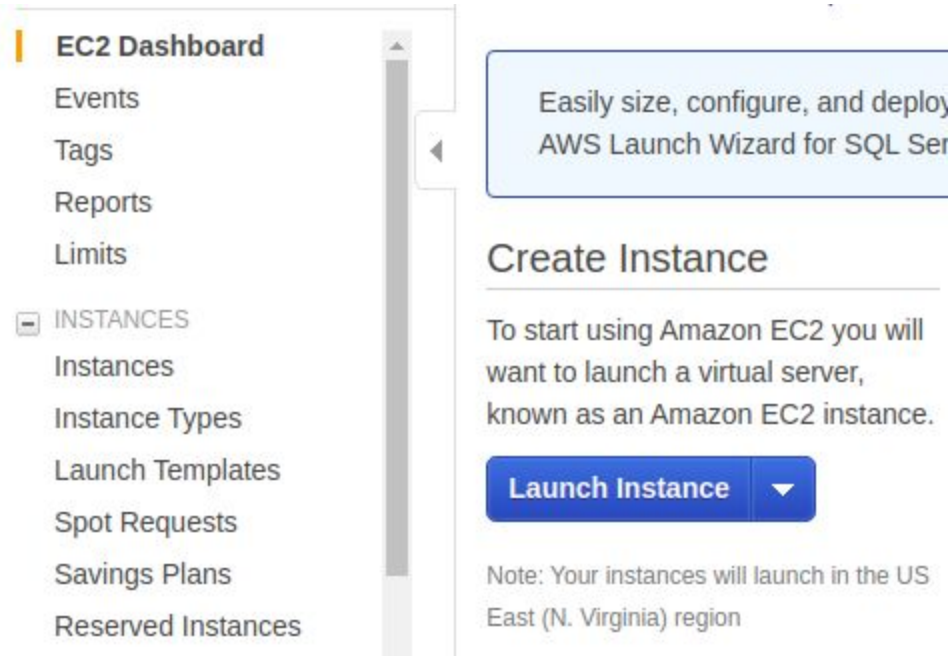
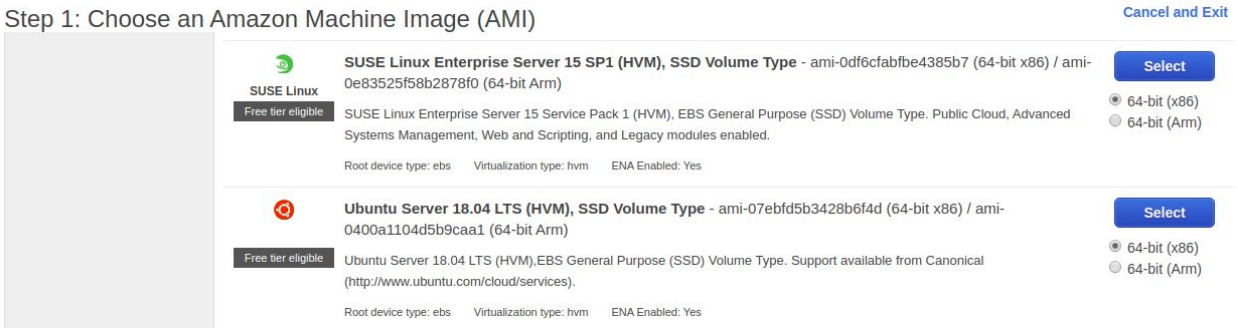
ASSESSMENT ON AWS EC2 AND EBS

1. Create an EC2 instance (Ubunutu 18.04, T3 nano).(instance A)

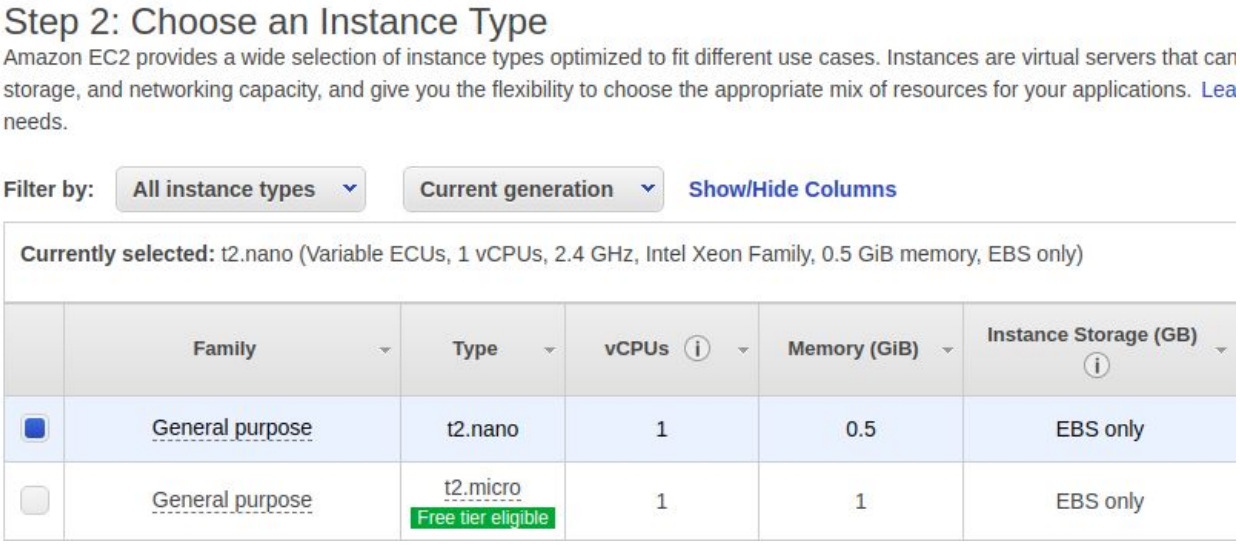
STEP 1: Click on EC2 and then click on launch instance.



