Get started with Amazon Elastic File System

Amazon EFS is a simple, serverless, elastic, set-and-forget file system that automatically grows and shrinks as you add and remove files with no need for management or provisioning. You can use Amazon EFS with Amazon EC2, AWS Lambda, Amazon ECS, Amazon EKS and other AWS compute instances, or with on-premises servers.

In this getting started exercise, you can learn how to quickly create an Amazon Elastic File System (Amazon EFS) file system. As part of this process, you mount your file system on two Amazon Elastic Compute Cloud (Amazon EC2) instances in your virtual private cloud (VPC). You also test the end-to-end setup.

Steps

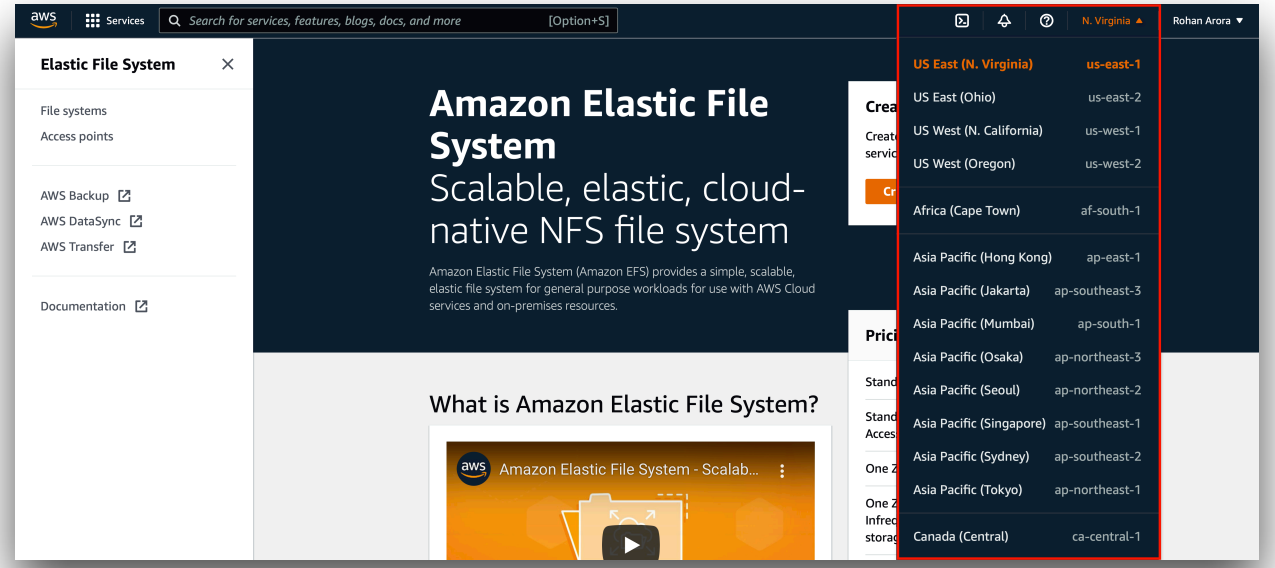
Following are the steps to be performed to complete this exercise:

- Create your Amazon EFS file system
- Launch two Amazon Linux EC2 instances
- Test the file system

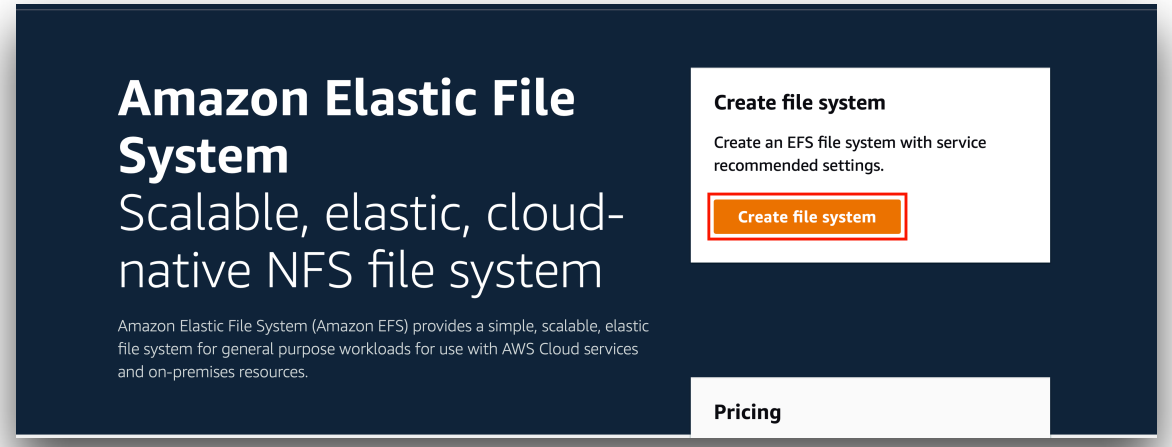
Create your Amazon EFS file system

1. Open the Amazon EFS Management Console at https://console.aws.amazon.com/efs/.

2. In the navigation bar at the top of the screen, the current AWS Region is displayed (for example, US East (N.Virginia)). Select a Region in which to deploy your resources including your EFS file system and EC2 instances.



3. Choose **Create file system** to open the **Create file system** dialog box.



4. (Optional) Enter a **Name** for your file system.

5. For **Virtual Private Cloud (VPC)**, choose your VPC, or keep it set to your default VPC.

6. For **Availability and Durability**, choose one of the following:

- **Regional** to create a file system that uses Standard storage classes. Standard storage classes store file system data and metadata redundantly across all Availability Zones within an AWS Region. **Regional** offers the highest levels of availability and durability.
- **One Zone** to create a file system that uses One Zone storage classes. One Zone storage classes store file system data and metadata redundantly within a single Availability Zone which makes it less expensive than Standard storage classes.

Because EFS One Zone storage classes store data in a single AWS Availability Zone, data stored in these storage classes may be lost in the event of a disaster or other fault that affects all copies of the data within the Availability Zone, or in the event of Availability Zone destruction resulting from disasters, such as earthquakes and floods. If you choose One Zone, choose the **Availability Zone** that you want the file system created in, or leave the default setting.

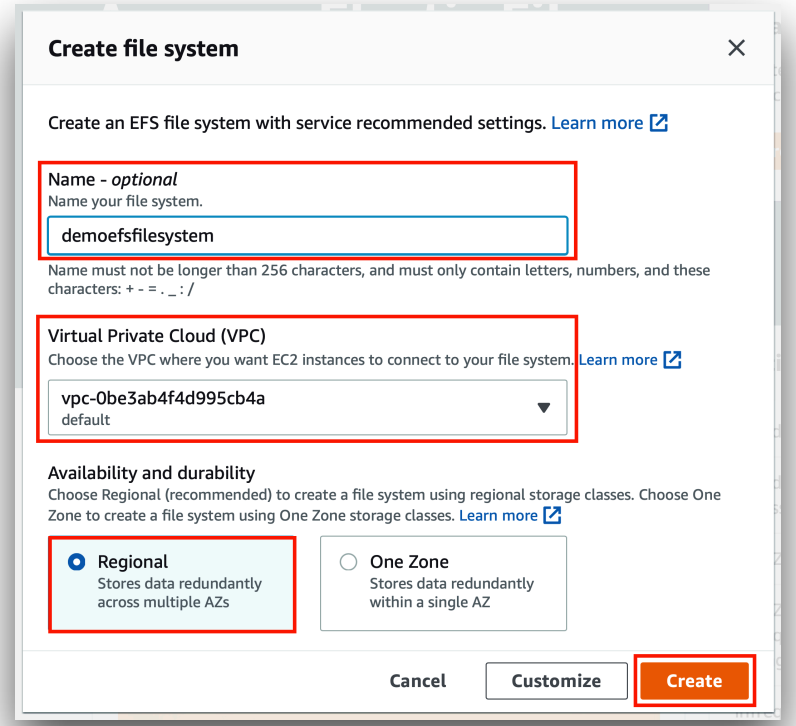
For this exercise, we will have the **Regional** option selected.

