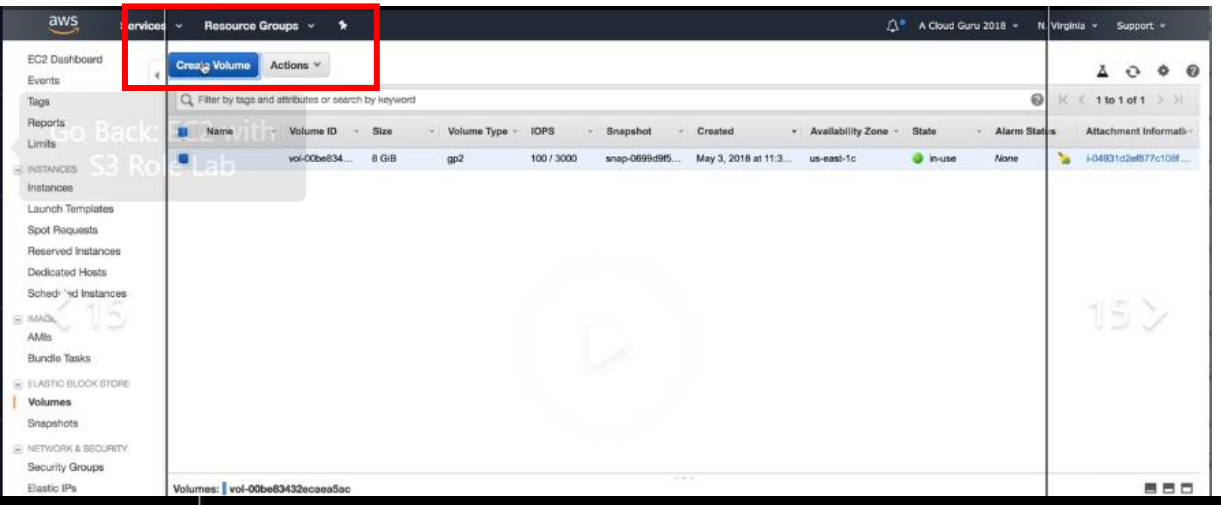
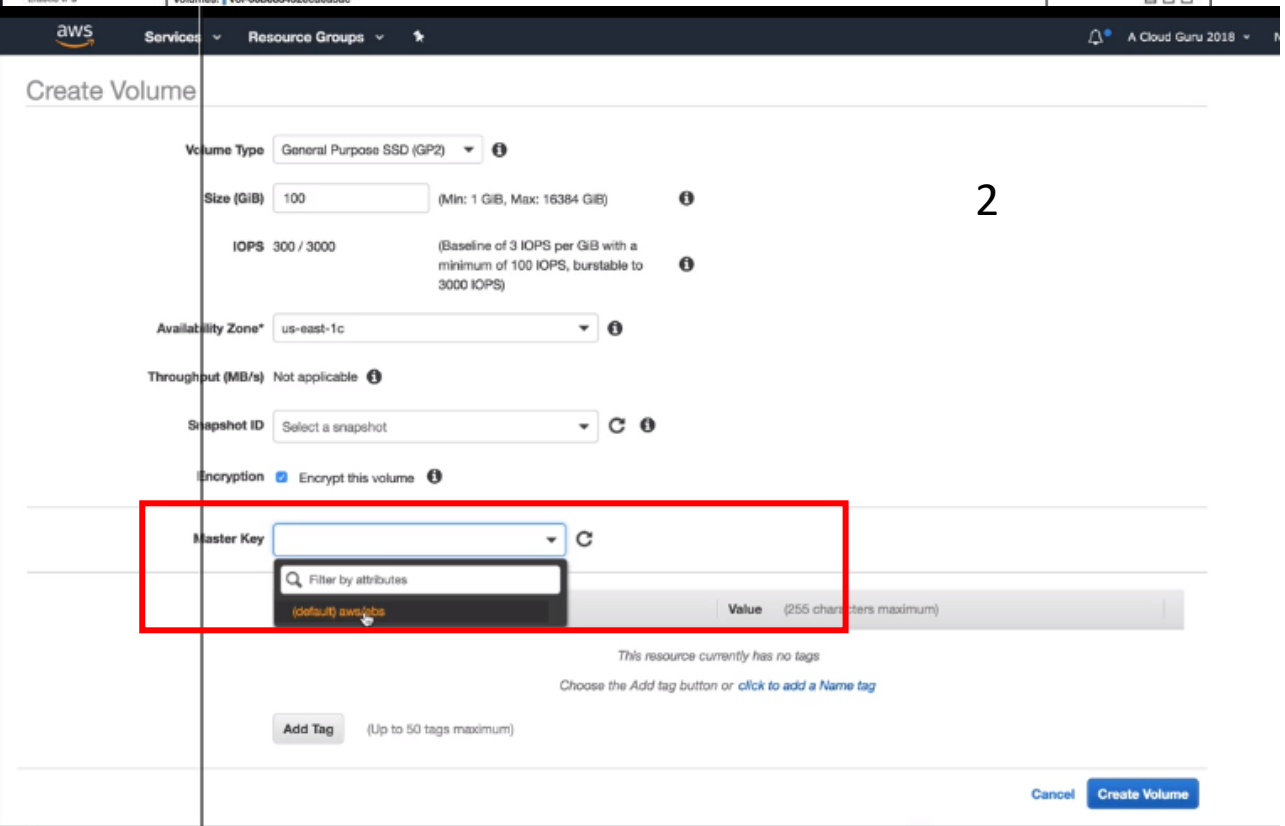
Creating the volume

