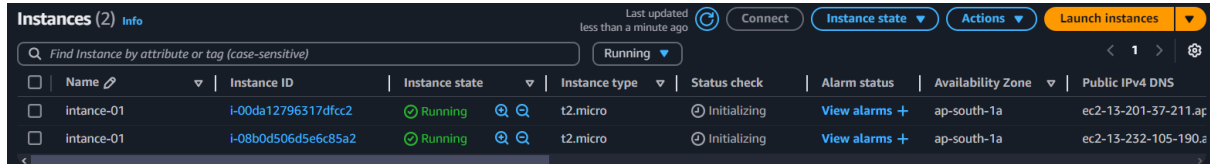


# EFS – Elastic File Systems

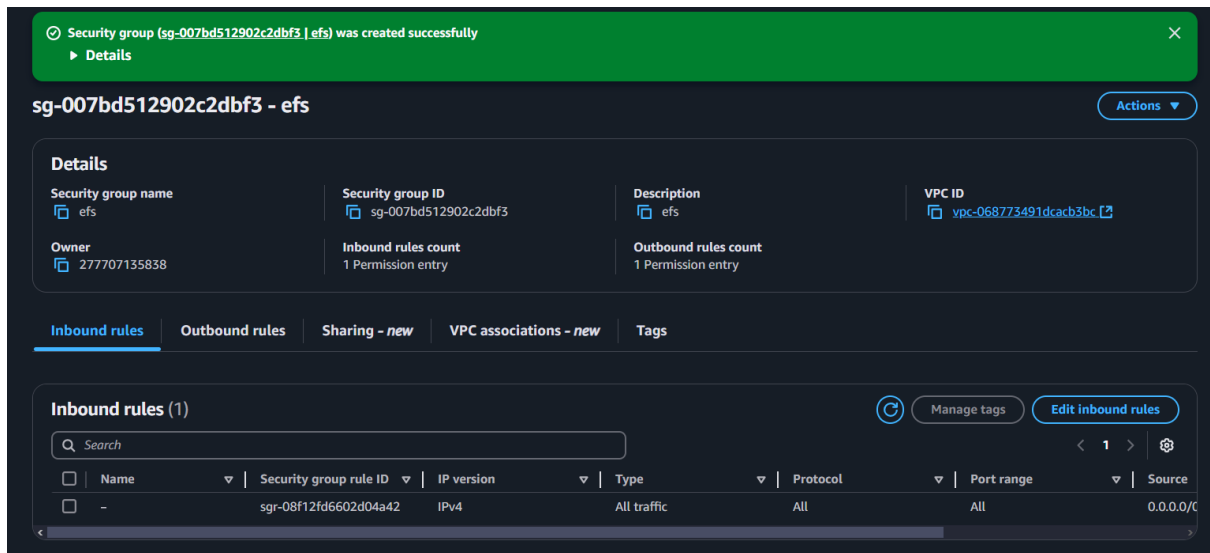
Step-1: Create 2 instance on same availability zone



The screenshot shows the AWS EC2 Instances console. At the top, there's a search bar and a filter set to 'Running'. Below the header, there's a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. Two instances are listed, both in the 'ap-south-1a' availability zone and in a 'Running' state.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
instance-01	i-00da12796317dfcc2	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-13-201-37-211.ap
instance-01	i-08b0d506d5e6c85a2	Running	t2.micro	Initializing	View alarms +	ap-south-1a	ec2-13-232-105-190.a

Step-2: Create a security group



The screenshot shows the AWS Security Groups console. A green notification banner at the top states: 'Security group (sg-007bd512902c2dbf3 | efs) was created successfully'. Below this, the details for the security group 'efs' are shown. It includes the security group name, ID, description, VPC ID, owner, and rule counts. The 'Inbound rules' tab is selected, showing a single rule that allows all traffic from 0.0.0.0/0.

