

Problem Statement:

You work for XYZ Corporation. Your corporation is working on an application and they require secured web servers on Linux to launch the application.

Tasks To Be Performed:

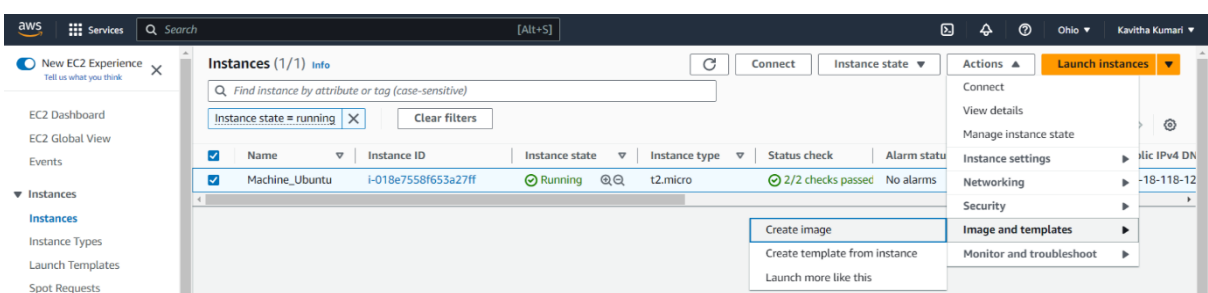
1. Create an instance in the US-East-2 (Ohio) region with Linux OS and manage the requirement of web servers of your company using AMI.
2. Replicate the instance in the US-West-2 (Oregon) region.
3. Build two EBS volumes and attach them to the instance in the US-East-2 (Ohio) region.
4. Delete one volume after detaching it and extend the size of the other volume.
5. Take backup of this EBS volume

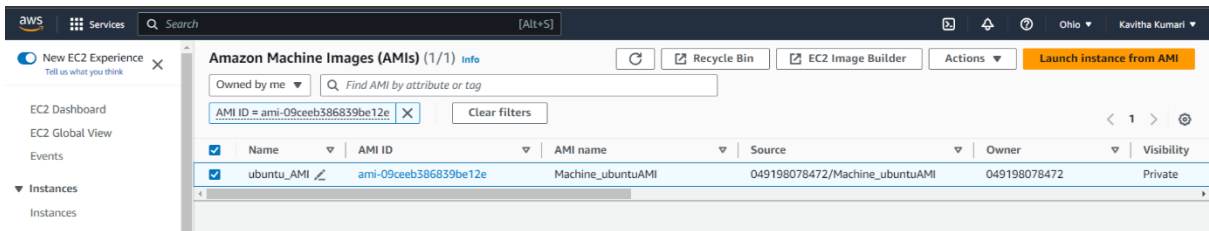
Procedure:

- First, we will launch an instance with the ubuntu machine. And connect the instance with cli and install the server that is apache2. Run the following commands.
 - `sudo apt-get update`
 - `sudo apt-get install apache2 -y`
 - therefore, we have now installed the web server apache 2 into our machine.

