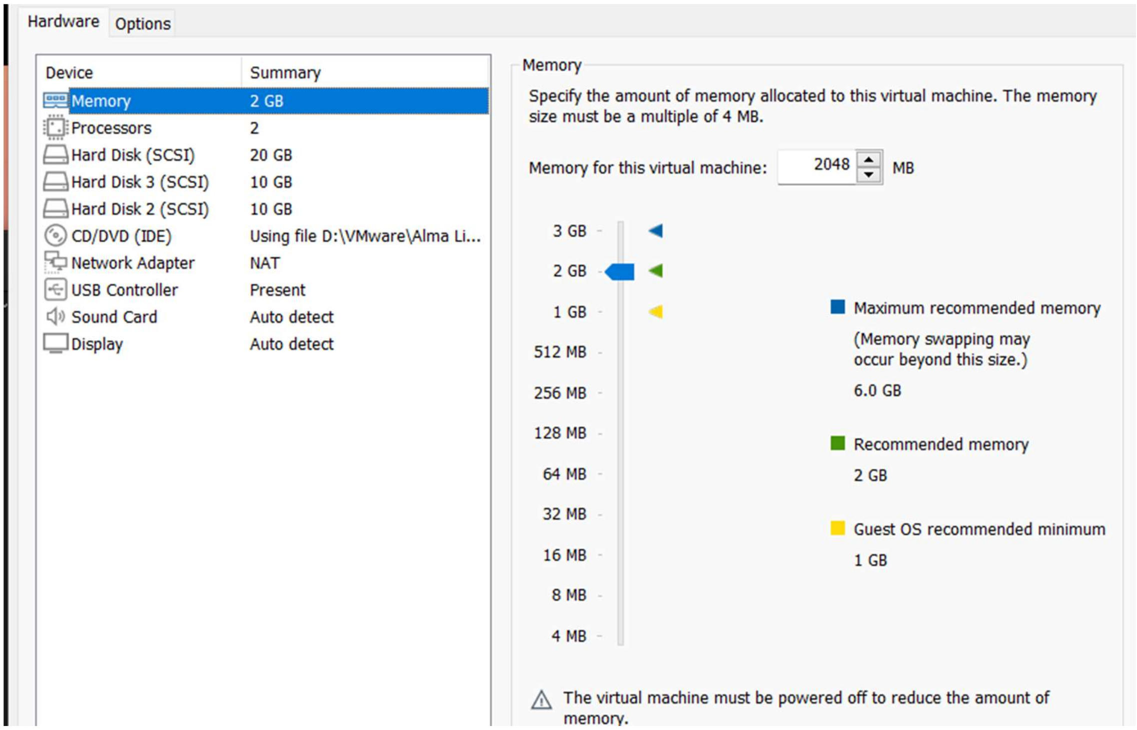
LVM

LVM (Logical Volume Manager) in Linux is a system for managing disk storage that allows for flexible and dynamic management of disk space.

It enables the creation of logical volumes, which abstract the physical storage, allowing you to resize, move, or combine disks without being limited by the physical structure.



Device	Summary
Memory	2 GB
Processors	2
Hard Disk (SCSI)	20 GB
Hard Disk 3 (SCSI)	10 GB
Hard Disk 2 (SCSI)	10 GB
CD/DVD (IDE)	Using file D:\VMware\Alma Li...
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: 2048 MB

🟦 Maximum recommended memory
(Memory swapping may occur beyond this size.)
6.0 GB

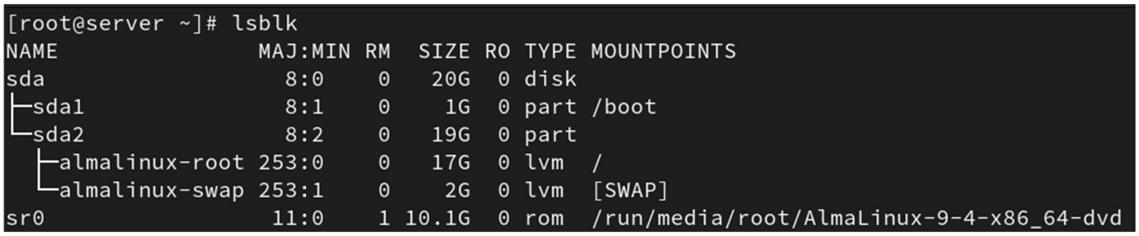
🟩 Recommended memory
2 GB

🟨 Guest OS recommended minimum
1 GB

⚠ The virtual machine must be powered off to reduce the amount of memory.

Checking whether the disk is added or not

By lsblk command:

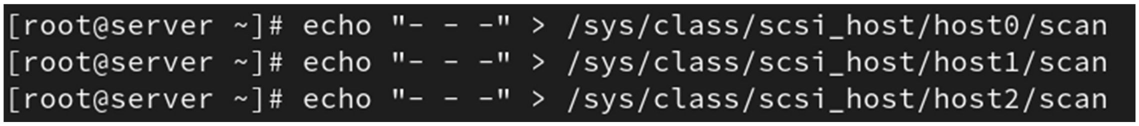


```
[root@server ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0   20G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   19G  0 part
  ├─almalinux-root 253:0    0   17G  0 lvm  /
  └─almalinux-swap 253:1    0    2G  0 lvm  [SWAP]
sr0                 11:0    1 10.1G  0 rom  /run/media/root/AlmaLinux-9-4-x86_64-dvd
```

Its not added because we didn’t restarted the system for that we use echo commad and scanning the host echo “- - -” > /sys/class/scsi_host/host0/scan

