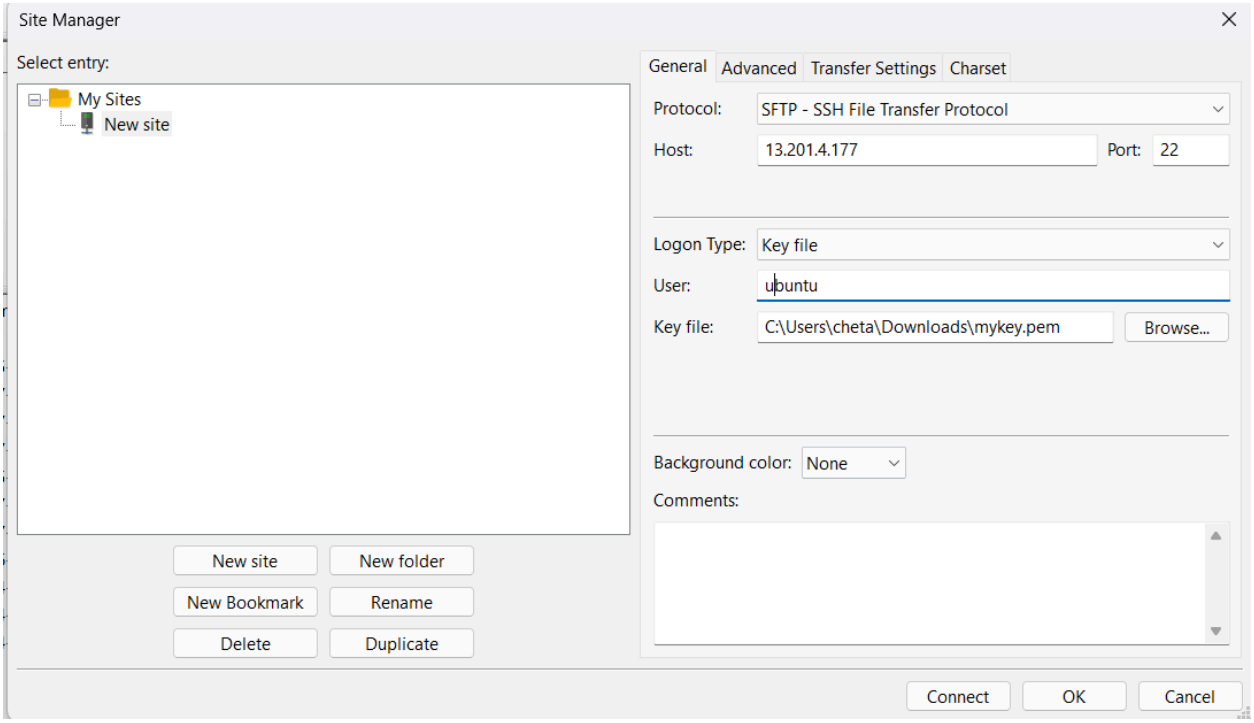
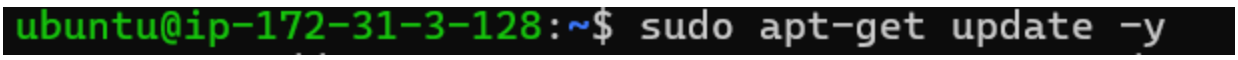
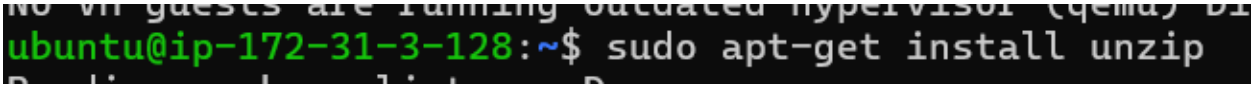
1. Custom AMI Creation

Step-by-Step:

1. **Launch Ubuntu EC2 Instance**
    - Choose Ubuntu 22.04 LTS
    - Use existing or new key pair
    - Allow ports 22 (SSH), 80 (HTTP) in security group
2. **Install Web Server**







```
ubuntu@ip-172-31-3-128:~$ sudo systemctl enable apache2
```