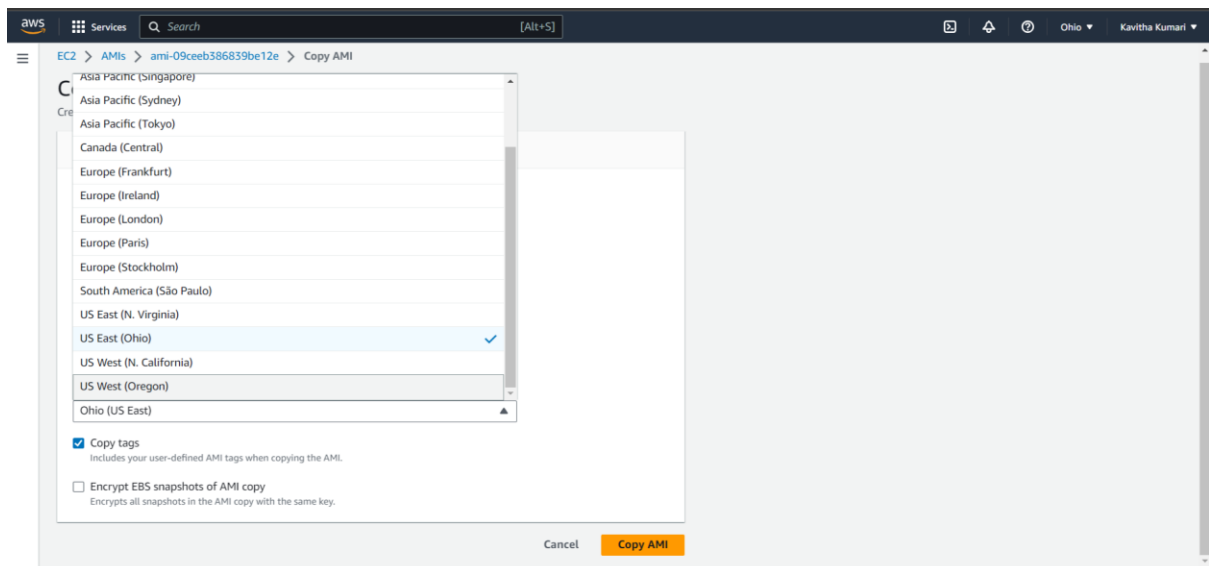
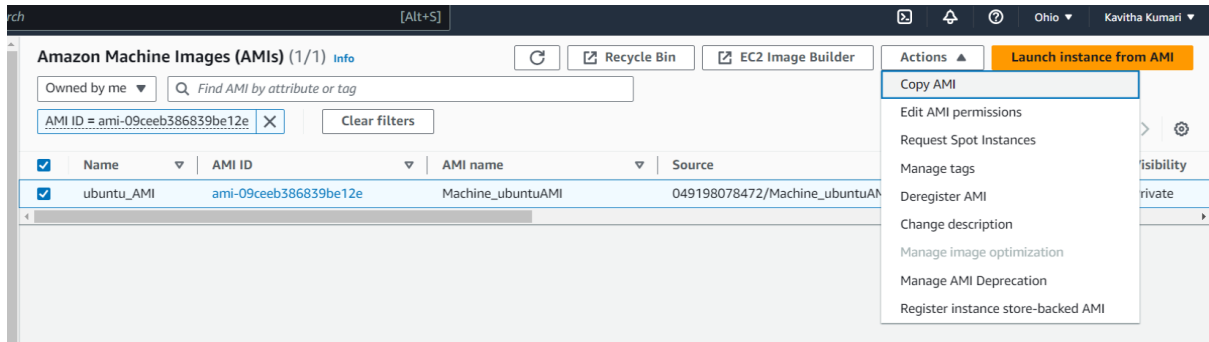


- Now we must replicate the region to the oregon region. For this first we will create the AMI of the existing machine and replicate that AMI to the Oregon region.
- Now give the name for AMI and click on create ami.

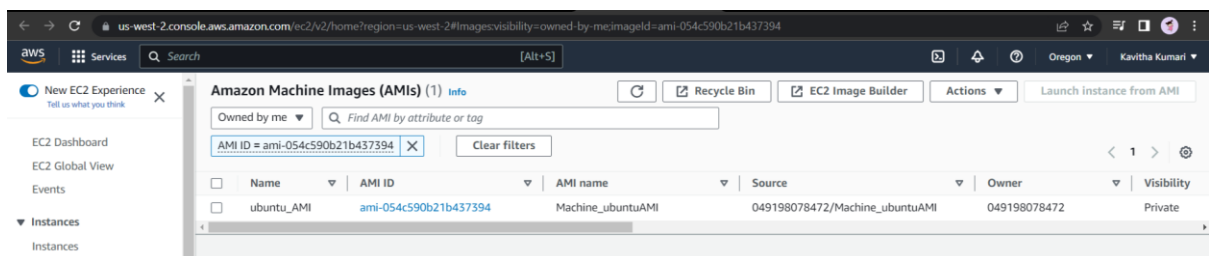




- Now we need to copy the AMI to the another region.



