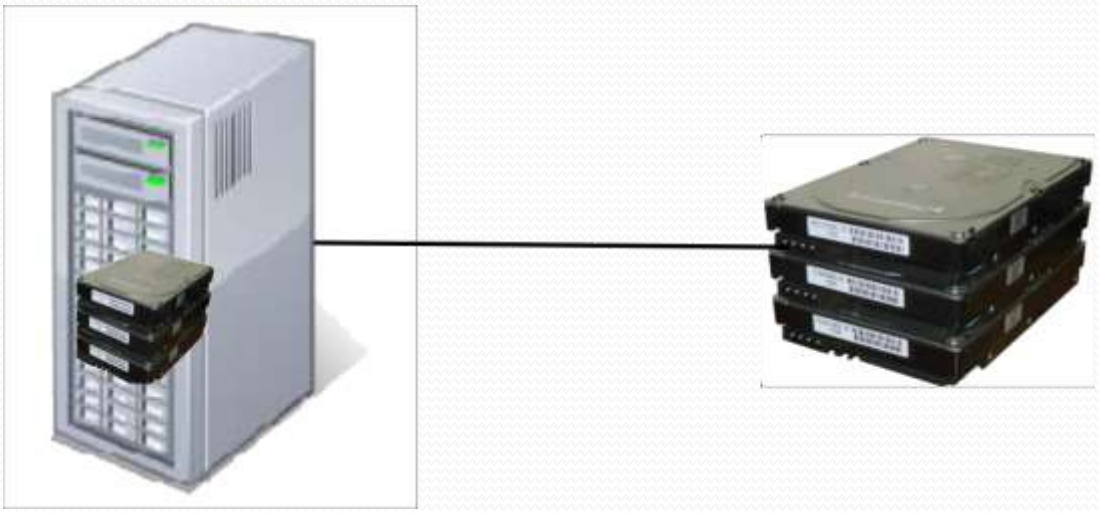
What is Elastic Block Store (EBS)?



Instance

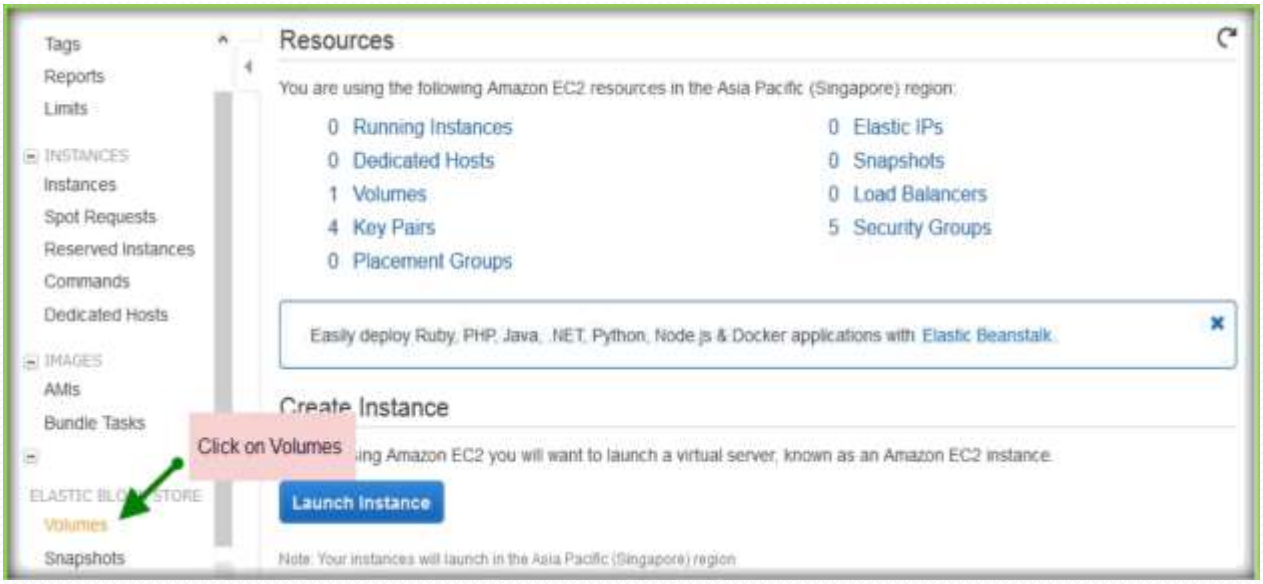
EBS storage

- **EBS storage is allocated in volumes**
  - A volume is a 'virtual disk' (size: 1gb – 16tb)
  - Basically, a raw block device
  - Can be attached to an instance (but only one at a time)
  - A single instance can access multiple volumes
- **Placed in specific availability zones**
  - Why is this useful?
  - Be sure to place it near instances (otherwise can't attach)
- **Replicated across multiple servers**
  - Data is not lost if a single server fails
  - Amazon: Annual failure rate is 0.1-0.2%

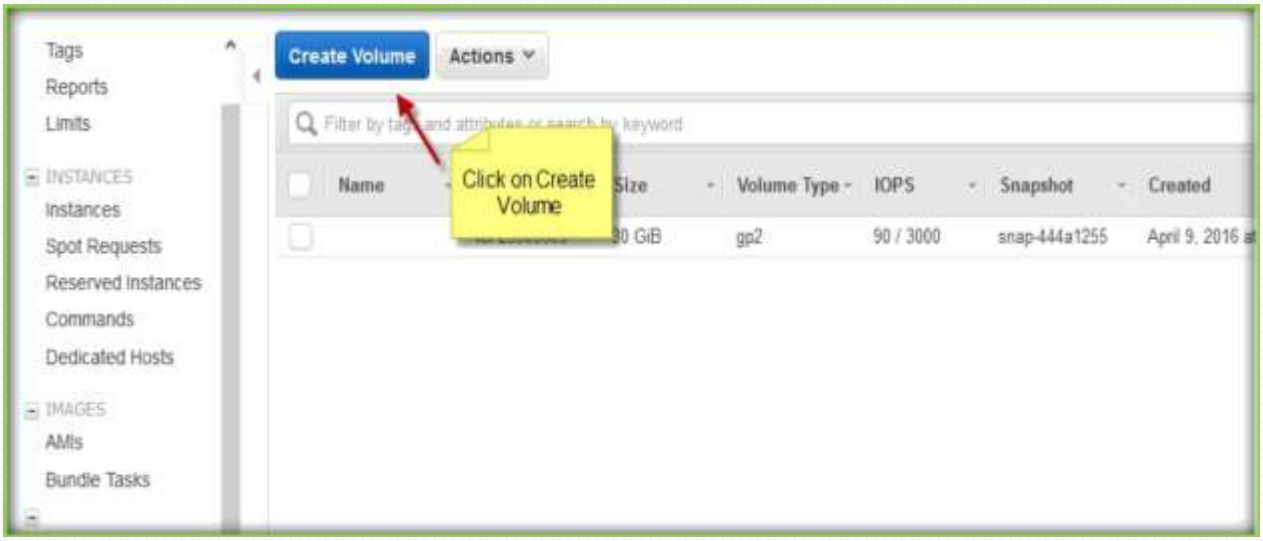
- **EC2 instances can have an EBS volume as their root device ("EBS boot")**
    - Result: Instance data persists independently from the lifetime of the instance
    - You can stop and restart the instance, like suspending and resuming a laptop
        - You won't be charged for the instance while it is stopped (only for EBS)
    - You can enable termination protection for the instance
        - Blocks attempts to terminate the instance (e.g., by accident) until termination protection is disabled again

CREATE EBS VOLUMES

Once you are in the EC2 page, click Volumes under ELASTIC BLOCK STORE on the left pane.

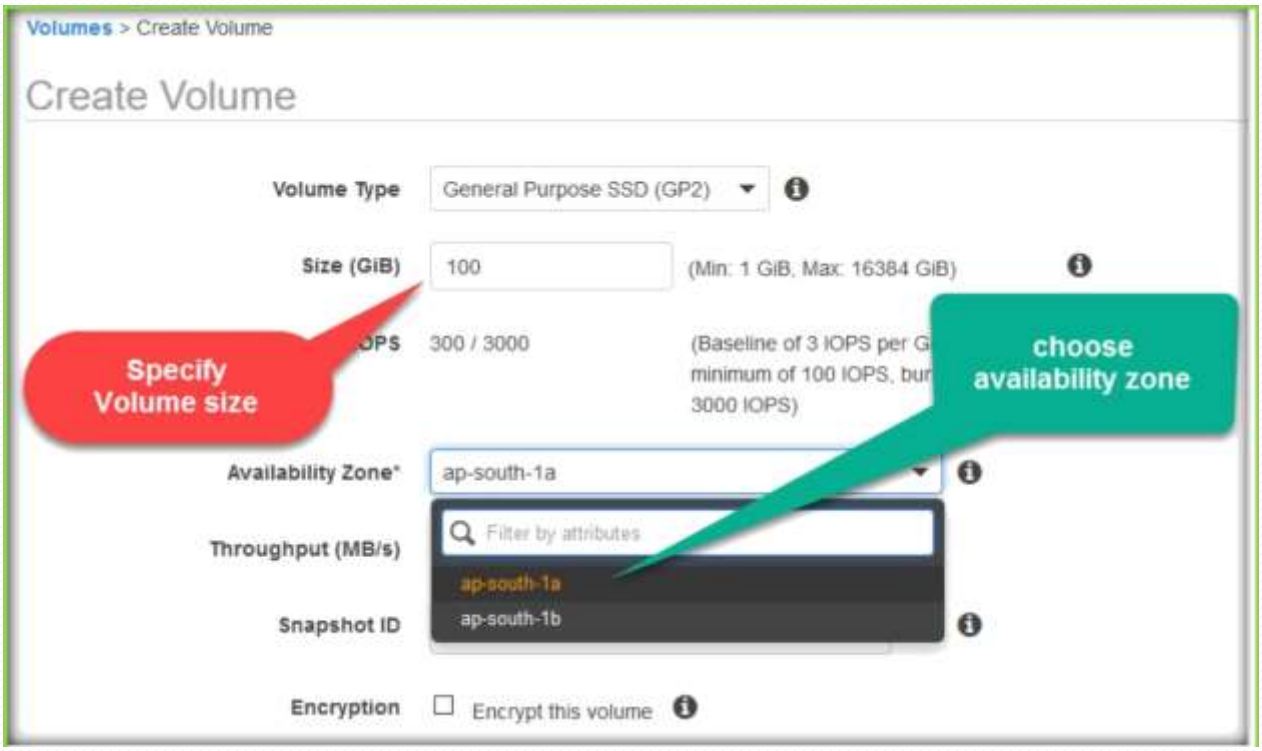


Once you are in the Volumes page, click on Create Volume to create a new volume.



