Amazon EBS Volumes

An Amazon EBS volume is a durable, block-level storage device that you can attach to a single EC2 instance.

It is a disk where operating system is installed.

It is not a part of virtual machine, It is a separate external disk.

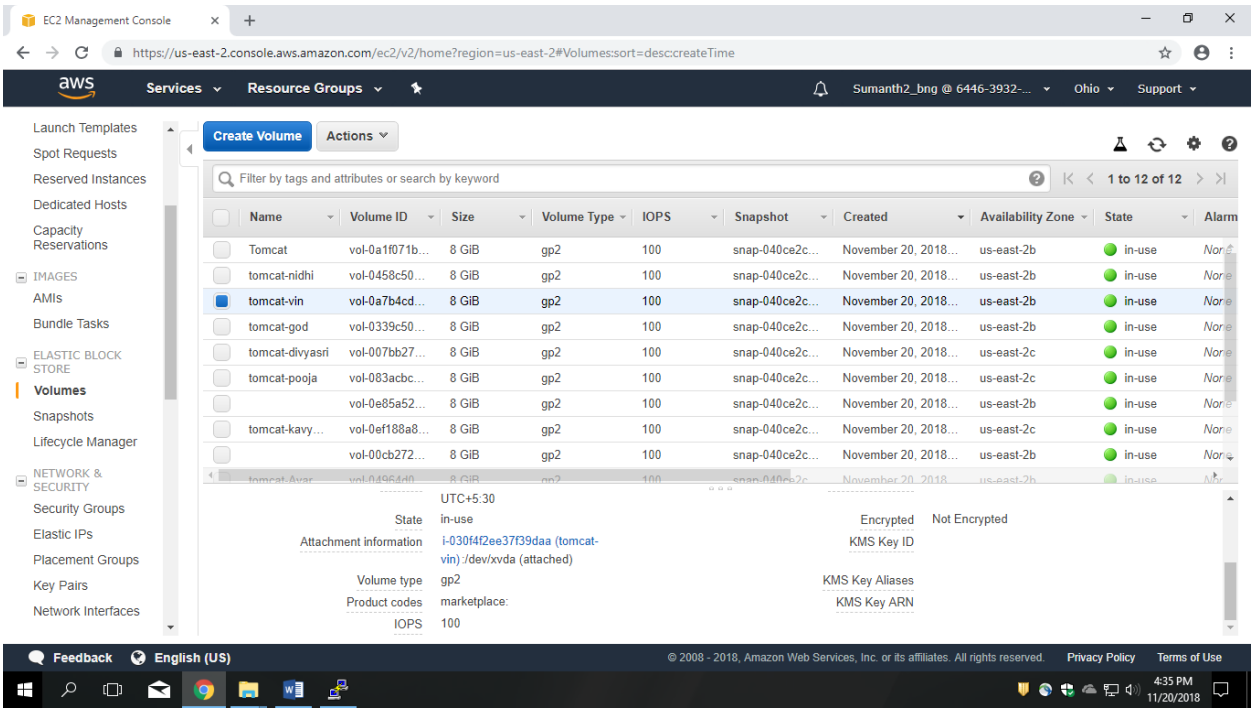
AWS Storage Service contains:-

a) EBS (Elastic Block Store)
b) S3 (Simple Storage Service)

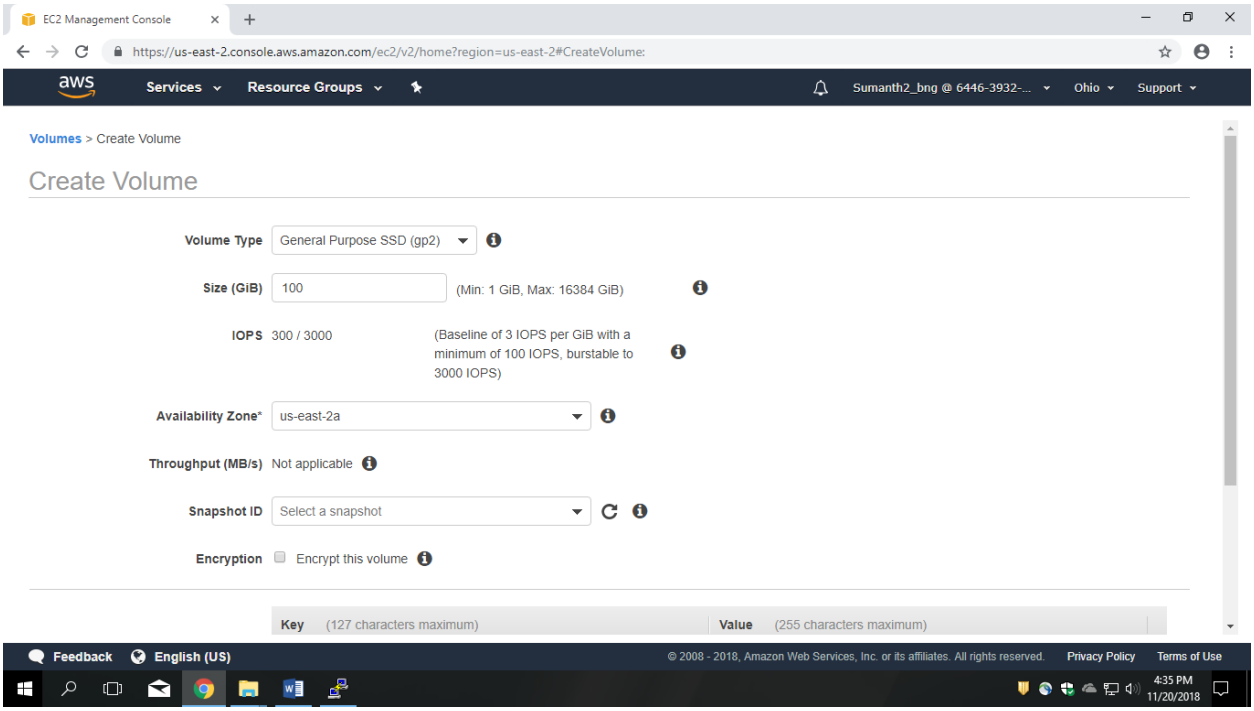
EBS is a hard disk for EC2, It is block level storage.

It is highly durable and highly available, your data is replicated on a different physical component, if one component fails it auto failovers to other component.

