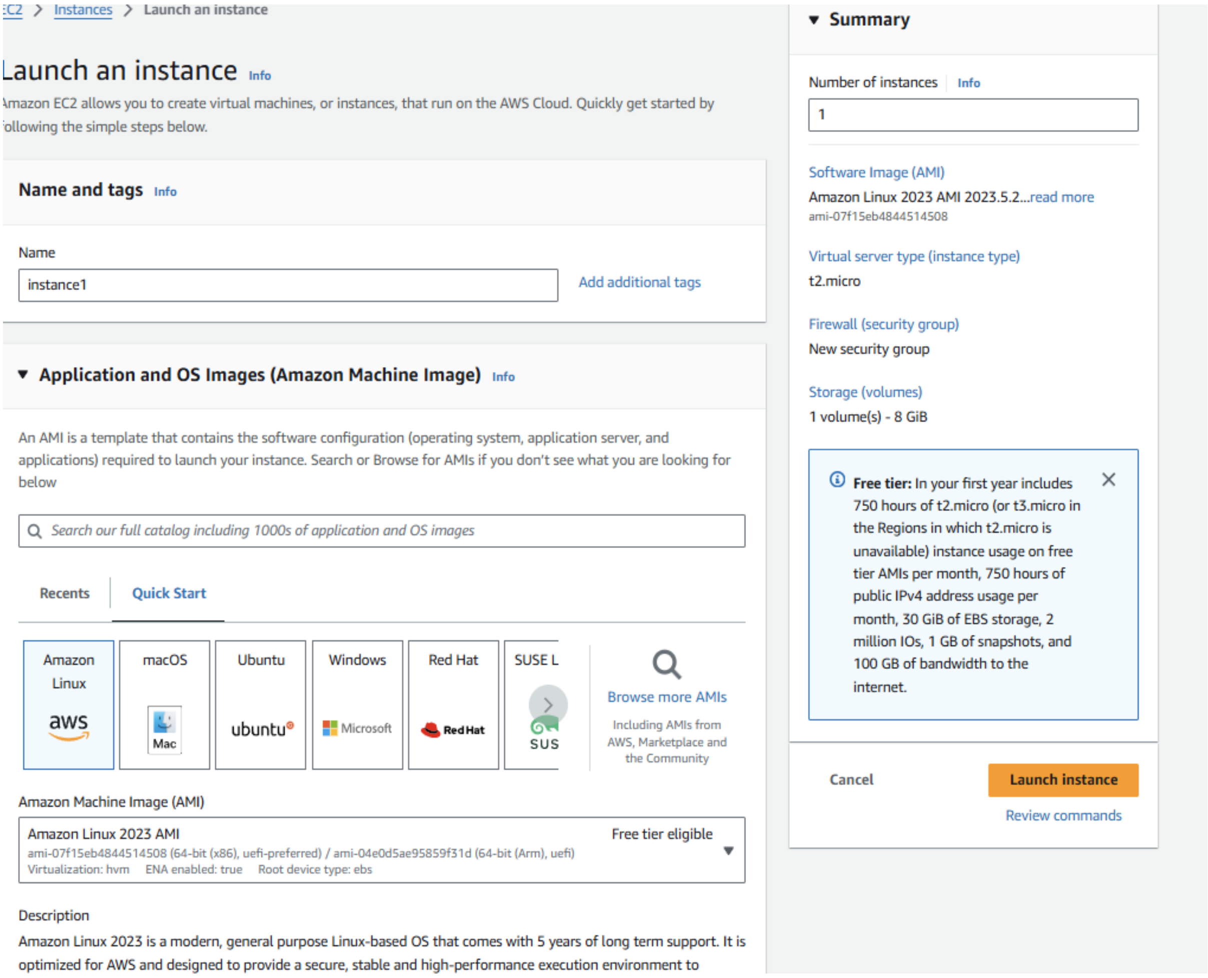
ELASTIC BLOCK STORAGE

Amazon Elastic Block Store (EBS) is Amazon's block-level storage solution used with the **EC2** cloud service to store **persistent data**. This means that the data is kept on the AWS EBS servers even when the EC2 instances are shut down. EBS provides scalable, high-performance block storage resources that you can attach to Amazon EC2 instances

Storage Volume
EC2 Instance

Create EBS and attach EBS to 3 different instances:

Step1: Go AWS console and create an EC2 instance. I was named as instance 1 and created in ap-northeast-2a availability zone in Asia pacific(seoul) region as shown in below.