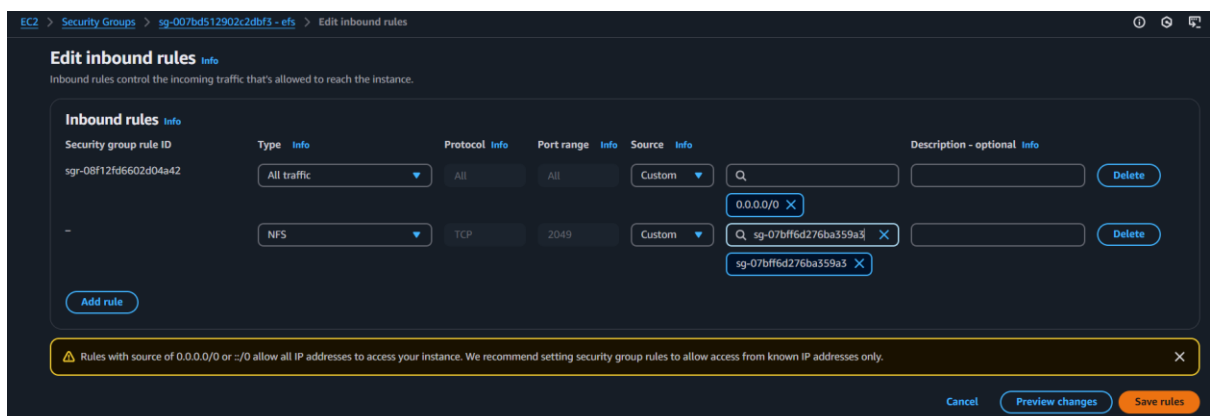
**Details**

- Security group name: efs
- Security group ID: sg-007bd512902c2dbf3
- Description: efs
- VPC ID: vpc-068773491dcacb3bc
- Owner: 277707135838
- Inbound rules count: 1 Permission entry
- Outbound rules count: 1 Permission entry

**Inbound rules (1)**

Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
-	sgr-08f12fd6602d04a42	IPv4	All traffic	All	All	0.0.0.0/0

Step-3: add inbound rules for efs security group with ec2 security group only



The screenshot shows the 'Edit inbound rules' page for the security group 'efs'. It displays the existing rule and two new rules added for the 'ec2' security group. The rules are for NFS traffic on port 2049. A warning message at the bottom states: 'Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.'