7. Choose **Create** to create a file system.

A. Enter a name for your file.

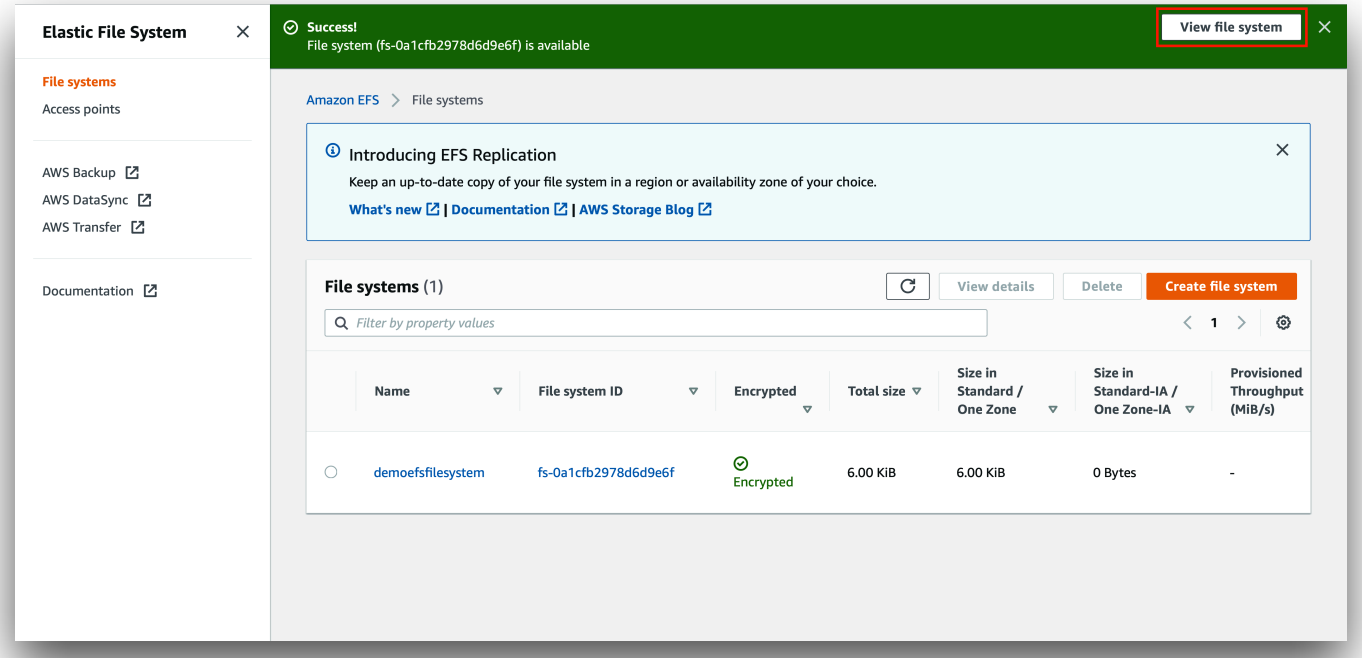
B. Keep it set to default VPC.

C. Keep it set to Regional.

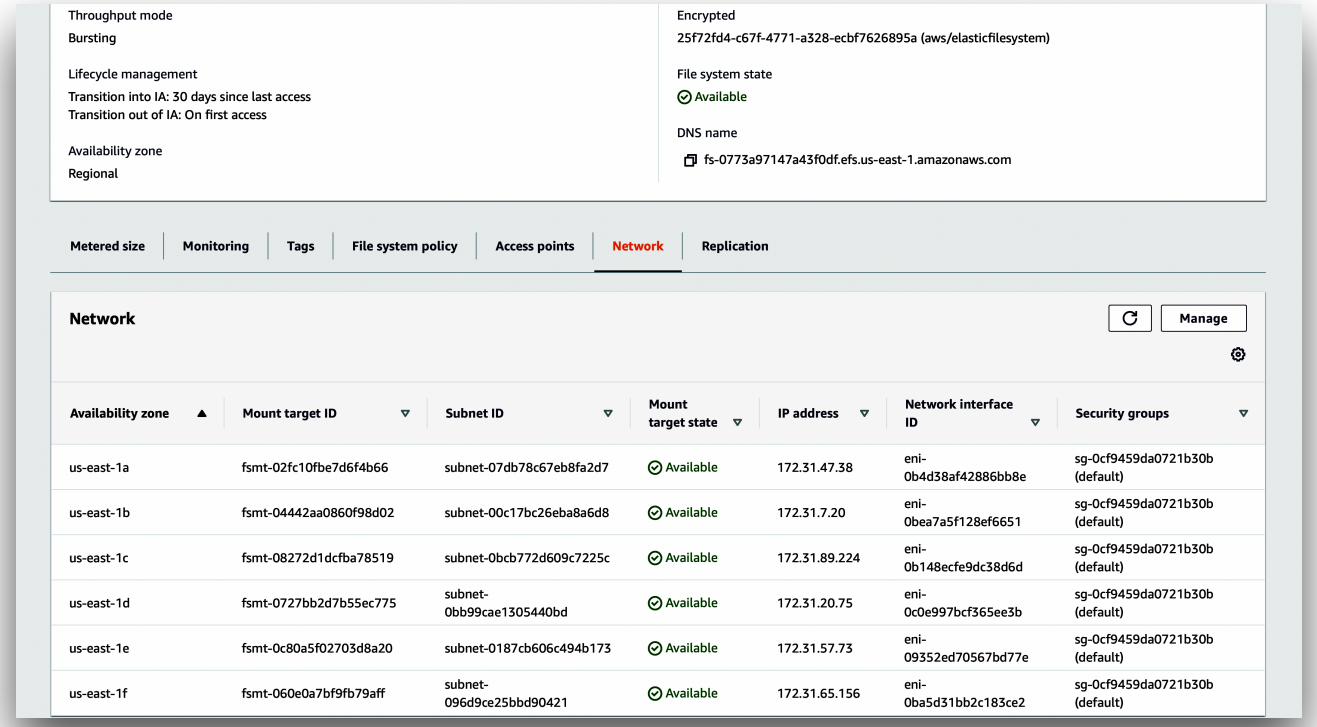


D. Choose Create to create a file system.

8. The **File systems** page appears with a banner across the top showing the status of the file system you created. A link to access the file system details page appears in the banner when the file system becomes available. Click **View file system**.



9. Within **Network**, you can see the number of mount targets created automatically. By default, this deployment will create a mount target in each Availability Zone in the AWS Region.



Availability zone	Mount target ID	Subnet ID	Mount target state	IP address	Network interface ID	Security groups
us-east-1a	fsmt-02fc10fbe7d6f4b66	subnet-07db78c67eb8fa2d7	Available	172.31.47.38	eni-0b4d38af42886bb8e	sg-0cf9459da0721b30b (default)
us-east-1b	fsmt-04442aa0860f98d02	subnet-00c17bc26eba8a6d8	Available	172.31.7.20	eni-0bea7a5f128ef6651	sg-0cf9459da0721b30b (default)
us-east-1c	fsmt-08272d1dcfba78519	subnet-0bcb772d609c7225c	Available	172.31.89.224	eni-0b148ecfe9dc38d6d	sg-0cf9459da0721b30b (default)
us-east-1d	fsmt-0727bb2d7b55ec775	subnet-0bb99cae1305440bd	Available	172.31.20.75	eni-0c0e997bcf365ee3b	sg-0cf9459da0721b30b (default)
us-east-1e	fsmt-0c80a5f02703d8a20	subnet-0187cb606c494b173	Available	172.31.57.73	eni-09352ed70567bd77e	sg-0cf9459da0721b30b (default)
us-east-1f	fsmt-060e0a7bf9fb79aff	subnet-096d9ce25bbd90421	Available	172.31.65.156	eni-0ba5d31bb2c183ce2	sg-0cf9459da0721b30b (default)

A *mount target* provides an IP address for an NFSv4 endpoint at which you can mount an Amazon EFS file system. You mount your file system using its Domain Name Service (DNS) name, which resolves to the IP address of the EFS mount target in the same Availability Zone as your EC2 instance. You can create one mount target in each Availability Zone in an AWS Region. If there are multiple subnets in an Availability Zone in your VPC, you create a mount target in one of the subnets. Then all EC2 instances in that Availability Zone share that mount target.

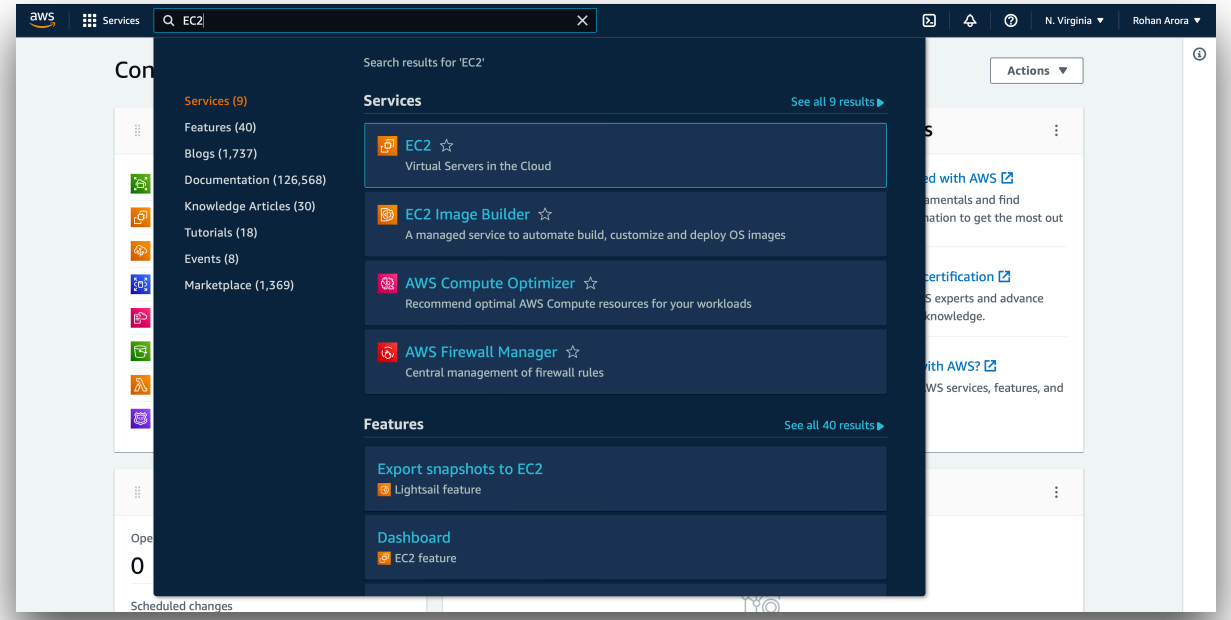
An Amazon EFS file system can only have mount targets in one VPC at a time.