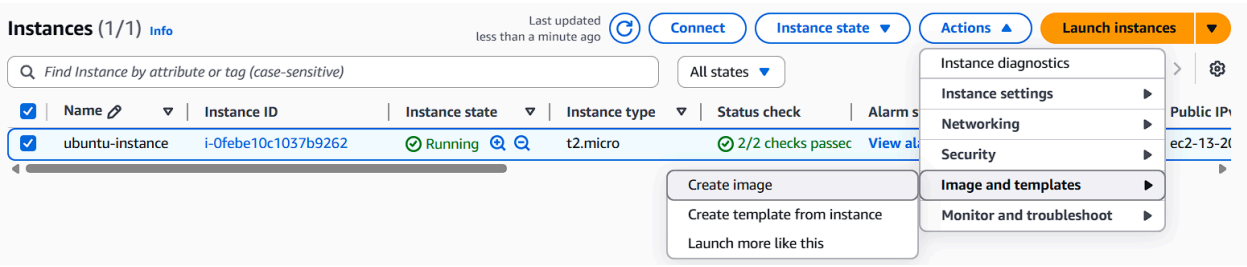
```
ubuntu@ip-172-31-3-128:~$ sudo systemctl restart apache2
```

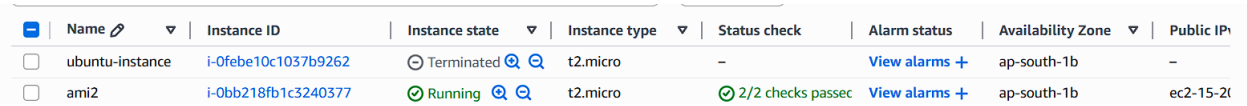
```
ubuntu@ip-172-31-3-128:~$ sudo systemctl status apache2
```

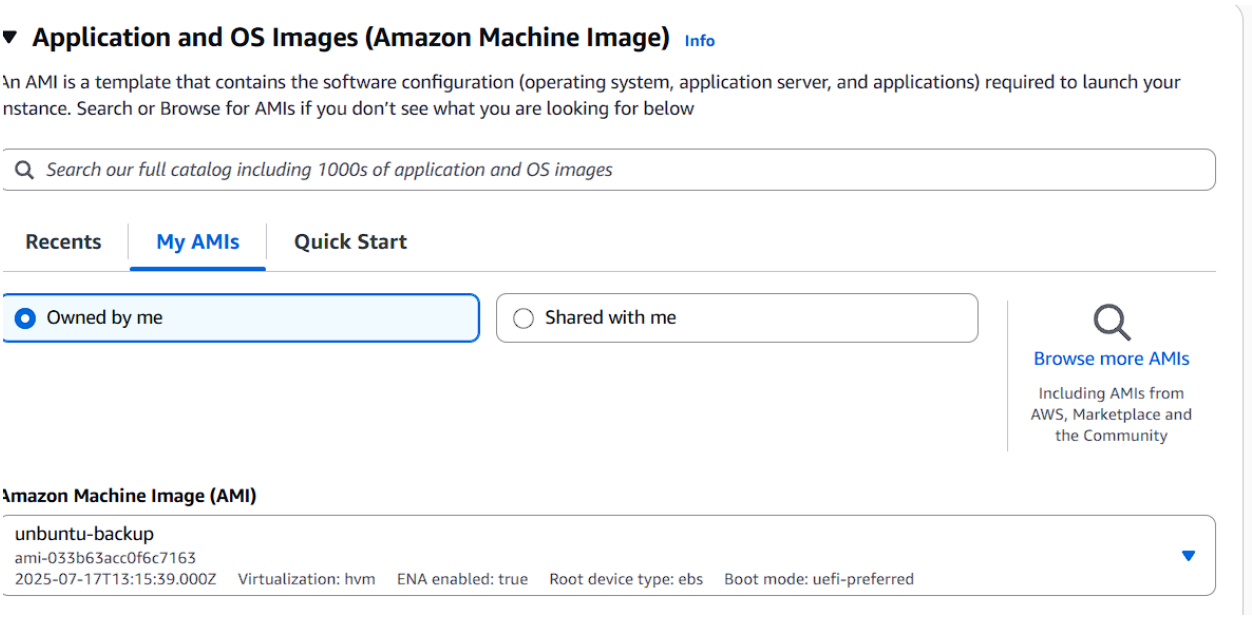
```
ubuntu@ip-172-31-3-128:~$ sudo systemctl restart apache2
ubuntu@ip-172-31-3-128:~$ unzip chef-website-template.zip
```