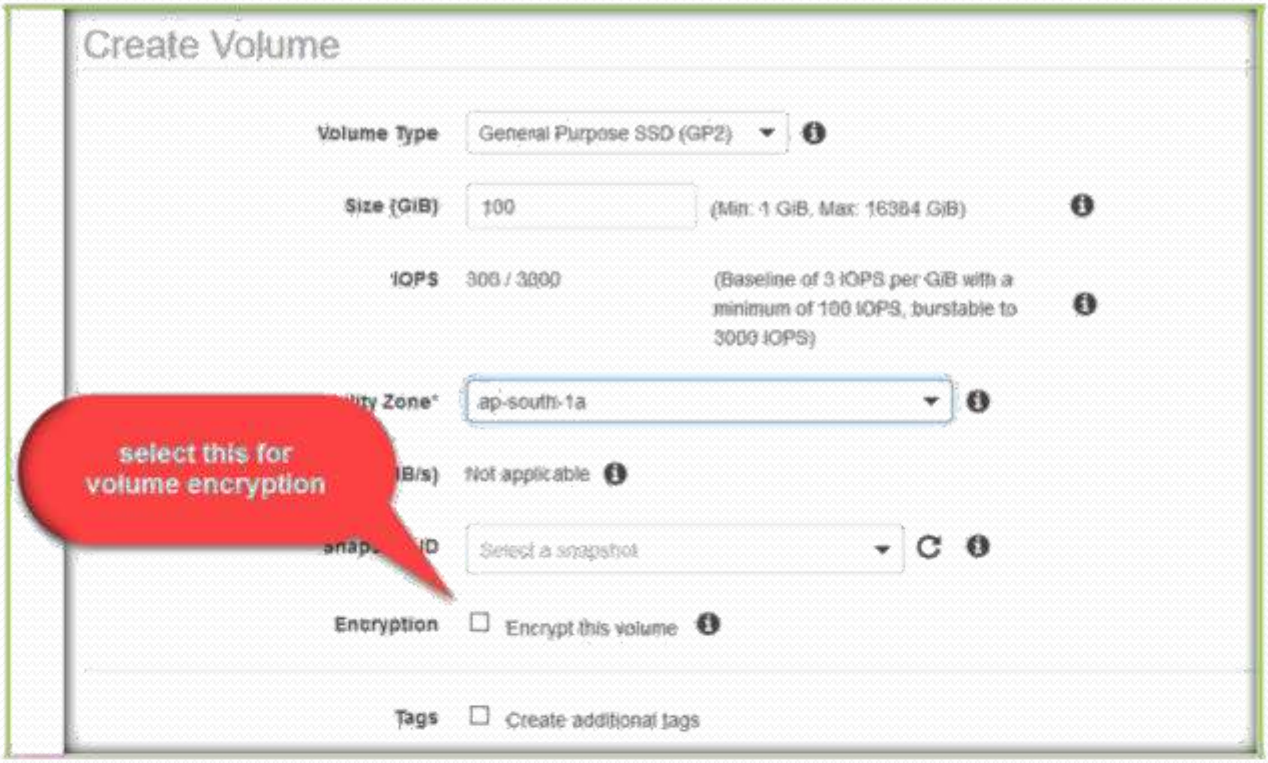
A pop-up window will come and there you need to specify the volume specifications.
Select Volume Type from the drop-down list.

Specify volume size in GBs in Size text field and select availability zone from drop down list in which availability zone you want to create your volume.



Specify Snapshot ID if you want your new volume to be copied data from the snapshot, otherwise leave blank.

And select encrypt option if you want to encrypt your newly creating volume, otherwise leave un selected.



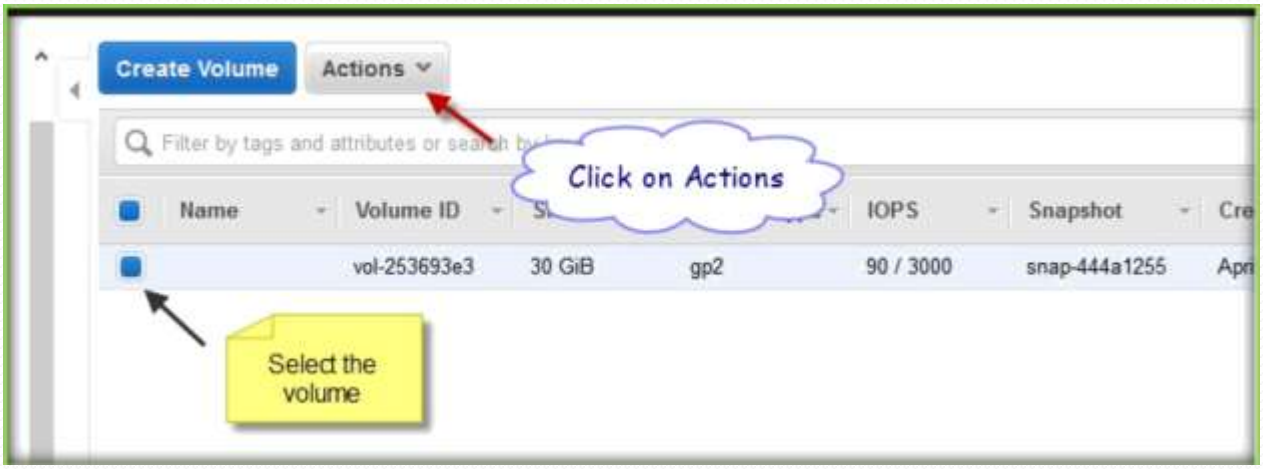
Then click on create after specifying the values required.

Your volume will start creating.

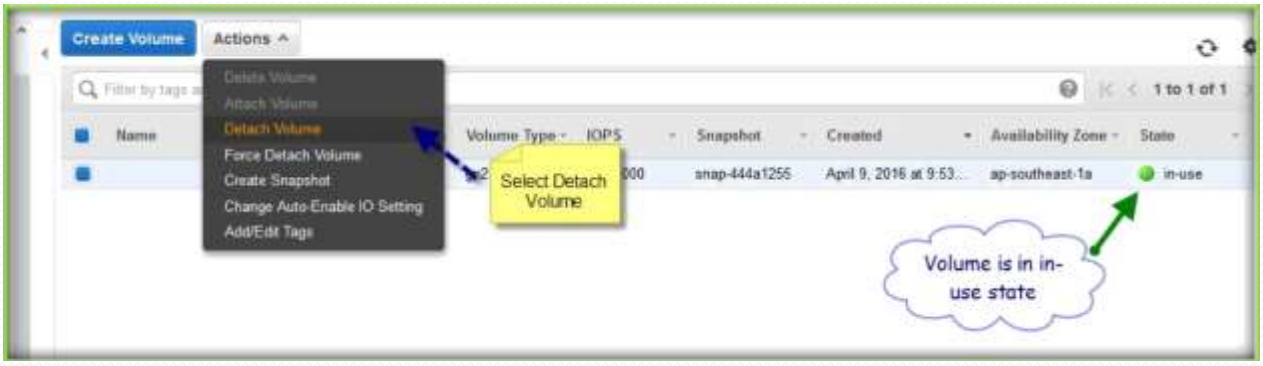
Once created it will be available state under volumes section.

DELETE EBS VOLUME

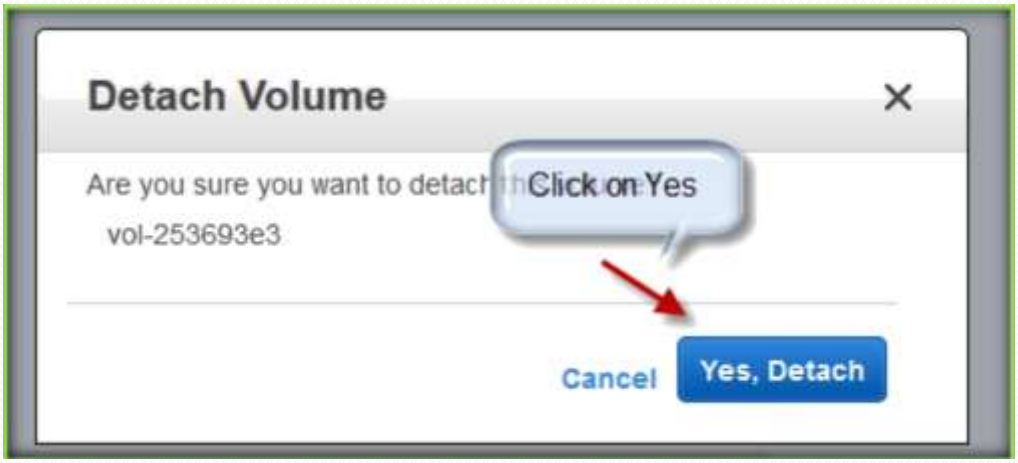
Once you logged in to AWS, go to EC2 section then go to Volumes section.
Click the volume which you want to delete and click on actions.



From the Actions menu select Detach Volume as you can see volume state is in **in-use** and those Delete volume option is un selectable.



Click on Yes, detach to detach volume from instance.



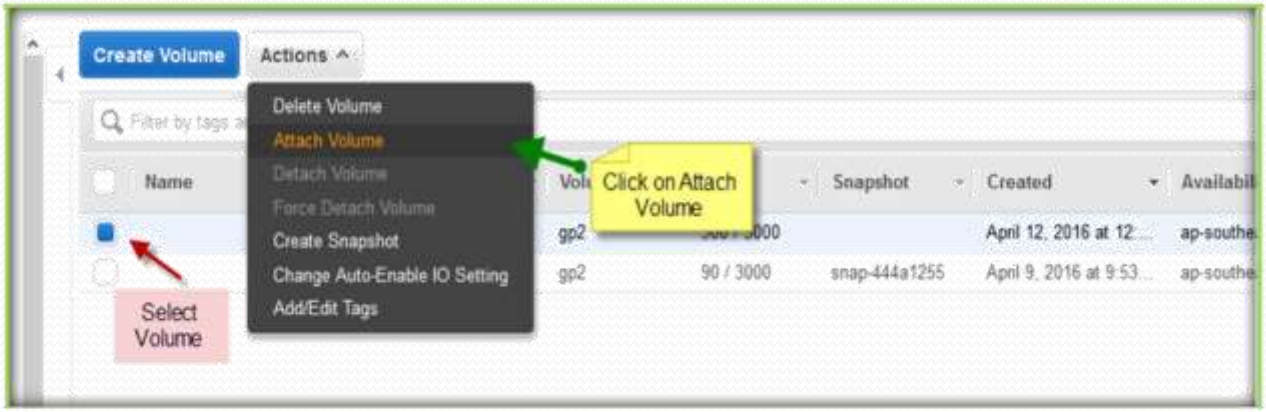
Select Volume which you want to delete then click on actions and select Delete Volume from the list.

A pop-up window will come for confirmation, then click on Yes, delete to delete.