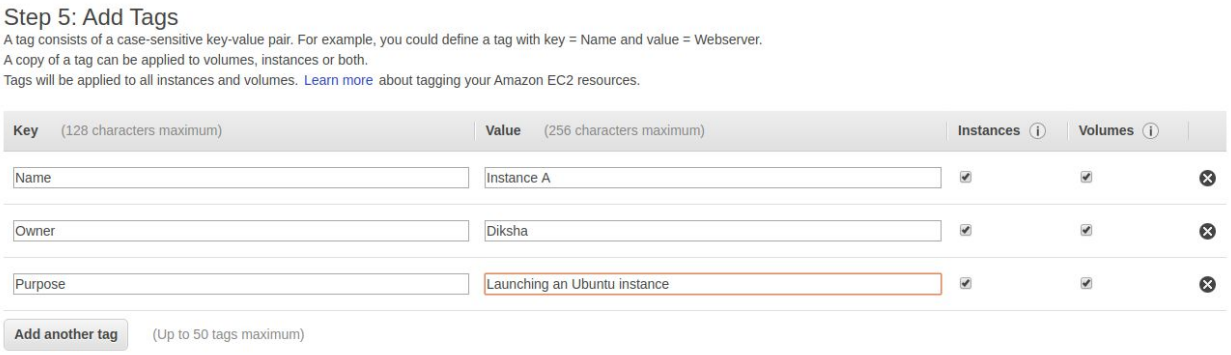
STEP 2: Choose an AMI



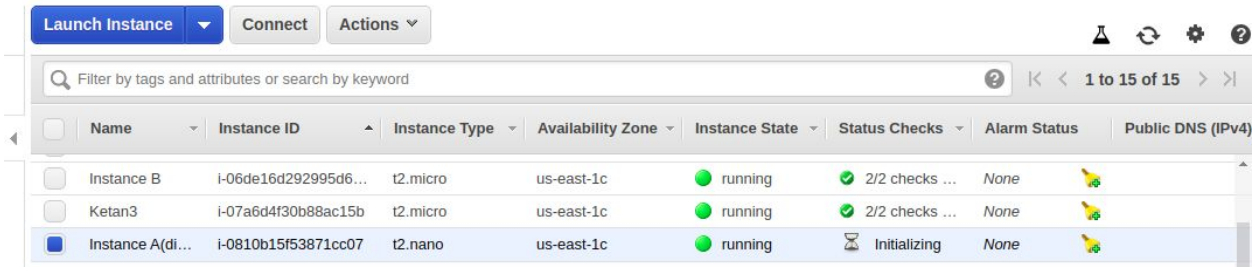
STEP 3: Choose an instance type



STEP 4: Add tags

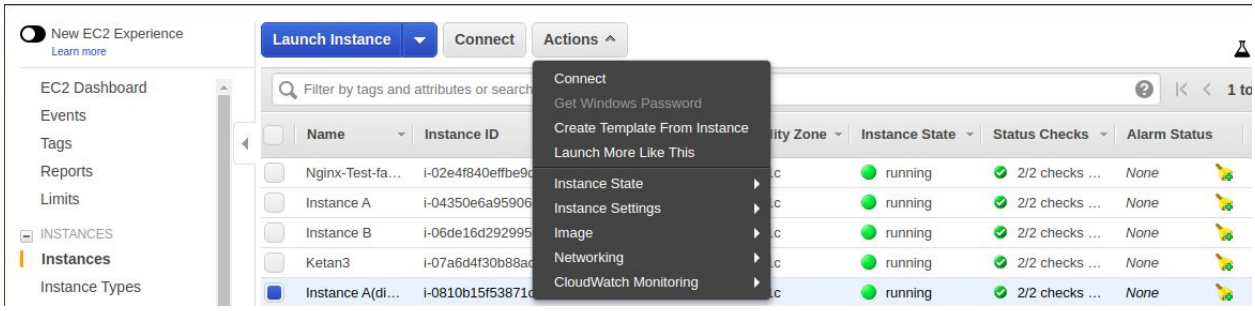


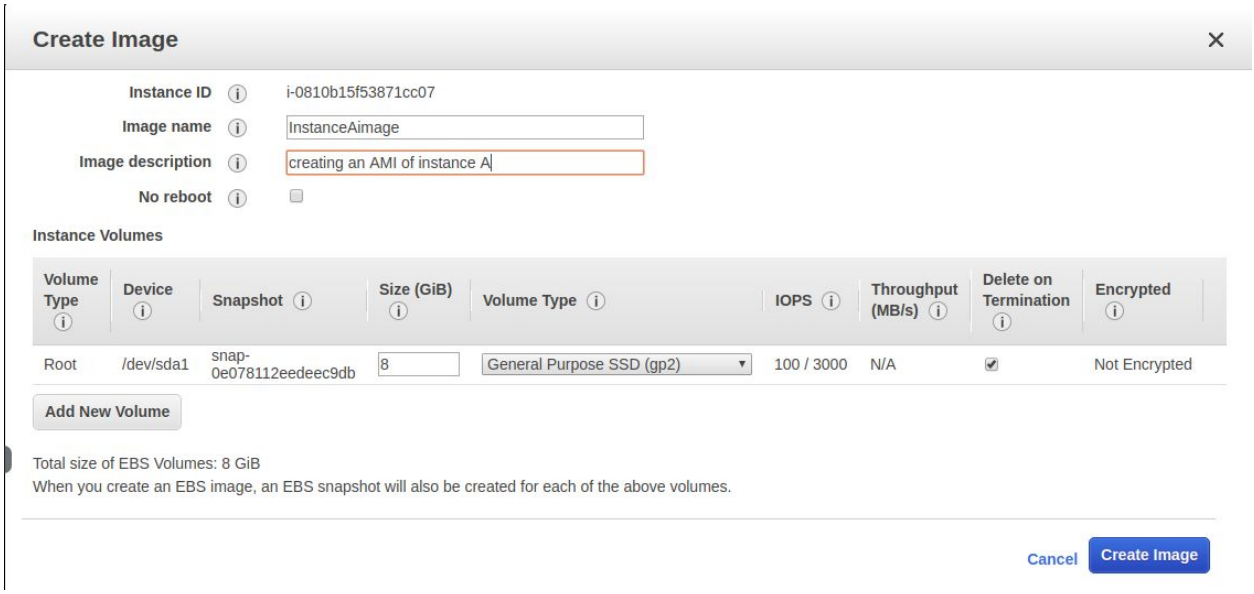
STEP 5: Instance is launched

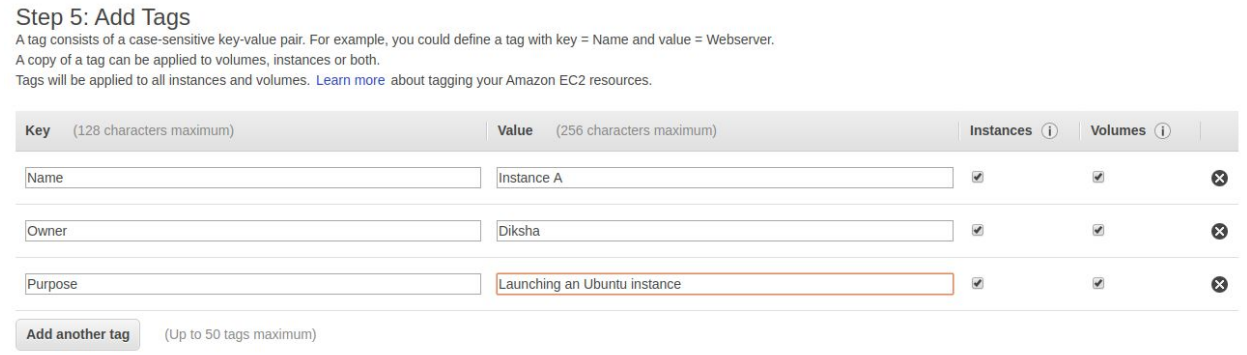


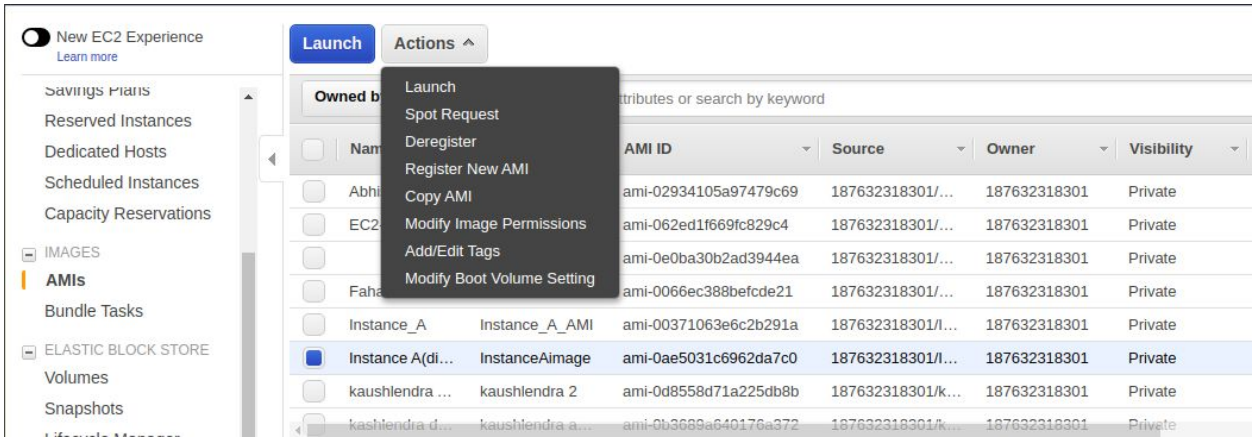
****

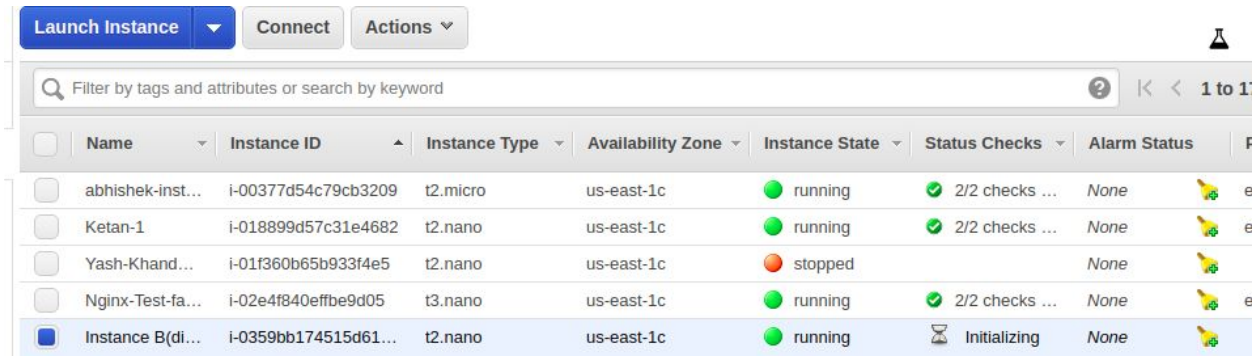
2. Create AMI of above instance and launch it. (instance B)