**Edit inbound rules**

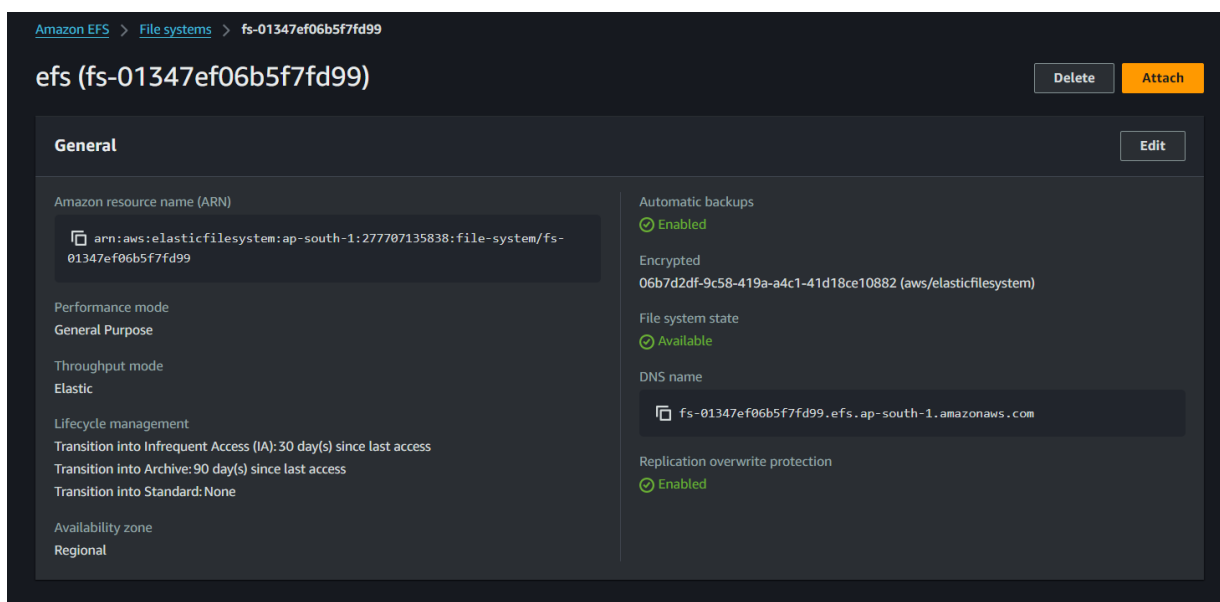
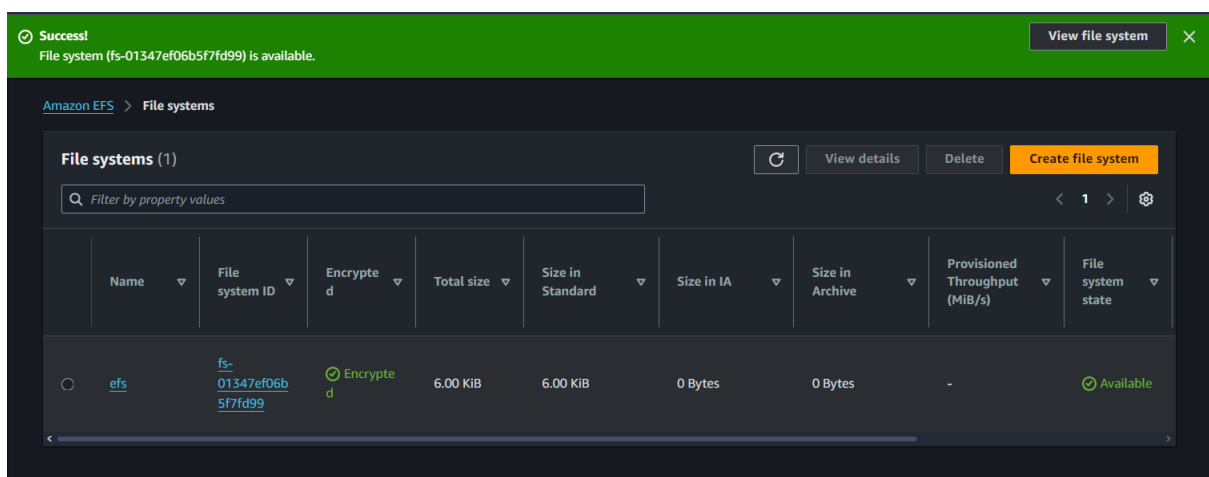
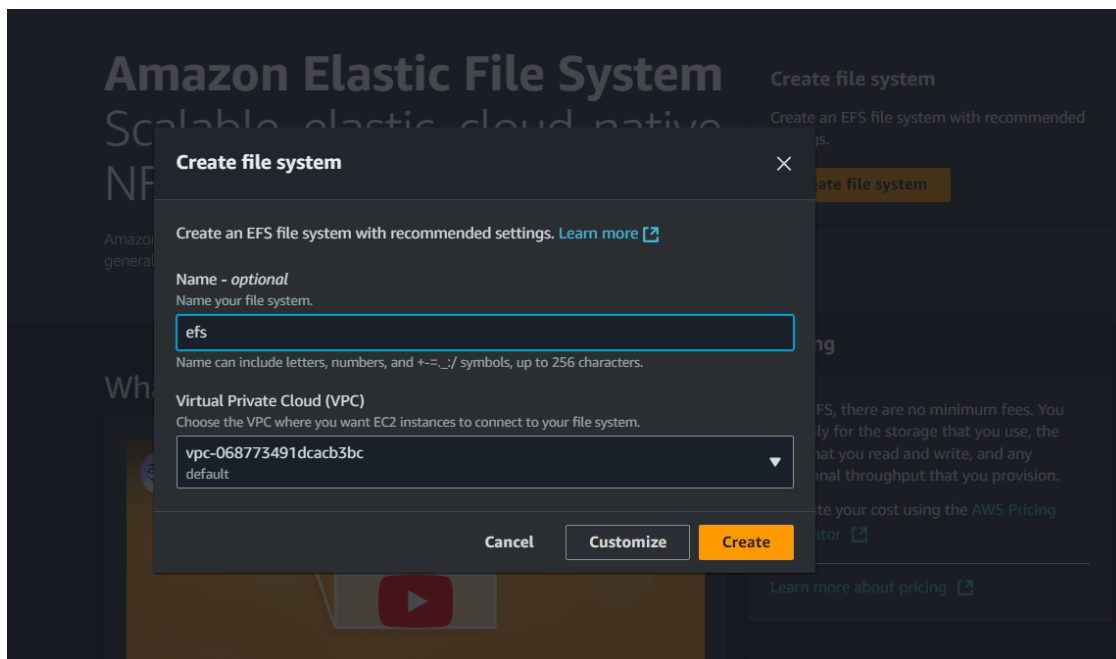
Inbound rules control the incoming traffic that's allowed to reach the instance.

