

# AWS EC2 实例磁盘空间扩容

[amazon-web-services](#) [ec2](#) 发布于 2019-01-02

## AWS Linux 实例磁盘空间扩容

大致步骤:

1. 调整卷大小
2. 调整分区大小
3. 调整文件系统大小

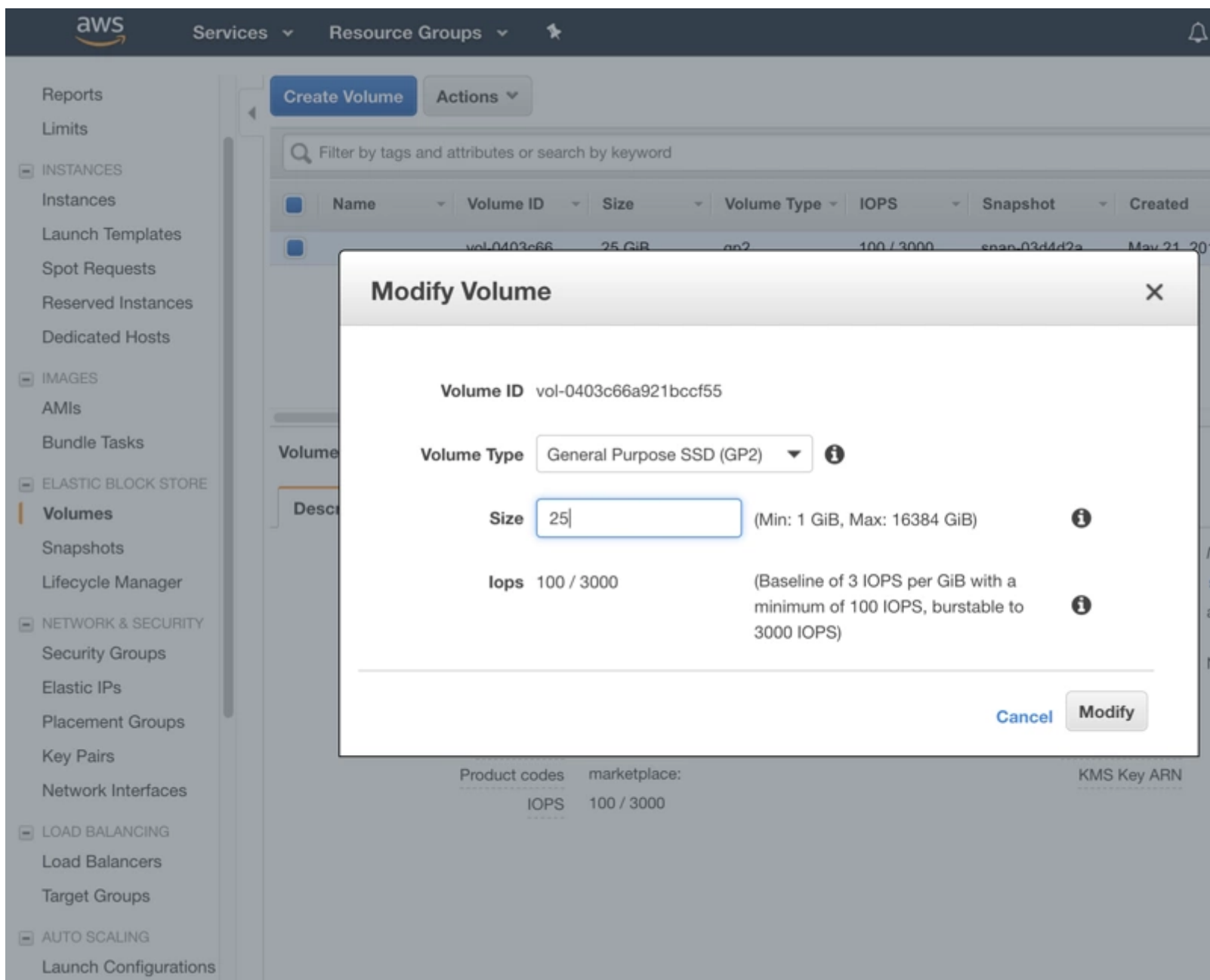
参考文章: [通过给EBS类型的根设备扩容解决AWS磁盘容量已满](#)