- Select the US West (Oregon) and click on Copy AMI.



- Therefore, the AMI is created in the Oregon region.
- Now the next step is to launch an instance in the Oregon region with the AMI we created.
- To create an instance, select the AMI and click on **Launch instance from AMI**.
- Allow HTTP and let everything be default and **Launch Instance**.

Application and OS Images (Amazon Machine Image)

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

[AMI from catalog](#) | [My AMIs](#) | [Quick Start](#)

Amazon Machine Image (AMI)
Machine_ubuntuAMI
ami-054c590b21b437394

Published: 2023-07-22T17:18:34.00Z
Architecture: x86_64
Virtualization: hvm
Root device type: ebs
ENA Enabled: Yes

[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Summary

Number of instances: 1

Software Image (AMI)
[Copied ami-09ceeb386839be12e ...read more]
ami-054c590b21b437394

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.

[Cancel](#) [Launch instance](#) [Review commands](#)

- Now if we copy the public IP of this instance search that in browser, we will see the default apache2 web page.
- Now we need to build 2 EBS volume and attach them to the instance of the **Ohio** region.
- **Note:** The volumes which we create must be in the same region. That is in **us-east-2a**.
- Navigate to the Elastic Block Store and click on volumes. Then click on **create volume**.
- Select the size and 1 and every thing be default and click on **create volume**.
- Similarly create another volume.

Successfully created volume vol-008ab01cb4f47a0a7.

Volumes (1/3)

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot	Created
-	vol-051c04d80ca4801b5	gp2	8 GiB	100	-	snap-07592d6...	2023/07/22 22:26 GMT+5:...
first_volume	vol-043ce7dff95524f2c	gp2	1 GiB	100	-	-	2023/07/22 23:17 GMT+5:...
second_volume	vol-008ab01cb4f47a0a7	gp2	2 GiB	100	-	-	2023/07/22 23:18 GMT+5:...

- Once both the Volume_state goes to available we can attach it to the instance.

Successfully created volume vol-008ab01cb4f47a0a7.

Volumes (1/3)

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot
-	vol-051c04d80ca4801b5	gp2	8 GiB	100	-	snap-
first_volume	vol-043ce7dff95524f2c	gp2	1 GiB	100	-	-
second_volume	vol-008ab01cb4f47a0a7	gp2	2 GiB	100	-	-

Volume ID: vol-043ce7dff95524f2c (first_volume)

Actions

- Modify volume
- Create snapshot
- Create snapshot lifecycle policy
- Delete volume
- Attach volume**
- Detach volume
- Force detach volume
- Manage auto-enabled I/O
- Manage tags
- Fault injection

Attach volume

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

Basic details

Volume ID
vol-043ce7dff95524f2c (first_volume)

