

# Experiment 03 - Cloud Computing

December 14, 2022

## Aim

Mount EFS to multiple EC2 instances in different availability zones.

## Theory

**Amazon Elastic File System** (EFS) is a scalable cloud file storage solution for use with EC2 instances. It is elastic because it will automatically grow and shrink as you add/remove files. It has a simple interface that enables you to create and configure file systems quickly and simply. It is similar to EBS, but with EBS you can only mount your virtual disk to one EC2 instance. You can have 2 instances sharing an EFS volume.

## Results

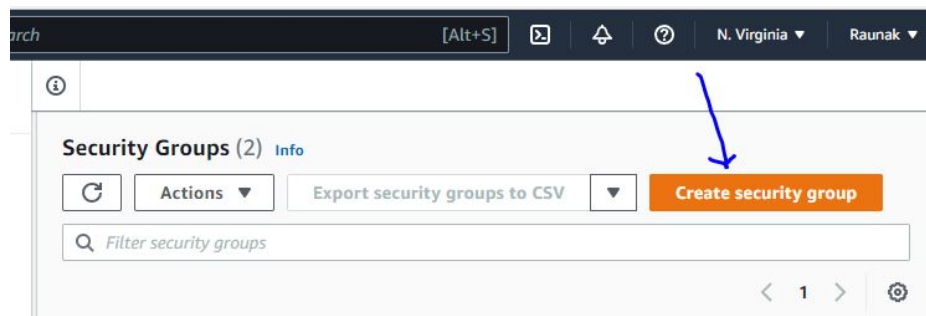


Figure 1: Start by creating a security group

Basic details

Security group name [Info](#)

genericSecurityGroup

Name cannot be edited after creation.

Description [Info](#)

Allows SSH access to developers

VPC [Info](#)

Q vpc-055b0f13119b0e718 X

Inbound rules [Info](#)

Inbound rule 1 [Delete](#)

Type [Info](#)

SSH

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere-IPv4

Source [Info](#)

Q 0.0.0.0/0 X

Description - optional [Info](#)

Add rule

Figure 2: Select type as SSH and source type as Anywhere-IPv4

Basic details

Security group name [Info](#)

genericEFSecurityGroup

Name cannot be edited after creation.

Description [Info](#)

Security for EFS

VPC [Info](#)

Q vpc-055b0f13119b0e718 X

Inbound rules [Info](#)

Inbound rule 1 [Delete](#)

Type [Info](#)

NFS

Source type [Info](#)

Custom

Source [Info](#)

Q

Description - optional [Info](#)

Add rule

:/48

:/64

Security Groups

launch-wizard-1 | sg-07d8e0243e1ace86e

genericSecurityGroup | sg-076b5964b521bea2d

default | sg-0b491988f2614d9a7

Prefix lists

com.amazonaws.us-eas... | pl-02cd2c6b

com.amazonaws.global... | pl-

Figure 3: Create one more security group and select the inbound rule with source of first security group