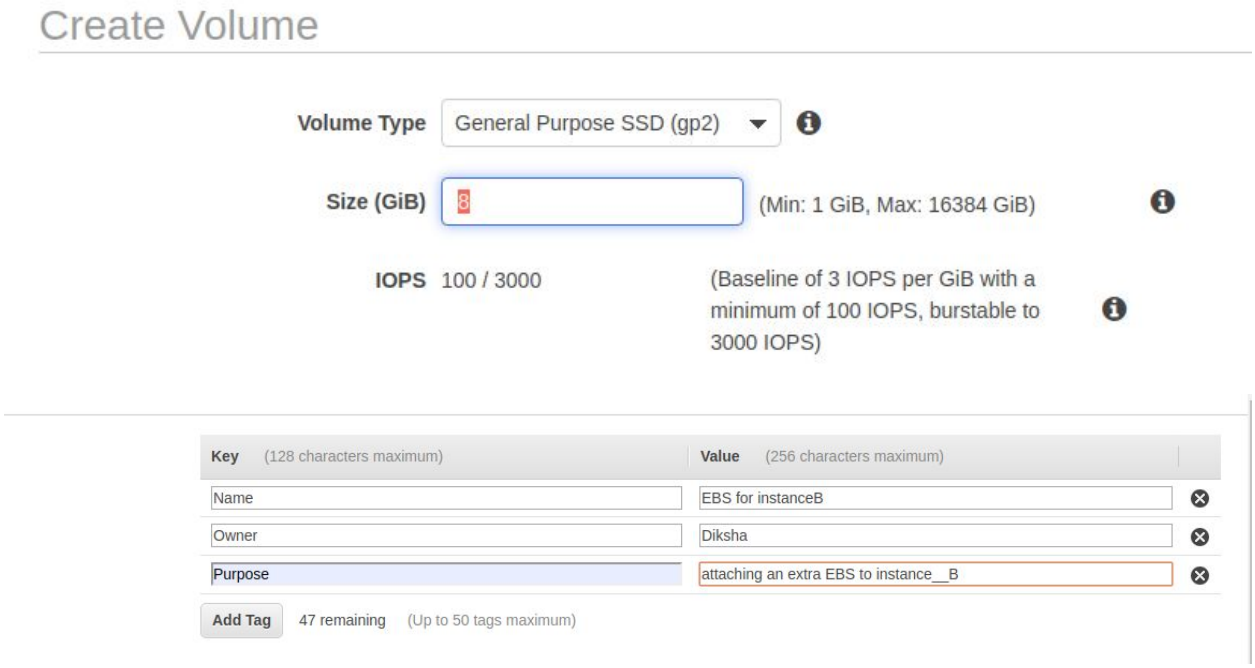




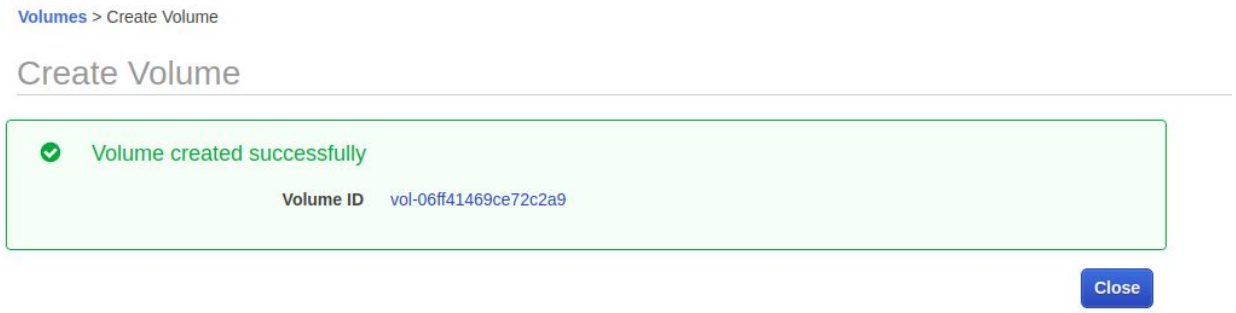
****

3. Attach EBS (8 GB) on that running instance.

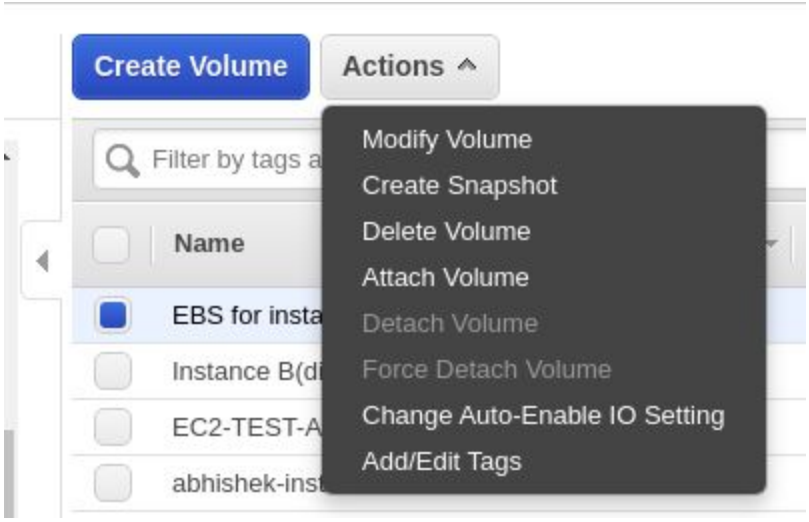
STEP 1: Click on create EBS and then click on Create volume.



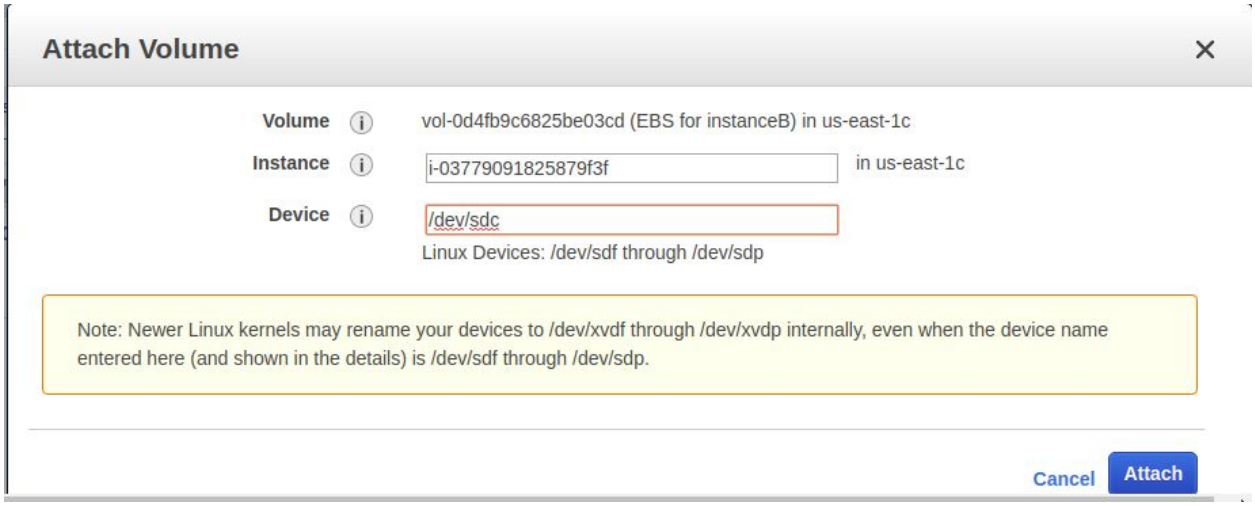
STEP 2: Make sure volume is in same availability zone as of the instance you want to attach.

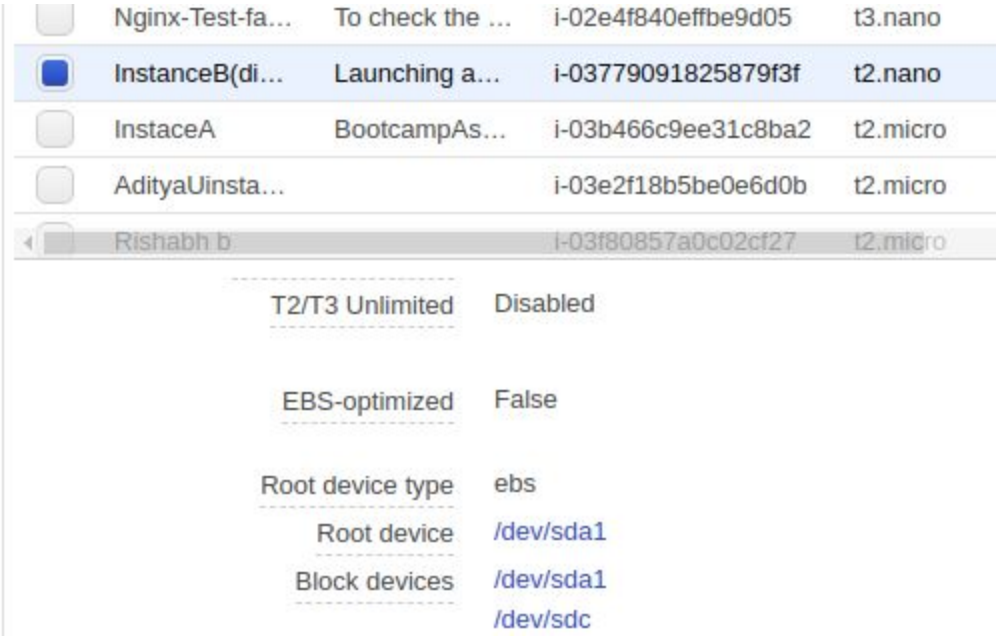


STEP 3: Click on the created volume and then go to Actions>Attach Volume



STEP 4: Copy and paste the instanc_ id of the instance on which you want to attach the volume.





****

4. Stop, Start, Restart that EBS (EBS must be auto-attached).

STEP 1: SSH into Instance B.Checking the block-devices and their mounting points.

```
ubuntu@ip-172-31-40-239:~$ lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0     7:0    0 89.1M  1 loop /snap/core/8268
loop1     7:1    0   18M  1 loop /snap/amazon-ssm-agent/1480
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0    8G  0 disk
ubuntu@ip-172-31-40-239:~$
```

*** We can see that /dev/xvda1 is the root (that has OS in it) and it is mounted to the root (“/”),**
but our attatched EBS is not mounted to any point (it is just the raw block).
*** Before mounting it, we have to provide a file-system to it.**

STEP 2: Create a file system using $mkfs.ext4 /dev/xvdb

```
root@ip-172-31-40-239:~# mkfs.ext4 /dev/xvdf
mke2fs 1.44.1 (24-Mar-2018)
/dev/xvdf contains a ext4 file system
        last mounted on Thu Feb 20 11:56:33 2020
Proceed anyway? (y,N) y
Creating filesystem with 2097152 4k blocks and 524288 inodes
Filesystem UUID: afddf037-4b28-4ff7-ab78-174ec14e38e0
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

root@ip-172-31-40-239:~# file -s /dev/xvdf
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=afddf037-4b28-4ff7-ab78-174e
c14e38e0 (extents) (64bit) (large files) (huge files)
root@ip-172-31-40-239:~#
```

STEP 3: Make a mount point “newvolume” in home
STEP 4: Mount the above file system

```
root@ip-172-31-40-239:~# mkdir /new_volume
root@ip-172-31-40-239:~# mount /dev/xvdf /new_volume
root@ip-172-31-40-239:~# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0     7:0    0 89.1M  1 loop /snap/core/8268
loop1     7:1    0   18M  1 loop /snap/amazon-ssm-agent/1480
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0    8G  0 disk /new_volume
root@ip-172-31-40-239:~#
```

*If we want to mount this EBS Volume on every startup, we must add an antry of this in
/etc/fstab. The default (root) EBS has the following entry in fstab.

```
root@ip-172-31-40-239:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            229M     0  229M   0% /dev
tmpfs            48M  760K   48M   2% /run
/dev/xvda1      7.7G  1.1G  6.7G  14% /
tmpfs           240M     0  240M   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           240M     0  240M   0% /sys/fs/cgroup
/dev/loop0       90M   90M     0 100% /snap/core/8268
/dev/loop1       18M   18M     0 100% /snap/amazon-ssm-agent/1480
tmpfs            48M     0   48M   0% /run/user/1000
/dev/xvdf       7.9G   36M  7.4G   1% /new_volume
root@ip-172-31-40-239:~#
```

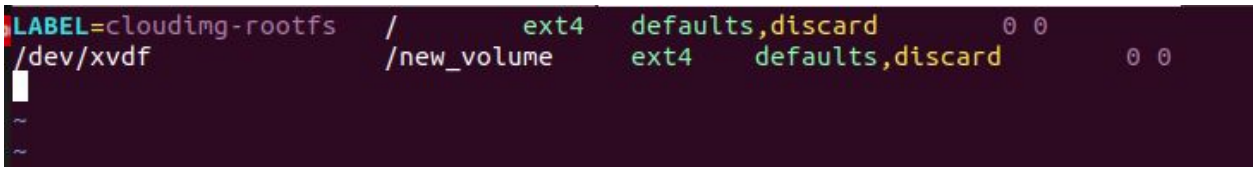
```
root@ip-172-31-40-239:~# cat /etc/fstab
LABEL=cloudimg-rootfs   /        ext4   defaults,discard        0 0
root@ip-172-31-40-239:~#
```