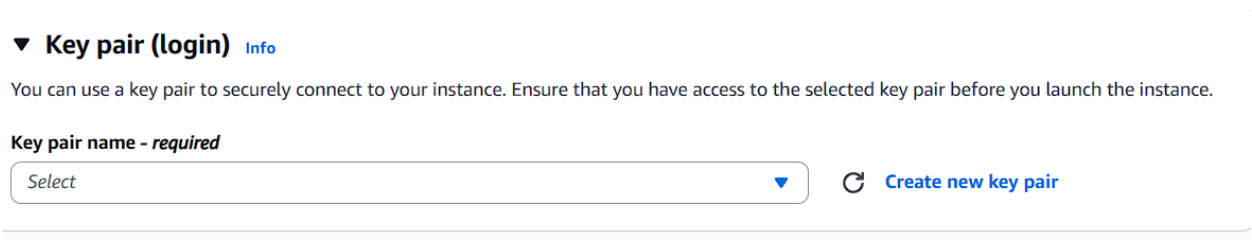
```
ubuntu@ip-172-31-3-128:~$ ls
chef-website-template  chef-website-template.zip
ubuntu@ip-172-31-3-128:~$ ls -la
total 828
drwxr-x--- 5 ubuntu ubuntu   4096 Jul 17 13:08 .
drwxr-xr-x 3 root   root     4096 Jul 17 12:36 ..
-rw-r--r-- 1 ubuntu ubuntu    220 Mar 31  2024 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu   3771 Mar 31  2024 .bashrc
drwx------ 2 ubuntu ubuntu   4096 Jul 17 12:39 .cache
-rw-r--r-- 1 ubuntu ubuntu    807 Mar 31  2024 .profile
drwx------ 2 ubuntu ubuntu   4096 Jul 17 12:36 .ssh
-rw-r--r-- 1 ubuntu ubuntu      0 Jul 17 12:39 .sudo_as_admin_successful
drwxrwxr-x 7 ubuntu ubuntu   4096 Jul 17 13:08 chef-website-template
-rw-rw-r-- 1 ubuntu ubuntu 814760 Jul 17 12:59 chef-website-template.zip
```

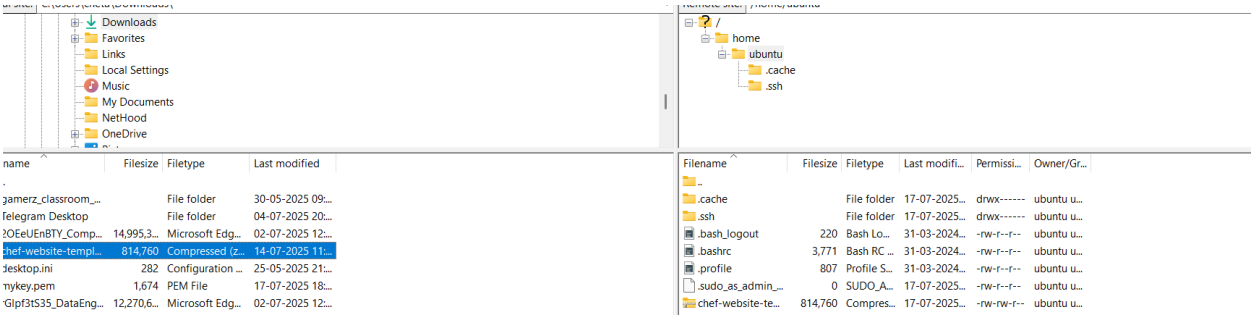
```
ubuntu@ip-172-31-3-128:~$ sudo cp -r chef-website-template/* /var/www/html
```











1. **Test Website**
   - Visit `http://<EC2-Public-IP>` in your browser → Should show "Hello from custom AMI"
2. **Create Custom AMI**
   - In EC2 Console → Select your instance → Actions → **Image and templates → Create Image**
   - Give it a name and create