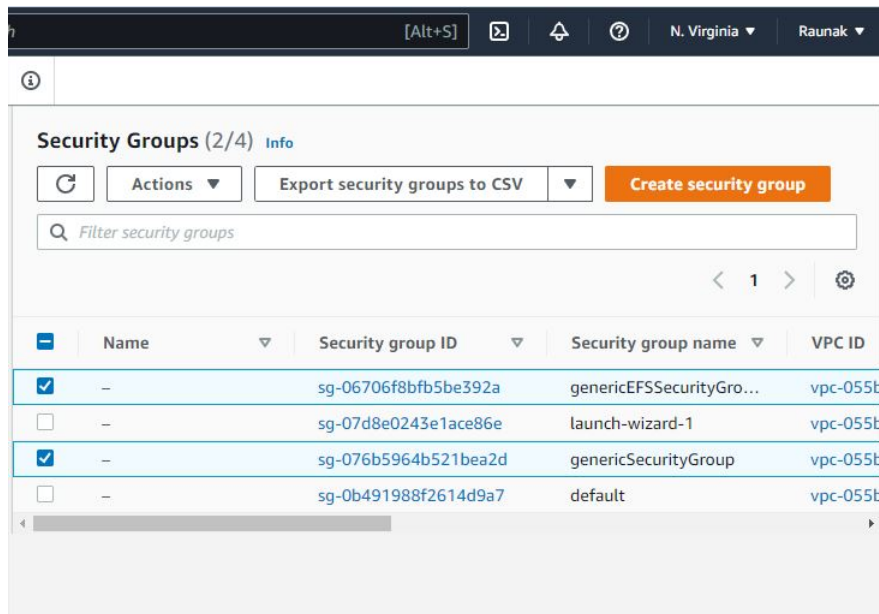


Figure 4: Successful creation of 2 security groups

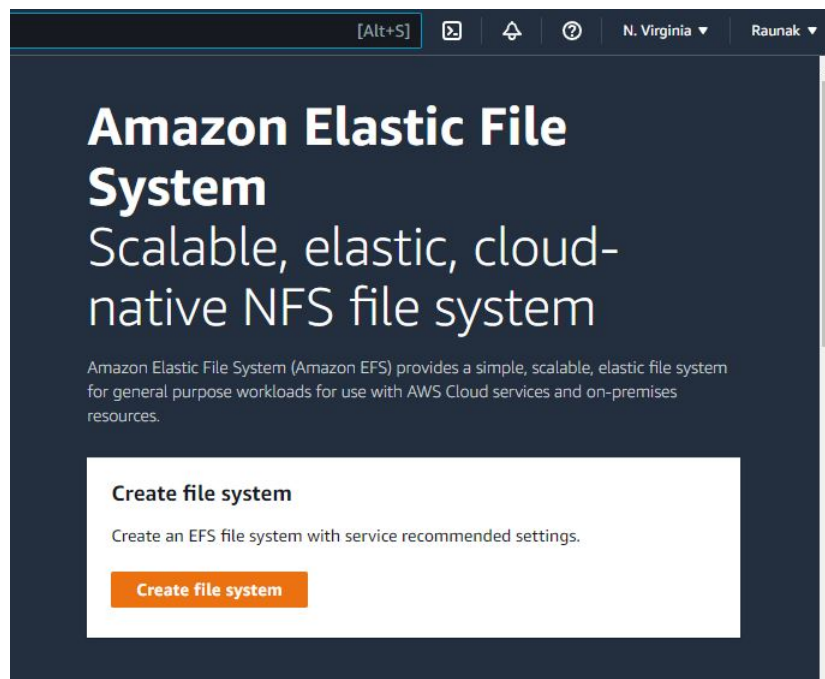


Figure 5: Create an EFS by searching for the service

**Create file system** [X]

Create an EFS file system with service recommended settings. [Learn more](#)

**Name - optional**  
Name your file system.  
genericEFS  
Name can include letters, numbers, and +-=.\_/ symbols, up to 256 characters.

**Virtual Private Cloud (VPC)**  
Choose the VPC where you want EC2 instances to connect to your file system. [Learn more](#)  
vpc-055b0f13119b0e718  
default

**Storage class** [Learn more](#)

☒ **Standard**  
Stores data redundantly across multiple AZs

☐ **One Zone**  
Stores data redundantly within a single AZ

Cancel Customize Create

Figure 6: Name the EFS and select the customization option before creating it

**General**

**Name - optional**  
Name your file system.  
genericEFS  
Name can include letters, numbers, and +-=.\_/ symbols, up to 256 characters.

