



**Department of Computer Science & Engineering  
Premier University.**

CSE 482: Contemporary Course of Computer Science  
Laboratory.

**Lab 4: Working with Amazon Elastic Block Store (EBS)**

**Submitted By:**

<b>Name</b>	<b>Sajjad Hosen Emon</b>
<b>ID</b>	<b>0222310005101105</b>
<b>Section</b>	<b>C</b>
<b>Semester</b>	<b>7<sup>th</sup></b>
<b>Submission Date</b>	<b>10/2/2026</b>

**Remarks**

--

# Lab 4: Working with Amazon Elastic Block Store (EBS)

## Objective

The objective of this lab is to develop practical knowledge about Amazon Elastic Block Store (EBS) by creating a storage volume, attaching it to an Amazon EC2 instance, configuring a file system, and performing data backup using snapshots. This lab also demonstrates how data can be restored from a snapshot to ensure storage reliability and persistence in a cloud computing environment.

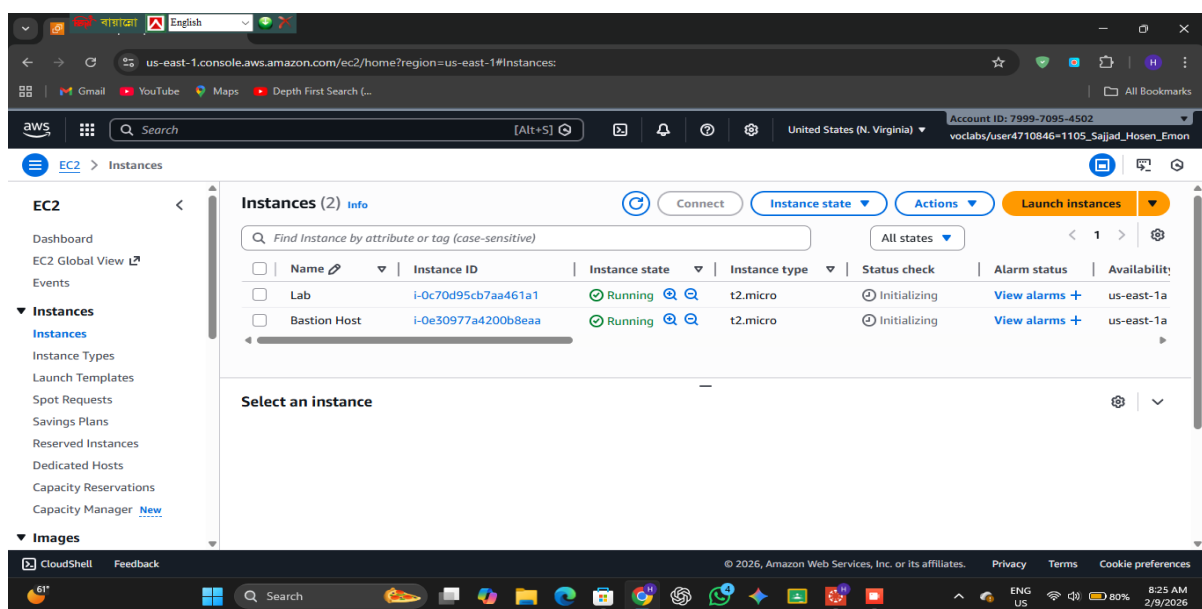
## Scenario

Amazon Elastic Block Store (EBS) is a block-level storage service designed for use with Amazon EC2 instances. It provides persistent storage that remains available even after an instance is stopped. In this lab, a running EC2 instance requires additional storage. Therefore, a new EBS volume is created and attached to the instance. After formatting and mounting the volume, data is written to it. A snapshot is then created to back up the stored data. Finally, the snapshot is used to create a new volume, which is attached and mounted to verify that the data has been successfully restored.

