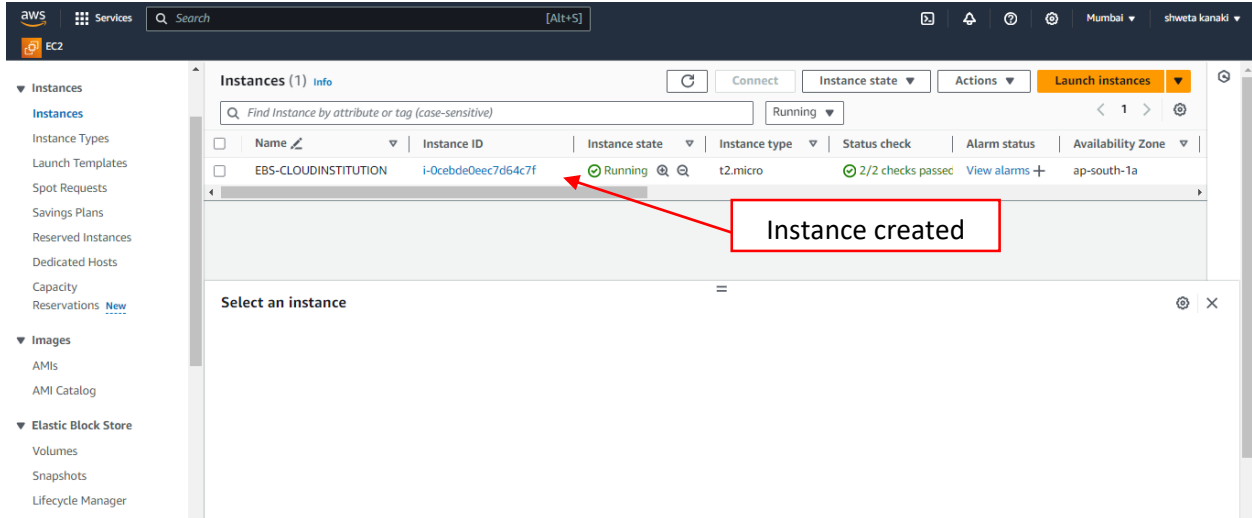


## ATTACH AND MOUNT EBS VOLUME TO LINUX INSTANCE

### Step 1 : Create a EC2 instance

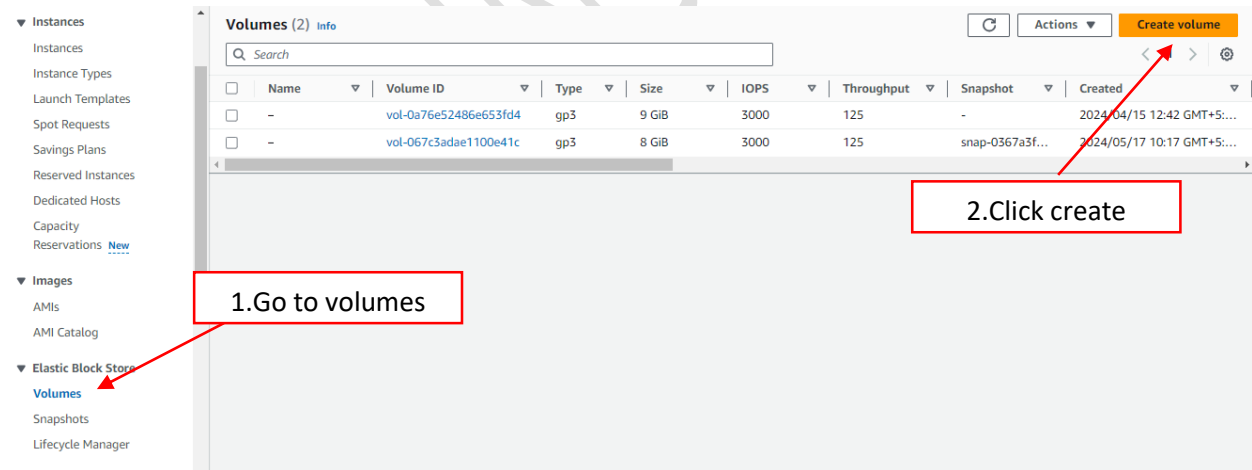


The screenshot shows the AWS Management Console with the 'Instances' page selected. The 'Instances (1)' section is highlighted with a red box. A red arrow points to the 'Running' status of the instance 'EBS-CLOUDINSTITUTION'.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
EBS-CLOUDINSTITUTION	i-0cebd0eec7d64c7f	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a

Instance created

### Step 2 : Create a volume



The screenshot shows the AWS Management Console with the 'Volumes' page selected. The 'Volumes (2)' section is highlighted with a red box. A red arrow points to the 'Create volume' button.

Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot	Created
-	vol-0a76e52486e653fd4	gp3	9 GiB	3000	125	-	2024/04/15 12:42 GMT+5:...
-	vol-067c3adae1100e41c	gp3	8 GiB	3000	125	snap-0367a3f...	2024/05/17 10:17 GMT+5:...

1.Go to volumes

2.Click create

[EC2](#) > [Volumes](#) > Create volume

## Create volume [Info](#)



Create an Amazon EBS volume to attach to any EC2 instance in the same Availability Zone.

### Volume settings

Select volume type

Volume type [Info](#)

General Purpose SSD (gp3)

 General Purpose SSD gp3 is now the default selection. gp3 provides up to 20% lower cost per GB than gp2.  
[Learn More](#) 

Size (GiB) [Info](#)

10

Min: 1 GiB, Max: 16384 GiB. The value must be an integer.

Specify size for your volume

IOPS [Info](#)

3000

Min: 3000 IOPS, Max: 16000 IOPS. The value must be an integer.

Throughput (MiB/s) [Info](#)

Min: 125 MiB, Max: 1000 MiB. Baseline: 125

Select the availability zone

Availability Zone [Info](#)Snapshot ID - optional [Info](#)Encryption [Info](#)

Use Amazon EBS encryption as an encryption solution for your EBS resources associated with your EC2 instances.

☐ Encrypt this volumeTags - optional [Info](#)

A tag 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.

No tags associated with the resource.

You can add 50 more tags.

### Tags - optional [Info](#)

A tag 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.

Key

Value - optional

You can add 49 more tags.

### Snapshot summary [Info](#)



⌚ Click refresh to view backup information

The volume type that you select and the tags that you assign determine whether the volume will be backed up by any Data Lifecycle Manager policies.

Click create

Cancel

Create volume

### Volumes (1/3) [Info](#)



Actions ▾

<input type="checkbox"/>	Name ▾	Volume ID ▾	Type ▾	Size ▾	IOPS ▾	Throughput ▾	Snapshot ▾	Cre
<input type="checkbox"/>	-	vol-0a76e52486e653fd4	gp3	9 GiB	3000	125	-	202
<input type="checkbox"/>	-	vol-067c3adae1100e41c	gp3	8 GiB	3000	125	snap-0367a3f...	202
<input checked="" type="checkbox"/>	-	vol-07fcbe7a6078a878a	gp3	10 GiB	3000	125	-	202