**Inbound rules**

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-08f12fd6602d04a42	All traffic	All	All	Custom	
-	NFS	TCP	2049	Custom	sg-07bfff6d276ba359a3
-	NFS	TCP	2049	Custom	sg-07bfff6d276ba359a3

Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

## Step-4: Create a EFS

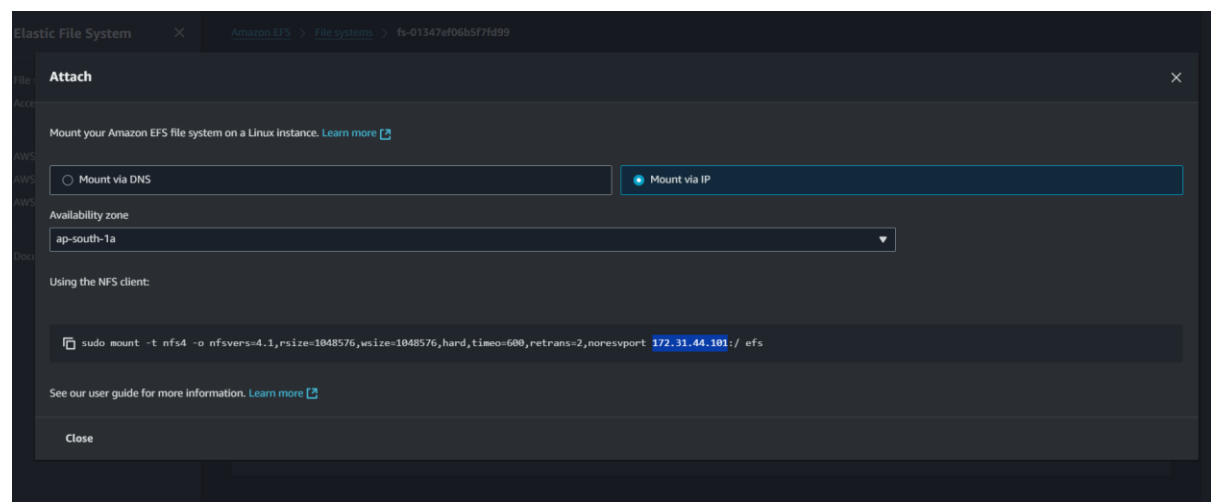


Step-5: Attach the efs in ec2 instance

Connect the instance-01

```
Installed:
telnet-1:0.17-83.amzn2023.0.2.x86_64

Complete!
[root@ip-172-31-44-120 ~]#
```



