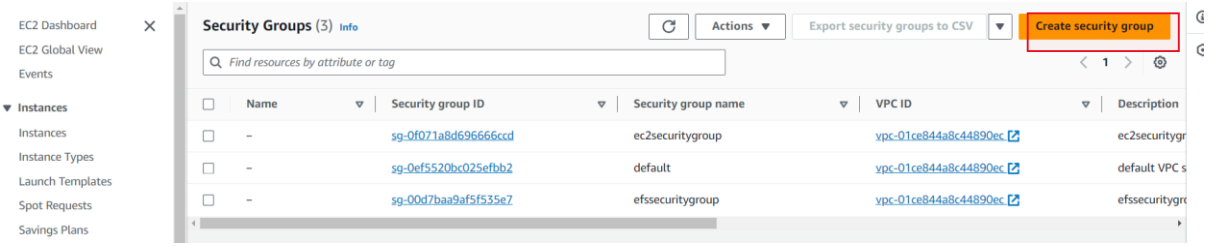
Day 3 - AWS - EBS ,Snapshot (28/06/2024)

Elastic file system (efs):

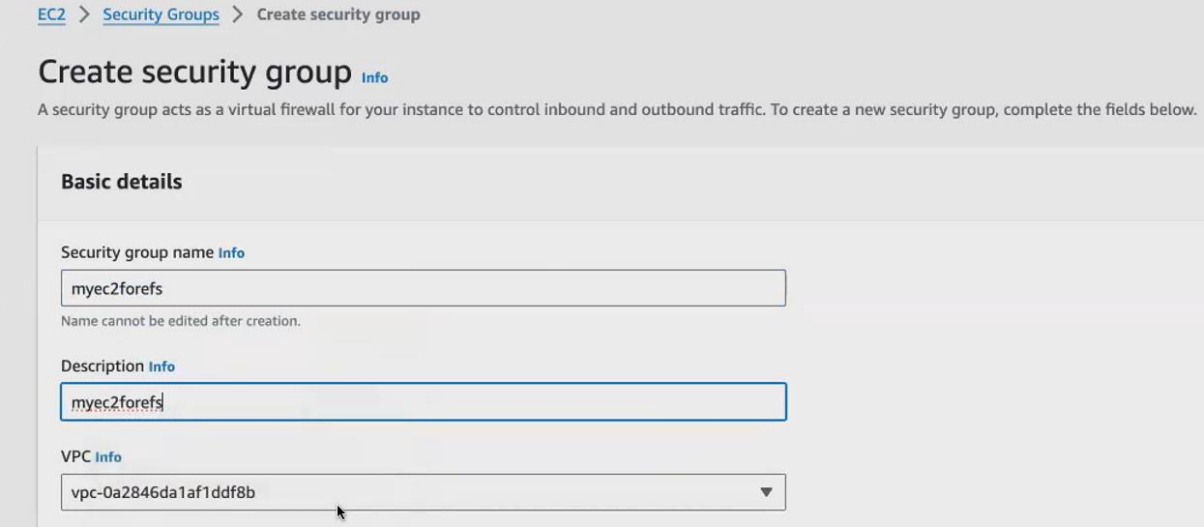
- elastic file system is one of your storage service in aws.
- you can access this Efs with multiple aws resources simultaneously. So you can connect this Effs with multiple Ec 2 missions.
- In EBS scaling not possible .in EFS scaling possible.
- Scaling means we can increase or decrease memory size /vol size.
- if this supports the network file system version 4, that is nfs version 4.1 and Nfs version 4.0 protocol.nfs – protocol.
- how do you create this protocol for Ec, 2. Mission to access your Ec 2. Mission? We need Ssh protocol. So you open up the port Number 22
- wherein for accessing your Effs volume ,they are saying that you need a network file system protocol, which means Nfs pro protocol has to be open.
- So you create a security group before you create before you launch. Efs, you create a security group with your Nfs protocol and you attach this security group to your Efs volume.

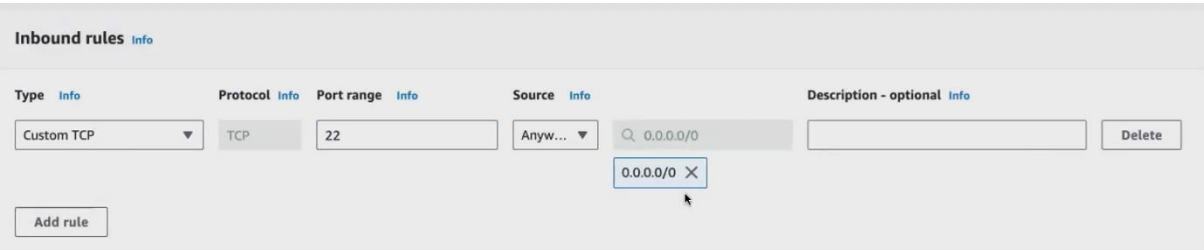
Create security group for ec2:

1. One for security group for ec2 (ec2securitygroup)
2. Another security group for efs (efs securitygroup)



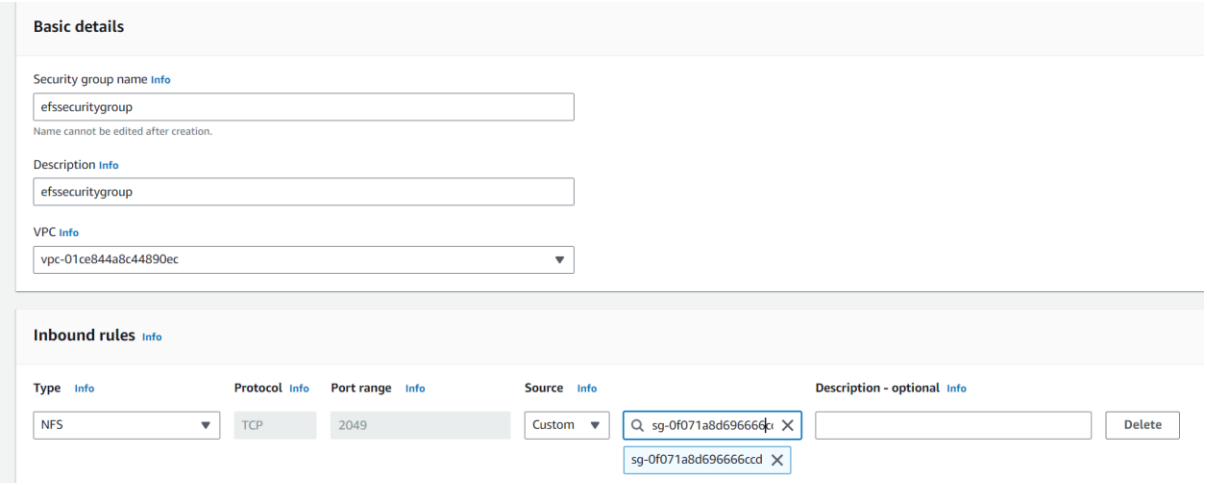
Yasmine o/p screen



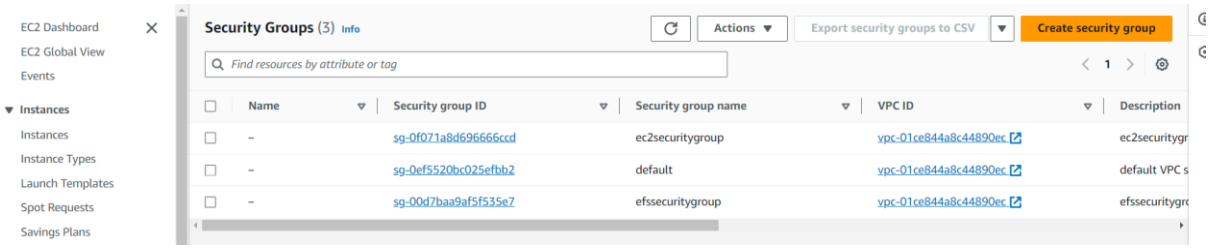


Click create security group

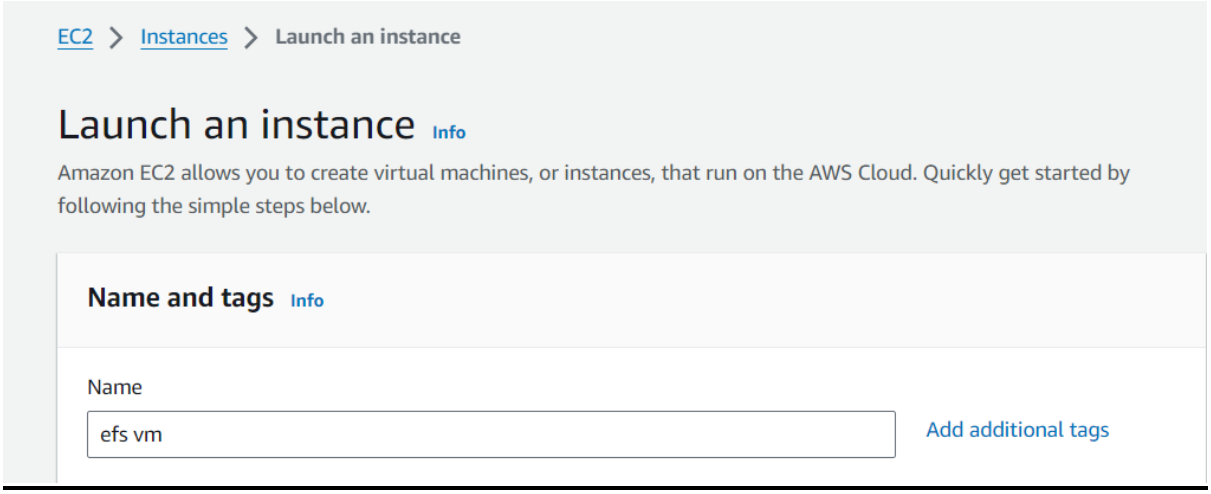
2. **<u>Another security group for efs</u>**

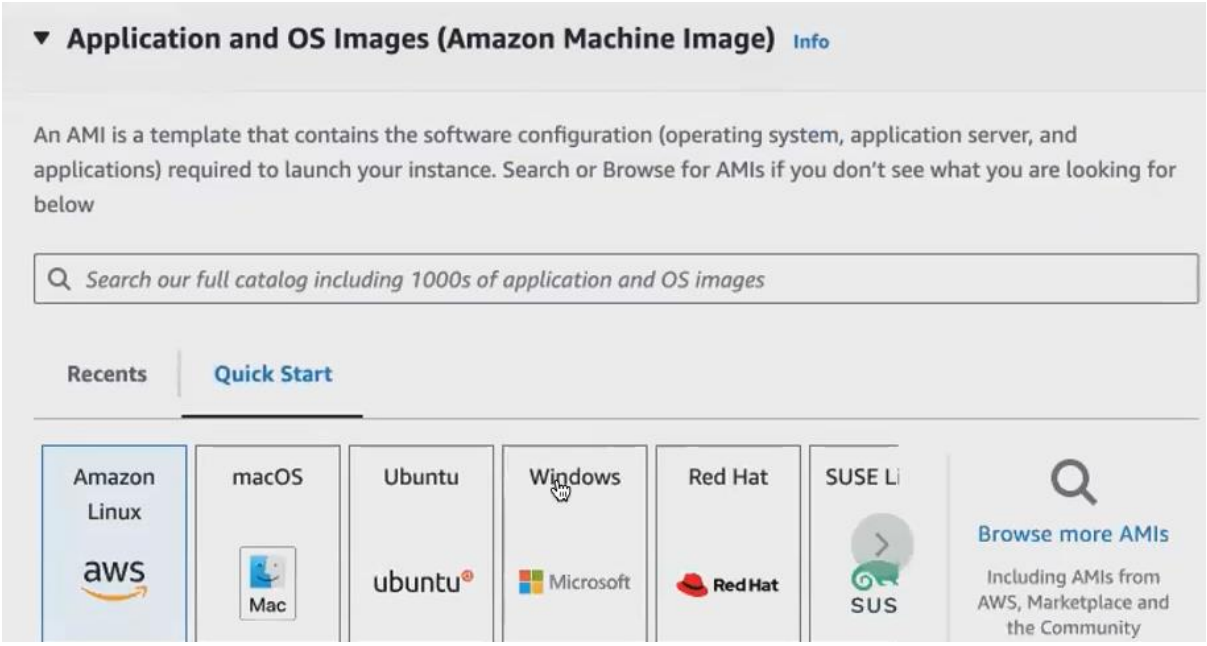


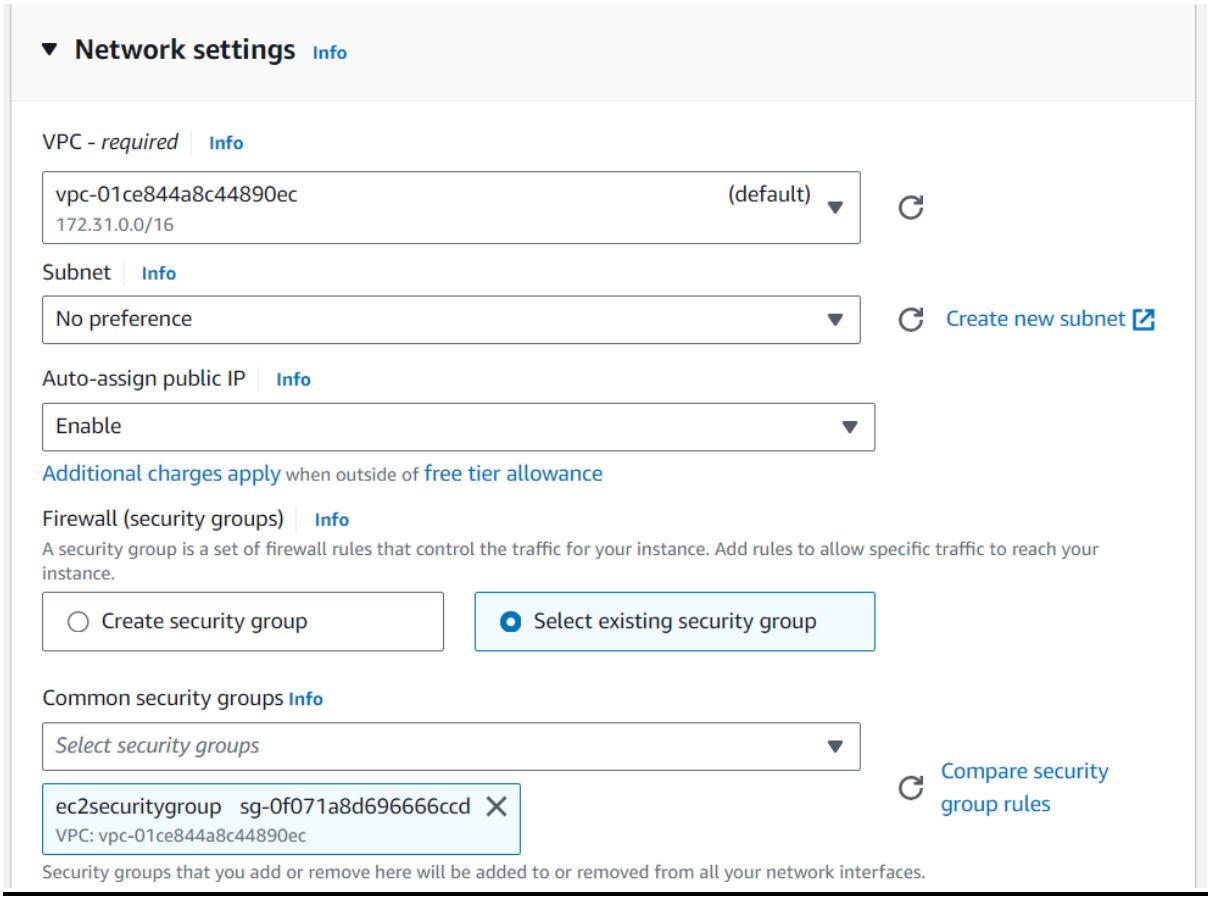
Efs run in NFS protocol .click myec2forfs (s.g name for ec2)