Volume ID: vol-07fcbe7a6078a878a

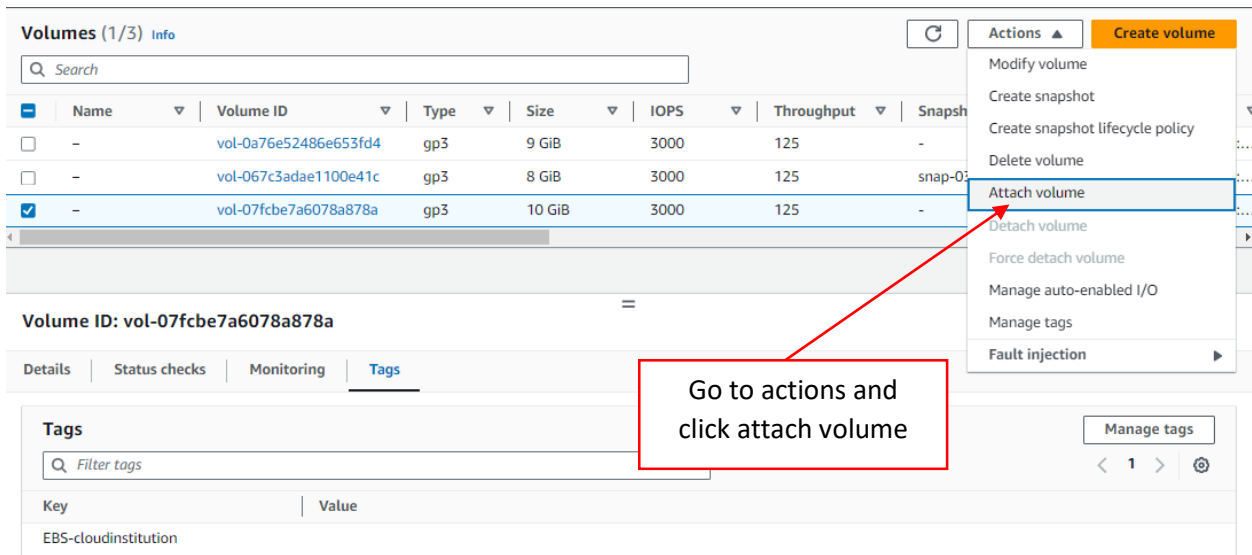
Volume created

Details | Status checks | Monitoring | **Tags**

#### Tags

Key	Value
EBS-cloudinstitution	

### Step 3 : Attach volume to the instance



The screenshot shows the AWS EBS Volumes console. A table lists three volumes. The third volume, **vol-07fcbe7a6078a878a**, is selected. The 'Actions' dropdown menu is open, and 'Attach volume' is highlighted. A red box with the text 'Go to actions and click attach volume' points to the 'Attach volume' option.

	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshots
<input type="checkbox"/>	-	vol-0a76e52486e653fd4	gp3	9 GiB	3000	125	-
<input type="checkbox"/>	-	vol-067c3adae1100e41c	gp3	8 GiB	3000	125	snap-03...
<input checked="" type="checkbox"/>	-	vol-07fcbe7a6078a878a	gp3	10 GiB	3000	125	-

Volume ID: vol-07fcbe7a6078a878a

Details | Status checks | Monitoring | **Tags**

Tags

Filter tags

Key	Value
EBS-cloudinstitution	

Manage tags

[EC2](#) > [Volumes](#) > [vol-07fcbe7a6078a878a](#) > **Attach volume**

## Attach volume Info

Attach a volume to an instance to use it as you would a regular physical hard disk drive.

### Basic details

Volume ID


 vol-07fcbe7a6078a878a

Availability Zone

ap-south-1a

Instance Info

i-0c2442ed6b9b8ff89

 i-0c2442ed6b9b8ff89  
(EBS-cloudinstitution) (running)

Select the created instance

### Basic details

Recommended for data volumes

/dev/sdb	Linux
/dev/sdc	Linux
/dev/sdd	Linux
/dev/sde	Linux
/dev/sdf	Linux <input checked="" type="checkbox"/>
/dev/sdg	Linux
/dev/sdh	Linux

Recommended device names for Linux: /dev/xvda for root volume. /dev/sd[f-p] for data volumes.

ⓘ Newer Linux kernels may rename your devices to **/dev/xvdf** through **/dev/xvdp** internally, even when the device name entered here (and shown in the details) is **/dev/sdf** through **/dev/sdp**.

Cancel
Attach volume

**Instances (1/2)** [Info](#)
Refresh
Connect
Instance state
Actions
Launch instances

All states

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
<input type="checkbox"/>	cloud	i-050dad788ba59373c	Terminated	t2.micro	-	<a href="#">View alarms</a>
<input checked="" type="checkbox"/>	EBS-cloudinstitution	i-0c2442ed6b9b8ff89	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>

**i-0c2442ed6b9b8ff89 (EBS-cloudinstitution)**
Settings
Close

Details
Status and alarms **New**
Monitoring
Security
Networking
**Storage**
Tags

▼ Root device details

Root device name	Root device type	EBS optimization
<input type="text"/> /dev/xvda	EBS	disabled

i-0c2442ed6b9b8ff89 (EBS-cloudinstitution)

