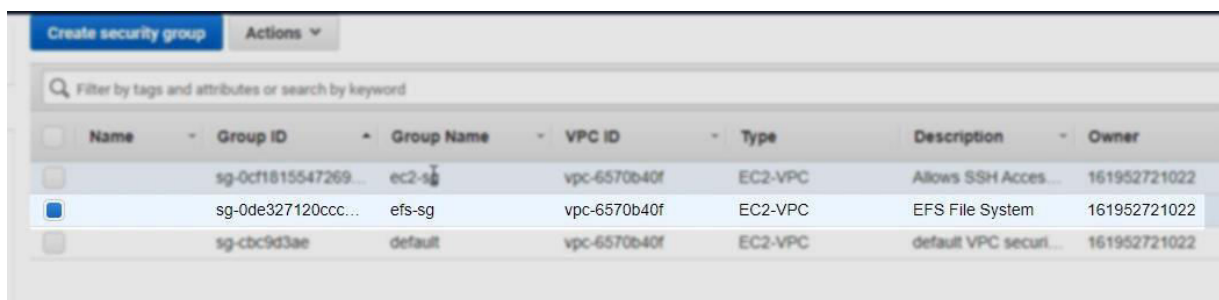
The screenshot shows the 'Create security group' form in the AWS Management Console. The form has three main input fields: 'Security group name*' with the value 'efs-sg', 'Description*' with the value 'EFS File System', and 'VPC' with a dropdown menu showing 'vpc-6570b40f'. There are information icons (i) next to each field. At the bottom right, there are 'Cancel' and 'Create' buttons. A note at the bottom left states '* Required'.

Another one for the EC2 Instances:



Configure EFS Security Group

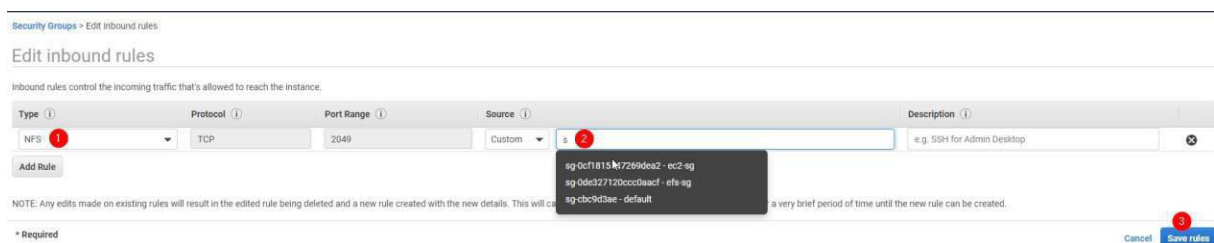
Edit the Inbound Rules from the EFS Security Rule:



Security Group: sg-0de327120ccc0aacf



Add the EC2 security group as NFS:



Configure EC2 Security Group

Edit the EC2 Security Group to allow inbound SSH access from anywhere:

Filter by tags and attributes or search by keyword

| <input type="checkbox"/> | Name | Group ID | Group Name | VPC ID | Type | Description | Owner |
|-------------------------------------|------|---------------------|------------|--------------|---------|-----------------------|--------------|
| <input checked="" type="checkbox"/> | | sg-0cf1815547269... | ec2-sg | vpc-6570b40f | EC2-VPC | Allows SSH Acces... | 161952721022 |
| <input type="checkbox"/> | | sg-0de327120ccc... | efs-sg | vpc-6570b40f | EC2-VPC | EFS File System | 161952721022 |
| <input type="checkbox"/> | | sg-cbc9d3ae | default | vpc-6570b40f | EC2-VPC | default VPC securi... | 161952721022 |

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

| Type | Protocol | Port Range | Source | Description |
|------|----------|------------|----------|----------------------------|
| SSH | TCP | 22 | Anywhere | e.g. SSH for Admin Desktop |

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

* Required

Cancel Save rules

Create an EFS File System

Open the EFS Dashboard:

Services
Resource Groups

History

VPC

Console Home

EC2

EFS

Elastic Container Service

S3

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Compute

EC2

Lightsail

Lambda

Batch

Elastic Beanstalk

Serverless Application Repository

AWS Outposts

EC2 Image Builder

Storage

S3

EFS

FSx

S3 Glacier

Storage Gateway

AWS Backup

Blockchain

Amazon Managed Blockchain

Satellite

Ground Station

Quantum Technologies

Amazon Braket

Management & Governance

AWS Organizations

CloudWatch

AWS Auto Scaling

CloudFormation

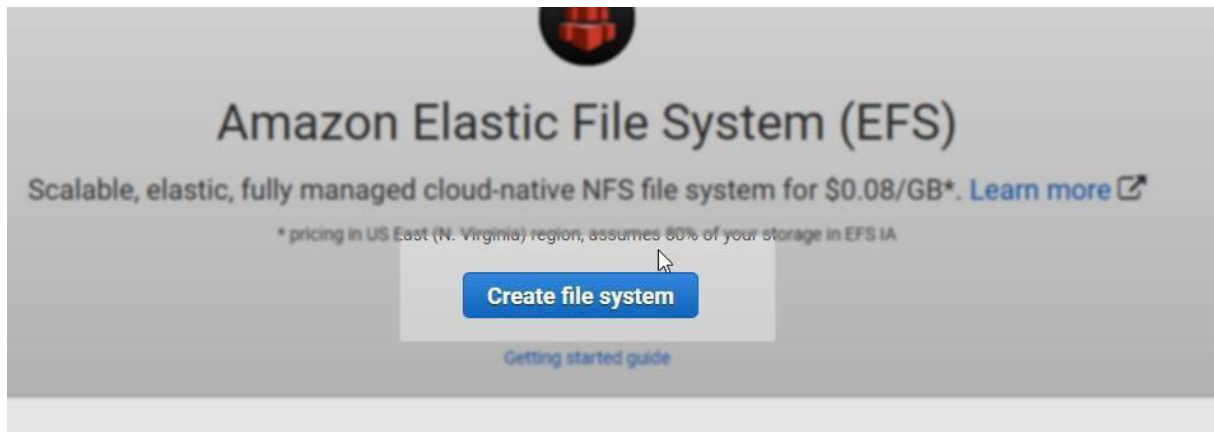
CloudTrail

Config

OpsWorks

Service Catalog

Let's use the creation wizard:



Place the EFS in all availability zones and assign the EFS Security Group to all mount points:

Configure network access

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

VPC:

Create mount targets

Instances connect to a file system by using mount targets you create. We recommend creating a mount target in each of your VPC's Availability Zones so that EC2 instances across your VPC can access the file system.

| | Availability Zone | Subnet | IP address | Security groups |
|---|---|--|------------|--|
| 1 | <input checked="" type="checkbox"/> eu-central-1a | <input type="text" value="subnet-cfd47ba5 (default)"/> | Automatic | <input type="text" value="sg-0de327120ccc0aacf - efs-sg"/> |
| 2 | <input checked="" type="checkbox"/> eu-central-1b | <input type="text" value="subnet-d5f6eca8 (default)"/> | Automatic | <input type="text" value=""/> |
| 3 | <input checked="" type="checkbox"/> eu-central-1c | <input type="text" value="subnet-bc21c8f0 (default)"/> | Automatic | <input type="text" value=""/> |

sg-0cf1815547269dea2 - ec2-sg

sg-0de327120ccc0aacf - efs-sg

sg-cbc9d3ae - default

Cancel **Next Step**

Leave all settings at their default value for file system settings:

Enable lifecycle management NEW!