:C2 > Instances > Launch an instance

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: instance1

Add additional tags

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI — Free tier eligible
ami-07f15eb4844514508 (64-bit (x86), uefi-preferred) / ami-04e0d5ae95859f31d (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to

▼ Summary

Number of instances Info: 1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...read more
ami-07f15eb4844514508

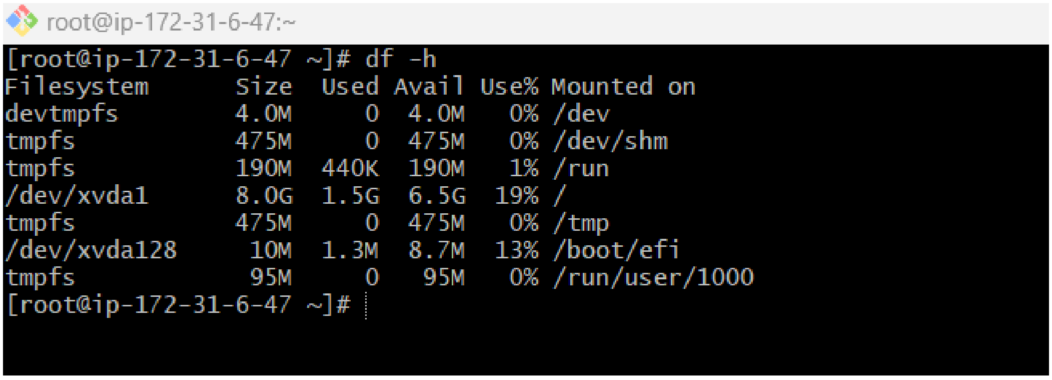
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

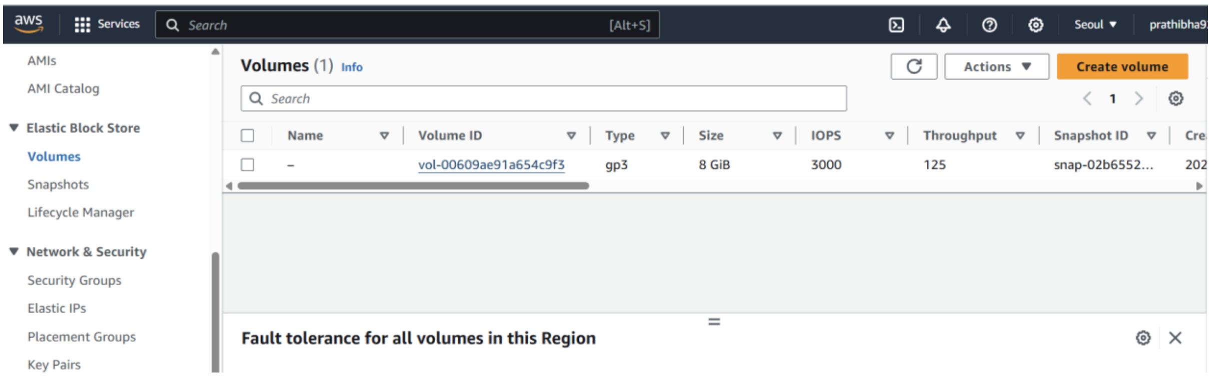
Once created the instance copy the ssh command and connect with that in git bash to the server. After that change to the root user with sudo -i command and then check the disk free space with the command df -h as shown in below. In AWS EC2 instance have 8GB default storage.



root@ip-172-31-6-47:~

```
[root@ip-172-31-6-47 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           475M     0  475M   0% /dev/shm
tmpfs           190M  440K  190M   1% /run
/dev/xvda1      8.0G  1.5G  6.5G  19% /
tmpfs           475M     0  475M   0% /tmp
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs            95M     0   95M   0% /run/user/1000
[root@ip-172-31-6-47 ~]#
```

Step 2: Go to Elastic block store and click on volumes as shown in below.