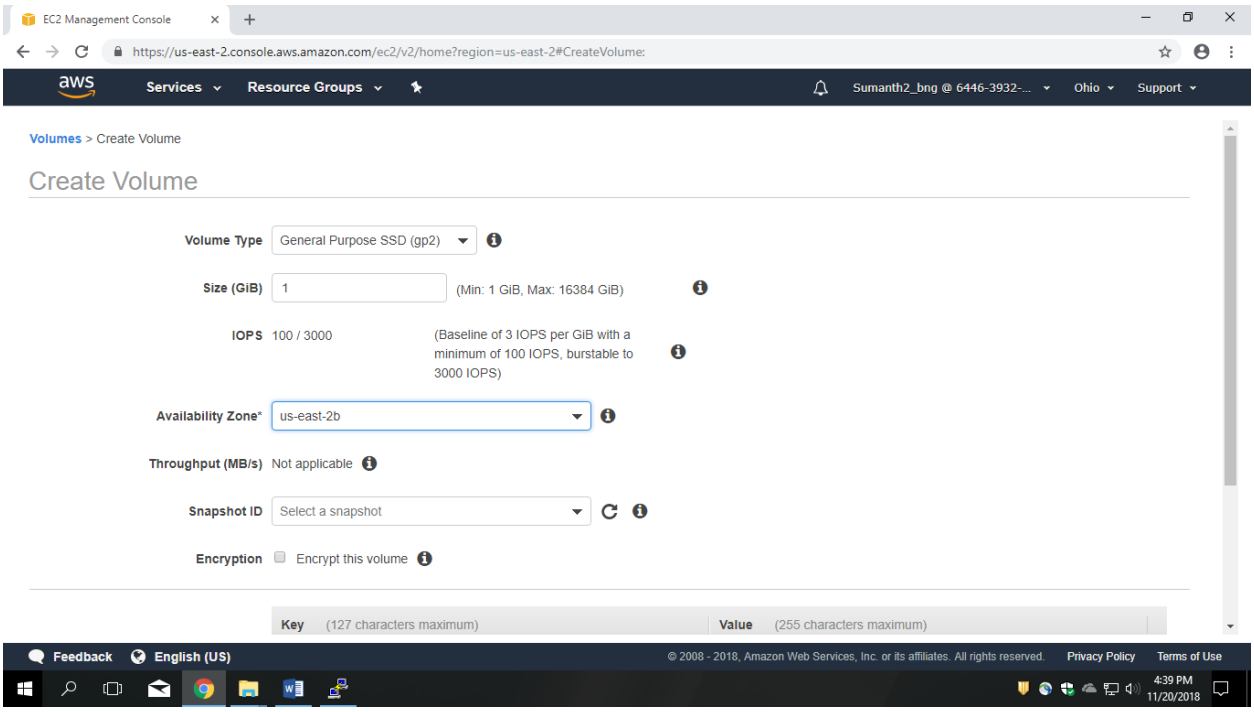
You can detach volume from one EC2 and attach to another EC2. You can change **size of EBS Volume (you can only increase and cannot decrease).**



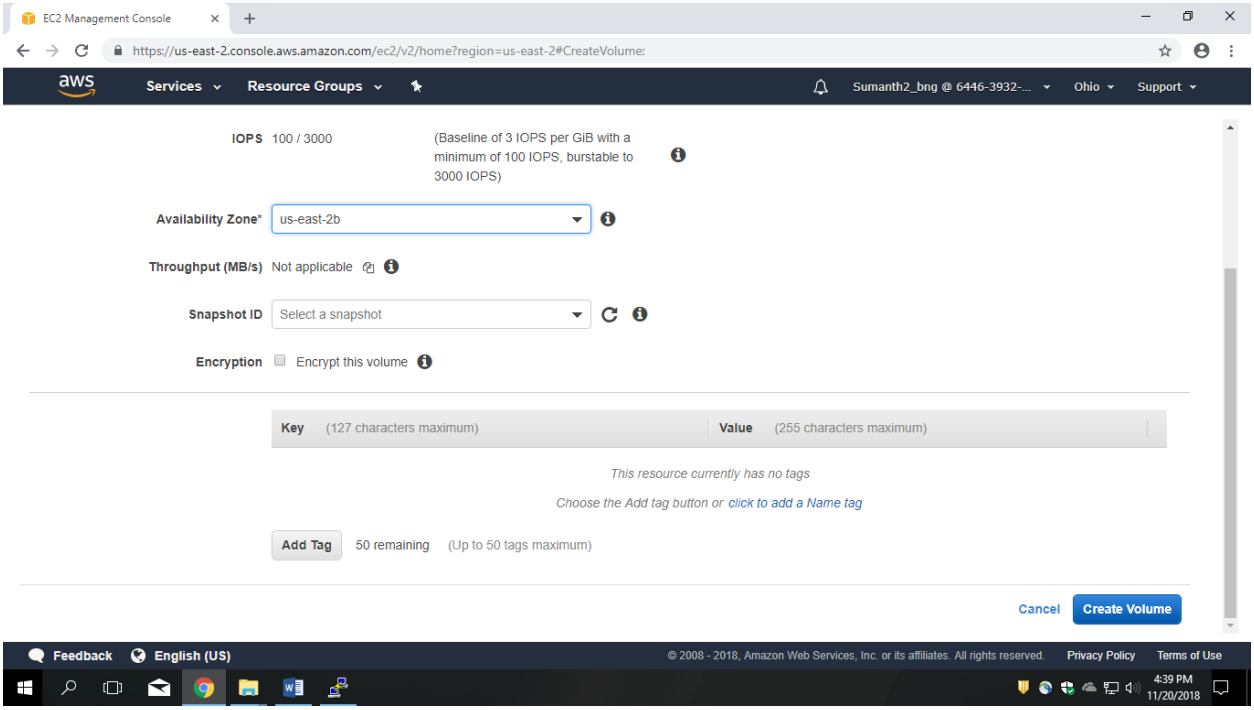
Step 1:After creating bucket in S3, Click on Create Volume button.
(Refer S3 file for creating a bucket)



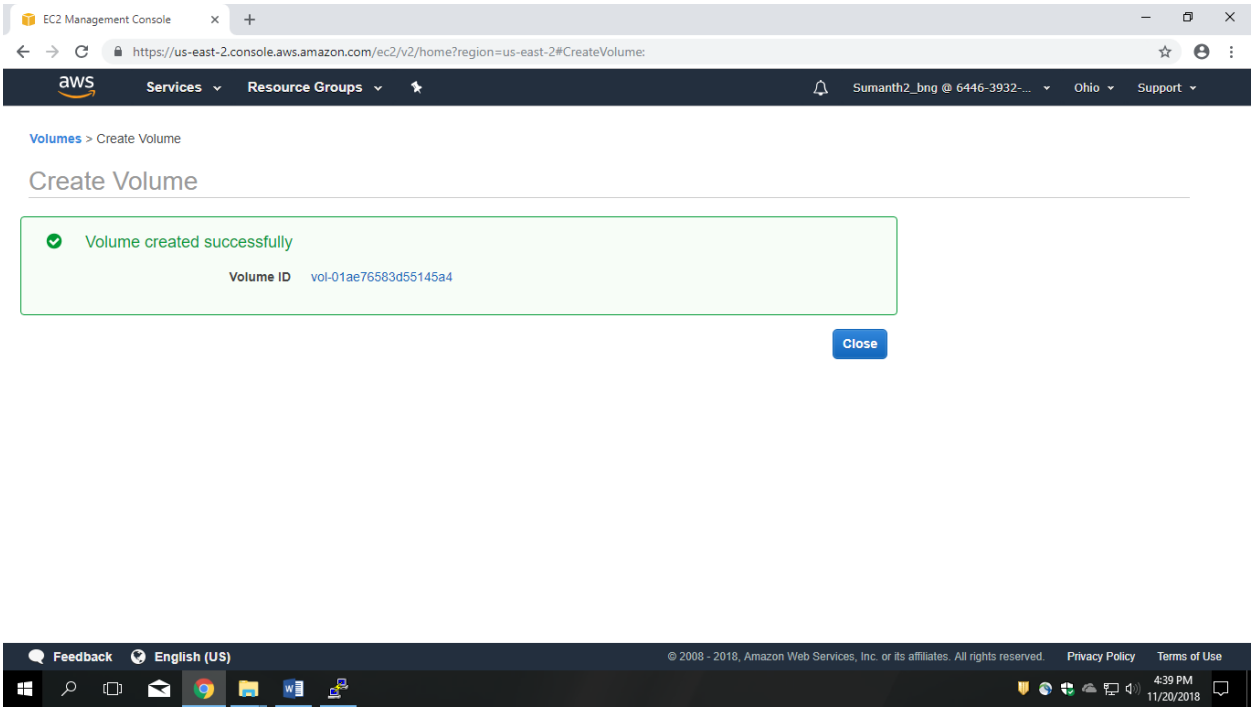
Step 2: Change the size of volume from 100 Gb to 1 Gb (depends upon user requirements).