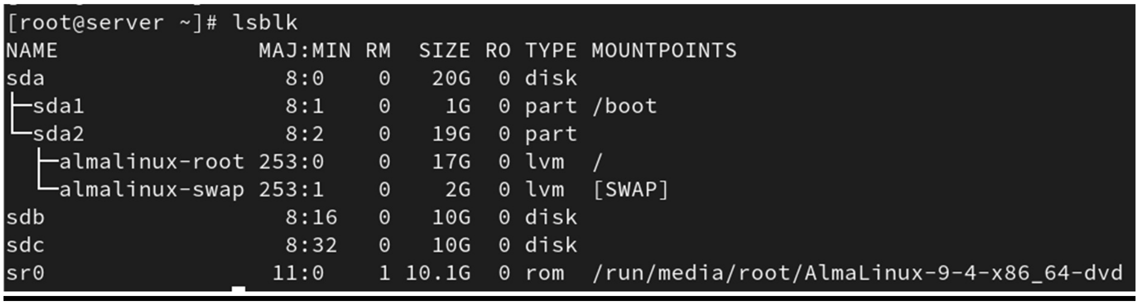
echo “- - -” > /sys/class/scsi_host/host1/scan

echo “- - -” > /sys/class/scsi_host/host2/scan



```
[root@server ~]# echo "- - -" > /sys/class/scsi_host/host0/scan
[root@server ~]# echo "- - -" > /sys/class/scsi_host/host1/scan
[root@server ~]# echo "- - -" > /sys/class/scsi_host/host2/scan
```

Now checking with lsblk command

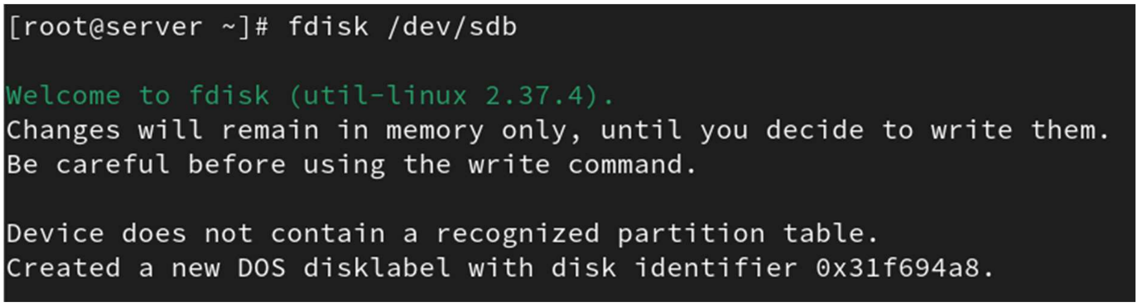


```
[root@server ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0   20G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   19G  0 part
  ├─almalinux-root 253:0    0   17G  0 lvm  /
  └─almalinux-swap 253:1    0    2G  0 lvm  [SWAP]
sdb                  8:16   0   10G  0 disk
sdc                  8:32   0   10G  0 disk
sr0                 11:0    1 10.1G  0 rom  /run/media/root/AlmaLinux-9-4-x86_64-dvd
```

Yes the hard disks where added.

Now partitioning the disks

For that we use fdisk command

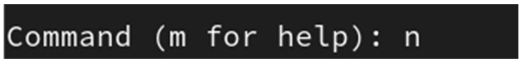


```
[root@server ~]# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x31f694a8.
```

For creating new partition we have to give n option in fdisk



```
Command (m for help): n
```

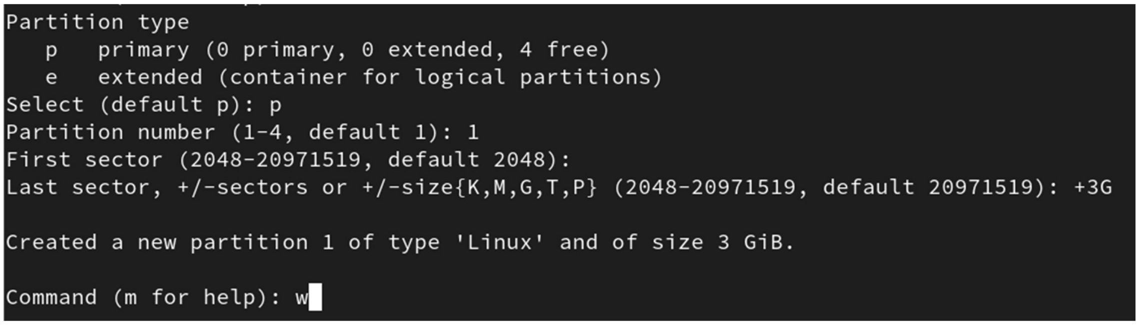
It will ask partition type primary or extended give default primary

And then partition number

First sector default

Last sector – how many gb storage that you want to patrtiotion

Finally give w to save the partition



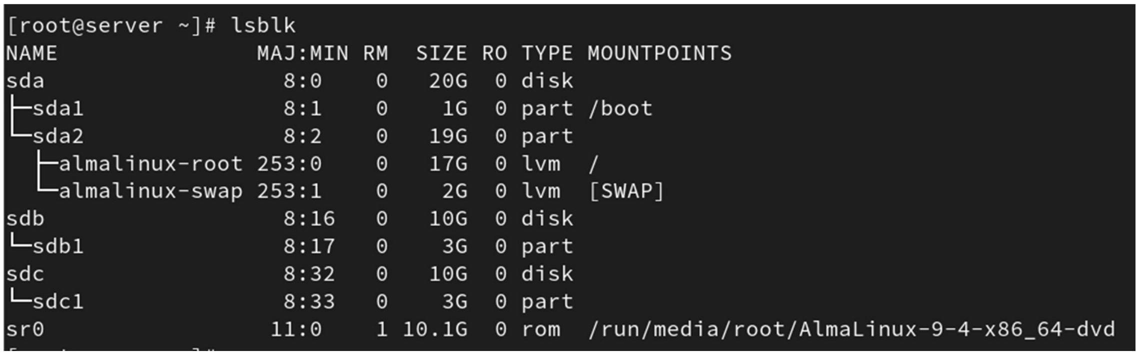
```
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-20971519, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-20971519, default 20971519): +3G

Created a new partition 1 of type 'Linux' and of size 3 GiB.

Command (m for help): w
```

Do the steps for the another disk also sdc

Check with lsblk command



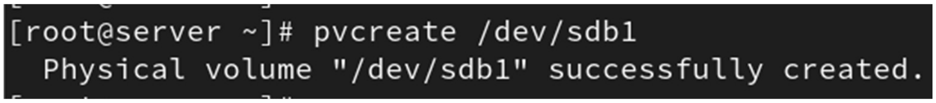
```
[root@server ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0   20G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   19G  0 part
  ├─almalinux-root 253:0    0   17G  0 lvm  /
  └─almalinux-swap 253:1    0    2G  0 lvm  [SWAP]
sdb                  8:16   0   10G  0 disk
└─sdb1               8:17   0    3G  0 part
sdc                  8:32   0   10G  0 disk
└─sdc1               8:33   0    3G  0 part
sr0                 11:0    1 10.1G  0 rom  /run/media/root/AlmaLinux-9-4-x86_64-dvd
```