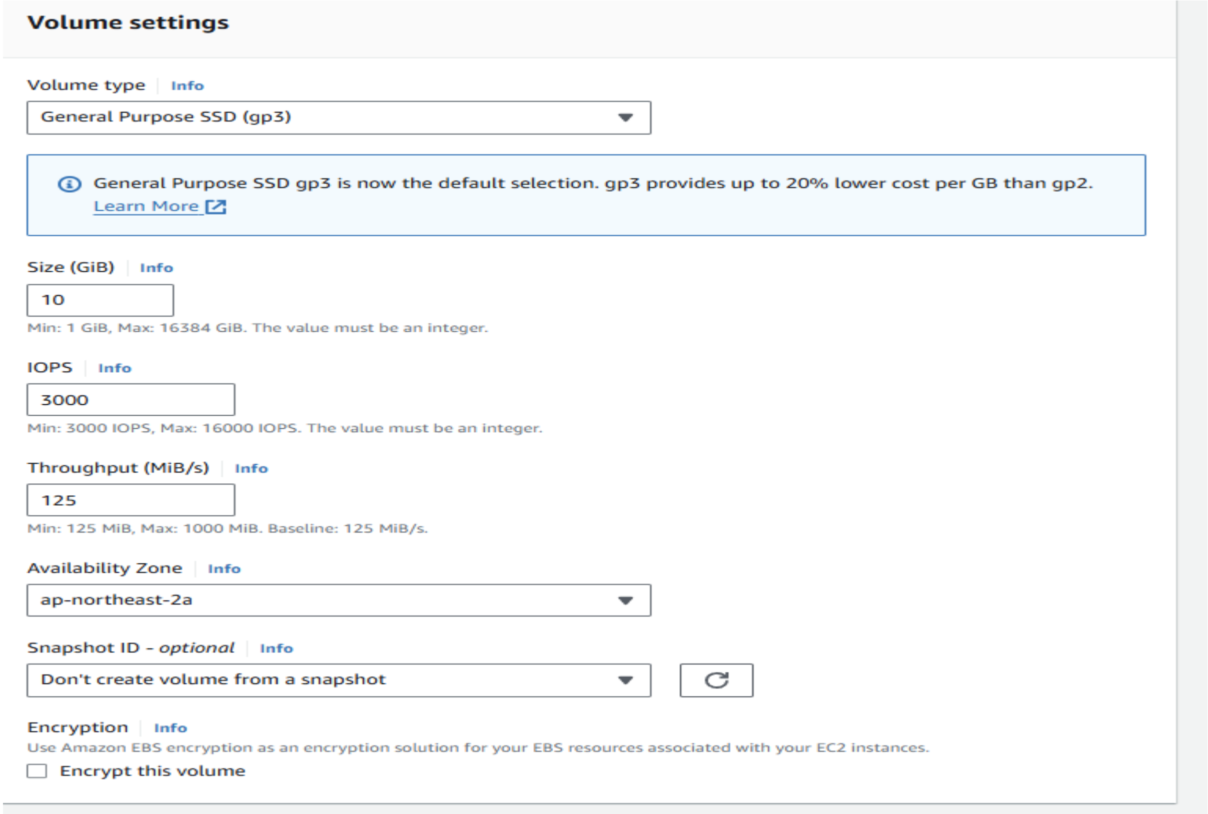


Volumes (1) Info

□	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Cre
□	–	vol-00609ae91a654c9f3	gp3	8 GiB	3000	125	snap-02b6552...	202

Fault tolerance for all volumes in this Region

Click on create volume and configure the settings as shown in below diagram.



Volume settings

Volume type Info

General Purpose SSD (gp3)

General Purpose SSD gp3 is now the default selection. gp3 provides up to 20% lower cost per GB than gp2. Learn More

Size (GiB) Info

10

Min: 1 GiB, Max: 16384 GiB. The value must be an integer.

IOPS Info

3000

Min: 3000 IOPS, Max: 16000 IOPS. The value must be an integer.

Throughput (MiB/s) Info

125

Min: 125 MiB, Max: 1000 MiB. Baseline: 125 MiB/s.