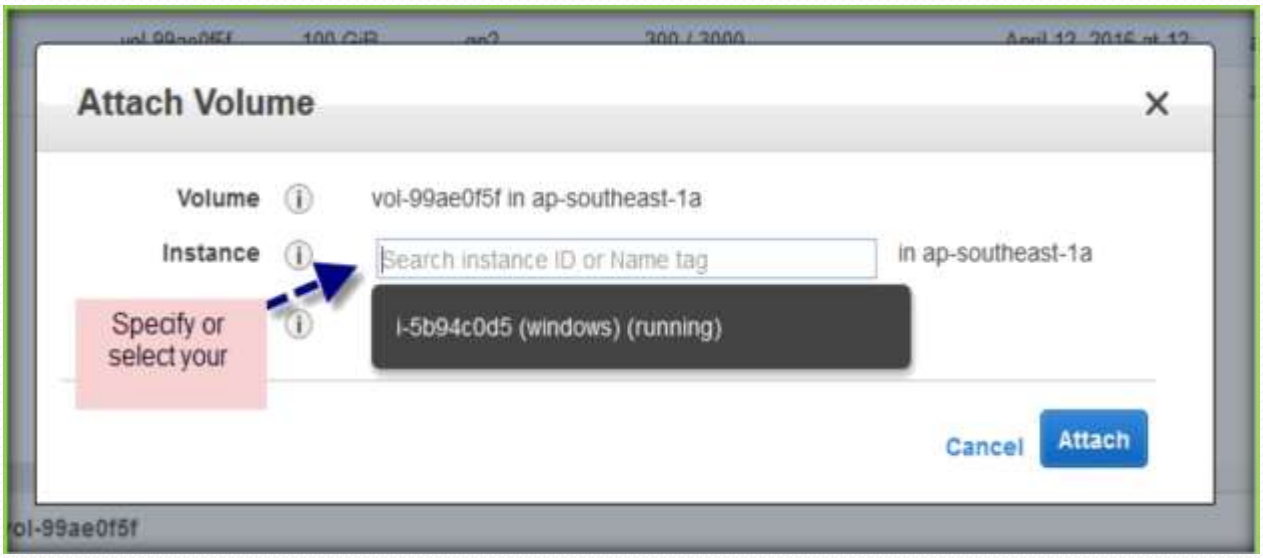
Volume will delete, and the State will change to deleting.

ATTACHING AND MOUNTING VOLUMES TO WINDOWS INSTANCES

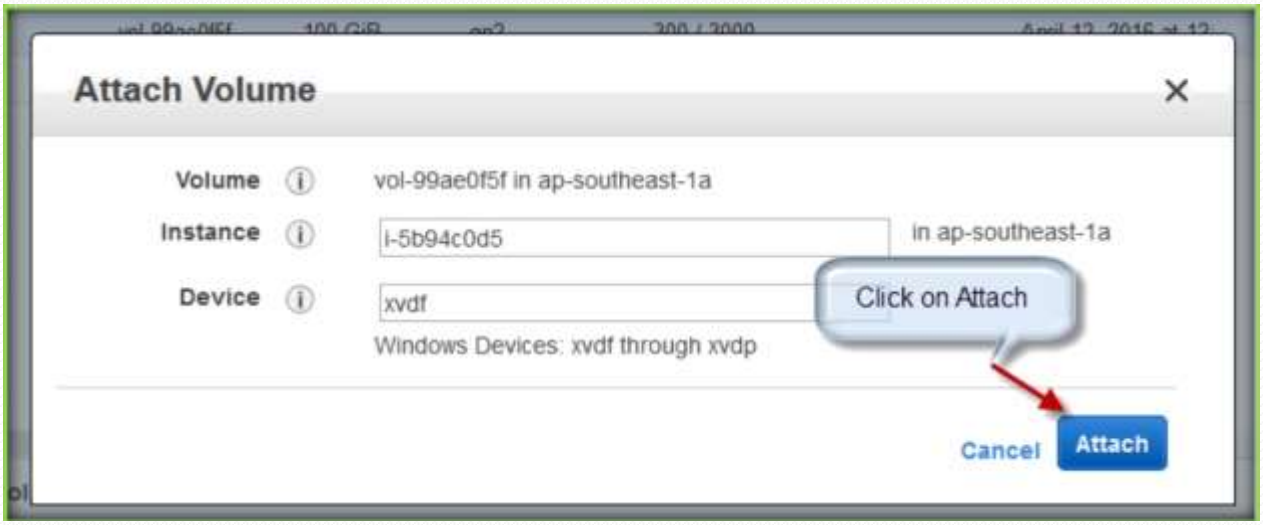
Go to volumes section and create a volume, make sure you select the availability zone same as your instance is residing.
Once created select the Volume and click on actions and select Attach Volume.



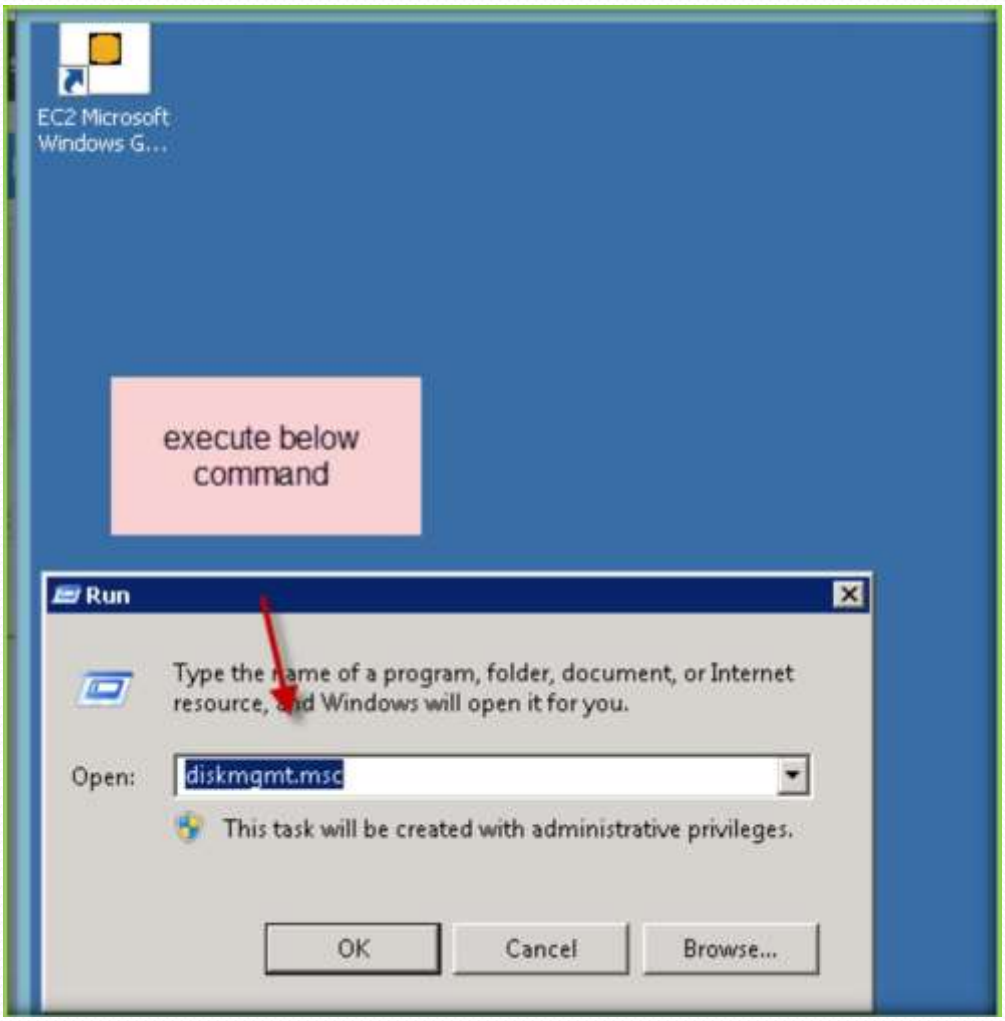
In the next page Select or search in the Instance text field and select your instance.



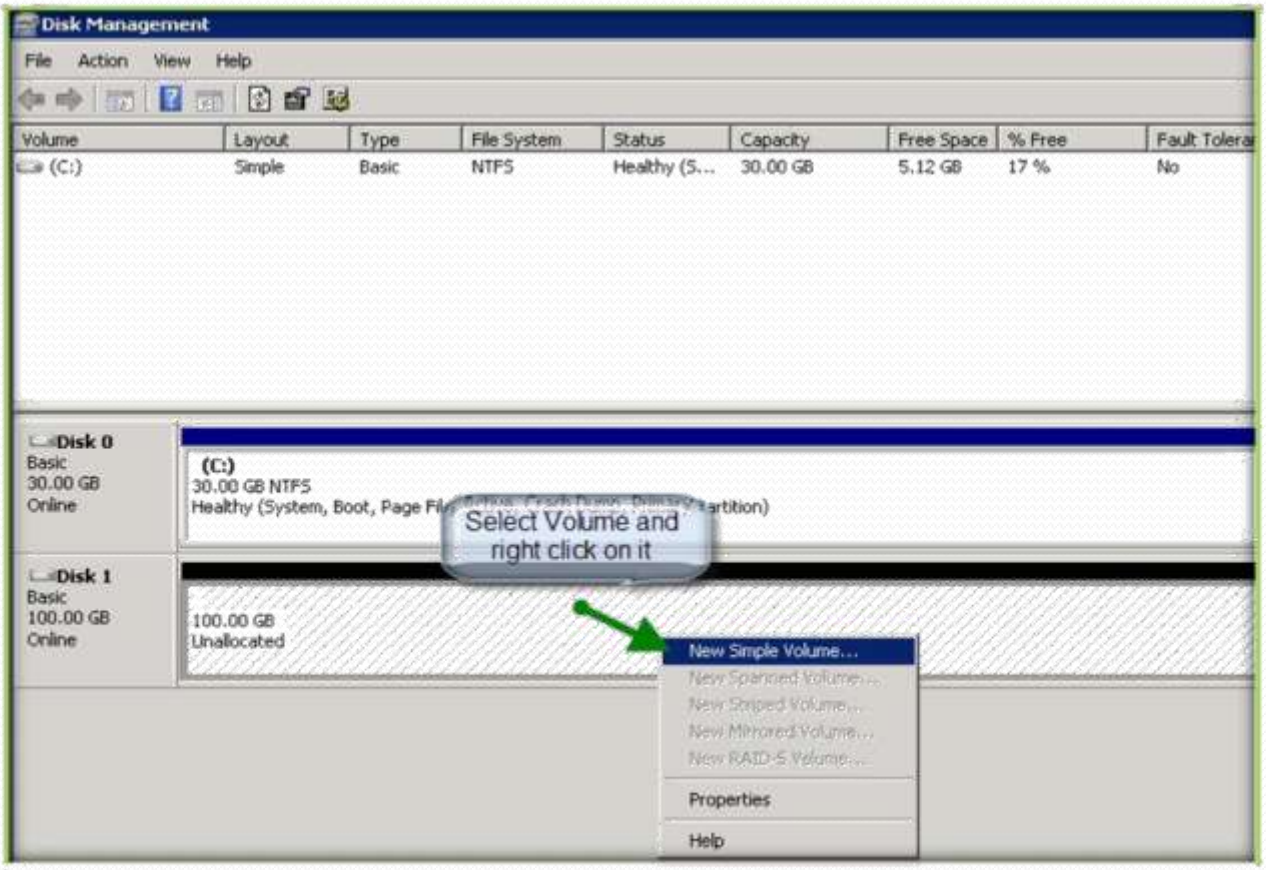
Then click on Attach button.



