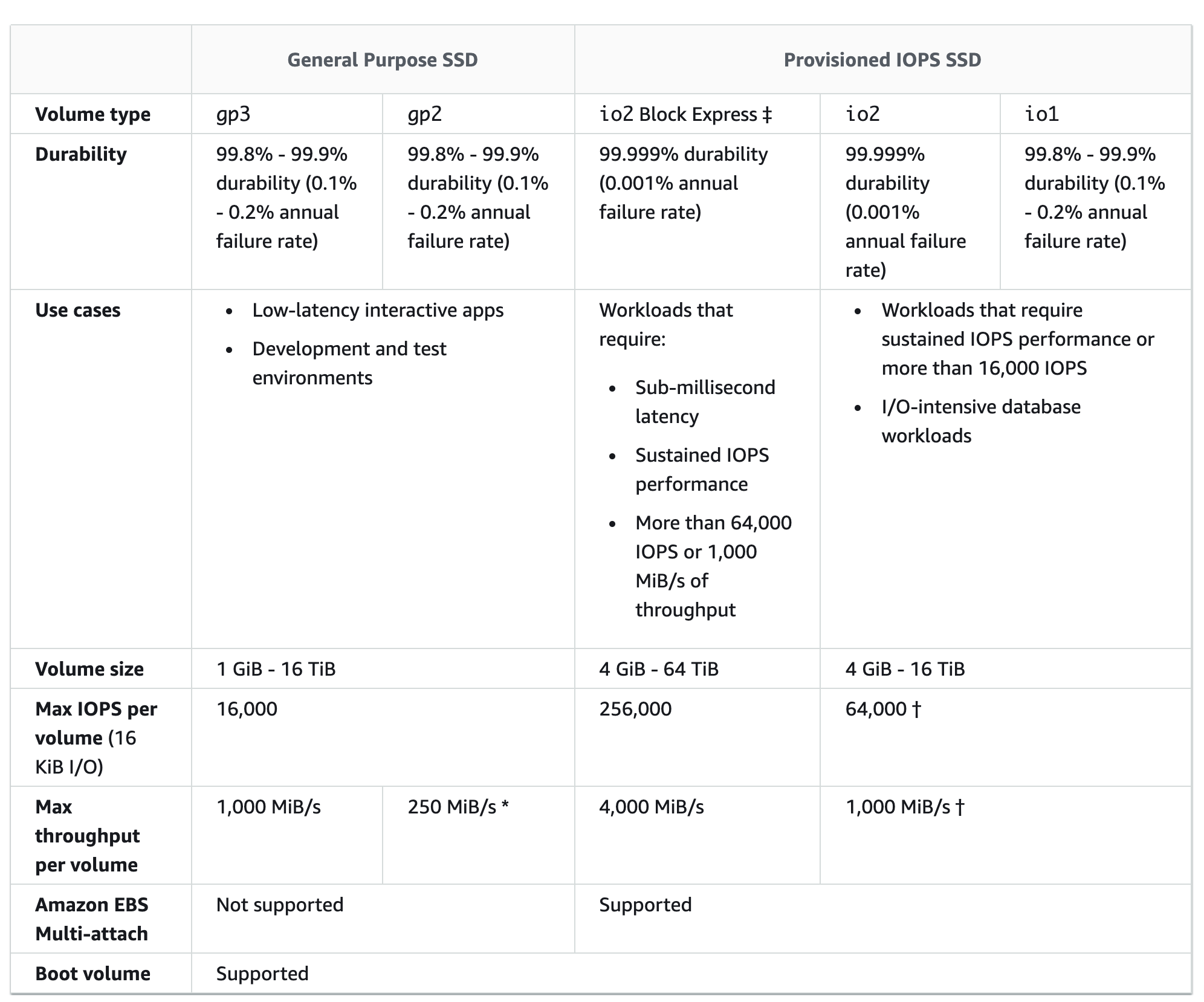
EBS Volume Types (New Generation)

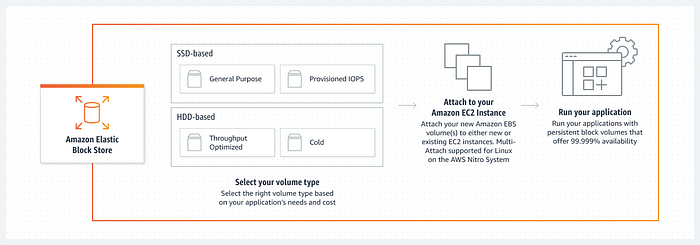
<https://jayendrapatil.com/aws-ebs-volume-types/> for ref



Amazon Elastic Block Store (EBS) provides scalable and high-performance block-level storage for your EC2 instances.