/dev/xvda

EBS

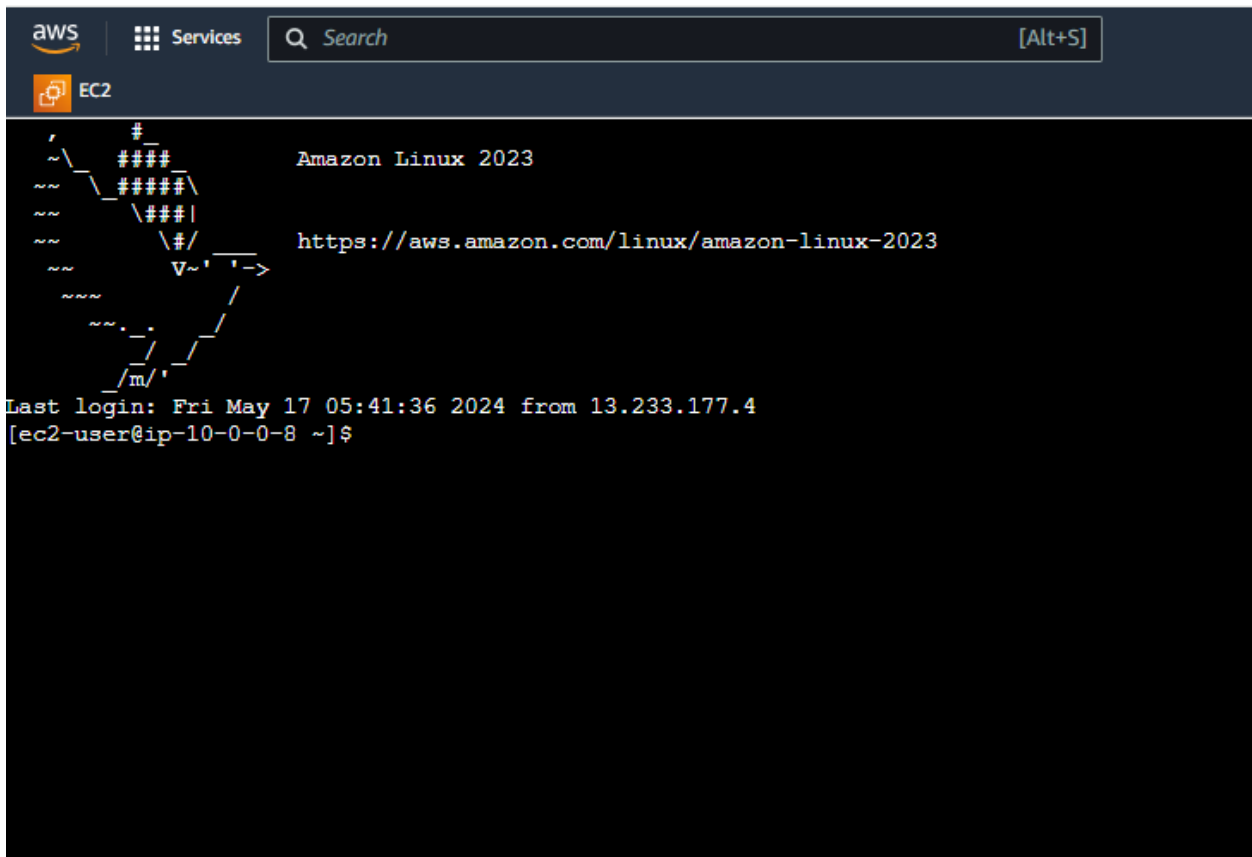
disabled

▼ Block devices

Volume attached to instance

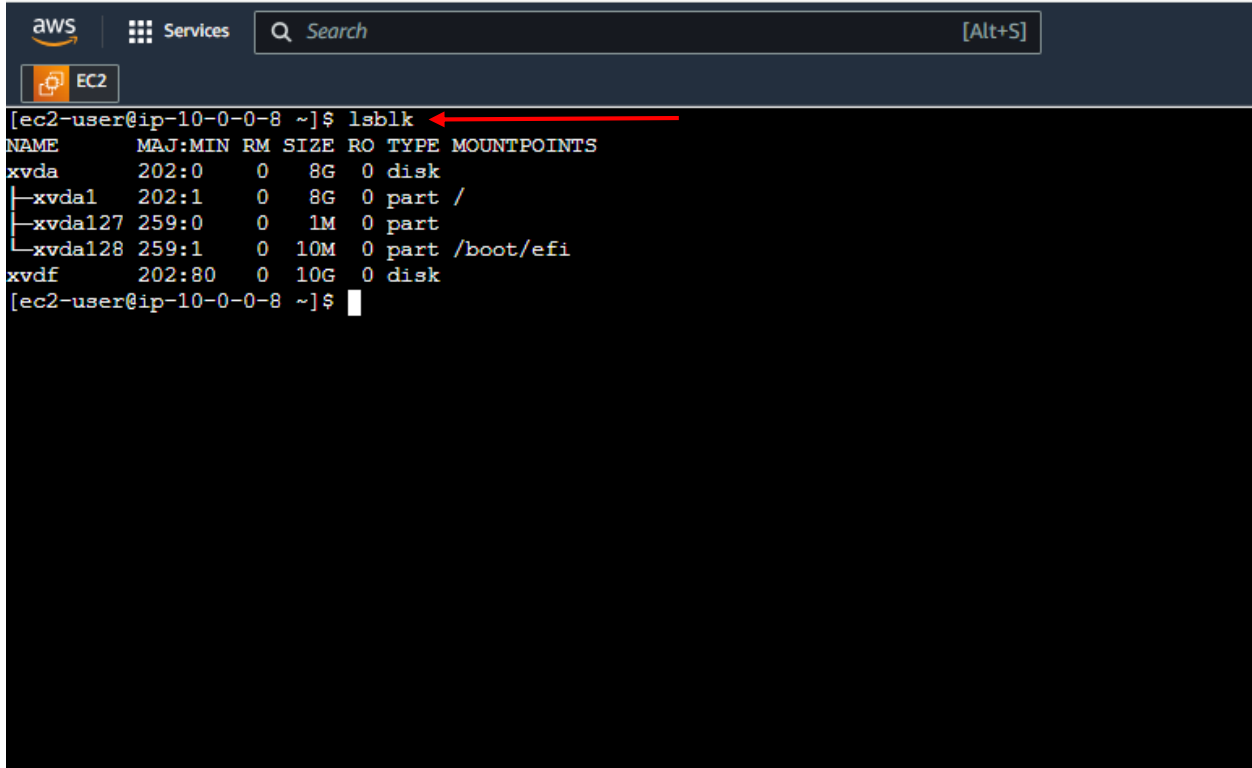
Filter block devices						
<input type="checkbox"/>	Volume ID	Device name	Volume size (GiB)	Attachment status	Attachment time	Encrypted
<input checked="" type="checkbox"/>	<a href="#">vol-067c3adae1100e41c</a>	/dev/xvda	8	✓ Attached	2024/05/17 10:17 GMT+5:30	No
<input type="checkbox"/>	<a href="#">vol-07fcb7a6078a878a</a>	/dev/sdf	10	✓ Attached	2024/05/17 10:49 GMT+5:30	No

Step 4 : Login to linux instance



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, 'Services' link, a search bar, and an '[Alt+S]' shortcut. Below the navigation bar, the 'EC2' icon is selected. The main content area displays a terminal window for an Amazon Linux 2023 instance. The terminal output includes the AWS logo, the text 'Amazon Linux 2023', the URL 'https://aws.amazon.com/linux/amazon-linux-2023', and the login message 'Last login: Fri May 17 05:41:36 2024 from 13.233.177.4'. The prompt is '[ec2-user@ip-10-0-0-8 ~]\$'.

The command **lsblk** (short for "list block devices") is used in Linux to display information about all available block devices.



The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with the AWS logo, a 'Services' menu, a search bar, and an '[Alt+S]' button. Below this, the 'EC2' service is selected. The main area displays a terminal window for an EC2 instance with IP address ip-10-0-0-8. The terminal shows the command `lsblk` being executed, with a red arrow pointing to it. The output is a table of block devices.