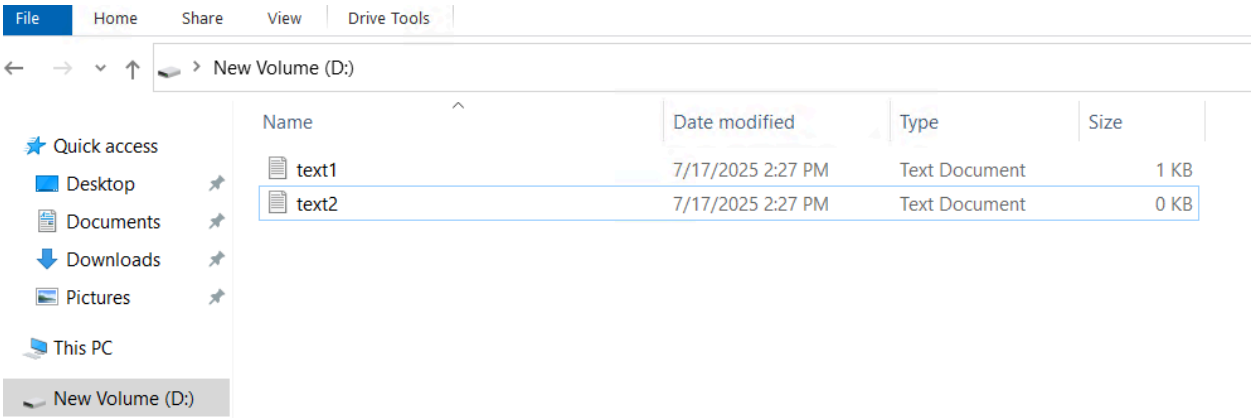
- Wait until AMI becomes **available** in EC2 → AMIs

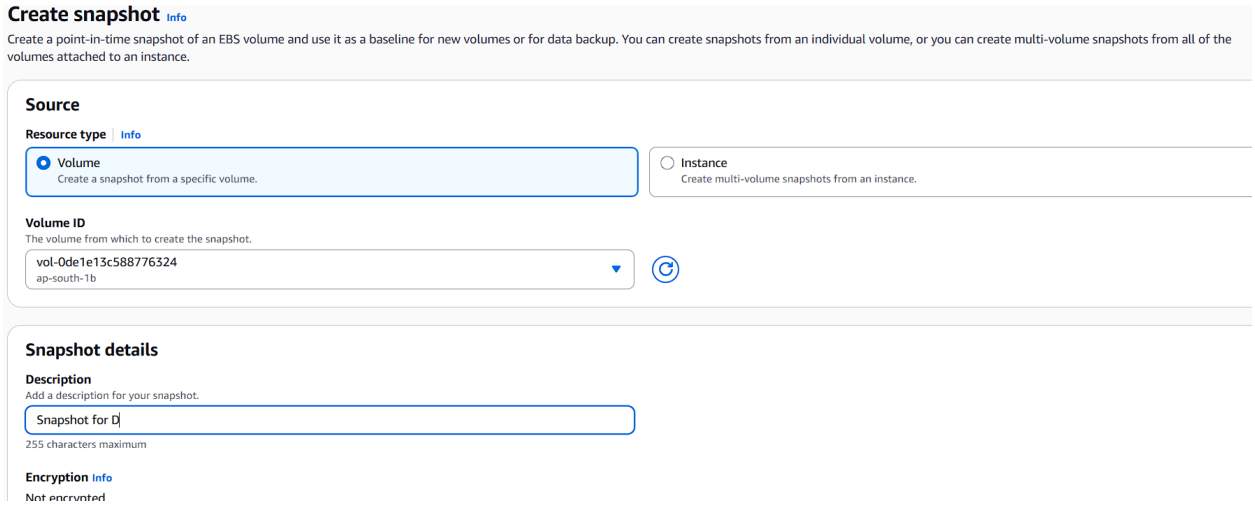
3. **Launch Instance from AMI**
   - Go to **AMIs**, select the custom AMI → Click **Launch instance from image**
   - Choose instance type, key pair, etc.
   - After launch, test in browser → Web server should still work