Then go and login to your instance, after logged in open Run and execute diskmgmt.msc command to open storage volumes attached to instance.



Then in the disk management screen select the volume which we have created and do a right click on the volume.

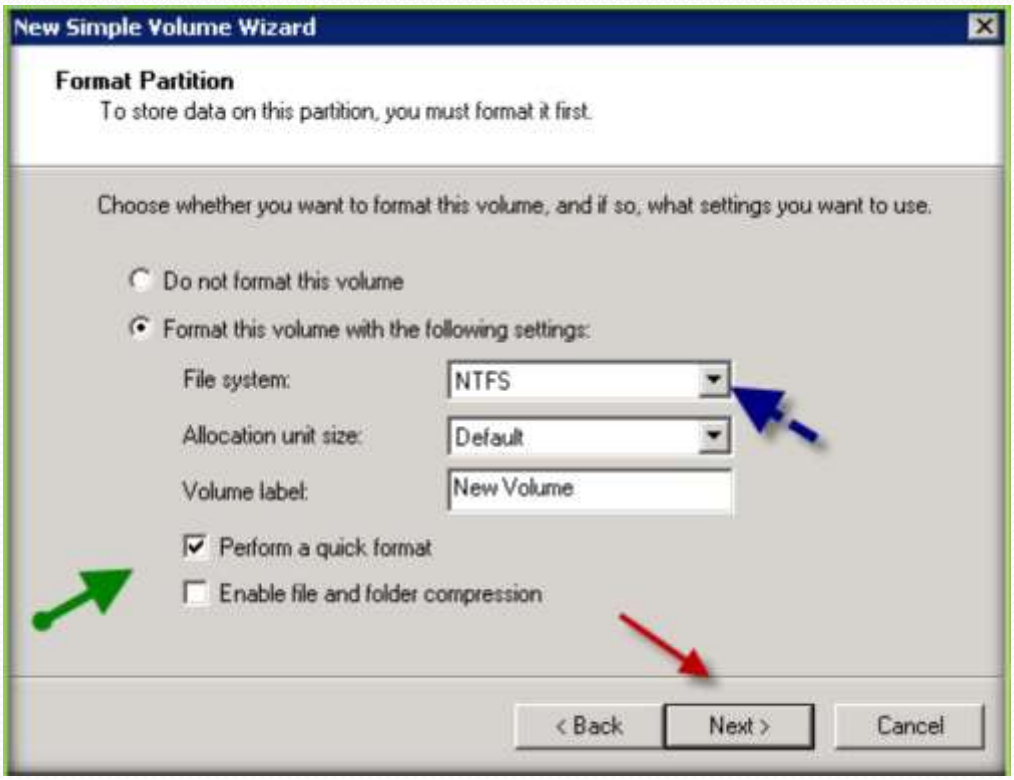


Click on Next to continue.

In the following screen specify size if you want to create more drives or leave as full size and click on Next.

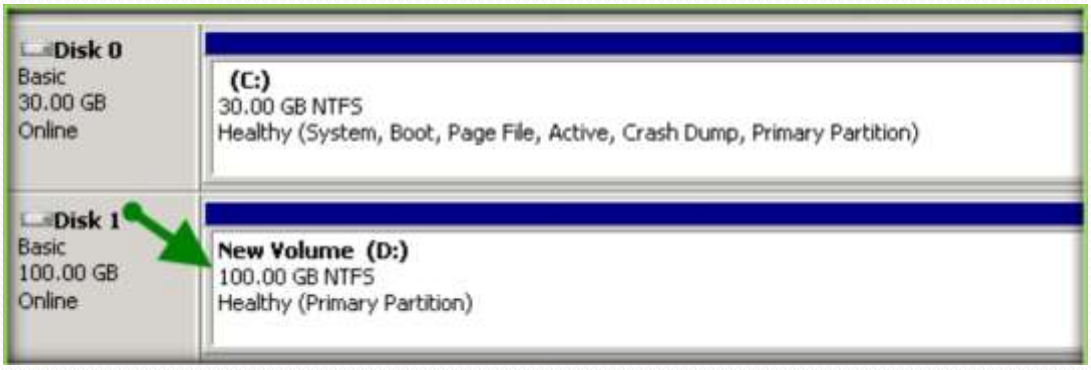
Specify a drive for the newly creating volume and click on Next.

In the screen go with default selections and click on Next.



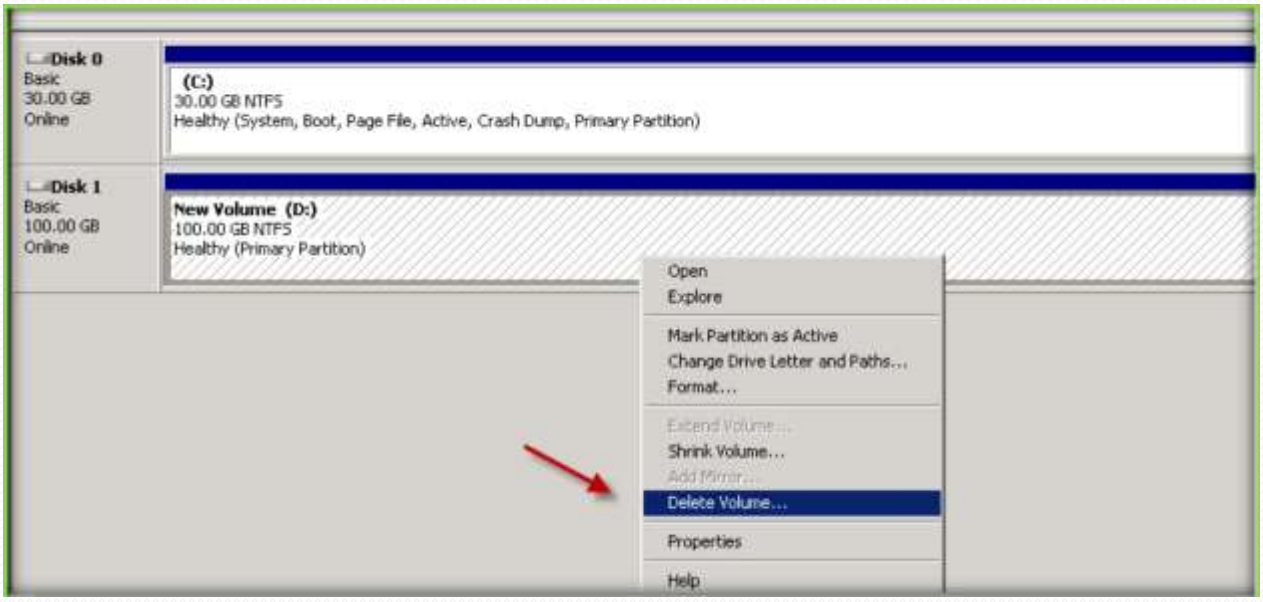
On the next screen click finish to complete the setup.

After formatting it will show as Healthy, then close disk management console.

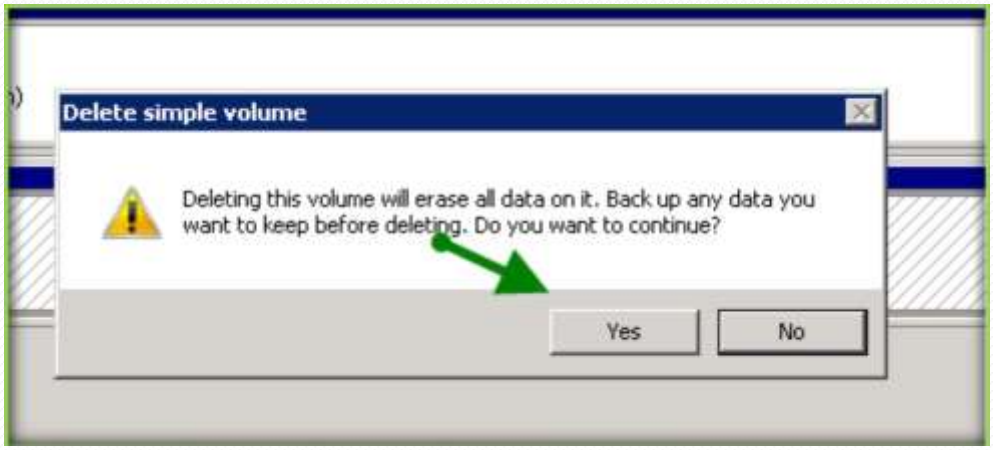


Open My Computer to see the newly formatted volume.

To unmount the Volume, once again open run and execute diskmgmt.msc to disk management console.
Then select your volume and right click on it and select Delete Volume.



Click yes to confirm.

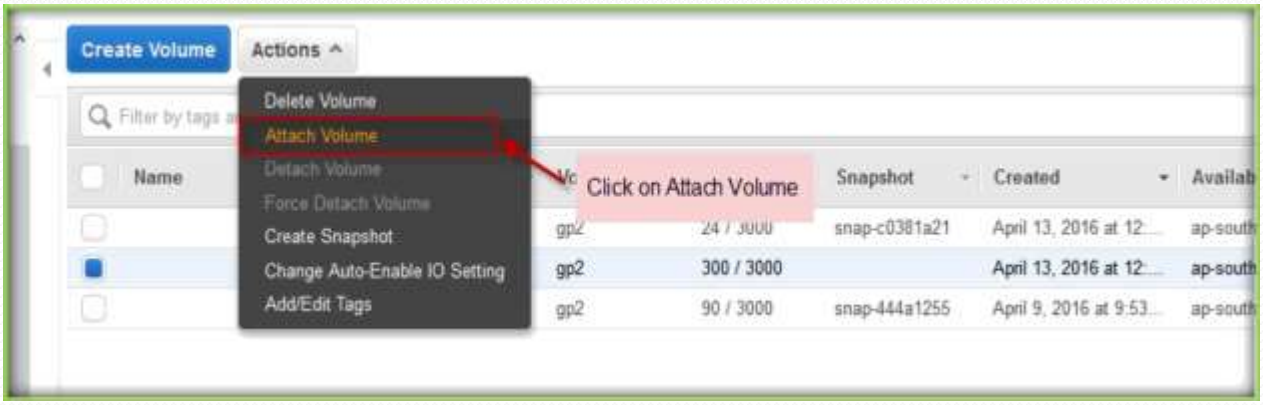


Then go to Volumes section on the EC2 page on AWS console.
Select the volume and go to Actions click on Detach Volume.

