

## EBS SNAPSHOT

Amazon Elastic Block Store (EBS) snapshots are point-in-time copies of data stored on Amazon EBS volumes in the Amazon Web Services (AWS) cloud. EBS snapshots are used for backup, recovery, and data migration. Here are some key points about EBS snapshots:

1. **Backup and Recovery:** EBS snapshots allow you to create backups of your EBS volumes. If you accidentally delete data or if there is a failure, you can use these snapshots to restore your volumes to a specific point in time.
2. **Incremental Backups:** EBS snapshots are incremental, meaning that only the data that has changed since the last snapshot is stored. This helps to save storage space and reduce costs.
3. **Consistency:** When you create a snapshot, it captures a consistent view of the entire EBS volume. This is particularly useful for databases and other applications that require data consistency.
4. **Volume Creation:** You can use EBS snapshots to create new EBS volumes. This is useful for creating multiple EBS volumes with the same data, or for migrating data between regions.
5. **Amazon Data Lifecycle Manager (DLM):** DLM is a service that automates the creation, retention, and deletion of EBS snapshots. You can define snapshot policies to schedule and manage the lifecycle of your EBS snapshots.
6. **Encryption:** EBS snapshots can be encrypted for additional security. You can choose to encrypt both the EBS volume and its snapshots using AWS Key Management Service (KMS) keys.
7. **Cross-Region Copy:** EBS snapshots can be copied to other AWS regions, allowing you to have backups of your data in different geographic locations for disaster recovery purposes.

## USE CASES OF EBS SNAPSHOT

EBS snapshots serve several use cases in AWS environments, providing flexibility, data protection, and ease of management. Here are some common use cases for EBS snapshots:

1. **Backup and Recovery: Data Protection:** EBS snapshots are used to create backups of your EBS volumes, ensuring data durability and protection against accidental data loss or corruption. **Point-in-Time Recovery:** Snapshots allow you to restore your EBS volumes to specific points in time, providing a mechanism for recovering from errors or unwanted changes.
2. **Disaster Recovery: Cross-Region Replication:** EBS snapshots can be copied to different AWS regions, providing a disaster recovery solution. In the event of a region-wide failure, you can launch new volumes from snapshots in another region.
3. **Testing and Development: Creating Clones:** Snapshots enable the creation of identical copies of EBS volumes, which is useful for testing and development environments. Developers can work with realistic datasets without impacting production data.

4. **Data Migration: Volume Migration:** Snapshots facilitate the migration of EBS volumes between EC2 instances, Availability Zones, or regions. You can create a snapshot of an existing volume and use it to create a new volume in a different location.
5. **Consistent Backups for Databases: Application Consistency:** For databases and other applications that require data consistency, EBS snapshots capture a point-in-time view of the entire volume. This ensures that the data is in a consistent state.
6. **Automated Snapshot Management: Data Lifecycle Policies:** Amazon Data Lifecycle Manager (DLM) can be used to automate the creation, retention, and deletion of EBS snapshots. This helps in managing snapshot lifecycles and reducing manual intervention.
7. **Security: Encryption:** EBS snapshots support encryption using AWS Key Management Service (KMS) keys. This adds an extra layer of security to sensitive data stored in EBS volumes.
8. **Cost Optimization: Incremental Backups:** Since EBS snapshots are incremental, they only store the changes made since the last snapshot. This helps in optimizing storage costs by avoiding redundant data storage.
9. **Custom AMIs (Amazon Machine Images): Image Creation:** EBS snapshots are often used in the process of creating custom Amazon Machine Images. Snapshots provide a reliable way to capture the state of an EBS volume and use it to launch instances with pre-configured data.

**By leveraging EBS snapshots, AWS users can enhance data resilience, streamline development and testing processes, and ensure the availability of critical data in various scenarios.**