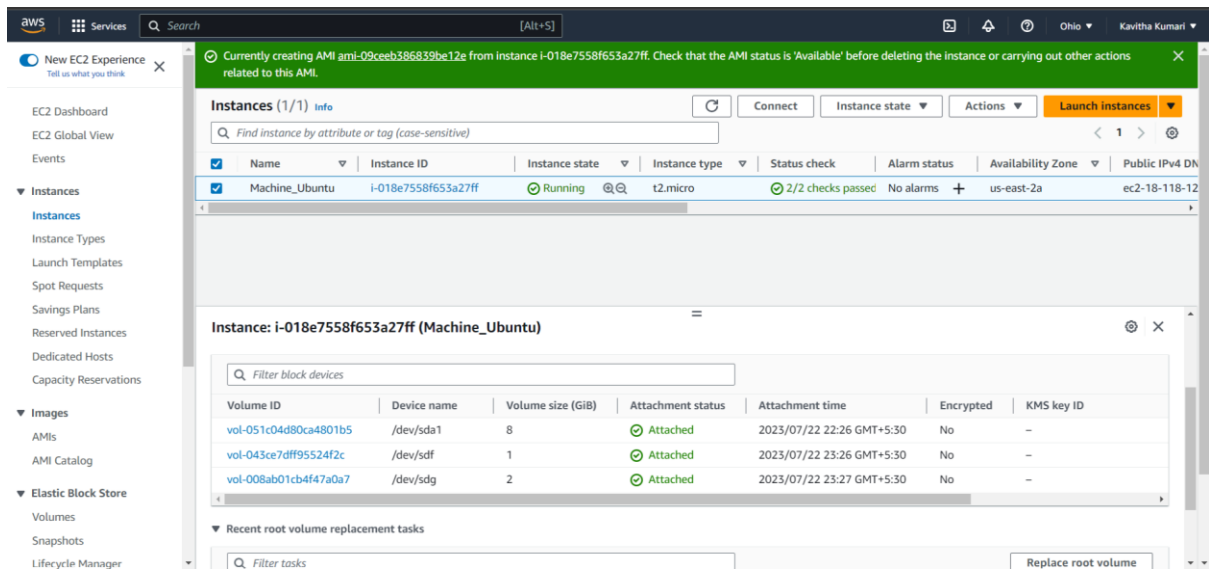
Availability Zone
us-east-2a

Instance

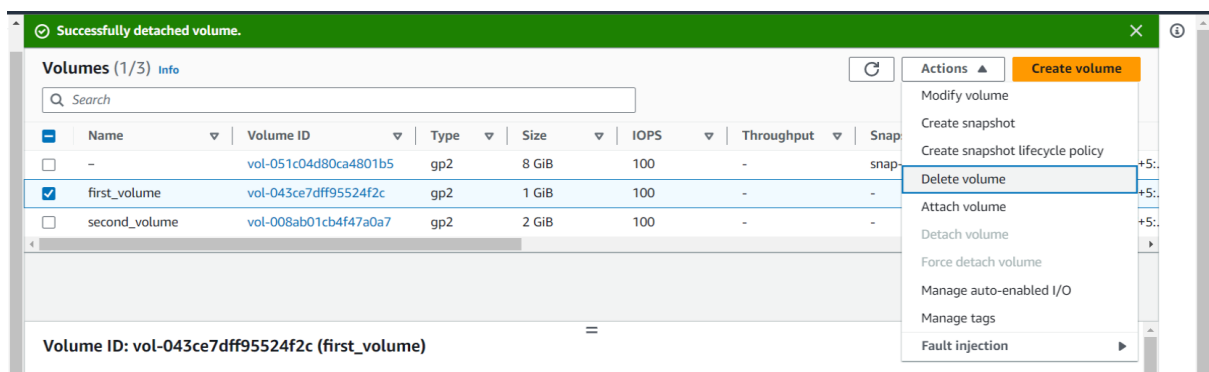
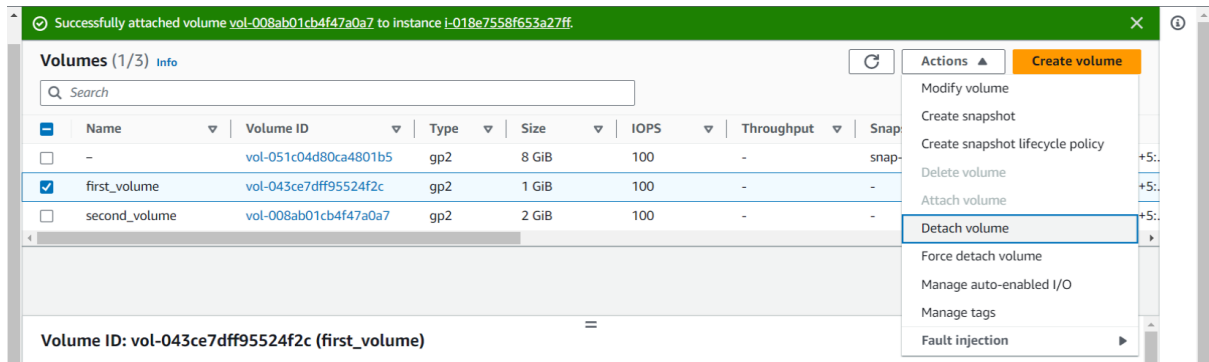
Device name