Select Availability Zone (make sure that the availability zone of your EC2 instance and Volume is same)

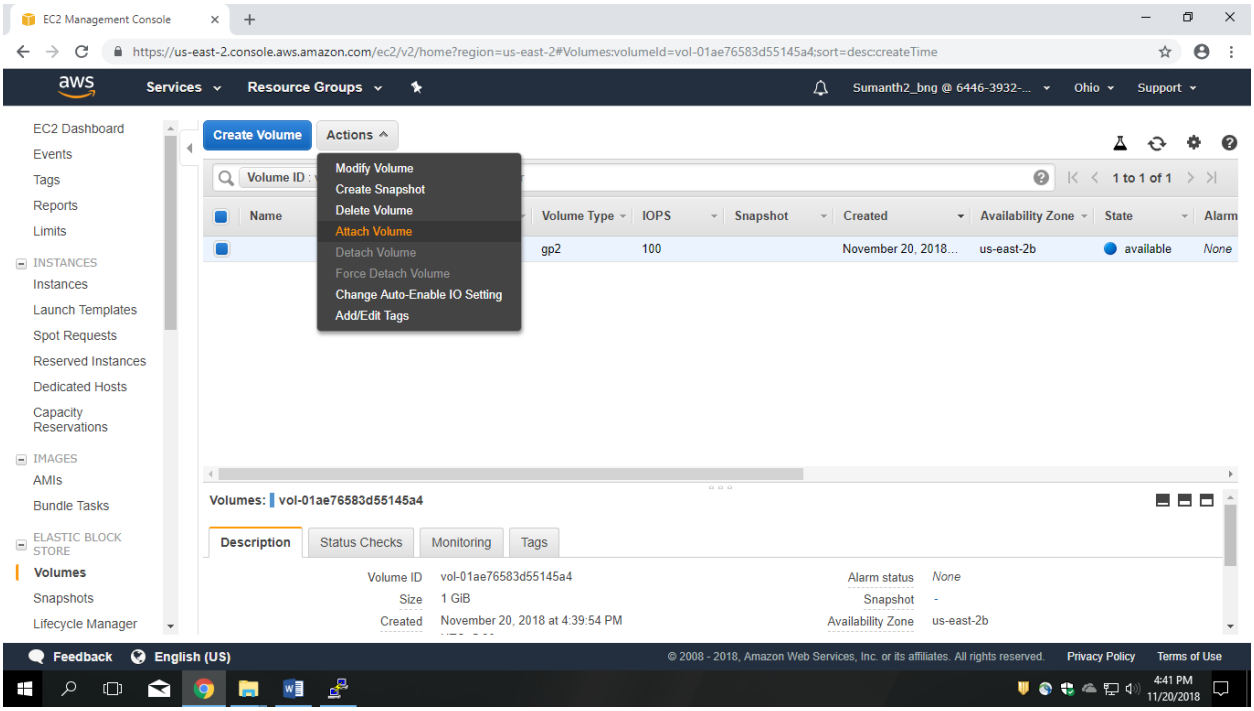


Step 3: Click on create Volume to create new AWS EBS Volume.

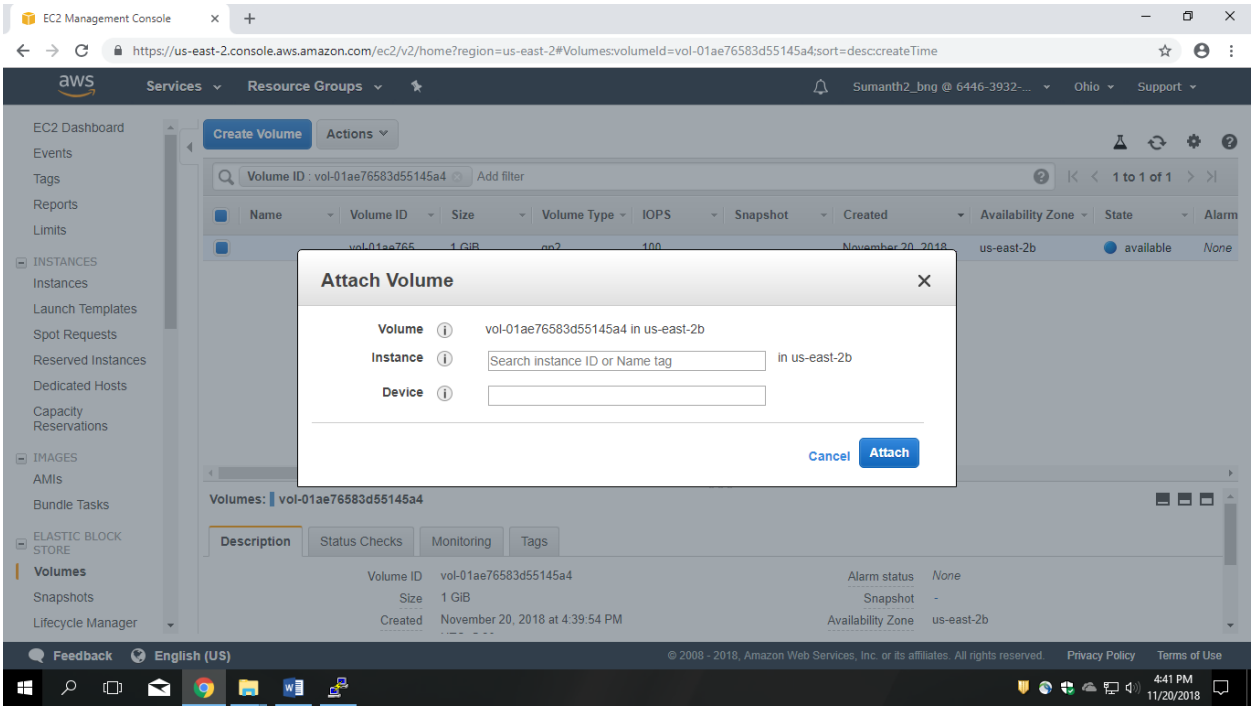


Note down Volume Id.