Mount targets themselves are designed to be highly available. As you design for high availability and failover to other Availability Zones, keep in mind that while the IP addresses and DNS for your mount targets in each Availability Zone are static, they are redundant components backed by multiple resources.

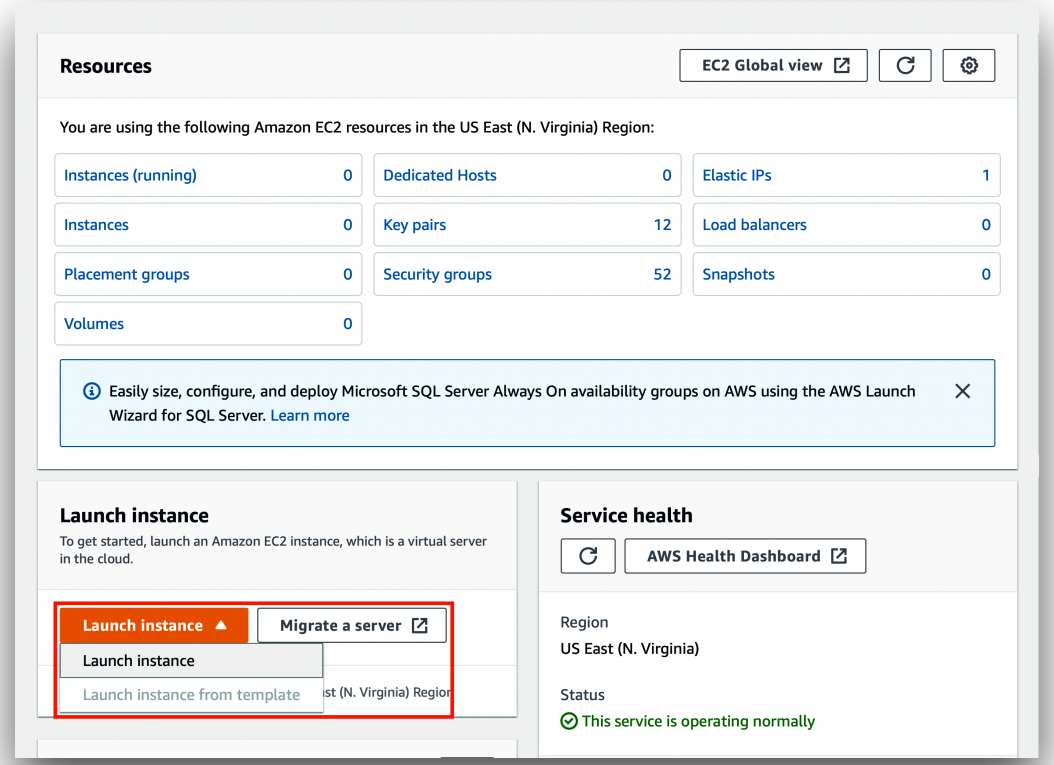
Launch two Amazon Linux EC2 instances

In this step you will create two new Amazon EC2 instances running Amazon Linux 2, and configure it to automatically mount the EFS file system you just created in the previous step.

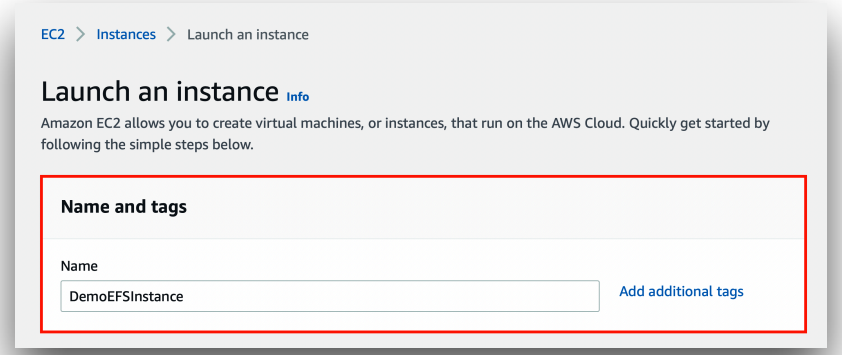
10. Navigate to EC2 console by searching for **EC2** on the top in the search bar and clicking on the presented option.



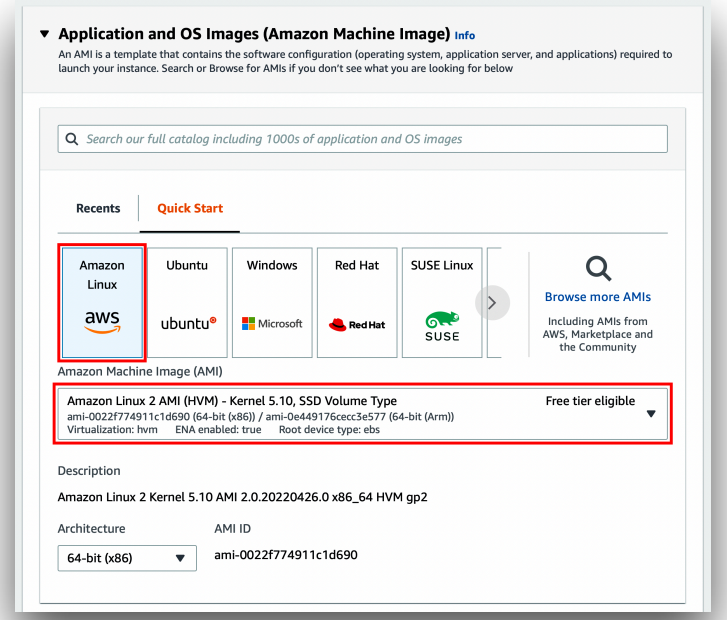
11. From the Amazon EC2 console dashboard, choose **Launch instance**



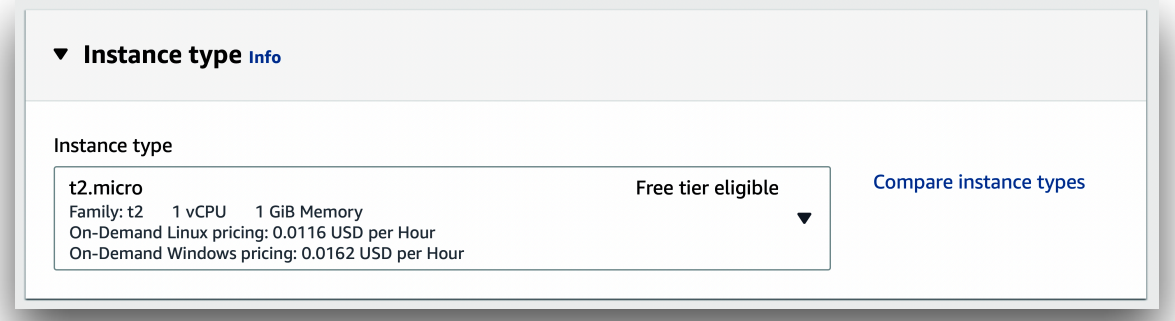
12. For **Name**, enter a descriptive name for the instances.



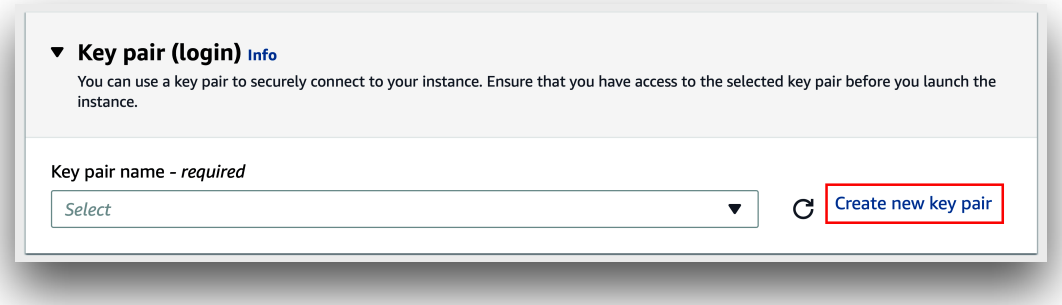
13. Under **Application and OS Images (Amazon Machine Image)**, choose **Quick Start**, and then choose the **Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type** for your instance.