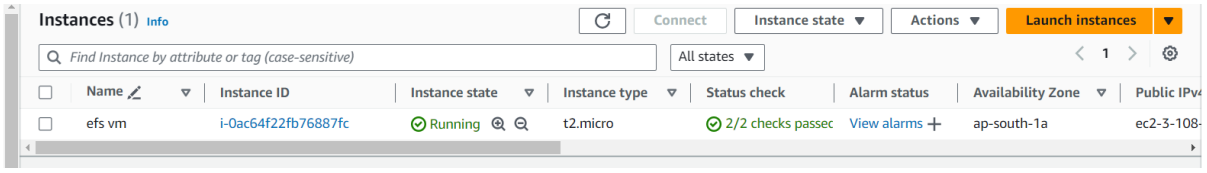
To launch new instance:

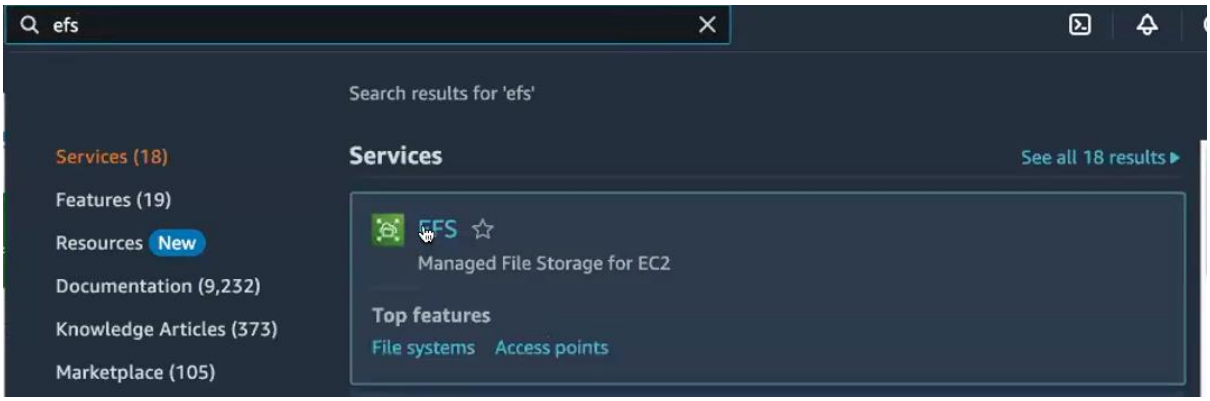




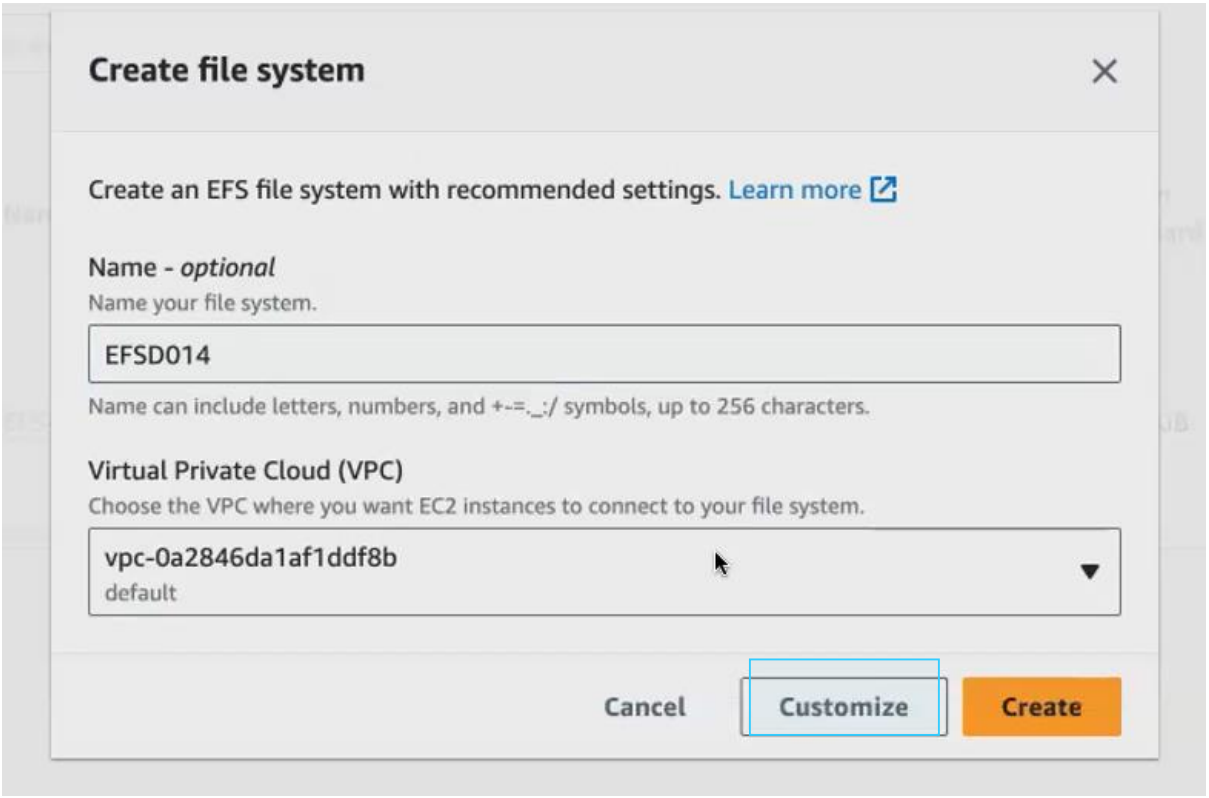


Click launch instance



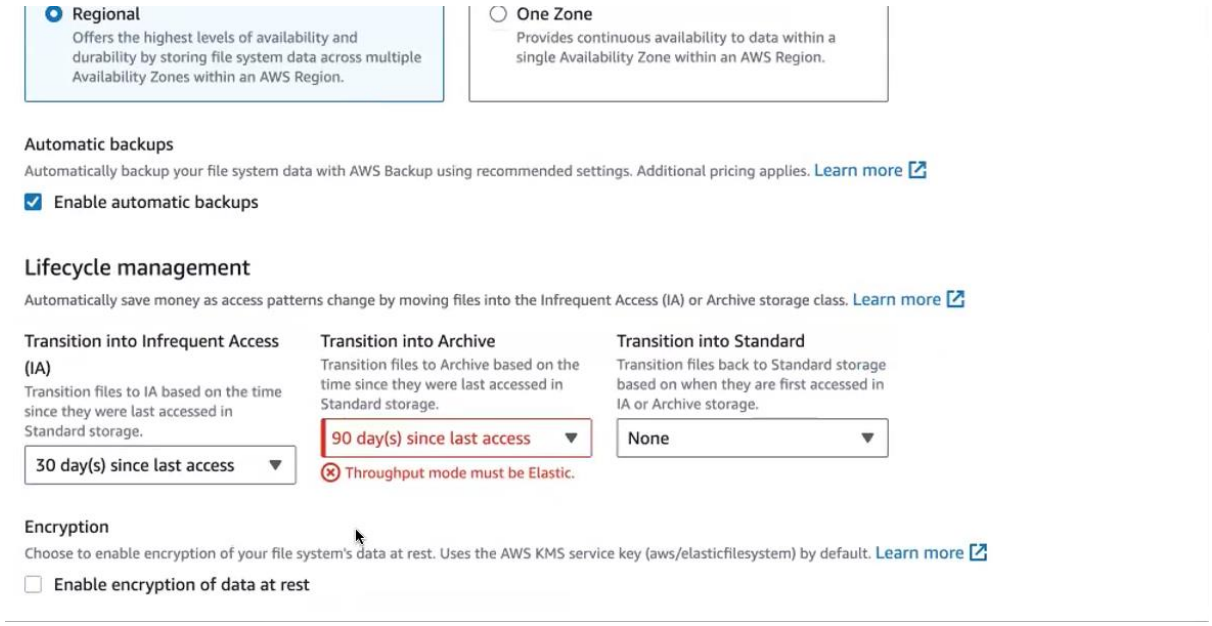


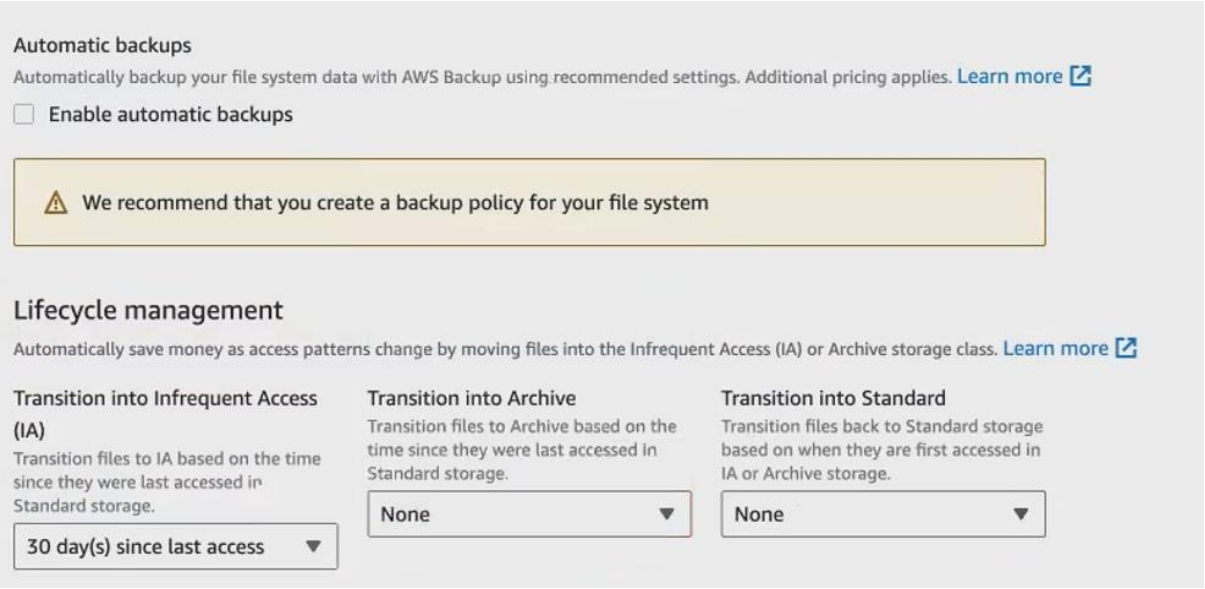

Elastic File System
File systems
Access points
AWS Backup
AWS DataSync
AWS Transfer
Documentation
Amazon Elastic File System
Scalable, elastic, cloud-native NFS file system
Amazon Elastic File System (Amazon EFS) provides a simple, scalable, elastic file system