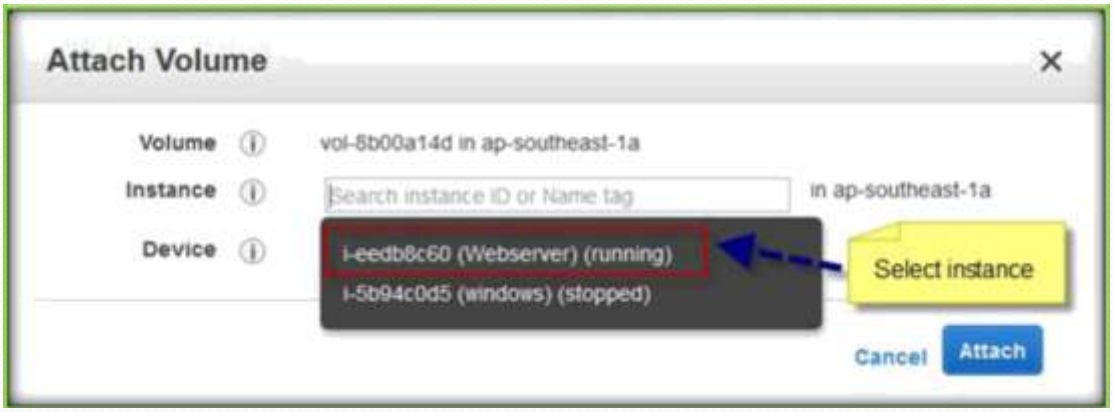
Click on Yes, detach to confirming detaching from the Instance.

ATTACHING AND MOUNTING VOLUMES TO LINUX INSTANCES

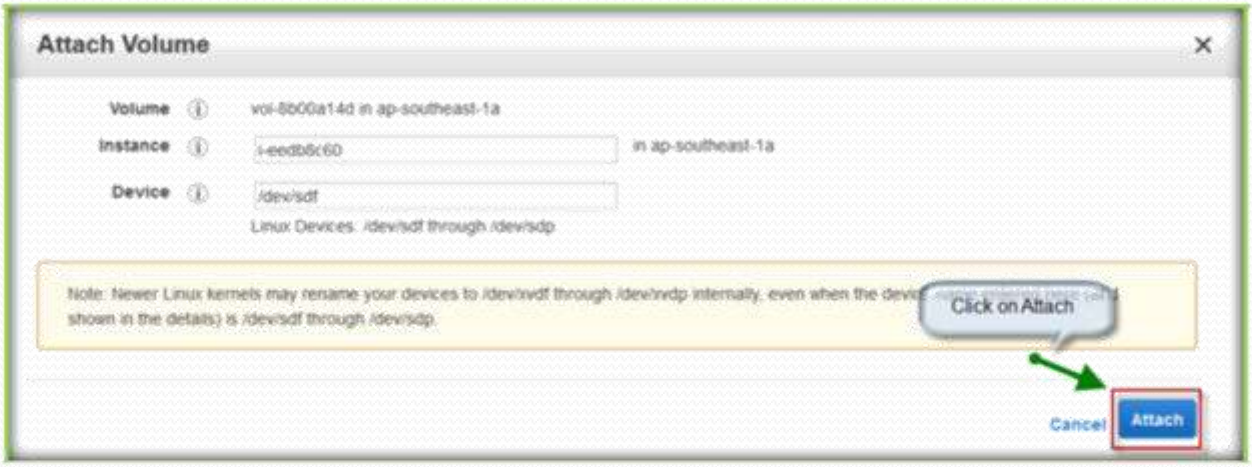
Once you logged in to AWS, go Volumes section under EC2.
Then create a volume as depicted above.
Once created select the volume and click Actions and choose Attach Volume.



Select instance from the Instance text field or specify Instance id.



Then click on Attach to attach volume to selected instance.



Then go and login to Linux and execute the below commands to format and mount to your instance.

Create the partition for newly attached volume.

```
[root@ip-172-31-25-51 ~]# fdisk /dev/xvdf
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x7ee209a3.

Command (m for help): n
Partition type:
   p primary (0 primary, 0 extended, 4
   free) e extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-209715199, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-209715199, default 209715199):
Using default value 209715199
Partition 1 of type Linux and of size 100 GiB is set

Command (m for help): p
```

Disk /dev/xvdf: 107.4 GB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes

```
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x7ee209a3

   Device Boot      Start End Blocks Id System 2048
/dev/xvdf1          209715199 104856576 83 Linux

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

Formatting newly creat ed partition.

```
[root@ip-172-31-25-51 ~]# mkfs.ext4 /dev/xvdf1
mke2fs 1.42.12 (29-Aug -2014)
Creating filesystem with  26214144 4k blocks and 6553600
inodes Filesystem UUID:  6459b7c0-440f-4fc0-9422-
94de2ee5dc34 Superblo ck backups stored on blocks:
        32768, 98304, 1638 40, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624,  11239424, 20480000, 23887872

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

Once formatted, we are  now mounting to a directory.

                         ~]# mount /dev/xvdf1
[root@ip-172-31-25-51 /mnt/
[root@ip-172-31-25-51 ~]# df -h
Filesystem      Size Used Avail Use% Mounted on
/dev/xvda1       7.8G 1.1G 6.6G 15% /
devtmpfs        490M   64K 490M 1% /dev
tmpfs           498M     0 498M 0% /dev/shm
```

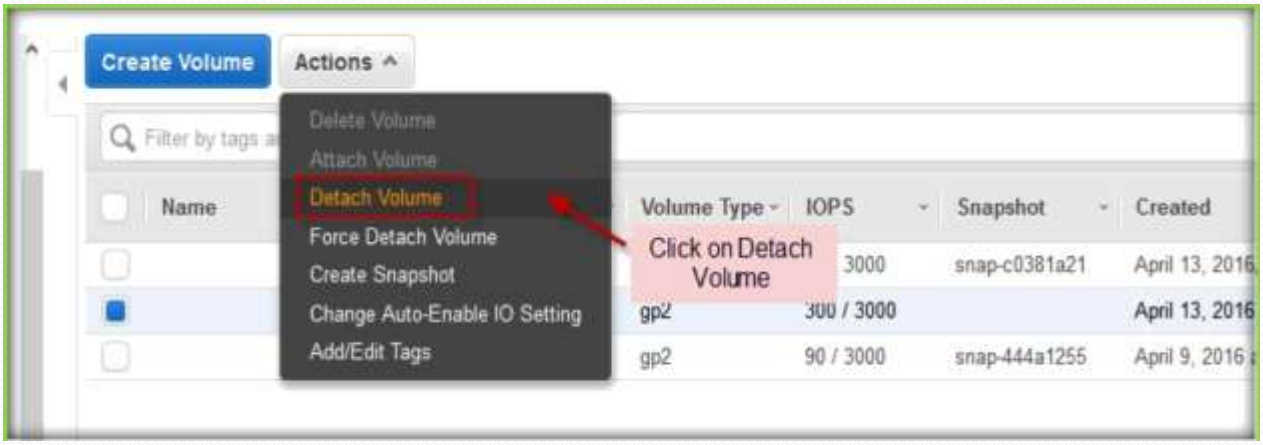
/dev/xvdf1 99G 60M 94G 1% /mnt

To unmount from the instance, follow below process.