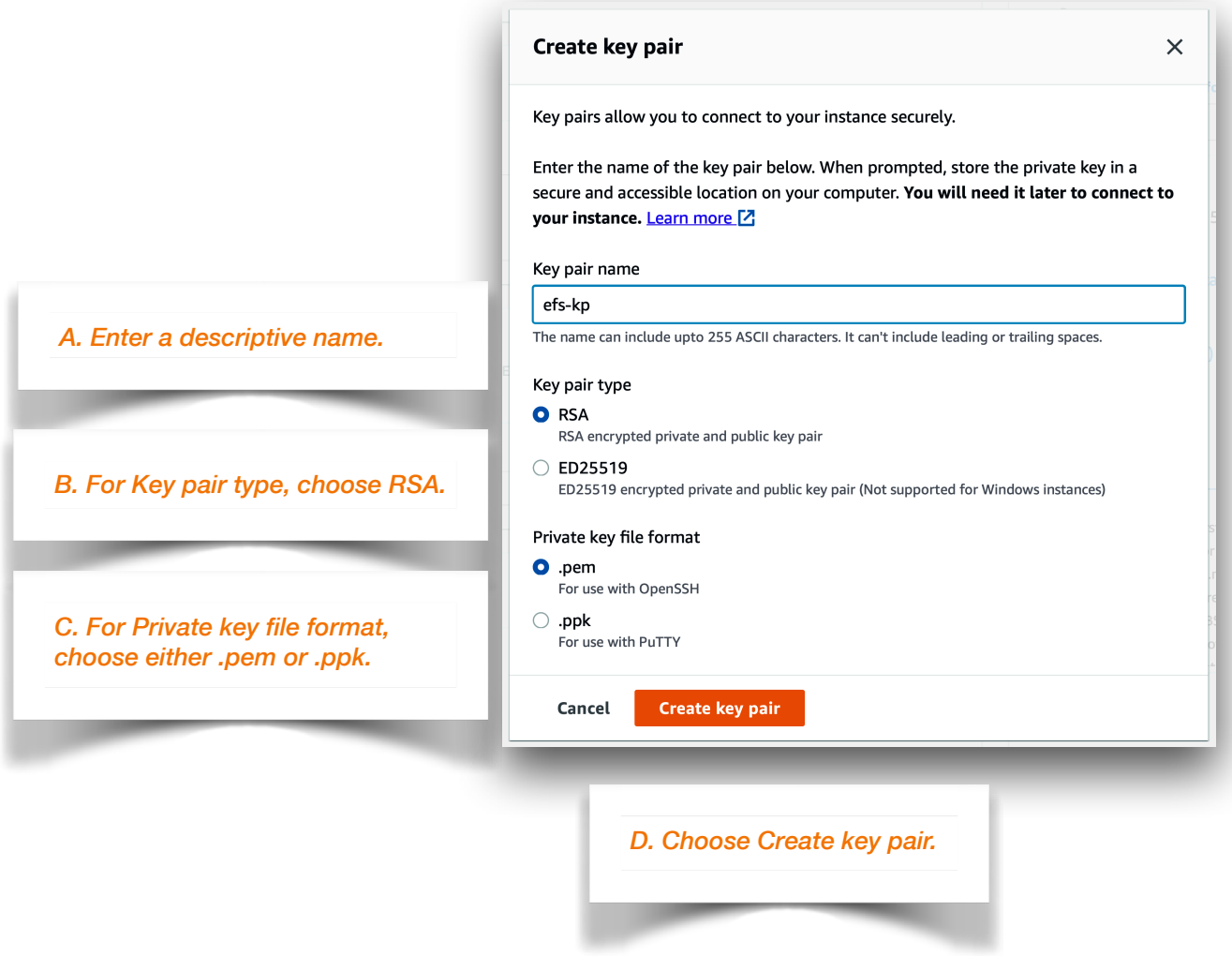
14. For **Instance type**, select either the **t2.micro** or **t3.micro** instance type (depending on the availability) for the instance.



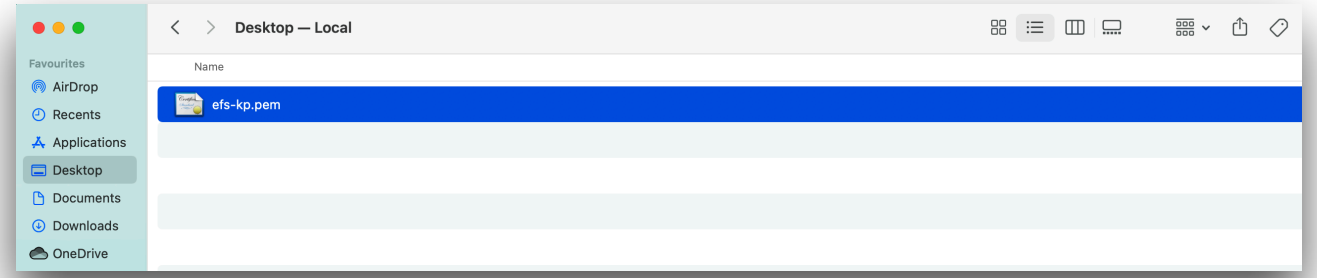
15. For **Key pair name**, choose an existing key pair, or choose **Create new key pair** to create a new one.



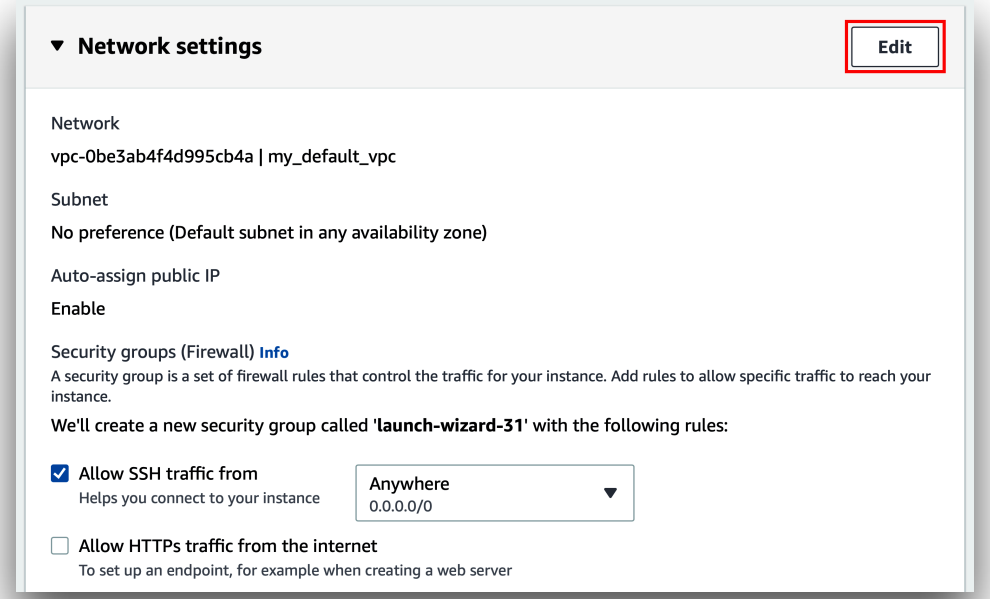
- For **Name**, enter a descriptive name for the key pair. Amazon EC2 associates the public key with the name that you specify as the key name. A key name can include up to 255 ASCII characters. It can't include leading or trailing space
- For **Key pair type**, choose either **RSA** or **ED25519**.
- For **Private key file format**, choose the format in which to save the private key. To save the private key in a format that can be used with OpenSSH, choose **pem**. To save the private key in a format that can be used with PuTTY, choose **ppk**.
- Choose **Create key pair**.



- The private key file is automatically downloaded by your browser. The base file name is the name that you specified as the name of your key pair, and the file name extension is determined by the file format that you chose. Save the private key file in a safe place.