## 😊 TO BEGIN WITH THE LAB

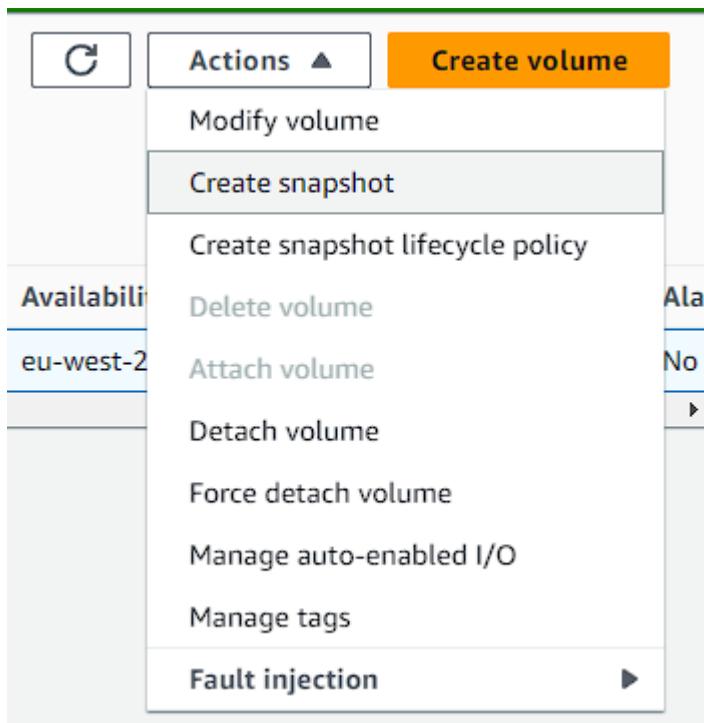
### 😊 STEP 1: CREATE AN EC2 INSTANCE

1. Go to AWS Console and create an EC2 instance based on Linux machine.
2. Create the instance in a different availability zone than your prior operating instances.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
appvm03	i-0566cd07666fc26c0	Running	t2.micro	2/2 checks passed	No alarms	eu-west-2b	ec2-18-130-107-66.eu...	18.130.107.66	-

### 😊 STEP 2: CREATE AN EBS SNAPSHOT

1. To create an EBS Snapshot, go to EBS Volumes.
2. There you will see your 16gb of volume which you have created before and that volume is mounted to another instance.
3. Select that volume then click on actions, then click on create snapshot.



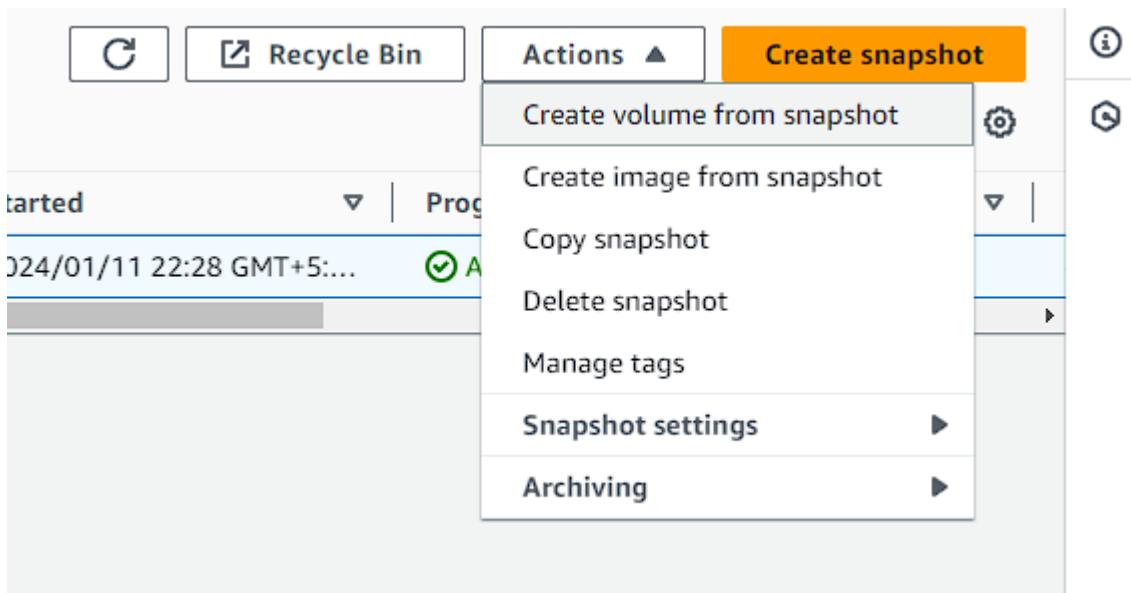
4. While creating a snapshot the only thing you need to do is, give it a name.
5. Then click on create snapshot.

The screenshot shows the 'Details' page for a new snapshot. It includes fields for 'Volume ID' (vol-03614324e8020e0f9 (appvm02-data)), 'Description' (appvm02-snapshot01), and 'Encryption Info' (Not encrypted).

6. Now go to snapshots, there you will see your newly created snapshot.
7. Currently it is in pending state, wait for time to let the progress to get completed.

The screenshot shows the 'Snapshots (1)' list view. The table has columns for Name, Snapshot ID, Volume size, Description, Storage tier, Snapshot status, Started, Progress, and Encryption. The single row shows 'Name: snap-0a88b1238c3b8a3e5', 'Snapshot ID: snap-0a88b1238c3b8a3e5', 'Volume size: 16 GiB', 'Description: appvm02-snapshot01', 'Storage tier: Standard', 'Snapshot status: Pending', 'Started: 2024/01/11 22:28 GMT+5...', 'Progress: Unavailable (39%)', and 'Encryption: Not encrypted'.

8. Once your snapshot is available, then you need to select this snapshot and click on actions.
9. Now from the actions' menu you can create a volume out of this snapshot.
10. So, click on create volume from snapshot.



