

Resolving Disk Full Issue on EC2 by Expanding the Root Volume

Problem Statement:

On an Ubuntu EC2 instance, the root filesystem (/) ran out of disk space. This caused:

- Package installations to fail (E: dpkg was interrupted)
- System updates to break
- Potential service instability

Objective:

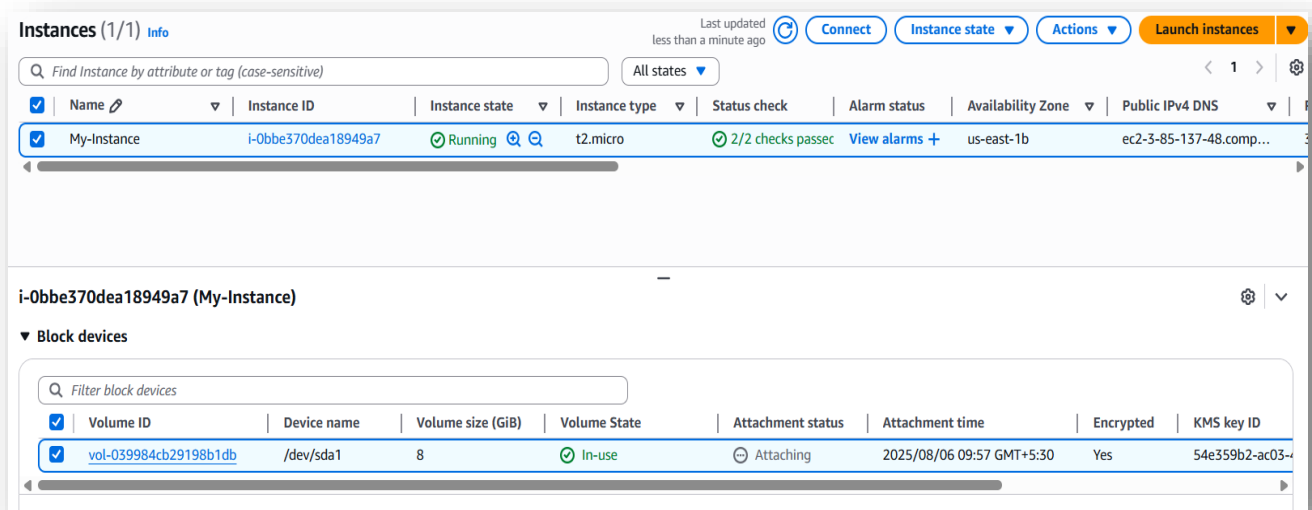
To fix the disk full issue by expanding the root EBS volume from its original size (e.g., 8 GB) to a larger size (e.g., 20 GB) without needing to create a new volume or reboot the instance.

Step-by-Step Guide:

Step1:

- Launch a New Ec2 ubuntu Instance.

Note: By default, most EC2 instances are launched with a root EBS volume of 8 GB.



Step2:

- Connect Instance via SSH and check the default volume and usage.

www.linkedin.com/in/arun-kumar-akula

- Verify Disk space using **df -h** command

```
ubuntu@ip-172-31-27-208:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        6.8G  1.8G  5.0G  26% /
tmpfs            479M   0  479M   0% /dev/shm
tmpfs            192M  868K  191M   1% /run
tmpfs            5.0M   0   5.0M   0% /run/lock
/dev/xvda16      881M   86M  734M  11% /boot
/dev/xvda15      105M   6.2M   99M   6% /boot/efi
tmpfs            96M   12K   96M   1% /run/user/1000
ubuntu@ip-172-31-27-208:~$
```

Here we can observe that most of the root space is free, with approximately 5 GB available.

- Here, I am going to install packages such as Jenkins and Docker to monitor disk usage and determine whether additional volume needs to be added.

```
ubuntu@ip-172-31-27-208:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        6.8G  6.7G   55M 100% /
tmpfs            479M   0  479M   0% /dev/shm
tmpfs            192M  888K  191M   1% /run
tmpfs            5.0M   0   5.0M   0% /run/lock
/dev/xvda16      881M   86M  734M  11% /boot
/dev/xvda15      105M   6.2M   99M   6% /boot/efi
tmpfs            96M   12K   96M   1% /run/user/1000
ubuntu@ip-172-31-27-208:~$
```

Now we can observe that after installing the packages, only 55 MB of storage is available, which is not sufficient for further operations.

Step3: Modify Volume in AWS Console.

- Go to EC2 → Volumes
- ind your root EBS volume
- Click Actions → Modify Volume

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

Saved filter sets
Choose filter set

Volume ID = vol-039984cb29198b1db

<input checked="" type="checkbox"/>	Name	Volume ID	Type	Size	IOPS	Throughput	Snapshot ID
<input checked="" type="checkbox"/>		vol-039984cb29198b1db	gp3	8 GiB	3000	125	snap-0bc1d350c2ac74

Last updated less than a minute ago

- Modify volume
- Create snapshot
- Create snapshot lifecycle policy
- Delete volume
- Attach volume
- Detach volume
- Force detach volume
- Manage auto-enabled I/O
- Manage tags
- Fault injection

Availability Zone: us-east-1b

- Change size from 8 GiB to 20 GiB.

