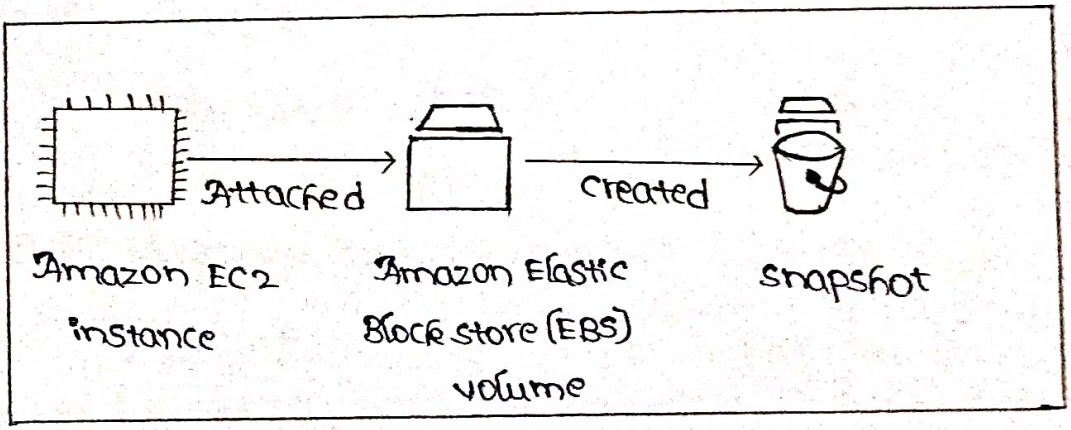
Lab-4.

Aim:- The working with amazon Elastic Block store.[Amazon EBS].

Description:-

Amazon Elastic Block store [Amazon EBS] offers persistent storage for Amazon EBS. instances. Amazon EBS volumes are network-attached and persist independently from the life of an instance. Amazon EBS volumes are highly available, highly reliable volumes that can be leveraged as an Amazon EC2 instances boot partition (or) attached to a running Amazon EC2 instance as a standard block device. when used as a boot partition, Amazon EC2 instances can be stopped and subsequently restarted, enabling you to pay only for the storage resources used while maintaining your instance's state.

Architecture:-



Steps followed to working with EBS console management:

* Click Start lab to launch a lab.
* Wait until you see the message "lab status: ready".
* Choose AWS.

Task-1: Create a New EBS volume:

* In AWS management console, on the services menu, click EC2.
* In navigation pane, choose Instances.
* The Availability zone of the instance. It will look similar to us-east-1a.
* In navigation pane, choose volumes.
* Choose create volume then configure:
  - Volume Type: General purpose SSD (gp2)
  - Size (GiB): 1. NOTE:- you may be restricted from creating large volumes.
  - Availability zone: select the same availability zone as our EC2 instance.
  - Choose Add Tag
  - In Tag Editor, enter:
    - key: Name
    - value: my volume
* Choose create volume.

Task-2: Attach the volume to an Instance:

* Select my volume.

* In the Action menu, choose Attach volume.
* Choose the Instance field.
* Choose Attach volume

Task-3: Connect to our Amazon EC2 instance:.

* Read through the three bullet points in this step before we start to complete the actions:
  - Choose the Details and then choose the Show. A credentials window will open.
  - Choose the Download ppk button and save the labuser.ppk file
  - Then exit the details panel
* Download needed software.
  - We will use PUTTY to SSH to Amazon EC2 instances.
* Open putty.exe.
* Configure PUTTY to not timeout:
  - Choose Connection
  - Set seconds between keepalives to 30
* Configure our PUTTY session:
  - Choose Session.
  - Host name : paste the public DNS or IPV4 address of Lab instance
  - Back in PUTTY, in the connection list, expand ⊞ SSH
  - Choose Auth and expand ⊞ credentials.
  - Under private key file for authentication: Choose Browse

- Browse to labsuser.ppk file, choose open.
- choose open again.

* To trust and connect to host, choose Accept
* when prompted login as, enter: ec2-user
  - This will connect we to the EC2 instance.

macos and linux users.

* Read through the three bullet points in this step before we start to complete the actions:
  - choose the Details and then choose the Show. A credentials window will open.
  - choose the Download button and save the labsuser.pem file.
  - Then exit the details panel.
* open a terminal window, and change directory cd to directory where the labsuser.pem file was downloaded.
  - for example, run this command.

```
cd ~/Downloads.
```

* change the permissions on key to be read only by running this command.

```
chmod 400 labsuser.pem.
```

* Return to aws management console, and in EC2 services, choose Instances.
* In details tab, copy the public IPV4 address value.

*Return to terminal window and run this command.

ssh -i labsuser.pem ec2-user@<public-ip>

*Type yes.

Task 4: Create and configure our file system.

*view the storage available on our instance:

df -h

*Create an ext3 file system on new volume:

sudo mkfs -t ext3 /dev/sdf

*create a directory for mounting the new storage volume:

sudo mkdir /mnt/data-store.

*mount the new volume:

sudo mount /dev/sdf /mnt/data-store.

*view the configuration file to see the setting on last line:

cat /etc/fstab

*view the available storage again:

df -h.

*create a file and add some text to it

sudo sh -c "echo some text has been written > /mnt/data-store/file.txt".

*verify that text has been written to our volume.

cat /mnt/data-store/file.txt.

Task 5 - create an Amazon EBS snapshot:

* In the AWS management console, choose volumes and select my volume.
* In the Actions menu, select create snapshot.
* choose Add tag then configure:
    - key : Name
    - value : my snapshot
    - choose Create snapshot
* In navigation pane, choose the Snapshots.
* In our remote SSH session, delete the file that you created on our volume.

```
sudo rm /mnt/data-store/file.txt
```

* verify that the file has been deleted.

```
ls /mnt/data-store/
```

Task-6 : Restore the Amazon EBS snapshot:

Create a volume using our snapshot.

* In the AWS management console, select my snapshot.
* In the Actions menu, select create volume from snapshot.
* For Availability zone select same availability zone.
* choose Add tag then configure:
    - key : Name
    - value : Restored volume
    - choose Create volume

Attach the Restored volume to our EC2 instance.

* In navigation pane, choose volumes
* select Restored volume.
* In the Action menu, select Attach volume.
* choose the Instance field.
* choose Attach volume.

The volume state is now in-use.

Mount the Restored volume.

*create a directory for mounting the new storage volume.

```
sudo mkdir /mnt/data-store2.
```

* Mount the new volume.

```
sudo mount /dev/sdg /mnt/data-store2.
```

*verify that volume we mounted has the file that we created earlier.

```
ls /mnt/data-store2/
```

*choose End Lab and then click Yes to confirm that we want to end the Lab.