Automatically save up to 92% on your EFS bill as your access patterns change by enabling **Lifecycle Management** for your file system. Based on the policy you choose, any files in your file system that are not accessed for a period of time will automatically move to the EFS Infrequent Access (EFS IA) storage class. EFS IA provides price/performance that's cost-optimized for files not accessed every day. [Learn more](#)

Lifecycle policy None

Choose throughput mode

We recommend **Bursting** throughput mode for most file systems. Use **Provisioned** throughput mode for applications that require more throughput than allowed by **Bursting** throughput. [Learn more](#)

- ☒ Bursting
- ☐ Provisioned

Choose performance mode

We recommend **General Purpose** performance mode for most file systems. **Max I/O** performance mode is optimized for applications where tens, hundreds, or thousands of EC2 instances are accessing the file system – it scales to higher levels of aggregate throughput and operations per second with a tradeoff of slightly higher latencies for file operations.

- ☒ General Purpose
- ☐ Max I/O

Enable encryption

If you enable encryption for your file system, all data on your file system will be encrypted at rest. You can select a KMS key from your account to protect your file system, or you can provide the ARN of a key from a different account. Encryption of data at rest can only be enabled during file system creation. Encryption of data in transit is configured when mounting your file system. [Learn more](#)

☐ Enable encryption of data at rest

[Cancel](#) [Previous](#) [Next Step](#)

Lave all settings at their default value for client access:

Configure client access

File system policy

A file system policy is an IAM resource policy that applies to all NFS clients connecting to this file system. You can use the check boxes below to create a simple file system policy. You can also use the JSON editor to create a more advanced policy, such as one that grants permissions to different IAM roles or a different AWS account. [Learn more](#)

Policy settings { } JSON

Select a combination of policy statements and set your policy.

- ☐ Disable root access by default*
- ☐ Enforce read-only access by default*
- ☐ Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

[Set policy](#)

Access points

You can create access points to provide applications access to your file system. Optionally, you can configure the POSIX identity and root directory for all connections to this access point. If you specify the owner for root directory, EFS will automatically create it with the ownership and permissions that you specify once a client connects to the access point. Once you create your file system, you can update its policy to apply to access points. [Learn more](#)

| | Name | Posix User | Directory | Owner |
|---|------|------------|-----------|-------|
| Add access points for your file system. | | | | |

[+ Add access point](#)

[Cancel](#) [Previous](#) [Next Step](#)

Review and Create the EFS:

Review and create

Review the configuration below before proceeding to create your file system.

File system access

| VPC | Availability Zone | Subnet | IP address | Security groups |
|------------------------|-------------------|---------------------------|------------|-------------------------------|
| vpc-6570b40f (default) | eu-central-1a | subnet-cfd47ba5 (default) | Automatic | sg-0de327120ccc0aacf - efs-sg |
| | eu-central-1b | subnet-d5f6eca8 (default) | Automatic | sg-0de327120ccc0aacf - efs-sg |
| | eu-central-1c | subnet-bc21c8f0 (default) | Automatic | sg-0de327120ccc0aacf - efs-sg |