Now the partition had created

CREATING PHYSICAL VOLUME

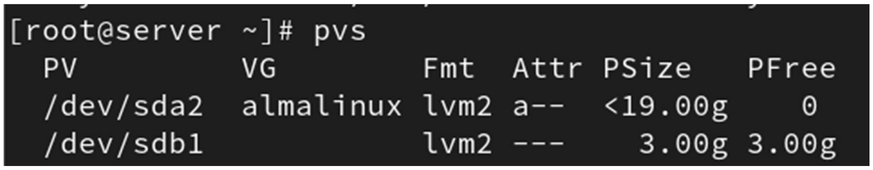
pvcreate /dev/sdb1

pvcreate /dev/sdc1



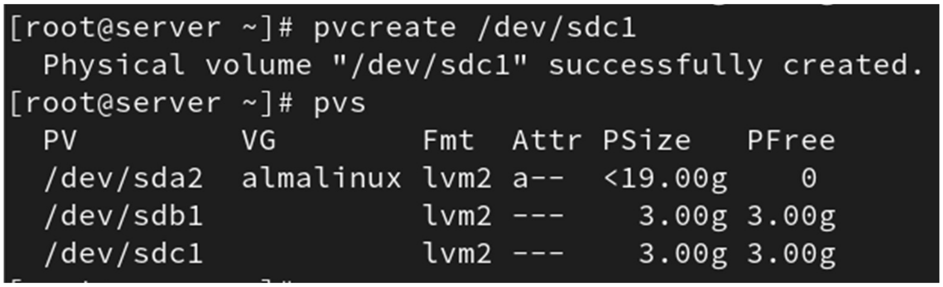
```
[root@server ~]# pvcreate /dev/sdb1
  Physical volume "/dev/sdb1" successfully created.
```

Checking whether the physical volume created or not by pvs command



```
[root@server ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  almalinux lvm2 a--  <19.00g    0
  /dev/sdb1            lvm2 ---    3.00g 3.00g
```

Now creating physical volume for another disk partition and checking pv

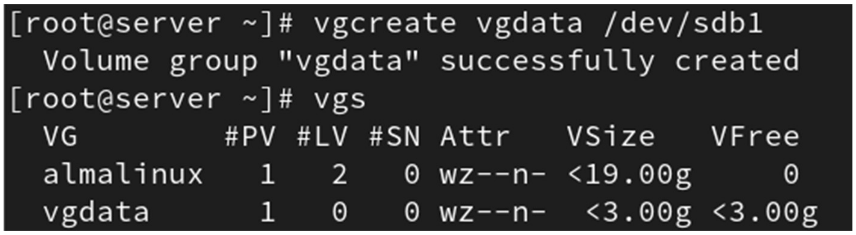


```
[root@server ~]# pvcreate /dev/sdc1
  Physical volume "/dev/sdc1" successfully created.
[root@server ~]# pvs
  PV         VG        Fmt  Attr PSize   PFree
  /dev/sda2  almalinux lvm2 a--  <19.00g    0
  /dev/sdb1            lvm2 ---    3.00g 3.00g
  /dev/sdc1            lvm2 ---    3.00g 3.00g
```

CREATING Volume Group

vgcreate vgdata /dev/sdb1

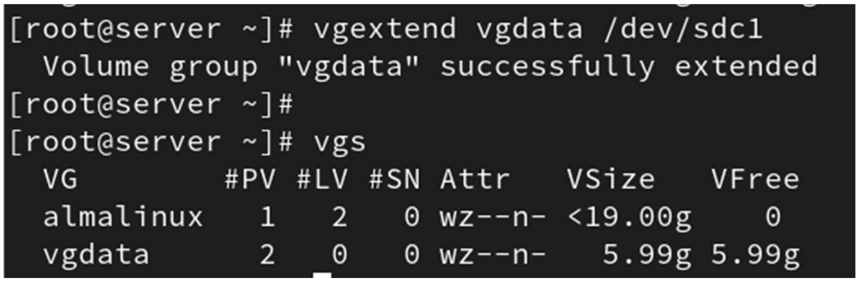
and checking the volume group by vgs command



```
[root@server ~]# vgcreate vgdata /dev/sdb1
  Volume group "vgdata" successfully created
[root@server ~]# vgs
  VG        #PV #LV #SN Attr   VSize   VFree
  almalinux   1   2   0 wz--n- <19.00g     0
  vgdata      1   0   0 wz--n-  <3.00g <3.00g
```

For Extending the VOLUME GROUP we use vgextend command

vgextend vgdata /dev/sdc1



```
[root@server ~]# vgextend vgdata /dev/sdc1
  Volume group "vgdata" successfully extended
[root@server ~]#
[root@server ~]# vgs
  VG        #PV #LV #SN Attr   VSize   VFree
  almalinux   1   2   0 wz--n- <19.00g    0
  vgdata      2   0   0 wz--n-   5.99g 5.99g
```