Availability Zone Info

ap-northeast-2a

Snapshot ID - *optional* Info

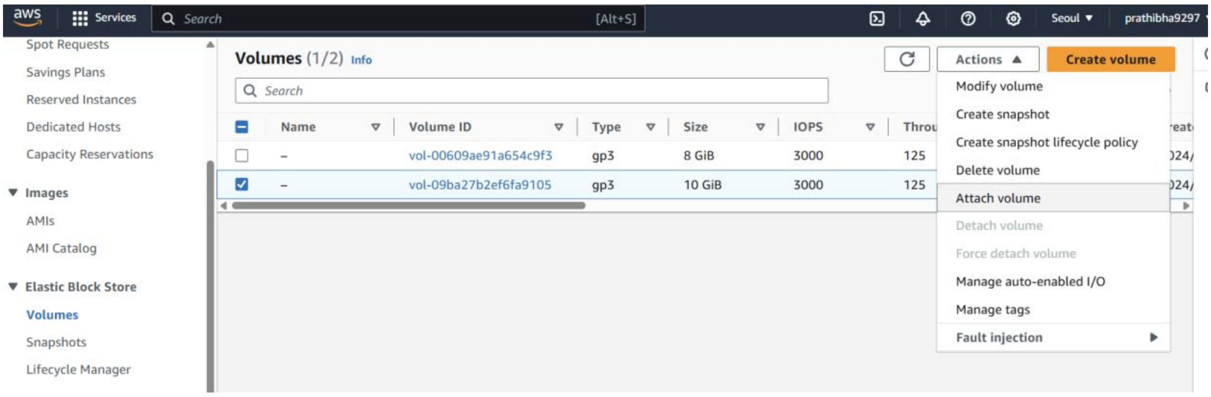
Don't create volume from a snapshot

Encryption Info

Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

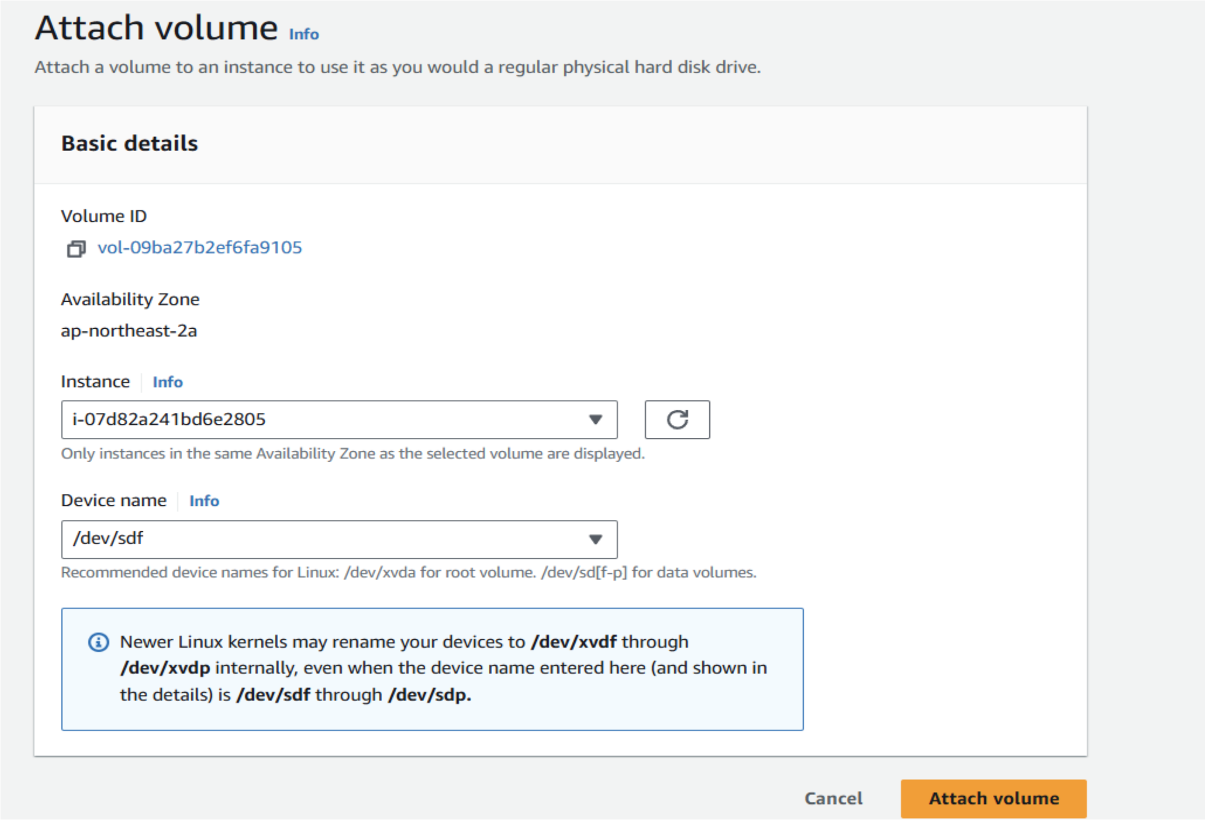
□ Encrypt this volume

I was created the volume with 10GiB.once created the volume attach the volume to the EC2 instance(instance 1). Ec2 instance and created EBS volume should be same availability zone then only we can able to attach the volume to the EC2 instance.