```
[root@ip-172-31-25-51 ~]# umount /mnt/

[root@ip-172-31-25-51 ~]# df -h
Filesystem      Size Used Avail Use% Mounted on
/dev/xvda1      7.8G 1.1G 6.6G 15% /
devtmpfs        490M   64K 490M 1% /dev
tmpfs           498M     0 498M 0% /dev/shm
```

After unmounting, go to Volumes section, select the volume and click on Actions. Under the Actions, click on Detach Volume.



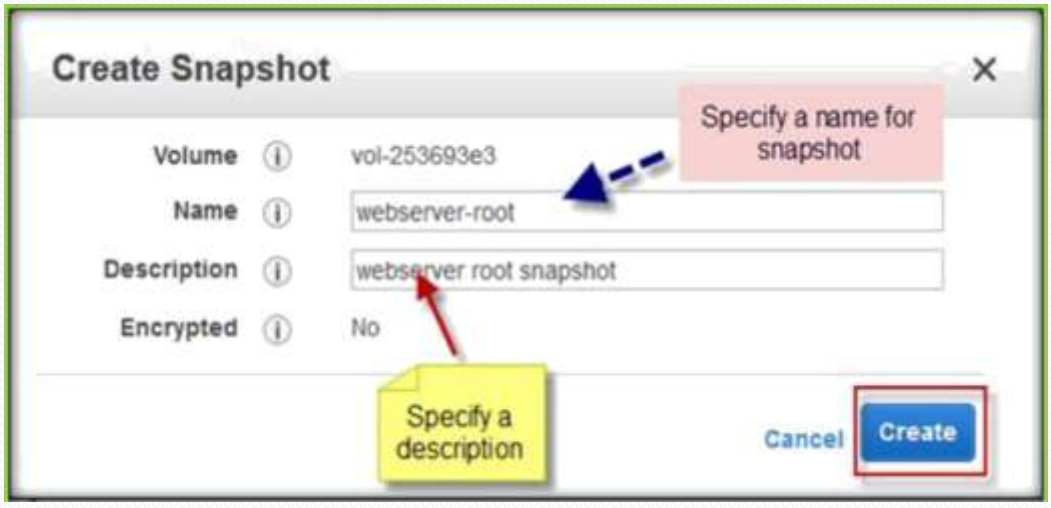
Click on Yes, detach to detach volume.

Once Volume detached, you can select and delete the volume.

CREATING AND DELETING SNAPSHOTS

Go to Volumes section under EC2 on AWS console.
Select the volume which you want to take a snapshot and click on Actions.
Under Actions menu, select Create Snapshot.

In the next page specify name for snapshot and add description, click on Create button to create.

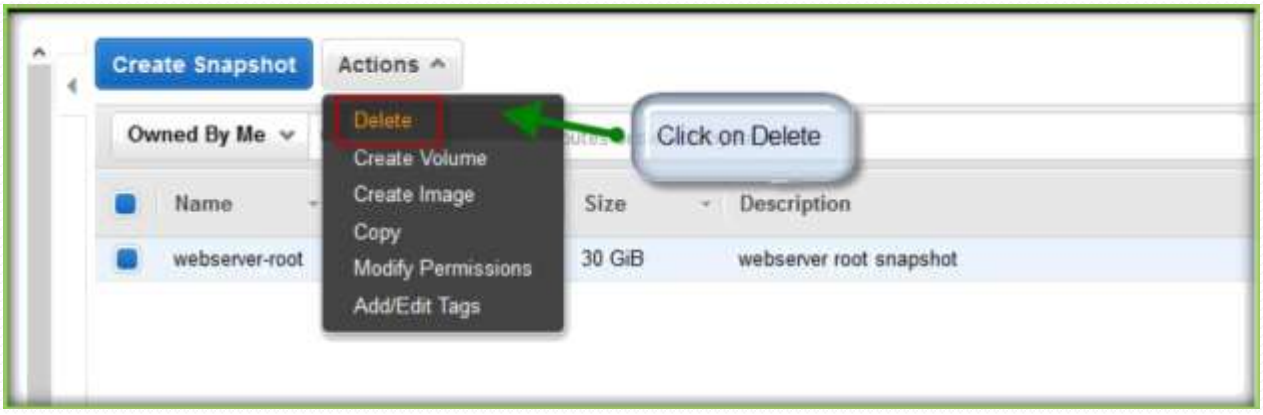


A confirmation box will pop up with Snapshot id.

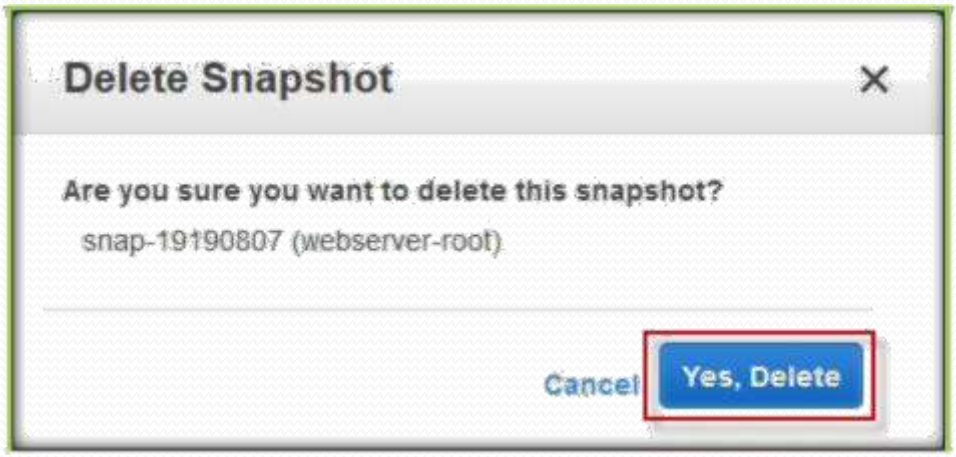
Click on the snapshot id to go the snapshot section or go to Snapshots under ELASTIC BLOCK STORE on EC2 left pane. You will find your snapshot which is created.

DELETING SNAPSHOTS

Go to Snapshots section under EC2 page, then select your snapshot which you want to delete.
Click on Actions, select Delete to delete the snapshot.



A confirmation dialog popup will show, click on Yes, delete to delete the snapshot.

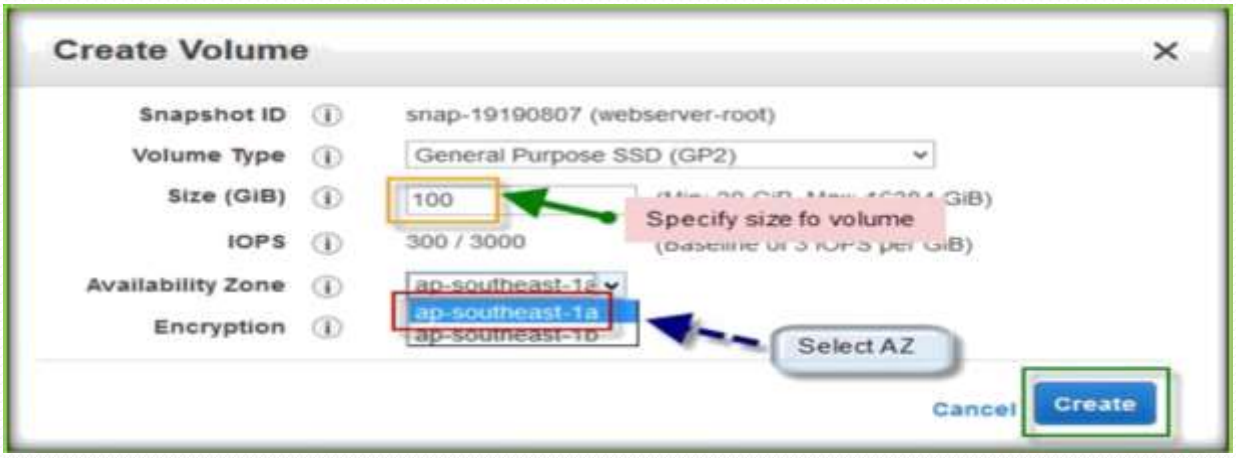


CREATING VOLUMES FROM SNAPSHOTS

Go to Snapshots section under EC2 page.
Select the snapshot and click on Actions, then select Create Volume to create a new volume.

In the next page, select the Volume Type from the drop-down list.

Specify size in size text field, select availability zone from drop down list.
Then click on create button to create a volume.



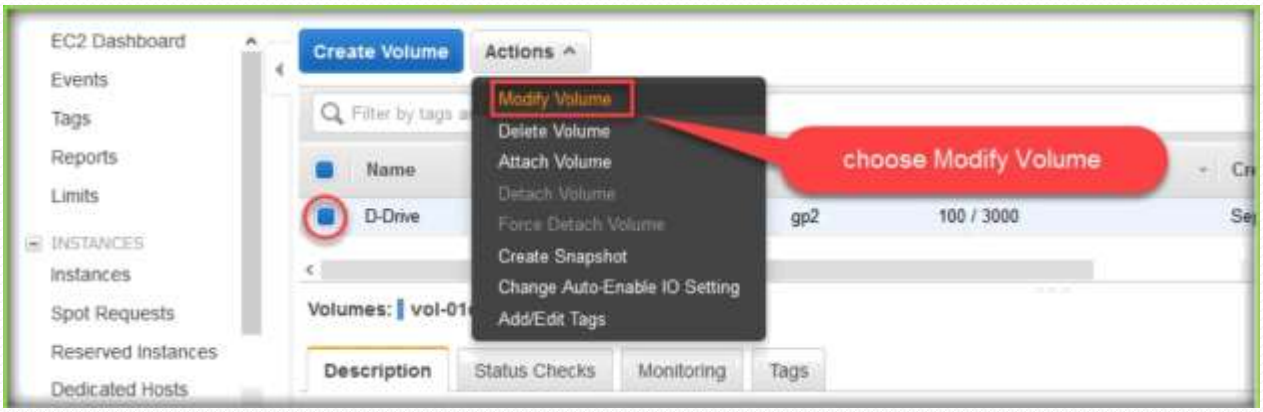
A confirmation pops up with volume id will show, click on volume id to go to the volume.

After creation volume, you can attach to other instances.

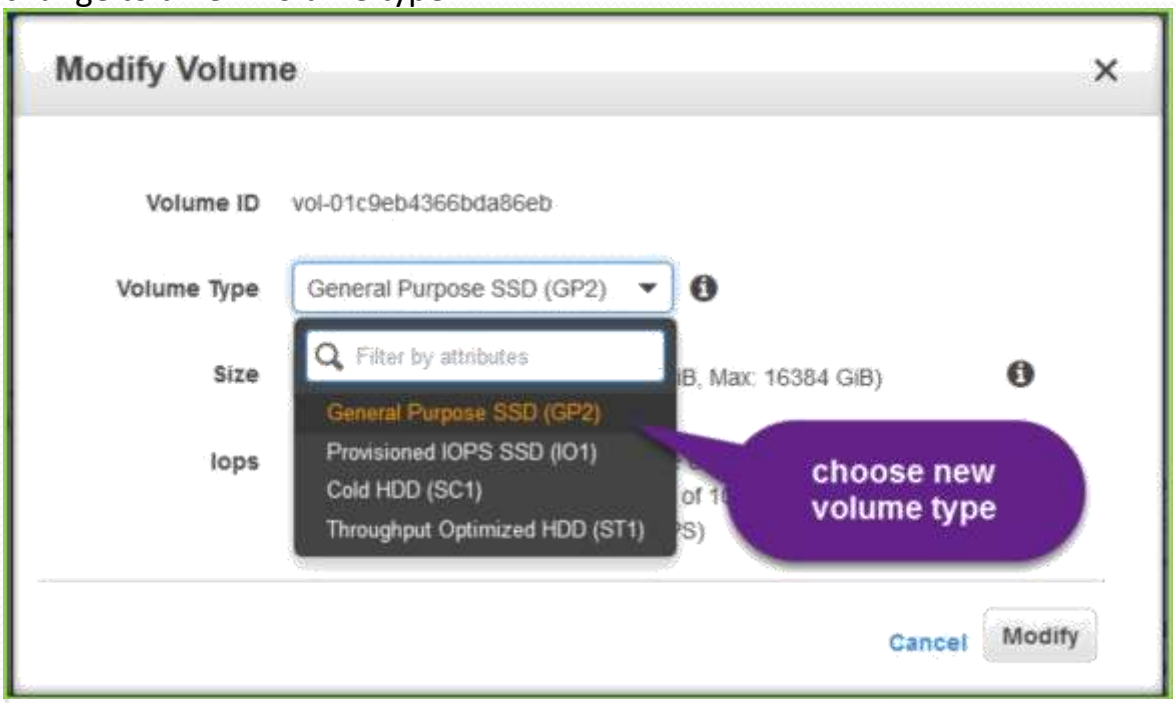
MODIFY VOLUMES