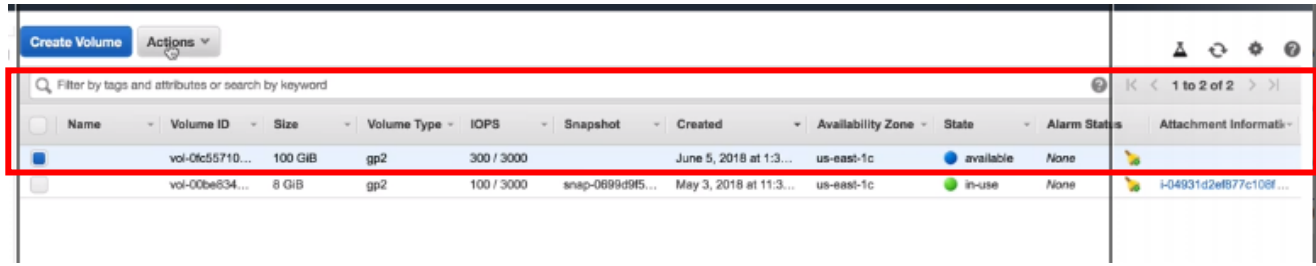
1



2

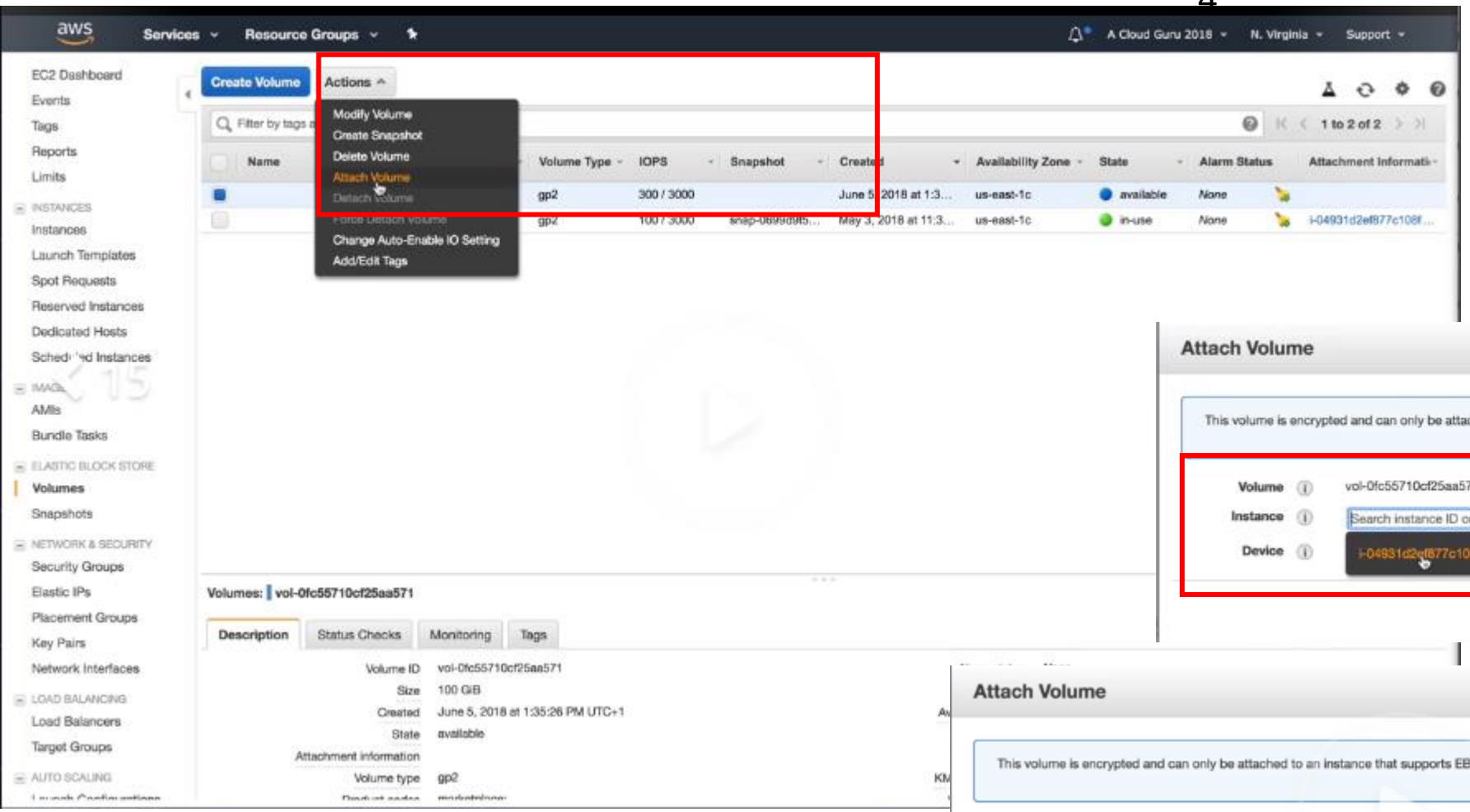


3



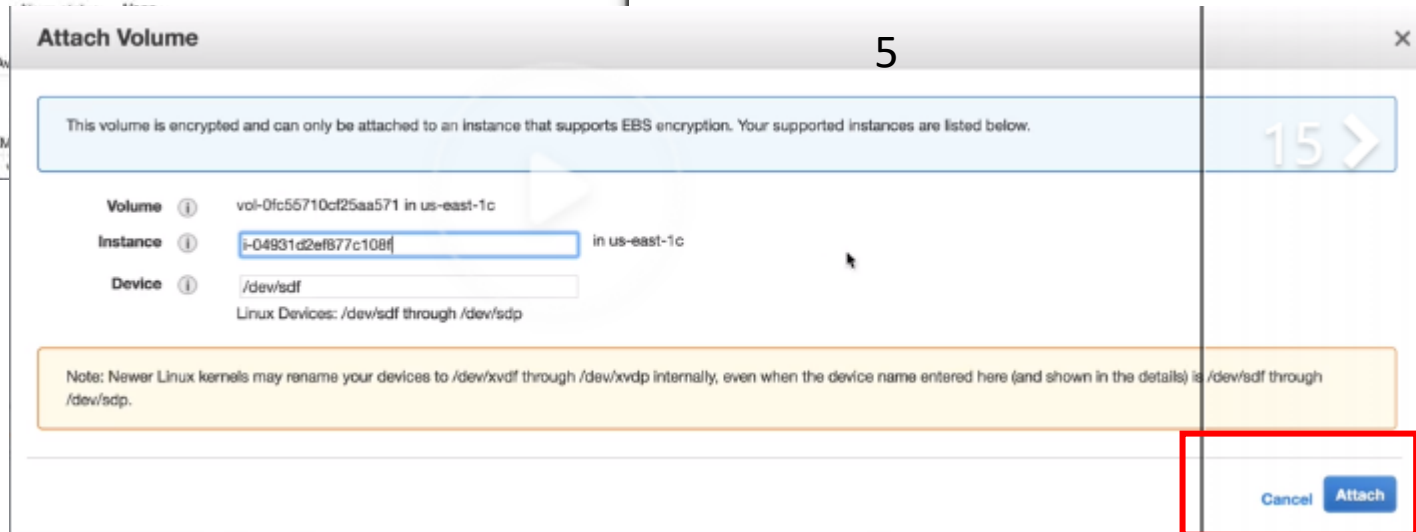
4

Attaching volume to Ec2



5

5



1

```
Ryans-iMac:Downloads acloudguru$ ssh ec2-user@35.153.184.58 -i MyNewKeyPair.pem
Last login: Tue Jun  5 10:27:17 2018 from 90.152.124.231

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
17 package(s) needed for security, out of 19 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-36-54 ~]$ sudo su
[root@ip-172-31-36-54 ec2-user]#
```

Creating a filesystem on the volume

3

```
[root@ip-172-31-36-54 ec2-user]# mkfs -t ext4 /dev/xvdf
mke2fs 1.42.12 (29-Aug-2014)
Creating filesystem with 26214400 4k blocks and 6553600 inodes
Filesystem UUID: 2276aec8-2a6e-4011-be2f-29f916e14f2e
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[root@ip-172-31-36-54 ec2-user]# file -s /dev/xvdf
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=2276aec8-2a6e-4011-be2f-29f916e14f2e (extents) (large files) (huge files)
```

List volumes

2

```
[root@ip-172-31-36-54 ec2-user]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk
[root@ip-172-31-36-54 ec2-user]# file -s /dev/xvdf
/dev/xvdf: data
```

Check if there is filesystem in the volume

```
[root@ip-172-31-36-54 ec2-user]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk
[root@ip-172-31-36-54 ec2-user]# cd /
[root@ip-172-31-36-54 /]# mkdir filesystem
[root@ip-172-31-36-54 /]# mount /dev/xvdf /filesystem
[root@ip-172-31-36-54 /]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk /filesystem
[root@ip-172-31-36-54 /]# cd filesystem
[root@ip-172-31-36-54 filesystem]# echo "Hello Cloud Gurus" > hello.txt
[root@ip-172-31-36-54 filesystem]# ls
hello.txt  lost+found
```

