
📦 AWS EBS Volume Resize – Step-by-Step

✅ You Can Resize:

- **gp2 / gp3 (General Purpose SSD)**
- **io1 / io2 (Provisioned IOPS SSD)**
- **st1 / sc1 (HDDs)**

> EBS supports **online resizing**—you **don't need to stop the instance** for most Linux/Windows systems.

🛠️ Step-by-Step: Resize an EBS Volume

🔹 Step 1: Modify the EBS Volume

Use AWS Console or CLI.

📌 CLI Command:

```
aws ec2 modify-volume \
  --volume-id vol-xxxxxxxxxxxxxxxxx \
  --size 100 \
  --region <your-region>
```

You can also modify IOPS or volume type at the same time:

```
aws ec2 modify-volume \
  --volume-id vol-xxxxxxxxxxxxxxxxx \
  --size 100 \
  --volume-type gp3 \
  --iops 3000 \
```

```
  --throughput 125
```

✅ No downtime required. AWS performs resizing in the background.

🔹 **Step 2: Wait for Modification to Complete**

```
aws ec2 describe-volumes-modifications \
  --volume-ids vol-xxxxxxxxxxxxxxxxx
```

Wait until the state is completed.

🔹 **Step 3: Resize the Partition (Linux)**

3.1. Check current partitions

```
lsblk
```

3.2. Grow the partition (if using GPT with growpart):

```
sudo growpart /dev/xvdf 1
```

> Replace /dev/xvdf with your volume device.

🔹 **Step 4: Resize the Filesystem**

For ext4:

```
sudo resize2fs /dev/xvdf1
```

For xfs:

```
sudo xfs_growfs /dev/xvda1
```

If using XFS, be sure you're resizing the **mounted root** or target mount point.

📌 Important Notes

Point	Description
Online	Resizing is **online** for most modern Linux distros (Ubuntu, Amazon Linux 2, RHEL 7+, etc.)
Shrinking	You **cannot** shrink EBS volumes. You must snapshot, create a new smaller volume
Root Volume	Resizing root volumes is supported, just be cautious with partition layout
File System	If your filesystem isn't expanded, even a bigger volume won't use more space

---

🧪 Example: Resize EBS Root Volume (Ubuntu)

```
# Modify volume in AWS or CLI (as above)

# On instance:
sudo growpart /dev/nvme0n1 1
sudo resize2fs /dev/nvme0n1p1
```

📄 (Optional) Automate with CloudWatch + Lambda

You can monitor volume usage and trigger **auto-resize** via a Lambda script based on thresholds.