NOTE: /etc/fstab has 6 columns:
1- Device Name - LABEL=cloudimg-rootfs
2- Mount Point - /
3- File System - ext4
4- Mount Options - defaults , discard
5- Backup Operation - 0
6- File System Check Order - 0

Mount Option:
How the kernel treats the file system (Mount options can be more than one).
Backup Operation:
0 - No backup for the file system.
1 - Takes backup for the file ystem.
File System Check Order (done at boot):
0 - fsck should not check the file for any error (fsck – file checking command for reparing the
partitions)
1 - fsck Only check the root partition

2 – fsck Check the rest partition (other than root)

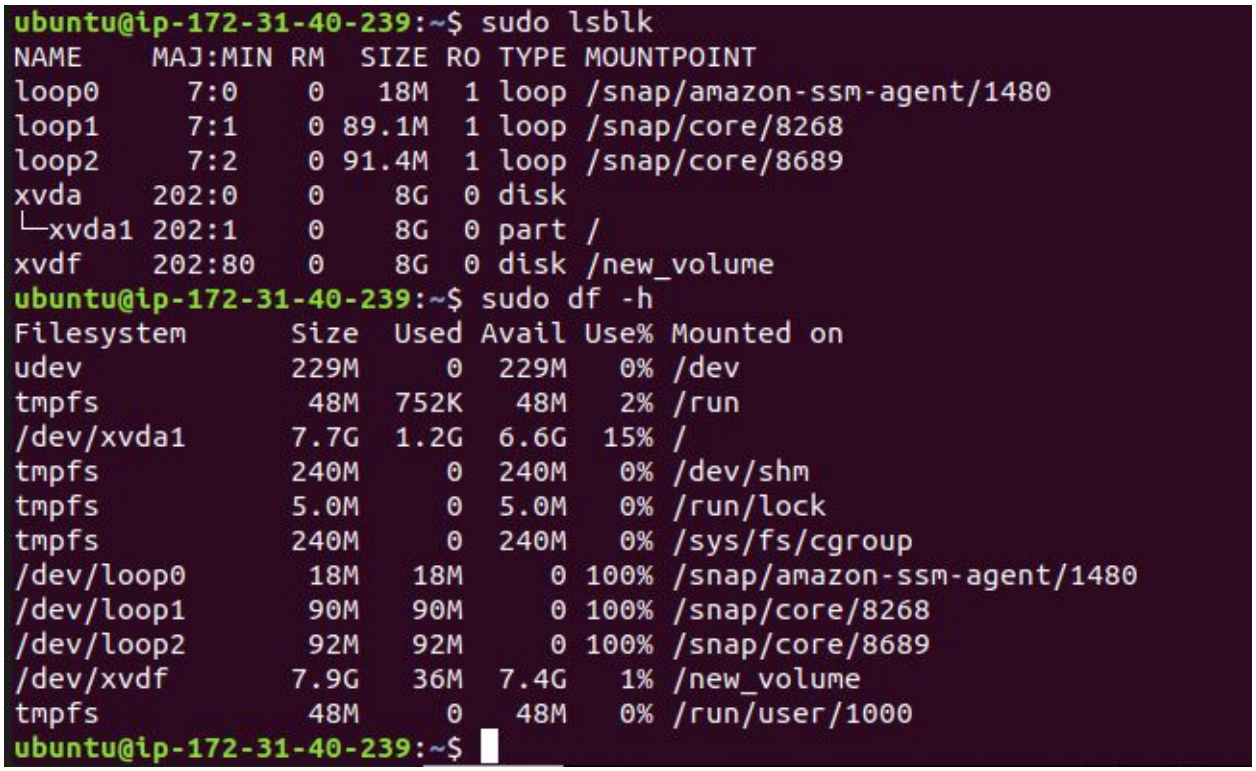
STEP 5: Entry in /etc/fstab file.



*** Now if we stop and start our instance, our EBS will be mounted automatically**

STEP 6: Stop the instance and then again start and check that the attached EBS is mounted

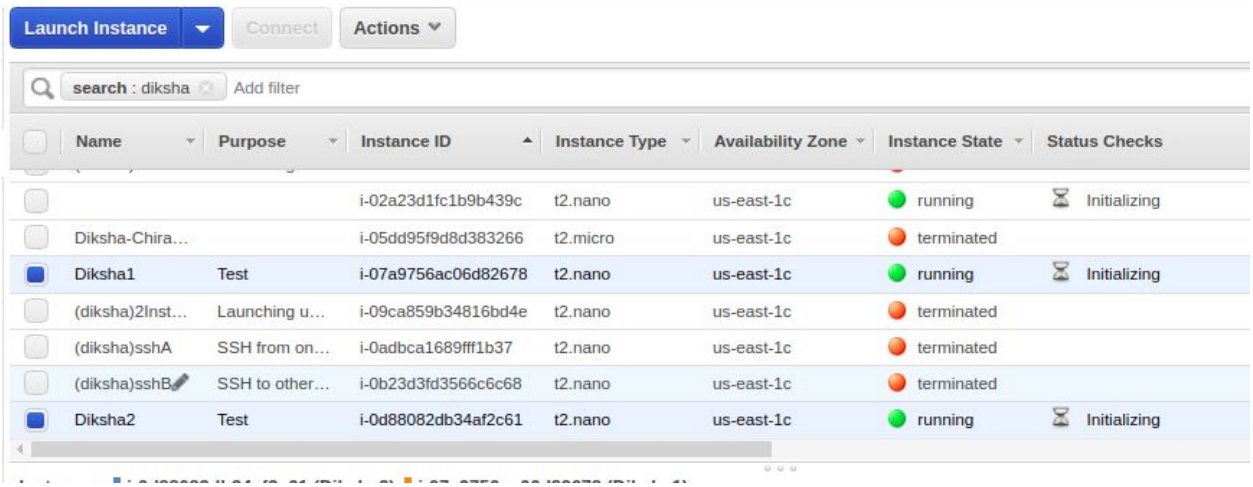
STEP 7: Starting again and verifying by using df -h and lsblk commands.



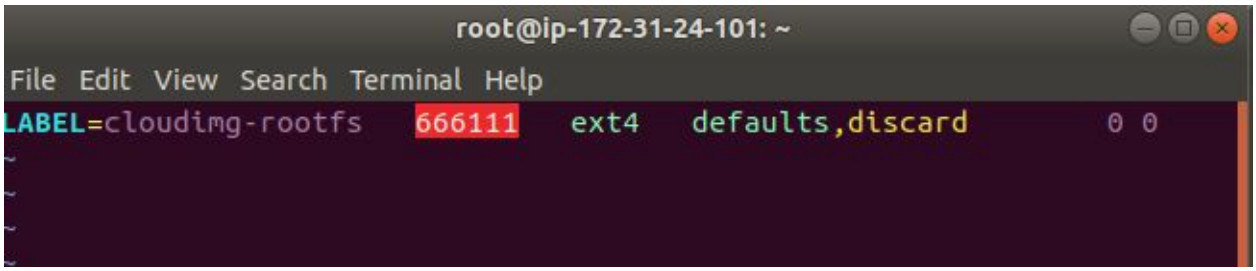
****

5. Make some mistake in fstab file, stop and start the instance, then troubleshoot it.

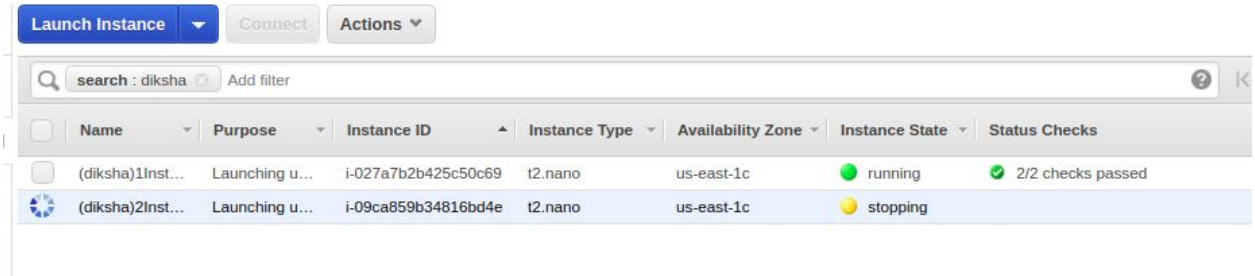
ANS:



STEP 1: Make errors in fstab file . Remove root's file extension(root: /*etc/*fstab)