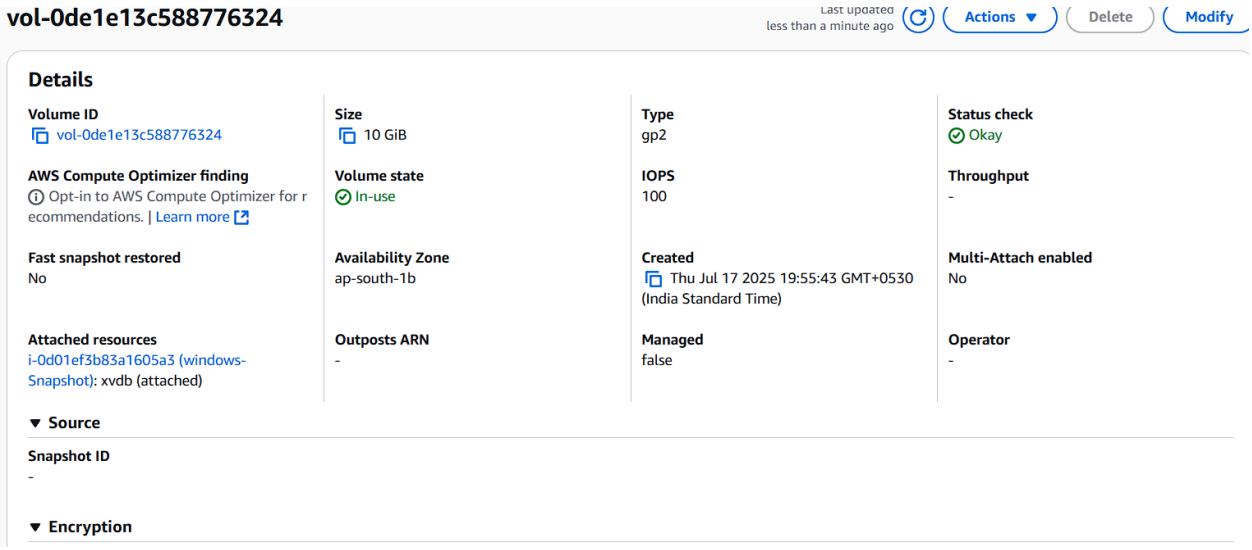
Snapshot for Windows EC2 Instance

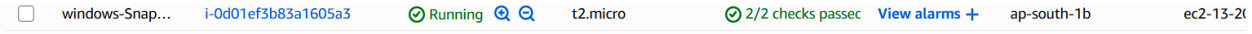
Objective:

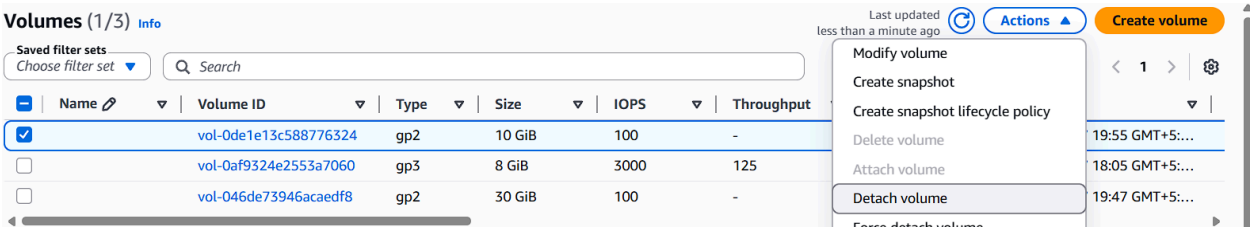
- Create a snapshot (backup) of the Windows EC2 instance's volume (usually the C: drive).
- Restore it later if needed, either fully or partially.