# telnet efs\_ip port\_no

```
[root@ip-172-31-44-120 ~]# telnet 172.31.44.101 2049
Trying 172.31.44.101...
Connected to 172.31.44.101.
Escape character is '^]'.
^]
telnet> Connection closed.
```

To exit telnet: ctrl + ]

ctrl + D

Step-6: now mount the efs to the ec2 instance

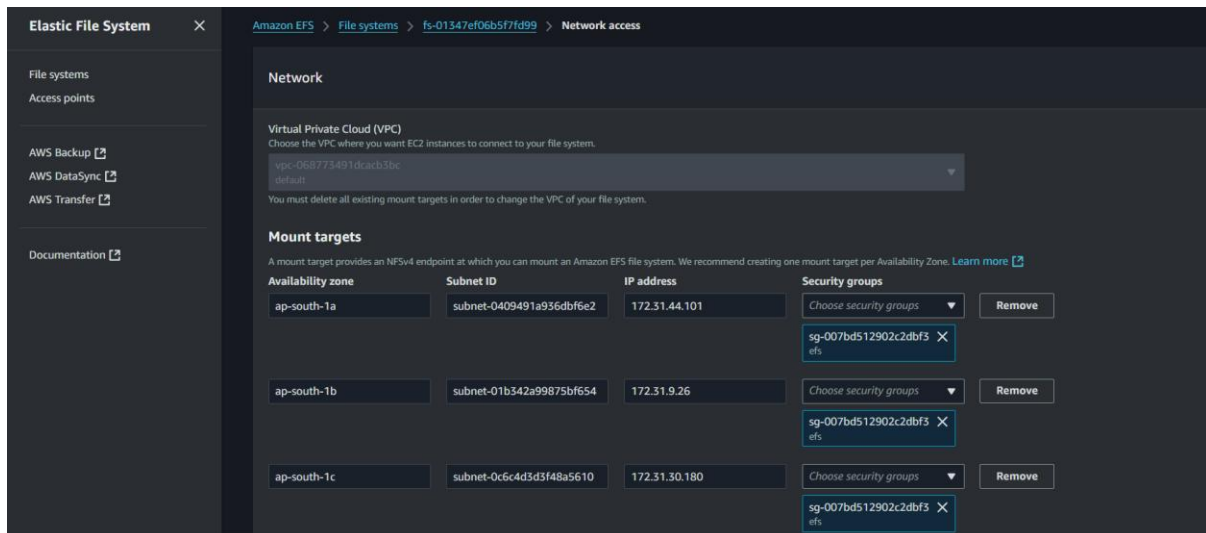
Create a directory (i.e. mkdir -p /share/efs)

```
[root@ip-172-31-44-120 ~]# mkdir -p /share/efs
[root@ip-172-31-44-120 ~]# sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport 172.31.44.101:/ /share/efs
```

```
Using the NFS client:
✓ copied
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport 172.31.44.101:/efs
```

Step-7: repeat step-5,6 in another instance for checking the files shared or not

Step-8: Change the network security group in EFS to efs security group



Step-9: Open the first instance and check in mounted directory and create some files and list it

```
[root@ip-172-31-44-120 ~]# cd /share/efs/
[root@ip-172-31-44-120 efs]# ll
total 0
[root@ip-172-31-44-120 efs]# touch file{1..5}
[root@ip-172-31-44-120 efs]# ls
file1 file2 file3 file4 file5
```

Step-10: Connect in 2<sup>nd</sup> instance and check the mounted directory.

```
[root@ip-172-31-37-43 ~]# cd /share/efs/
[root@ip-172-31-37-43 efs]# ls
file1 file2 file3 file4 file5
```

Step-11: create a new file in 2<sup>nd</sup> instance and check in 1<sup>st</sup> instance

```
[root@ip-172-31-37-43 efs]# touch smaple
[root@ip-172-31-37-43 efs]# ls
file1 file2 file3 file4 file5 smaple
```

```
[root@ip-172-31-44-120 efs]# ls
file1 file2 file3 file4 file5 smaple
```