## How it works

Amazon Elastic Block Store (Amazon EBS) is an easy-to-use, scalable, high-performance block-storage service designed for Amazon Elastic Compute Cloud (Amazon EC2).



In the realm of AWS capabilities, you have the flexibility to generate fresh EBS volumes, ready to be linked with instances for additional storage. Yet, the road to transforming an EBS volume into functional storage within an instance involves the crucial step of mounting it to a designated directory.

Source: <https://aws.amazon.com/ebs/>

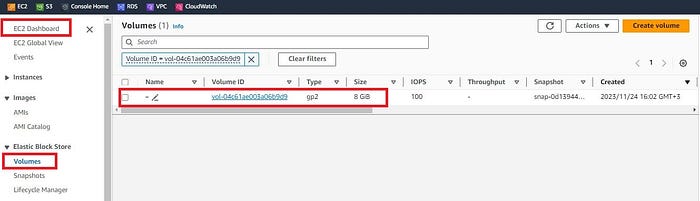
In this guide, we’ll walk through the process of attaching and mounting an EBS volume to an EC2 Linux instance running Ubuntu.

## Prerequisites:

* An active AWS account.
* An EC2 instance running Ubuntu.

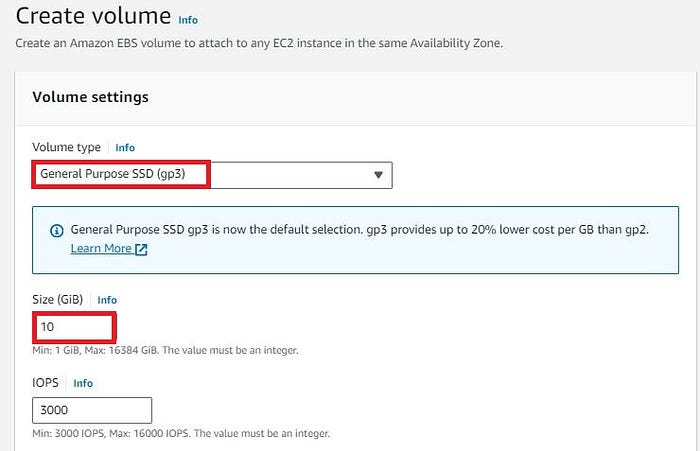
## Step 1: Create an EBS Volume:

1. Open the AWS Management Console and navigate to the EC2 dashboard.
2. In the left navigation pane, choose “Volumes” under the “Elastic Block Store” section.
3. Take note of the existing volumes. In this example, we have an 8GB volume.

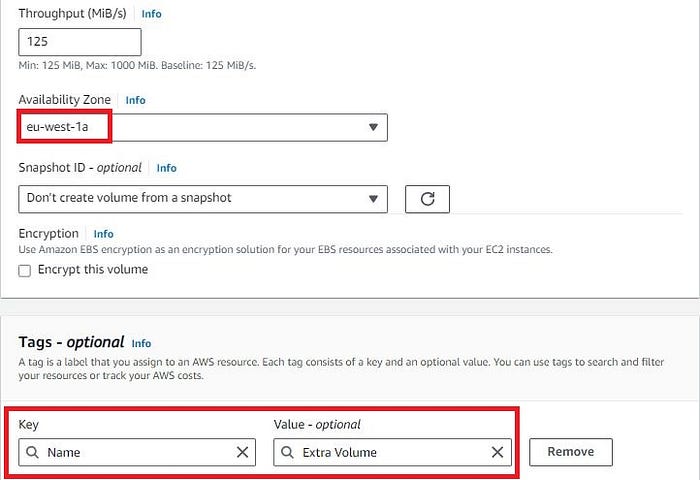


4. Click “Create Volume” to add a new EBS volume.

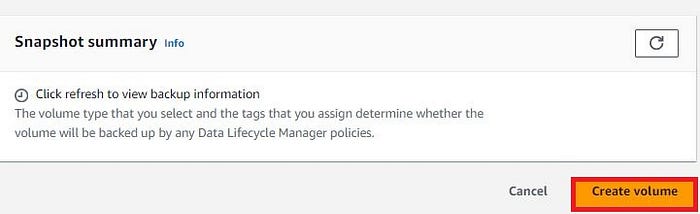
* Volume Type: General Purpose SSD (gp2)
* Size (GiB): Enter 10 for a 10GB volume.
* Availability Zone: Choose the same availability zone as your EC2 instance.