**Storage class** [Learn more](#)

☒ **Standard**  
Stores data redundantly across multiple AZs

☐ **One Zone**  
Stores data redundantly within a single AZ

**Automatic backups**  
Automatically backup your file system data with AWS Backup using recommended settings. Additional pricing applies. [Learn more](#)  
☐ **Enable automatic backups**

**Lifecycle management**  
EFS Intelligent-Tiering uses Lifecycle Management to automatically achieve the right price and performance blend for your application by moving your files between the Standard and Standard-Infrequent Access storage classes. [Learn more](#)

**Transition into IA**  
Transition files from Standard to Standard-Infrequent Access.  
30 day(s) since last access

**Transition out of IA**  
Transition files from Standard-Infrequent Access to Standard.  
On first access

**Encryption**  
Choose to enable encryption of your file system's data at rest. Uses the AWS KMS service key (aws/elasticfilesystem) by default. [Learn more](#)  
☐ **Enable encryption of data at rest**

Figure 7: Uncheck the automatic backups and encryption option

**Network**

Virtual Private Cloud (VPC)  
Choose the VPC where you want EC2 instances to connect to your file system. [Learn more](#)

vpc-055b0f13119b0e718  
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	Security groups	
us-east-1a	subnet-00ae...	Automatic	sg-06706f88fb5be392a genericEFSSecurityGroup	Remove
us-east-1b	subnet-0f40...	Automatic	sg-06706f88fb5be392a genericEFSSecurityGroup	Remove

Add mount target

Figure 8: Delete all the availability zones except shown in image and use the security group that was created earlier

Services Search [Alt+S] N. Virginia Raunak

Amazon EFS > File systems

File systems (1) View details Delete Create file system

Filter by property values

Name	File system ID	Encryption	Total size	Size in Standard / One Zone	Size in Standard-IA / One Zone-IA
genericEFS	fs-0c72e075a28313abd	Unencrypted	6.00 KiB	6.00 KiB	0 Bytes

Figure 9: Successful creation of the EFS

**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  
genericInstanceForEFS Add additional tags

Figure 10: Launching the instance for connecting it with EFS

▼ Network settings

Info

VPC - required

Info

vpc-055b0f13119b0e718

(default) ▼

172.31.0.0/16

↻

Subnet

Info

subnet-00ae66eba1b1ebf34

VPC: vpc-055b0f13119b0e718   Owner: 303164066091   Availability Zone: us-east-1a   ▼

IP addresses available: 4090   CIDR: 172.31.0.0/20)

↻

Create new subnet

Auto-assign public IP

Info

Enable

▼

Firewall (security groups)

Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group
 ☒ Select existing security group

Common security groups

Info

Select security groups

▼

genericSecurityGroup

sg-076b5964b521bea2d

×

VPC: vpc-055b0f13119b0e718

↻

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▶ Advanced network configuration

Figure 11: Select sub-net with first availability zone and select existing security group with created security group earlier

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

secondGenericInstance

Add additional tags

Figure 12: Launch another instance

▼ **Network settings** [Info](#)

VPC - required [Info](#)

vpc-055b0f13119b0e718 (default) ↕

172.31.0.0/16

Subnet [Info](#)

subnet-0f4047b6880473825

VPC: vpc-055b0f13119b0e718 Owner: 303164066091 Availability Zone: us-east-1b ↕

IP addresses available: 4089 CIDR: 172.31.80.0/20

Auto-assign public IP [Info](#)

Enable

**Firewall (security groups)** [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group ☒ Select existing security group

Common security groups [Info](#)

Select security groups

genericSecurityGroup sg-076b5964b521bea2d ✕

VPC: vpc-055b0f13119b0e718

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

► Advanced network configuration

Figure 13: Select sub-net with second availability zone and select existing security group with created security group earlier

th [Alt+S] [Icons] N. Virginia ▼ Raunak ▼

**Instances (2/3)** [Info](#)

↻ Connect Instance state ▼ Actions ▼ Launch instances ▼

Find instance by attribute or tag (case-sensitive)

	Name ▼	Instance ID	Instance state ▼	Instance type
<input type="checkbox"/>	myFirstInstance	i-0b9bea4c3cfd31876	⊖ Stopped	t2.micro
<input checked="" type="checkbox"/>	genericInstanceForEFS	i-0579550348dcfee95	⊕ Running	t2.micro
<input checked="" type="checkbox"/>	secondGenericInstance	i-040101b0ec14aec2e	⊕ Running	t2.micro