Volumes (1/2) Info

	Name	Volume ID	Type	Size	IOPS	Throu
□	–	vol-00609ae91a654c9f3	gp3	8 GiB	3000	125
☑	–	vol-09ba27b2ef6fa9105	gp3	10 GiB	3000	125



Attach volume Info

Attach a volume to an instance to use it as you would a regular physical hard disk drive.

Basic details

Volume ID
vol-09ba27b2ef6fa9105

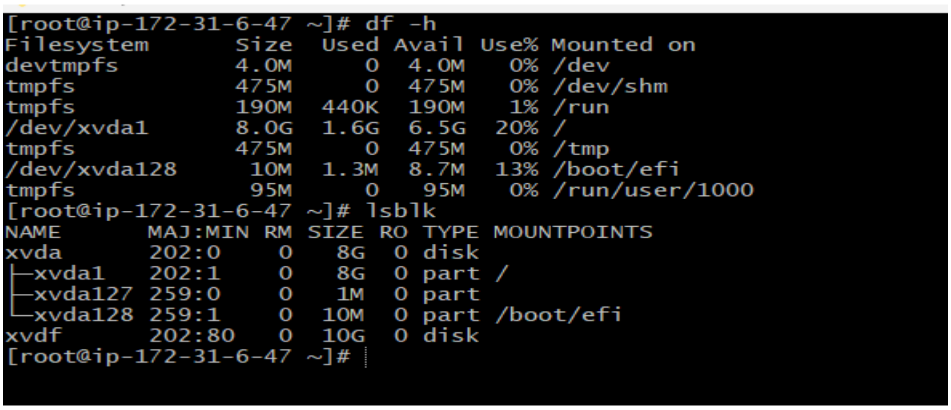
Availability Zone
ap-northeast-2a

Instance Info
i-07d82a241bd6e2805
Only instances in the same Availability Zone as the selected volume are displayed.

Device name Info
/dev/sdf
Recommended device names for Linux: /dev/xvda for root volume. /dev/sd[f-p] for data volumes.

Newer Linux kernels may rename your devices to **/dev/xvdf** through **/dev/xvdp** internally, even when the device name entered here (and shown in the details) is **/dev/sdf** through **/dev/sdp.**

Now for EC2 instance total storage will be 18GiB(8 GiB default storage+10 GiB attached volume).



```
[root@ip-172-31-6-47 ~]# df -h
Filesystem       Size  Used Avail Use% Mounted on
devtmpfs         4.0M     0  4.0M   0% /dev
tmpfs            475M     0  475M   0% /dev/shm
tmpfs            190M  440K  190M   1% /run
/dev/xvda1       8.0G  1.6G  6.5G  20% /
tmpfs            475M     0  475M   0% /tmp
/dev/xvda128      10M  1.3M  8.7M  13% /boot/efi
tmpfs             95M     0   95M   0% /run/user/1000
[root@ip-172-31-6-47 ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda      202:0    0    8G  0 disk
├─xvda1   202:1    0    8G  0 part /
├─xvda127 259:0    0    1M  0 part
└─xvda128 259:1    0   10M  0 part /boot/efi
xvdf      202:80   0   10G  0 disk
[root@ip-172-31-6-47 ~]#
```

Attached volume will be available in block storage, for checking that storage in git bash use lsblk command.