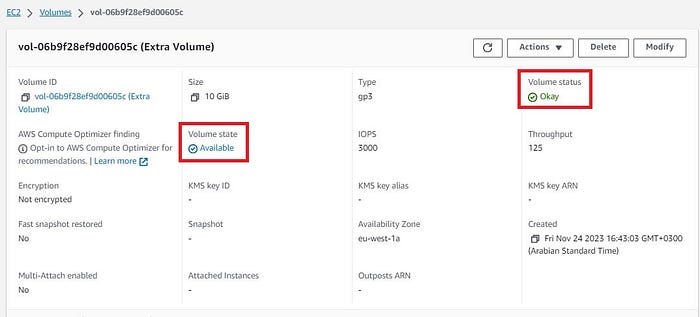
We can a tag which is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.



Finally, Click “Create” to create the EBS volume.

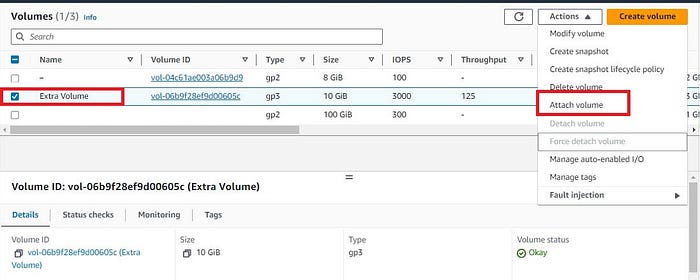


Now if we can check the status of the newly created volume will be available.

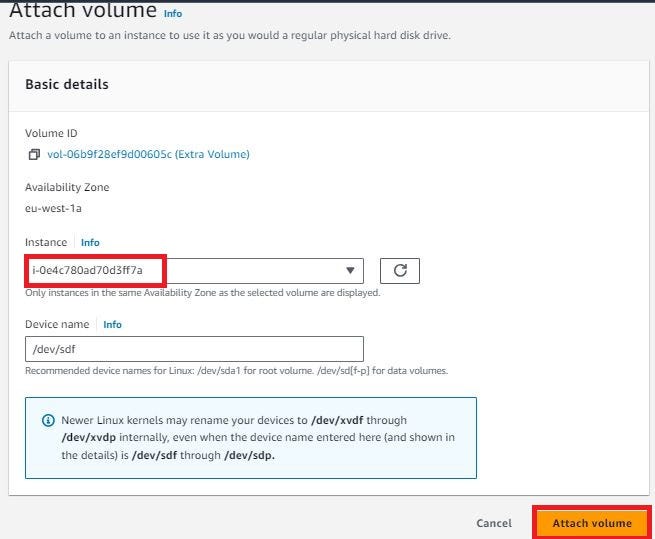


## Step 2: Attach EBS Volume to EC2 Instance:

1. In the Volumes dashboard, select the newly created volume.
2. Click “Actions” > “Attach Volume” and choose your EC2 instance.
3. Confirm the attachment by clicking “Attach.”



Make sure to select the instance on which you have to attach the additional volume.



To verify that the volume has been successfully attached to the EC2 Instance you can go to the Volume section, select the newly created volume, and check the attached instance section as shown below:

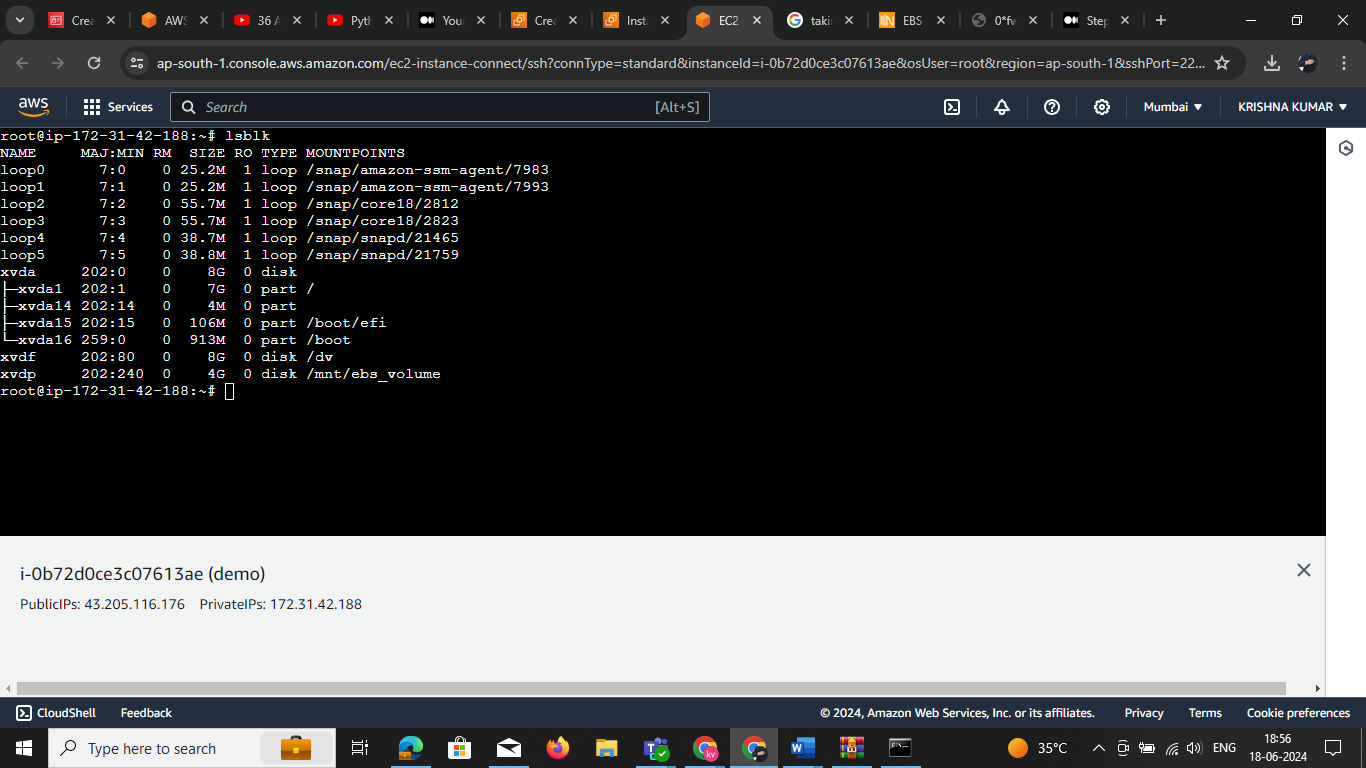


## ****Step 3: Connect to your EC2 Instance:****

1. Use SSH to connect to your EC2 instance.
2. Use the following command to list the available block devices:

lsblk

Identify your attached volume (e.g., /dev/xvdf).



## Step 4: Create a Filesystem on the EBS Volume:

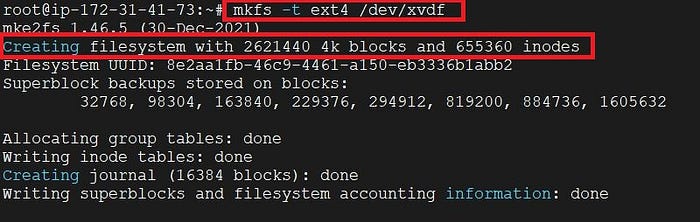
1. Let’s first create a directory to be used as the mount point:

mkdir -p /mnt/ebs\_volume

mkdir dv

2. Now use the below command to create a filesystem on the attached volume:

mkfs -t ext4 /dev/xvdf



Now if we verify if the new file system exists use the below command and you should see the output:

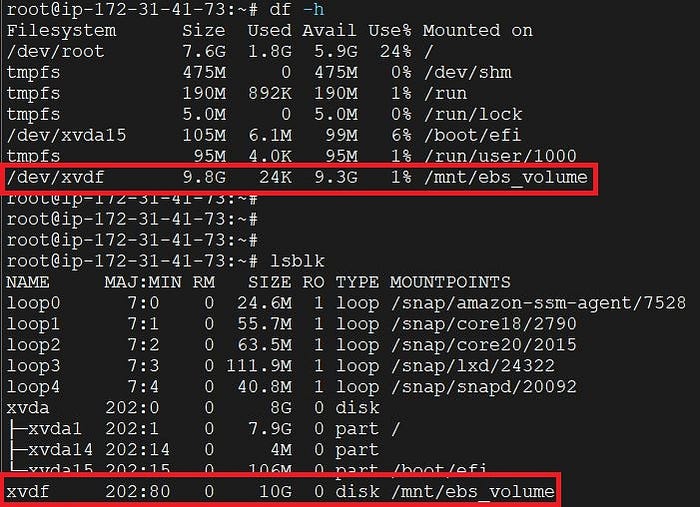
file -s /dev/xvdf  
/dev/xvdf: Linux rev 1.0 ext4 filesystem data, UUID=8e2aa1fb-46c9-4461-a150-eb3336b1abb2 (extents) (64bit) (large files) (huge files)

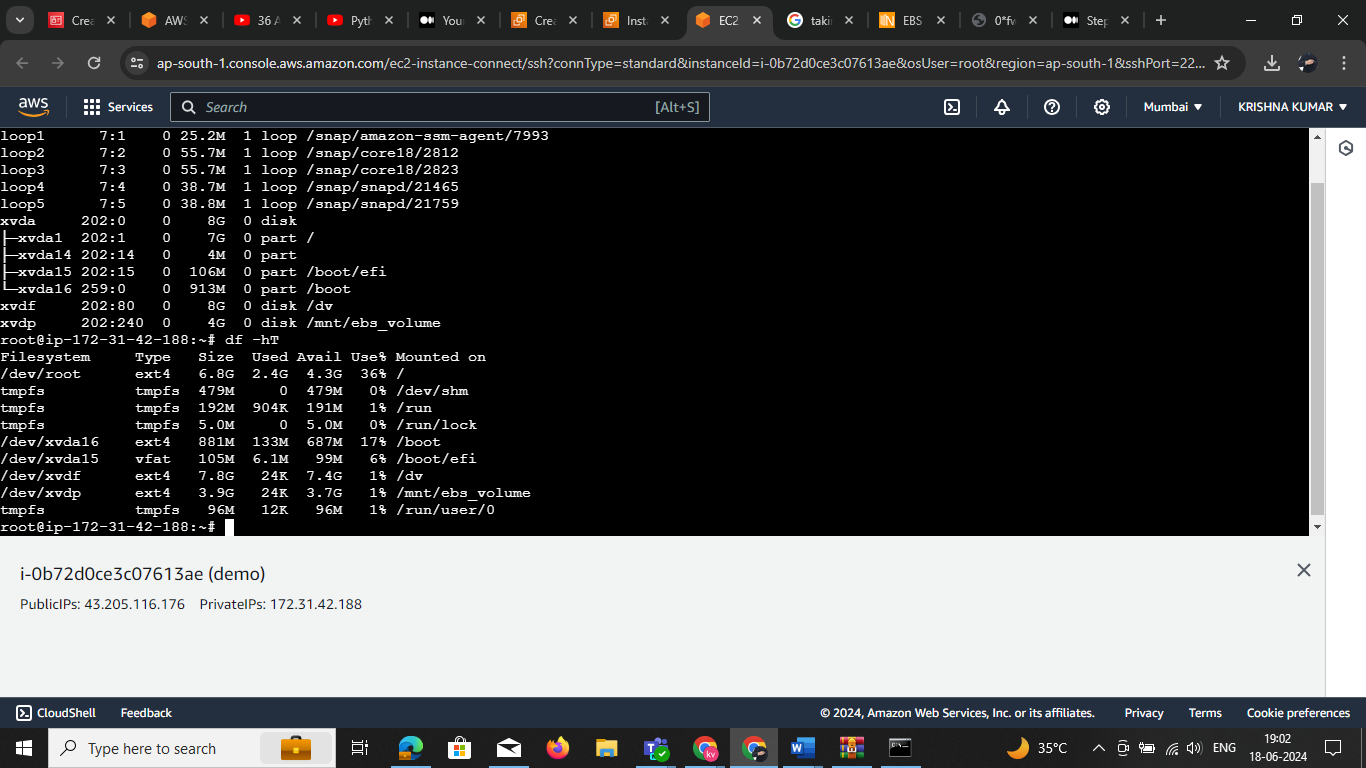
## Step 5: Mount the EBS Volume:

1. Mount the EBS volume to the specified mount point:

mount /dev/xvdf /mnt/ebs\_volume

Now you should see the new EBS volume mounted to the directory you created in the earlier steps:





## Step 6: Configure Automatic Mount on Boot:

1. Open the /etc/fstab file in a text editor:

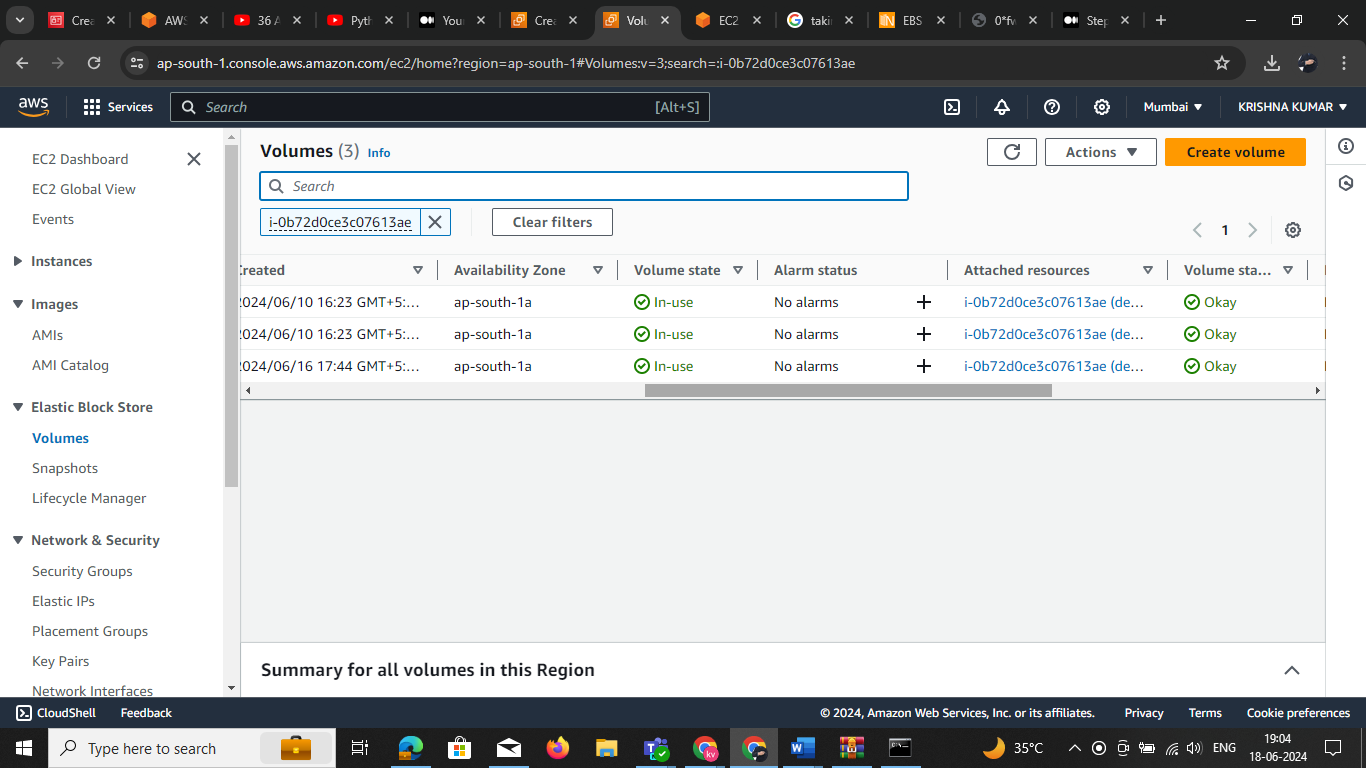
nano /etc/fstab

2. Add the following line to the end of the file:

/dev/xvdf /mnt/ebs\_volume ext4 defaults,nofail 0 2

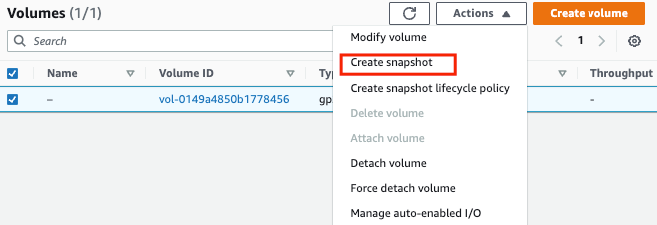
Save and exit the editor.