Figure 14: Successful creation of 2 Linux based instances

```
login as: ec2-user
Authenticating with public key "genericKeyPair"

  _ | _ | _ |
  _ | ( _ | _ | /
  _ | \ _ | _ |
  _ | \ _ | _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-10-136 ~]$ sudo yum -y install amazon-efs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.34.1-1.amzn2 will be installed
--> Processing Dependency: stunnel5 for package: amazon-efs-utils-1.34.1-1.amzn2.noarch
--> Running transaction check
--> Package stunnel5.x86_64 0:5.58-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                                Arch      Version                                Repository    Size
=====
Package                                Arch      Version                                Repository    Size
=====
```

Figure 15: Connect with SSH and run `sudo yum -y install amazon-efs-utils` command

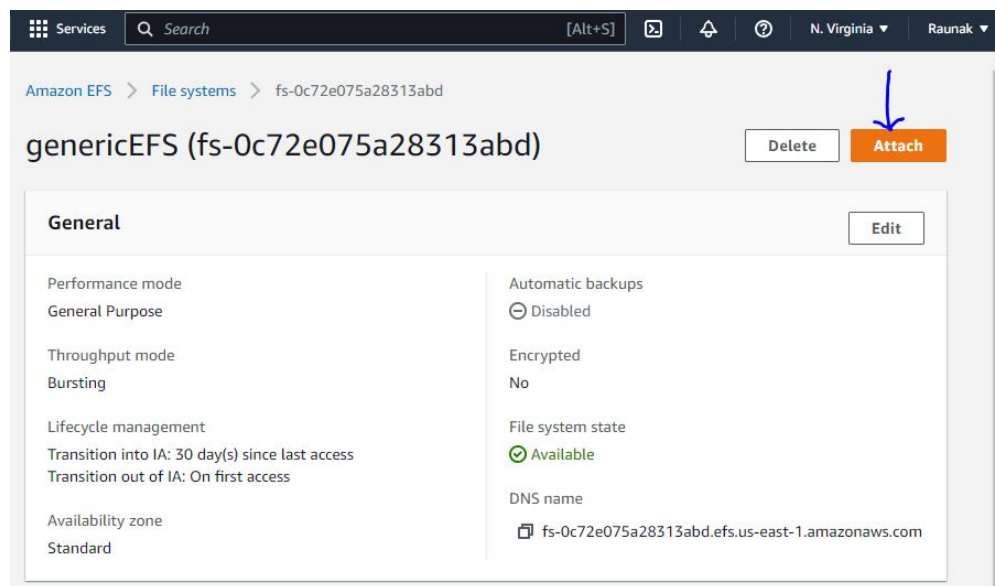


Figure 16: Attach the EFS by selecting the option



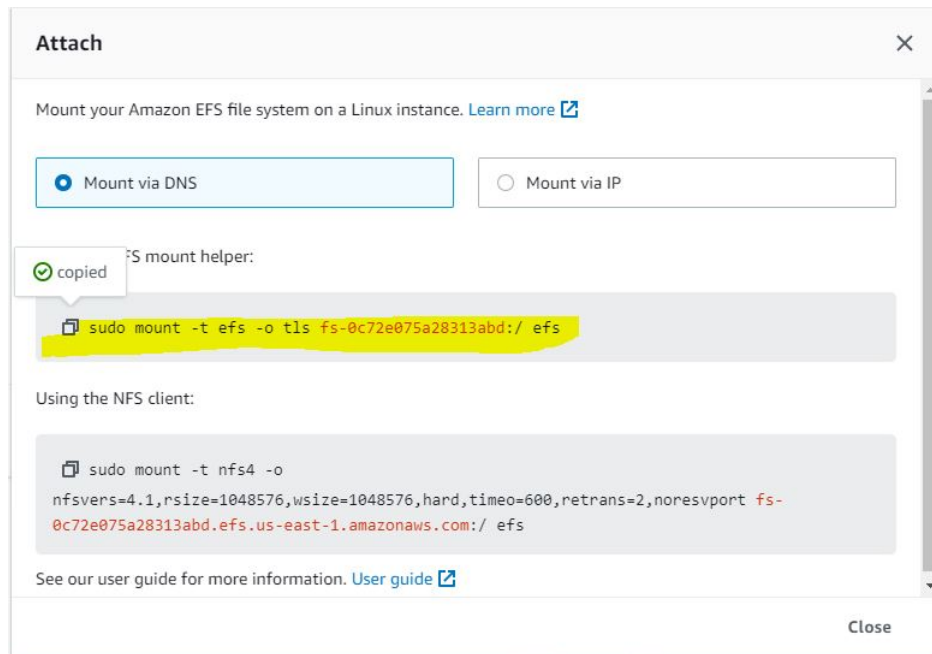


Figure 17: Copy the mounting command of the EFS

```
[ec2-user@ip-172-31-10-136 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-10-136 ~]$ sudo mount -t efs -o tls fs-0c72e075a28313abd:/ efs
b'mount.nfs4: mount point efs does not exist'
[ec2-user@ip-172-31-10-136 ~]$ sudo mount -t efs -o tls fs-0c72e075a28313abd:/ /efs
[ec2-user@ip-172-31-10-136 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        474M   0   474M   0% /dev
tmpfs           483M   0   483M   0% /dev/shm
tmpfs           483M 520K   482M   1% /run
tmpfs           483M   0   483M   0% /sys/fs/cgroup
/dev/xvda1      8.0G  1.6G   6.5G  20% /
tmpfs           97M   0    97M   0% /run/user/1000
127.0.0.1:/     8.0E   0    8.0E   0% /efs
[ec2-user@ip-172-31-10-136 ~]$ cd /efs