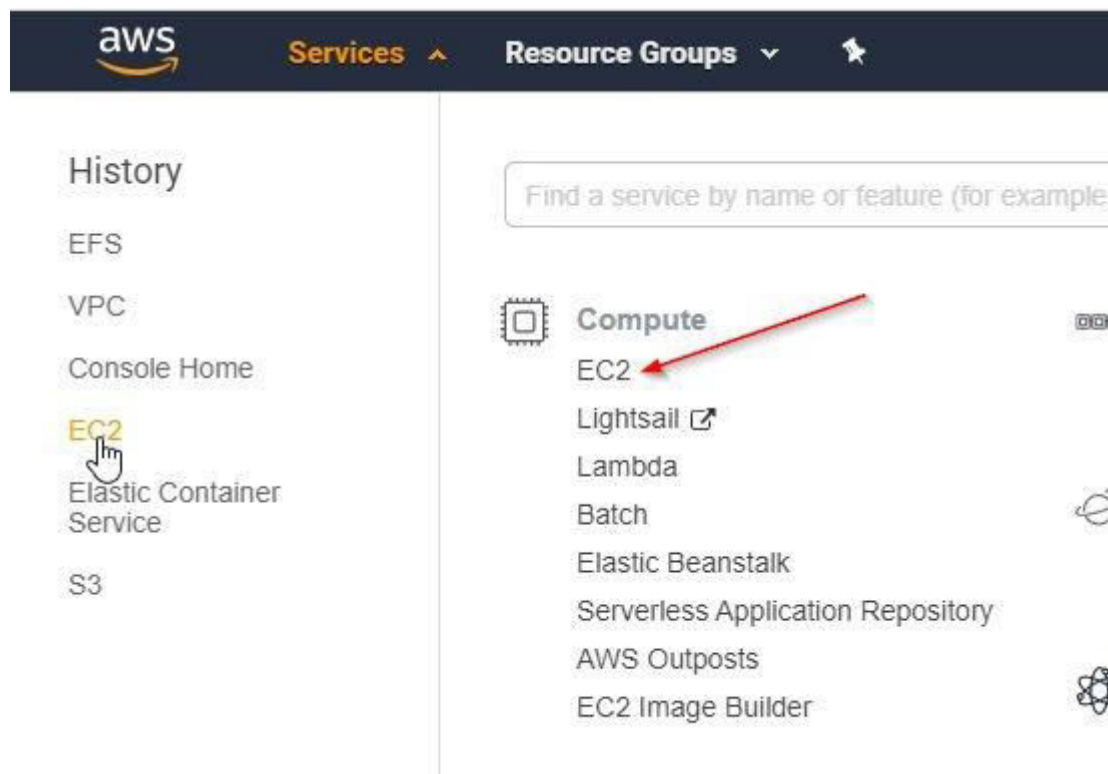
Optional settings

| | |
|-------------------------|-----------------|
| Tags | No tags added |
| Performance mode | General Purpose |
| Throughput mode | Bursting |
| Encrypted | No |
| Lifecycle policy | None |
| Number of access points | 0 |
| File system policy | None |

Cancel Previous **Create File System**

EC2 Creation

Head over to the EC2 Dashboard:



Launch an Instance

Launch a new Instance.

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Launch instance ▼

Note: Your instances will launch in the Europe (Frankfurt) Region

Select the Amazon Linux 2 AMI and the t2.micro instance type.

Start 2 Instances right away:

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.

| | | |
|-----------------------|---|--------------------------------|
| Number of instances | 2 | Launch into Auto Scaling Group |
| Purchasing option | <input type="checkbox"/> Request Spot instances | |
| Network | vpc-6570b40f (default) | Create new VPC |
| Subnet | No preference (default subnet in any Availability Zone) | Create new subnet |
| Auto-assign Public IP | Use subnet setting (Enable) | |

Add a File System

File systems ⓘ

Add file system



Create new file system

Observe that a script is automatically inserted into the user-data field:

File systems ⓘ **1** fs-92fa5eca /mnt/efs/fs1 ⓘ

Add file system Create new file system

▼ Advanced Details

| | |
|-------------------------------------|--|
| Metadata accessible ⓘ | Enabled |
| Metadata version ⓘ | V1 and V2 (token optional) |
| Metadata token response hop limit ⓘ | 1 |
| User data ⓘ | <p><input checked="" type="radio"/> As text <input type="radio"/> As file <input type="checkbox"/> Input is already base64 encoded</p> <p>2 #cloud-config repo_update: true repo_upgrade: all runcmd:</p> |

Security Group

Make sure you select the EC2 Security group we created earlier:

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon

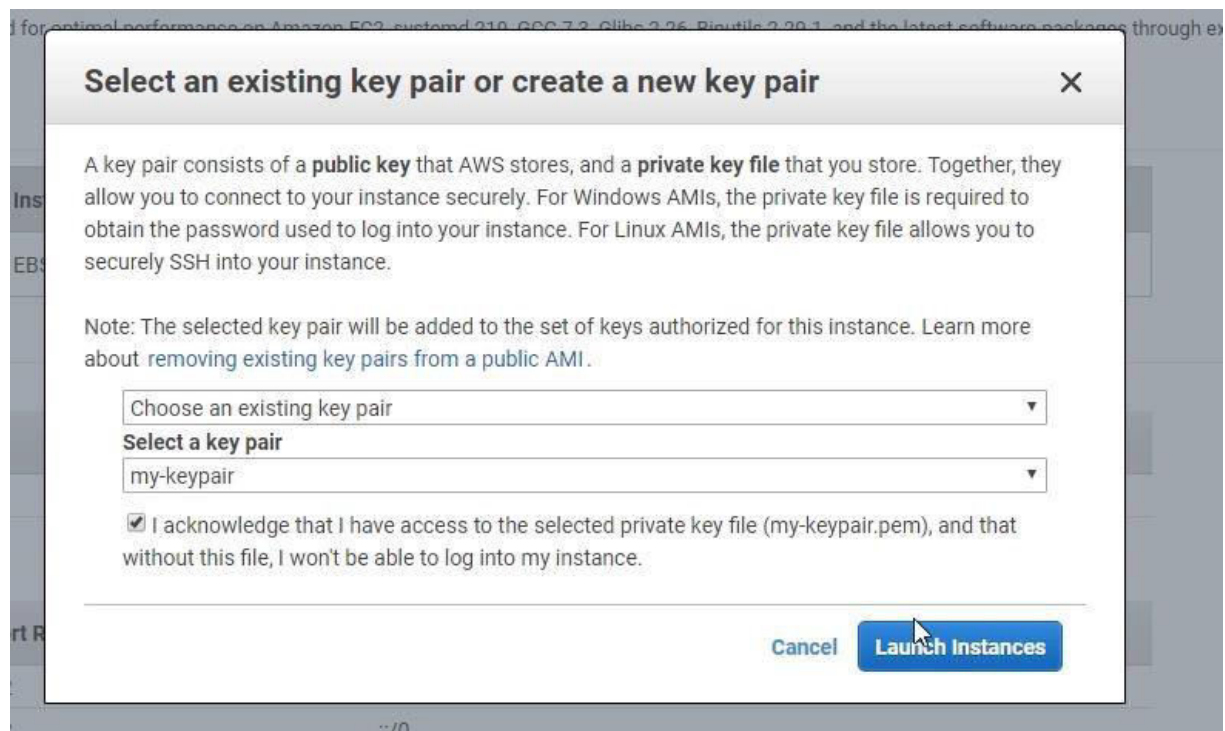
Assign a security group: ☐ Create a new security group

1 ☒ Select an **existing** security group

| Security Group ID | Name |
|---|---------|
| <input type="checkbox"/> sg-cbc9d3ae | default |
| <input checked="" type="checkbox"/> sg-0cf1815547269dea2 2 | ec2-sg |
| <input type="checkbox"/> sg-0de327120ccc0aacf | efs-sg |

Launch instances

Launch the instances and select your key-pair so you can login via ssh:



Use Elastic File System

Login to EC2 Instance 1

Login to your first instance:


```
Course 14 - Understanding Docker with AWS ECS and Fargate> ssh -i "my-keypair.pem" ec2-
user@ec2-52-59-253-174.eu-central-1.compute.amazonaws.com
The authenticity of host 'ec2-52-59-253-174.eu-central-1.compute.amazonaws.com (52.59.2
53.174)' can't be established.
ECDSA key fingerprint is SHA256:OkY9dPs3jucJHD4HnqUSrW3kBmi8s5DhatdtYhLoVRk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-59-253-174.eu-central-1.compute.amazonaws.com,52.59.
253.174' (ECDSA) to the list of known hosts.
```

```

 _|_  _|_  _|_
 _|_  _|_  _|_  Amazon Linux 2 AMI

```

```
https://aws.amazon.com/amazon-linux-2/
1 package(s) needed for security, out of 1 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-37-175 ~]$ cd /mnt/efs/fs1/
[ec2-user@ip-172-31-37-175 fs1]$ ls
[ec2-user@ip-172-31-37-175 fs1]$
```

And change to the directory /mnt/efs/fs1

```
cd /mnt/efs/fs1
```

Change the permissions

For the ec2-user to be able to write files from the EFS we need to change the permissions.

```
sudo chown ec2-user:ec2-user .
```

