Roll No: 20BCE204
Name: Dhyan Patel
Course: Cloud Computing
Practical No: 5

Aim: Working with EBS

EBS – Elastic block store, a key underlying storage mechanism for Amazon EC2 instances.

Amazon Elastic Block Store (EBS) is a scalable block storage service offered by Amazon Web Services (AWS). It is designed to provide persistent block-level storage volumes for use with Amazon Elastic Compute Cloud (EC2) instances. EBS volumes are highly available, reliable, and durable storage systems that can be attached to EC2 instances as virtual hard drives.

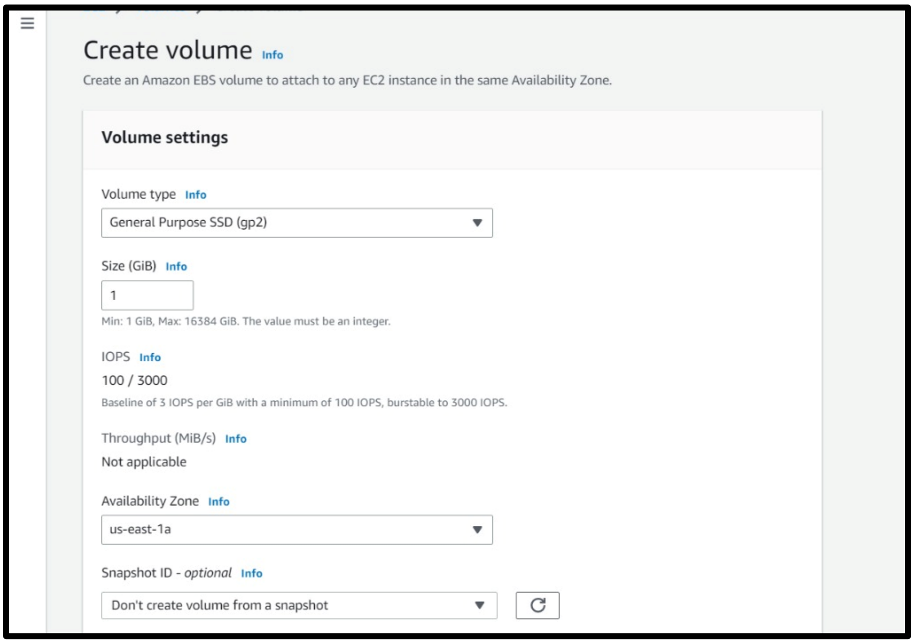
EBS volumes are stored in Amazon's redundant storage infrastructure, which provides high durability and availability. EBS volumes can be created and deleted independently of EC2 instances, and can be moved between instances. They are also designed to be resilient to failures, and can automatically recover from hardware or software failures.

EBS is a reliable and scalable storage solution for EC2 instances, providing persistent block-level storage that is highly available and durable.

EBS volumes are available in two types: solid-state drives (SSDs) and hard disk drives (HDDs).

Now, let’s start the lab

Here, we are already given with an instance name Lab. Now, let’s create a volume from the EC2 dashboard.



Create volume

Info

Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.

Volume settings

Volume type Info

General Purpose SSD (gp2)

Size (GiB) Info

1

Min: 1 GiB, Max: 16384 GiB. The value must be an integer.

IOPS Info

100 / 3000

Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS.

Throughput (MiB/s) Info

Not applicable

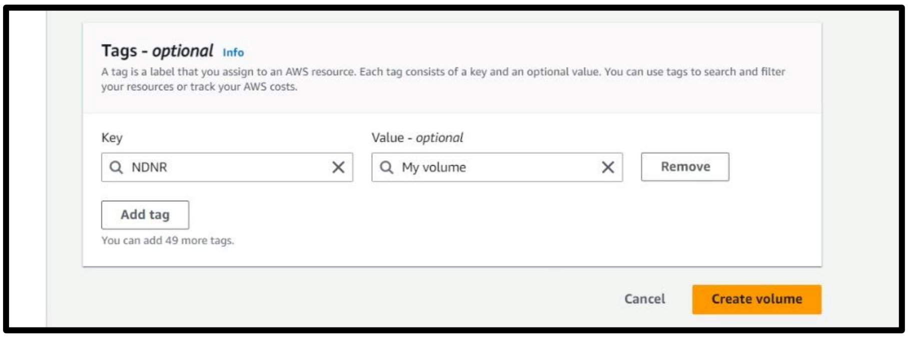
Availability Zone Info

us-east-1a

Snapshot ID - *optional* Info

Don't create volume from a snapshot

Here to create our volume we need to specify our volume type, size of volume and availability zone. After that we will be assign a tag by providing a key value pair to this volume.



Tags - *optional* Info

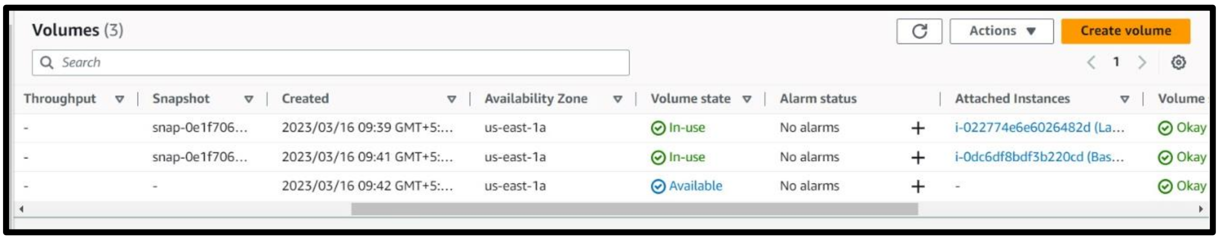
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*
NDNR	My volume

You can add 49 more tags.

Now, click on ***Create Volume*** button.

After creating the volume, it will take few minutes to move from ***Creating state*** to ***available state***.