For checking is there any file system in device ->file -s /dev/xvdf

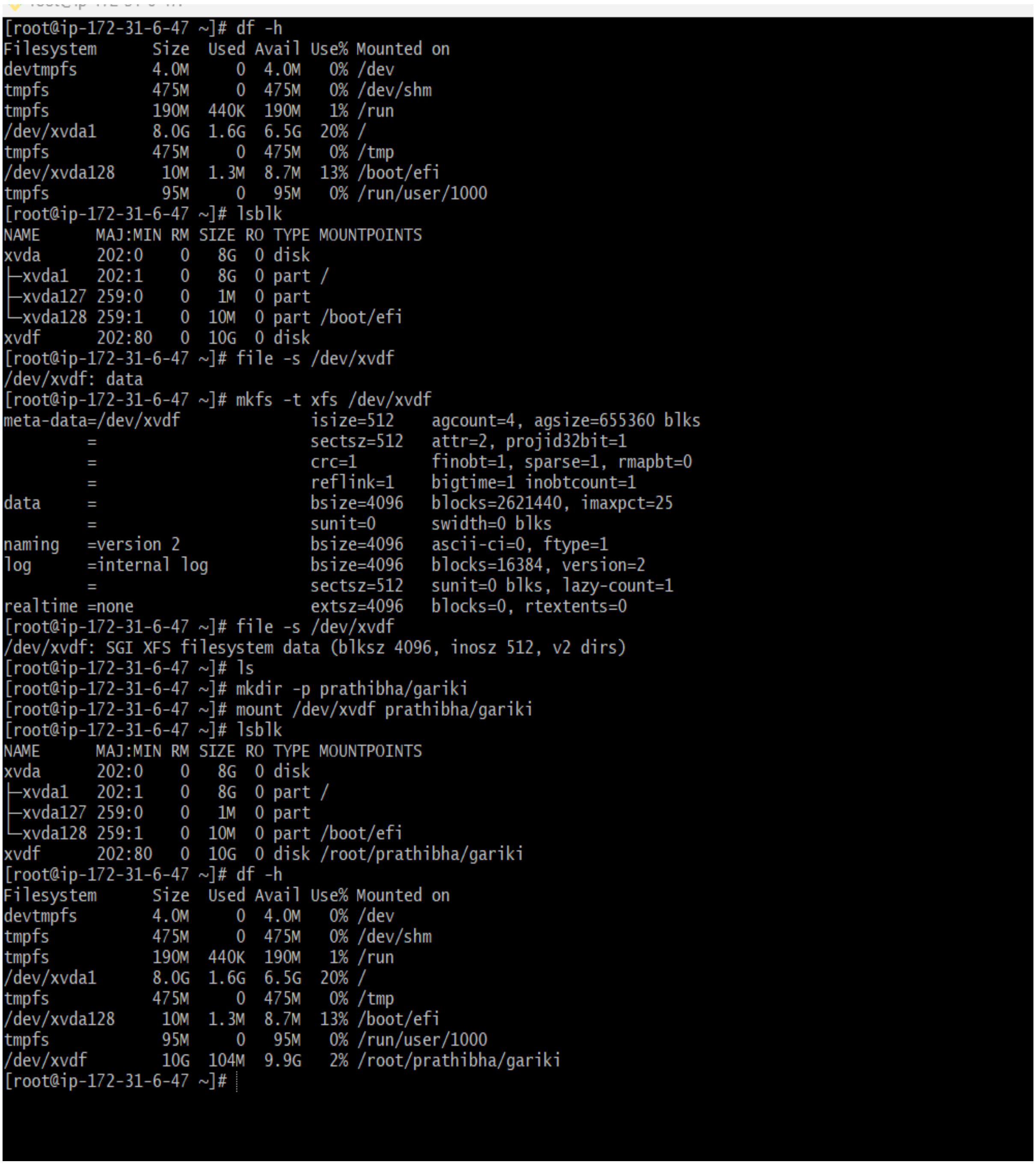
For creating file system ->mkfs -t xfs /dev/xvdf

Creating a nested directory->mkdir -p apps/volume

For mounting the storage block storage to the root storage, we need to use

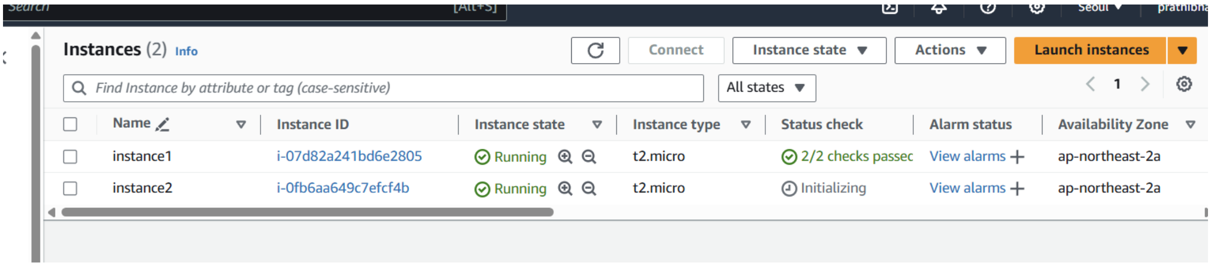
Mount /dev/xvdf apps/volume.

After mounting the storage total output storage for the root user is 18 GiB as shown in below.