Amazon EBS volume is attached to an EC2 instance type, can increase its size, change its volume type, or (for an io1 volume) adjust its IOPS performance, all without detaching it. You can apply these changes to detached volumes as well.

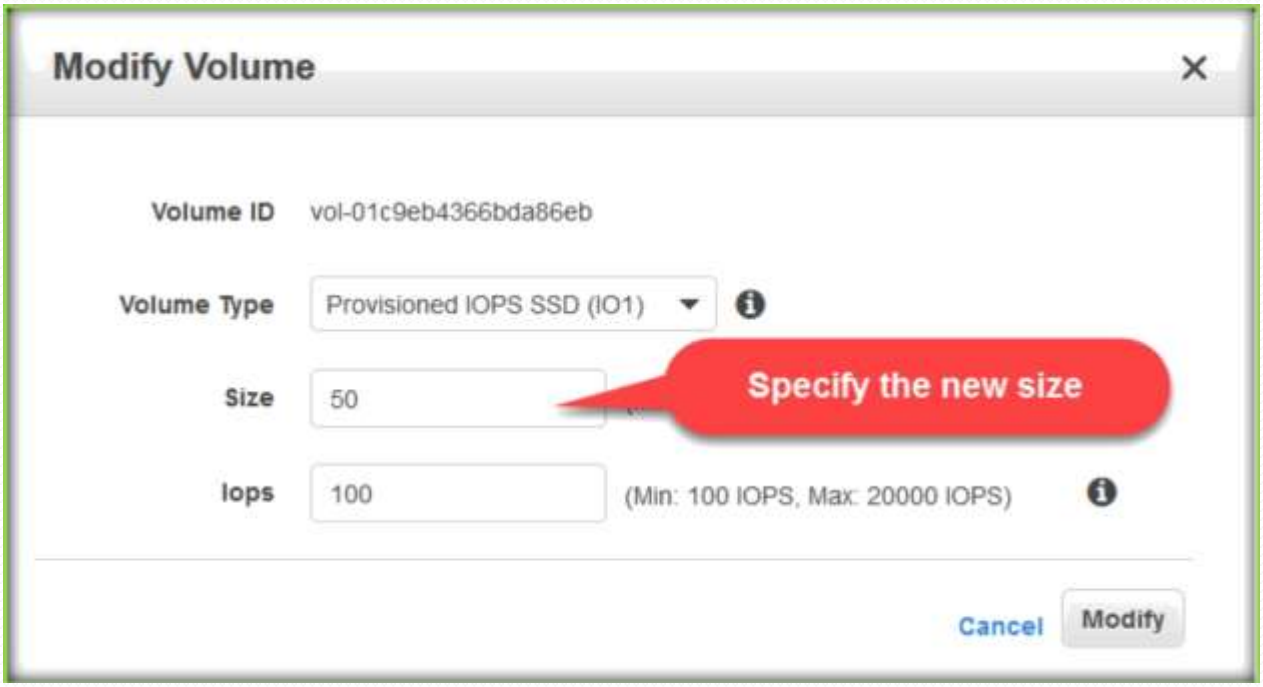
Go to volumes, choose the volume which you want to modify, then select and go to Actions tab then choose modify volume.



On the modify volume dialog box, choose a different volume type if you want to change to a new volume type.



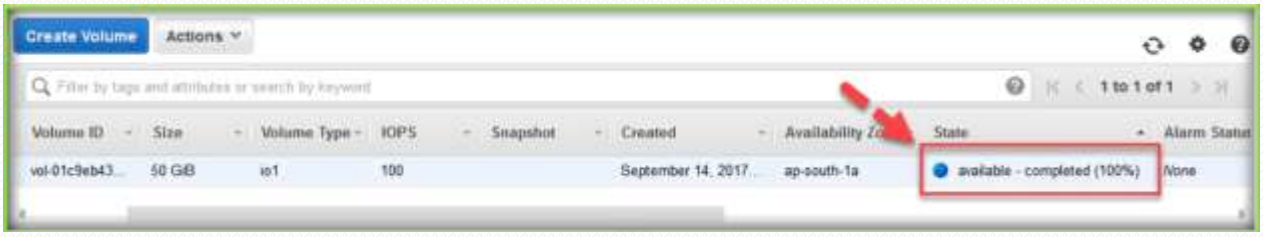
Specify the new size and IOPS if it is Provisioned IOPS volume.



Then click on modify.

You can view the volume state about the modification.

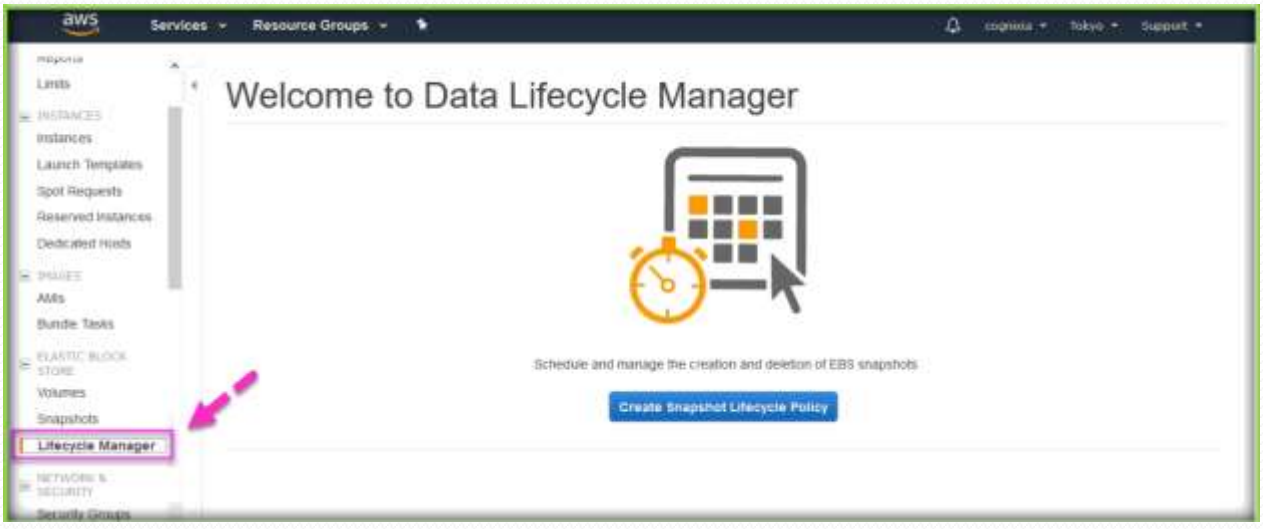
Once done state will show completed.



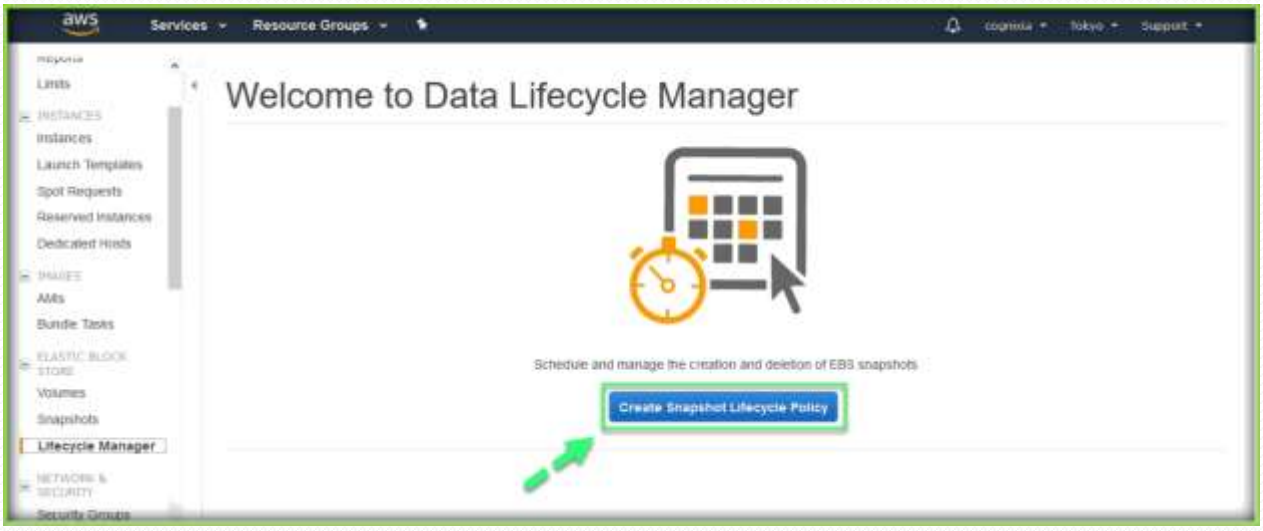
Then we must go to the respective operating system and extend the volume sizes to view the new size in the operating systems.

Creating and Using a Lifecycle Policy

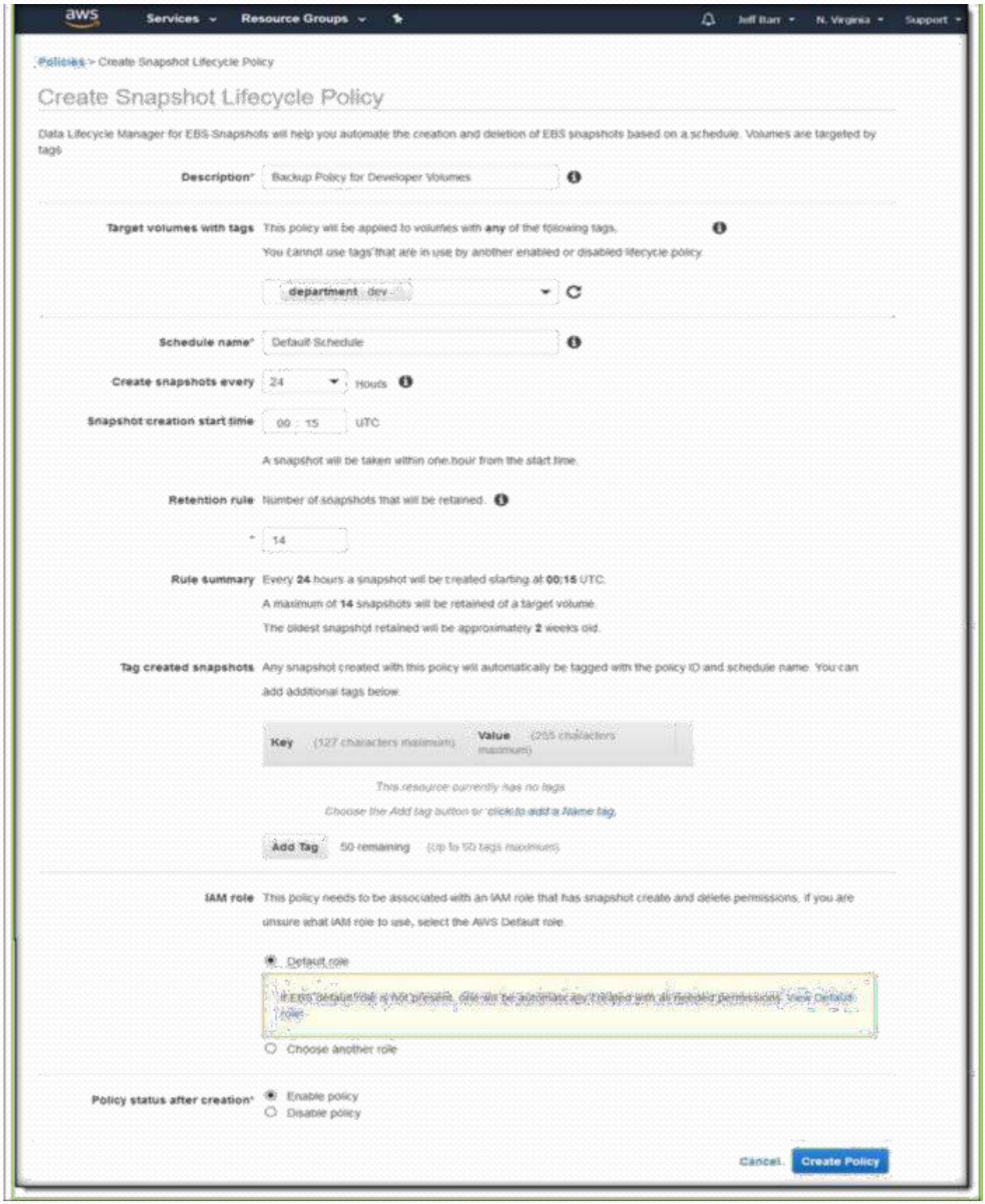
Lifecycle Manager uses lifecycle policies to figure out when to run, which volumes to snapshot, and how long to keep the snapshots around.
I access the Lifecycle Manager from the Elastic Block Store section of the menu



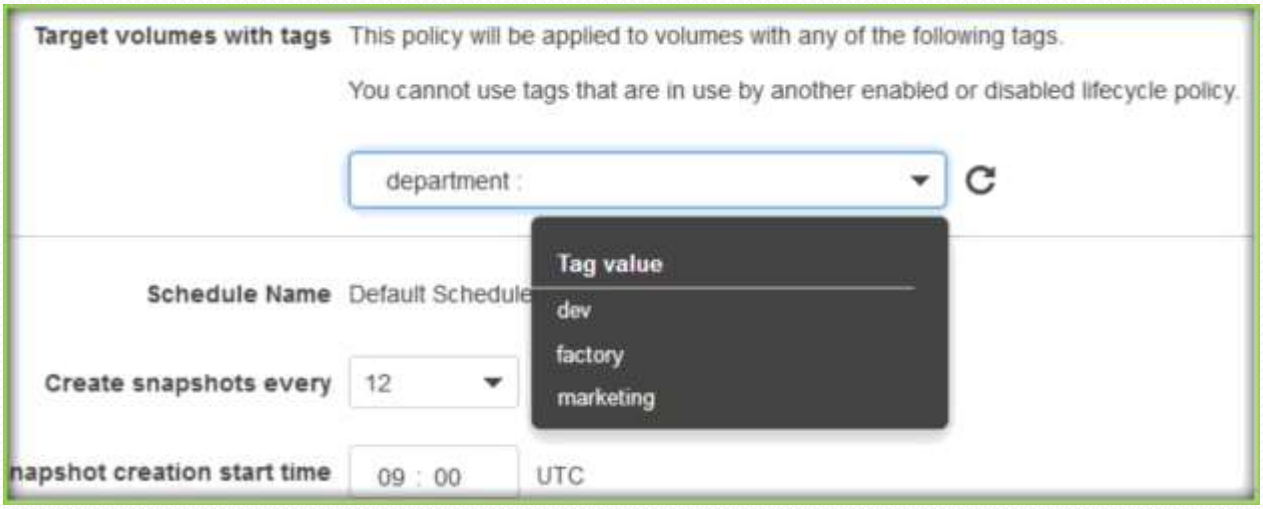
Then I click Create Snapshot Lifecycle Policy to proceed:



Then I create my first policy


aws
Services
Resource Groups
Jeff Barr
N. Virginia
Support
Policies > Create Snapshot Lifecycle Policy
Create Snapshot Lifecycle Policy
Data Lifecycle Manager for EBS Snapshots will help you automate the creation and deletion of EBS snapshots based on a schedule. Volumes are targeted by tags
Description*
Backup Policy for Developer Volumes
Target volumes with tags
This policy will be applied to volumes with any of the following tags.
You cannot use tags that are in use by another enabled or disabled lifecycle policy.
department dev
Schedule name*
Default Schedule
Create snapshots every
24
Hours
Snapshot creation start time
00 : 15
UTC
A snapshot will be taken within one hour from the start time.
Retention rule
Number of snapshots that will be retained.
14
Rule summary
