

Amazon EBS GP2 to GP3 Migration

Amazon EBS GP3 volumes are the latest generation of general-purpose SSD-based storage. With GP3, customers can scale IOPS (input/output operations per second) and throughput independently of capacity. This ensures customers only pay for the storage they actually need.

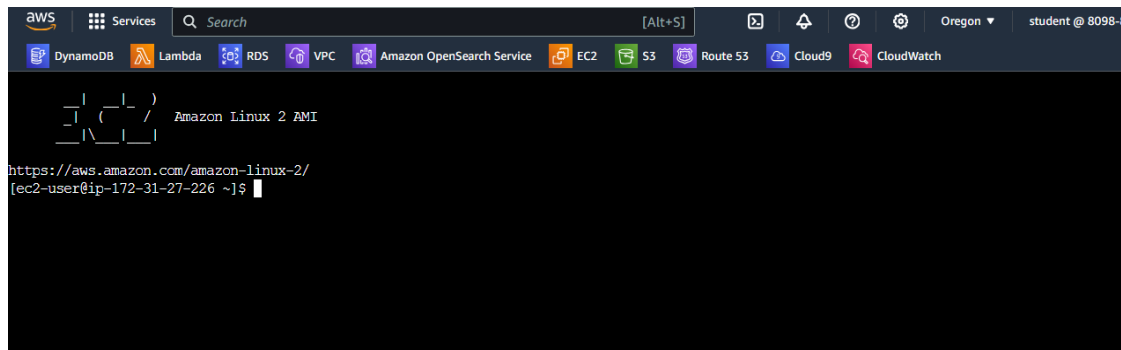
I'll first start by provisioning and attaching a new EBS GP2 volume to a provided running EC2 instance. You'll then be walked through the process of formatting and mounting the EBS GP2 volume into the filesystem. Next, I'll be instructed to migrate the EBS GP2 volume to GP3. Finally, I'll be shown how to benchmark the EBS GP3 volume, and in doing so I'll get to see first hand how GP3 provides improved performance

Steps:

- 1- Logging In to the Amazon Web Services Console
- 2- Connect to the Workload Instance
- 3- Create and Attach GP2 Volume
- 4- Format and Mount Volume
- 5- Migrate EBS GP2 Volume to GP3
- 6- Performance Test GP3 EBS Volume

2- Connect to the Workload Instance:

- Right-click on the instance and select the Connect option
- Confirm that you have now connected successfully



3- Create and Attach GP2 Volume:

- 1- Use the AWS CLI to create a new EBS GP2 volume. The new volume should be 2Gb in size, and placed in the same availability zone as the running EC2 instance. In the terminal run the following command

```
$ aws ec2 create-volume --size 2 --volume-type gp2 --availability-zone $LAB_INSTANCE_AZ
```



- 2- Use the AWS CLI to again examine the existing EBS volumes. Confirm that there is now a second EBS GP2 volume in an available state. In the terminal run the following command

```
$ aws ec2 describe-volumes
```

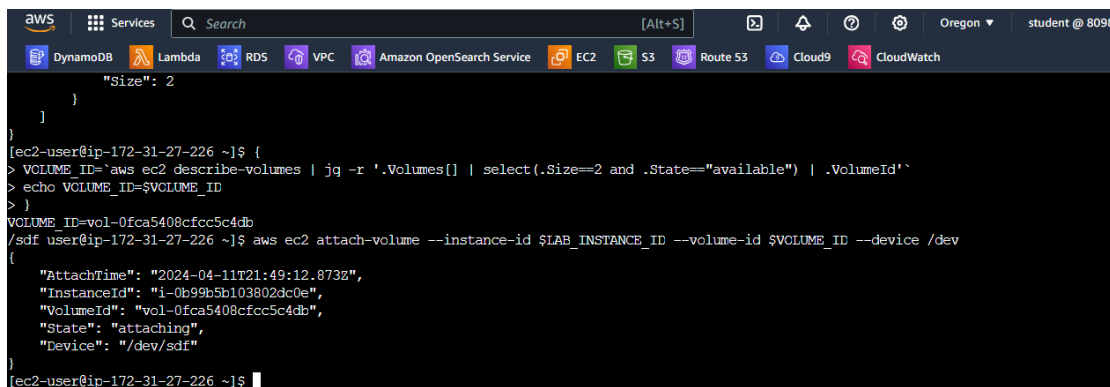
- 3- Retrieve the volume id for the newly provisioned EBS GP2 volume

```
$ {  
VOLUME_ID=`aws ec2 describe-volumes | jq -r '.Volumes[] | select(.Size==2  
and .State=="available") | .VolumeId`
```

```
echo VOLUME_ID=$VOLUME_ID
}
```

- 4- Use the AWS CLI to attach the newly provisioned EBS GP2 volume to the provided EC2 instance. In the terminal run the following command

```
aws ec2 attach-volume --instance-id $LAB_INSTANCE_ID --volume-id
$VOLUME_ID --device /dev/sdf
```