11. The process is same to create the volume. You need to select a volume type then size is fixed.
12. The only major this here is you need to select the same availability zone as your instance on which you are going to attach this volume.
13. Then click on create volume.

**Volume settings**

Snapshot ID  
 snap-0a88b1238c3b8a3e5

Volume type | [Info](#)

Size (GiB) | [Info](#)  
  
Min: 1 GiB, Max: 16384 GiB. The value must be an integer.

IOPS | [Info](#)  
100 / 3000  
Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS.

Throughput (MiB/s) | [Info](#)  
Not applicable

Availability Zone | [Info](#)

Fast snapshot restore | [Info](#)  
 Not enabled for selected snapshot

14. So, now if you go back to volumes, there you will see your new volume, just rename it and attach it with the instance.

Volumes (1/5) <a href="#">Info</a>											
<a href="#">Actions</a> <a href="#">Create volume</a>											
Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot	Created	Availability Zone	Volume state	Alarm	
appvm03-root	vol-0ca5e8092b0ed044c	gp2	8 GiB	100	-	snap-06005d1...	2024/01/11 22:13 GMT+5....	eu-west-2b	<span style="color: green;">In-use</span>	No ala	
<input checked="" type="checkbox"/> appvm03-data	vol-0ddcd3772061783287	gp2	16 GiB	100	-	snap-0a88bd1...	2024/01/11 22:32 GMT+5....	eu-west-2b	<span style="color: blue;">Available</span>	No ala	
appvm01-root	vol-0ccff0b3d69227b518	gp2	8 GiB	100	-	snap-06005d1...	2024/01/11 14:28 GMT+5....	eu-west-2a	<span style="color: green;">In-use</span>	No ala	
appvm02-data	vol-03614324e8020e0f9	gp2	16 GiB	100	-	-	2024/01/11 15:24 GMT+5....	eu-west-2a	<span style="color: green;">In-use</span>	No ala	
appvm02-root	vol-0a83ff382e81a4e79	gp2	8 GiB	100	-	snap-06005d1...	2024/01/11 19:47 GMT+5....	eu-west-2a	<span style="color: green;">In-use</span>	No ala	



## STEP 3: LOGIN TO INSTANCE AND MOUNT THE VOLUME

- After attaching the volume to your snapshot. Now you have to log in to your instance using putty.

```

ubuntu@ip-172-31-34-100: ~
[1] login as: ubuntu
[2] Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1017-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Thu Jan 11 17:08:57 UTC 2024

 System load:  0.0          Processes:           97
 Usage of /:   20.5% of 7.57GB  Users logged in:    0
 Memory usage: 21%          IPv4 address for eth0: 172.31.34.100
 Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-34-100:~$ 

```

- The steps are same as you did in previous labs.

3. First do a listing of bulk drives. Then create a directory for your volume.
4. After that mount the volume with your directory.

```
ubuntu@ip-172-31-34-100:~$ lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
loop0    7:0    0 24.9M  1 loop /snap/amazon-ssm-agent/7628
loop1    7:1    0 55.7M  1 loop /snap/core18/2812
loop2    7:2    0 63.5M  1 loop /snap/core20/2015
loop3    7:3    0 111.9M 1 loop /snap/lxd/24322
loop4    7:4    0 40.9M  1 loop /snap/snapd/20290
xvda    202:0   0     8G  0 disk
└─xvda1  202:1   0    7.9G 0 part /
└─xvda14 202:14  0     4M 0 part
└─xvda15 202:15  0   106M 0 part /boot/efi
xvdf    202:80   0    16G  0 disk
ubuntu@ip-172-31-34-100:~$ sudo mkdir /data
ubuntu@ip-172-31-34-100:~$ sudo mount /dev/xvdf /data
ubuntu@ip-172-31-34-100:~$ 
```

5. Now go into your directory. There does a listing of files that exist in this volume, because you have created this volume from a snapshot.
6. So, the data which was store into this volume should be in here.

```
ubuntu@ip-172-31-34-100:~$ cd /data
ubuntu@ip-172-31-34-100:/data$ ls
file.txt
ubuntu@ip-172-31-34-100:/data$ more file.txt
This is the data volume
ubuntu@ip-172-31-34-100:/data$ 
```

## **NOW FOR THE CLEAN UP PROCESS.**

**FIRST YOU NEED TO UNMOUNT THIS VOLUME FROM THIS INSTANCE.**

```
ubuntu@ip-172-31-34-100:/data$ sudo umount -l /dev/xvdf
ubuntu@ip-172-31-34-100:/data$ 
```

**AFTER THAT GO BACK TO THE CONSOLE AND DETACH THE VOLUME FROM YOUR INSTANCE. ONCE YOUR VOLUME HAS BEEN DETACHED DELETE IT.**

**THEN DELETE YOUR SNAPSHOT ALSO. AND TERMINATE YOUR INSTANCE.**