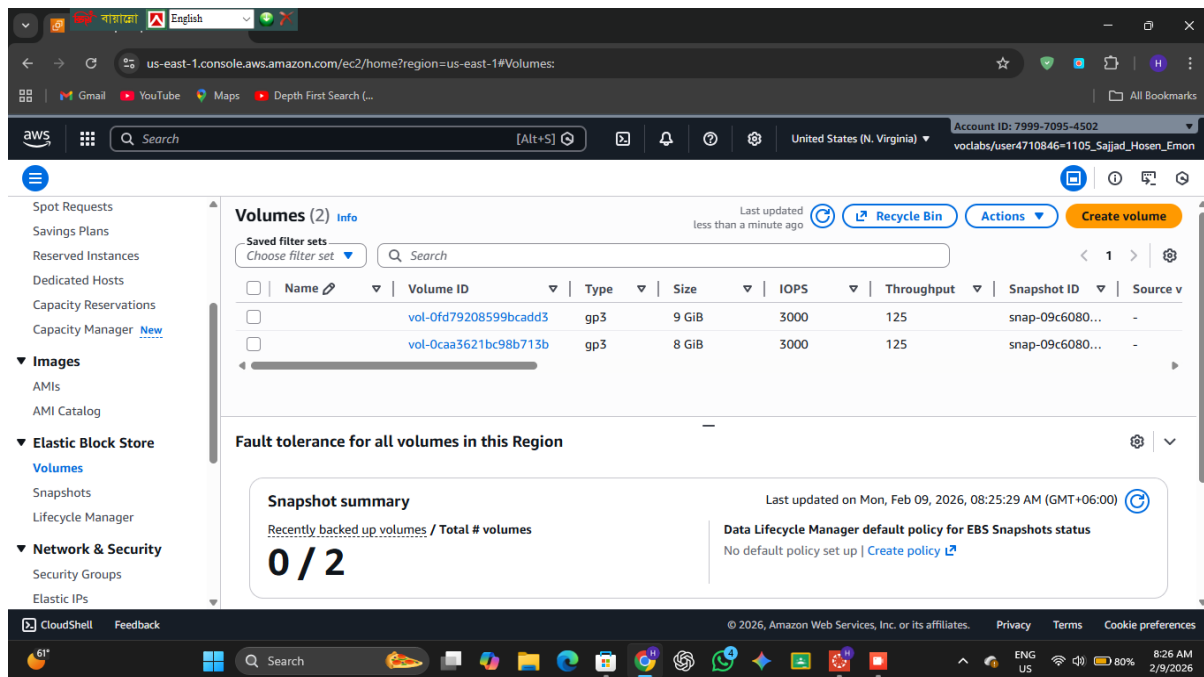


## Working Procedure

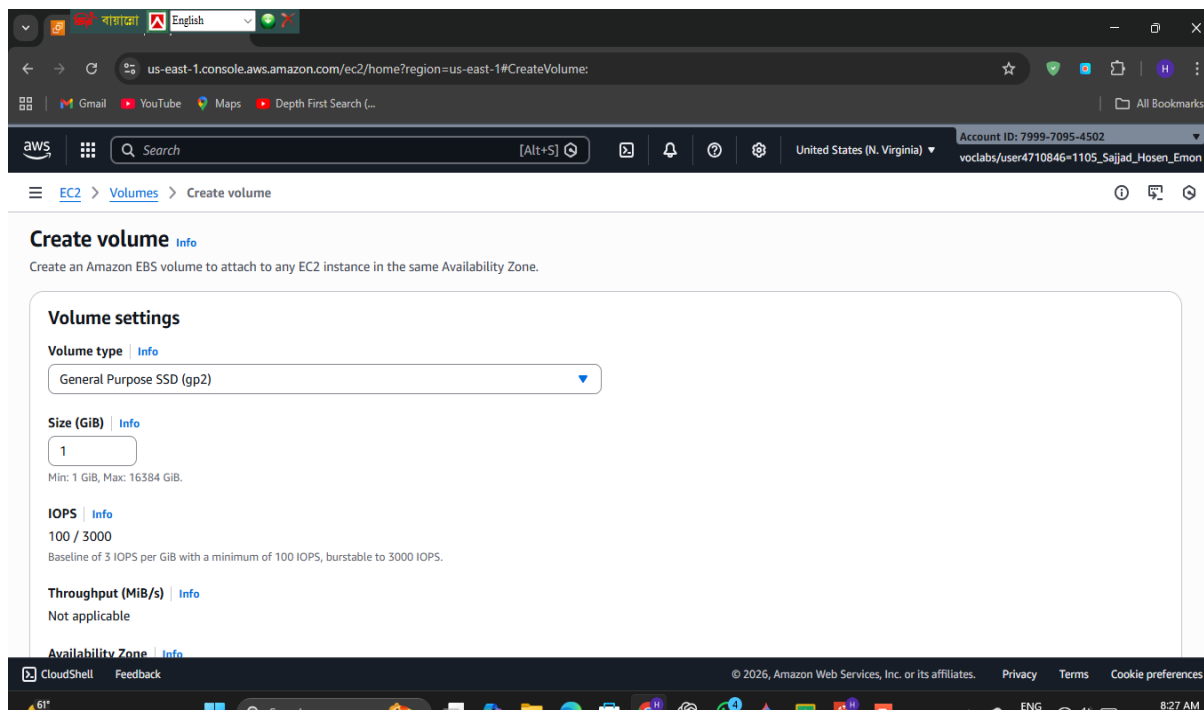
1. Logged in to the AWS Management Console and opened the EC2 Dashboard.



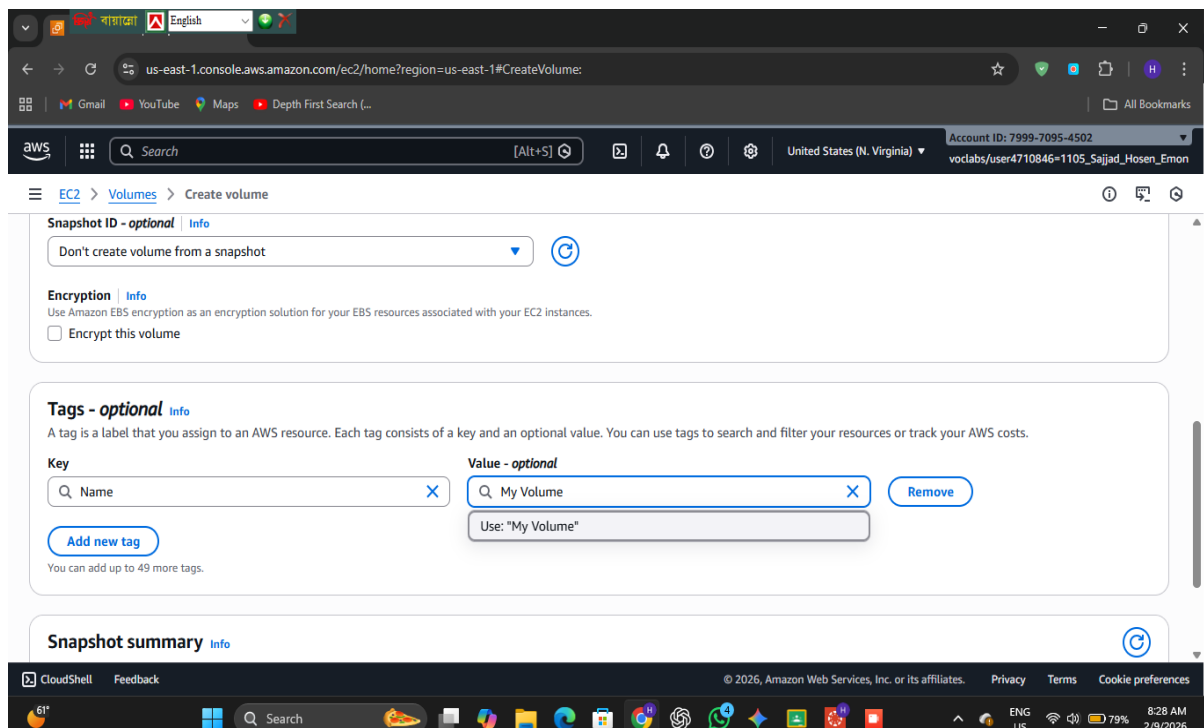
2. Navigated to the Volumes section and selected Create Volume.