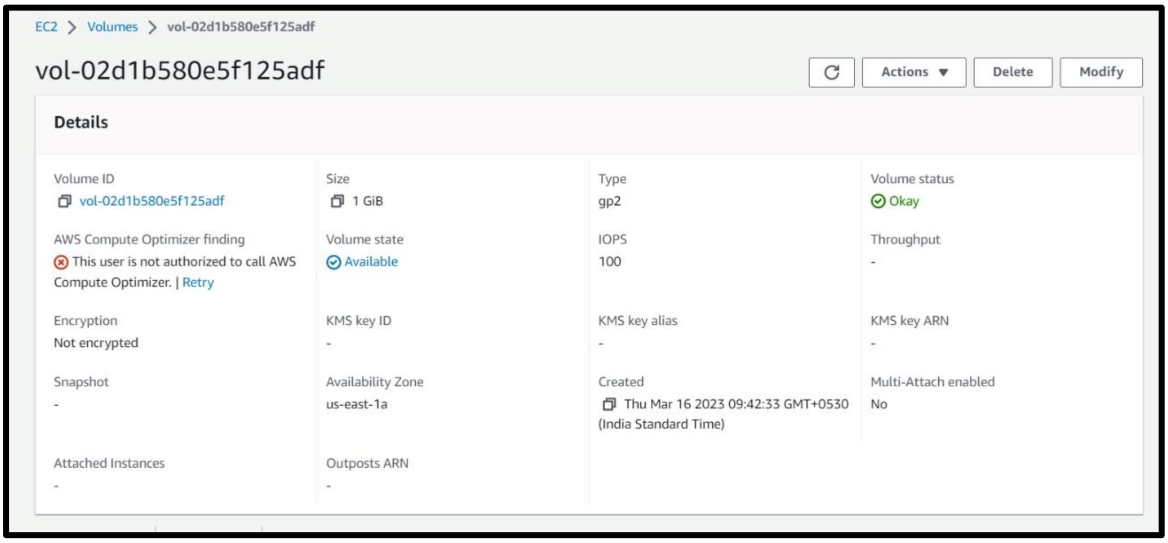
Volumes (3)

Throughput	Snapshot	Created	Availability Zone	Volume state	Alarm status	Attached Instances	Volume
-	snap-0e1f706...	2023/03/16 09:39 GMT+5:...	us-east-1a	✓ In-use	No alarms +	i-022774e6e6026482d (La...	✓ Okay
-	snap-0e1f706...	2023/03/16 09:41 GMT+5:...	us-east-1a	✓ In-use	No alarms +	i-0dc6df8bdf3b220cd (Bas...	✓ Okay
-	-	2023/03/16 09:42 GMT+5:...	us-east-1a	✓ Available	No alarms +	-	✓ Okay

Here, our volume is now available after refreshing the page.

Let’s now attach this volume to our EC2 instance name LAB.

For that, open the volume that we have just created.



vol-02d1b580e5f125adf

Details

Volume ID: vol-02d1b580e5f125adf

Size: 1 GiB

Type: gp2

Volume status: Okay

AWS Compute Optimizer finding: This user is not authorized to call AWS Compute Optimizer. | Retry

Volume state: Available

IOPS: 100

Throughput: -

Encryption: Not encrypted

KMS key ID: -

KMS key alias: -

KMS key ARN: -

Snapshot: -

Availability Zone: us-east-1a

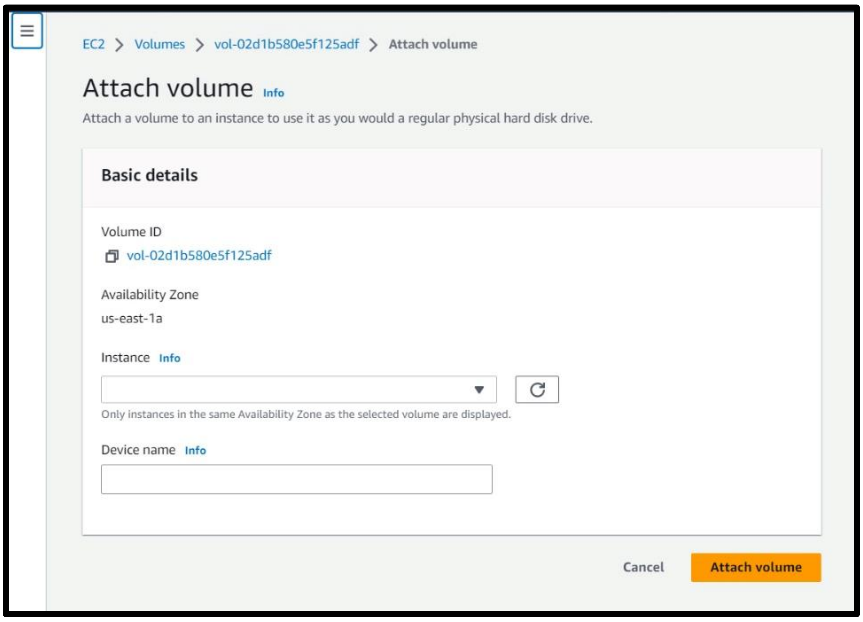
Created: Thu Mar 16 2023 09:42:33 GMT+0530 (India Standard Time)

Multi-Attach enabled: No

Attached Instances: -

Outposts ARN: -

Click on ***Action*** and then on ***Attach Volume.***



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-02d1b580e5f125adf

Availability Zone: us-east-1a

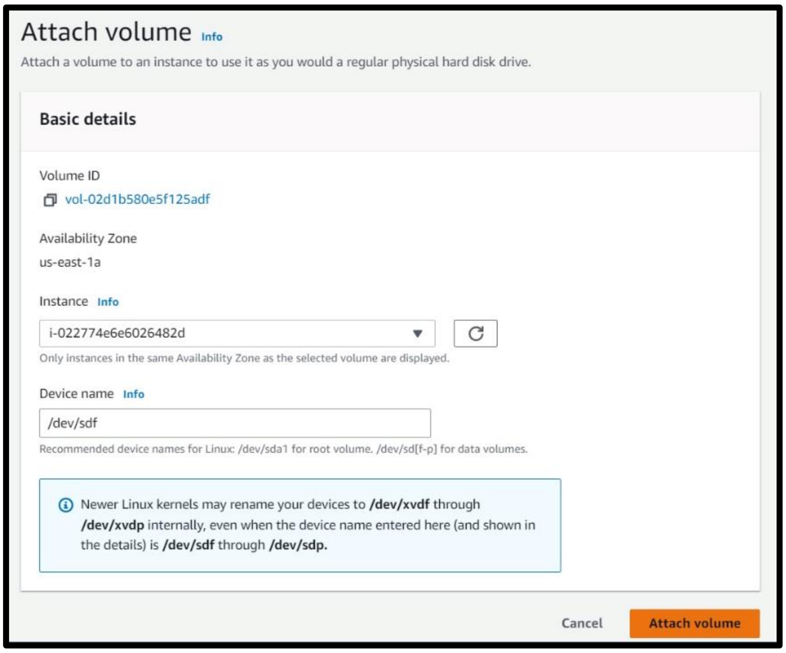
Instance Info

Only instances in the same Availability Zone as the selected volume are displayed.