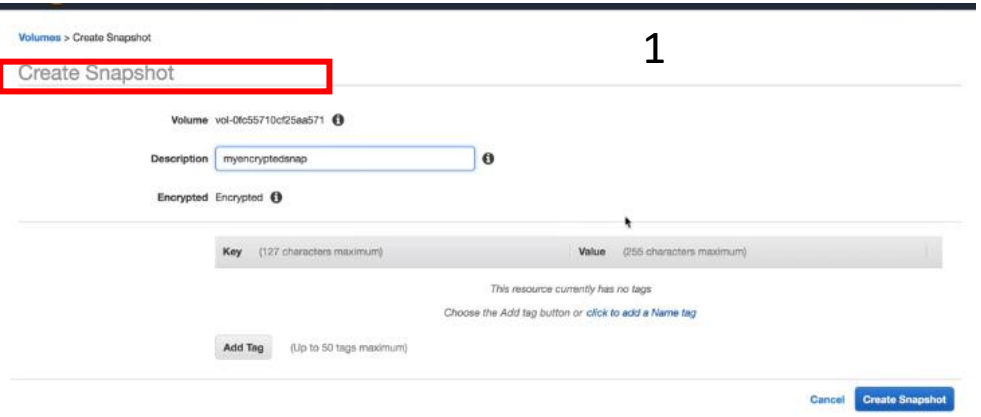
Mounting volume to the filesystem

Unmounting the volume to file system

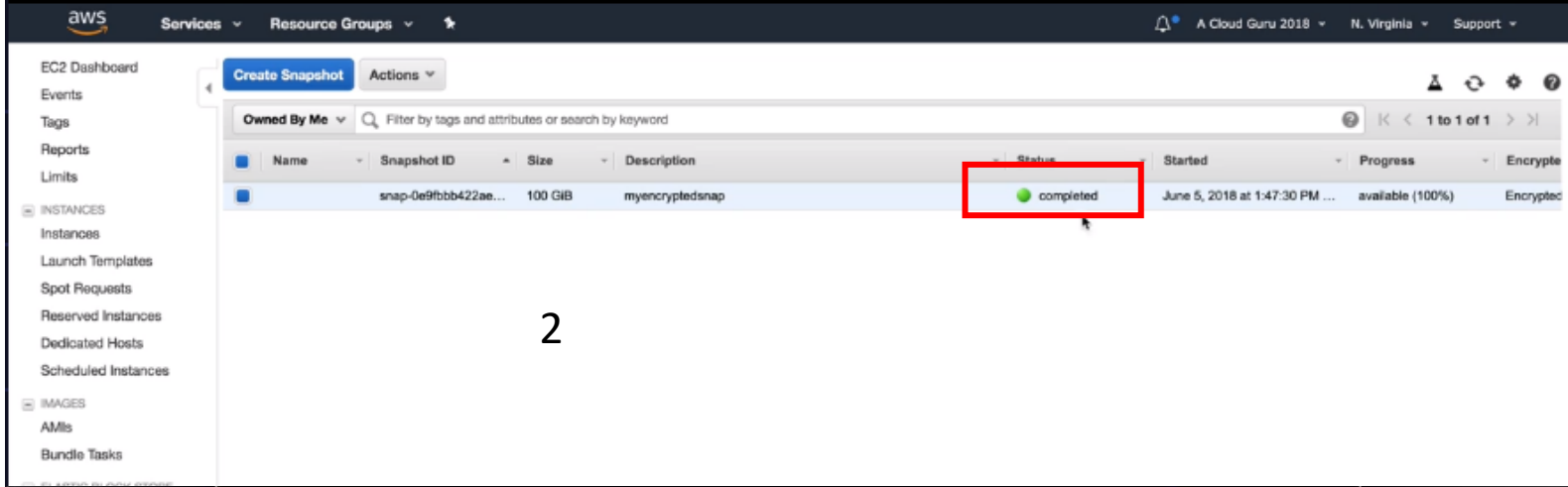
```
[root@ip-172-31-36-54 filesystem]# cd /
[root@ip-172-31-36-54 /]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk /filesystem
[root@ip-172-31-36-54 /]# umount -d /dev/xvdf
[root@ip-172-31-36-54 /]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk
[root@ip-172-31-36-54 /]#
```

- Detach the volume
- Restore it with the snapshot
- Mount back
- Able to access the txt file

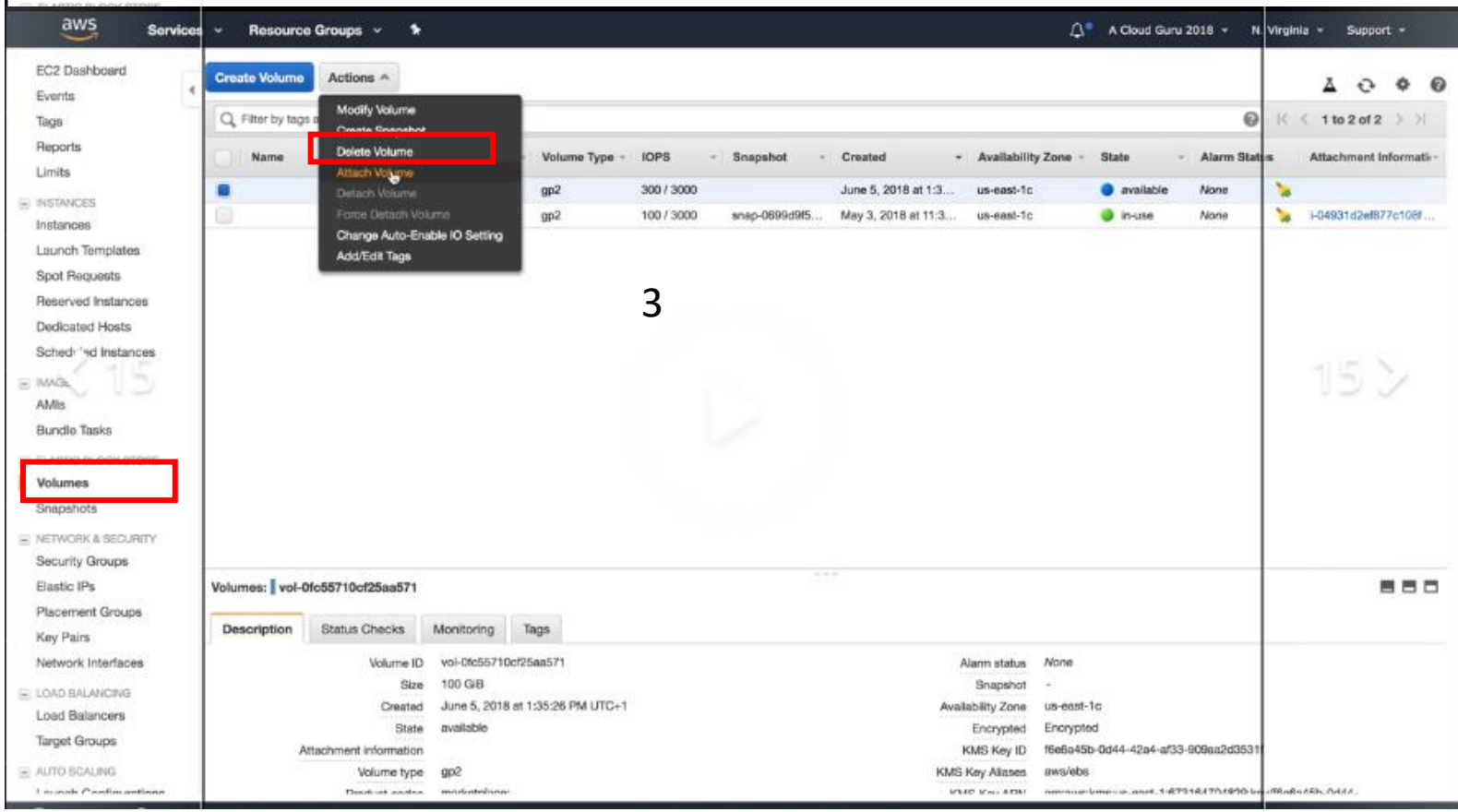
1. EC2 instance → Actions → detach volume (unmount the volume) → status should be available
2. Actions → create snapshots