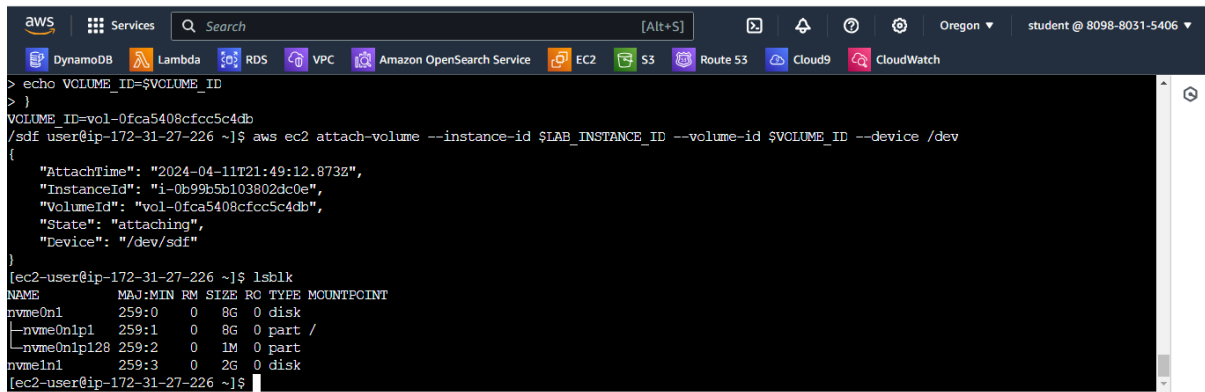
```
aws
Services Search [Alt+S] Oregon student @ 8098
DynamoDB Lambda RDS VPC Amazon OpenSearch Service EC2 S3 Route 53 Cloud9 CloudWatch

"Size": 2
}
]
}
[ec2-user@ip-172-31-27-226 ~]$ {
> VOLUME_ID=$(aws ec2 describe-volumes | jq -r '.Volumes[] | select(.Size==2 and .State=="available") | .VolumeId')
> echo VOLUME_ID=$VOLUME_ID
> }
VOLUME_ID=vol-0fca5408cfcc5c4db
/sdf user@ip-172-31-27-226 ~]$ aws ec2 attach-volume --instance-id $LAB_INSTANCE_ID --volume-id $VOLUME_ID --device /dev
{
  "AttachTime": "2024-04-11T21:49:12.873Z",
  "InstanceId": "i-0b99b5b103802dc0e",
  "VolumeId": "vol-0fca5408cfcc5c4db",
  "State": "attaching",
  "Device": "/dev/sdf"
}
[ec2-user@ip-172-31-27-226 ~]$
```

In previous steps I used the AWS CLI to provision a new EBS GP2 volume. then used the AWS CLI to attach the newly provisioned EBS GP2 volume to the provided EC2 instance. In the next lab step I will format and mount the new EBS volume into the filesystem on the provided EC2 instance, before eventually migrating the same EBS volume to GP3

4-Format and Mount Volume:

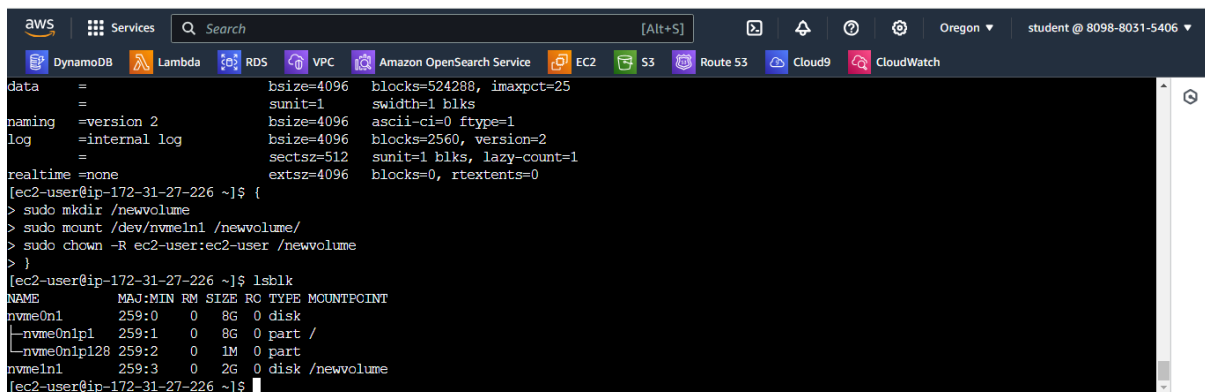
- 1- Use the lsblk command to list the block devices currently available.
- 2- Use the mkfs command to format the device using the XFS format



```
aws
Services
Search [Alt+S]
DynamoDB Lambda RDS VPC Amazon OpenSearch Service EC2 S3 Route 53 Cloud9 CloudWatch
> echo VOLUME_ID=$VOLUME_ID
>
VOLUME_ID=vol-0fca5408cfcc5c4db
/sdf user@ip-172-31-27-226 ~]$ aws ec2 attach-volume --instance-id $LAB_INSTANCE_ID --volume-id $VOLUME_ID --device /dev
{
  "AttachTime": "2024-04-11T21:49:12.873Z",
  "InstanceId": "i-0b99b5b103802dc0e",
  "VolumeId": "vol-0fca5408cfcc5c4db",
  "State": "attaching",
  "Device": "/dev/sdf"
}
[ec2-user@ip-172-31-27-226 ~]$ lsblk
NAME        MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme0n1     259:0    0  8G  0 disk
├─nvme0n1p1 259:1    0  8G  0 part /
├─nvme0n1p128 259:2    0 1M  0 part
└─nvme1n1   259:3    0  2G  0 disk
[ec2-user@ip-172-31-27-226 ~]$
```