Device name Info

Now choose the instance for which we need to use this volume.

Also specify the device name as ***/dev/sdf*** and then click on the ***Attach Volume*** button



Attach volume Info

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

Basic details

Volume ID
vol-02d1b580e5f125adf

Availability Zone
us-east-1a

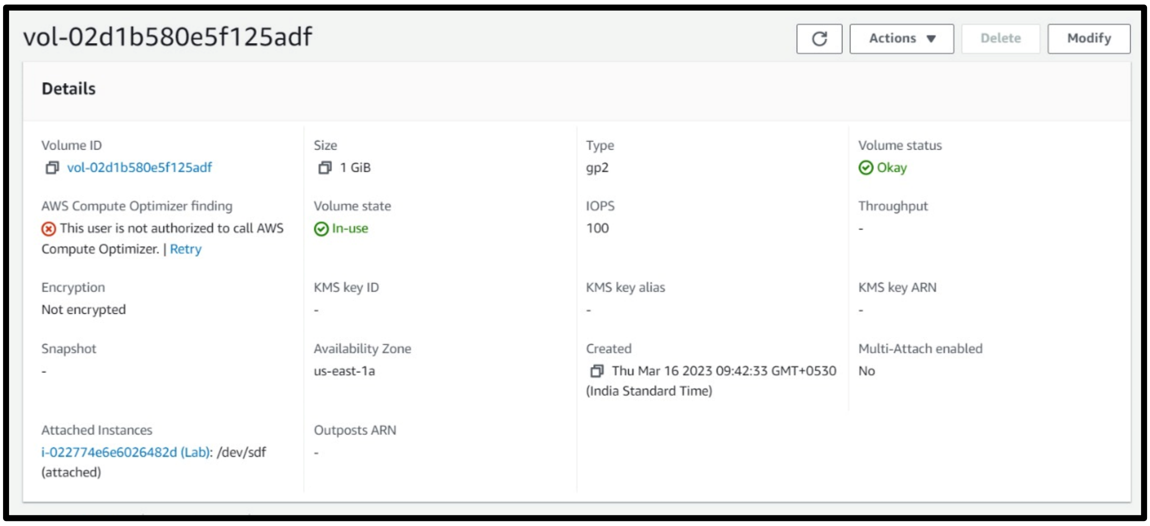
Instance Info
i-022774e6e6026482d

Only instances in the same Availability Zone as the selected volume are displayed.

Device name Info
/dev/sdf

Recommended device names for Linux: /dev/sda1 for root volume. /dev/sd[f-p] for data volumes.

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.**



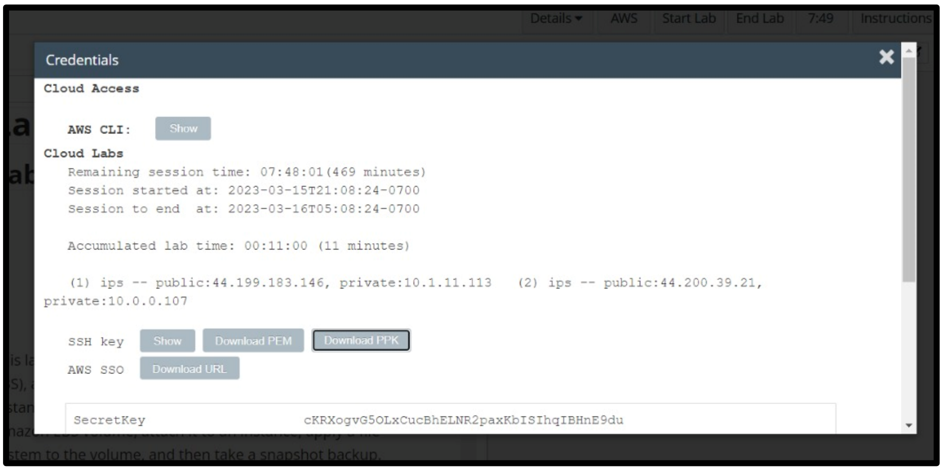
vol-02d1b580e5f125adf

Details

Volume ID vol-02d1b580e5f125adf	Size 1 GiB	Type gp2	Volume status Okay
AWS Compute Optimizer finding This user is not authorized to call AWS Compute Optimizer. | Retry	Volume state In-use	IOPS 100	Throughput -
Encryption Not encrypted	KMS key ID -	KMS key alias -	KMS key ARN -
Snapshot -	Availability Zone us-east-1a	Created Thu Mar 16 2023 09:42:33 GMT+0530 (India Standard Time)	Multi-Attach enabled No
Attached Instances i-022774e6e6026482d (Lab): /dev/sdf (attached)	Outposts ARN -		

Now, we will mount this volume on over EC2 instance named LAB from our PC using ***Putty.***

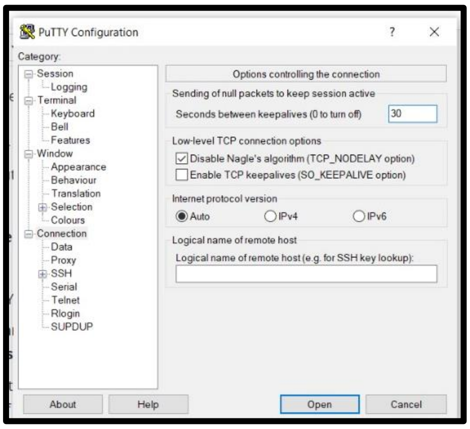
For that we will require the credential, to get that we will click on ***details*** button which is available on the Lab manual page. From their we will download the ppk credentials.