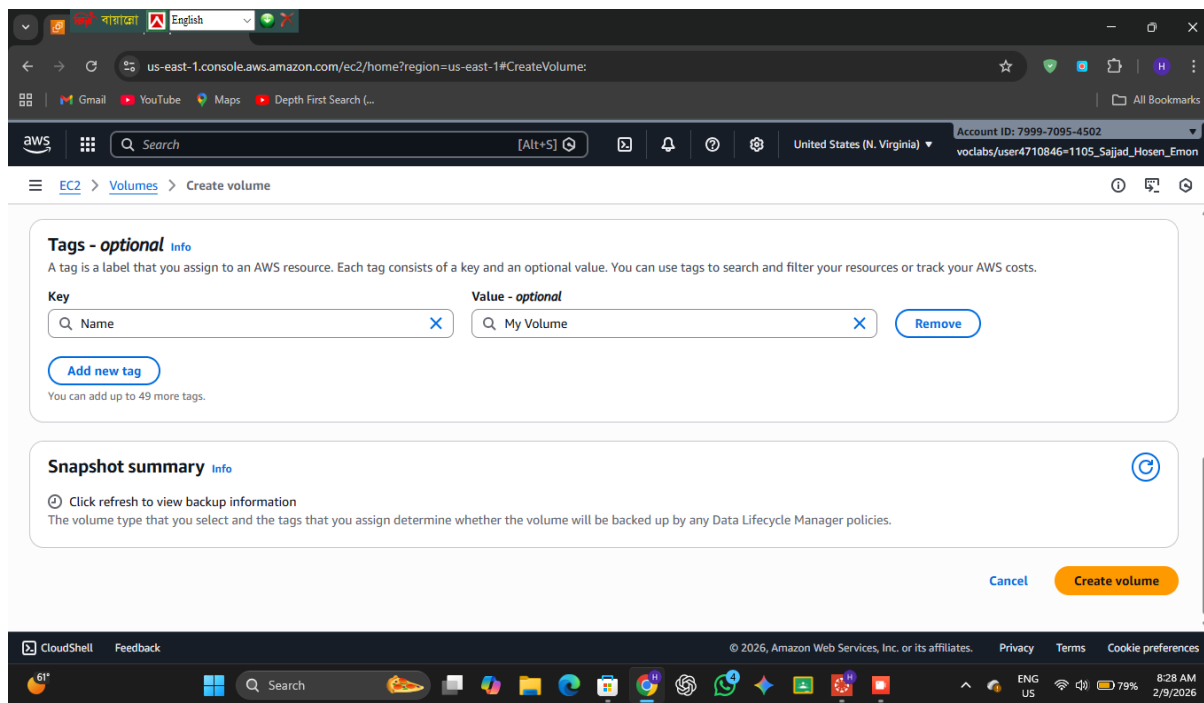
3. Selected **General Purpose SSD (gp2)** as the volume type.
4. Set the volume size to **1 GiB**.



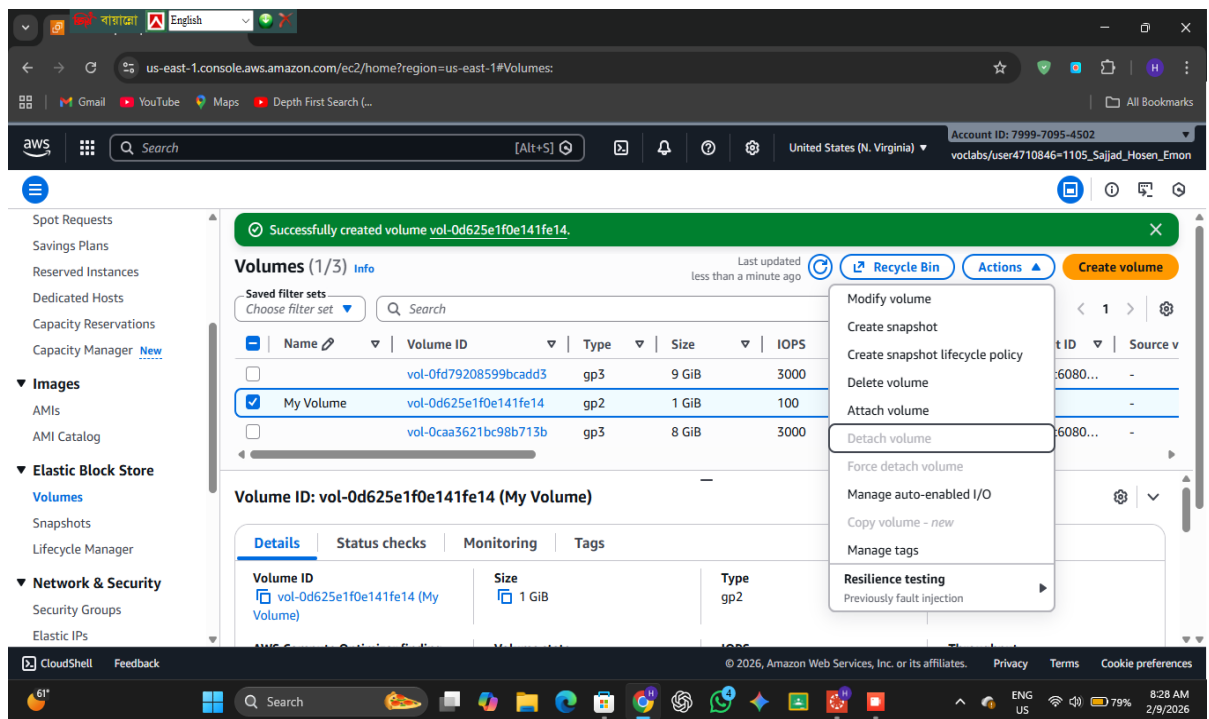
5. Chose the same Availability Zone as the running EC2 instance.
6. Added a tag with the **name My Volume** for identification.



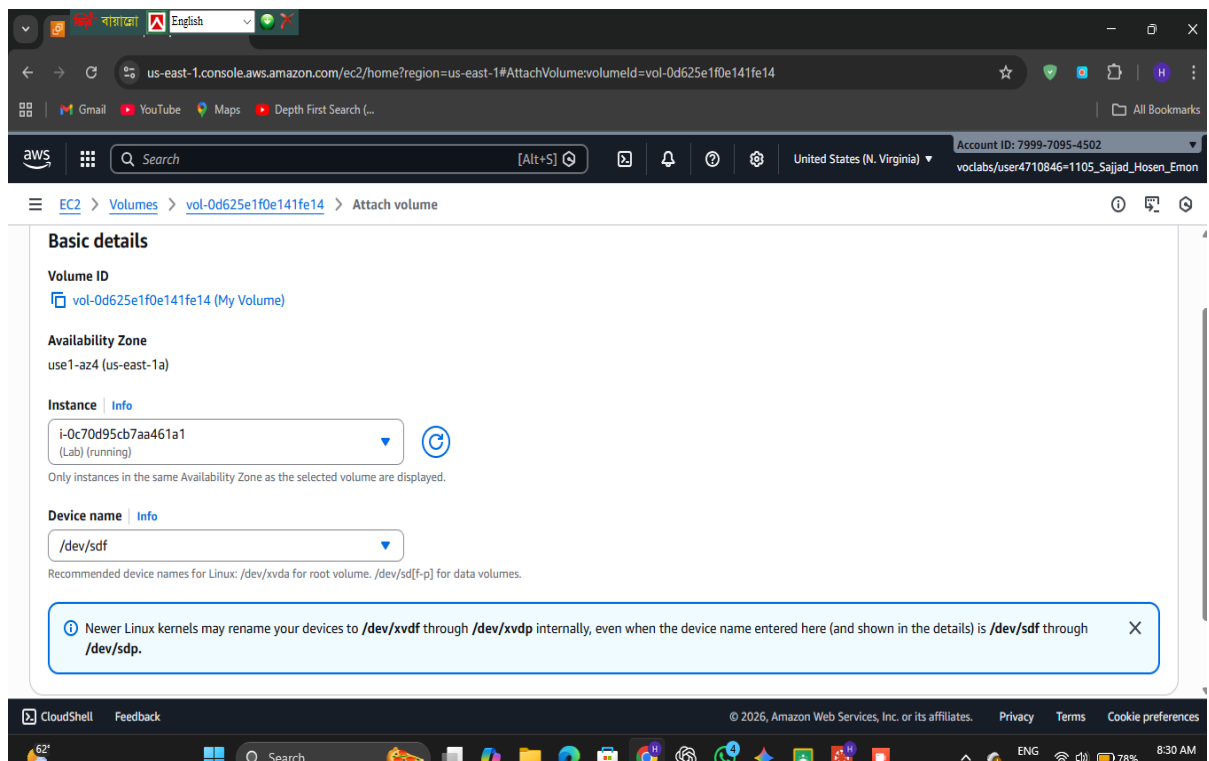
7. Created the volume and waited until its status became Available.