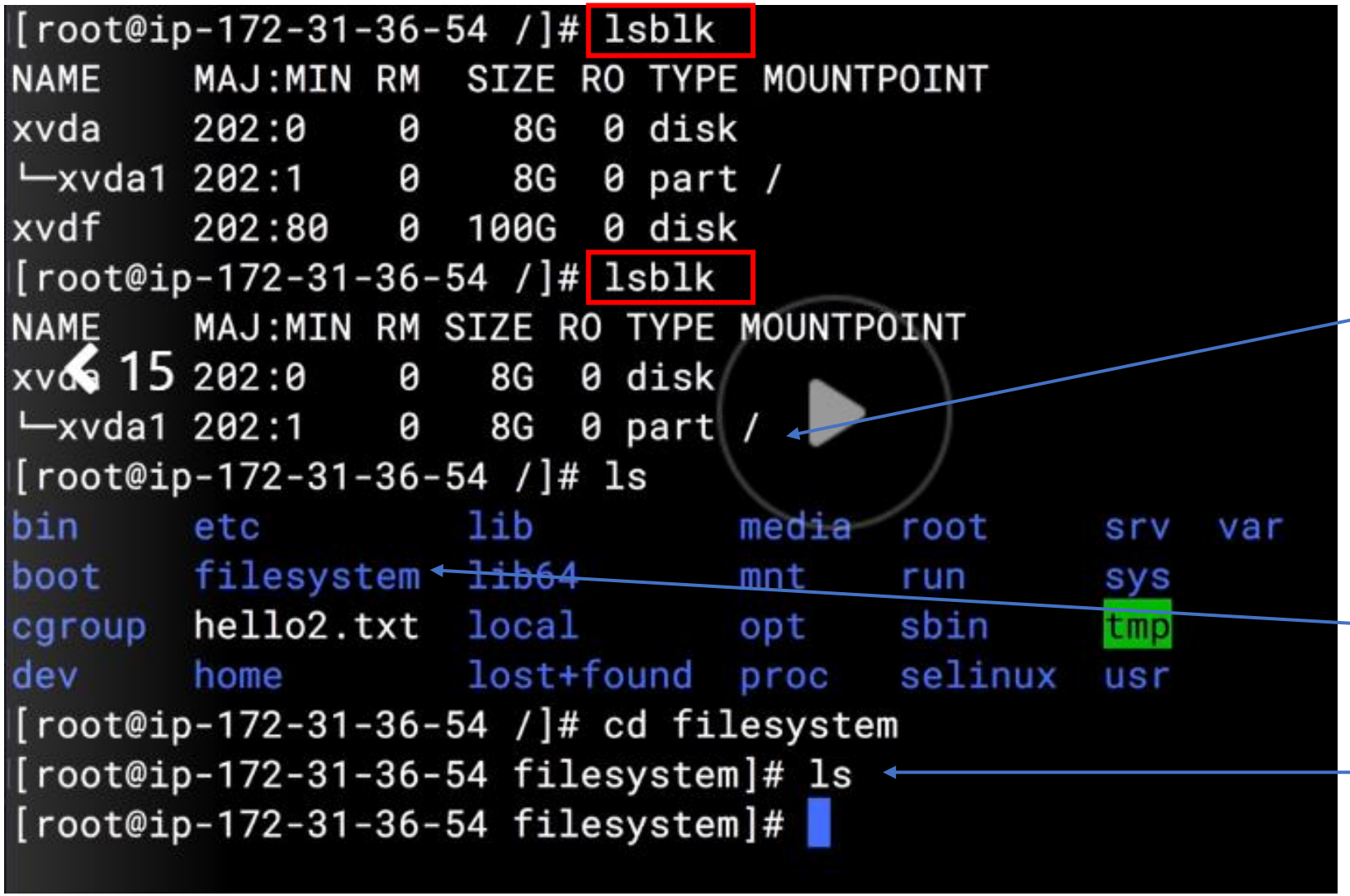
1



2



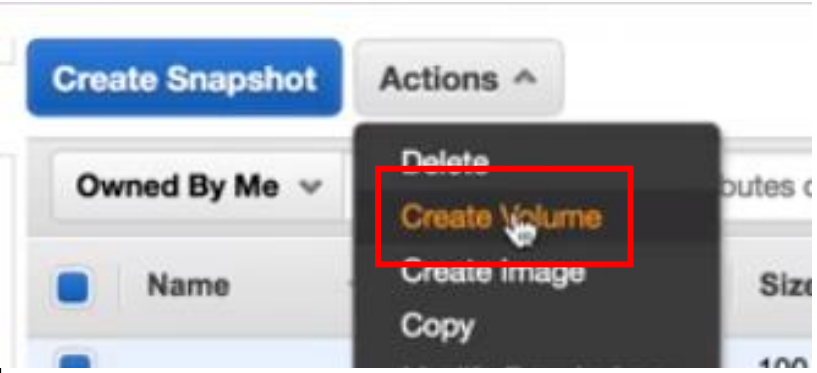
3

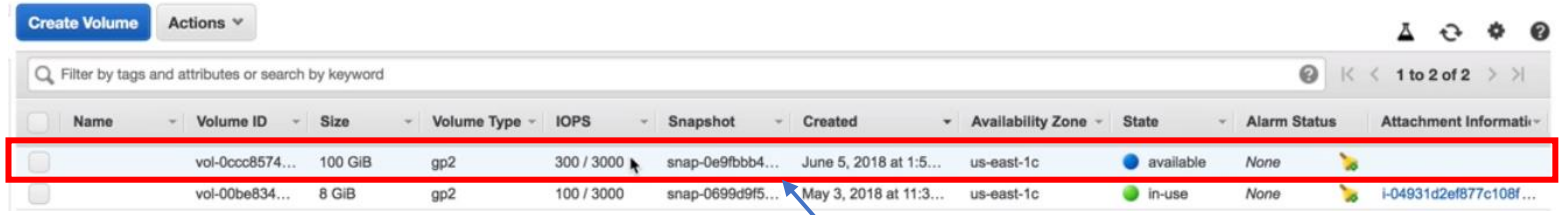


We cant see xvdf

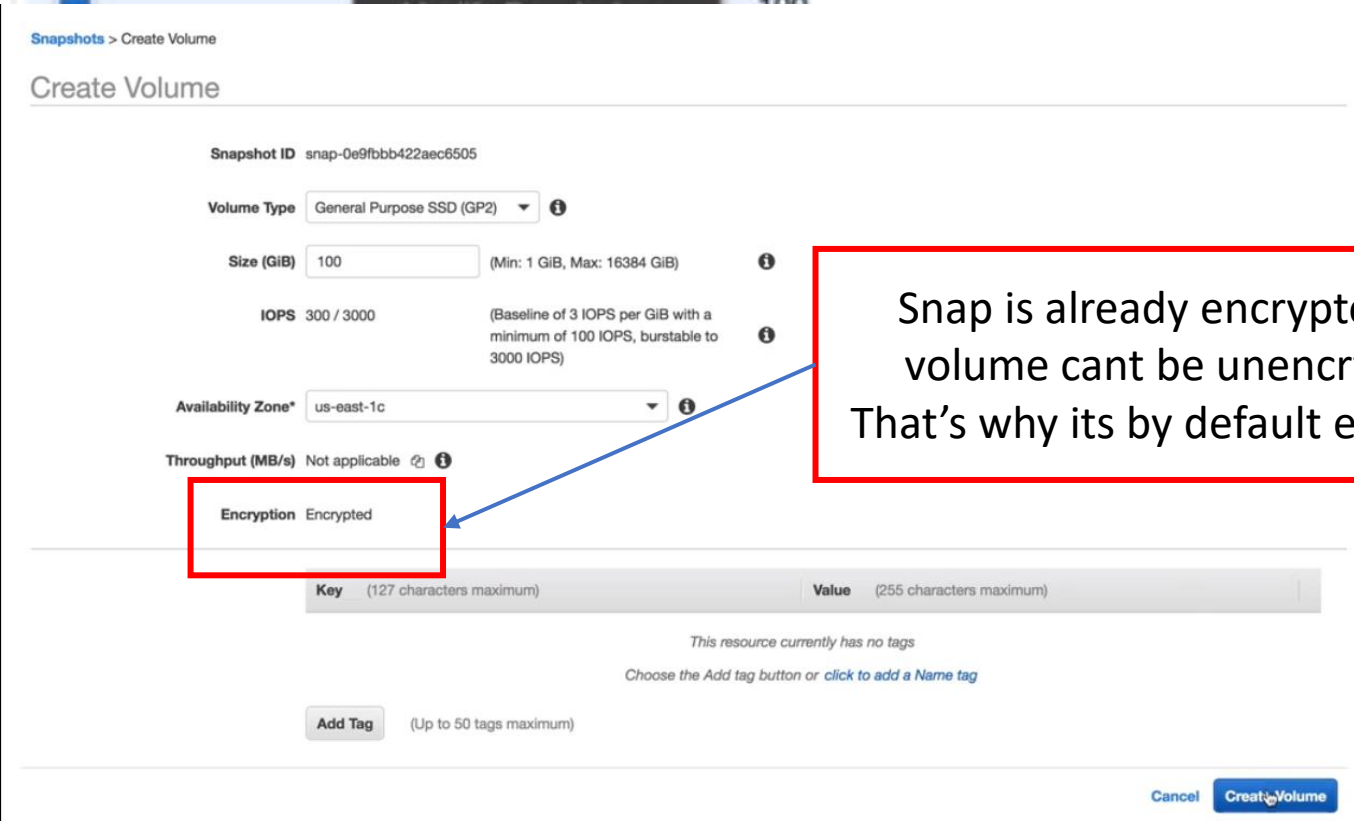
We can see folder filesystem but content of it is not visible as the volume is unmounted / detached