for general purpose workloads for use with AWS Cloud services and on-premises resources.
Create file system
Create an EFS file system with recommended settings.
Create file system


Create file system
Create an EFS file system with recommended settings. Learn more
Name - optional
Name your file system.
EFSD014
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.
vpc-0a2846da1af1ddf8b
default
Cancel
Customize
Create

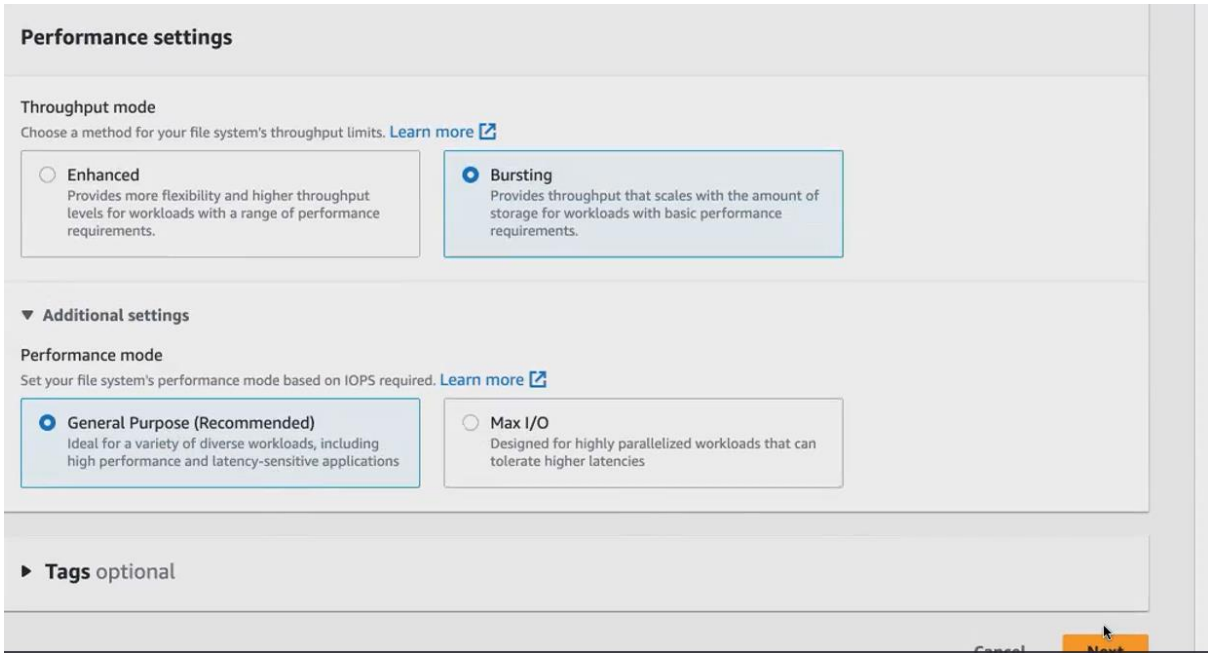
Click customize ,below settings available .total 4 steps

Step 1:

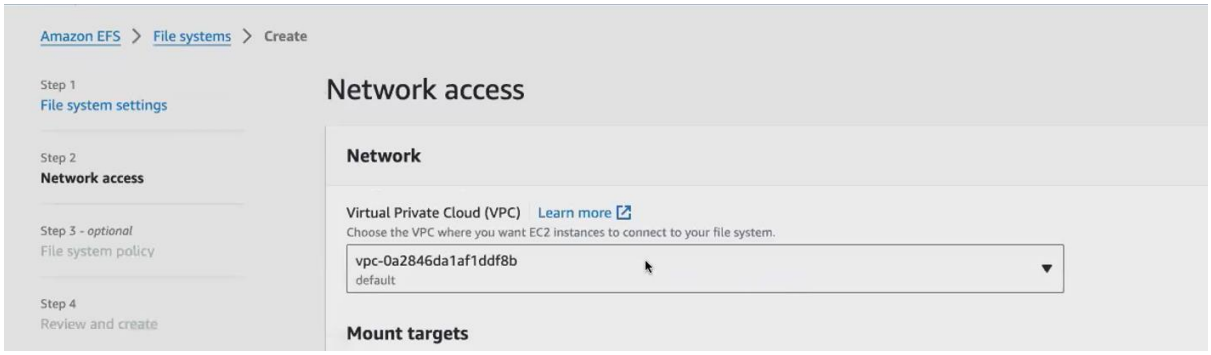




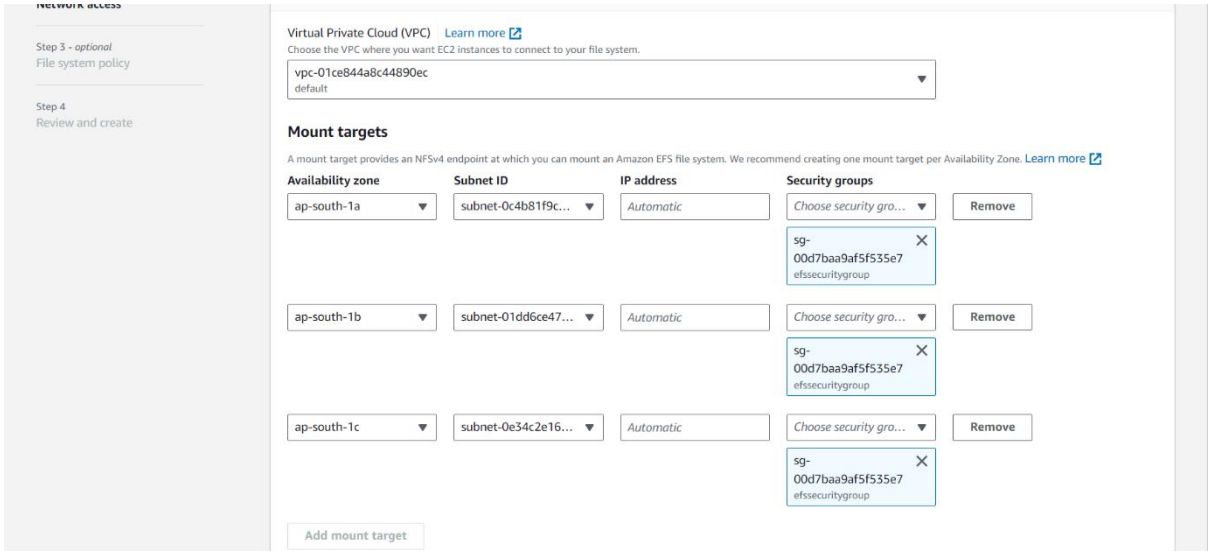
Bursting is free so we select this



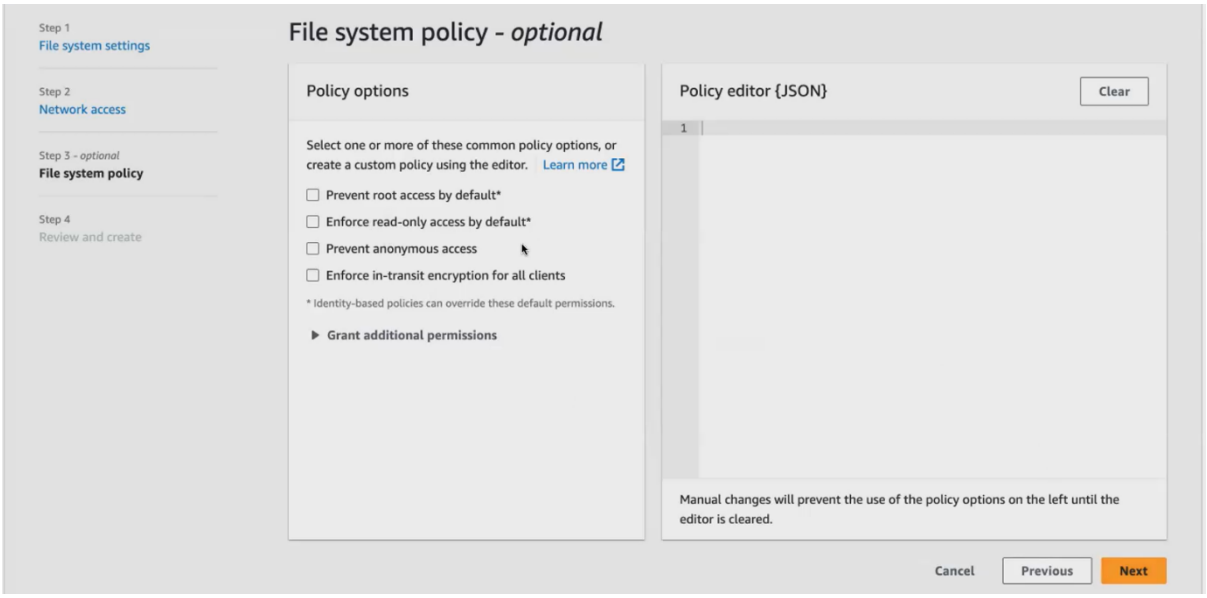
Step 2:



Delete all default security group add our created security group. Click next

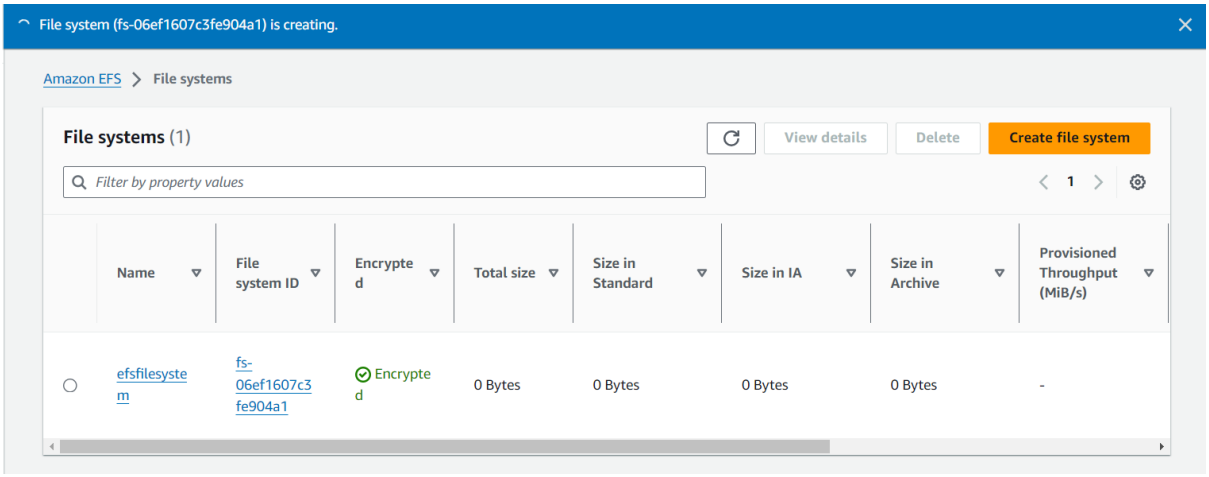


Step 3:

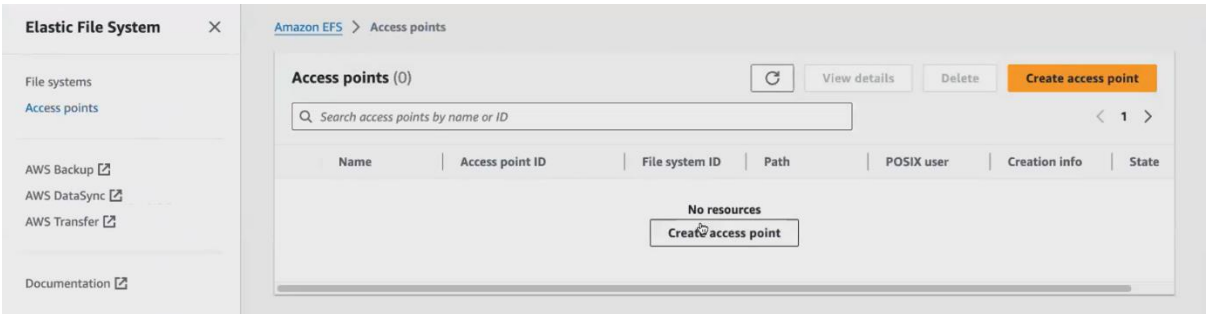


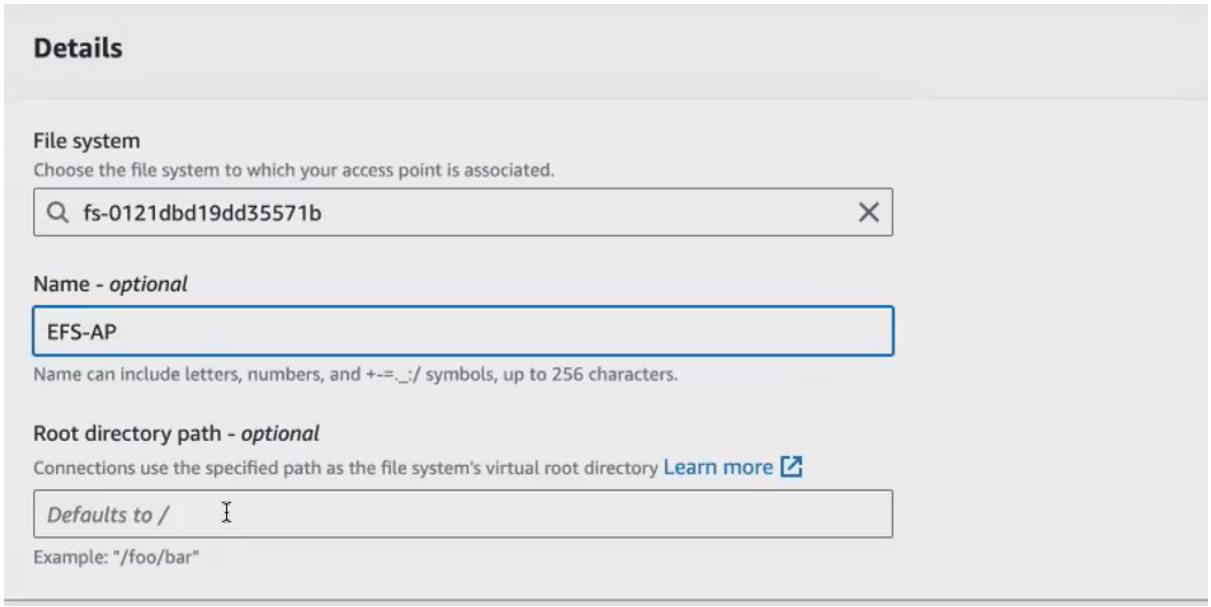
Step 4:

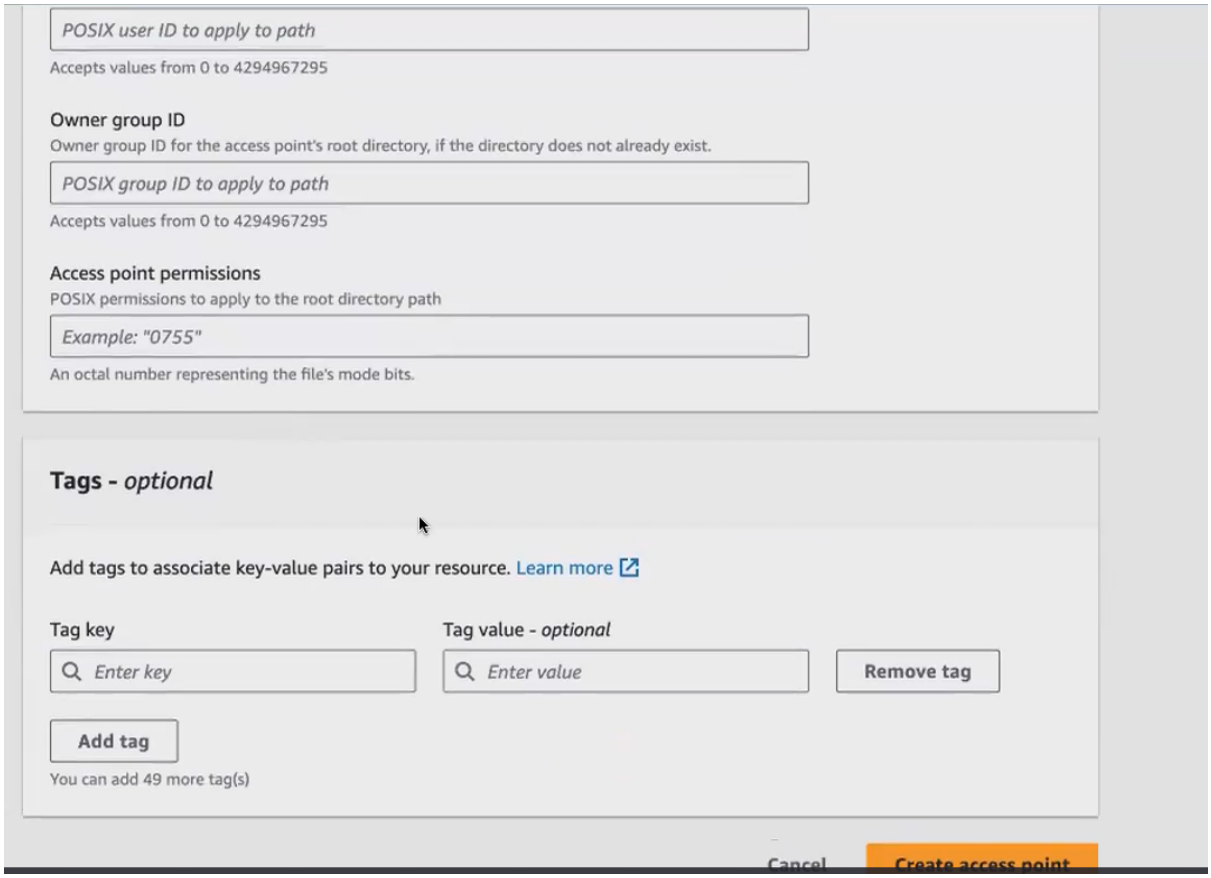
No change ..click create f.s.file system created

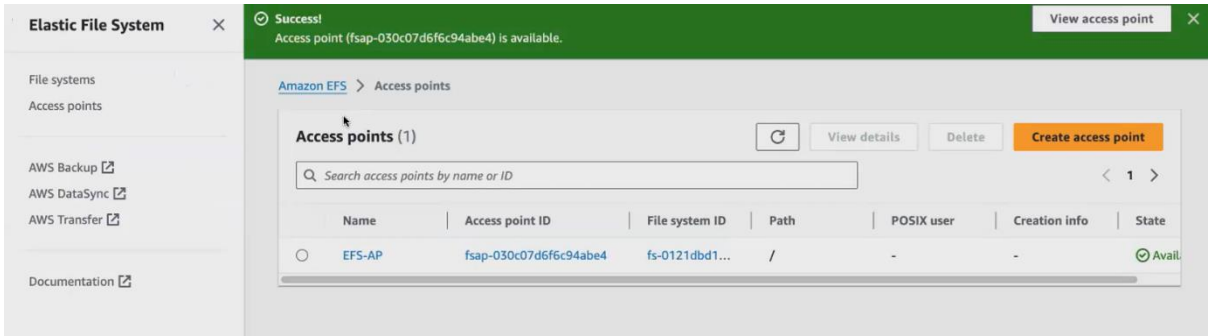


Now f.s created .how to attach this with ec2. 1st create <u>access point</u>




Details
File system
Choose the file system to which your access point is associated.
fs-0121dbd19dd35571b
Name - optional
EFS-AP
Name can include letters, numbers, and +-=._:/ symbols, up to 256 characters.
Root directory path - optional
Connections use the specified path as the file system's virtual root directory Learn more
Defaults to /
Example: "/foo/bar"


POSIX user ID to apply to path
Accepts values from 0 to 4294967295
Owner group ID
Owner group ID for the access point's root directory, if the directory does not already exist.
POSIX group ID to apply to path
Accepts values from 0 to 4294967295
Access point permissions
POSIX permissions to apply to the root directory path
Example: "0755"
An octal number representing the file's mode bits.
Tags - optional
Add tags to associate key-value pairs to your resource. Learn more
Tag key
Enter key
Tag value - optional
Enter value
Remove tag
Add tag
You can add 49 more tag(s)
Cancel
Create access point