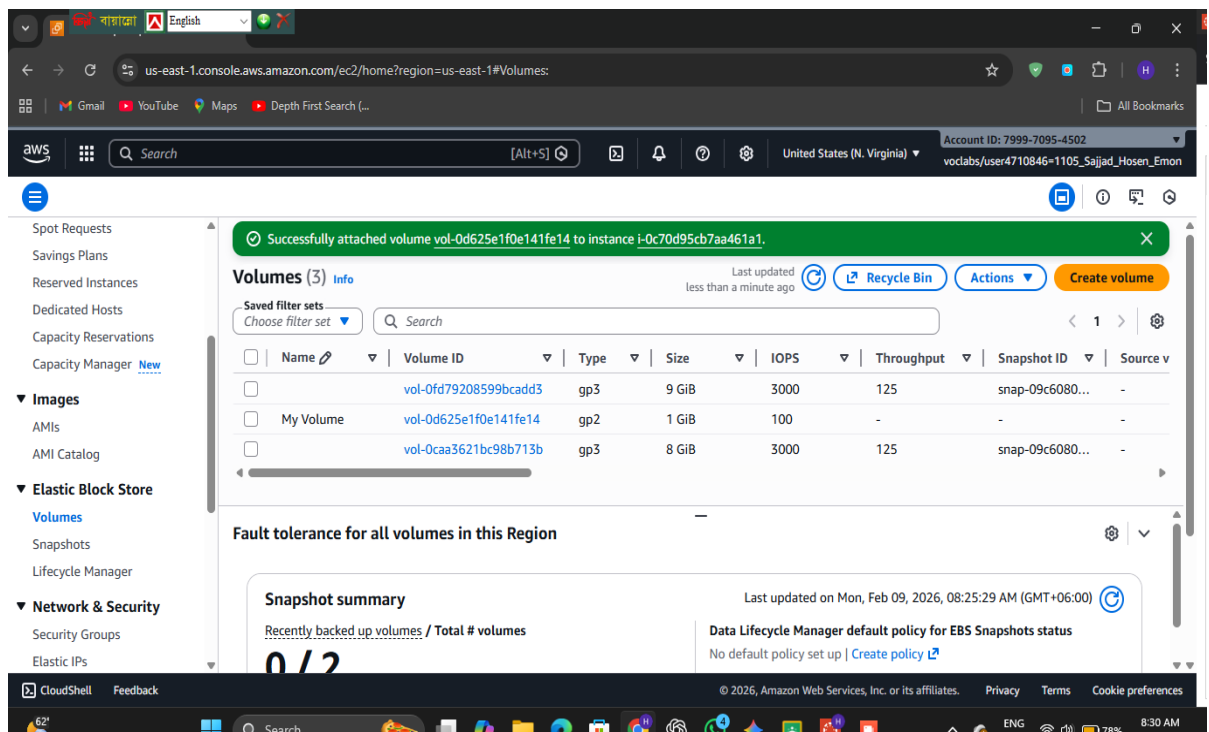
8. Selected the newly created volume and clicked Attach Volume.



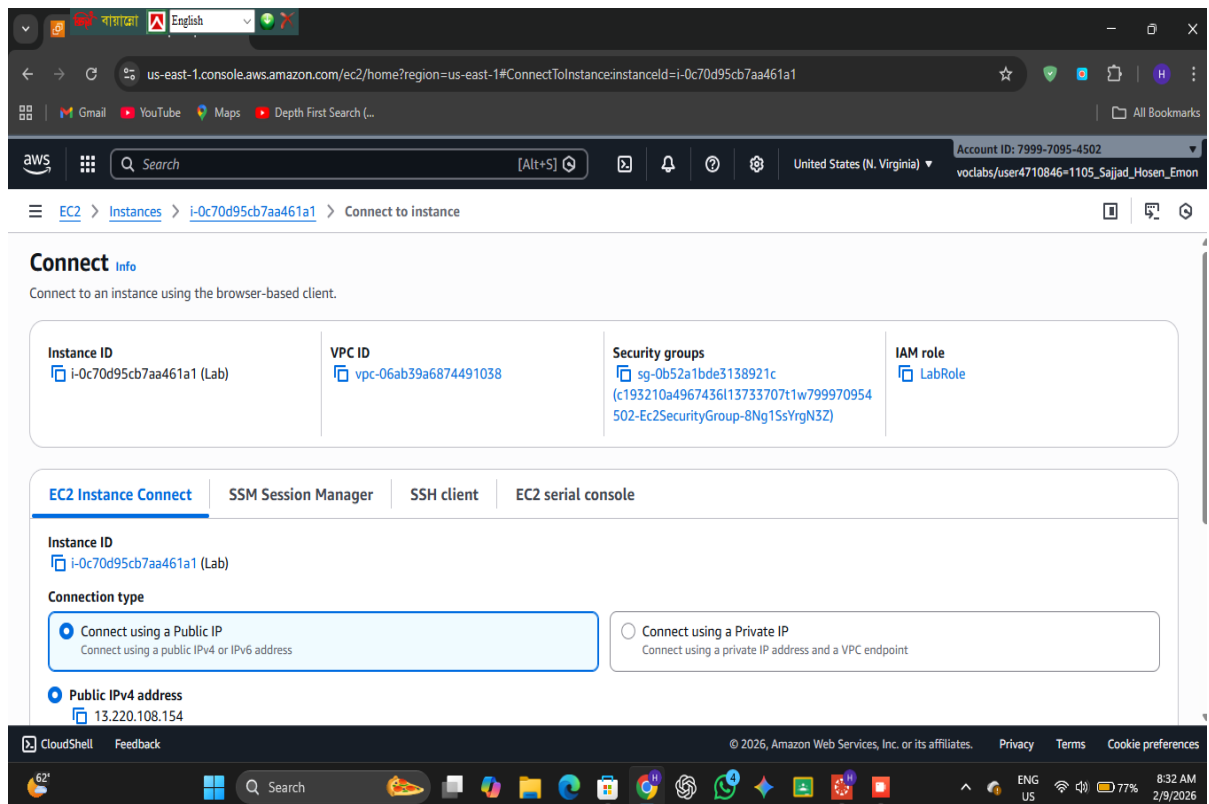
9. Chose the Lab EC2 instance and assigned the device name `/dev/sdf`.