Then login to the second instance:

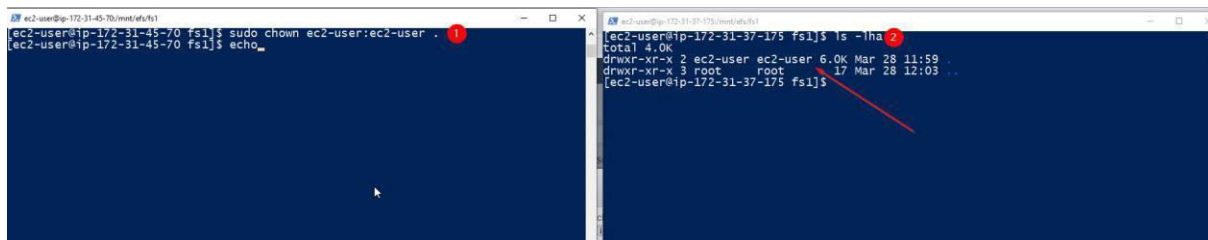
Login to EC2 Instance 2

Open a second powershell/terminal and login to instance 2.

Cd into the same directory /mnt/efs/fs1.

Observe that the directory now already has the correct permissions

```
ls -lha
```



Create a file

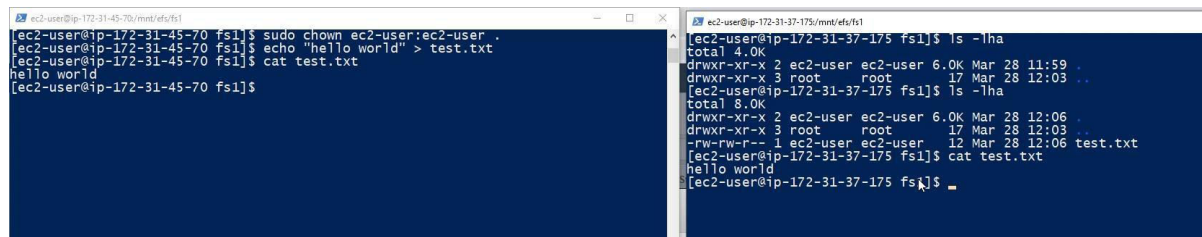
Let's create a file on instance 1 and observe the same file appearing on instance 2:

On Instance 1 run:

```
echo "hello world" > test.txt
```

On Instance 2 run:

```
cat test.txt
```



The image shows two terminal windows side-by-side. The left window is titled 'ec2-user@ip-172-31-45-70/mnt/efs/fs1' and shows the following commands and output:
[ec2-user@ip-172-31-45-70 fs1]\$ sudo chown ec2-user:ec2-user .
[ec2-user@ip-172-31-45-70 fs1]\$ echo "hello world" > test.txt
[ec2-user@ip-172-31-45-70 fs1]\$ cat test.txt
hello world
[ec2-user@ip-172-31-45-70 fs1]\$
The right window is titled 'ec2-user@ip-172-31-37-175/mnt/efs/fs1' and shows the following commands and output:
[ec2-user@ip-172-31-37-175 fs1]\$ ls -lha
total 4.0K
drwxr-xr-x 2 ec2-user ec2-user 6.0K Mar 28 11:59 .
drwxr-xr-x 3 root root 17 Mar 28 12:03 ..
[ec2-user@ip-172-31-37-175 fs1]\$ ls -lha
total 8.0K
drwxr-xr-x 2 ec2-user ec2-user 6.0K Mar 28 12:06 .
drwxr-xr-x 3 root root 17 Mar 28 12:03 ..
-rw-rw-r-- 1 ec2-user ec2-user 12 Mar 28 12:06 test.txt
[ec2-user@ip-172-31-37-175 fs1]\$ cat test.txt
hello world
[ec2-user@ip-172-31-37-175 fs1]\$

Cleanup

1. Terminate both Instances
2. Delete the EFS File System (costs might occur otherwise)
3. Delete the security groups