Credentials

```
Cloud Access

   AWS CLI:  Show
Cloud Labs
   Remaining session time: 07:48:01(469 minutes)
   Session started at: 2023-03-15T21:08:24-0700
   Session to end  at: 2023-03-16T05:08:24-0700

   Accumulated lab time: 00:11:00 (11 minutes)

   (1) ips -- public:44.199.183.146, private:10.1.11.113   (2) ips -- public:44.200.39.21,
private:10.0.0.107

   SSH key  Show  Download PEM  Download PPK
   AWS SSO  Download URL

   SecretKey                cKRXogvG5OLxCucBhELNR2paxKbISIhqIBHnE9du
```

Now open putty, go to ***Connections*** and in that change the ***Seconds between keepalives*** to ***30.***



PuTTY Configuration

Options controlling the connection

Sending of null packets to keep session active

Seconds between keepalives (0 to turn off) 30

Low-level TCP connection options

☑ Disable Nagle's algorithm (TCP_NODELAY option)

☐ Enable TCP keepalives (SO_KEEPALIVE option)

Internet protocol version

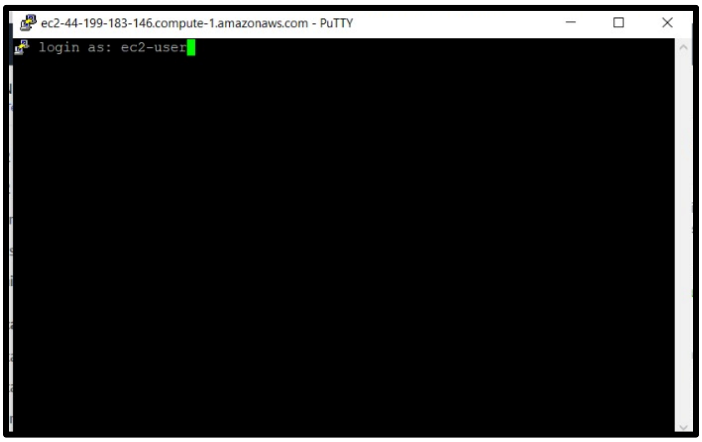
● Auto ○ IPv4 ○ IPv6

Logical name of remote host

Logical name of remote host (e.g. for SSH key lookup):

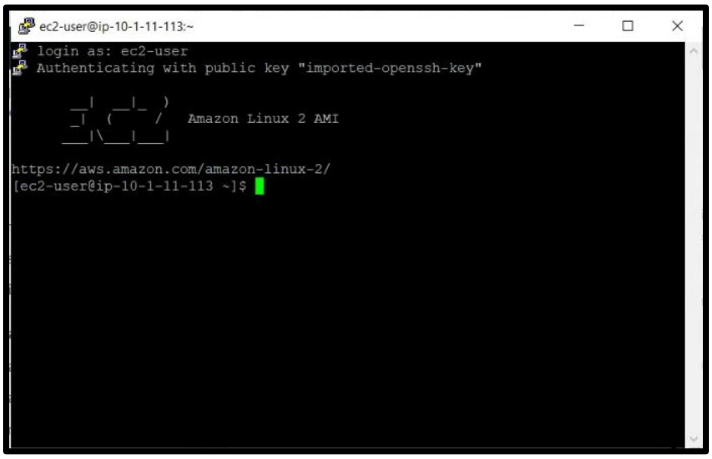
Now copy the ***public DNS or IPv4 address*** of our instance name LAB.

Configure the putty using this public DNS and also authenticate with the key which we have downloaded and start the session.



ec2-44-199-183-146.compute-1.amazonaws.com - PuTTY

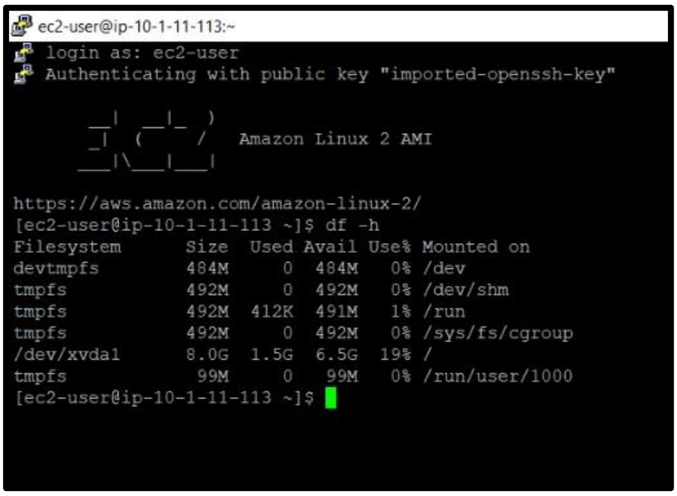
```
login as: ec2-user
```



ec2-user@ip-10-1-11-113:~

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-1-11-113 ~]$
```

Now let’s view the storage available on your instance by write the command ***df -h***

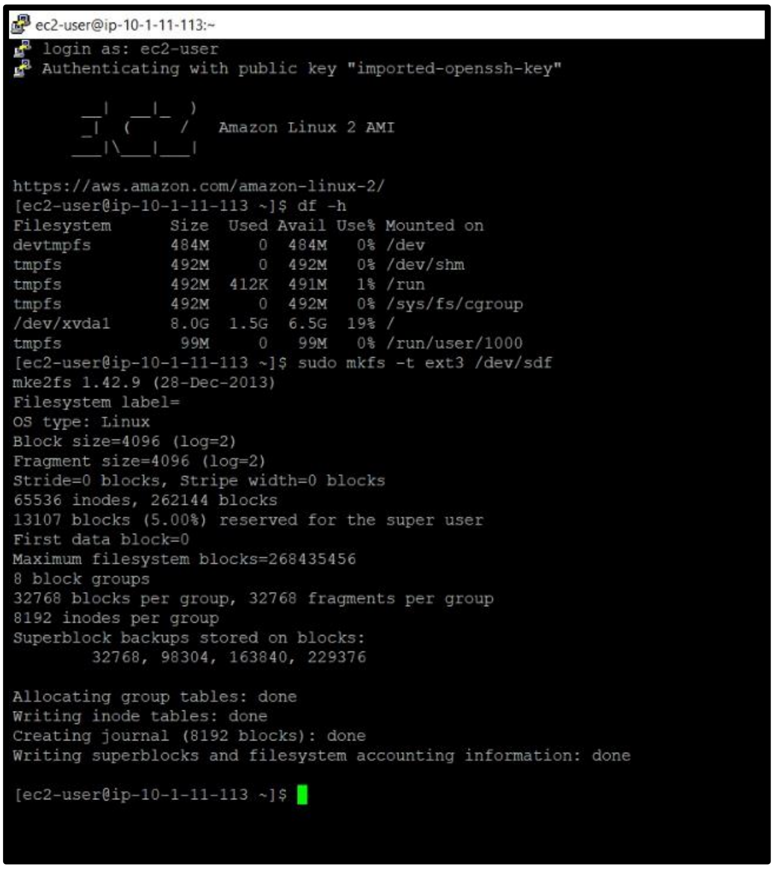


ec2-user@ip-10-1-11-113:~