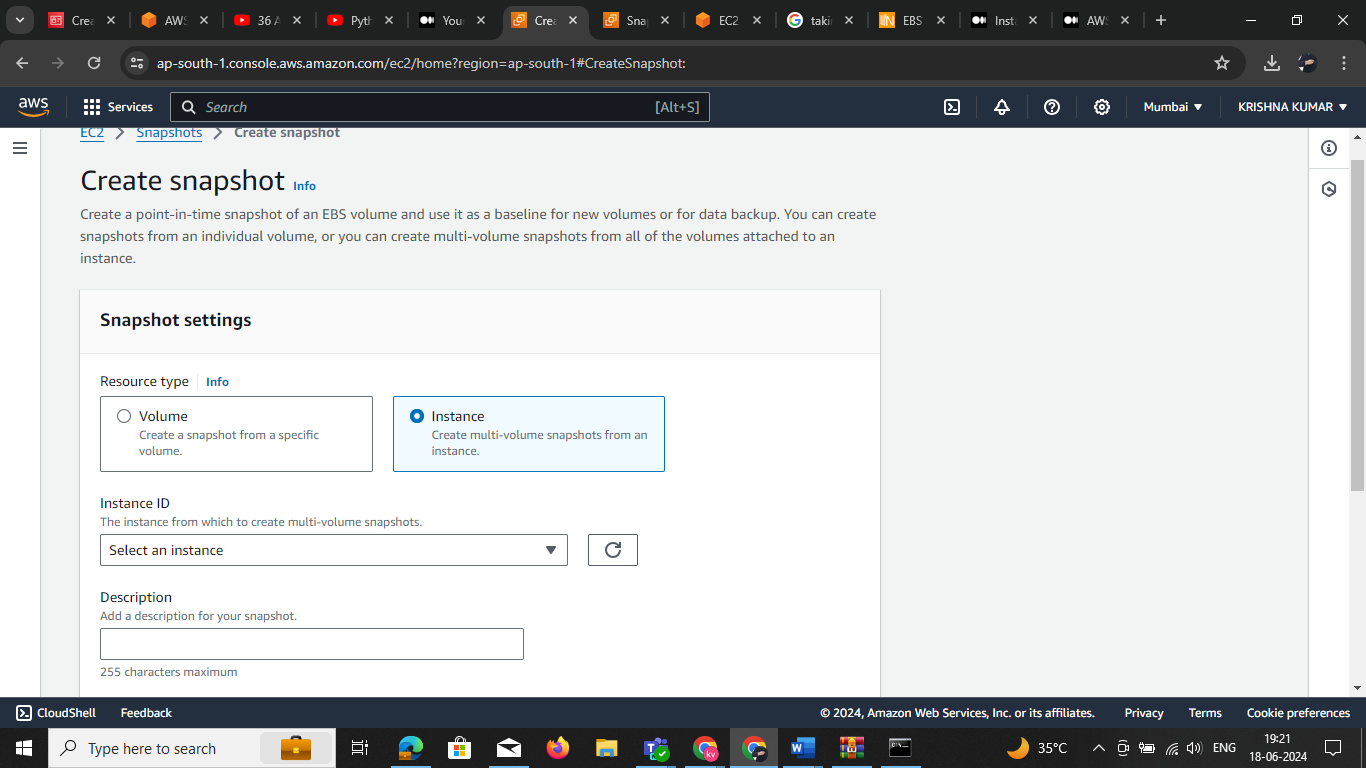
You can also verify on the AWS console the new EBS volume has been successfully added and mounted:



**Creating a snapshot from a single EC2 Volume**

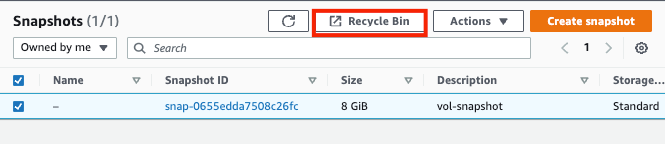
Clicking the Create snapshot is pretty easy and you should select the volumes on the left panel and one volume should be created for your EC2 image. From this EBS volume, we can create the snapshot as shown below:



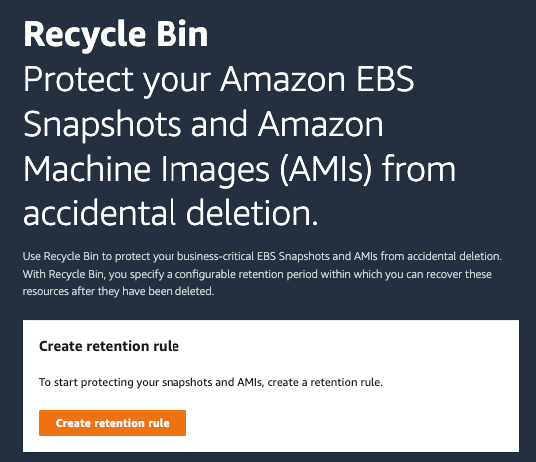


**Note : taking snapshot we need to stop ec2 instances**

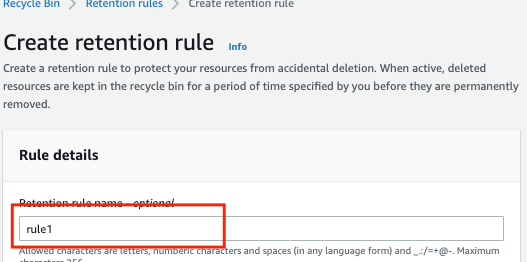
**Initiating Recycle Bin**



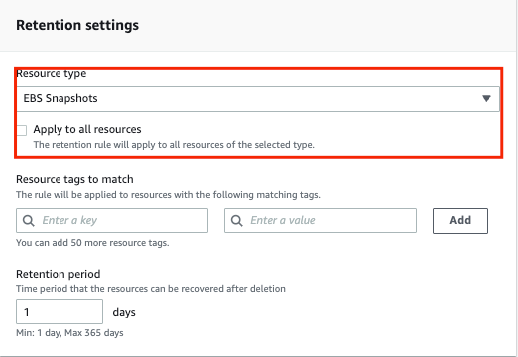
Our intention is to recover our EBS, when it is deleted. In case we click the Recycle Bin button on the selected snapshot, it will navigate us to the following page, in which we will press *Create retention rule* button.



First, the retention rule name and descriptions are required.



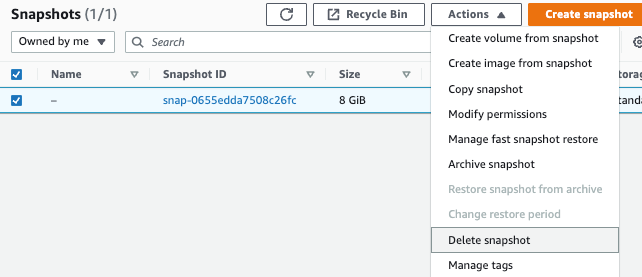
Retention settings enable us to define the resource type, whether it is an image or EBS snapshot, resource tags to match, whether we target a specific group of resource types, and finally retention period, which informs us about the recovery time after the deletion.



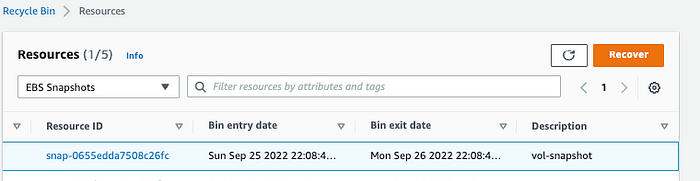
In our example, we clicked *Apply to all resources* to simplify the process.

**Recovering Deleted Snapshot**

Below we delete the snapshot, and it will not be listed any more under Snapshots list. However, we noticed that we did a mistake, therefore we navigate us to the Recycle Bin.



Under Resources in Recycle Bin, we can recover from the deleted EBS volume as illustrated below:



We can even create an EBS volume from this snapshot in any AZ region depicted in the following figure.