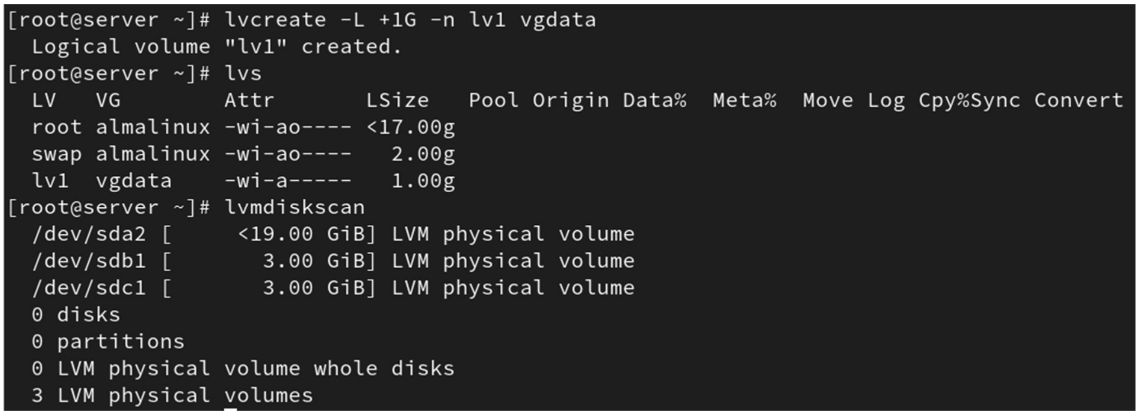
CREATING LOGICAL VOLUME

lvcreate -L +1G -n lv1 vgdata

lvcreate -l +100%FREE -n lv2 vgdata

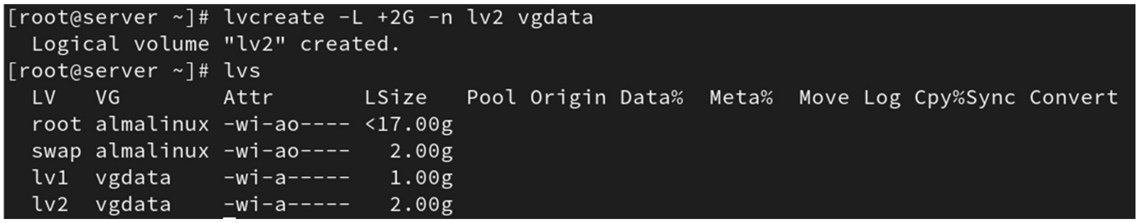
lvs

lvmdiskscan



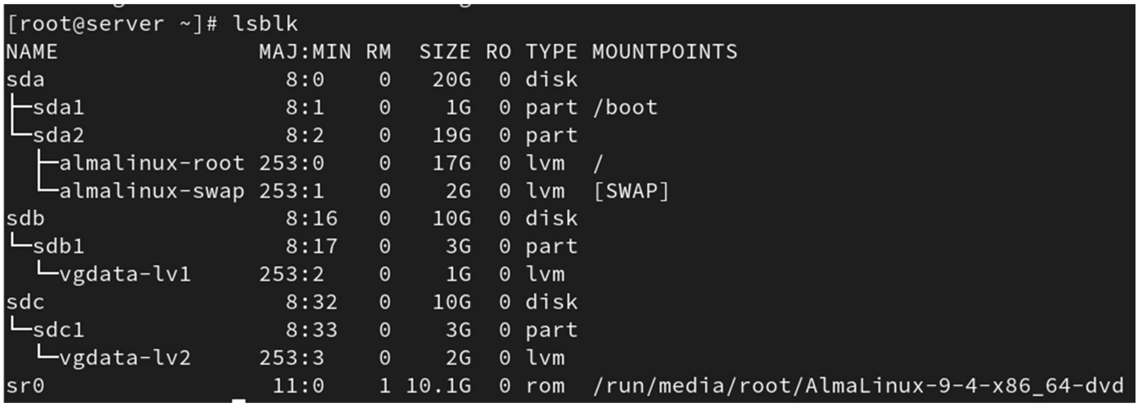
```
[root@server ~]# lvcreate -L +1G -n lv1 vgdata
  Logical volume "lv1" created.
[root@server ~]# lvs
  LV   VG        Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root almalinux -wi-ao---- <17.00g
  swap almalinux -wi-ao----   2.00g
  lv1  vgdata    -wi-a-----   1.00g
[root@server ~]# lvmdiskscan
  /dev/sda2 [     <19.00 GiB] LVM physical volume
  /dev/sdb1 [       3.00 GiB] LVM physical volume
  /dev/sdc1 [       3.00 GiB] LVM physical volume
  0 disks
  0 partitions
  0 LVM physical volume whole disks
  3 LVM physical volumes
```

creating one more lv named lv2



```
[root@server ~]# lvcreate -L +2G -n lv2 vgdata
  Logical volume "lv2" created.
[root@server ~]# lvs
  LV   VG        Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root almalinux -wi-ao---- <17.00g
  swap almalinux -wi-ao----   2.00g
  lv1  vgdata    -wi-a-----   1.00g
  lv2  vgdata    -wi-a-----   2.00g
```

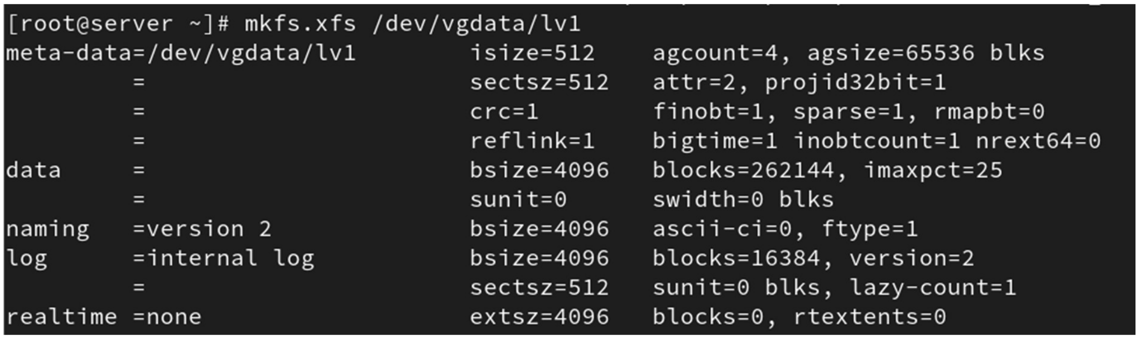
Checking with lsblk



```
[root@server ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                   8:0    0   20G  0 disk
├─sda1                8:1    0    1G  0 part /boot
└─sda2                8:2    0   19G  0 part
  ├─almalinux-root 253:0    0   17G  0 lvm  /
  └─almalinux-swap 253:1    0    2G  0 lvm  [SWAP]
sdb                   8:16   0   10G  0 disk
└─sdb1                8:17   0    3G  0 part
  └─vgdata-lv1     253:2    0    1G  0 lvm
sdc                   8:32   0   10G  0 disk
└─sdc1                8:33   0    3G  0 part
  └─vgdata-lv2     253:3    0    2G  0 lvm
sr0                  11:0    1 10.1G  0 rom  /run/media/root/AlmaLinux-9-4-x86_64-dvd
```