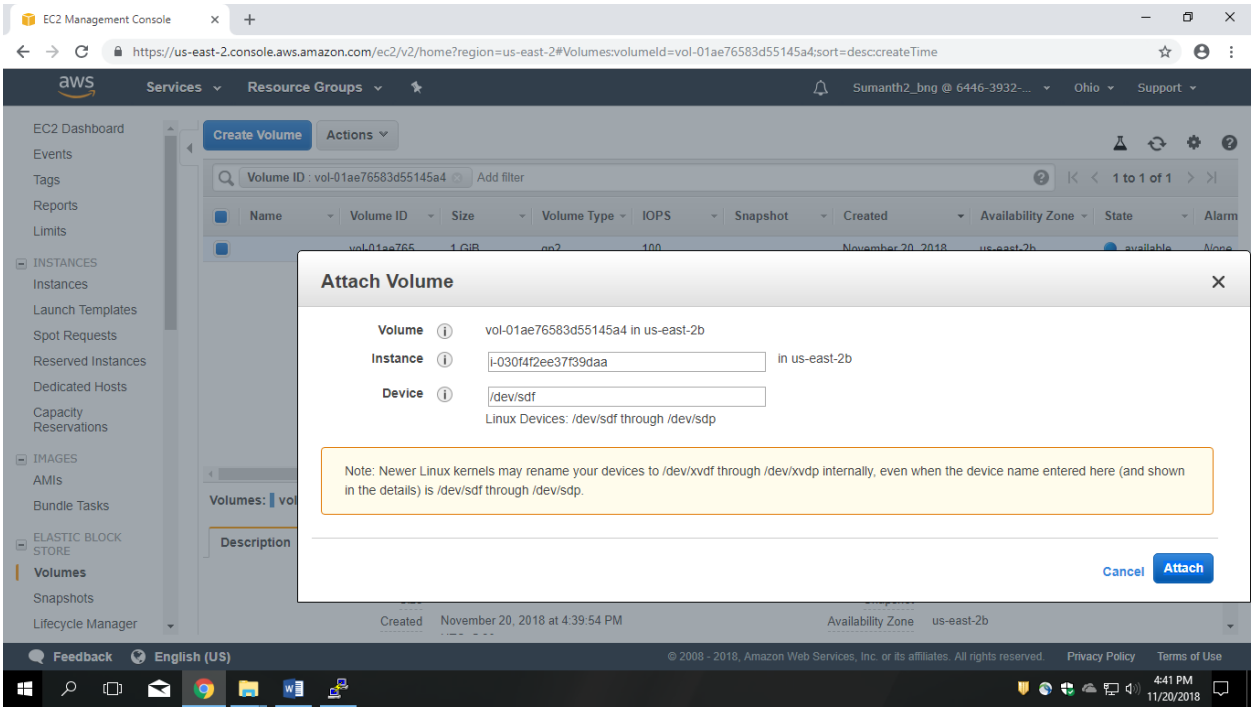
Step 4: Click on Volume Id.



Step 5: Click on Actions, Select Attach Volume

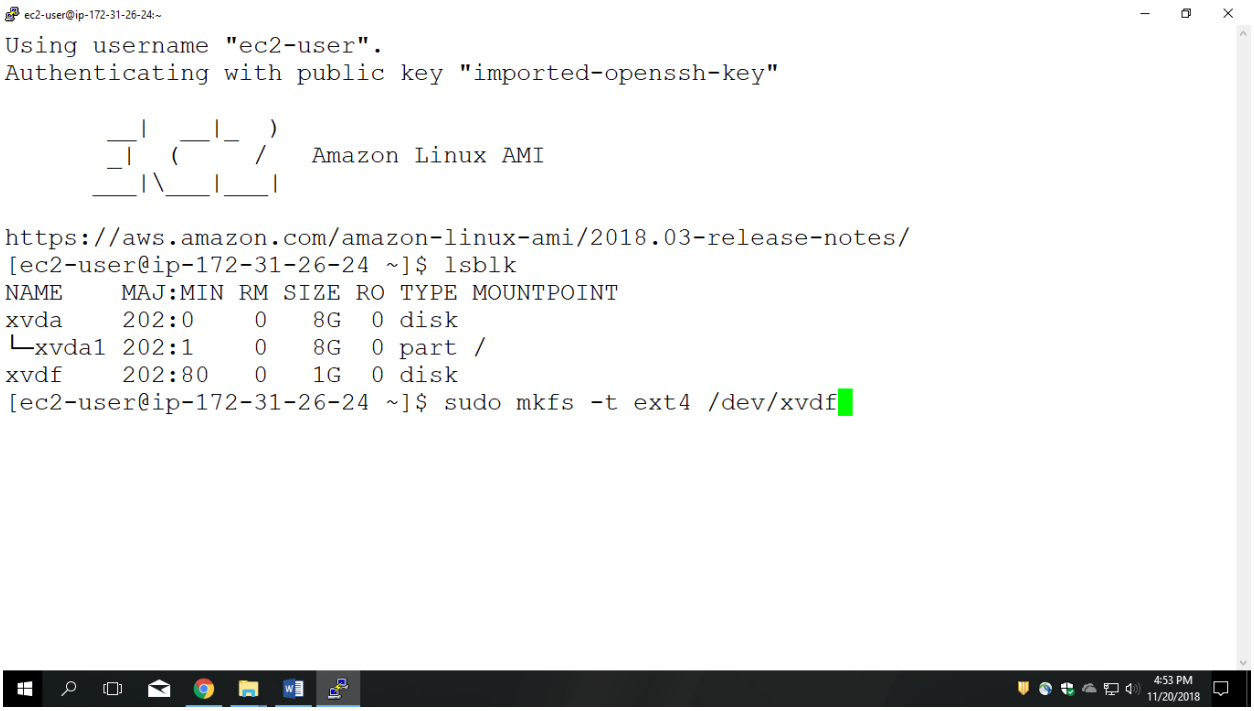


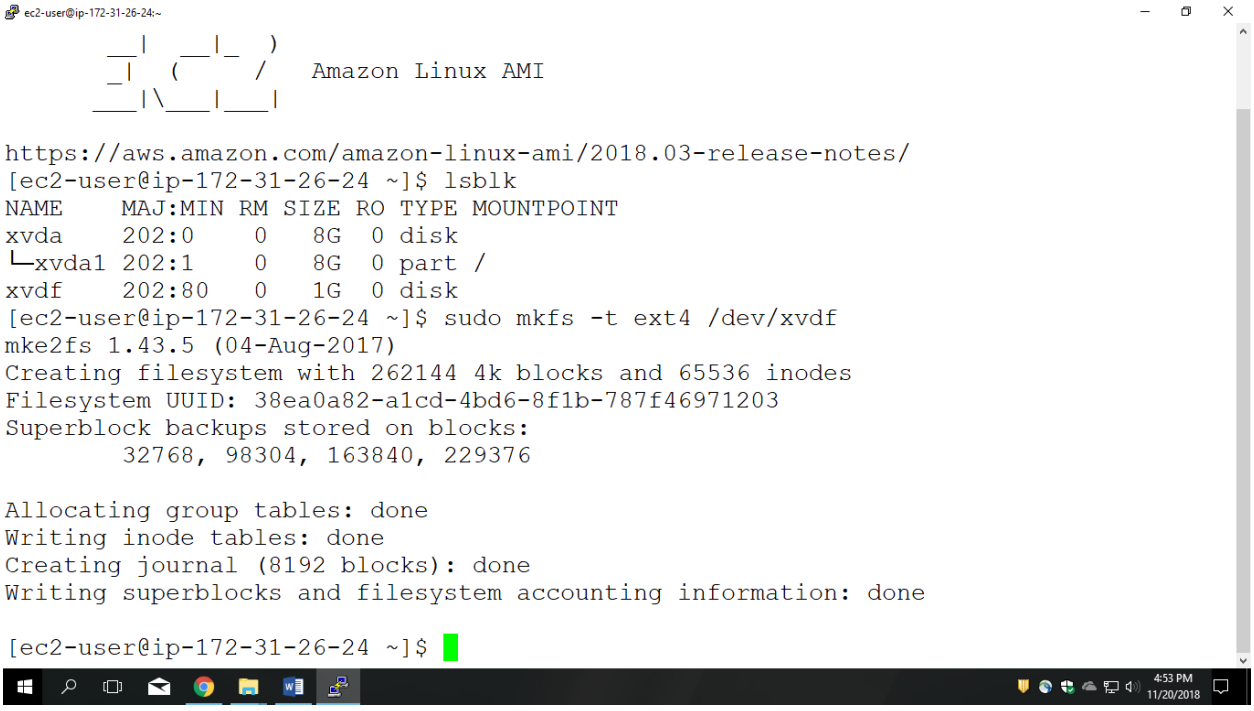
Step 6: Enter your EC2 instance Id and device name.