[Cancel](#) [Attach volume](#)

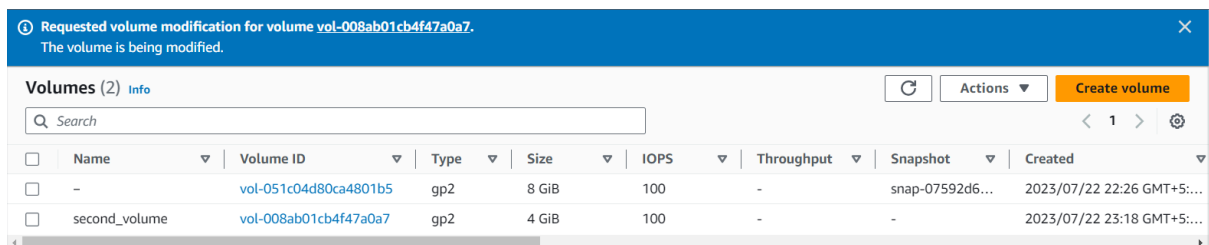
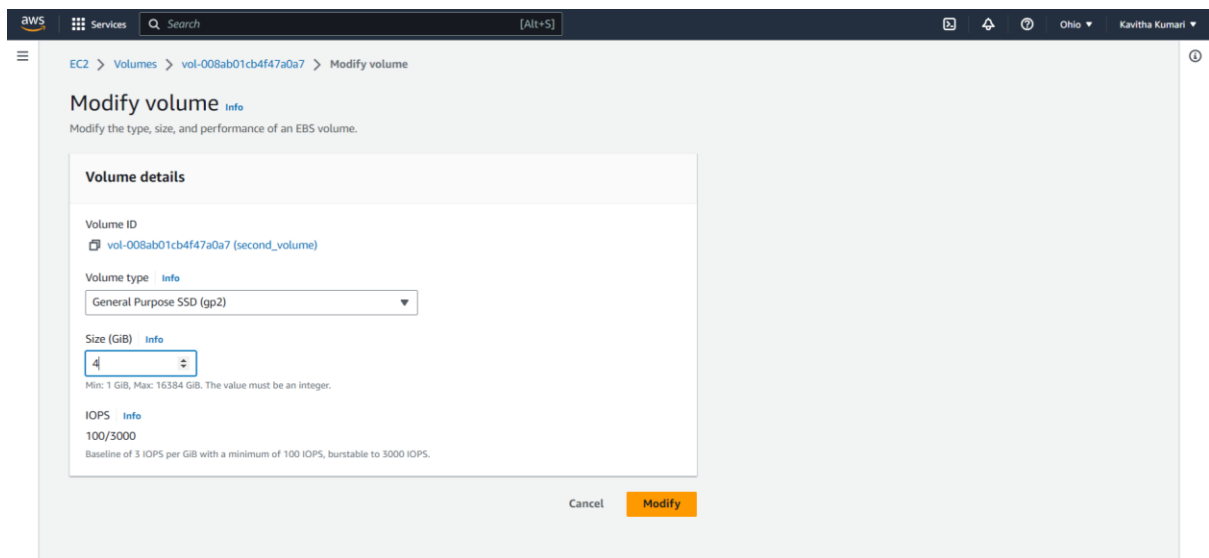
- Select the instance and click on **Attach Volume**.
- Similarly, we need to attach another volume.



- To delete an EBS volume we need to first detach it and delete it.
- Select the volume and go to action and click on detach volume.