Every 24 hours a snapshot will be created starting at 00:15 UTC.
A maximum of 14 snapshots will be retained of a target volume.
The oldest snapshot retained will be approximately 2 weeks old.
Tag created snapshots
Any snapshot created with this policy will automatically be tagged with the policy ID and schedule name. You can add additional tags below.
Key
(127 characters maximum)
Value
(255 characters maximum)
This resource currently has no tags
Choose the Add tag button or click to add a Name tag.
Add Tag
50 remaining
(Up to 50 tags maximum)
IAM role
This policy needs to be associated with an IAM role that has snapshot create and delete permissions. If you are unsure what IAM role to use, select the AWS Default role.
Default role
If this default role is not present, one will be automatically created with all needed permissions. View Default role
Choose another role
Policy status after creation*
Enable policy
Disable policy
Cancel
Create Policy

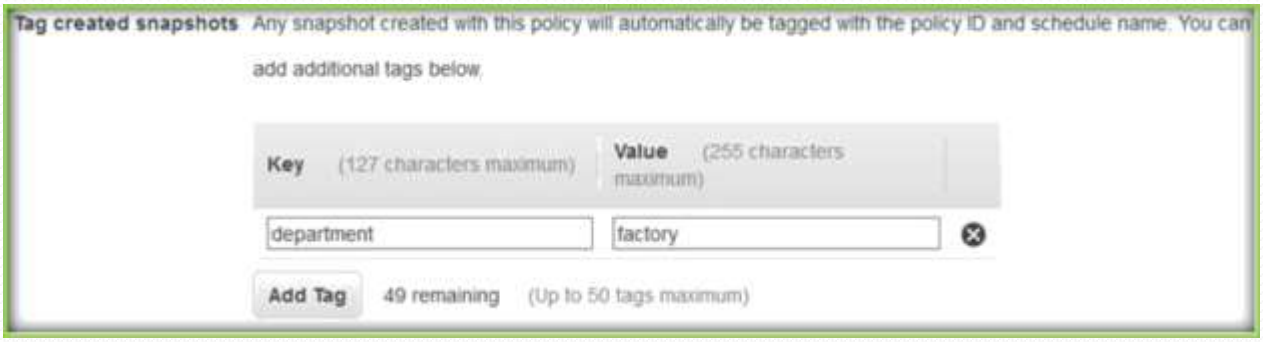
I use tags to specify the volumes that the policy applies to. If I specify multiple tags, then the policy applies to volumes that have any of the tags



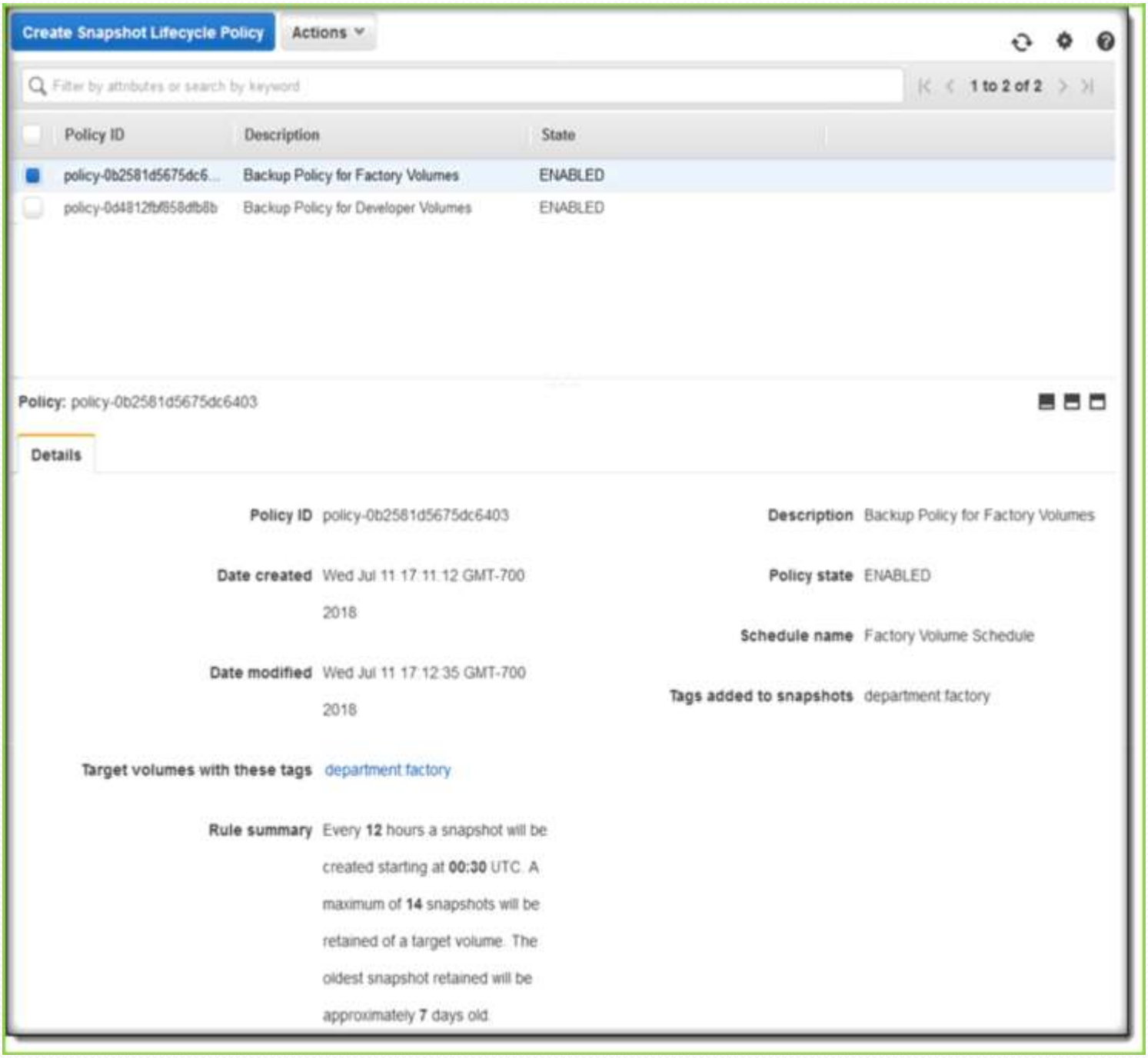
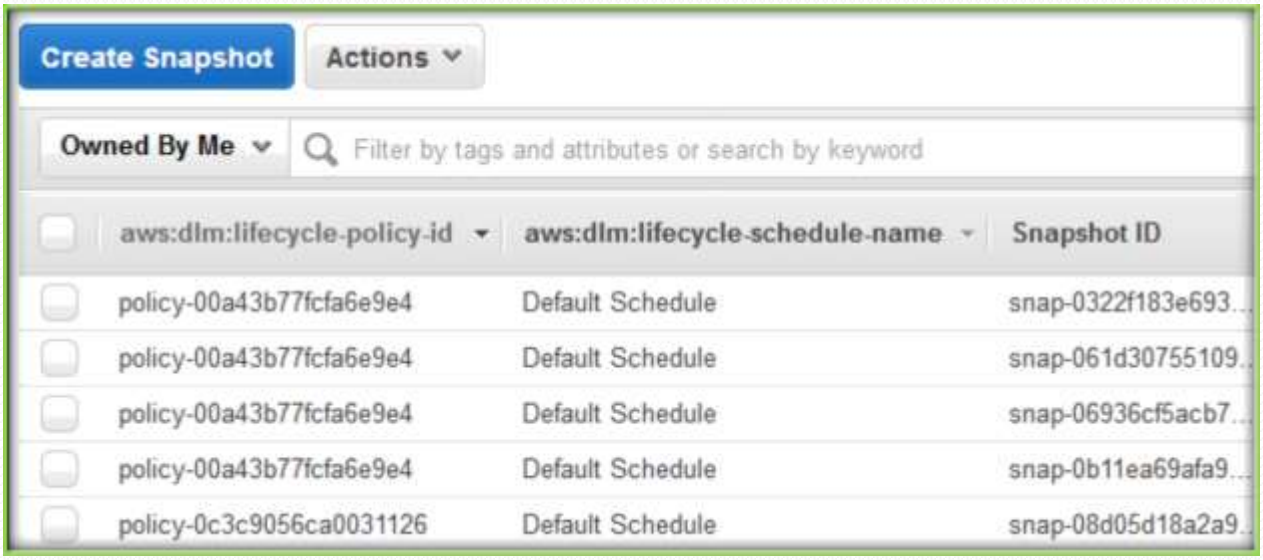
I can create snapshots at 12- or 24-hour intervals, and I can specify the desired snapshot time. Snapshot creation will start no more than an hour after this time, with completion based on the size of the volume and the degree of change since the last snapshot.

I can use the built-in default IAM role or I can create one of my own. If I use my own role, I need to enable the EC2 snapshot operations and all the DLM (Data Lifecycle Manager) operations.

Newly created snapshots will be tagged with the aws: dlm:lifecycle-policy-id and aws:dlm:lifecycle-schedule-nameautomatically; I can also specify up to 50 additional key/value pairs for each policy



I can see all my policies at a glance



I took a short break and came back to find that the first set of snapshots had been created, as expected (I configured the console to show the two tags created on the snapshots)