Snapshot → actions → create volume

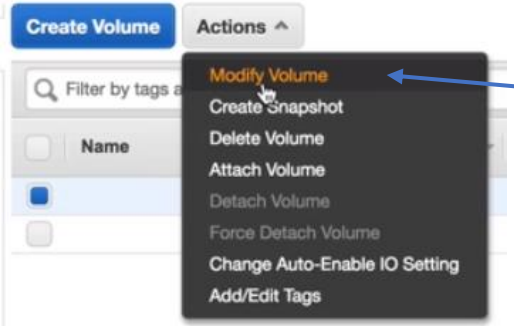




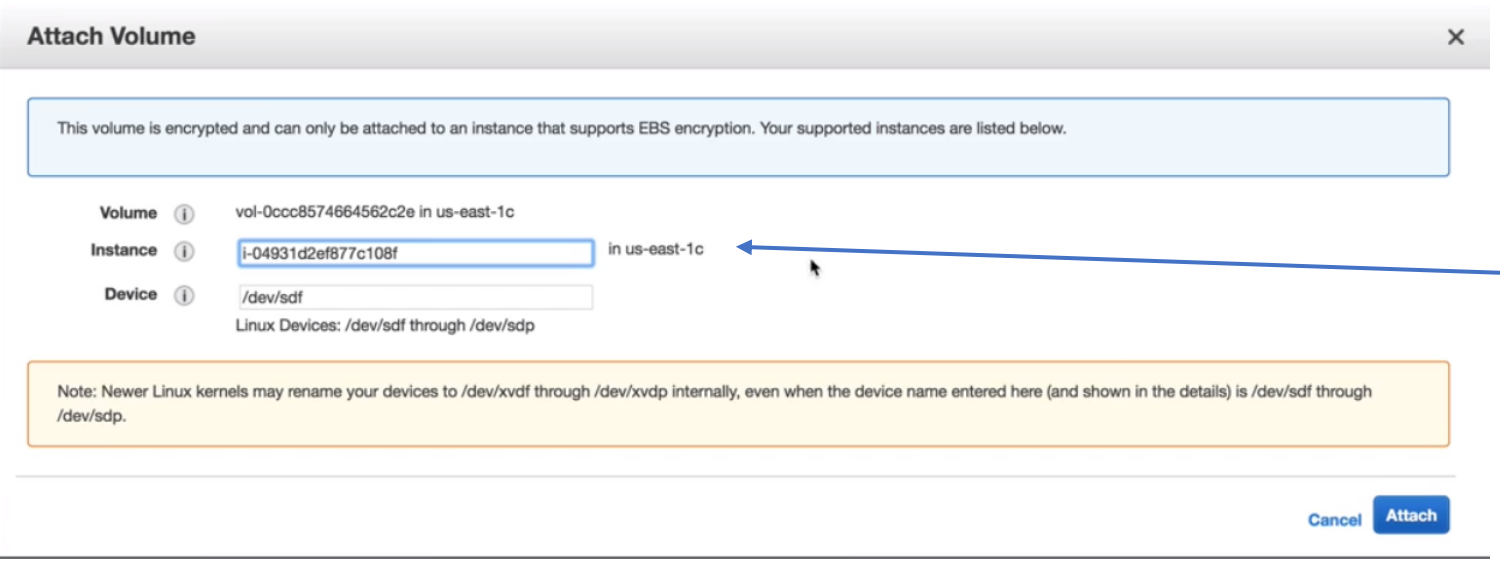
Go back to volume and status is available



Snap is already encrypted and volume cant be unencrypted. That's why its by default encrypted