- Now we need to increase the size of the volume.
- Select the volume and go to actions there you will find the option of **Modify volume** click on that and then you can modify the volume.
- **Note:** You can only increase the volume but not decrease.



- Therefore, the volume has changed to the 4 GiB.
- Now we need to **format and Mount** the EBS volume which we have created.
- This can be done with the help you the Terminal of the instance. Run the following command.
 - lsblk (It list out all the block level storage)
 - sudo file -s /dev/xvda (here we are checking whther the root volume is being formatted or not)

```
xvda    202:0    0    8G    0 disk
├─xvda1  202:1    0    7.9G  0 part /
├─xvda14 202:14   0     4M    0 part
└─xvda15 202:15   0   106M   0 part /boot/efi
xvdg    202:96   0     4G    0 disk
ubuntu@ip-172-31-4-134:~$ sudo file -s /dev/xvda
/dev/xvda: DOS/MBR boot sector, extended partition table (last)
ubuntu@ip-172-31-4-134:~$ ^C
ubuntu@ip-172-31-4-134:~$
```

- The above statement means it has been formatted.
- sudo file -s /dev/xvdg (here we are checking whether the ebs volume is formatted on not.

```
xvdg    202:96   0     4G    0 disk
ubuntu@ip-172-31-4-134:~$ sudo file -s /dev/xvda
/dev/xvda: DOS/MBR boot sector, extended partition table (last)
ubuntu@ip-172-31-4-134:~$ ^C
ubuntu@ip-172-31-4-134:~$ sudo file -s /dev/xvdg
/dev/xvdg: data
ubuntu@ip-172-31-4-134:~$
```

- It means the volume has been not formatted.
- `sudo mkfs -t ext4 /dev/xvdg` (here we are doing the format of the ebs volume attached separately)

```

Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

ubuntu@ip-172-31-4-134:~$ sudo file -s /dev/xvdg
/dev/xvdg: Linux rev 1.0 ext4 filesystem data, UUID=c3e5cfb9-928c-462b-8efe-0cbd3f7403c2 (extents) (64bit) (large files) (huge files)
ubuntu@ip-172-31-4-134:~$

```

- Now we have to create the directory.
- `sudo mkdir ebs` (We created the directory named ebs)
- `ls` (it lists out the directories or files)

```

/dev/xvdg: Linux rev 1.0 ext4 filesystem data, UUID=c3e5cfb9-928c-462b-8efe-0cbd3f7403c2 (extents) (64bit) (large files) (huge files)
ubuntu@ip-172-31-4-134:~$ sudo mkdir ebs
ubuntu@ip-172-31-4-134:~$ ls
ebs
ubuntu@ip-172-31-4-134:~$

```

- `sudo mount /dev/xvdg ebs` (here we are mounting the particular volume to the ebs directory)
- Now if we run the command `lsblk` we will see the following output.

```

ubuntu@ip-172-31-4-134:~$ ls
ebs
ubuntu@ip-172-31-4-134:~$ sudo mount /dev/xvdg ebs
ubuntu@ip-172-31-4-134:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0        7:0      0   24.4M  1 loop /snap/amazon-ssm-agent/6312
loop1        7:1      0   55.6M  1 loop /snap/core18/2745
loop2        7:2      0   63.3M  1 loop /snap/core20/1879
loop3        7:3      0  111.9M  1 loop /snap/lxd/24322
loop4        7:4      0   53.2M  1 loop /snap/snapd/19122
xvda         202:0     0    8G    0 disk
├─xvda1      202:1     0    7.9G  0 part /
├─xvda14     202:14    0     4M   0 part
└─xvda15     202:15    0   106M  0 part /boot/efi
xvdg         202:96    0     4G    0 disk /home/ubuntu/ebs
ubuntu@ip-172-31-4-134:~$

```

- Now we can create the files in the ebs volume.

```

ubuntu@ip-172-31-4-134:~$ cd ebs
ubuntu@ip-172-31-4-134:~/ebs$ sudo nano file.txt
ubuntu@ip-172-31-4-134:~/ebs$ ls
file.txt  lost+found
ubuntu@ip-172-31-4-134:~/ebs$

```

- `cd` (to exit the directory)
- To unmount the directory, we can run the following command.
- `sudo umount /dev/xvdg`
- Now we need to take the back ups of the ebs volumes.
- To take the backup select the volume and go to the actions and from there click on **create snapshot**.