```
login as: ec2-user
Authenticating with public key "imported-openssh-key"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-1-11-113 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        484M     0  484M   0% /dev
tmpfs           492M     0  492M   0% /dev/shm
tmpfs           492M  412K  491M   1% /run
tmpfs           492M     0  492M   0% /sys/fs/cgroup
/dev/xvda1      8.0G  1.5G  6.5G  19% /
tmpfs            99M     0   99M   0% /run/user/1000
[ec2-user@ip-10-1-11-113 ~]$
```

Our new volume is not yet shown.

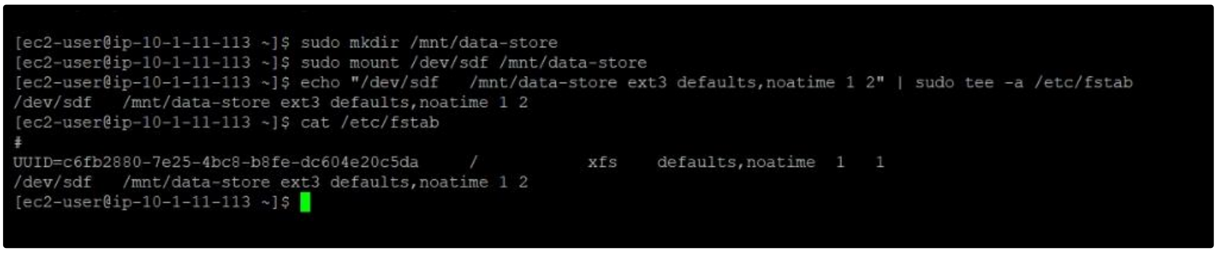
First of all, we need to create a file system. For example, let’s create a file system named ***ext3***



```
ec2-user@ip-10-1-11-113:~
login as: ec2-user
Authenticating with public key "imported-openssh-key"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-1-11-113 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        484M     0  484M   0% /dev
tmpfs           492M     0  492M   0% /dev/shm
tmpfs           492M  412K  491M   1% /run
tmpfs           492M     0  492M   0% /sys/fs/cgroup
/dev/xvda1      8.0G  1.5G  6.5G  19% /
tmpfs            99M     0   99M   0% /run/user/1000
[ec2-user@ip-10-1-11-113 ~]$ sudo mkfs -t ext3 /dev/sdf
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
65536 inodes, 262144 blocks
13107 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=268435456
8 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
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-10-1-11-113 ~]$
```

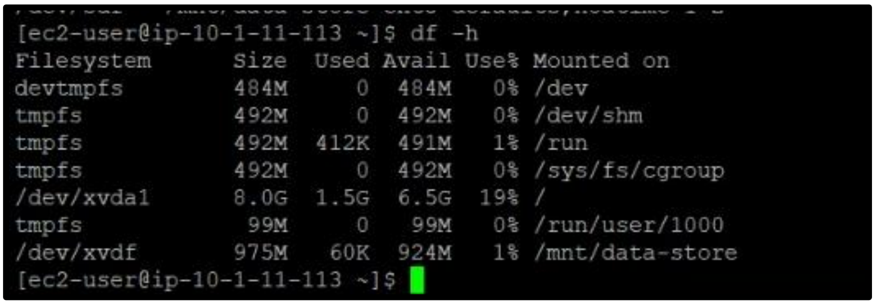
Now we will create a directory to mount our volume. Let’s say we have created a directory ***/mnt/data-store.***

Now we will mount our ***/dev/sdf*** volume to this directory ***/mnt/data-store***



```
[ec2-user@ip-10-1-11-113 ~]$ sudo mkdir /mnt/data-store
[ec2-user@ip-10-1-11-113 ~]$ sudo mount /dev/sdf /mnt/data-store
[ec2-user@ip-10-1-11-113 ~]$ echo "/dev/sdf   /mnt/data-store ext3 defaults,noatime 1 2" | sudo tee -a /etc/fstab
/dev/sdf   /mnt/data-store ext3 defaults,noatime 1 2
[ec2-user@ip-10-1-11-113 ~]$ cat /etc/fstab
#
UUID=c6fb2880-7e25-4bc8-b8fe-dc604e20c5da     /           xfs    defaults,noatime  1   1
/dev/sdf   /mnt/data-store ext3 defaults,noatime 1 2
[ec2-user@ip-10-1-11-113 ~]$
```

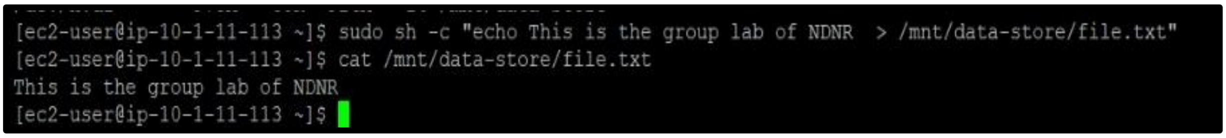
Now again let’s see the storage of our EC2 instance



```
[ec2-user@ip-10-1-11-113 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        484M     0  484M   0% /dev
tmpfs           492M     0  492M   0% /dev/shm
tmpfs           492M  412K  491M   1% /run
tmpfs           492M     0  492M   0% /sys/fs/cgroup
/dev/xvda1      8.0G  1.5G  6.5G  19% /
tmpfs            99M     0   99M   0% /run/user/1000
/dev/xvdf       975M   60K  924M   1% /mnt/data-store
[ec2-user@ip-10-1-11-113 ~]$
```

As we can see an additional line is present ***/dev/xvdf*** , which our volume which we just mount on our EC2 instance.

Let’s create a file named as ***file.txt*** and write some content inside that file.

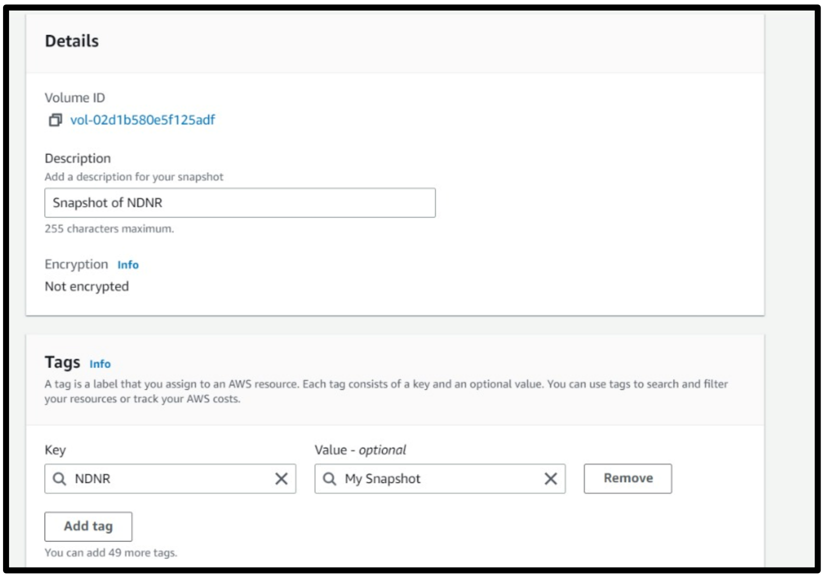