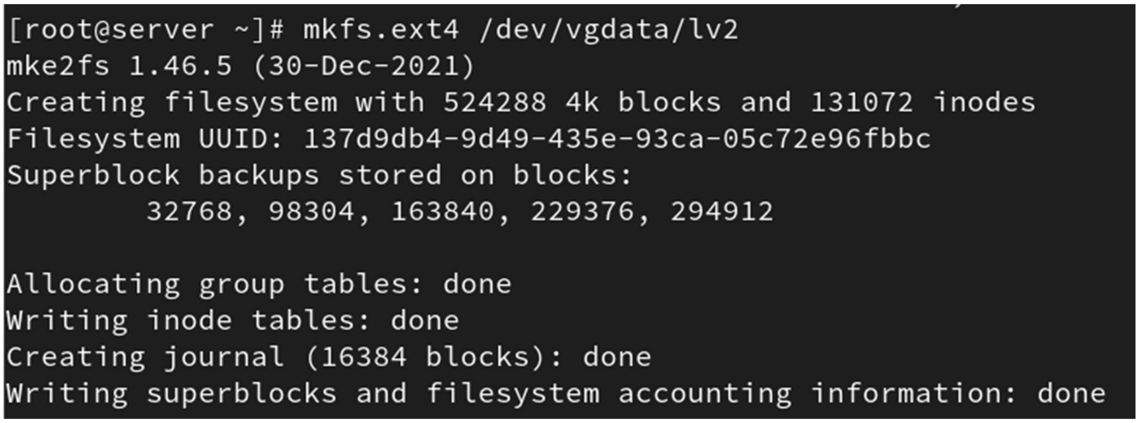
CREATING FILE SYSTEM for both Logical Volume one as xfs and another one as ext4 filesystem type

mkfs .xfs /dev/vgdata/lv1



```
[root@server ~]# mkfs.xfs /dev/vgdata/lv1
meta-data=/dev/vgdata/lv1        isize=512    agcount=4, agsize=65536 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
data     =                       bsize=4096   blocks=262144, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

mkfs.ext4 /dev/vgdata/lv2

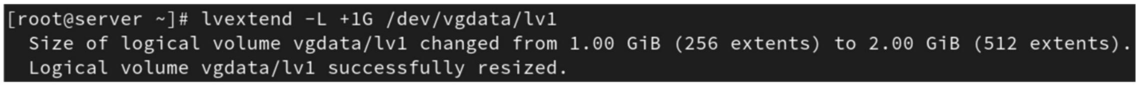


```
[root@server ~]# mkfs.ext4 /dev/vgdata/lv2
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 524288 4k blocks and 131072 inodes
Filesystem UUID: 137d9db4-9d49-435e-93ca-05c72e96fbbc
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

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

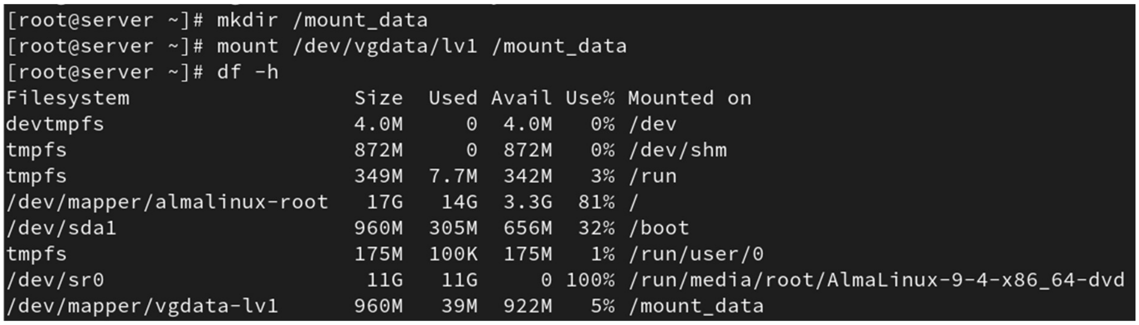
Extending the LOGICAL VOLUME

lvextend -L +1G /dev/vgdata/lv1



```
[root@server ~]# lvextend -L +1G /dev/vgdata/lv1
  Size of logical volume vgdata/lv1 changed from 1.00 GiB (256 extents) to 2.00 GiB (512 extents).
  Logical volume vgdata/lv1 successfully resized.
```

After extending the logical volume mount the lv with mount point:



```
[root@server ~]# mkdir /mount_data
[root@server ~]# mount /dev/vgdata/lv1 /mount_data
[root@server ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       872M     0  872M   0% /dev/shm
tmpfs                       349M  7.7M  342M   3% /run
/dev/mapper/almalinux-root   17G   14G  3.3G  81% /
/dev/sda1                   960M  305M  656M  32% /boot
tmpfs                       175M  100K  175M   1% /run/user/0
/dev/sr0                     11G   11G     0 100% /run/media/root/AlmaLinux-9-4-x86_64-dvd
/dev/mapper/vgdata-lv1      960M   39M  922M   5% /mount_data
```

Whenever we extending the logical volume we have to resize the file system

For that we have to use the command

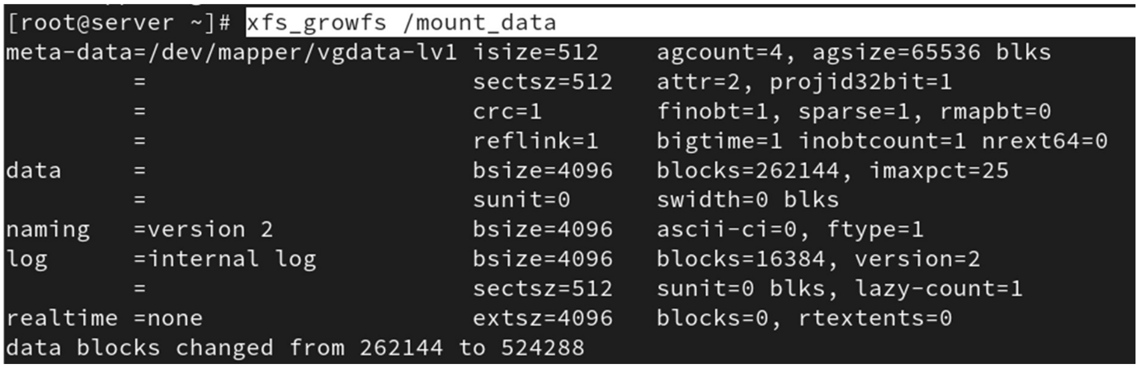
If the file system type is ext4 means : resize2fs /dev/vg_name/lvname