16. To configure some of the network settings, click **Edit** in **Network settings** section.



17. Configure the networking settings as following:

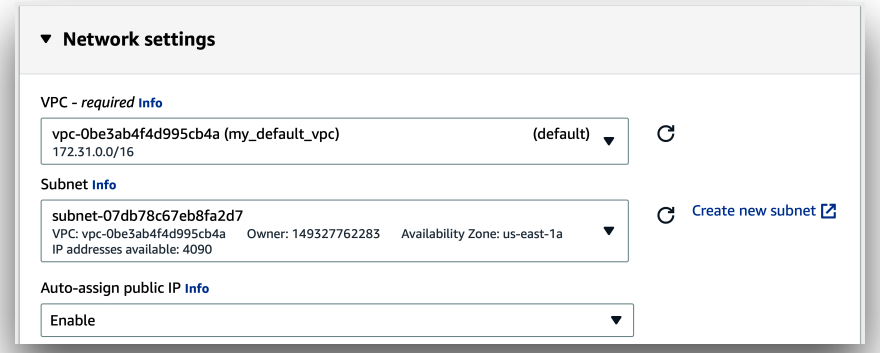
- **Network:** Here the default VPC is selected automatically. You will be launching this EC2 instance in the same default VPC of the region you're working in.

- **Subnet**: Select any of the available subnets linked to any specific availability zone. This is an important step to be performed before you can add a file system.

- **Auto-assign Public IP**: Specify whether your instance receives a public IPv4 address. By default, instances in a default subnet receive a public IPv4 address, and instances in a non-default subnet don't. You can select **Enable** or **Disable** to override the subnet's default setting. For this hands-on, keep **Auto-assign public IP** to **Enable**.

A. Keep the VPC set to default.

B. Select any one of the subnets.



C. Keep Auto-assign public IP to Enable state.

18. Use a security group to define firewall rules for your instance. These rules specify which incoming network traffic is delivered to your instance. All other traffic is ignored. Create a new security group which allows inbound SSH access from a known IP address.

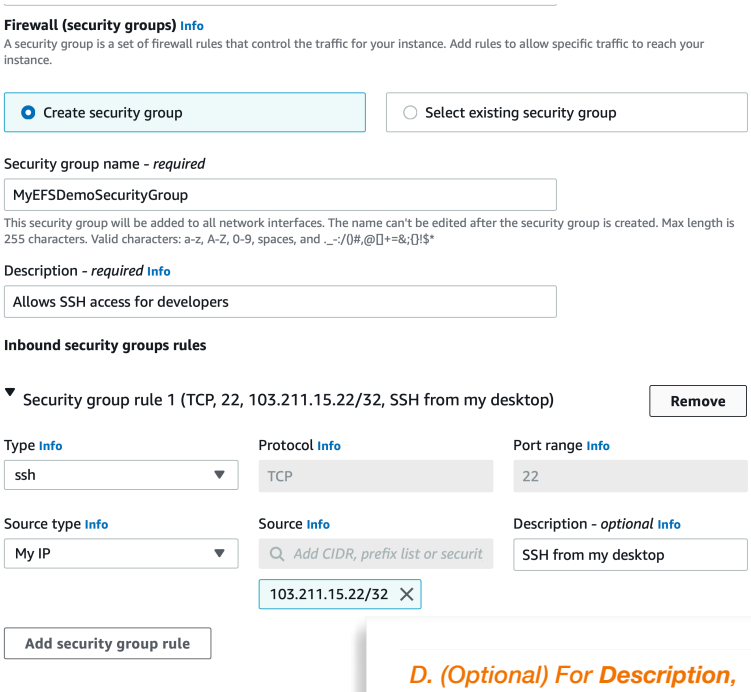
Create a security group as follows:

- To create a new security group, choose **Create security group**.
- Allocate a name to this security group while including a description as well.
- To let the launch instance wizard add your computer's public IP address, choose **My IP**. Optionally, you can include a description for this rule.

*A. Choose **Create Security Group**.*

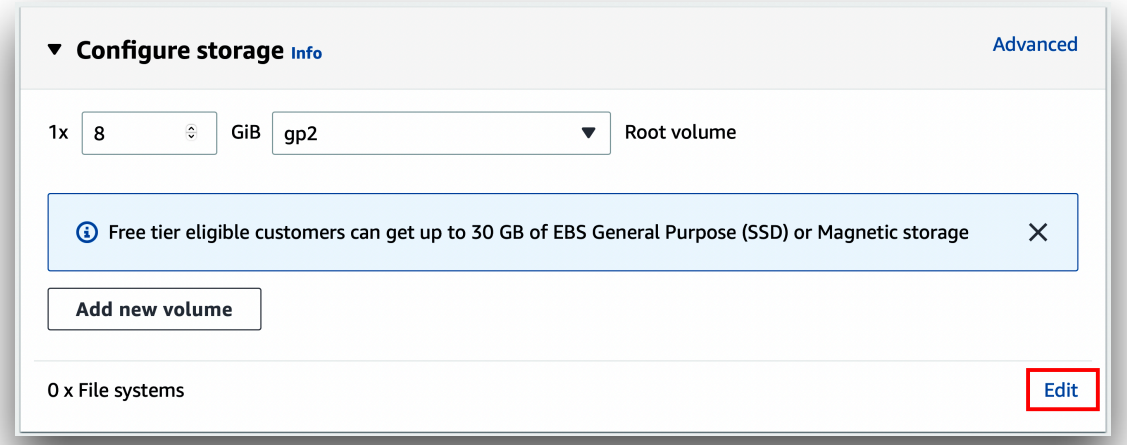
B. Enter a name and description for the security group.