```
[ec2-user@ip-10-1-11-113 ~]$ sudo sh -c "echo This is the group lab of NDNR  > /mnt/data-store/file.txt"
[ec2-user@ip-10-1-11-113 ~]$ cat /mnt/data-store/file.txt
This is the group lab of NDNR
[ec2-user@ip-10-1-11-113 ~]$
```

Next, we will create the snapshot of our volume.

To create snapshot, we have to go to our volume and in action we will find option of creating snapshot of the volume, on click we will find below dialogue box.

Here, we can add the description of our snapshot. We also have to add a tag which is key value pair.



Details

Volume ID
vol-02d1b580e5f125adf

Description
Add a description for your snapshot
Snapshot of NDNR
255 characters maximum.

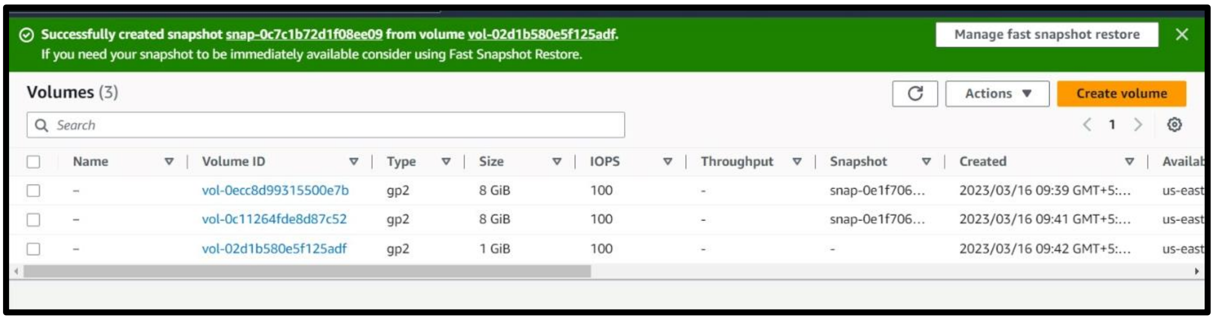
Encryption Info
Not encrypted

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.

Key: NDNR
Value - *optional*: My Snapshot

You can add 49 more tags.

After creating the snapshot, we will get a popup message of successfully creation of snapshot of our volume.

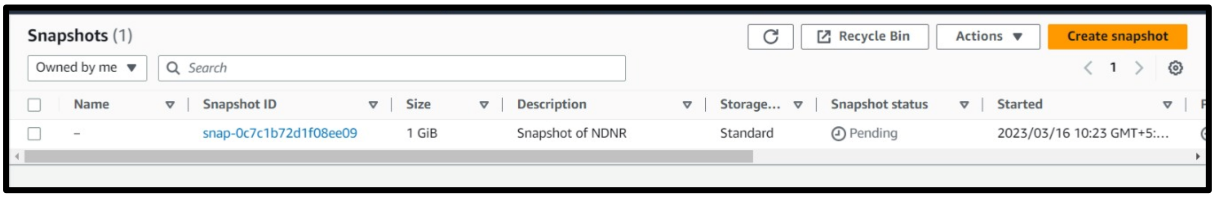


Successfully created snapshot snap-0c7c1b72d1f08ee09 from volume vol-02d1b580e5f125adf.
If you need your snapshot to be immediately available consider using Fast Snapshot Restore.

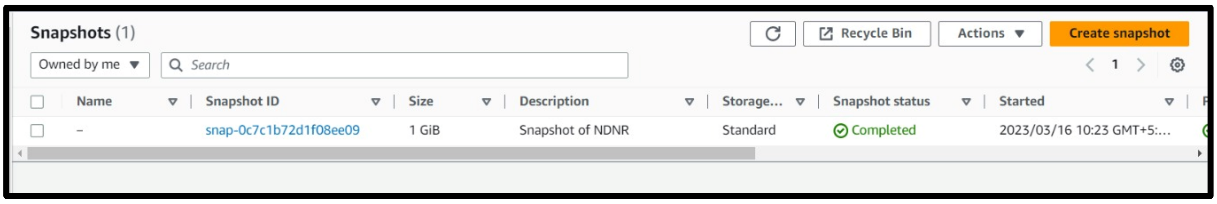
□	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot	Created	Availab
□	–	vol-0ecc8d99315500e7b	gp2	8 GiB	100	-	snap-0e1f706...	2023/03/16 09:39 GMT+5:...	us-east
□	–	vol-0c11264fde8d87c52	gp2	8 GiB	100	-	snap-0e1f706...	2023/03/16 09:41 GMT+5:...	us-east
□	–	vol-02d1b580e5f125adf	gp2	1 GiB	100	-	-	2023/03/16 09:42 GMT+5:...	us-east

Now we will go to Snapshots which will be available in the left navigation plane of the EC2 dashboard.

Inside we will find a snapshot will be existing, which the one we created just now. First, it will be in ***pending stat***e and after few minutes the creation of snapshot will be completed and it will change to ***completed state***

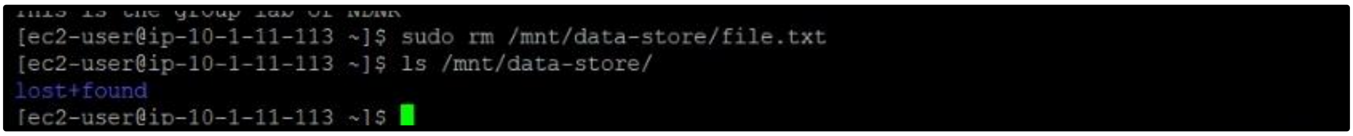


□	Name	Snapshot ID	Size	Description	Storage...	Snapshot status	Started
□	–	snap-0c7c1b72d1f08ee09	1 GiB	Snapshot of NDNR	Standard	Pending	2023/03/16 10:23 GMT+5:...



□	Name	Snapshot ID	Size	Description	Storage...	Snapshot status	Started
□	–	snap-0c7c1b72d1f08ee09	1 GiB	Snapshot of NDNR	Standard	✓ Completed	2023/03/16 10:23 GMT+5:...

Now, we will remove our file.txt from our volume using ***rm*** command.

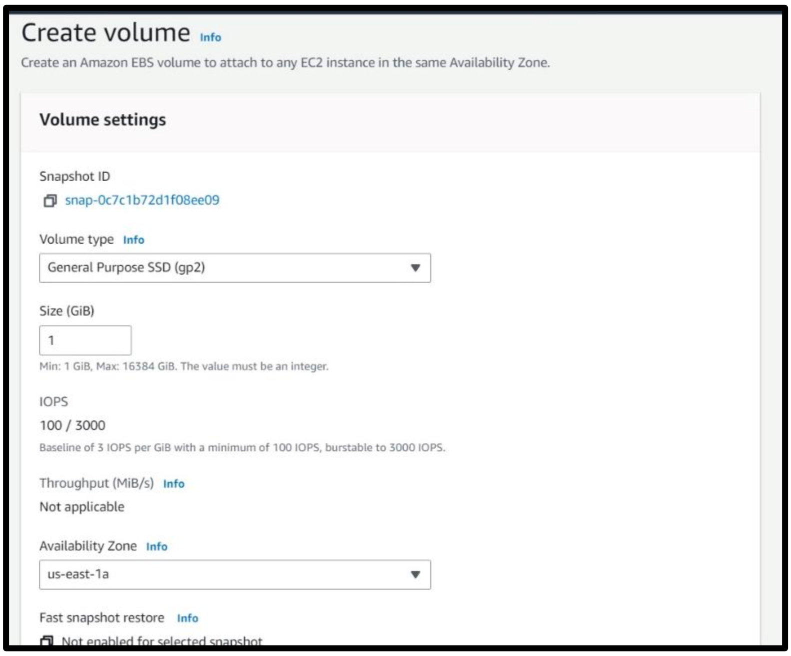