If the file system type is xfs means : xfs_growfs /mountpoint

NOW DEALING WITH XFS TYPE

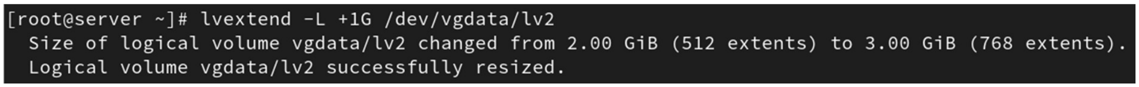
xfs_growfs /mountpoint

xfs_growfs /mount_data



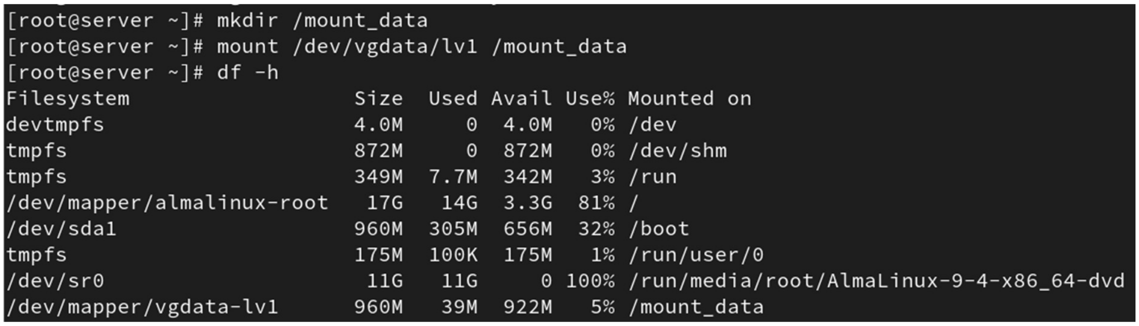
```
[root@server ~]# xfs_growfs /mount_data
meta-data=/dev/mapper/vgdata-lv1 isize=512    agcount=4, agsize=65536 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
data     =                       bsize=4096   blocks=262144, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 262144 to 524288
```

NOW DEALING WITH ext4 TYPE:



```
[root@server ~]# lvextend -L +1G /dev/vgdata/lv2
  Size of logical volume vgdata/lv2 changed from 2.00 GiB (512 extents) to 3.00 GiB (768 extents).
  Logical volume vgdata/lv2 successfully resized.
```

After extending the logical volume mount the lv with mount point:



```
[root@server ~]# mkdir /mount_data
[root@server ~]# mount /dev/vgdata/lv1 /mount_data
[root@server ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       872M     0  872M   0% /dev/shm
tmpfs                       349M  7.7M  342M   3% /run
/dev/mapper/almalinux-root   17G   14G  3.3G  81% /
/dev/sda1                   960M  305M  656M  32% /boot
tmpfs                       175M  100K  175M   1% /run/user/0
/dev/sr0                     11G   11G     0 100% /run/media/root/AlmaLinux-9-4-x86_64-dvd
/dev/mapper/vgdata-lv1      960M   39M  922M   5% /mount_data
```

REDUCING THE LOGICAL VOLUME we have to unmount the lv which is mounted in mount point:

FOR

ext4 type: 1. e2fsck -f /dev/vgdata/lv2

2. resize2fs /dev/vgdata/lv2 1G

3. lvreduce -L 1G /dev/vgdata/lv2

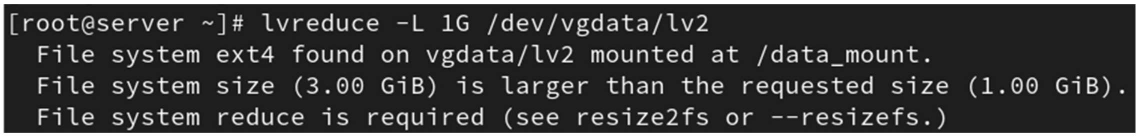
FOR

xfs type: lvreduce -L 1G /dev/vgdata/lv1

Reducing the size of an XFS file system is not supported.

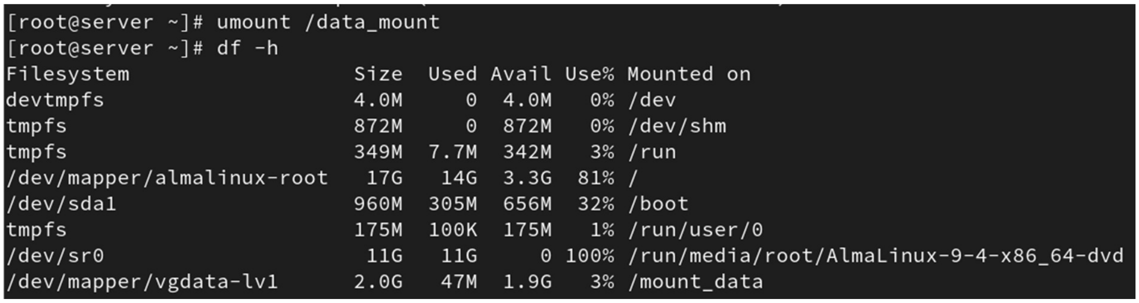
XFS allows only growing the file system, not shrinking it.