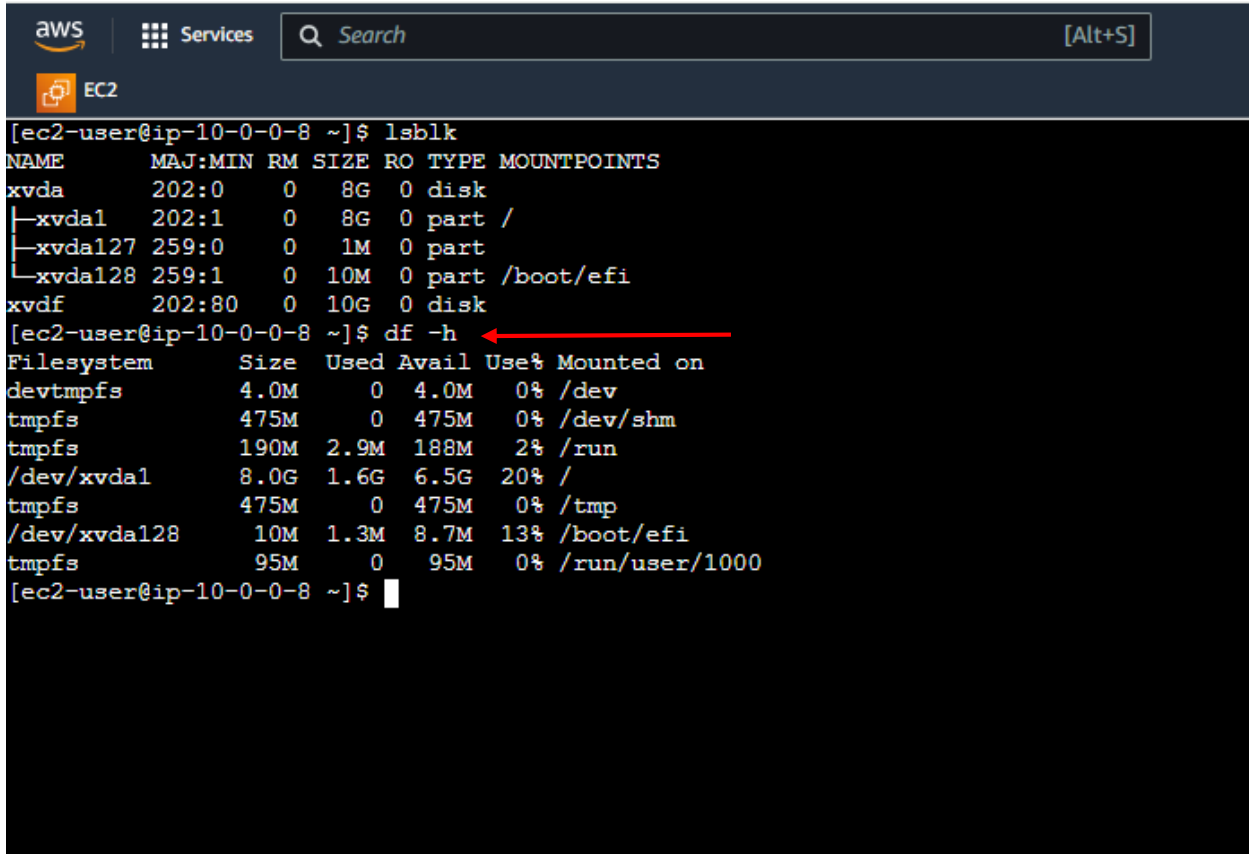
```
[ec2-user@ip-10-0-0-8 ~]$ lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
xvda	202:0	0	8G	0	disk	
_xvda1	202:1	0	8G	0	part	/
_xvda127	259:0	0	1M	0	part	
_xvda128	259:1	0	10M	0	part	/boot/efi
xvdf	202:80	0	10G	0	disk	

```
[ec2-user@ip-10-0-0-8 ~]$
```



The **df -h** command is used in Linux to display information about the disk space usage of all mounted filesystems



```
aws | Services | Search [Alt+S]
EC2
[ec2-user@ip-10-0-0-8 ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda        202:0    0   8G  0 disk
├─xvda1     202:1    0   8G  0 part /
├─xvda127   259:0    0    1M  0 part
└─xvda128   259:1    0   10M  0 part /boot/efi
xvdf        202:80   0  10G  0 disk
[ec2-user@ip-10-0-0-8 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0   4.0M  0% /dev
tmpfs           475M   0   475M  0% /dev/shm
tmpfs           190M  2.9M  188M  2% /run
/dev/xvda1       8.0G  1.6G  6.5G  20% /
tmpfs           475M   0   475M  0% /tmp
/dev/xvda128     10M  1.3M  8.7M  13% /boot/efi
tmpfs           95M   0    95M  0% /run/user/1000
[ec2-user@ip-10-0-0-8 ~]$
```

Create a directory