```
This is the group lab of NDNR
[ec2-user@ip-10-1-11-113 ~]$ sudo rm /mnt/data-store/file.txt
[ec2-user@ip-10-1-11-113 ~]$ ls /mnt/data-store/
lost+found
[ec2-user@ip-10-1-11-113 ~]$ 
```

Now if we want to retrieve our ***file.txt***, is it possible? how?

- ⇨ Yes, it is possible
- ⇨ We can retrieve our file.txt from our snapshot because the snapshot will contain all the files of the volume.
- ⇨ How to do that
    - ○ First, we have to create a volume from the snapshot

- Then we have to attach this volume to our EC2 instance
- After attaching, we will mount the volume to our EC2 instance and then we will be available to access our ***file.txt***



Create volume

Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.

Volume settings

Snapshot ID
snap-0c7c1b72d1f08ee09

Volume type
General Purpose SSD (gp2)

Size (GiB)
1
Min: 1 GiB, Max: 16384 GiB. The value must be an integer.

IOPS
100 / 3000
Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS.

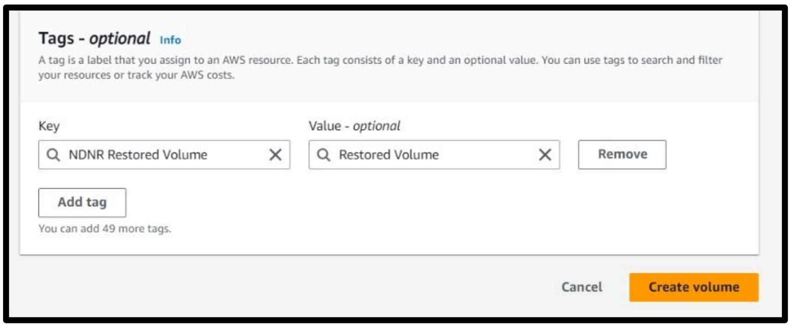
Throughput (MiB/s)
Not applicable

Availability Zone
us-east-1a

Fast snapshot restore
Not enabled for selected snapshot

At the time of creating volume from snapshot we can change the size of our volume disc and also the type.

We also have to provide tag which a key value pair.

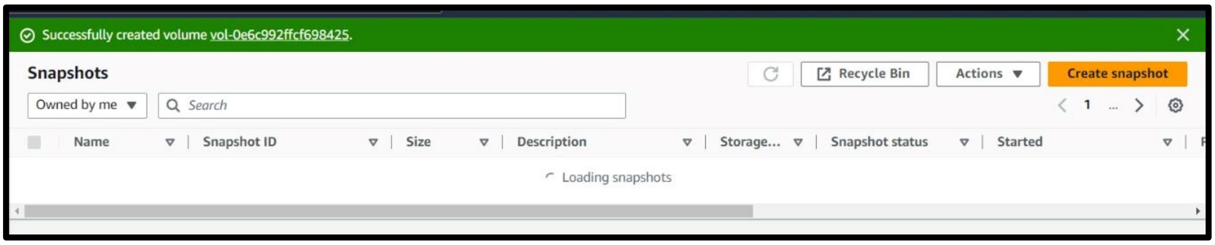


Tags - *optional*

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.

Key: NDNR Restored Volume
Value - *optional*: Restored Volume

You can add 49 more tags.



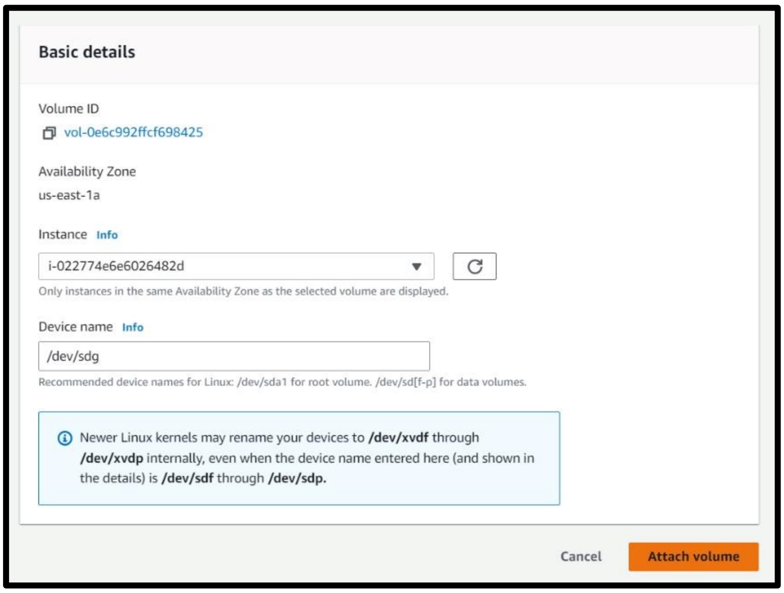
Successfully created volume vol-0e6c992ffcf698425.

Snapshots

Name	Snapshot ID	Size	Description	Storage...	Snapshot status	Started

Loading snapshots

Now we will attach this volume to our EC2 instance named LAB.