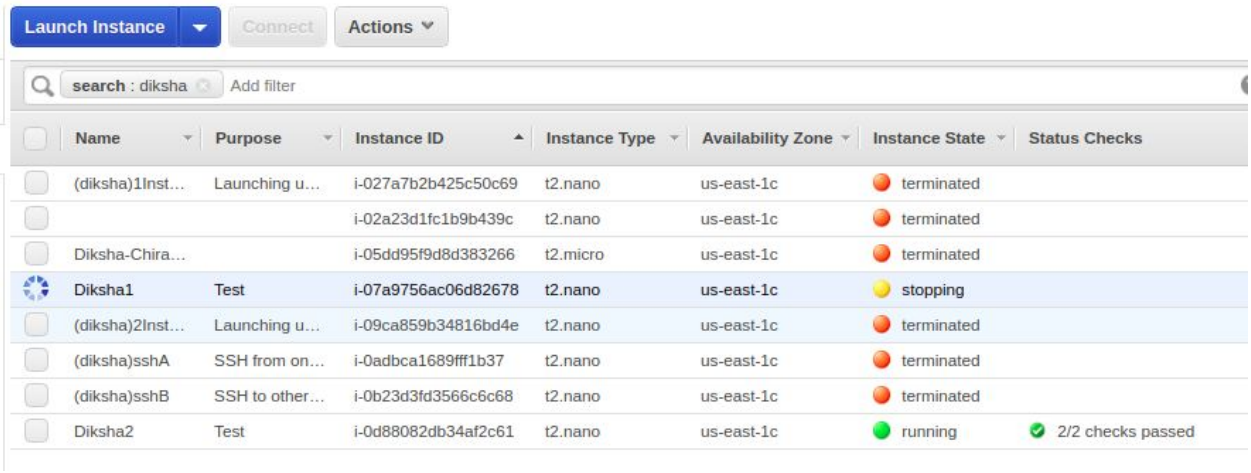


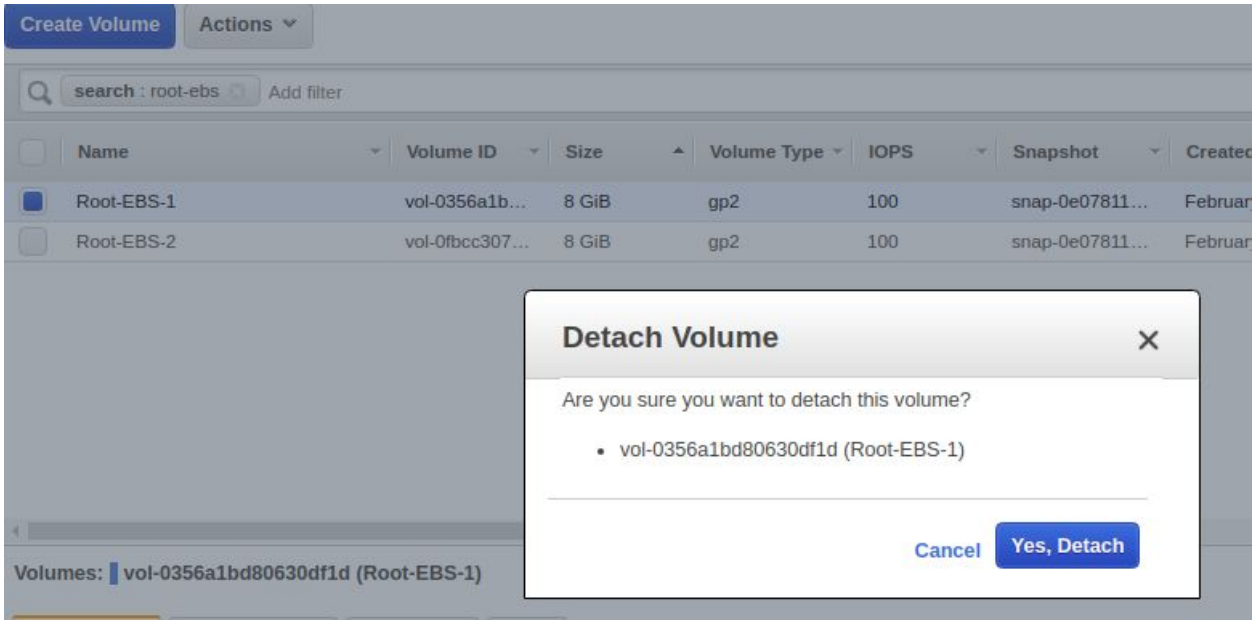
STEP 2: Now stop and start the Instance.(*** We noticed that instance is running at only "Read Only Mode", because of the error is file system in fstab.)**



```
ubuntu@ip-172-31-40-239:~$ Connection to ec2-52-91-106-106.compute-1.amazonaws.c
om closed by remote host.
Connection to ec2-52-91-106-106.compute-1.amazonaws.com closed.
diksha@diksha:~/Downloads$ ssh -i "diksha_awskey.pem" ubuntu@ec2-18-234-201-159.
compute-1.amazonaws.com
The authenticity of host 'ec2-18-234-201-159.compute-1.amazonaws.com (18.234.201
.159)' can't be established.
ECDSA key fingerprint is SHA256:0CgewlRboDNN/sr0P2ff6MGRWM+P/GavxkJZXmWuH5A.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-18-234-201-159.compute-1.amazonaws.com,18.234.20
1.159' (ECDSA) to the list of known hosts.
Last login: Thu Feb 20 17:51:57 2020 from 182.69.242.57
ubuntu@ip-172-31-40-239:~$ sudo bash
sudo: unable to resolve host ip-172-31-40-239: Resource temporarily unavailable
root@ip-172-31-40-239:~# cat >file
bash: file: Read-only file system
root@ip-172-31-40-239:~#
```

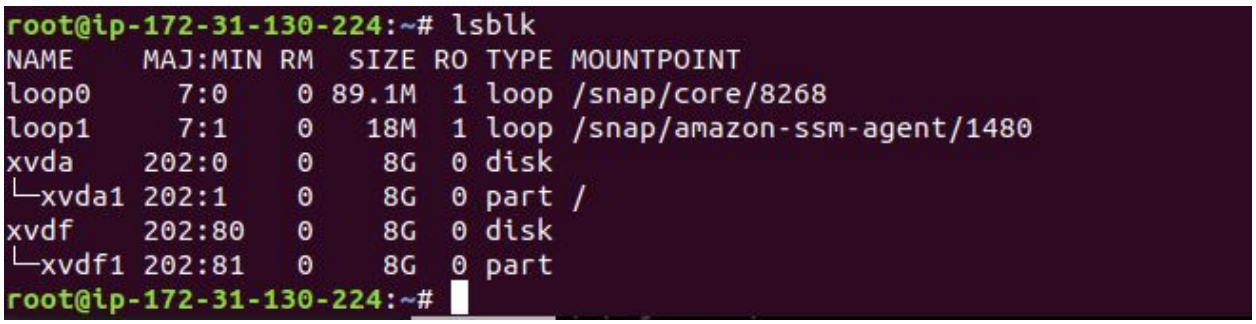
STEP 3: Stop the instance again and then detatch the root EBS for correcting the file system.(Detatch volume from instance 1
)



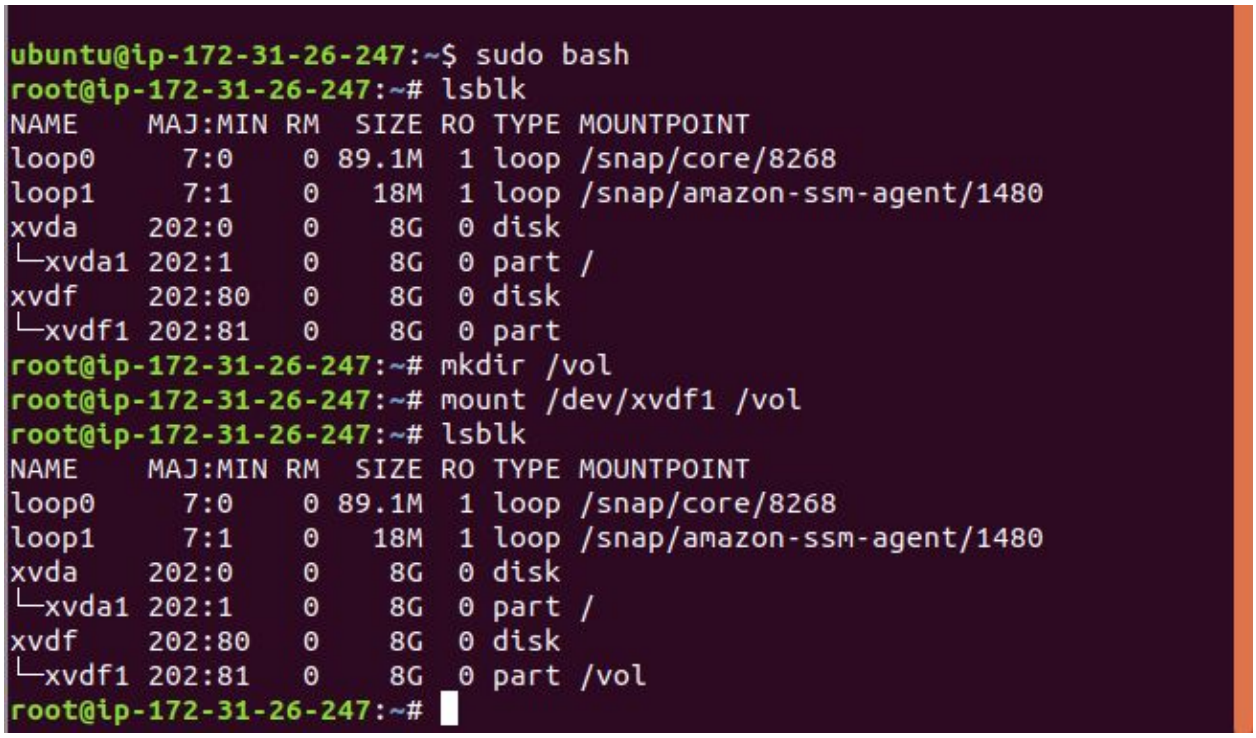


STEP 4: Attach that root EBS to the other “InstanceA” as secondary EBS, and see the status in console.(Attach in Instance 2 as secondary EBS)

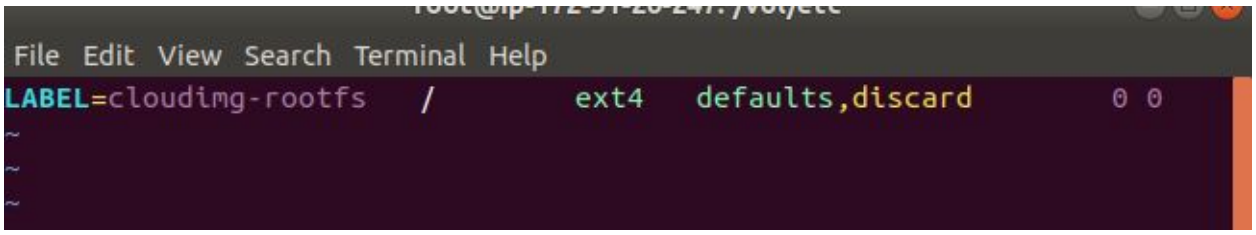
STEP 5: SSH into “InstanceA” and list all block devices.