3- Use the mkfs command to format the device using the XFS format

4- Create a new directory and mount it to the newly formatted XFS device

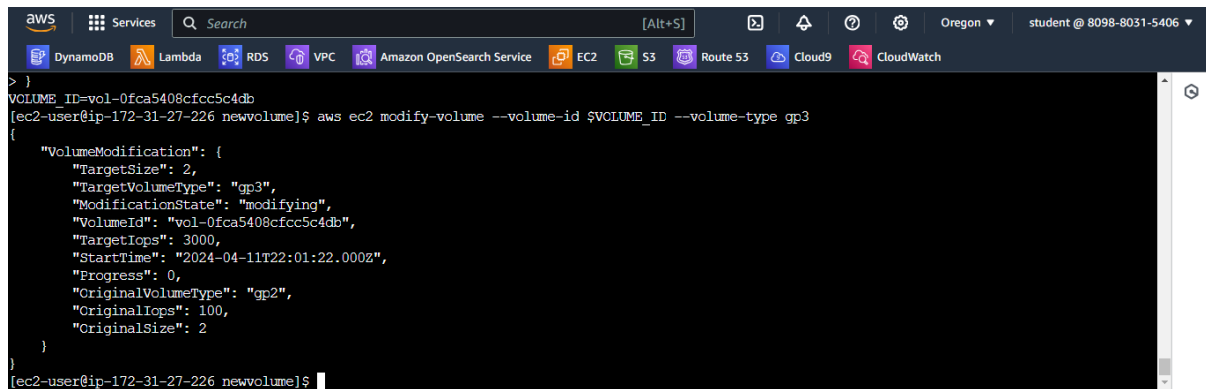


```
aws
Services
Search [Alt+S]
DynamoDB Lambda RDS VPC Amazon OpenSearch Service EC2 S3 Route 53 Cloud9 CloudWatch
data      =                               bsize=4096  blocks=524288, imaxpct=25
          =                               sunit=1    swidth=1 blks
naming    =version 2                   bsize=4096  ascii-ci=0  ftype=1
log       =internal log                bsize=4096  blocks=2560, version=2
          =                               sectsz=512   sunit=1 blks, lazy-count=1
realtime  =none                        extsz=4096  blocks=0, rtextents=0
[ec2-user@ip-172-31-27-226 ~]$ {
> sudo mkdir /newvolume
> sudo mount /dev/nvme1n1 /newvolume/
> sudo chown -R ec2-user:ec2-user /newvolume
> }
[ec2-user@ip-172-31-27-226 ~]$ lsblk
NAME        MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
nvme0n1     259:0    0  8G  0 disk
├─nvme0n1p1 259:1    0  8G  0 part /
├─nvme0n1p128 259:2    0 1M  0 part
└─nvme1n1   259:3    0  2G  0 disk /newvolume
[ec2-user@ip-172-31-27-226 ~]$
```

5- Migrate EBS GP2 Volume to GP3:

1- Use the AWS CLI to migrate the EBS GP2 volume to GP3

```
$ aws ec2 modify-volume --volume-id $VOLUME_ID --volume-type gp3
```



```
> }
VOLUME_ID=vol-0fca5408cfcc5c4db
[ec2-user@ip-172-31-27-226 newvolume]$ aws ec2 modify-volume --volume-id $VOLUME_ID --volume-type gp3
{
  "VolumeModification": {
    "TargetSize": 2,
    "TargetVolumeType": "gp3",
    "ModificationState": "modifying",
    "VolumeId": "vol-0fca5408cfcc5c4db",
    "TargetIops": 3000,
    "StartTime": "2024-04-11T22:01:22.000Z",
    "Progress": 0,
    "OriginalVolumeType": "gp2",
    "OriginalIops": 100,
    "OriginalSize": 2
  }
}
[ec2-user@ip-172-31-27-226 newvolume]$
```

- 2- Use the AWS CLI to examine the state of the EBS volumes
\$aws ec2 describe-volumes

6- Performance Test GP3 EBS Volume

1. Install the Flexible I/O tester. This will be used to test the I/O throughout of the migrated GP3 disk
\$sudo yum install -y fio
2. Perform an I/O benchmark using the fio utility. Configure fio to perform a 60 second read/write test against the /newvolume directory which is mounted to the EBS GP3 volume. In the terminal run the following command:

```
{
cd /newvolume
sudo fio --directory=. \
--ioengine=psync \
--name fio_test_file \
--direct=1 \
--rw=randwrite \
--bs=16k \
--size=100M \
```

$$\}$$

approximately 3000 IOPs