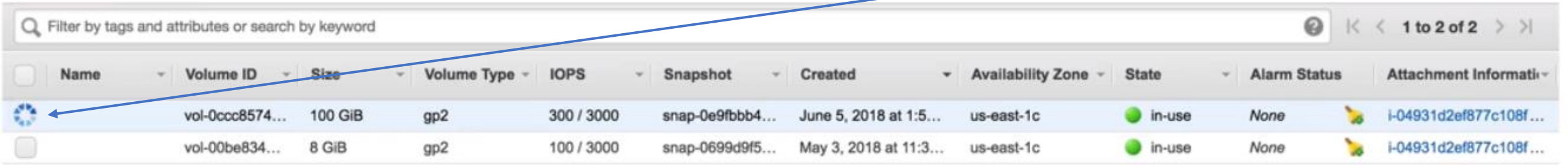


Select the volume and select modify volume



Provide instance id and the region same as ec2 instance

Attached to the ec2 instance



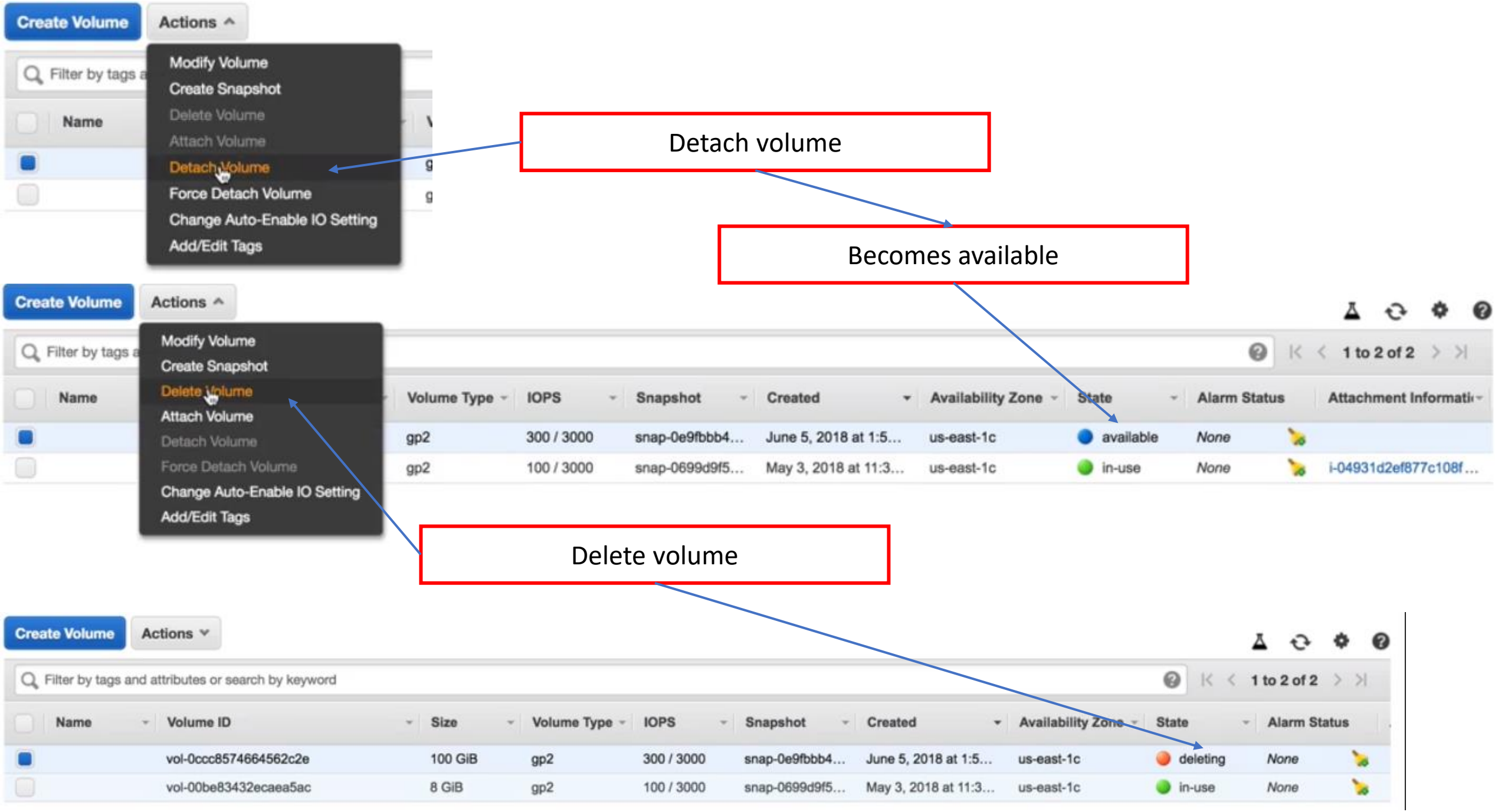
```
[root@ip-172-31-36-54 filesystem]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk
[root@ip-172-31-36-54 filesystem]# file -s /dev/xvdf
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=2276aec8-2a6e-4011-b
e2f-29f916e14f2e (extents) (large files) (huge files)
[root@ip-172-31-36-54 filesystem]# cd ..
[root@ip-172-31-36-54 /]# mount /dev/xvdf /filesystem
[root@ip-172-31-36-54 /]# cd filesystem/
[root@ip-172-31-36-54 filesystem]# ls
hello.txt  lost+found
[root@ip-172-31-36-54 filesystem]# nano hello.txt
[root@ip-172-31-36-54 filesystem]#
```

Xvdf is visible
Has the filesystem of type ext4
Content of the file are safe

To unmount

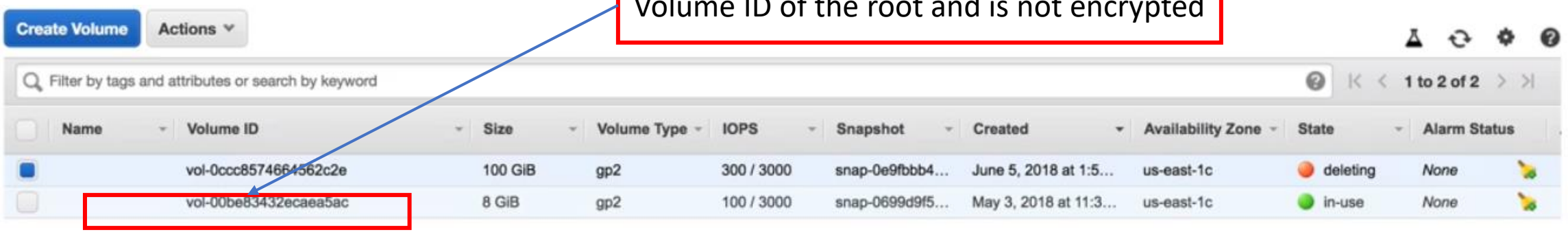
```
[root@ip-172-31-36-54 filesystem]# cd ..
[root@ip-172-31-36-54 /]# umount -d /dev/xvdf
[root@ip-172-31-36-54 /]# lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
└─xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  100G  0 disk
[root@ip-172-31-36-54 /]#
```

The overall steps we did:
Encrypted additional volume
Took snapshot of this
Deleted the volume
Restored the volume from the snapshot


Create Volume
Actions
Modify Volume
Create Snapshot
Delete Volume
Attach Volume
Detach Volume
Force Detach Volume
Change Auto-Enable IO Setting
Add/Edit Tags
Filter by tags
Name
Detach volume
Becomes available
Create Volume
Actions
Modify Volume
Create Snapshot
Delete Volume
Attach Volume
Detach Volume
Force Detach Volume
Change Auto-Enable IO Setting
Add/Edit Tags
Filter by tags
1 to 2 of 2
Name
Volume Type
IOPS
Snapshot
Created
Availability Zone
State
Alarm Status
Attachment Informati
gp2
300 / 3000
snap-0e9fbbb4...
June 5, 2018 at 1:5...
us-east-1c
available
None
gp2
100 / 3000
snap-0699d9f5...
May 3, 2018 at 11:3...
us-east-1c
in-use
None
i-04931d2ef877c108f...
Delete volume
Create Volume
Actions
Filter by tags and attributes or search by keyword
1 to 2 of 2
Name
Volume ID
Size
Volume Type
IOPS
Snapshot
Created
Availability Zone
State
Alarm Status
vol-0ccc8574664562c2e
100 GiB
gp2
300 / 3000
snap-0e9fbbb4...
June 5, 2018 at 1:5...
us-east-1c
deleting
None
vol-00be83432ecaea5ac
8 GiB
gp2
100 / 3000
snap-0699d9f5...
May 3, 2018 at 11:3...
us-east-1c
in-use
None