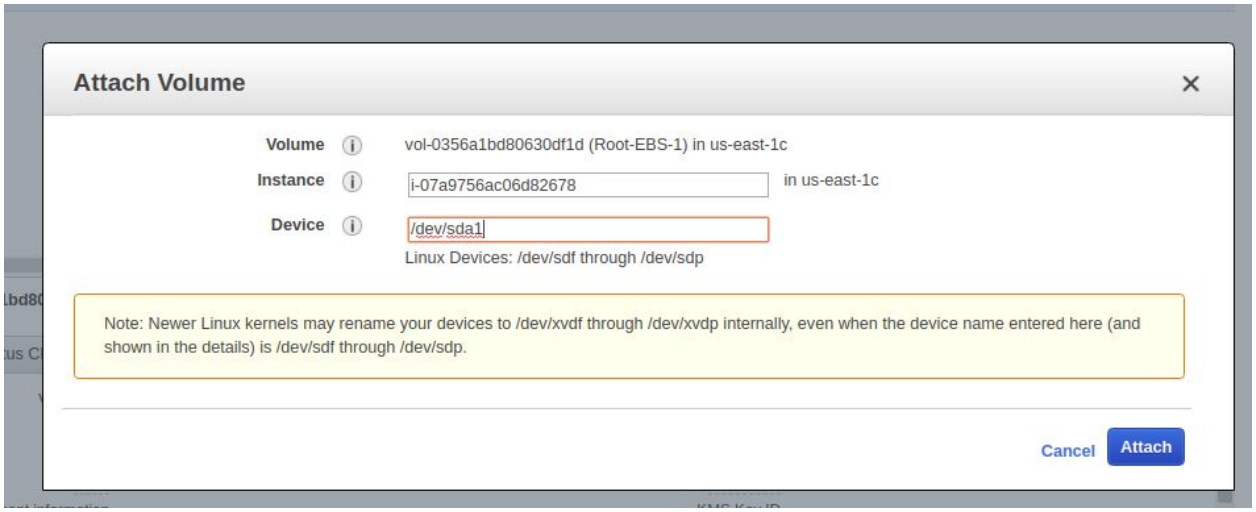
STEP 6: Mount that EBS in “Instance2” (checking file system also)



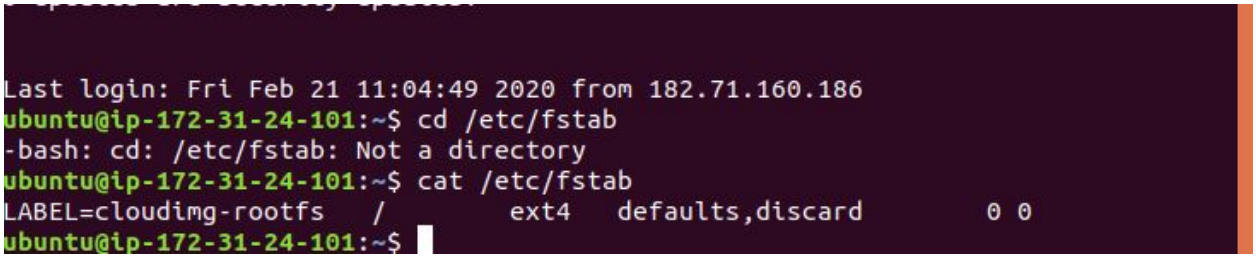
STEP 7: Correcting the fstab file of the mounted disk here.



STEP 8: Now detach this EBS from “Instance2” and attach back to original “Instance1”.

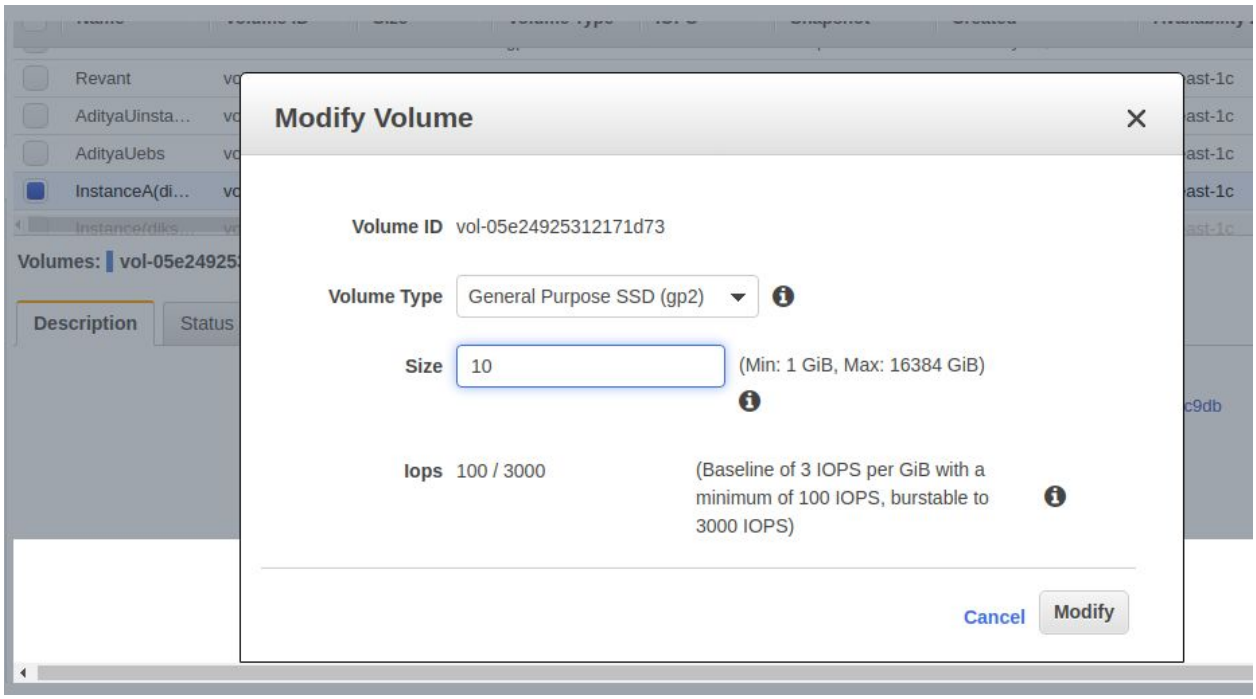

Attach Volume
Volume
vol-0356a1bd80630df1d (Root-EBS-1) in us-east-1c
Instance
i-07a9756ac06d82678
in us-east-1c
Device
/dev/sda1
Linux Devices: /dev/sdf through /dev/sdp
Note: Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.
Cancel
Attach

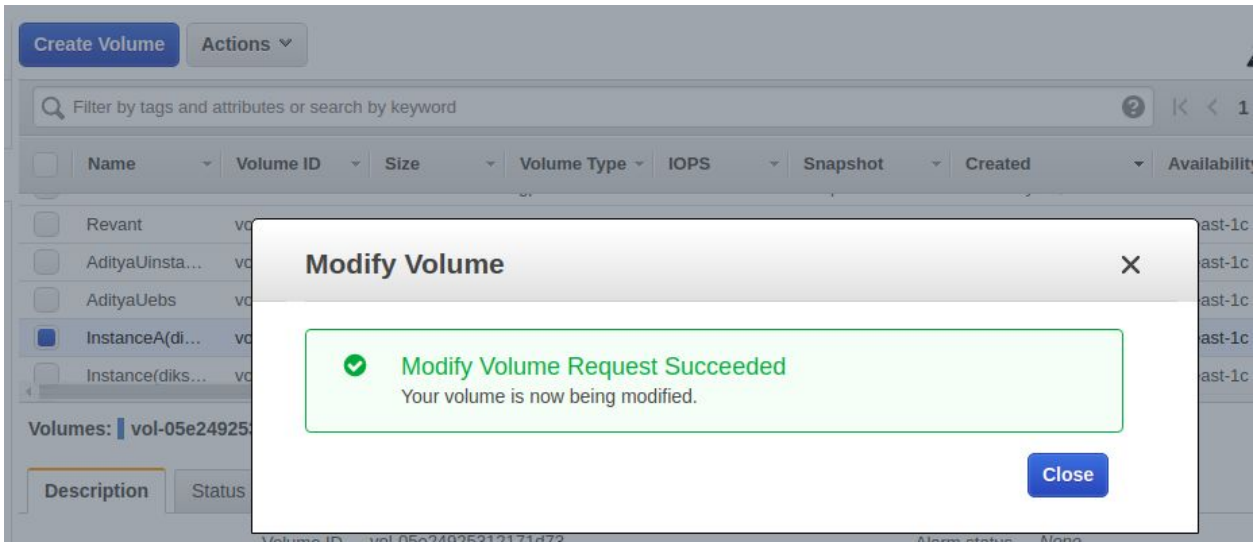
STEP 9: Start “Instance1” and SSH into it ,checking whether the file system is mounted perfectly or not (It is perfectly fine).



****

6. Resize the EBS from 8 to 10GB





InstanceA(di...	vol-05e2492...	10 GiB	gp2	100	snap-0e07811...	February 20, 2020 ...	us-east-1c

****

7. SSH from one instance A to instance B.

STEP 1: Copy pem file to your instance.

```
diksha@diksha:~/Downloads$ scp -i diksha_awskey.pem diksha_awskey.pem ubuntu@ec2
-184-72-121-108.compute-1.amazonaws.com:
diksha_awskey.pem                             100% 1692     5.5KB/s   00:00
diksha@diksha:~/Downloads$
```

STEP 2: Now ssh into the instance

```
diksha@diksha:~/Downloads$ ssh -i "diksha_awskey.pem" ubuntu@ec2-184-72-121-108.
compute-1.amazonaws.com
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Feb 21 04:58:35 UTC 2020

  System load:  0.01              Processes:           89
  Usage of /:   15.0% of 7.69GB   Users logged in:     0
  Memory usage: 30%               IP address for eth0: 172.31.155.240
  Swap usage:   0%

 * Multipass 1.0 is out! Get Ubuntu VMs on demand on your Linux, Windows or
   Mac. Supports cloud-init for fast, local, cloud devops simulation.
```