Ext4:



```
[root@server ~]# lvreduce -L 1G /dev/vgdata/lv2
  File system ext4 found on vgdata/lv2 mounted at /data_mount.
  File system size (3.00 GiB) is larger than the requested size (1.00 GiB).
  File system reduce is required (see resize2fs or --resizefs.)
```

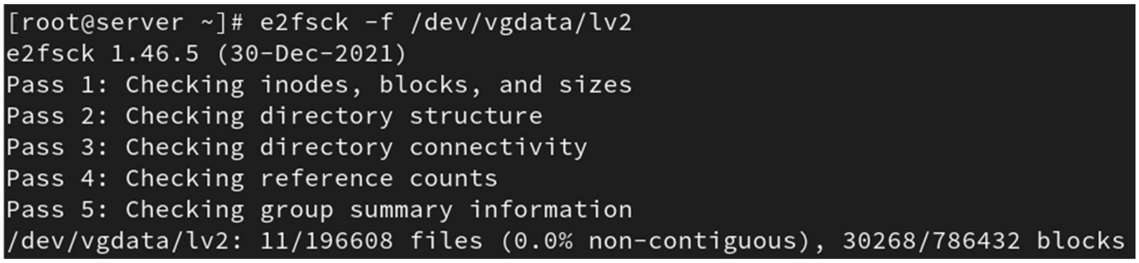
Umount



```
[root@server ~]# umount /data_mount
[root@server ~]# df -h
Filesystem                 Size  Used Avail Use% Mounted on
devtmpfs                   4.0M     0  4.0M   0% /dev
tmpfs                      872M     0  872M   0% /dev/shm
tmpfs                      349M  7.7M  342M   3% /run
/dev/mapper/almalinux-root   17G   14G  3.3G  81% /
/dev/sda1                  960M  305M  656M  32% /boot
tmpfs                      175M  100K  175M   1% /run/user/0
/dev/sr0                    11G   11G     0 100% /run/media/root/AlmaLinux-9-4-x86_64-dvd
/dev/mapper/vgdata-lv1     2.0G   47M  1.9G   3% /mount_data
```

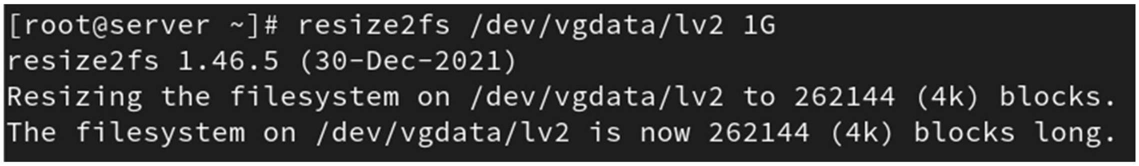
File system checking

e2fsck -f /dev/vgdata/lv2



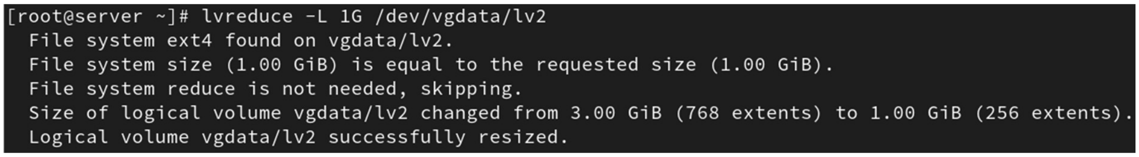
```
[root@server ~]# e2fsck -f /dev/vgdata/lv2
e2fsck 1.46.5 (30-Dec-2021)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/vgdata/lv2: 11/196608 files (0.0% non-contiguous), 30268/786432 blocks
```

resize2fs /dev/vgdata/lv2 1G



```
[root@server ~]# resize2fs /dev/vgdata/lv2 1G
resize2fs 1.46.5 (30-Dec-2021)
Resizing the filesystem on /dev/vgdata/lv2 to 262144 (4k) blocks.
The filesystem on /dev/vgdata/lv2 is now 262144 (4k) blocks long.
```

Now reducing the logical volume



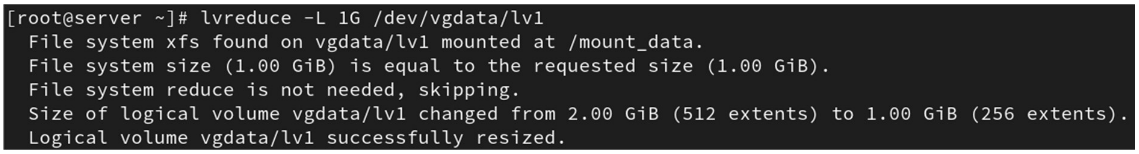
```
[root@server ~]# lvreduce -L 1G /dev/vgdata/lv2
  File system ext4 found on vgdata/lv2.
  File system size (1.00 GiB) is equal to the requested size (1.00 GiB).
  File system reduce is not needed, skipping.
  Size of logical volume vgdata/lv2 changed from 3.00 GiB (768 extents) to 1.00 GiB (256 extents).
  Logical volume vgdata/lv2 successfully resized.
```

I can able to reduce the logical volume

<u>FOR:</u>

xfs type

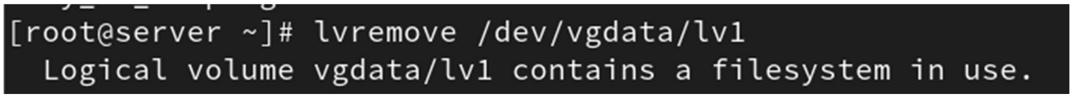
lvreduce -L 1G /dev/vgdata/lv1



```
[root@server ~]# lvreduce -L 1G /dev/vgdata/lv1
  File system xfs found on vgdata/lv1 mounted at /mount_data.
  File system size (1.00 GiB) is equal to the requested size (1.00 GiB).
  File system reduce is not needed, skipping.
  Size of logical volume vgdata/lv1 changed from 2.00 GiB (512 extents) to 1.00 GiB (256 extents).
  Logical volume vgdata/lv1 successfully resized.
```

<u>Note:</u> If we mounted it permanently we can't able to extend or reduce the logical volume.