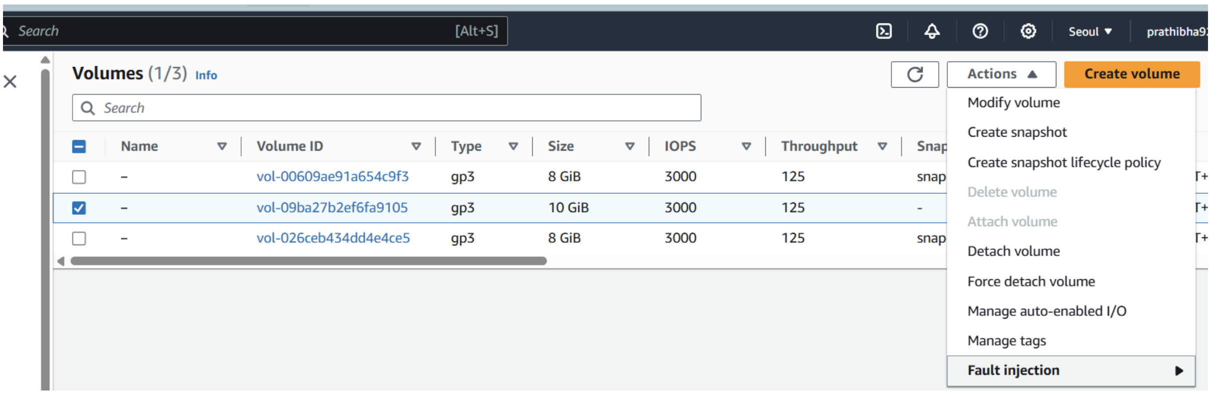
```
[root@ip-172-31-6-47 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           475M     0  475M   0% /dev/shm
tmpfs           190M  440K  190M   1% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M     0  475M   0% /tmp
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs            95M     0   95M   0% /run/user/1000
[root@ip-172-31-6-47 ~]# lsblk
NAME      MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
xvda      202:0    0   8G  0 disk
├─xvda1   202:1    0   8G  0 part /
├─xvda127 259:0    0   1M  0 part
└─xvda128 259:1    0  10M  0 part /boot/efi
xvdf      202:80   0  10G  0 disk
[root@ip-172-31-6-47 ~]# file -s /dev/xvdf
/dev/xvdf: data
[root@ip-172-31-6-47 ~]# mkfs -t xfs /dev/xvdf
meta-data=/dev/xvdf              isize=512    agcount=4, agsize=655360 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1
data     =                       bsize=4096   blocks=2621440, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@ip-172-31-6-47 ~]# file -s /dev/xvdf
/dev/xvdf: SGI XFS filesystem data (blksz 4096, inosz 512, v2 dirs)
[root@ip-172-31-6-47 ~]# ls
[root@ip-172-31-6-47 ~]# mkdir -p prathibha/gariki
[root@ip-172-31-6-47 ~]# mount /dev/xvdf prathibha/gariki
[root@ip-172-31-6-47 ~]# lsblk
NAME      MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
xvda      202:0    0   8G  0 disk
├─xvda1   202:1    0   8G  0 part /
├─xvda127 259:0    0   1M  0 part
└─xvda128 259:1    0  10M  0 part /boot/efi
xvdf      202:80   0  10G  0 disk /root/prathibha/gariki
[root@ip-172-31-6-47 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           475M     0  475M   0% /dev/shm
tmpfs           190M  440K  190M   1% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M     0  475M   0% /tmp
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs            95M     0   95M   0% /run/user/1000
/dev/xvdf        10G  104M  9.9G   2% /root/prathibha/gariki
[root@ip-172-31-6-47 ~]#
```

Step3:Create another instance in AWS console and names as instance 2 as shown in below