### 1. 调整EBS存储卷大小

进入控制台, 选择 EBS - Volumes -Action - Modify Volume

The screenshot displays the AWS Management Console interface. On the left, the navigation pane shows the 'Elastic Block Store' section with 'Volumes' selected. The main content area shows a list of volumes. A context menu is open over a volume, listing actions such as 'Modify Volume', 'Create Snapshot', 'Delete Volume', 'Attach Volume', 'Detach Volume', 'Force Detach Volume', 'Change Auto-Enable IO Setting', and 'Add/Edit Tags'. Below the volume list, the details for a specific volume (vol-0403c66a921bccf55) are shown, including its ID, size (25 GiB), creation time, state, and attachment information.

Volume ID	Size	Created	State	Attachment information	Volume type	Product codes	IOPS
vol-0403c66a921bccf55	25 GiB	May 21, 2018 at 4:32:05 PM UTC+12	in-use - completed (100%)	/dev/sda1 (attached)	gp2	marketplace:	100 / 3000



## 2. 登录实例调整分区

首先使用`lsblk`可以查看附加到实例上的所有存储卷的真实size以及分区情况

```
ubuntu@ip-172-31-8-121:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda         202:0    0   25G  0 disk
└─xvda1      202:1    0    8G  0 part /
```

存储卷已经调整为25G

其上只有1个分区【1】显示为 xvda1

并没有占用全部空间

下面调整分区【1】使其占用全部卷可用空间

`sudo growpart /dev/xvda 1`

```
ubuntu@ip-172-31-8-121:~$ sudo growpart /dev/xvda 1
CHANGED: partition=1 start=2048 old: size=16775135 end=16777183 new: size=52426719,end=52428767
```

分区大小跟EBS大小一样了

```
ubuntu@ip-172-31-8-121:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda         202:0    0   25G  0 disk
└─xvda1      202:1    0   25G  0 part /
```

但文件系统空间还没有识别到这部分新增的空间

```
ubuntu@ip-172-31-8-121:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            488M   0    488M   0% /dev
tmpfs           100M  11M   89M   11% /run
/dev/xvda1       7.7G  4.4G  3.4G   57% /
tmpfs           496M   0    496M   0% /dev/shm
tmpfs           5.0M   0    5.0M   0% /run/lock
tmpfs           496M   0    496M   0% /sys/fs/cgroup
tmpfs           100M   0    100M   0% /run/user/1000
```

### 3. 调整实例的文件系统进行扩容：

```
sudo resize2fs /dev/xvda1
```

```
ubuntu@ip-172-31-8-121:~$ sudo resize2fs /dev/xvda1
resize2fs 1.42.13 (17-May-2015)
Filesystem at /dev/xvda1 is mounted on /; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 2
The filesystem on /dev/xvda1 is now 6553339 (4k) blocks long.

ubuntu@ip-172-31-8-121:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            488M   0    488M   0% /dev
tmpfs           100M  11M   89M   11% /run
/dev/xvda1      25G   4.4G   20G   18% /
tmpfs           496M   0    496M   0% /dev/shm
tmpfs           5.0M   0     5.0M   0% /run/lock
tmpfs           496M   0    496M   0% /sys/fs/cgroup
tmpfs           100M   0    100M   0% /run/user/1000
```

Size变为25G， Done

注：

- 挂载在根目录上的是EBS存储卷上的分区而不是这个卷，而且卷被扩容但是分区大小不变依旧不能扩容
- xvda是一个存储卷，xvda1是存储卷上的一个分区，但是发现并不是一个存储卷被挂载到根而是存储卷上的一个分区被挂在跟上
- EBS 存储卷视为容器，则分区是卷中的另一个容器，并且数据位于分区上。
- EBS 相当于一块外部磁盘，如果实例损坏，无法启动可用挂载到别到实例上读取数据。

阅读 4.2k · 发布于 2019-01-02

👍 赞

🔖 收藏

🔗 分享

本作品系 原创， 采用《署名-非商业性使用-禁止演绎 4.0 国际》许可协议



思考的虫子

🔖 13

关注作者