*C. For SSH, select **Source type** as **MyIP**.*

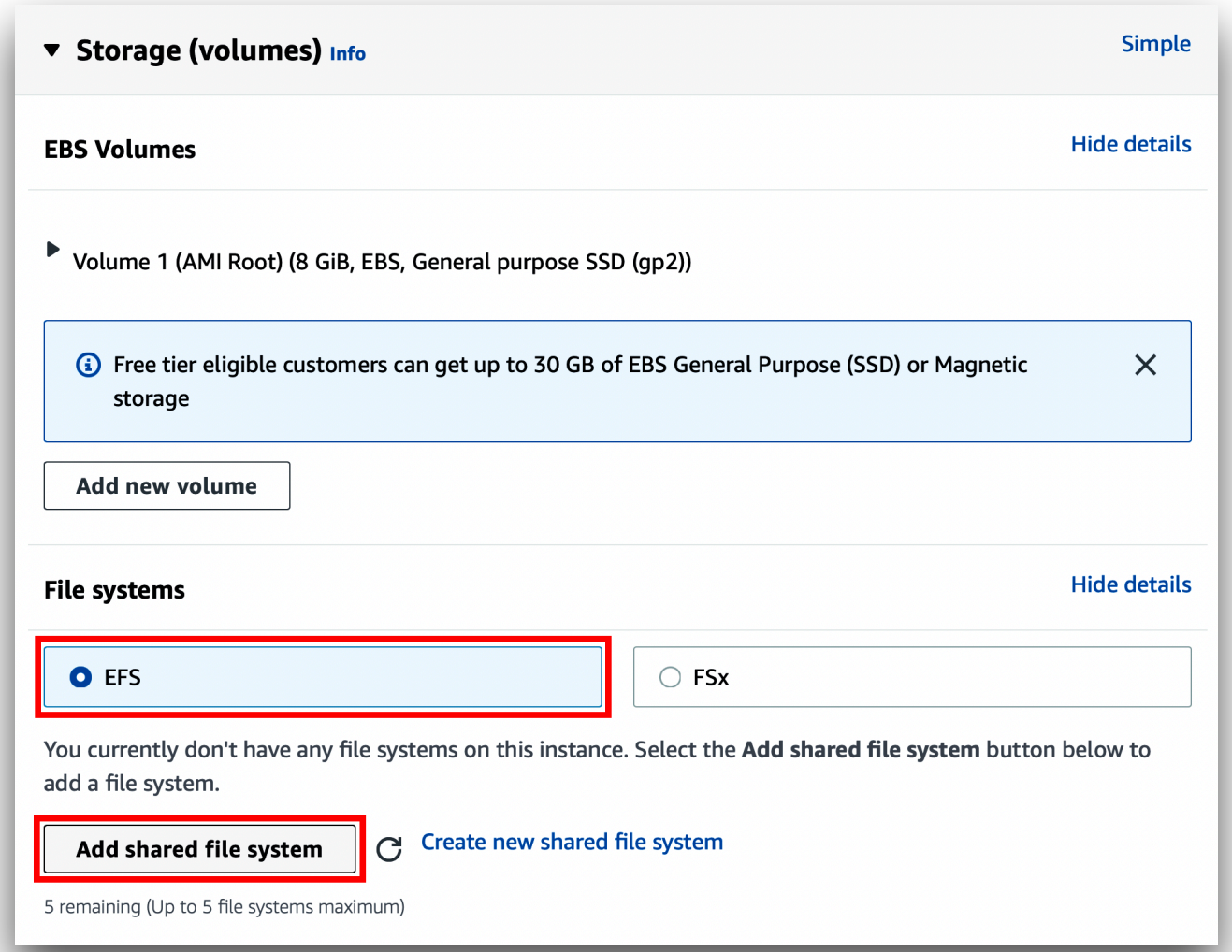


*D. (Optional) For **Description**, specify a brief description for the rule.*

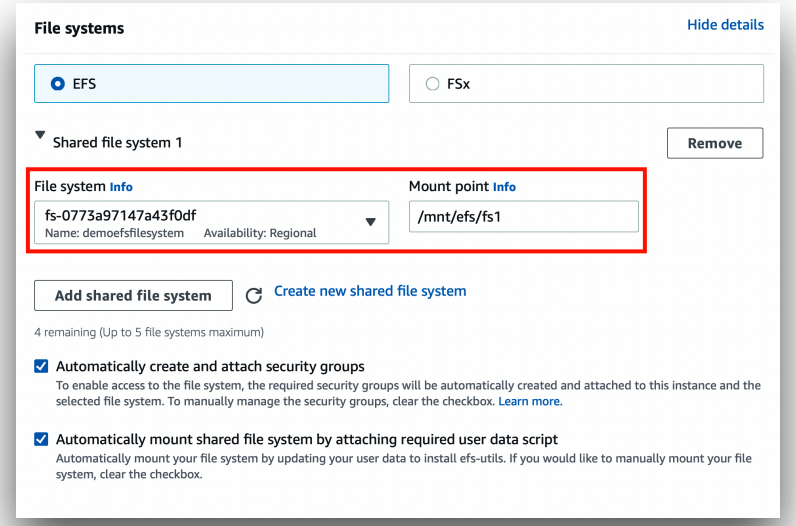
19. Within **Configure storage**, click **Edit** to mount the EFS file system to the instances.



20. Select **EFS** within **File systems**, and then click **Add shared file system**.



21. Ensure that the value matches the file system ID that you created in the previous step. The path shown next to the file system ID is the mount point that the instance will use, which you can change.



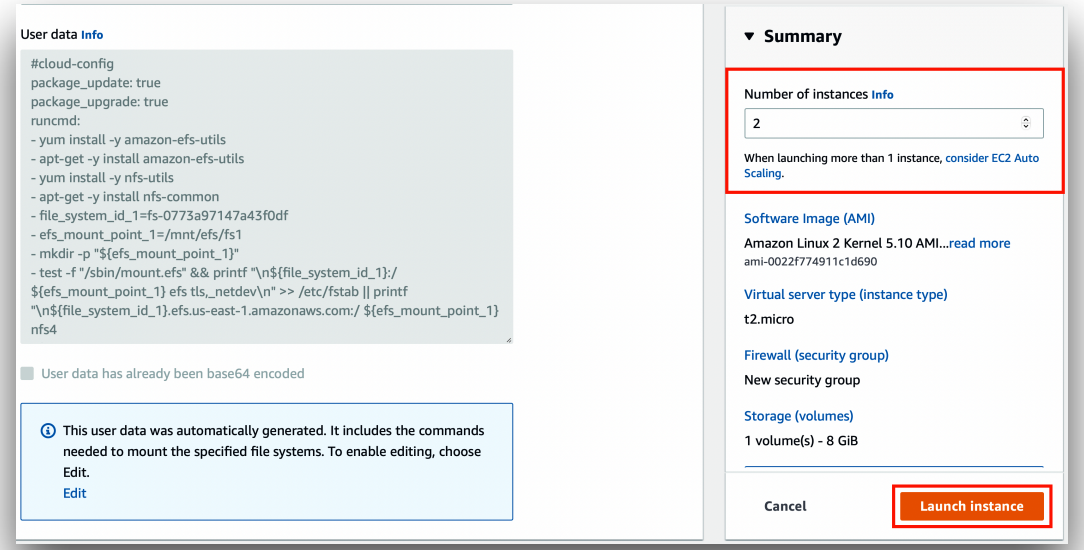
Note that upon adding the shared file system following actions will be performed automatically:

- Another security group will be created and attached to the instances, which will allow inbound NFS connections to the file system via the EFS mount target from the EC2 instances that are associated with this security group (the source is the security group itself).

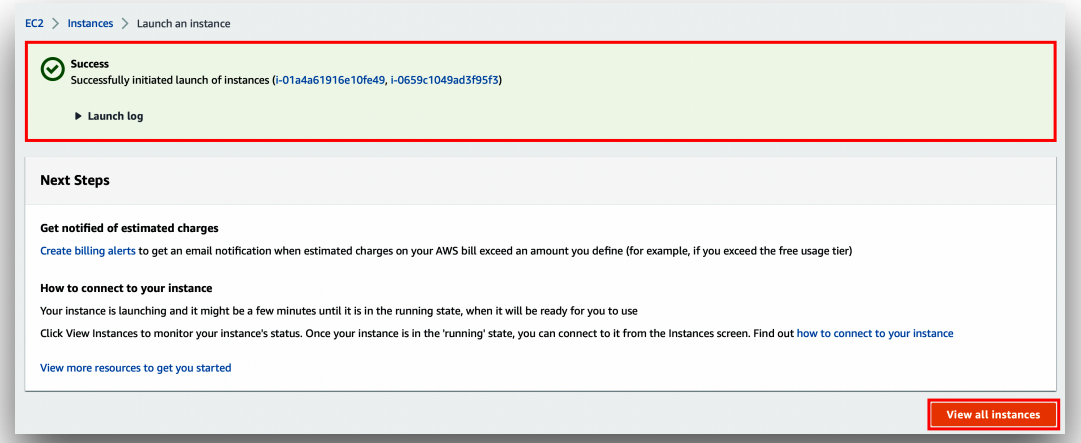
- Under **Advanced Details**, the **User data** is automatically generated, and includes the commands needed to mount the file system.

Hence, you should ensure that options **Automatically create and attach security groups** and **Automatically mount shared file system by attaching required user data script** remain selected.

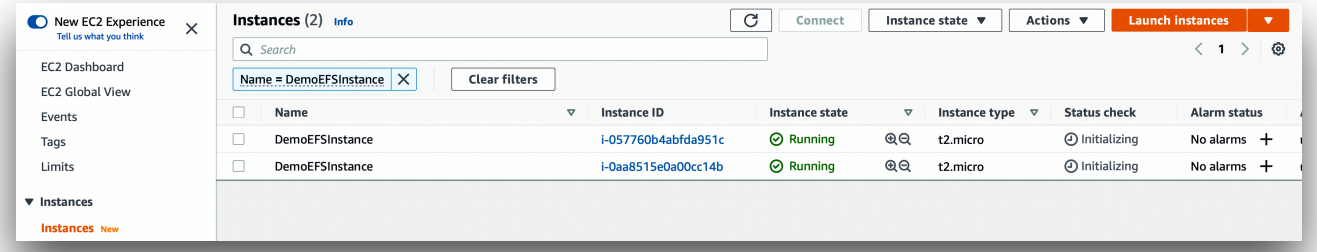
22. Within **Summary**, mention **2** for the **Number of instances** and click **Launch instance**. This will allow you to create two Amazon EC2 Linux instances while adding the shared EFS file system to them.



23. The EC2 instances will now be successfully launched. Click **View all instances** to view and access these newly launched EC2 instances on the instance dashboard.



24. The newly launched Amazon EC2 linux instances will be displayed as follows. Initially, their status is `pending`. After the status changes to `running`, your instances are ready for use.



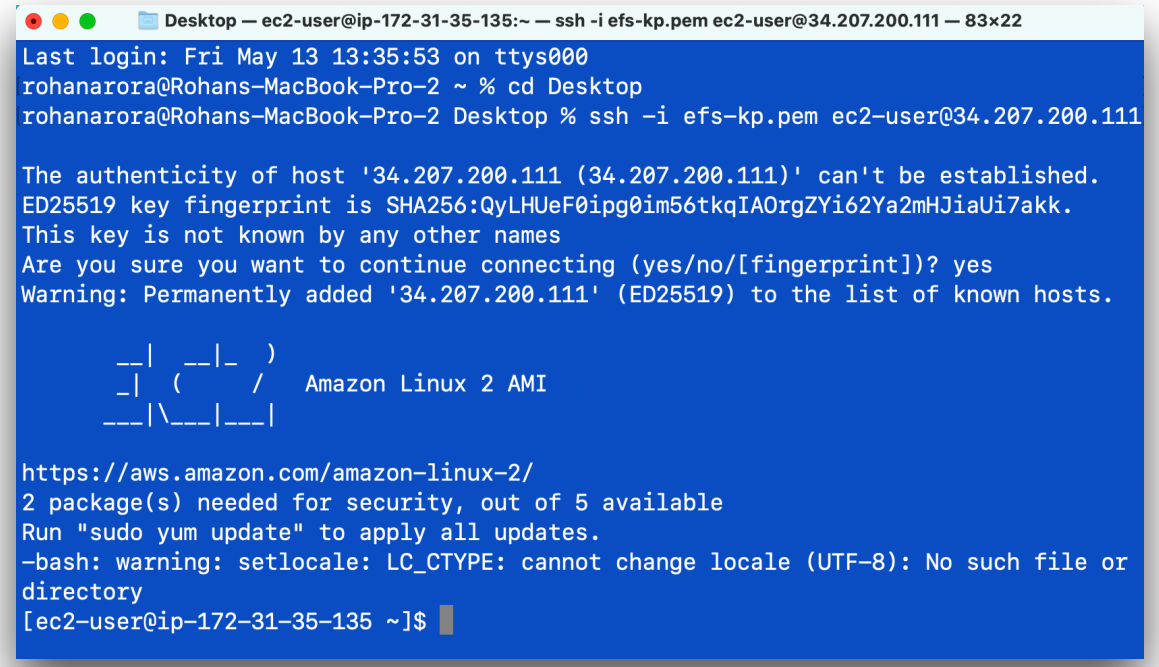
Your instances are now configured to mount the Amazon EFS file system at launch and whenever they're rebooted.