Elastic File System
File systems
Access points
AWS Backup
AWS DataSync
AWS Transfer
Documentation
Success!
Access point (fsap-030c07d6f6c94abe4) is available.
View access point
Amazon EFS > Access points
Access points (1)
View details
Delete
Create access point
Search access points by name or ID
Name
Access point ID
File system ID
Path
POSIX user
Creation info
State
EFS-AP
fsap-030c07d6f6c94abe4
fs-0121dbd1...
/
-
-
Avail

Access point created .then go to ec2 attach that efs.click ec2 →connect that ec2

This is amazon linux

https://docs.aws.amazon.com/efs/latest/ug/installing-amazon-efs-utils.html

installation step for efs

```
       _/m/'
[ec2-user@ip-172-31-39-25 ~]$ sudo yum update
Last metadata expiration check: 0:12:04 ago on Fri Jul  5 14:44:03 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-39-25 ~]$
```

That cmd copy from installation docu.link is above

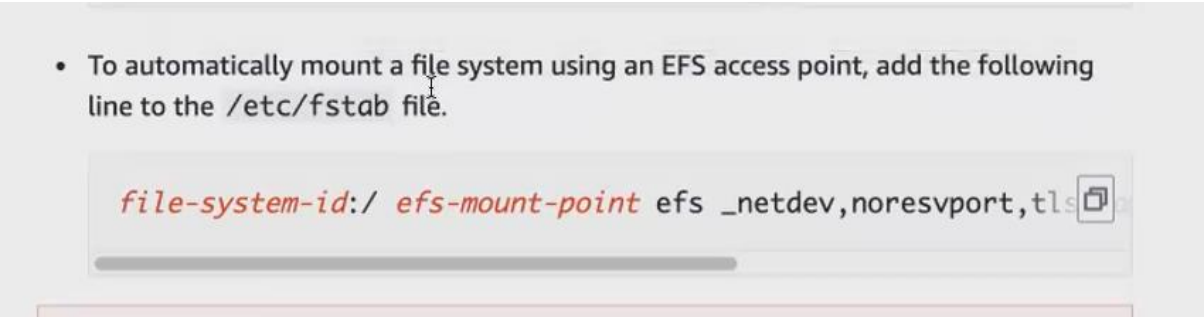
```
[ec2-user@ip-172-31-39-25 ~]$ sudo yum update
Last metadata expiration check: 0:12:04 ago on Fri Jul  5 14:44:03 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-39-25 ~]$ sudo yum install -y amazon-efs-utils
Last metadata expiration check: 0:13:48 ago on Fri Jul  5 14:44:03 2024.
Dependencies resolved.
================================================================================
 Package                    Architecture                Version
================================================================================
Installing:
 amazon-efs-utils           x86_64                      2.0.2-1.amzn2023
Installing dependencies:
 stunnel                    x86_64                      5.58-1.amzn2023.0.2
```

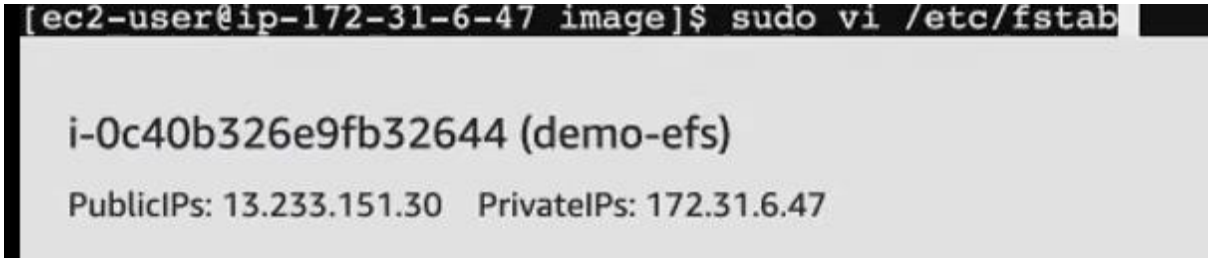
Efs connected with ec2 via access point

1. **Create mount point then attach it to efs volume**
2. **Copy that cmd in vi editor**

https://docs.aws.amazon.com/efs/latest/ug/automount-with-efs-mount-helper.html

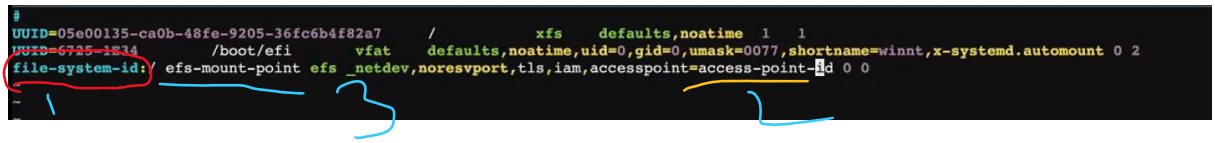
below cmd available in that link



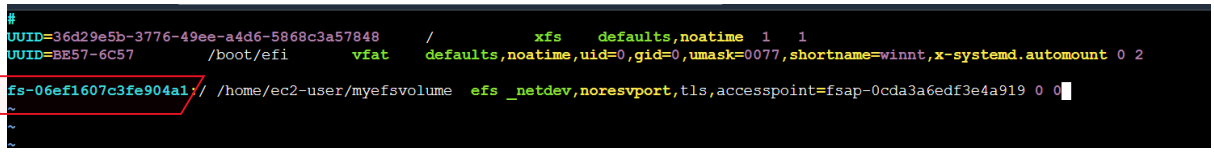


Get file system id and access point that we created .give init.give mountpoint also