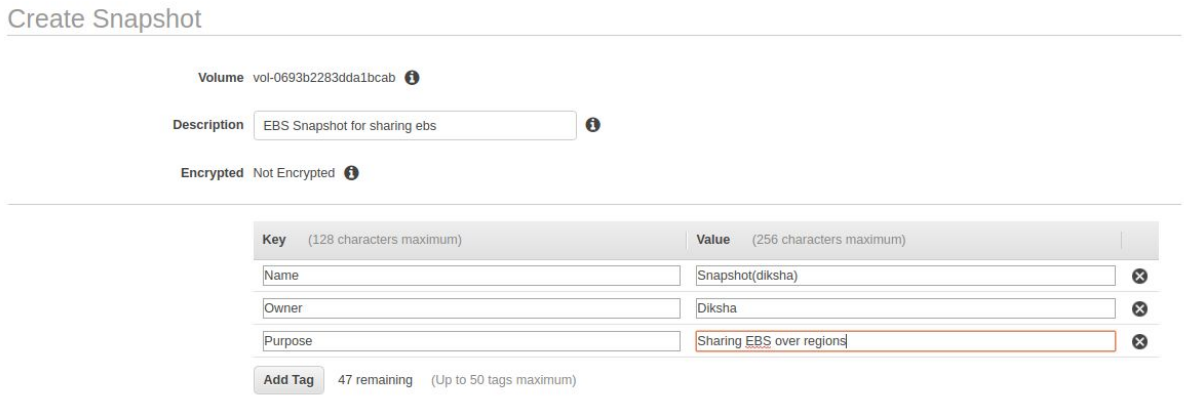
STEP 3: Then ssh into *Server B* from A

```
ubuntu@ip-172-31-155-240:~$ ls
diksha_awskey.pem
ubuntu@ip-172-31-155-240:~$ ssh -i diksha_awskey.pem  ubuntu@172.31.20.8
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1101-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Multipass 1.0 is out! Get Ubuntu VMs on demand on your Linux, Windows or
   Mac. Supports cloud-init for fast, local, cloud devops simulation.

     https://multipass.run/

0 packages can be updated.
0 updates are security updates.

New release '18.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

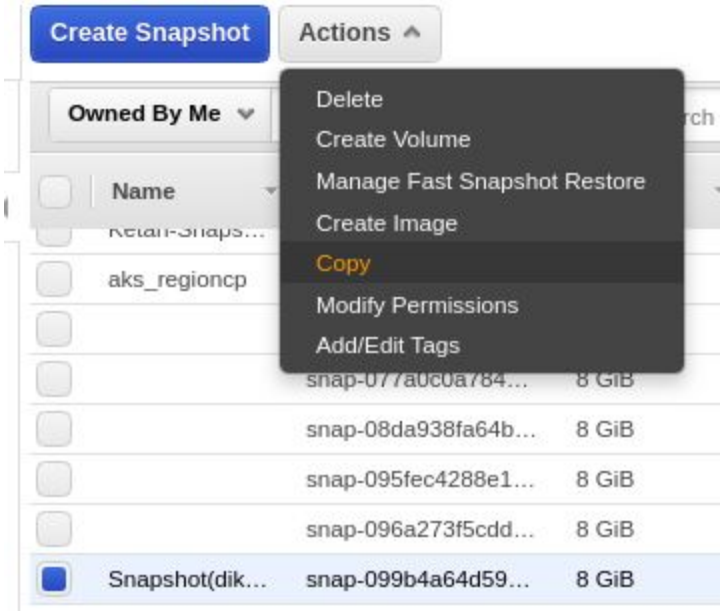
****

8. Copy the EBS in different region(oregon).

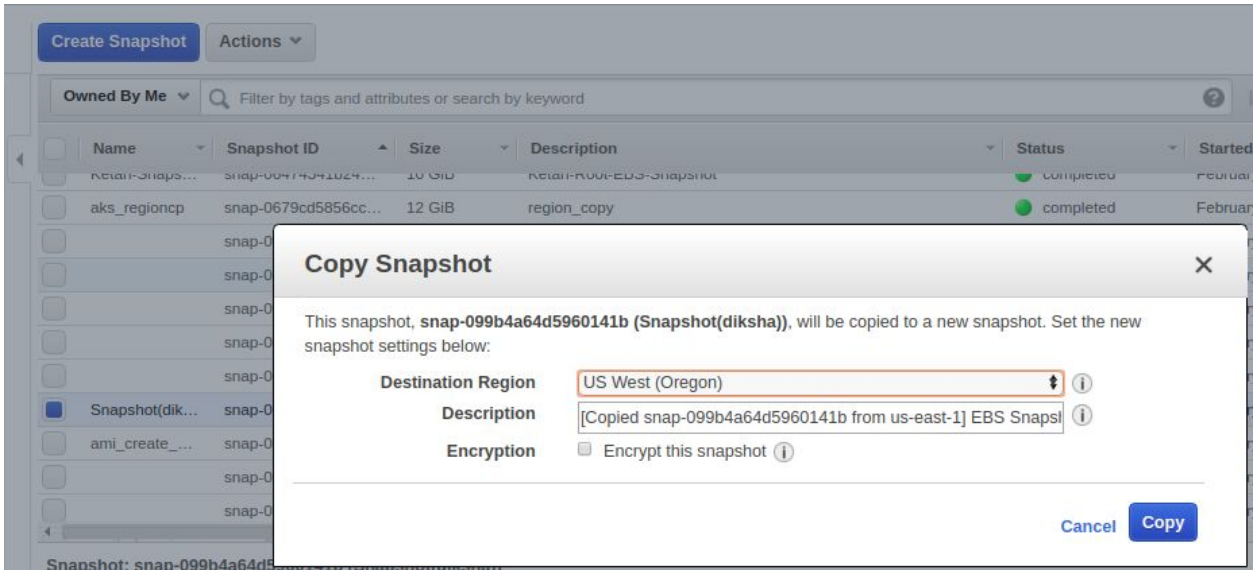
STEP 1: First Create Snapshot of the EBS.



STEP 2: Copy Snapshot.



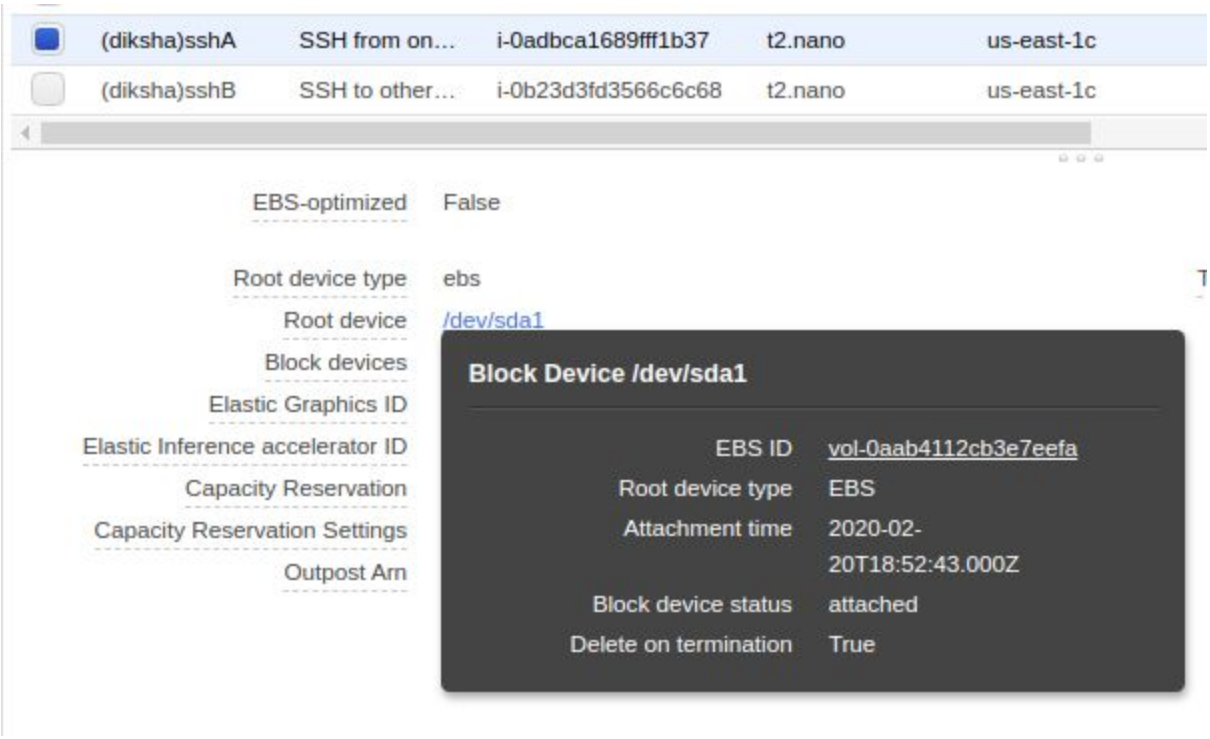
STEP 3: Give the region to which it will be copied (In our case it is Oregon).



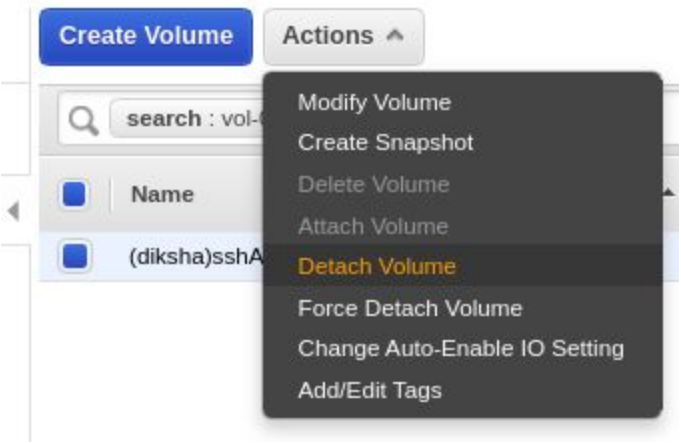
****

9. Detach the root EBS, create its snapshot, than create the AMI and run it as instance such that nginx should be preinstalled at the boot time of instance.

STEP 1: Detach the root EBS



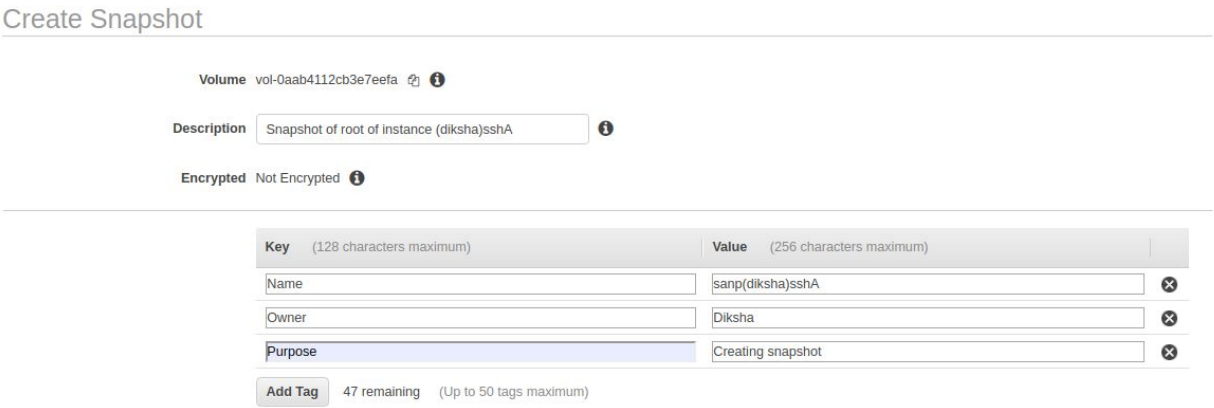
STEP 2: Detach volume



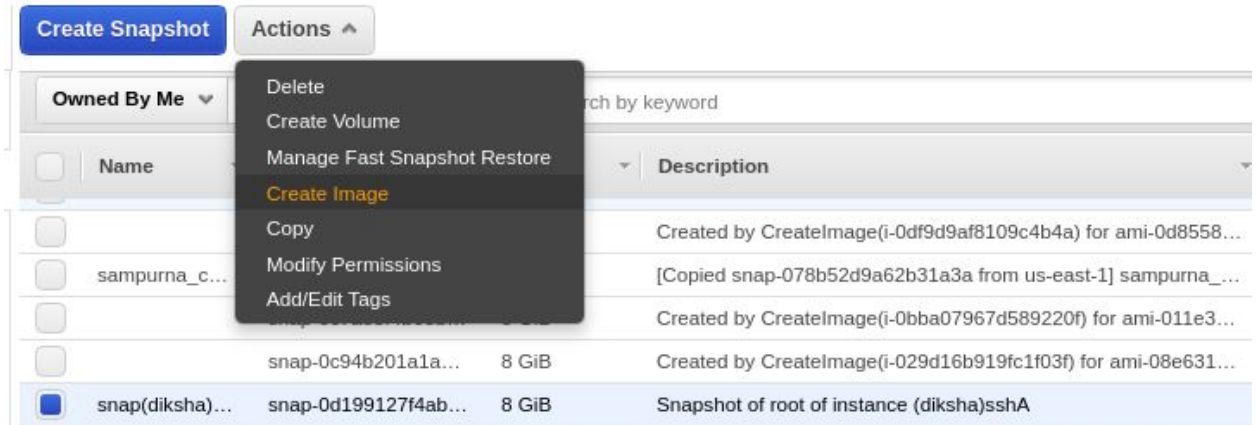
STEP 3:

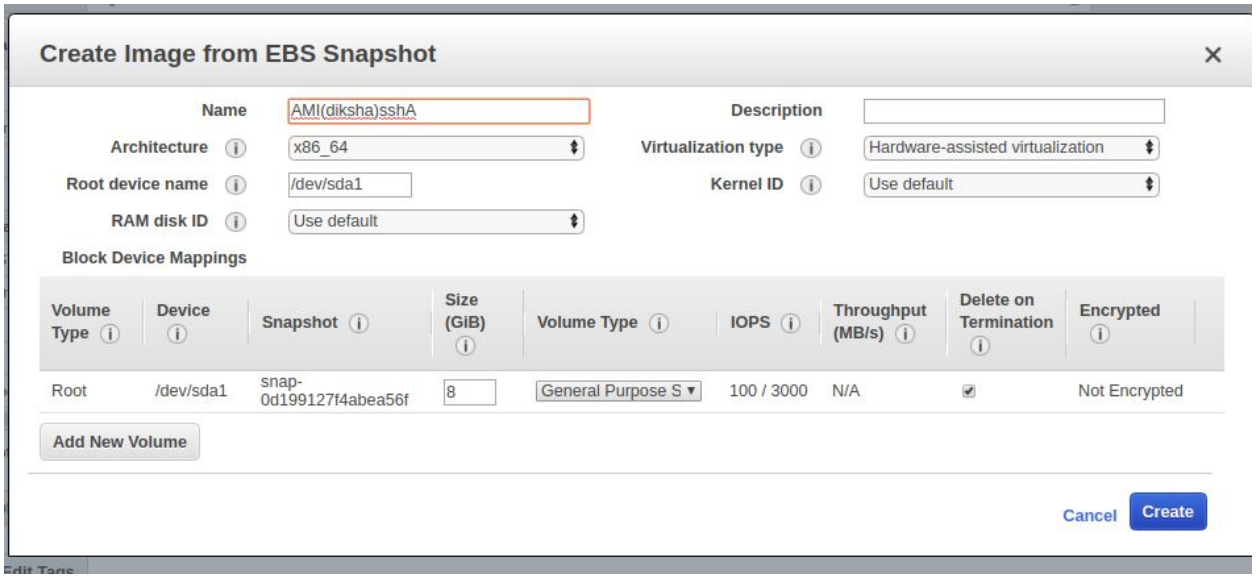
STEP 4: Create its snapshot

STEP 5:

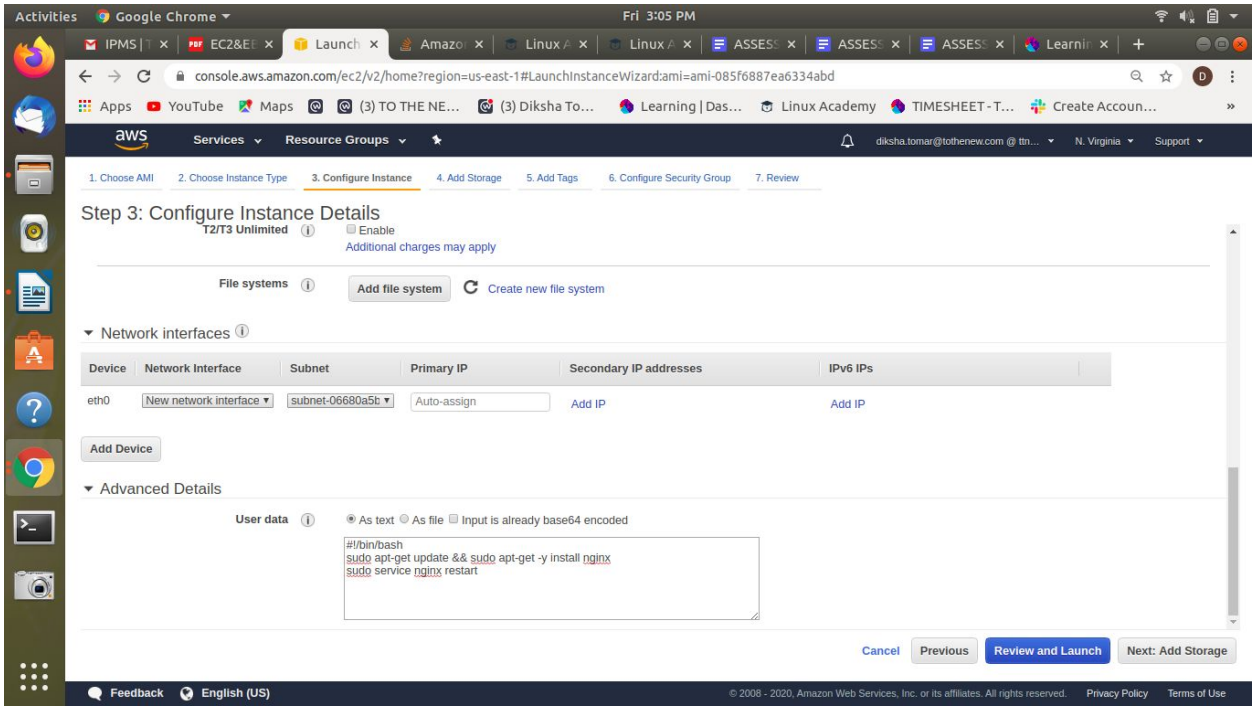


STEP 6: Create AMI





STEP 7: Launch instance with this AMI and save it with userdata file having installation of nginx.



Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	
Name	(diksha)nginxInstancesshA	✓	✓	⊗
Owner	Diksha	✓	✓	⊗
Purpose	Launching instance with installed nginx	✓	✓	⊗

Add another tag (Up to 50 tags maximum)