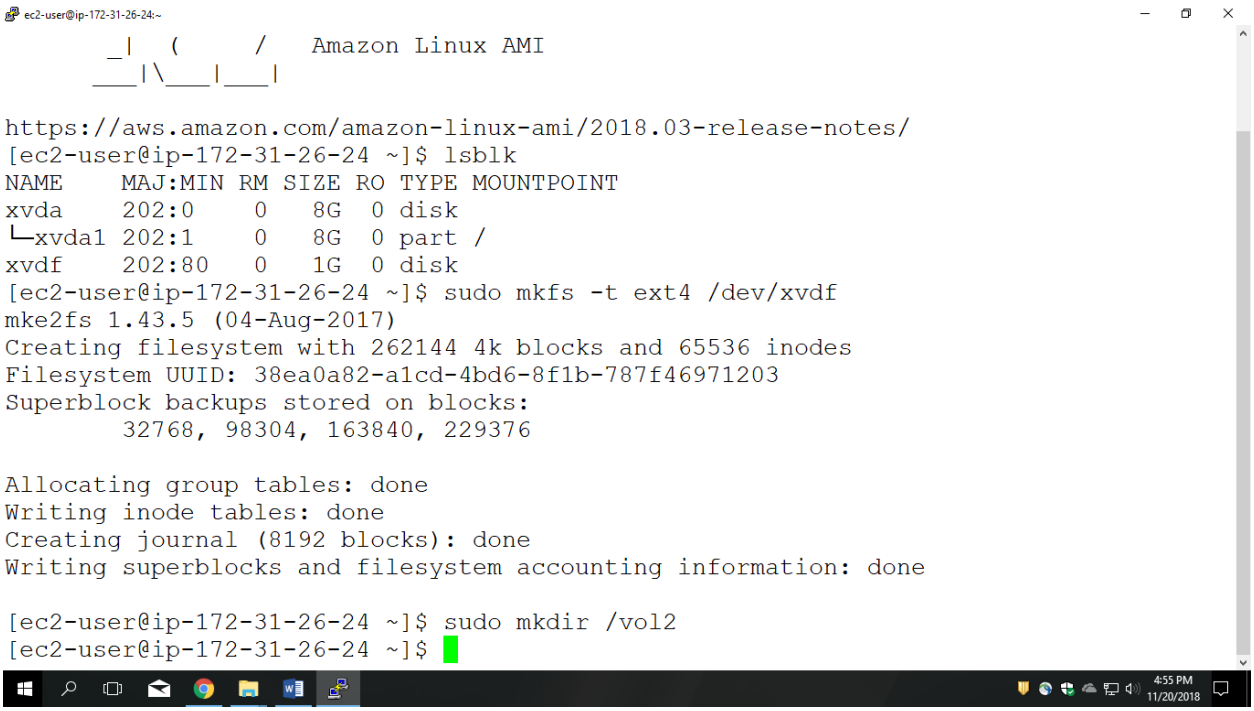


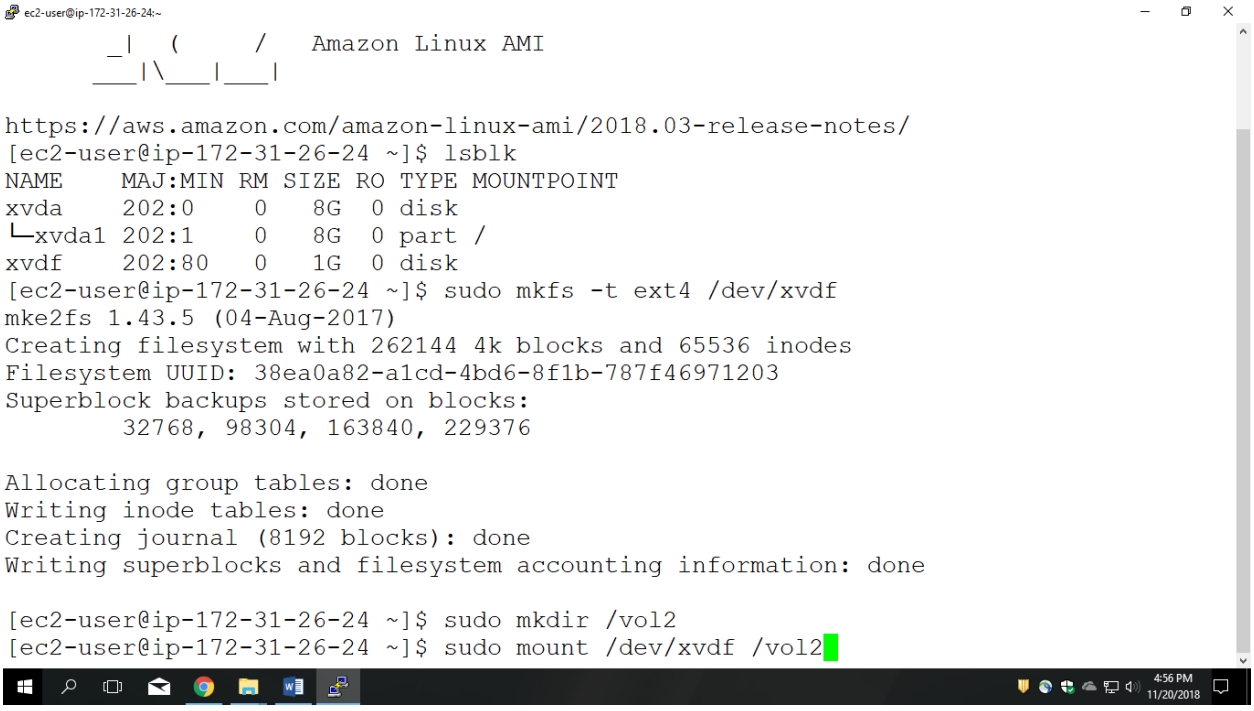
Step 7: Click on Attach.

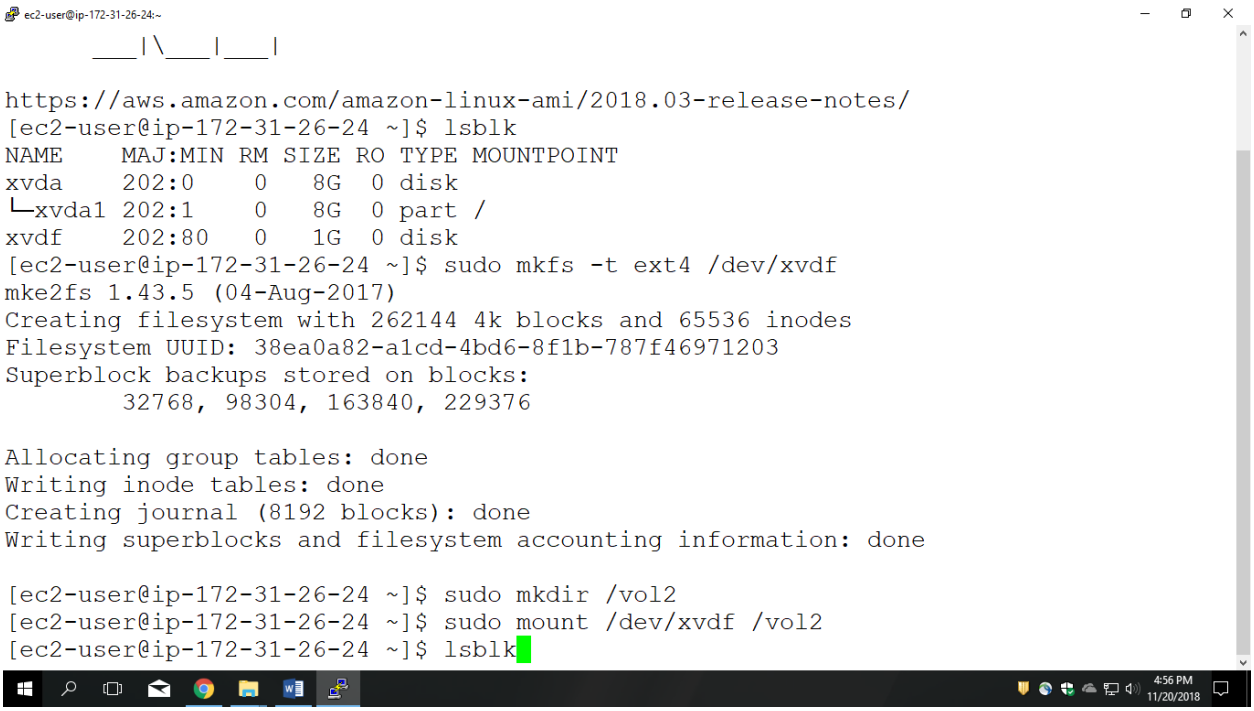
After attaching, we have to mount the volume. Before mounting, open putty.exe and enter the following commands.