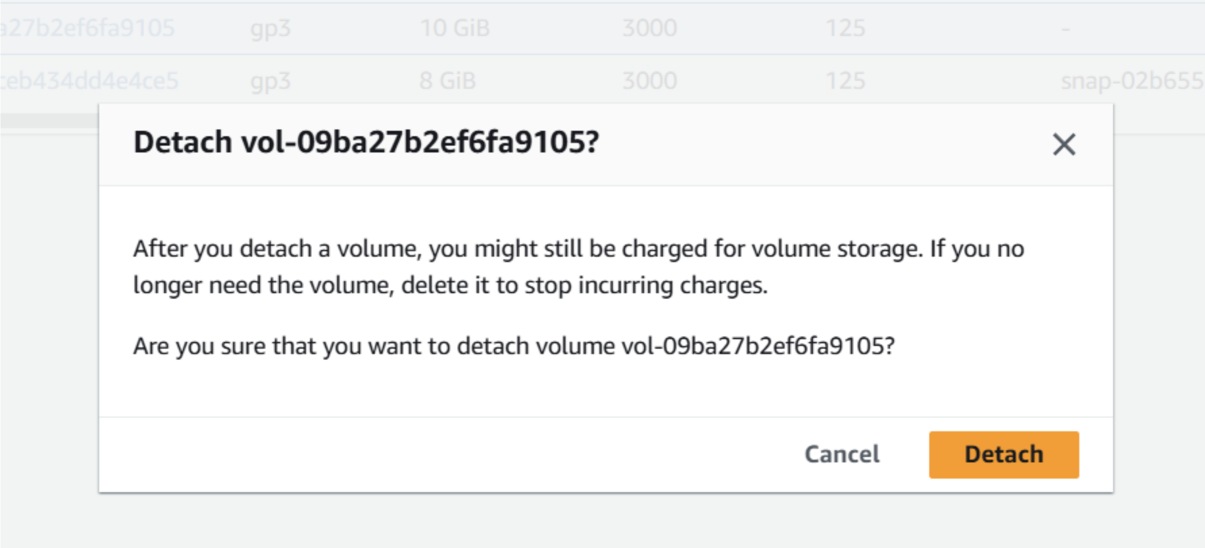


□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
□	instance1	i-07d82a241bd6e2805	Running	t2.micro	2/2 checks passec	View alarms +	ap-northeast-2a
□	instance2	i-0fb6aa649c7efcf4b	Running	t2.micro	Initializing	View alarms +	ap-northeast-2a

Once created the instance try to attach the instance to the EBS volume but can’t able to attach as shown below it will be disables because the EBS volume attached with instance 2. At a time, we can’t able to attach multiple EC2 instances to the same EBS volume. But we detach the EBS volume from the instance 1 and able to attach to the instance 2 as shown in below diagrams.



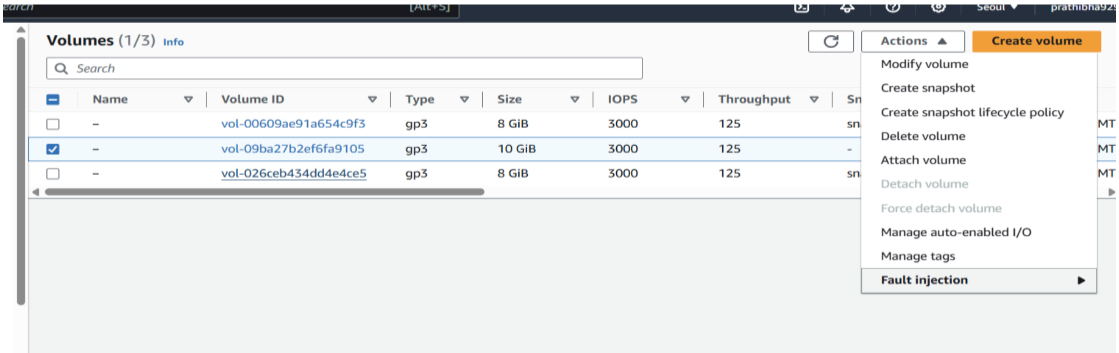
	Name	Volume ID	Type	Size	IOPS	Throughput	Snap
□	–	vol-00609ae91a654c9f3	gp3	8 GiB	3000	125	snap
☑	–	vol-09ba27b2ef6fa9105	gp3	10 GiB	3000	125	-
□	–	vol-026ceb434dd4e4ce5	gp3	8 GiB	3000	125	snap


27b2ef6fa9105 gp3 10 GiB 3000 125 -
eb434dd4e4ce5 gp3 8 GiB 3000 125 snap-02b655

Detach vol-09ba27b2ef6fa9105?

After you detach a volume, you might still be charged for volume storage. If you no longer need the volume, delete it to stop incurring charges.

Are you sure that you want to detach volume vol-09ba27b2ef6fa9105?



	Name	Volume ID	Type	Size	IOPS	Throughput	Sn
□	–	vol-00609ae91a654c9f3	gp3	8 GiB	3000	125	sn
☑	–	vol-09ba27b2ef6fa9105	gp3	10 GiB	3000	125	-
□	–	vol-026ceb434dd4e4ce5	gp3	8 GiB	3000	125	sn

Like wise we need to create another EC2 instance and attach to the EBS volume after detaching from the instance 2.

Conclusion:

- Elastic block storage is a scalable, high-performance and block storage service designed for amazon EC2 and it can be attached or detach from EC2.
- EBS and EC2 instances both are in same availability zone only we can able to attach or detach the volume.
- Multiple EBS can attach to the single EC2 instance but multiple EC2 instances can’t able to attach to the same EBS.