[ec2-user@ip-172-31-10-136 efs]$ sudo mkdir myData
[ec2-user@ip-172-31-10-136 efs]$ cd myData/
[ec2-user@ip-172-31-10-136 myData]$ cd ..
[ec2-user@ip-172-31-10-136 efs]$ sudo chown ec2-user myData
[ec2-user@ip-172-31-10-136 efs]$ ls -lrth
total 4.0K
drwxr-xr-x 2 ec2-user root 6.0K Dec 25 16:39 myData
[ec2-user@ip-172-31-10-136 efs]$ cd myData
[ec2-user@ip-172-31-10-136 myData]$ touch genericTestFile.txt
[ec2-user@ip-172-31-10-136 myData]$ ls -lrth
total 4.0K
-rw-rw-r-- 1 ec2-user ec2-user 0 Dec 25 16:40 genericTestFile.txt
[ec2-user@ip-172-31-10-136 myData]$ cat genericTestFile.txt
[ec2-user@ip-172-31-10-136 myData]$ vim genericTestFile.txt
[ec2-user@ip-172-31-10-136 myData]$ cat genericTestFile.txt
Hello World
[ec2-user@ip-172-31-10-136 myData]$
```

Figure 18: Follow all the commands that are highlighted in yellow

```

login as: ec2-user
Authenticating with public key "genericKeyPair"

      _|_  _|_  )
      _|_  ( _|_ /   Amazon Linux 2 AMI
      _|_  _|_  |

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-81-144 ~]$ sudo yum -y install amazon-efs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                     | 3.7 kB    00:00
Resolving Dependencies
--> Running transaction check
--> Package amazon-efs-utils.noarch 0:1.34.1-1.amzn2 will be installed
--> Processing Dependency: stunnel5 for package: amazon-efs-utils-1.34.1-1.amzn2.noarch
--> Running transaction check
--> Package stunnel5.x86_64 0:5.58-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch      Version              Repository           Size
=====
Installing:
amazon-efs-utils        noarch    1.34.1-1.amzn2       amzn2-core           53 k
Installing for dependencies:
stunnel5                x86_64    5.58-1.amzn2.0.1     amzn2-core           165 k
=====