10. Attached the volume successfully and confirmed the status changed to In-use.



11. Opened the Instances section and connected to the EC2 instance using EC2 Instance Connect.



12. Checked the available storage using the df -h command.

The screenshot shows the AWS CloudShell interface. The terminal output includes the Amazon Linux 2023 logo and version information. A command to check disk usage is executed, showing the following table:

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	4.0M	0	4.0M	0%	/dev
tmpfs	481M	0	481M	0%	/dev/shm
tmpfs	193M	448K	192M	1%	/run
/dev/xvda1	8.0G	1.6G	6.4G	20%	/
tmpfs	481M	0	481M	0%	/tmp
/dev/xvda128	10M	1.3M	8.7M	13%	/boot/efi
tmpfs	97M	0	97M	0%	/run/user/1000

Below the table, the instance ID **i-0c70d95cb7aa461a1 (Lab)** is displayed, along with PublicIPs: 13.220.108.154 and PrivateIPs: 10.1.11.217.

13. Created a file system on the new volume using `sudo mkfs -t ext3 /dev/sdf`.

The screenshot shows the AWS CloudShell terminal with the same disk usage table as before. The user then runs the command `sudo mkfs -t ext3 /dev/sdf`. The output shows the successful creation of the file system:

```
[ec2-user@ip-10-1-11-217 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M    0  4.0M   0% /dev
tmpfs           481M    0  481M   0% /dev/shm
tmpfs           193M  448K  192M   1% /run
/dev/xvda1      8.0G  1.6G  6.4G  20% /
tmpfs           481M    0  481M   0% /tmp
/dev/xvda128    10M  1.3M  8.7M  13% /boot/efi
tmpfs           97M    0   97M   0% /run/user/1000

[ec2-user@ip-10-1-11-217 ~]$ sudo mkfs -t ext3 /dev/sdf
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 0c8dc999-b281-421a-8542-3adb291f2b9d
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-10-1-11-217 ~]$
```

The instance ID **i-0c70d95cb7aa461a1 (Lab)** and IP addresses are also visible at the bottom of the terminal window.

- 14. Created a directory `/mnt/data-store` to use as a mount point.
- 15. Mounted the volume to the directory using `sudo mount /dev/sdf /mnt/data-store`.
- 16. Updated the `/etc/fstab` file to enable automatic mounting after reboot.
- 17. Created a text file inside the mounted volume to store data.

The screenshot shows a terminal window in AWS CloudShell. The user is logged in as `ec2-user@ip-10-1-11-217`. The terminal output shows the following commands and results:

```
[ec2-user@ip-10-1-11-217 ~]$ sudo mount /dev/sdf /mnt/data-store
[ec2-user@ip-10-1-11-217 ~]$ 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-10-1-11-217 ~]$ cat /etc/fstab
#
UUID=298c4de6-f9e5-428a-96bf-cc54c4bf3079 / xfs defaults,noatime 1 1
UUID=8FB3-6A0A /boot/efi vfat defaults,noatime,uid=0,gid=0,umask=0077,shortname=winnt,x-systemd.automount 0 2
/dev/sdf /mnt/data-store ext3 defaults,noatime 1 2
[ec2-user@ip-10-1-11-217 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M  0  4.0M   0% /dev
tmpfs           481M  0  481M   0% /dev/shm
tmpfs           193M  448K 192M   1% /run
/dev/xvda1       8.0G  1.6G  6.4G  20% /
tmpfs           481M  0  481M   0% /tmp
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs           97M  0  97M   0% /run/user/1000
/dev/xvdf       975M  60K  924M   1% /mnt/data-store
[ec2-user@ip-10-1-11-217 ~]$ sudo sh -c "echo some text has been written > /mnt/data-store/file.txt"
[ec2-user@ip-10-1-11-217 ~]$ cat /mnt/data-store/file.txt
some text has been written
[ec2-user@ip-10-1-11-217 ~]$
```

Below the terminal output, the instance details are shown:

**i-0c70d95cb7aa461a1 (Lab)**  
PublicIPs: 13.220.108.154 PrivateIPs: 10.1.11.217

18. Verified the file to confirm that data was written successfully.
19. • Returned to the EC2 console and selected the previously created volume **My Volume**.

The screenshot shows the AWS Management Console for the Elastic Block Store (EBS) Volumes page. The left sidebar shows the navigation menu with categories: Elastic Block Store, Network & Security, Load Balancing, and Auto Scaling. The main content area shows a list of volumes with the following columns: Name, Volume ID, Type, Size, IOPS, Throughput, Snapshot ID, and Source v.

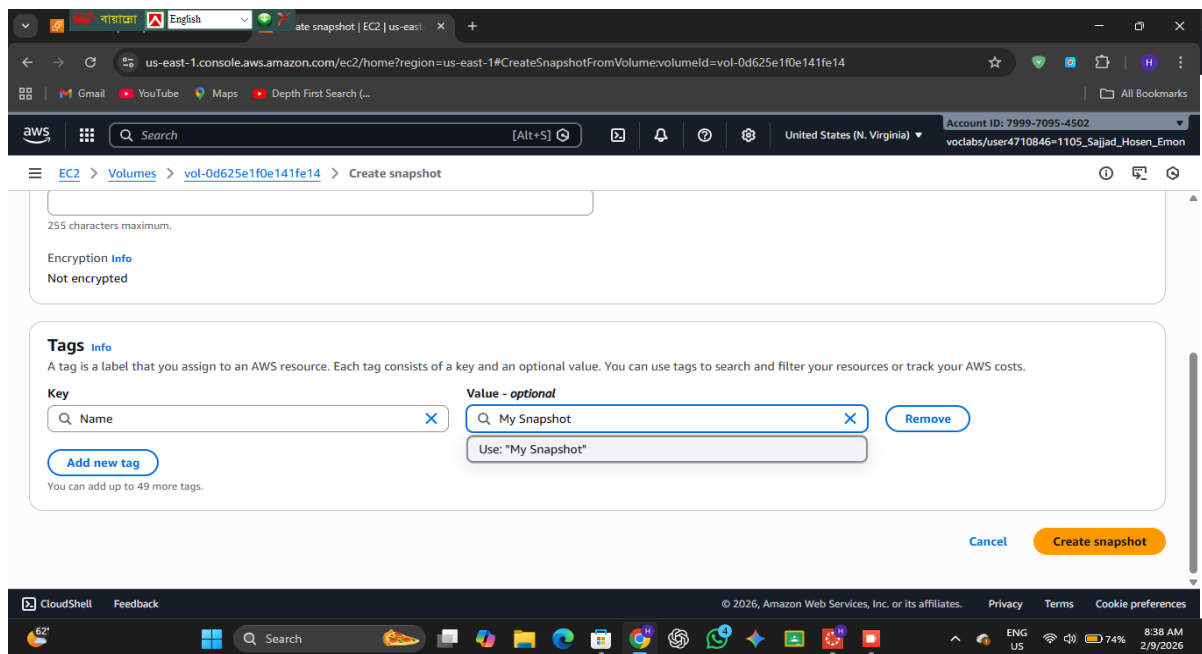
Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID	Source v
	vol-0fd79208599bcadd3	gp3	9 GiB	3000	125	snap-09c6080...	-
<input checked="" type="checkbox"/> My Volume	vol-0d625e1f0e141fe14	gp2	1 GiB	100	-	-	-
	vol-0caa3621bc98b713b	gp3	8 GiB	3000	125	snap-09c6080...	-