Services

Search

[Alt+S]

Details

Volume ID

vol-008ab01cb4f7a0a7 (second_volume)

Description

Add a description for your snapshot

snap_of_volume

255 characters maximum.

Encryption

Info

Not encrypted

Tags

Info

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

Add tag

You can add 50 more tags.

Cancel

Create snapshot

- To check the snapshots, navigate to the snapshots and there you will see the snapshots available

Snapshots (3) Info

Owned by me

Search

Refresh

Recycle Bin

Actions

Create snapshot

	Name	Snapshot ID	Volume size	Description	Storage tier	Snapshot status	Started
<input type="checkbox"/>	-	snap-00075d41ae0740fc6	4 GiB	snap_of_volume	Standard	Pending	2023/07/23 00:23 GMT
<input type="checkbox"/>	-	snap-0ae28f80818c3944a	8 GiB	Created by CreateImage(-...	Standard	Completed	2023/07/22 22:41 GMT
<input type="checkbox"/>	-	snap-0bc8cfe4dbc024f44	8 GiB	Created by CreateImage(-...	Standard	Completed	2023/07/16 00:41 GMT

EC2 > Snapshots > snap-00075d41ae0740fc6

Refresh

Delete

Actions

Snapshot ID

snap-00075d41ae0740fc6

Volume size

4 GiB

Progress

Available (100%)

Snapshot status

Completed

Owner

049198078472

Volume ID

vol-008ab01cb4f7a0a7

Started

Sun Jul 23 2023 00:23:00 GMT+0530 (India Standard Time)

Product codes

-

Encryption

Not encrypted

KMS key ID

-

KMS key alias

-

KMS key ARN

-

Fast snapshot restore

-

Description

snap_of_volume

Permissions

Storage tier

Tags

Snapshot share permissions

Private

The snapshot is shared only with AWS accounts that you specified.

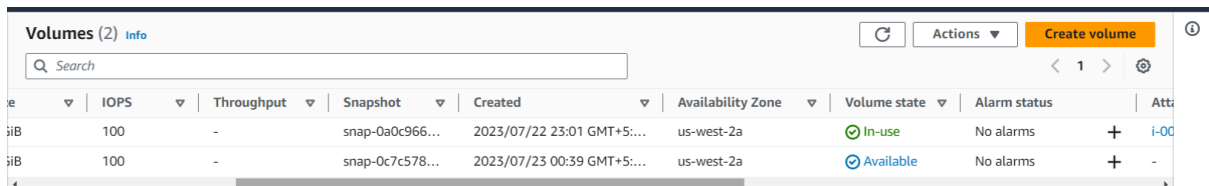
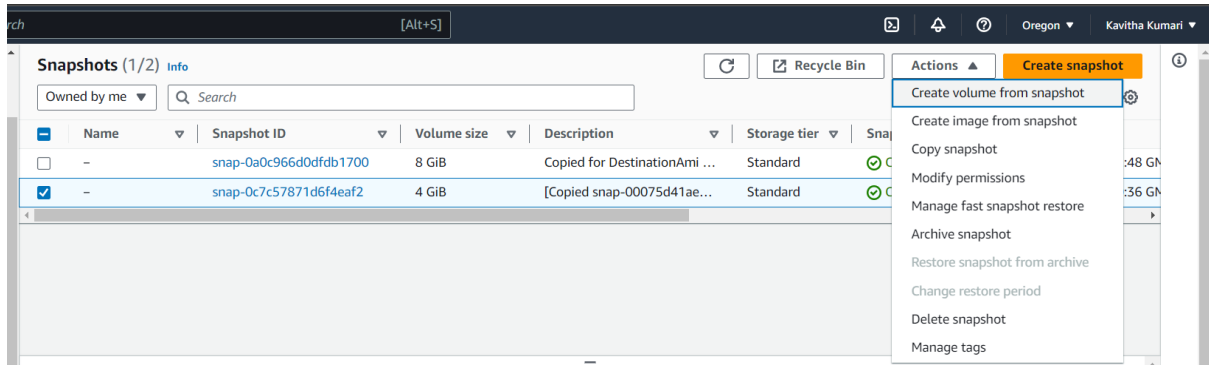
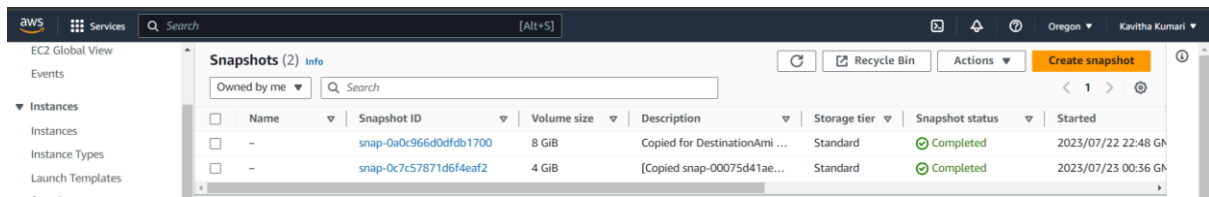
Modify permissions