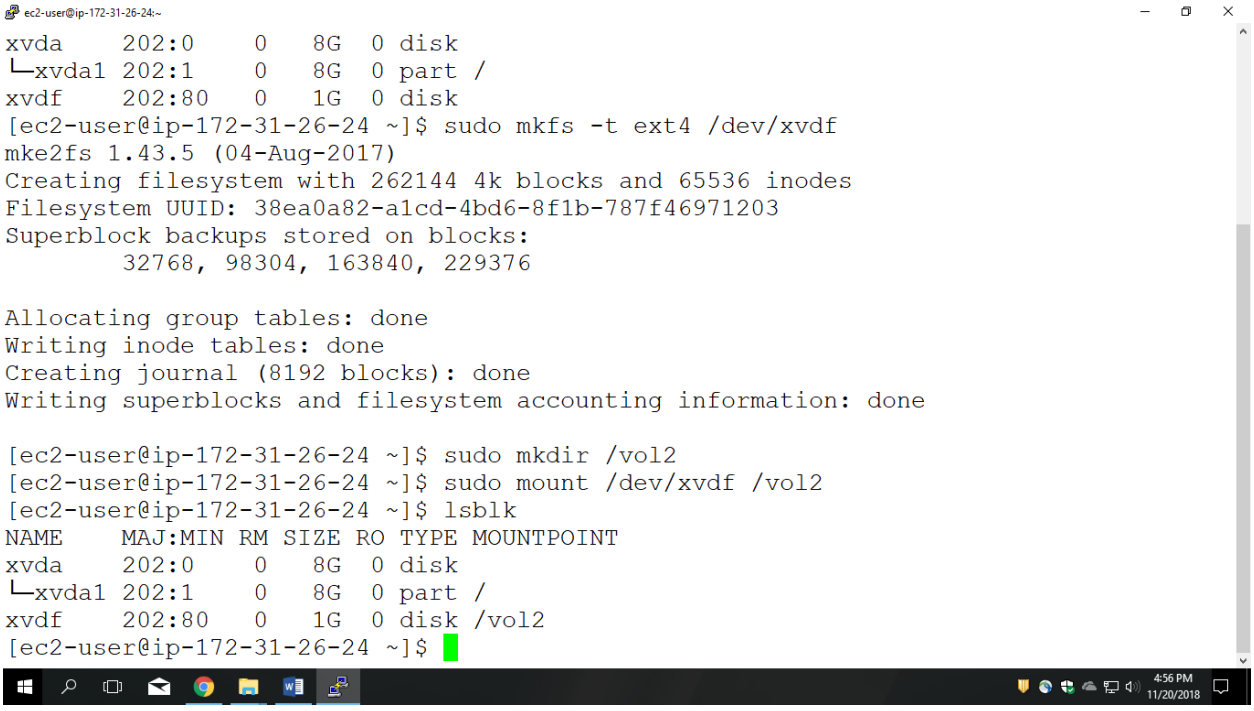

ec2-user@ip-172-31-26-24:~
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
__| __|_)
_| (/   Amazon Linux AMI
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
[ec2-user@ip-172-31-26-24 ~]$ lsblk
NAME   MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda   202:0   0   8G 0 disk
└─xvda1 202:1   0   8G 0 part /
xvdf   202:80   0   1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
4:53 PM
11/20/2018

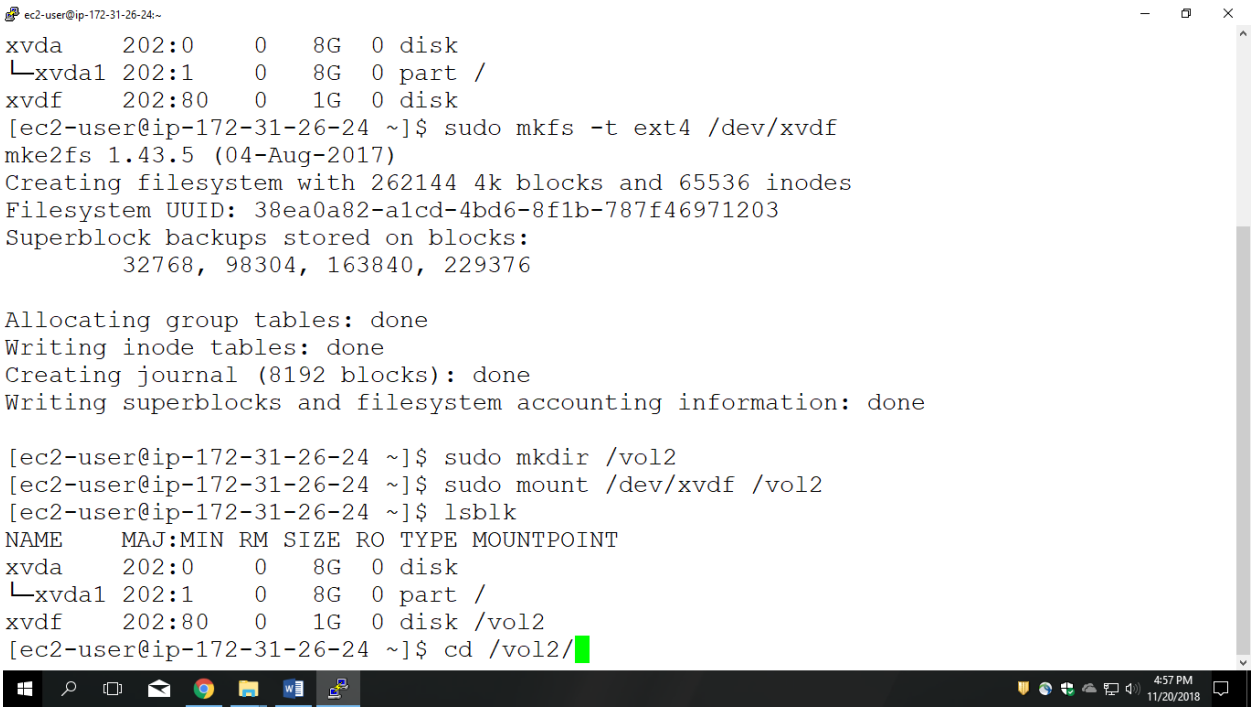

ec2-user@ip-172-31-26-24:~
__| __|_)
_| (/   Amazon Linux AMI
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
[ec2-user@ip-172-31-26-24 ~]$ lsblk
NAME   MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda   202:0   0   8G 0 disk
└─xvda1 202:1   0   8G 0 part /
xvdf   202:80   0   1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.43.5 (04-Aug-2017)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 38ea0a82-a1cd-4bd6-8f1b-787f46971203
Superblock backups stored on blocks:
32768, 98304, 163840, 229376
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-26-24 ~]$
4:53 PM
11/20/2018