Below the table, the details for the selected volume (vol-0d625e1f0e141fe14) are shown. The details include:

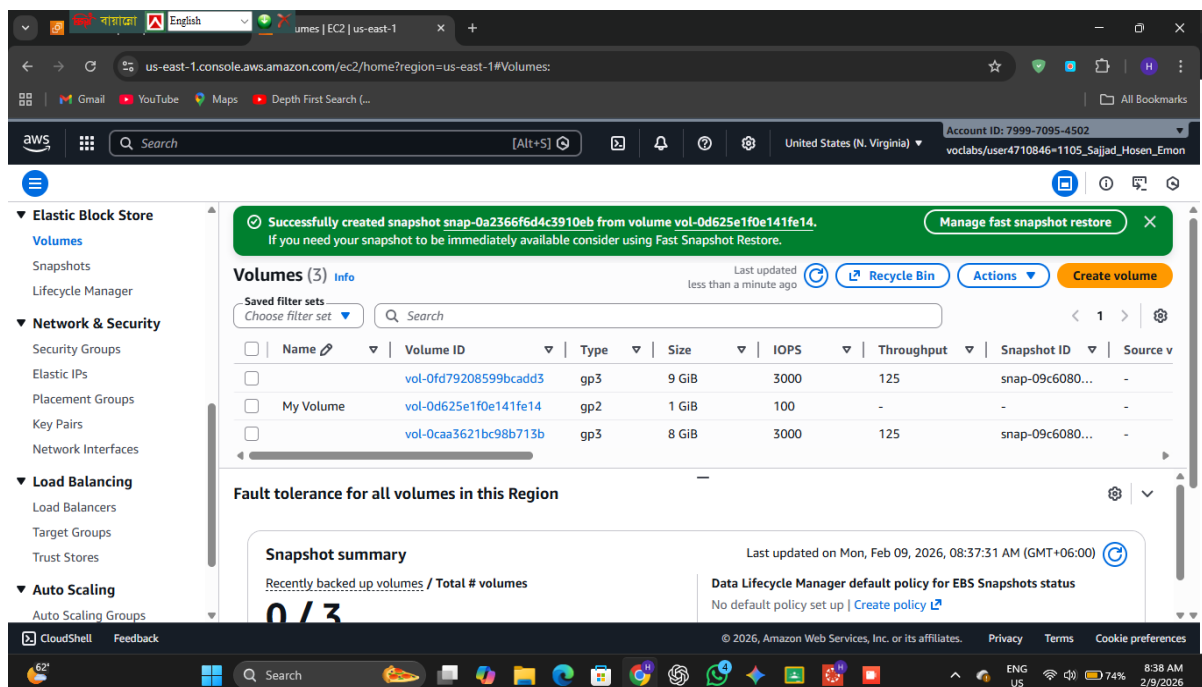
- Volume ID: vol-0d625e1f0e141fe14 (My Volume)
- Size: 1 GiB
- Type: gp2
- Status check: ✔ Okay

20. • Clicked on **Actions** → **Create Snapshot** to create a backup of the volume.

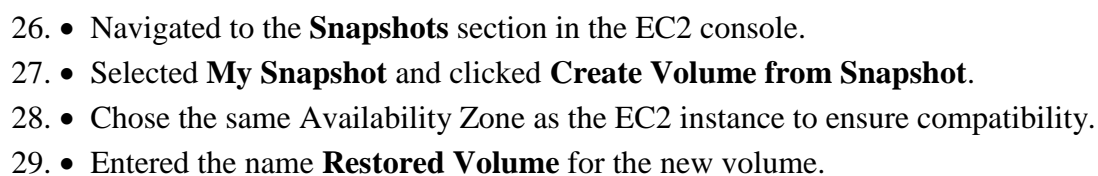




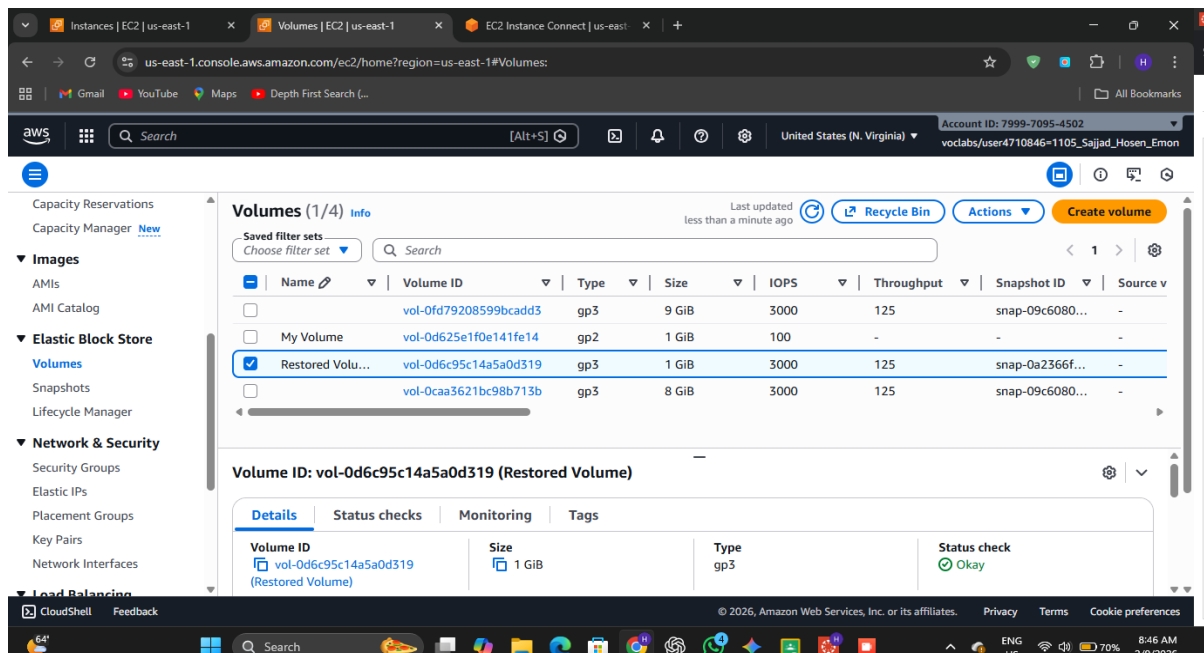
21. • Provided the snapshot name **My Snapshot** for easy identification.
22. • Initiated the snapshot creation process and waited until the status changed from **Pending** to **Completed**.