1. **file system id**

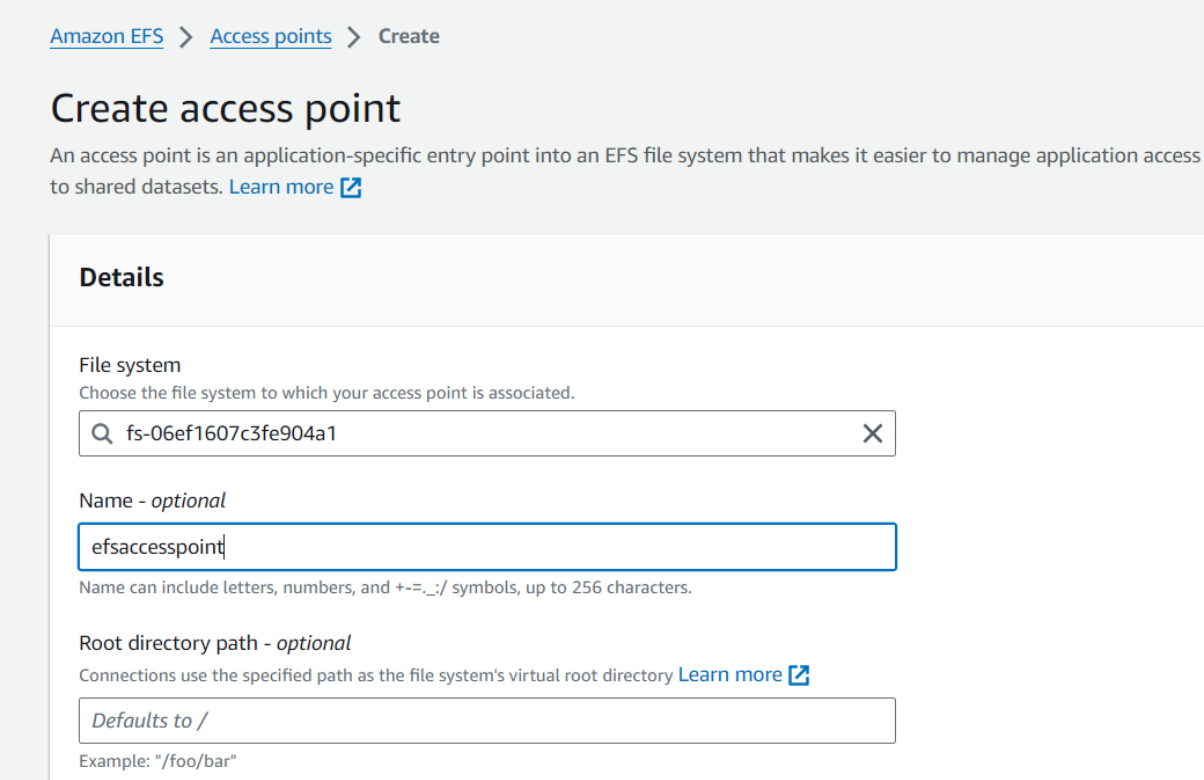


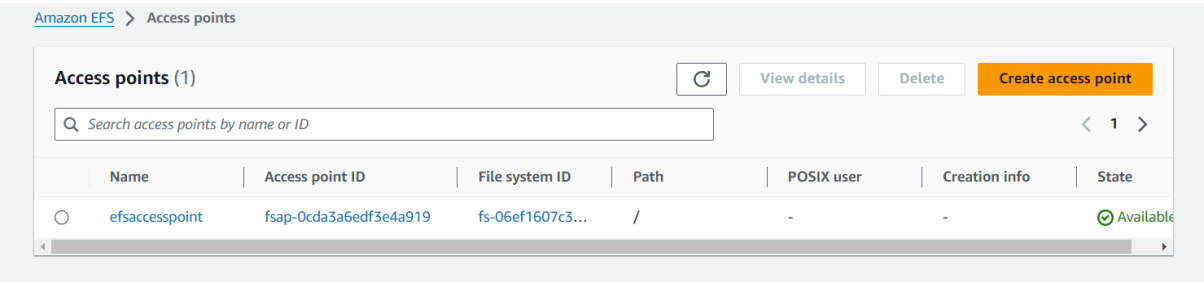
Search efs → getinto that → already created file system→file system id available .copy and paste it in vi editor

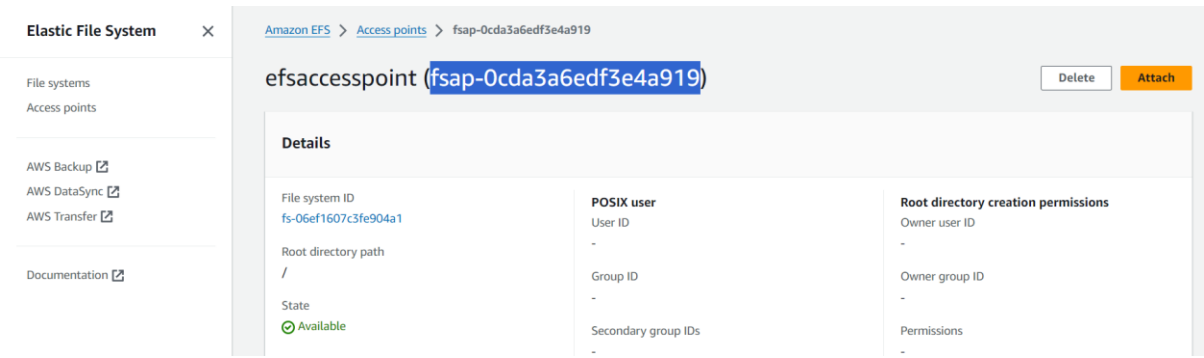


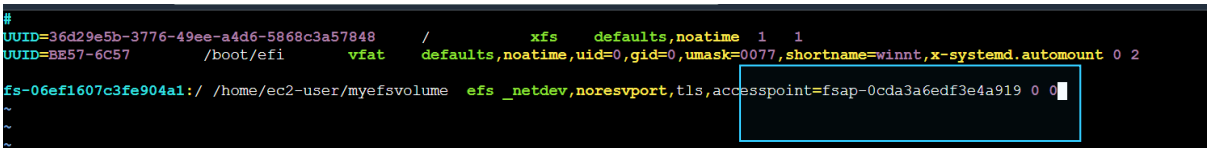
2. **access point id:**

click create access point .give name and all


Amazon EFS > Access points > Create
Create access point
An access point is an application-specific entry point into an EFS file system that makes it easier to manage application access to shared datasets. Learn more
Details
File system
Choose the file system to which your access point is associated.
fs-06ef1607c3fe904a1
Name - optional
efsaccesspoint
Name can include letters, numbers, and +-=._:/ symbols, up to 256 characters.
Root directory path - optional
Connections use the specified path as the file system's virtual root directory Learn more
Defaults to /
Example: "/foo/bar"


Amazon EFS > Access points
Access points (1)
View details
Delete
Create access point
Search access points by name or ID
1
Name
Access point ID
File system ID
Path
POSIX user
Creation info
State
efsaccesspoint
fsap-0cda3a6edf3e4a919
fs-06ef1607c3...
/
-
-
Available


Elastic File System
File systems
Access points
AWS Backup
AWS DataSync
AWS Transfer
Documentation
Amazon EFS > Access points > fsap-0cda3a6edf3e4a919
efsaccesspoint (fsap-0cda3a6edf3e4a919)
Delete
Attach
Details
File system ID
fs-06ef1607c3fe904a1
Root directory path
/
State
Available
POSIX user
User ID
-
Group ID
-
Secondary group IDs
-
Root directory creation permissions
Owner user ID
-
Owner group ID
-
Permissions
-


#
UUID=36d29e5b-3776-49ee-a4d6-5868c3a57848 / xfs defaults,noatime 1 1
UUID=BE57-6C57 /boot/efi vfat defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
fs-06ef1607c3fe904a1:/ /home/ec2-user/myefsvolume efs _netdev,noresvport,tls,accesspoint=fsap-0cda3a6edf3e4a919 0 0

3. **mountpoint: duplicate the tab check the m.p (/home /)**

```
       _/m/'
Last login: Fri Jul  5 14:55:49 2024 from 13.233.177.4
[ec2-user@ip-172-31-39-25 ~]$ mkdir myefsvolume
[ec2-user@ip-172-31-39-25 ~]$ cd myefsvolume/
[ec2-user@ip-172-31-39-25 myefsvolume]$ pwd
/home/ec2-user/myefsvolume
[ec2-user@ip-172-31-39-25 myefsvolume]$ ^C
[ec2-user@ip-172-31-39-25 myefsvolume]$
```

Remove iam

```
#
UUID=05e00135-ca0b-48fe-9205-36fc6b4f82a7     /           xfs    defaults,noatime  1   1
UUID=6725-1E34        /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
fs-0121dbd19dd35571b:/ /home/ec2-user/image efs _netdev,noresvport,tls,iam,accesspoint=fsap-030c07d6f6c94abe4 0 0
~
```

```
#
UUID=05e00135-ca0b-48fe-9205-36fc6b4f82a7     /           xfs    defaults,noatime  1   1
UUID=6725-1E34        /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
fs-0121dbd19dd35571b: /home/ec2-user/image efs _netdev,noresvport,tls,accesspoint=fsap-030c07d6f6c94abe4 0 0
```

My output file:

```
#
UUID=36d29e5b-3776-49ee-a4d6-5868c3a57848     /           xfs    defaults,noatime  1   1
UUID=BE57-6C57        /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2

fs-06ef1607c3fe904a1:/ efs-mount-point efs _netdev,noresvport,tls,accesspoint=fsap-0cda3a6edf3e4a919 0 0
~
```

```
#
UUID=36d29e5b-3776-49ee-a4d6-5868c3a57848     /           xfs    defaults,noatime  1   1
UUID=BE57-6C57        /boot/efi       vfat    defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2

fs-06ef1607c3fe904a1:/ /home/ec2-user/myefsvolume efs _netdev,noresvport,tls,accesspoint=fsap-0cda3a6edf3e4a919 0 0
~
~
~
```

```
[ec2-user@ip-172-31-39-25 ~]$ sudo mount -fav
/                          : ignored
/boot/efi                  : already mounted
/home/ec2-user/myefsvolume: successfully mounted
[ec2-user@ip-172-31-39-25 ~]$
```

Df-h cmd used for check the what are all drive attached efs vol mounted in that particular path