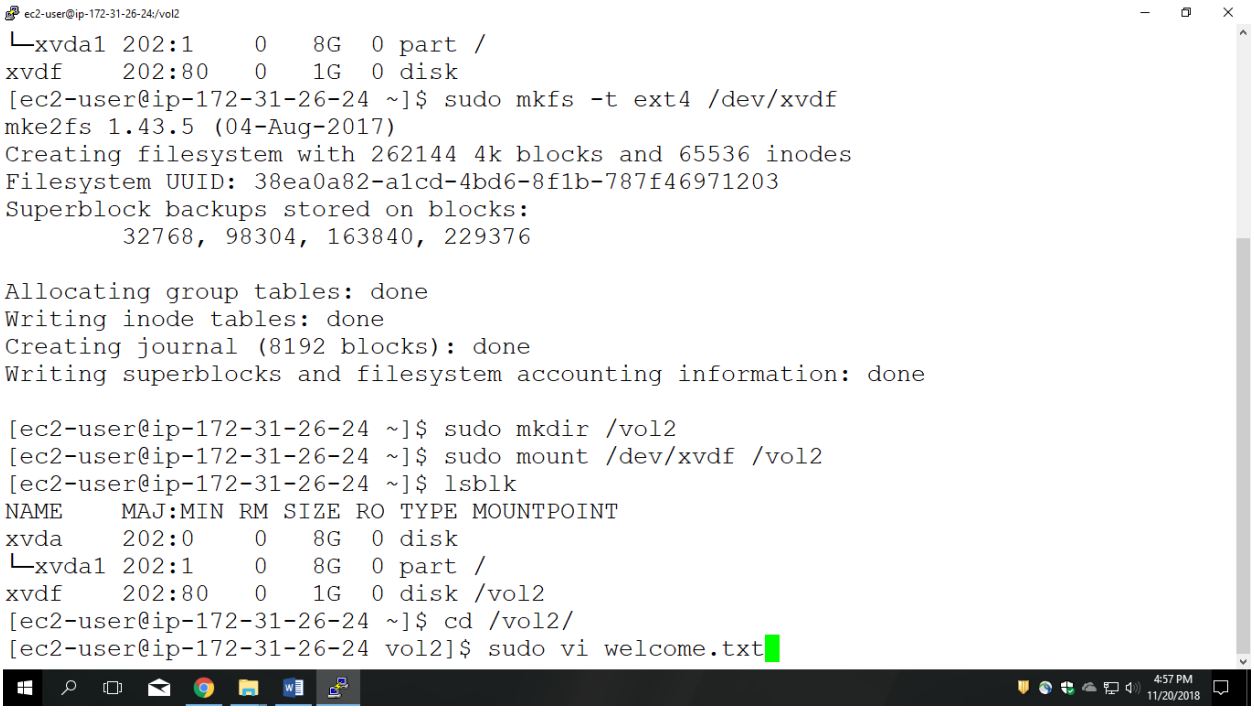

ec2-user@ip-172-31-26-24:~
_| (/ Amazon Linux AMI
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
[ec2-user@ip-172-31-26-24 ~]$ lsblk
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda 202:0 0 8G 0 disk
└─xvda1 202:1 0 8G 0 part /
xvdf 202:80 0 1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.43.5 (04-Aug-2017)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 38ea0a82-a1cd-4bd6-8f1b-787f46971203
Superblock backups stored on blocks:
32768, 98304, 163840, 229376
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-26-24 ~]$ sudo mkdir /vol2
[ec2-user@ip-172-31-26-24 ~]$
4:55 PM
11/20/2018


ec2-user@ip-172-31-26-24:~
_| (/ Amazon Linux AMI
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
[ec2-user@ip-172-31-26-24 ~]$ 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 1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.43.5 (04-Aug-2017)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 38ea0a82-a1cd-4bd6-8f1b-787f46971203
Superblock backups stored on blocks:
32768, 98304, 163840, 229376
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-26-24 ~]$ sudo mkdir /vol2
[ec2-user@ip-172-31-26-24 ~]$ sudo mount /dev/xvdf /vol2
4:56 PM
11/20/2018


ec2-user@ip-172-31-26-24:~
___|___|___|
https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
[ec2-user@ip-172-31-26-24 ~]$ 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   1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.43.5 (04-Aug-2017)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 38ea0a82-a1cd-4bd6-8f1b-787f46971203
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-26-24 ~]$ sudo mkdir /vol2
[ec2-user@ip-172-31-26-24 ~]$ sudo mount /dev/xvdf /vol2
[ec2-user@ip-172-31-26-24 ~]$ lsblk
4:56 PM
11/20/2018