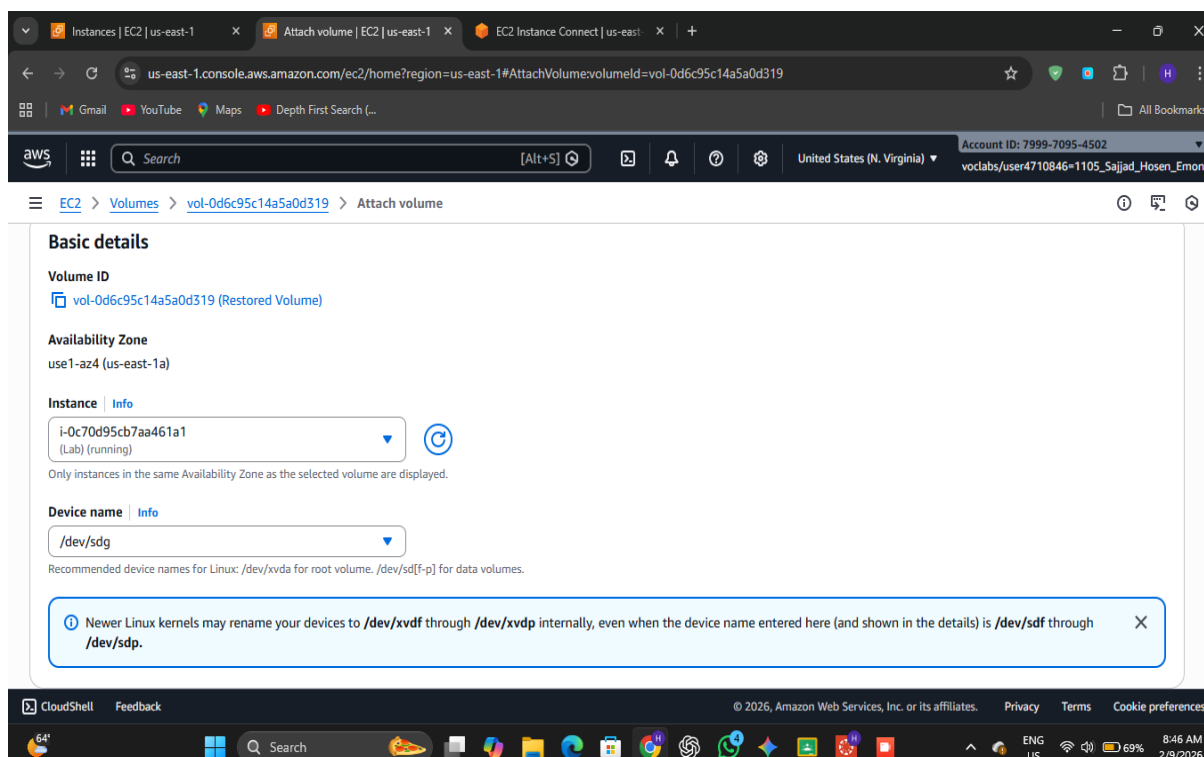
23. • After the snapshot was created, connected back to the EC2 instance terminal.
24. • Deleted the previously created file from the original volume to simulate data loss.
25. • Verified that the file was successfully deleted from the directory.



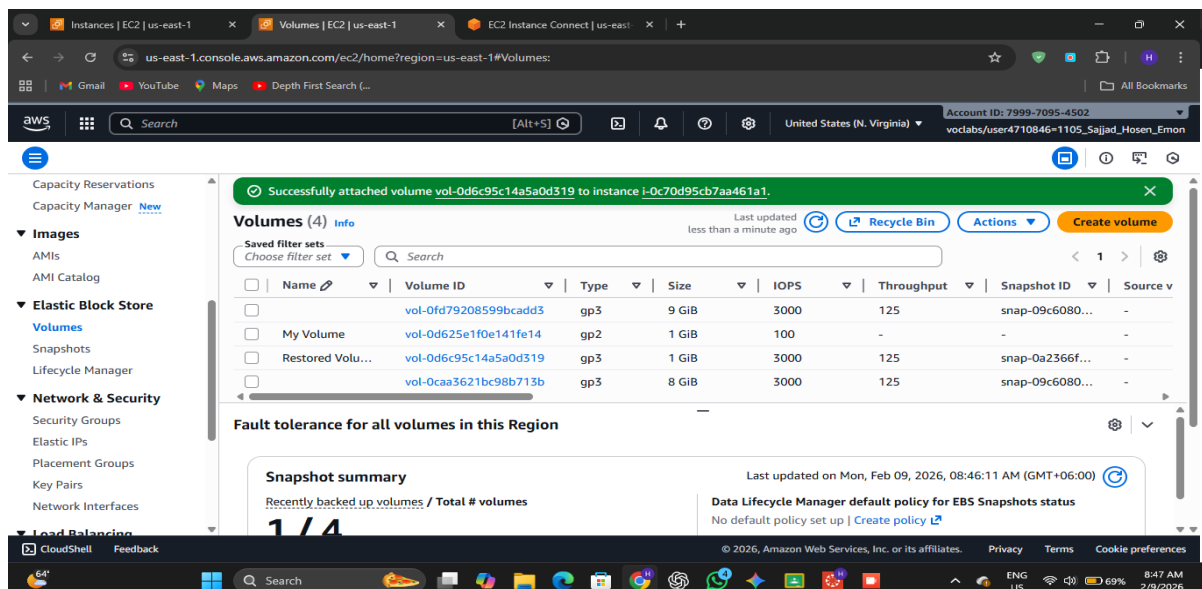
31. • Selected the restored volume and clicked **Attach Volume**.