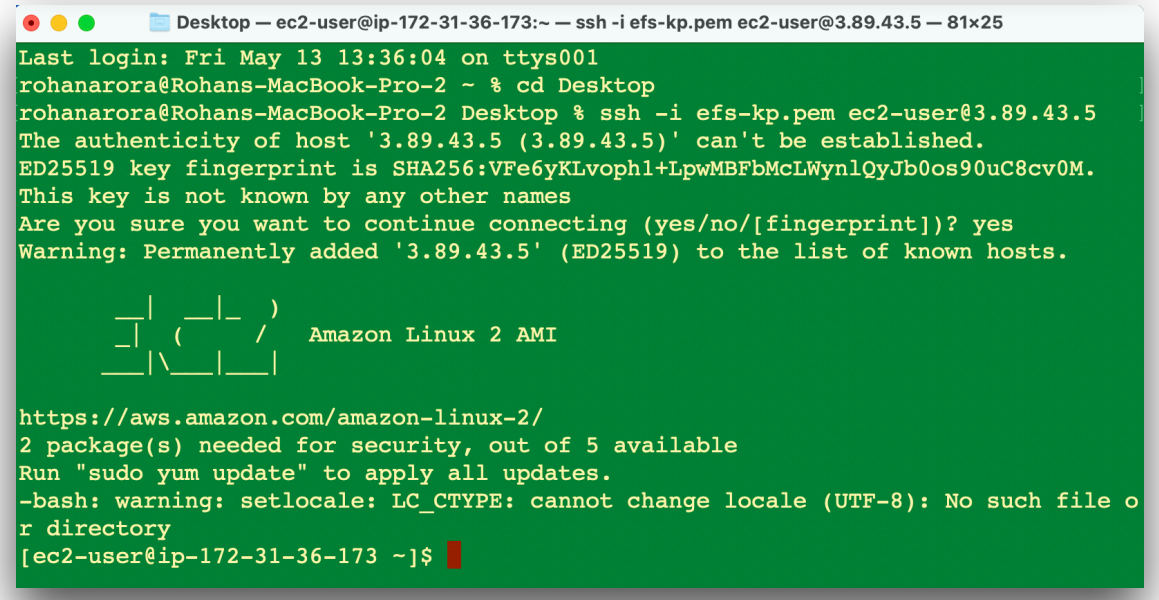
Test the file system

You can connect to your instances and verify that the file system is mounted to the directory that you specified (for example, /mnt/efs).

To verify that the file system is mounted

25. Connect to your instances. Initiate an SSH connection to these Amazon EC2 Linux instances.





26. From the terminal window for each instance, run the **df -T** command to verify that the EFS file system is mounted.

```
[ec2-user@ip-172-31-35-135 ~]$ df -T
Filesystem     Type            1K-blocks     Used        Available Use% Mounted on
devtmpfs       devtmpfs           485340        0           485340   0% /dev
tmpfs          tmpfs              493848        0           493848   0% /dev/shm
tmpfs          tmpfs              493848      524           493324   1% /run
tmpfs          tmpfs              493848        0           493848   0% /sys/fs/cgroup
/dev/xvda1     xfs               8376300  1612628          6763672  20% /
127.0.0.1:/    nfs4     9007199254739968        0 9007199254739968   0% /mnt/efs/fs1
tmpfs          tmpfs               98772        0            98772   0% /run/user/0
tmpfs          tmpfs               98772        0            98772   0% /run/user/1000
[ec2-user@ip-172-31-35-135 ~]$
```

```
[ec2-user@ip-172-31-36-173 ~]$ df -T
Filesystem     Type            1K-blocks     Used        Available Use% Mounted on
devtmpfs       devtmpfs           485340        0           485340   0% /dev
tmpfs          tmpfs              493848        0           493848   0% /dev/shm
tmpfs          tmpfs              493848      524           493324   1% /run
tmpfs          tmpfs              493848        0           493848   0% /sys/fs/cgroup
/dev/xvda1     xfs               8376300  1609892          6766408  20% /
127.0.0.1:/    nfs4     9007199254739968        0 9007199254739968   0% /mnt/efs/fs1
tmpfs          tmpfs               98772        0            98772   0% /run/user/0
tmpfs          tmpfs               98772        0            98772   0% /run/user/1000
[ec2-user@ip-172-31-36-173 ~]$
```

27. Create a file in the file system from one instance, and then verify that you can view the file from the other instance.

- From the first instance, run the following command to create the file.

```
sudo touch /mnt/efs/fs1/test-file.txt
```

```
[ec2-user@ip-172-31-35-135 ~]$ df -T
Filesystem     Type            1K-blocks     Used        Available Use% Mounted on
devtmpfs       devtmpfs           485340        0           485340   0% /dev
tmpfs          tmpfs              493848        0           493848   0% /dev/shm
tmpfs          tmpfs              493848      472           493376   1% /run
tmpfs          tmpfs              493848        0           493848   0% /sys/fs/cgroup
/dev/xvda1     xfs               8376300  1621264          6755036  20% /
127.0.0.1:/    nfs4     9007199254739968        0 9007199254739968   0% /mnt/efs/fs1
tmpfs          tmpfs               98772        0            98772   0% /run/user/1000
[ec2-user@ip-172-31-35-135 ~]$ sudo touch /mnt/efs/fs1/test-file.txt
[ec2-user@ip-172-31-35-135 ~]$
```

- From the second instance, run the following command to view the file.

```
ls /mnt/efs/fs1
```

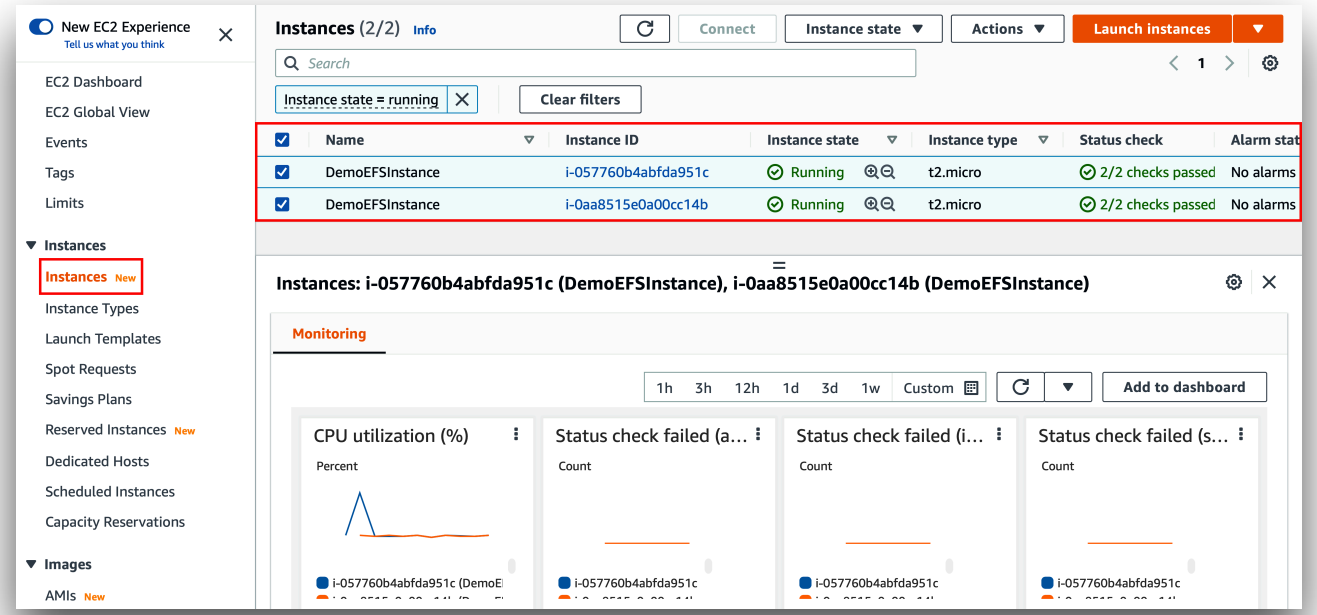
```
[ec2-user@ip-172-31-36-173 ~]$ df -T
Filesystem     Type            1K-blocks     Used        Available Use% Mounted on
devtmpfs       devtmpfs           485340        0           485340   0% /dev
tmpfs          tmpfs              493848        0           493848   0% /dev/shm
tmpfs          tmpfs              493848      472           493376   1% /run
tmpfs          tmpfs              493848        0           493848   0% /sys/fs/cgroup
/dev/xvda1     xfs               8376300  1621240          6755060  20% /
127.0.0.1:/    nfs4      9007199254739968        0 9007199254739968   0% /mnt/efs/fs1
tmpfs          tmpfs               98772        0            98772   0% /run/user/1000
[ec2-user@ip-172-31-36-173 ~]$ ls /mnt/efs/fs1
test-file.txt
[ec2-user@ip-172-31-36-173 ~]$
```