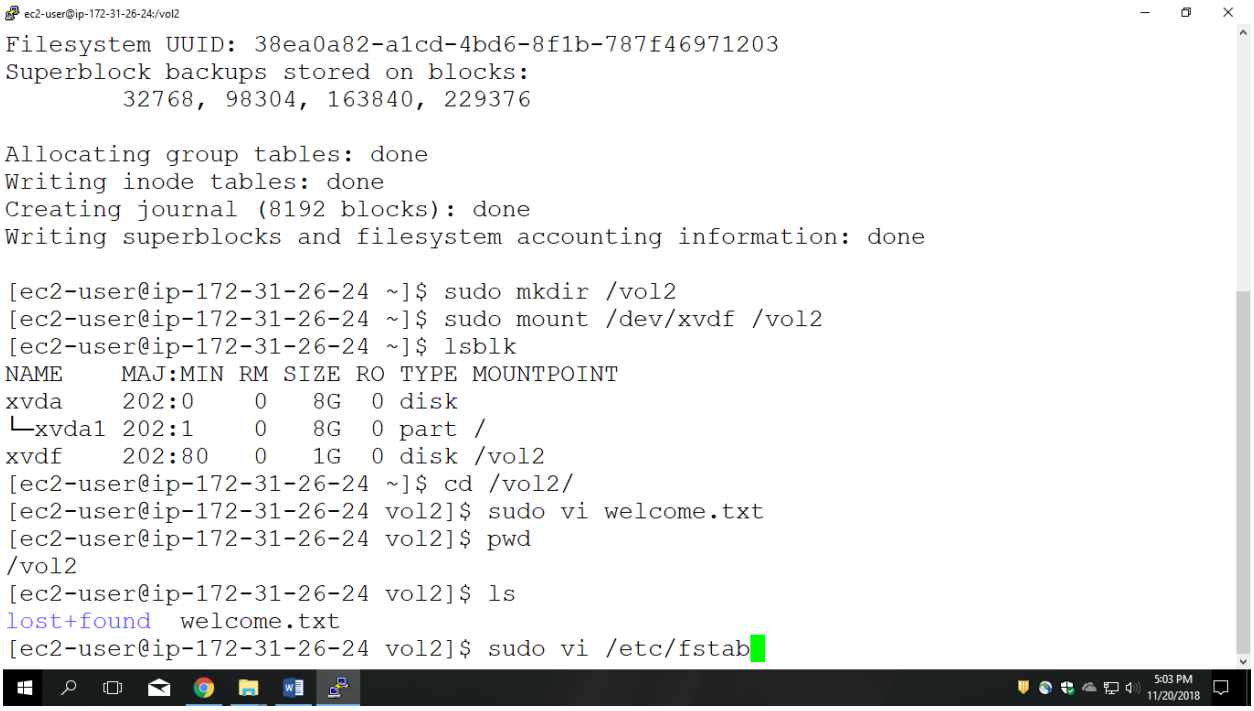

ec2-user@ip-172-31-26-24:~
xvda   202:0   0   8G 0 disk
└─xvda1 202:1   0   8G 0 part /
xvdf   202:80   0   1G 0 disk
[ec2-user@ip-172-31-26-24 ~]$ sudo mkfs -t ext4 /dev/xvdf
mke2fs 1.43.5 (04-Aug-2017)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 38ea0a82-a1cd-4bd6-8f1b-787f46971203
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
[ec2-user@ip-172-31-26-24 ~]$ sudo mkdir /vol2
[ec2-user@ip-172-31-26-24 ~]$ sudo mount /dev/xvdf /vol2
[ec2-user@ip-172-31-26-24 ~]$ 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   1G 0 disk /vol2
[ec2-user@ip-172-31-26-24 ~]$
4:56 PM
11/20/2018





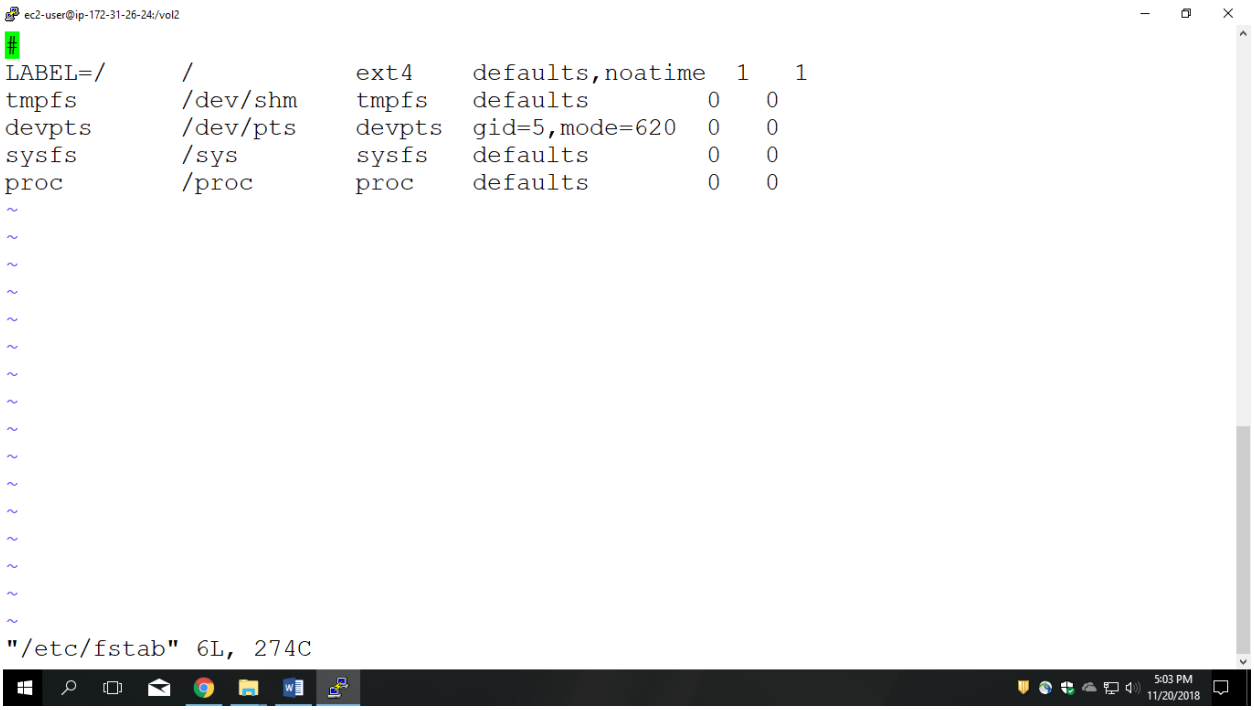
Step 8: Create Welcome.txt file and edit this file using vi editor.

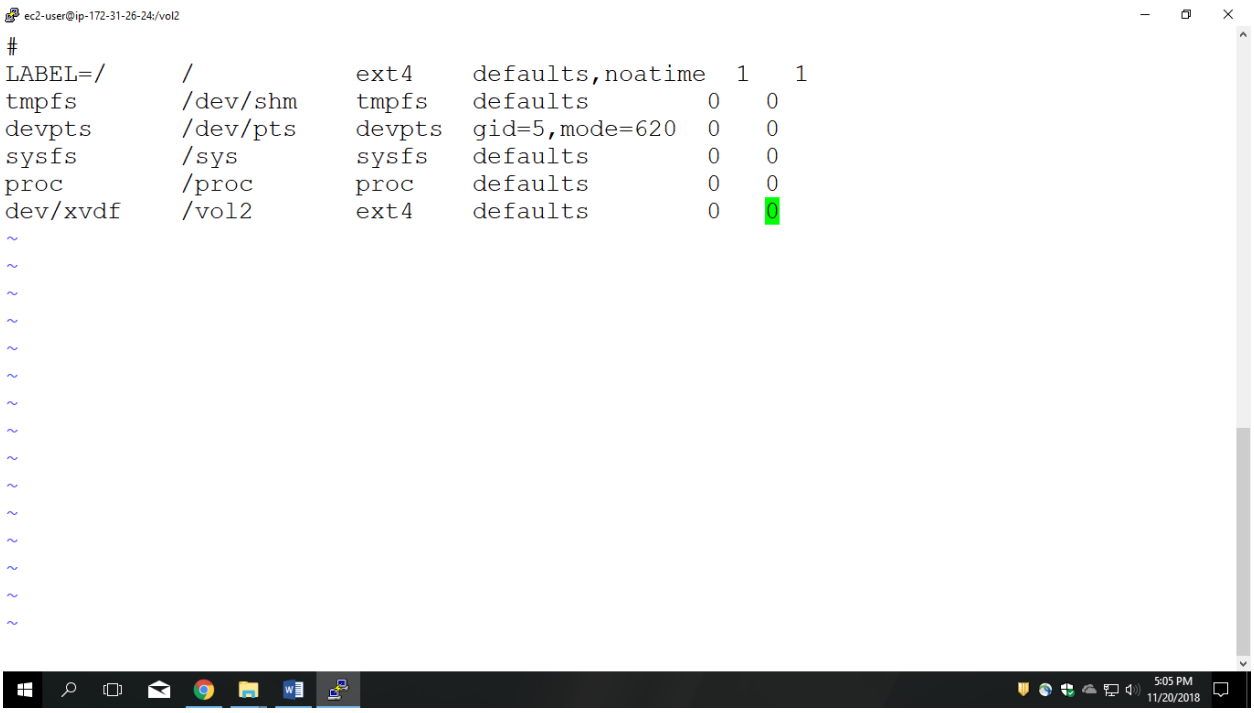
Step 9: To Save the file and exit from editor, press Escape key and then type :wq



Step 10: To auto-mount volume, follow the following commands.

sudo vi /etc/fstab





Step 11: Insert

dev/xvdf /vol2 ext4 defaults 0 0

Save the file and exit from vi editor.

AWS EBS Volume is created.