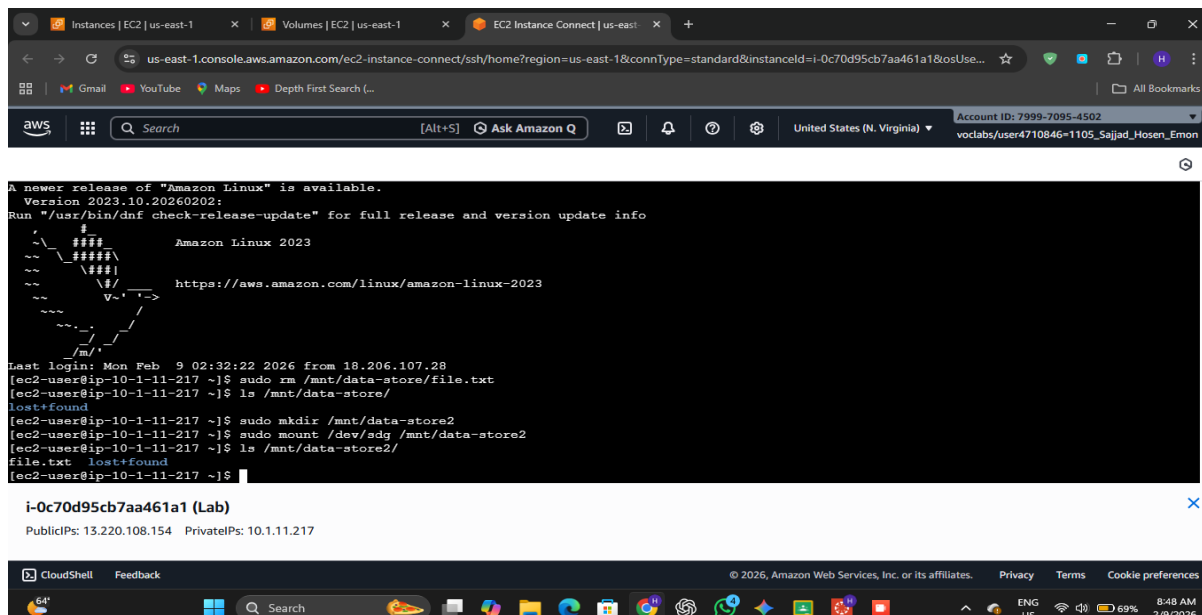
32. • Chose the Lab EC2 instance and assigned the device name `/dev/sdg`.



33. • Confirmed that the restored volume status changed to **In-use**.



34. • Opened the EC2 terminal again to configure the restored storage.
35. • Created a new directory `/mnt/data-store2` to serve as the mount point.
36. • Mounted the restored volume using the command `sudo mount /dev/sdg /mnt/data-store2`.
37. • Listed the files in the mounted directory to verify the restoration process.
38. • Confirmed that **file.txt** was recovered successfully, proving that the snapshot restoration worked correctly.
39. Created a new directory `/mnt/data-store2` for mounting the restored volume.
40. Mounted the restored volume using `sudo mount /dev/sdg /mnt/data-store2`.
41. Checked the directory and verified that the previously stored file was successfully recovered.



## 42. Submission report:

Submission Report

[Executed at: Sun Feb 8 18:49:04 PST 2026]

gradeFile = /mnt/vocwork5/grader/eee\_G\_2692329/asn4967435\_6/asn4967436\_1/tmp/temp\_uf\_02082026/.eb16ki260bd31EAhTHJ  
reportFile =/mnt/vocwork5/grader/eee\_G\_2692329/asn4967435\_6/asn4967436\_1/tmp/temp\_uf\_02082026/.rbjk1rhsgf15cQNc3Jd  
/mnt/vocwork5/grader/eee\_G\_2692329/asn4967435\_6/asn4967436\_1/tmp/temp\_uf\_02082026/.eb16ki260bd31EAhTHJ  
Started: 2026-02-08 18:48:57  
region: us-east-1  
profile: default

Evaluating Task 1 - Create EBS volume  
Lab instance AZ: us-east-1a  
Lab instance public IP: 13.220.108.154  
found volume size: 9  
found volume size: 1  
Volume name: My Volume  
New volume id: vol-0d625e1f0e141fe14

For any checks where you did not receive full points, there are sometimes helpful details  
ded in the submission report

Submission Report

New volume AZ: us-east-1a  
New volume state: in-use  
Task 1 - Success! Evidence found that a new 1 GB EBS volume named My Volume was created in the same Availability Zone as the  
found volume size: 1  
Volume name: Restored Volume  
Found restored volume id: vol-0d6c95c14a5a0d319  
found volume size: 8

Evaluating Task 2 - Attach volume  
New EBS volume vol-0d625e1f0e141fe14 attached to Lab instance  
Task 2 - Success! Evidence found that a new EBS volume named My Volume was attached to the instance.

Evaluating Task 4 - Volume mounted  
Found key for instance.  
Result of df -h query on Lab instance:  
"Filesystem Size Used Avail Use% Mounted on  
ndevtmpfs 4.0M 0 4.0M 0% /dev/ndevtmpfs 481M 0  
Task 4 - Success! Evidence found in the df -h output that the EBS volume was mounted as /dev/xvdf.

Submission Report

Evaluating Task 4 - Volume mounted  
Found key for instance.  
Result of df -h query on Lab instance:  
"Filesystem Size Used Avail Use% Mounted on  
ndevtmpfs 4.0M 0 4.0M 0% /dev/ndevtmpfs 481M 0  
Task 4 - Success! Evidence found in the df -h output that the EBS volume was mounted as /dev/xvdf.

Evaluating Task 5 - Snapshot created  
Found snapshot of volume ID: vol-0d625e1f0e141fe14  
Volume ID of the 1 GB volume attached to Lab instance:vol-0d625e1f0e141fe14  
Task 5 - Success! Evidence found that a snapshot was created and that the volume ID of the 1GB volume attached to the Lab i

Evaluating Task 6 - Snapshot restored  
evidence found of /dev/xvdf and /mnt/data-store2 in df -h output.  
Task 6 - Success! Evidence found that the snapshot was restored and mounted.

Completed: 2026-02-08 18:49:03

For any checks where you did not receive full points, there are sometimes helpful details  
ded in the submission report

43. Total score

Submit		Submission Report	Grades
Total score		25/25	
Task 1 - Create EBS volume		5/5	
Task 2 - Attach volume		5/5	
Task 4 - Volume mounted		5/5	
Task 5 - Snapshot created		5/5	
Task 6 - Snapshot restored		5/5	

44.

## Conclusion

In this lab, Amazon EBS was used to create persistent storage for an EC2 instance. The volume was successfully attached, formatted, and mounted for data storage. A snapshot was created to ensure backup, and the data was restored by creating a new volume from the snapshot. This lab highlighted the importance of Amazon EBS in providing reliable, scalable, and durable storage solutions in cloud computing.