Clean up

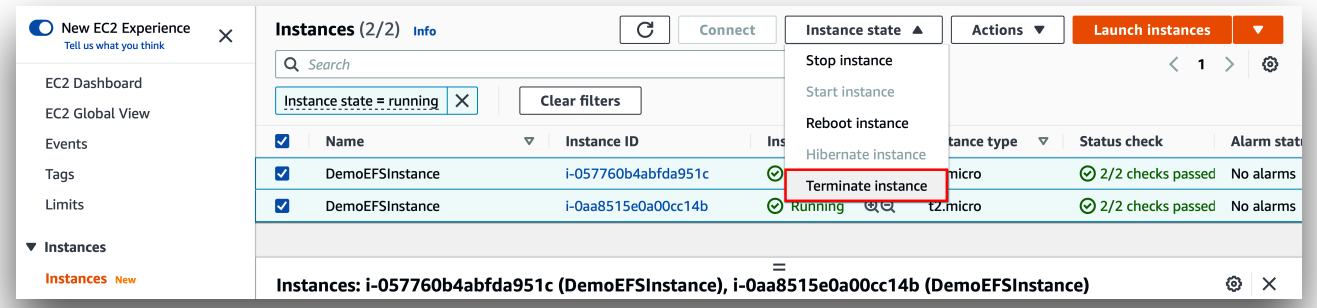
When you are finished with this tutorial, you can terminate the instances and delete the file system.

To terminate the instances

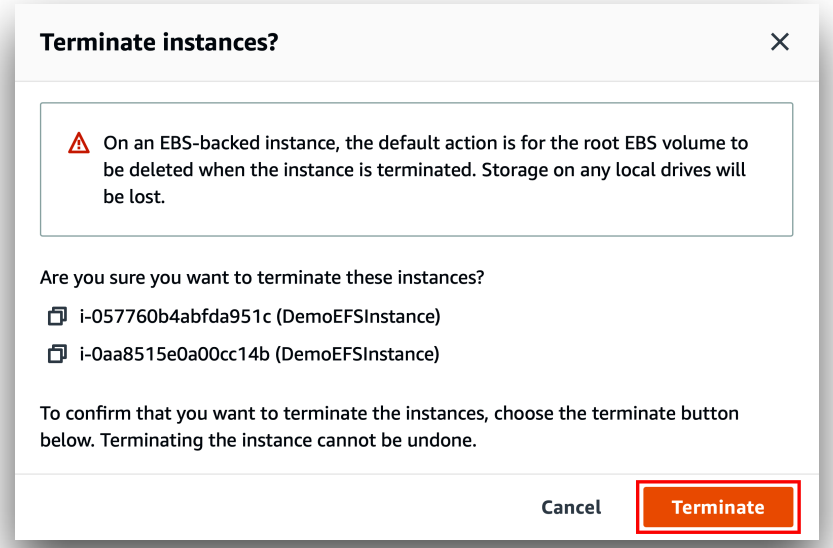
28. Go back to EC2 dashboard, choose **Instances** from the navigation pane and then select the instances to terminate.



29. Choose **Instance state**, **Terminate instance**.

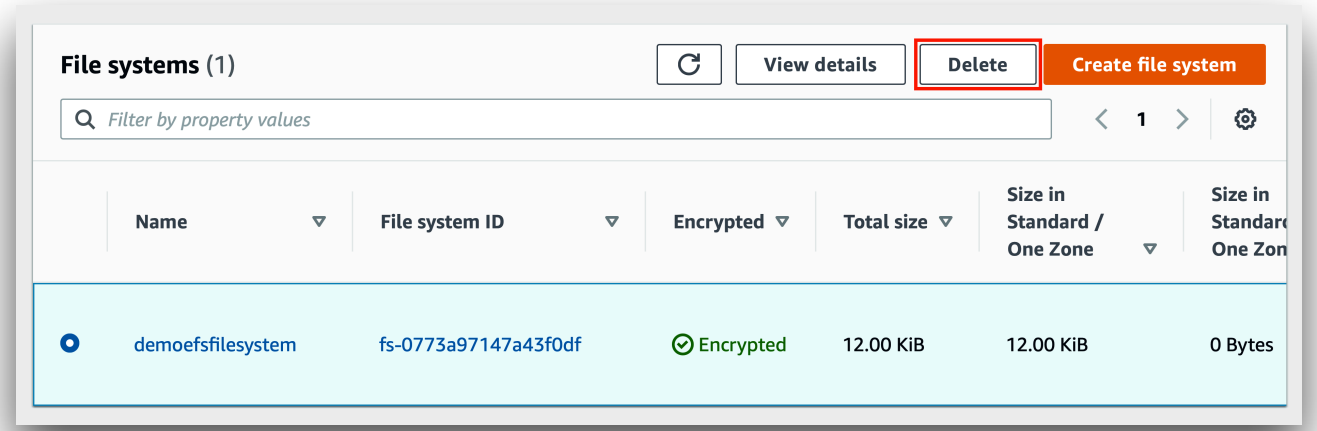


30. Choose **Terminate** when prompted for confirmation.

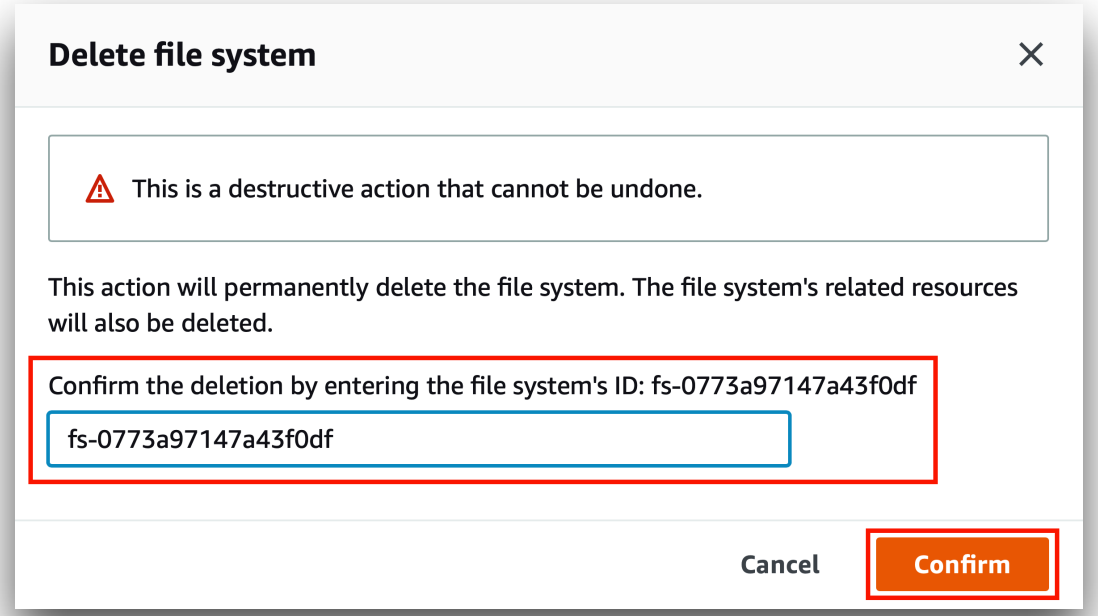


To delete the file system

31. Open the Amazon Elastic File System console and select the file system to delete, and click **Delete**.



32. When prompted for confirmation, enter the file system ID and choose **Confirm**.



Summary

This completes the hands-on exercise on creating an EFS file system and mounting it on two Amazon Elastic Compute Cloud (Amazon EC2) instances in your virtual private cloud (VPC). You then tested the end-to-end setup by creating a file in the file system from one instance, and then verified that you can view the file from the other instance.

After testing, resources including both EC2 instances and EFS file system were terminated to clean up and avoid any unexpected charges.