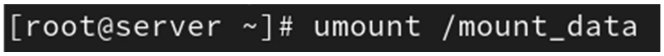
When we remove the logical volume sometimes it will show you cannot remove the file system is in is use this is because of the file system is mounted in the mount point



```
[root@server ~]# lvremove /dev/vgdata/lv1
  Logical volume vgdata/lv1 contains a filesystem in use.
```

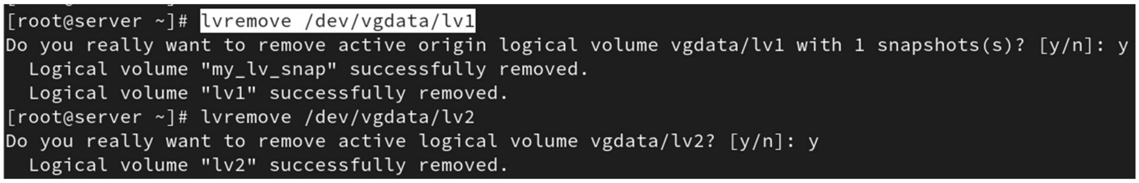
To remove all those logical volume and volume group follow the steps:

unmount the mount point



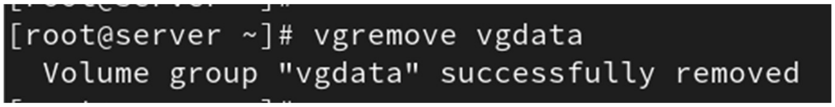
```
[root@server ~]# umount /mount_data
```

And then remove it you can able to remove



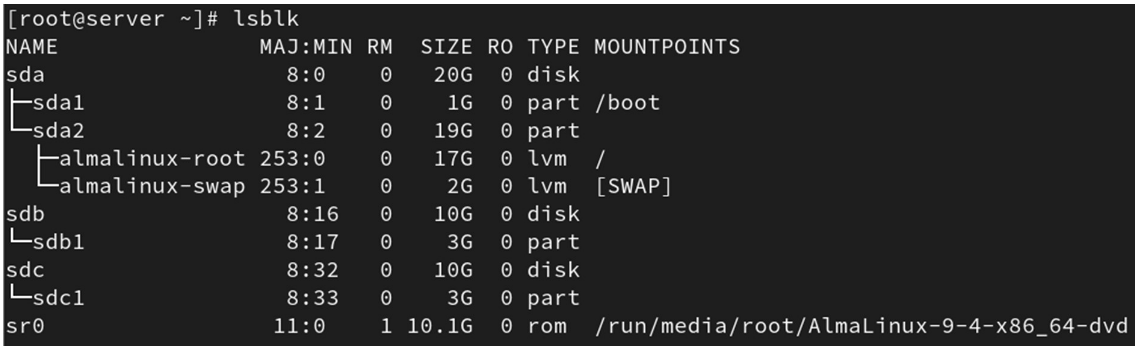
```
[root@server ~]# lvremove /dev/vgdata/lv1
Do you really want to remove active origin logical volume vgdata/lv1 with 1 snapshots(s)? [y/n]: y
  Logical volume "my_lv_snap" successfully removed.
  Logical volume "lv1" successfully removed.
[root@server ~]# lvremove /dev/vgdata/lv2
Do you really want to remove active logical volume vgdata/lv2? [y/n]: y
  Logical volume "lv2" successfully removed.
```

Same to volume group also



```
[root@server ~]# vgremove vgdata
  Volume group "vgdata" successfully removed
```

Check with lsblk



```
[root@server ~]# lsblk
NAME               MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0   20G  0 disk
├─sda1               8:1    0    1G  0 part /boot
└─sda2               8:2    0   19G  0 part
  ├─almalinux-root 253:0    0   17G  0 lvm  /
  └─almalinux-swap 253:1    0    2G  0 lvm  [SWAP]
sdb                  8:16   0   10G  0 disk
└─sdb1               8:17   0    3G  0 part
sdc                  8:32   0   10G  0 disk
└─sdc1               8:33   0    3G  0 part
sr0                 11:0    1 10.1G  0 rom  /run/media/root/AlmaLinux-9-4-x86_64-dvd
```

SNAPSHOT

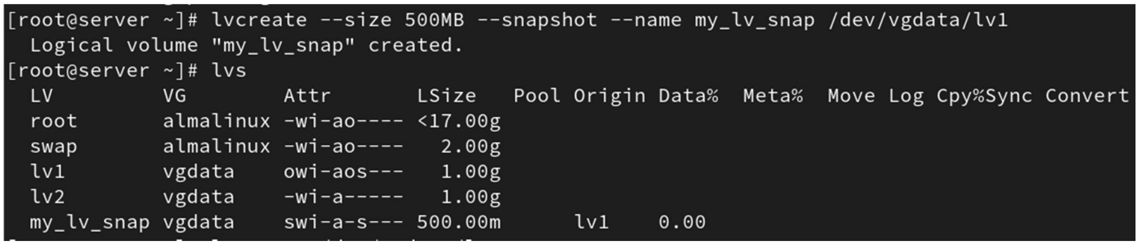
LVM snapshots use a copy-on-write (CoW) mechanism. When you create a snapshot, it does not immediately duplicate the entire contents of the original volume. Instead, it only keeps track of changes to the original volume after the snapshot is taken. The snapshot retains the original data in its pre-change state, allowing you to restore or access the volume as it was at the time the snapshot was created.

- Snapshot Origin (Source Volume): The original logical volume from which the snapshot is created.
- Snapshot Volume: The copy of the original logical volume at the time the snapshot was taken.

Use Cases

1. Backup: Snapshots allow you to back up a consistent version of a logical volume without interrupting applications that are using the volume.
2. Testing: You can create snapshots before making changes to the system, so if something goes wrong, you can easily revert back to the snapshot.
3. Recovery: In the event of data corruption or other failures, snapshots can be used to recover the data as it was at the time the snapshot was taken.

lvcreate --size 500MB --snapshot --name my_lv_snap /dev/vgdata/lv1



```
[root@server ~]# lvcreate --size 500MB --snapshot --name my_lv_snap /dev/vgdata/lv1
  Logical volume "my_lv_snap" created.
[root@server ~]# lvs
  LV         VG        Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root       almalinux -wi-ao---- <17.00g
  swap       almalinux -wi-ao----   2.00g
  lv1        vgdata    owi-aos---   1.00g
  lv2        vgdata    -wi-a-----   1.00g
  my_lv_snap vgdata    swi-a-s--- 500.00m      lv1    0.00
```