Basic details

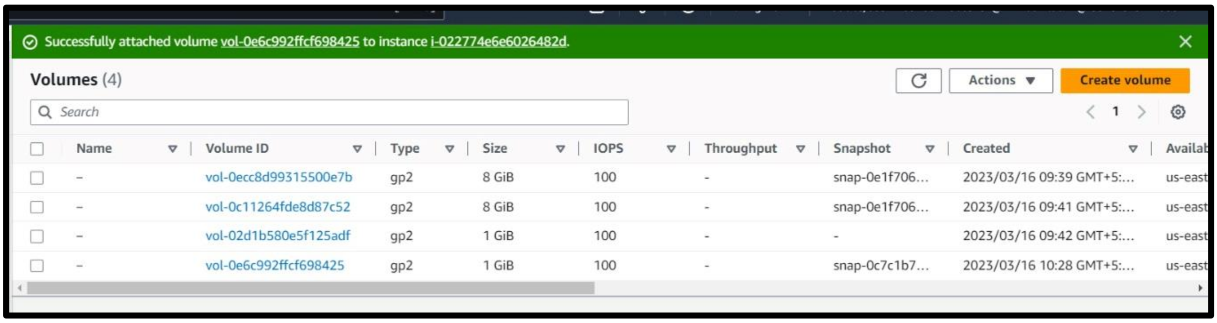
Volume ID
vol-0e6c992ffcf698425

Availability Zone
us-east-1a

Instance Info
i-022774e6e6026482d
Only instances in the same Availability Zone as the selected volume are displayed.

Device name Info
/dev/sdg
Recommended device names for Linux: /dev/sda1 for root volume. /dev/sd[f-p] for data volumes.

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.**

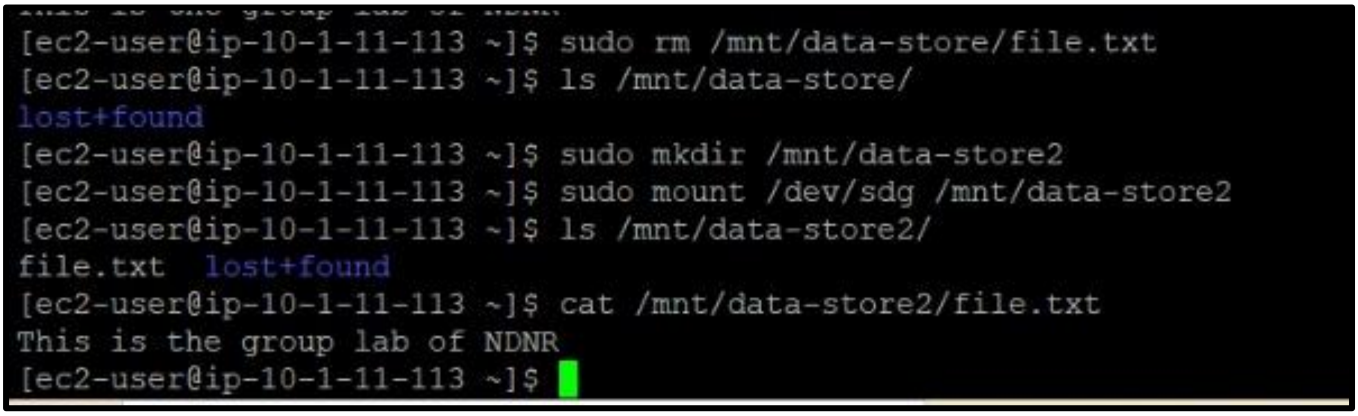


Successfully attached volume vol-0e6c992ffcf698425 to instance i-022774e6e6026482d.

Volumes (4)

□	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot	Created	Availat
□	–	vol-0ecc8d99315500e7b	gp2	8 GiB	100	-	snap-0e1f706...	2023/03/16 09:39 GMT+5:...	us-east
□	–	vol-0c11264fde8d87c52	gp2	8 GiB	100	-	snap-0e1f706...	2023/03/16 09:41 GMT+5:...	us-east
□	–	vol-02d1b580e5f125adf	gp2	1 GiB	100	-	-	2023/03/16 09:42 GMT+5:...	us-east
□	–	vol-0e6c992ffcf698425	gp2	1 GiB	100	-	snap-0c7c1b7...	2023/03/16 10:28 GMT+5:...	us-east

Now we will mount this volume on our EC2 instance in a directory ***/mnt/data-store2***



```
This is the group lab of NDNR
[ec2-user@ip-10-1-11-113 ~]$ sudo rm /mnt/data-store/file.txt
[ec2-user@ip-10-1-11-113 ~]$ ls /mnt/data-store/
lost+found
[ec2-user@ip-10-1-11-113 ~]$ sudo mkdir /mnt/data-store2
[ec2-user@ip-10-1-11-113 ~]$ sudo mount /dev/sdg /mnt/data-store2
[ec2-user@ip-10-1-11-113 ~]$ ls /mnt/data-store2/
file.txt  lost+found
[ec2-user@ip-10-1-11-113 ~]$ cat /mnt/data-store2/file.txt
This is the group lab of NDNR
[ec2-user@ip-10-1-11-113 ~]$ 
```

As we can see that ***file.txt*** is not available inside the ***data-store*** but is available inside the ***data-store2***. This is because we have mounted the volume which is the snapshot of the previous volume.