```
diksha@diksha:~/Downloads$ sudo ssh -i "diksha_awskey.pem" ubuntu@ec2-52-91-24-1
88.compute-1.amazonaws.com
[sudo] password for diksha:
The authenticity of host 'ec2-52-91-24-188.compute-1.amazonaws.com (52.91.24.188
)' can't be established.
ECDSA key fingerprint is SHA256:ULWfItfIQeOvS4YT9hodOsXlwanSJT/NHtt2qjUtA+E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-52-91-24-188.compute-1.amazonaws.com,52.91.24.18
8' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
```

```
ubuntu@ip-172-31-74-80:~$
ubuntu@ip-172-31-74-80:~$ sudo service nginx status
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: en
   Active: active (running) since Fri 2020-02-21 09:37:15 UTC; 18s ago
     Docs: man:nginx(8)
  Process: 1835 ExecStop=/sbin/start-stop-daemon --quiet --stop --retry QUIT/5 -
  Process: 1849 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code
  Process: 1838 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process
 Main PID: 1853 (nginx)
    Tasks: 2 (limit: 1152)
   CGroup: /system.slice/nginx.service
           ├─1853 nginx: master process /usr/sbin/nginx -g daemon on; master_pro
           └─1855 nginx: worker process

Feb 21 09:37:15 ip-172-31-74-80 systemd[1]: Starting A high performance web serv
Feb 21 09:37:15 ip-172-31-74-80 systemd[1]: nginx.service: Failed to parse PID f
Feb 21 09:37:15 ip-172-31-74-80 systemd[1]: Started A high performance web serve
lines 1-16/16 (END)
```