Shared accounts (0)

The snapshot is shared with the following AWS accounts.

Add account ID

- From here we can copy the snapshot to the another region named Oregon and create the volume from the snapshot and attach to the instance.



- Attach the volume to the instance which we created in the Oregon region.
- After attaching we need to connect our instance to the terminal. Then run the following command.
 - `sudo apt-get update`
 - `lsblk`
 - `sudo file -s /dev/xvdf` (to check whether it is mounted or not)
 - `sudo mkdir ebs`
 - `sudo mount /dev/xvdf ebs`
 - now if we run `lsblk` we will see that it is successfully available.
 - `cd ebs`
 - `ls`
 - we will see the file `file.txt`
 - `sudo cat file.txt` (to see the content of the file)

```

-xvda1 202:1 0 7.9G 0 part /
-xvda14 202:14 0 4M 0 part
-xvda15 202:15 0 106M 0 part /boot/efi
xvdf 202:80 0 4G 0 disk
ubuntu@ip-172-31-16-192:~$ sudo file -s /dev/xvdf
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=c3e5cfb9-928c-462b-8efe-0cb3f7403c2 (needs journal recovery) (extents) (64bit) (large files) (huge files)
ubuntu@ip-172-31-16-192:~$ sudo mkdir ebs
ubuntu@ip-172-31-16-192:~$ ls
ebs
ubuntu@ip-172-31-16-192:~$ sudo mount /dev/xvdf ebs
ubuntu@ip-172-31-16-192:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0       7:0      0   55.6M 1 loop /snap/core18/2745
loop1       7:1      0   63.3M 1 loop /snap/core20/1879
loop2       7:2      0   111.9M 1 loop /snap/lxd/24322
loop3       7:3      0   24.4M 1 loop /snap/amazon-ssm-agent/6312
loop4       7:4      0   53.2M 1 loop /snap/snapd/19122
xvda1       202:1    0    7.9G 0 part /
xvda14      202:14   0     4M 0 part
xvda15      202:15   0   106M 0 part /boot/efi
xvdf        202:80   0     4G 0 disk /home/ubuntu/ebs
ubuntu@ip-172-31-16-192:~$ cd ebs
ubuntu@ip-172-31-16-192:~/ebs$ ls
file.txt  lost+found
ubuntu@ip-172-31-16-192:~/ebs$ sudo cat file.txt
This is the file on ebs
ubuntu@ip-172-31-16-192:~/ebs$

```

-----END-----