```
aws | Services | Search [Alt+S]
EC2
[ec2-user@ip-10-0-0-8 ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda        202:0    0   8G  0 disk
├─xvda1     202:1    0   8G  0 part /
├─xvda127   259:0    0   1M  0 part
└─xvda128   259:1    0  10M  0 part /boot/efi
xvdf        202:80   0  10G  0 disk
[ec2-user@ip-10-0-0-8 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M    0  4.0M   0% /dev
tmpfs           475M    0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M    0  475M   0% /tmp
/dev/xvda128    10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M    0   95M   0% /run/user/1000
[ec2-user@ip-10-0-0-8 ~]$ sudo mkdir cloudinstitution
[ec2-user@ip-10-0-0-8 ~]$
```

the **mkfs.ext4** command is used to create an ext4 filesystem on a specified device

```
aws | Services | Search [Alt+S]
EC2
├─xvda1     202:1    0   8G  0 part /
├─xvda127   259:0    0   1M  0 part
└─xvda128   259:1    0  10M  0 part /boot/efi
xvdf        202:80   0  10G  0 disk
[ec2-user@ip-10-0-0-8 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M    0  4.0M   0% /dev
tmpfs           475M    0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M    0  475M   0% /tmp
/dev/xvda128    10M   1.3M   8.7M  13% /boot/efi
tmpfs           95M    0   95M   0% /run/user/1000
[ec2-user@ip-10-0-0-8 ~]$ sudo mkdir cloudinstitution
[ec2-user@ip-10-0-0-8 ~]$ sudo mkfs.ext4 /dev/xvdf
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: d20804c1-6baa-4566-b4d5-a30b59d58bd4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-10-0-0-8 ~]$
```

Now mount the Volume to the directory using `sudo mount /dev/xvdf <dir_name>/` command

```
aws | Services | Search [Alt+S]
EC2
[xvda127 259:0 0 1M 0 part
[xvda128 259:1 0 10M 0 part /boot/efi
xvdf 202:80 0 10G 0 disk
[ec2-user@ip-10-0-0-8 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0  4.0M   0% /dev
tmpfs           475M   0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M   0  475M   0% /tmp
/dev/xvda128    10M  1.3M  8.7M  13% /boot/efi
tmpfs           95M    0   95M   0% /run/user/1000
[ec2-user@ip-10-0-0-8 ~]$ sudo mkdir cloudinstitution
[ec2-user@ip-10-0-0-8 ~]$ sudo mkfs.ext4 /dev/xvdf
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: d20804c1-6baa-4566-b4d5-a30b59d58bd4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-10-0-0-8 ~]$ sudo mount /dev/xvdf cloudinstitution/
[ec2-user@ip-10-0-0-8 ~]$
```

```
aws | Services | Search [Alt+S]
EC2
/dev/xvda128    10M  1.3M  8.7M  13% /boot/efi
tmpfs           95M    0   95M   0% /run/user/1000
[ec2-user@ip-10-0-0-8 ~]$ sudo mkdir cloudinstitution
[ec2-user@ip-10-0-0-8 ~]$ sudo mkfs.ext4 /dev/xvdf
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2621440 4k blocks and 655360 inodes
Filesystem UUID: d20804c1-6baa-4566-b4d5-a30b59d58bd4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-10-0-0-8 ~]$ sudo mount /dev/xvdf cloudinstitution/
[ec2-user@ip-10-0-0-8 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M   0  4.0M   0% /dev
tmpfs           475M   0  475M   0% /dev/shm
tmpfs           190M  2.9M  188M   2% /run
/dev/xvda1      8.0G  1.6G  6.5G  20% /
tmpfs           475M   0  475M   0% /tmp
/dev/xvda128    10M  1.3M  8.7M  13% /boot/efi
tmpfs           95M    0   95M   0% /run/user/1000
/dev/xvdf       9.8G   24K  9.3G   1% /home/ec2-user/cloudinstitution
[ec2-user@ip-10-0-0-8 ~]$
```

Volume is now mounted