Modify volume [Info](#)

Modify the type, size, and performance of an EBS volume.

Volume details

Volume ID
vol-039984cb29198b1db

Volume type | [Info](#)
General Purpose SSD (gp3)

Size (GiB) | [Info](#)
20
Min: 1 GiB, Max: 16384 GiB.

IOPS | [Info](#)
3000
Min: 3000 IOPS, Max: 16000 IOPS.

Throughput (MiB/s) | [Info](#)
125
Min: 125 MiB, Max: 1000 MiB. Baseline: 125 MiB/s.

After modifying the volume, we can see that the EC2 instance's root volume has increased from 8 GB to 20 GB. Now, we can expand the /root (or /) filesystem to utilize the full 20 GB.

Instances (1/1) [Info](#)

Find Instance by attribute or tag (case-sensitive) All states

Last updated less than a minute ago

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
<input checked="" type="checkbox"/>	My-Instance	i-0bbe370dea18949a7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-3-85-137-48.comp...

i-0bbe370dea18949a7 (My-Instance)

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

Root device details

Root device name /dev/sda1	Root device type EBS	EBS optimization disabled
-------------------------------	-------------------------	------------------------------

Block devices

Filter block devices

<input checked="" type="checkbox"/>	Volume ID	Device name	Volume size (GiB)	Volume State	Attachment status	Attachment time	Encrypted	KMS key ID
<input checked="" type="checkbox"/>	vol-039984cb29198b1db	/dev/sda1	20	In-use	Attached	2025/08/06 09:57 GMT+5:30	Yes	54e359b2-ac03-

Step4: Install Required packages and Resize the Volume.

```
sudo apt update
```

```
sudo apt install cloud-guest-utils
```

```
ubuntu@ip-172-31-27-208:~$  
ubuntu@ip-172-31-27-208:~$ sudo apt install cloud-guest-utils  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
cloud-guest-utils is already the newest version (0.33-1).  
cloud-guest-utils set to manually installed.  
You might want to run 'apt --fix-broken install' to correct these.  
The following packages have unmet dependencies:  
python3-pyqt5.qtchart : Depends: python3-pyqt5 (>= 5.15.4) but it is not going to be installed  
python3-pyqt5.qtsvg : Depends: python3-pyqt5 (= 5.15.10+dfsg-1build6) but it is not going to be installed
```

Installs growpart, which is needed to expand the root partition after increasing the EC2 volume size.

- **Expand the Partition**

```
ubuntu@ip-172-31-27-208:~$  
ubuntu@ip-172-31-27-208:~$ lsblk  
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS  
loop0        7:0    0   27.2M  1 loop /snap/amazon-ssm-agent/11320  
loop1        7:1    0   73.9M  1 loop /snap/core22/1981  
loop2        7:2    0   50.9M  1 loop /snap/snapd/24505  
loop3        7:3    0   63.8M  1 loop /snap/core20/2599  
loop4        7:4    0  401.5M  1 loop /snap/blender/6438  
xvda        202:0    0    20G  0 disk  
├─xvda1     202:1    0     7G  0 part /  
├─xvda14    202:14   0     4M  0 part  
├─xvda15    202:15   0   106M  0 part /boot/efi  
└─xvda16    259:0    0    913M  0 part /boot  
ubuntu@ip-172-31-27-208:~$ sudo growpart /dev/xvda 1  
CHANGED: partition=1 start=2099200 old: size=14677983 end=16777182 new: size=39843807 end=41943006  
ubuntu@ip-172-31-27-208:~$  
ubuntu@ip-172-31-27-208:~$ sudo resize2fs /dev/xvda1  
resize2fs 1.47.0 (5-Feb-2023)  
Filesystem at /dev/xvda1 is mounted on /; on-line resizing required  
old_desc_blocks = 1, new_desc_blocks = 3  
The filesystem on /dev/xvda1 is now 4980475 (4k) blocks long.
```

```
sudo growpart /dev/xvda 1 (This expands partition 1 (xvda1) to use the full disk space (20 GB).)
```

```
sudo resize2fs /dev/xvda1 ( This resizes the ext4 filesystem to match the new partition size.)
```

After completing the following steps, the expanded volume will be attached and available under the root (/) filesystem.

- Verify Final output using **df -h**

```
ubuntu@ip-172-31-27-208:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       19G   5.0G   14G   28% /
tmpfs           479M    0   479M    0% /dev/shm
tmpfs           192M  916K   191M    1% /run
tmpfs           5.0M    0   5.0M    0% /run/lock
/dev/xvda16      881M   86M   734M   11% /boot
/dev/xvda15     105M   6.2M    99M    6% /boot/efi
tmpfs           96M   12K    96M    1% /run/user/1000
ubuntu@ip-172-31-27-208:~$ |
```