How to encrypt root device volume

Volume ID of the root and is not encrypted

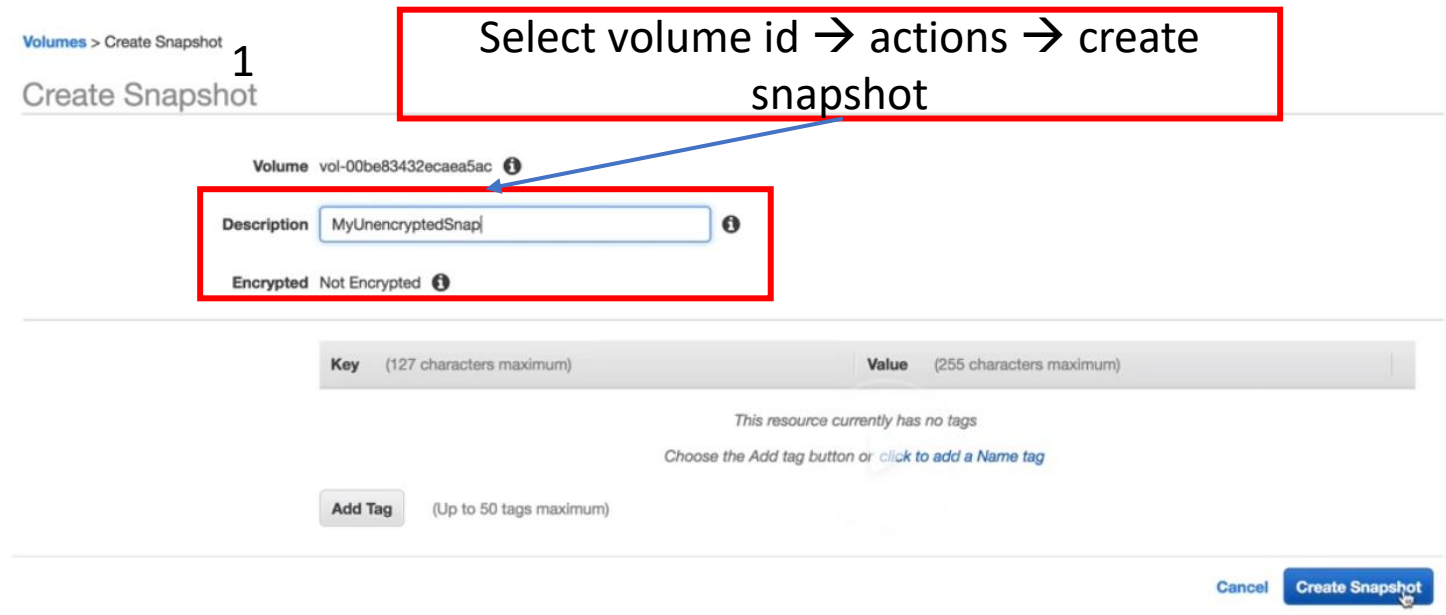


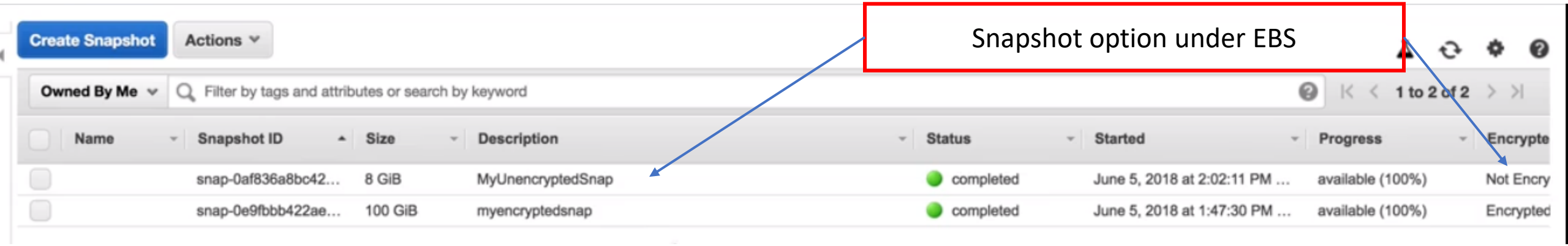
Other columns Encrypted = not encrypted.

Two ways to encrypt:

1. Use OS to encrypt the volume (windows = bitlocker)
2. Take snapshot of the volume (cleaner way)

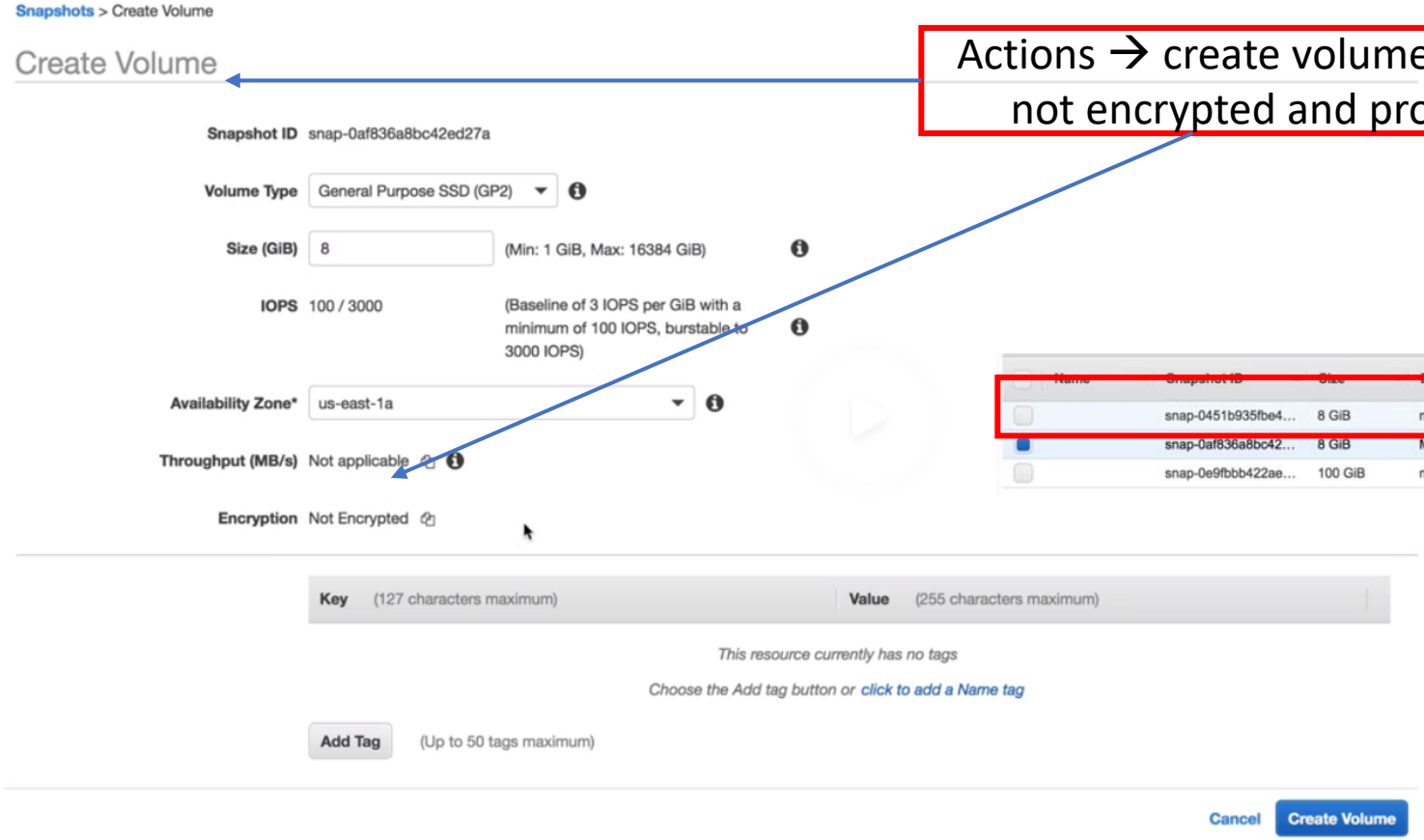
Take snapshot of the volume to encrypt root volume(cleaner way)