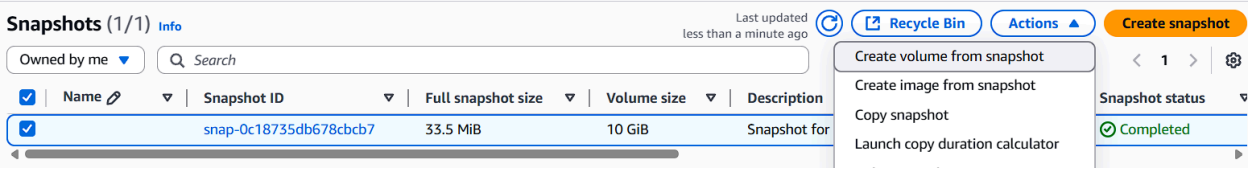


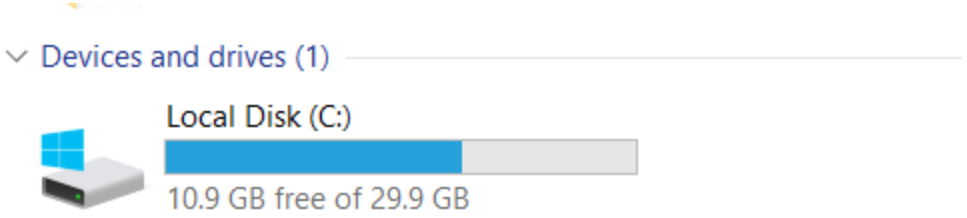


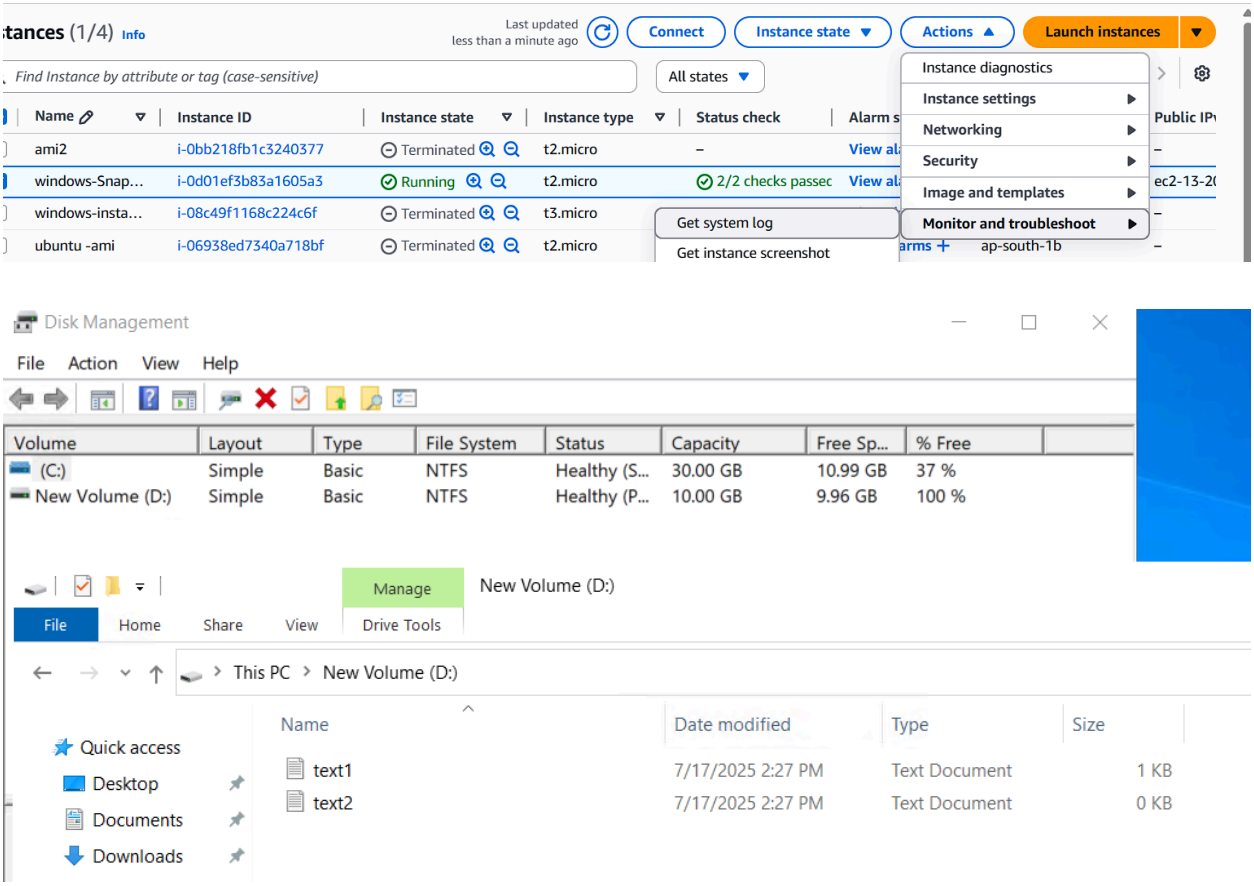

vol-0de1e13c588776324
Last updated less than a minute ago
Actions
Delete
Modify
Details
Volume ID
vol-0de1e13c588776324
Size
10 GiB
Type
gp2
Status check
Okay
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more
Volume state
In-use
IOPS
100
Throughput
-
Fast snapshot restored
No
Availability Zone
ap-south-1b
Created
Thu Jul 17 2025 19:55:43 GMT+0530 (India Standard Time)
Multi-Attach enabled
No
Attached resources
i-0d01ef3b83a1605a3 (windows-Snapshot): xvdb (attached)
Outposts ARN
-
Managed
false
Operator
-
Source
Snapshot ID
-
Encryption


windows-Snap...
i-0d01ef3b83a1605a3
Running
t2.micro
2/2 checks passec
View alarms +
ap-south-1b
ec2-13-2(


Volumes (1/3) Info
Last updated less than a minute ago
Actions
Create volume
Saved filter sets
Choose filter set
Search
Modify volume
Create snapshot
Create snapshot lifecycle policy
Delete volume
Attach volume
Detach volume
Name
Volume ID
Type
Size
IOPS
Throughput
vol-0de1e13c588776324 gp2 10 GiB 100 - 19:55 GMT+5:...
vol-0af9324e2553a7060 gp3 8 GiB 3000 125 18:05 GMT+5:...
vol-046de73946acaedf8 gp2 30 GiB 100 - 19:47 GMT+5:...


Snapshots (1/1) Info
Last updated less than a minute ago
Recycle Bin
Actions
Create snapshot
Owned by me
Search
Create volume from snapshot
Create image from snapshot
Copy snapshot
Launch copy duration calculator
Name
Snapshot ID
Full snapshot size
Volume size
Description
Snapshot status
snap-0c18735db678cbcb7 33.5 MiB 10 GiB Snapshot for Completed


Devices and drives (1)
Local Disk (C:)
10.9 GB free of 29.9 GB



Part 1: Create a Snapshot of the Windows EC2 Volume

1. **Identify the Volume:**
   - Go to the AWS EC2 Console.
   - In the left menu, click on **Volumes** under **Elastic Block Store**.
   - Find the volume attached to your Windows EC2 instance (it will likely show as `/dev/sda1` or "Root device").