```

Figure 19: Connect the second EC2 via SSH and install the amazon utils using yum command

```

[ec2-user@ip-172-31-81-144 ~]$ sudo mkdir /efs
[ec2-user@ip-172-31-81-144 ~]$ sudo mount -t efs -o tls fs-0c72e075a28313abd:/ /efs
[ec2-user@ip-172-31-81-144 ~]$ cd /efs
[ec2-user@ip-172-31-81-144 efs]$ ls -lrth
total 4.0K
drwxr-xr-x 2 ec2-user root 6.0K Dec 25 16:41 myData
[ec2-user@ip-172-31-81-144 efs]$ cd myData/
[ec2-user@ip-172-31-81-144 myData]$ ls -lrth
total 4.0K
-rw-rw-r-- 1 ec2-user ec2-user 12 Dec 25 16:41 genericTestFile.txt
[ec2-user@ip-172-31-81-144 myData]$ cat genericTestFile.txt
Hello World
[ec2-user@ip-172-31-81-144 myData]$

```

Figure 20: Follow the highlighted commands as given and are similar to last process

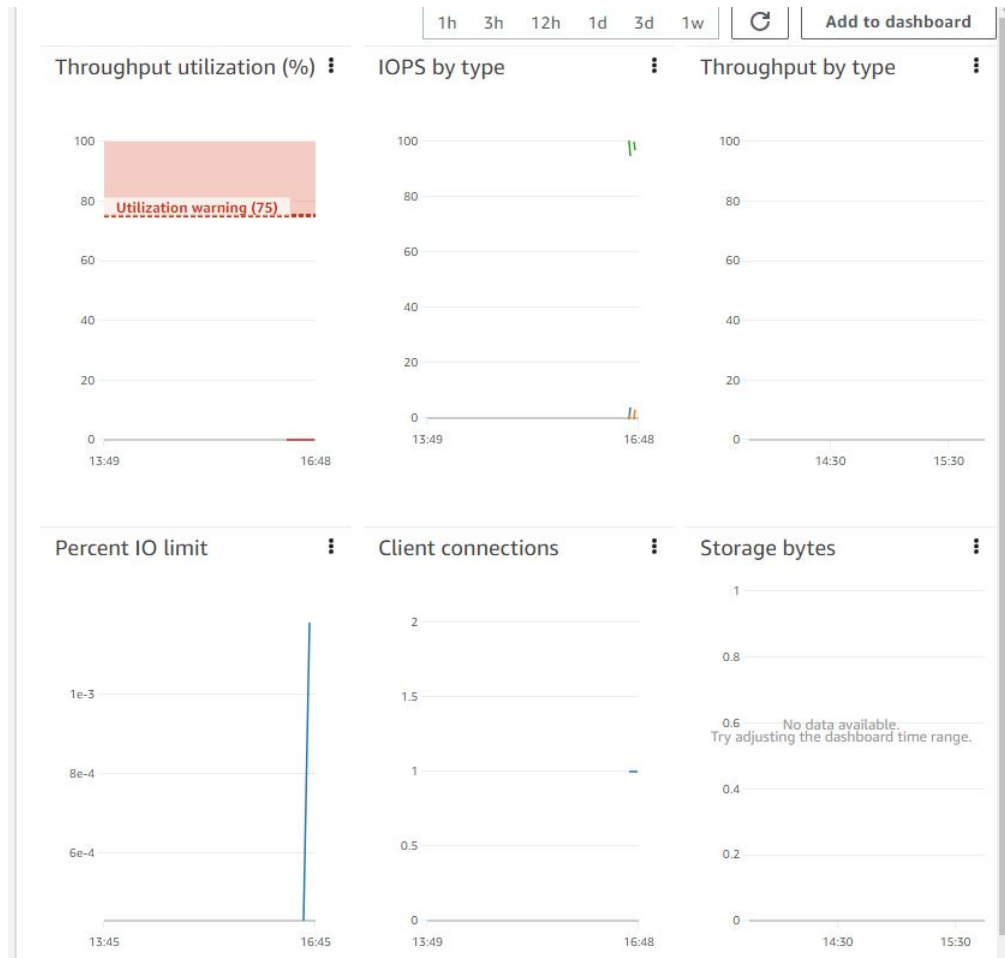


Figure 21: Monitoring the currently used EFS can be seen on its dashboard

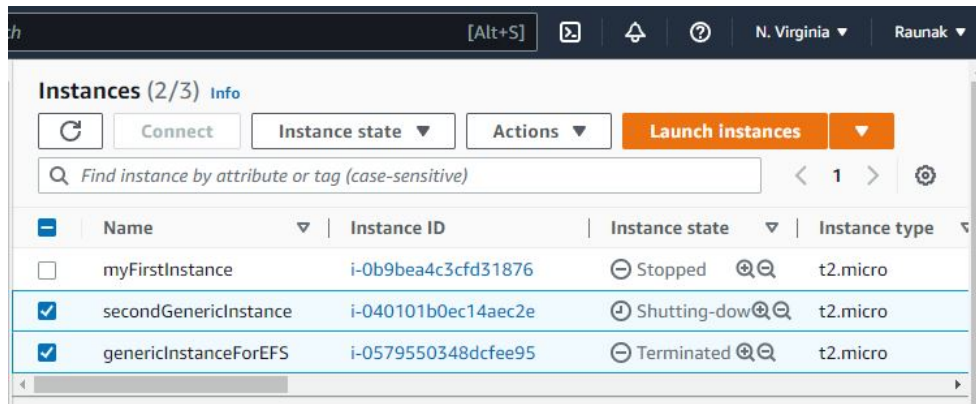


Figure 22: Stopping and terminating both the instances

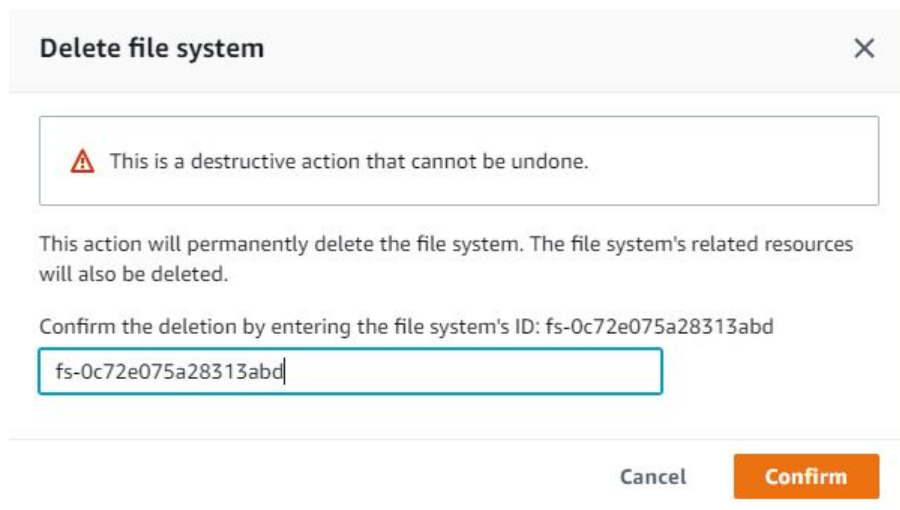


Figure 23: Deletion of the created EFS after successful experimentation

## Conclusion

This experiment is successful implementation of creating an Amazon Elastic File System and connecting it with two separate EC2 instances. The motive of entire experiment was to prove that EFS can be connected simultaneously to different machine unlike EBS which connects to a single instance. The experiment gives comprehensive outlook towards the creation to termination process in a seamless manner.