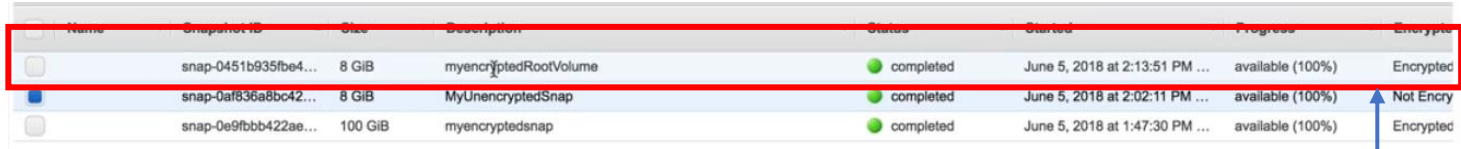




2

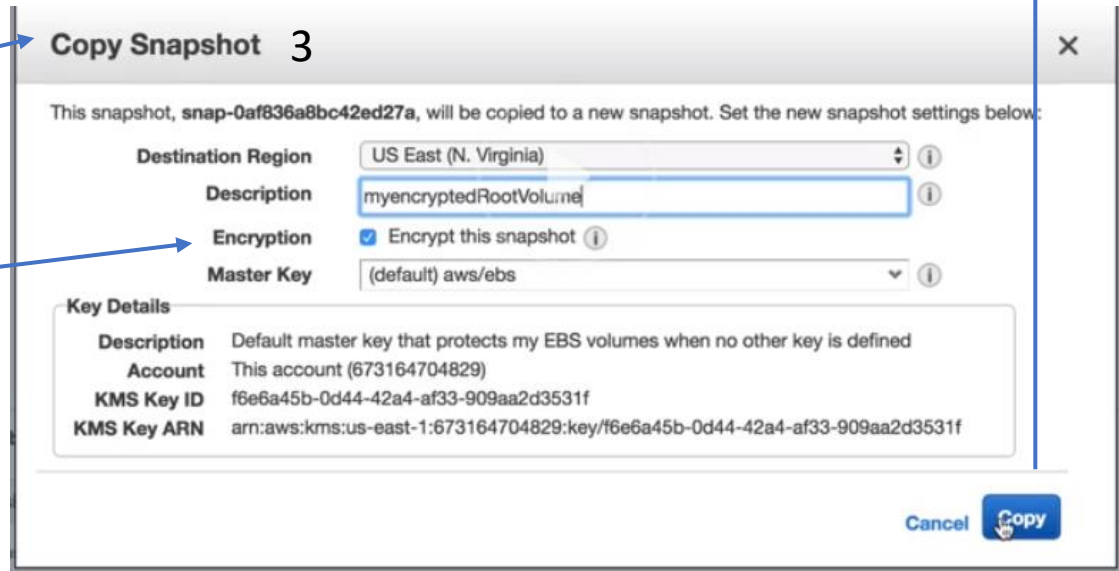
Actions → create volume (doesn't help as by default its not encrypted and provides no option to encrypt)





Instead of creating volume, copy the snapshot.
Actions → copy → gives option to encrypt the snapshot

3



4

Actions → create image (this is AMI). Use this AMI to deploy new EC2 instance)

Create Image from EBS Snapshot

Name	MyEC2EncryptedRootDeviceVolume	Description	MyEC2EncryptedRootDeviceVolume
Architecture	x86_64	Virtualization type	Paravirtual
Root device name	/dev/sda1	Kernel ID	Use default
RAM disk ID	Use default		

Block Device Mappings

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0451b935fbe40932b	8	General Purpose SSD (GP2)	100 / 3000	N/A	☑	Encrypted

Add New Volume

Cancel Create

Images → AMI

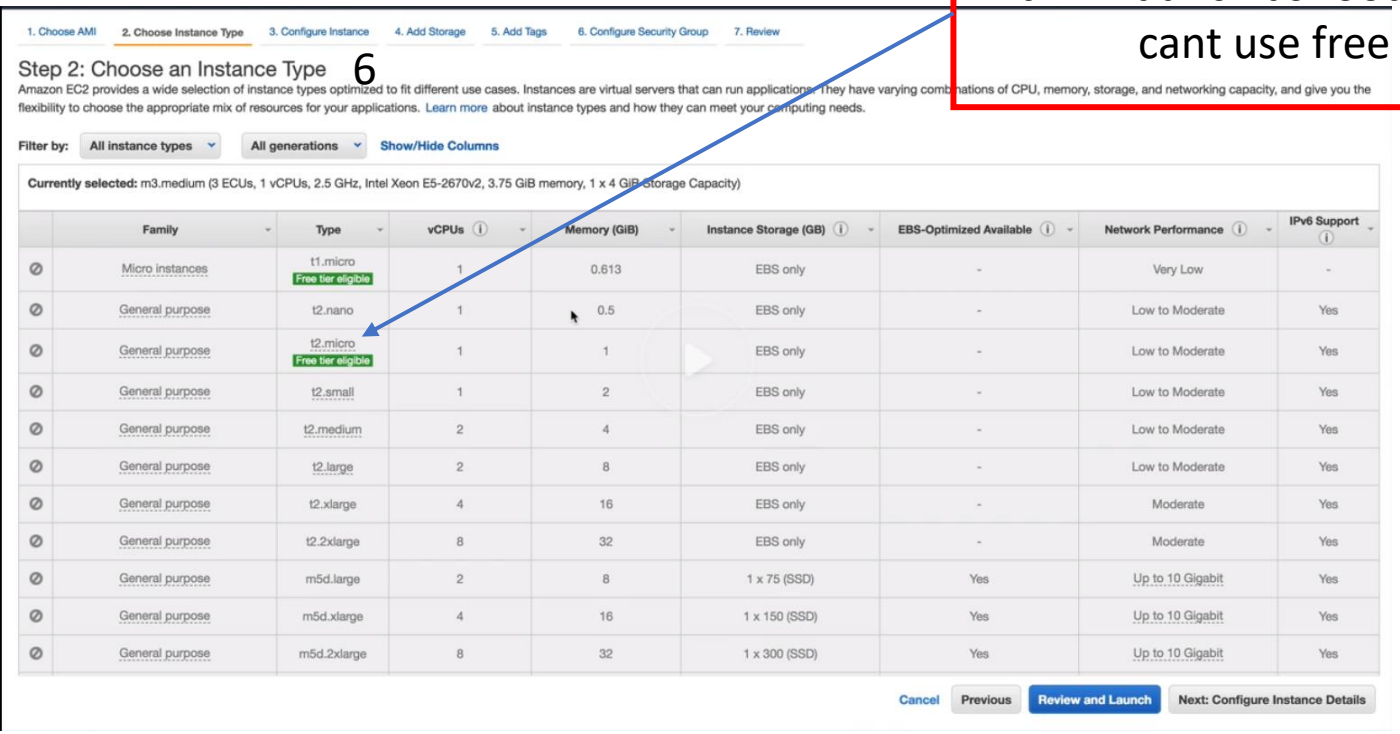
5

Launch Actions

Owned by me | Filter by tags and attributes or search by keyword | 1 to 1 of 1

Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date	Platform	Root Device T	Virt
	MyEC2Encrypt...	ami-9bd9aae4	673164704829/...	673164704829	Private	available	June 5, 2018 at 2:19:59 PM ...	Other Linux	ebs	para

It will launch as root volume. Since root volume is encrypted, it cant use free tier type instead use large instances



EC2 Encryption

EXAM TIPS

A CLOUD

- You can encrypt the root device volume (the volume the OS is installed on) using Operating System level encryption
- You can encrypt the root device volume by first taking a snapshot of that volume, and then creating a copy of that snap with encryption. You can then make an AMI of this snap and deploy the encrypted root device volume
- You can encrypt additional attached volumes using the console, CLI or API