2. **Create the Snapshot:**
   - Select the volume.
   - Click **Actions > Create snapshot**.
   - Provide a name and description for the snapshot (e.g., "Windows backup July 17").
   - Click **Create snapshot**.

3. **Monitor Progress:**
    - Go to the **Snapshots** section in EC2.
    - Wait for the status to change from "pending" to "completed".

Part 2: Restore the Snapshot

Option A: Restore as a New Root Volume (Full Recovery)

1. **Create a Volume from the Snapshot:**
    - In the EC2 Console, go to **Snapshots**.
    - Select the snapshot and click **Actions > Create volume**.
    - Select the same Availability Zone (AZ) as the original EC2 instance.
2. **Detach the Old Volume (Optional if full restore):**
    - Stop the Windows EC2 instance.
    - Go to the **Volumes** section, find the root volume attached to the instance.
    - Select it and click **Actions > Detach volume**.
3. **Attach the New Volume:**
    - Go to the new volume created from the snapshot.
    - Click **Actions > Attach volume**.
    - Select the instance and set the device name as `/dev/sda1` (this is critical to make it the root device).
4. **Start the EC2 Instance:**
    - After attaching the volume, start the instance.
    - It should now boot using the restored volume.

3. Cost Optimization (Mumbai Region)

Step-by-Step:

1. **Go to AWS Cost Explorer**

- Enable it if it’s the first time
- Review usage and filter by **Mumbai (ap-south-1)** region

2. **Check for Orphaned Resources**
    - Go to:
        - **EC2 → Volumes** → Look for unattached volumes
        - **EC2 → Snapshots** → Look for old, unused snapshots
        - **Elastic IPs** → Check for unassociated IPs
        - **AMIs** → Unused custom images
3. **Cleanup**
    - Delete unattached EBS volumes
    - Delete old snapshots
    - Release unused Elastic IPs
    - Deregister and delete AMIs if not needed