RAID:

RAID (Redundant Array of Independent Disks) in Linux is a technology that combines multiple physical drives into a single logical unit.

What is RAID 1 (Mirroring)?

- **Definition:** Copies (mirrors) data identically across two or more disks simultaneously.
- **Purpose:** Provides redundancy—if one disk fails, data remains accessible from the other disk(s).
- **Tradeoff:** No capacity gain (you only get capacity of one drive), and write performance is pegged to the slowest disk.

Why RAID 1?

- Ensures data safety: complete duplication on each disk.
- Read performance may improve (reads can happen from any mirrored disk), but writes go to all disks, so write speed ≈ single disk speed.
- If any single disk fails, the volume stays intact; completed with hot- spare use or redundancy.
- Minimum of two disks required.

To configure **RAID 1 (mirroring)** on Linux:

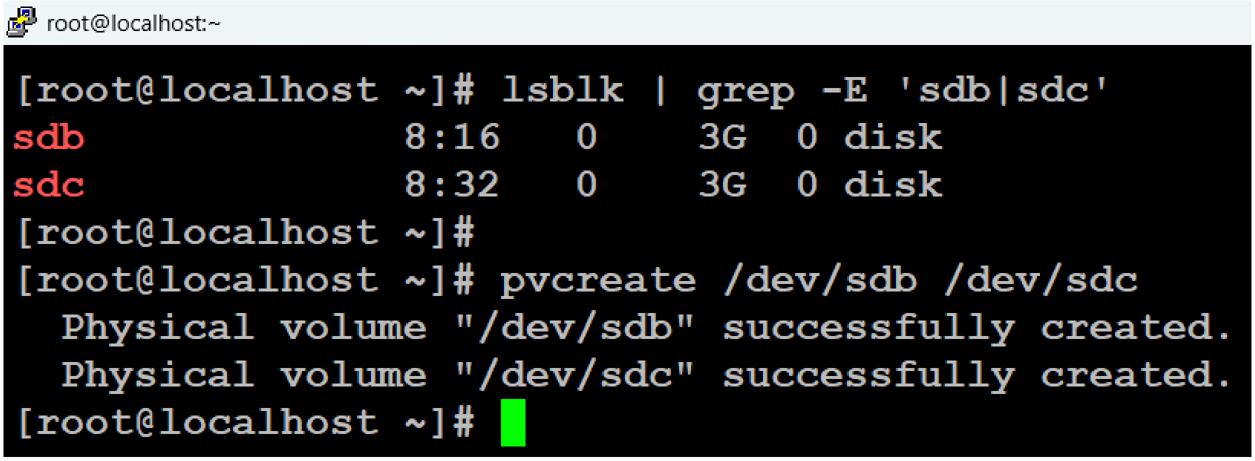
1. LVM (Logical Volume Manager)
2. MDADM (Multiple Device Administration)

RAID 1 Mirroring with LVM – Full Workflow

Step 1: Add New Disks & Initialize as PVs

Once you physically attach or virt- add your disks (e.g., /dev/sdb, /dev/sdc), format them as LVM PVs:

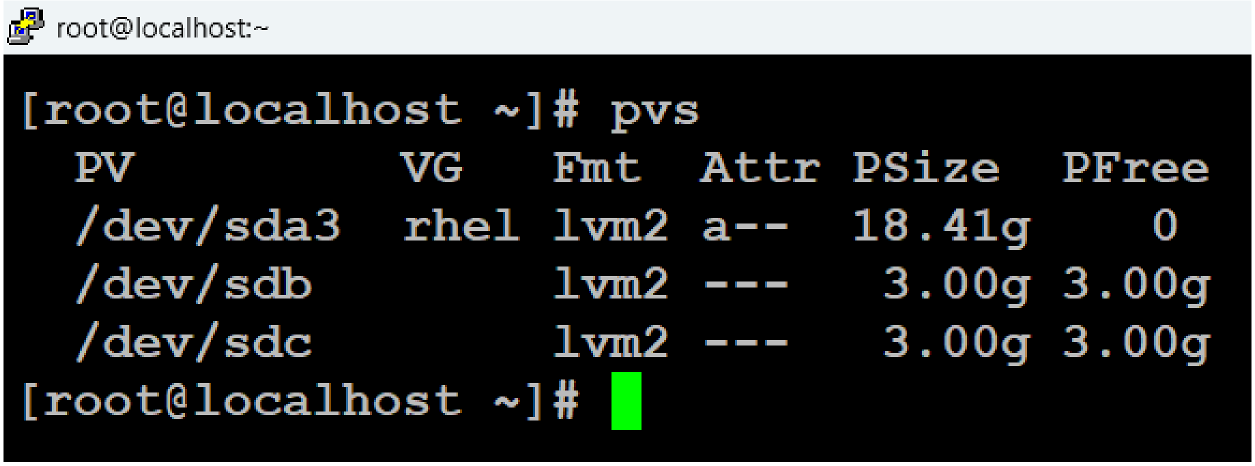
Command: pvcreate /dev/sdb /dev/sdc



```
[root@localhost ~]# lsblk | grep -E 'sdb|sdc'
sdb             8:16   0    3G  0 disk
sdc             8:32   0    3G  0 disk
[root@localhost ~]#
[root@localhost ~]# pvcreate /dev/sdb /dev/sdc
  Physical volume "/dev/sdb" successfully created.
  Physical volume "/dev/sdc" successfully created.
[root@localhost ~]#
```

Creates three PVs ready for LVM usage.

Check with:

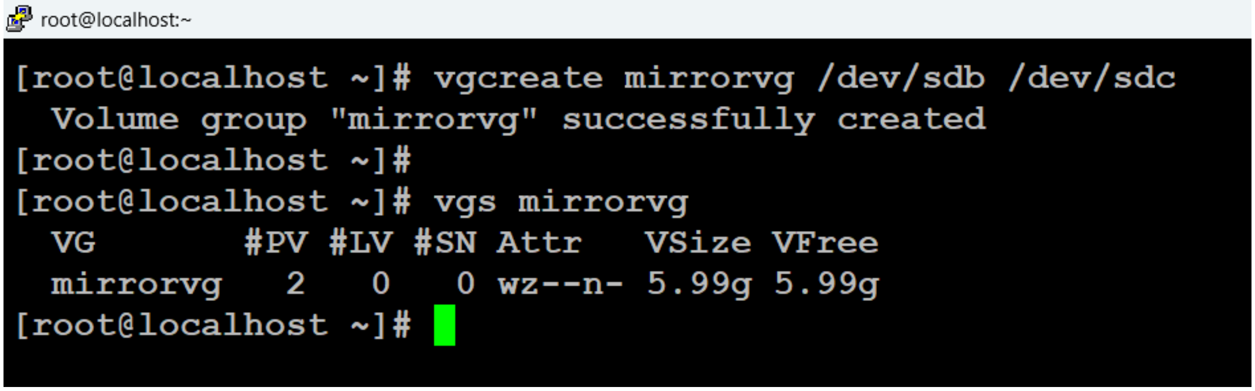


```
[root@localhost ~]# pvs
  PV         VG   Fmt  Attr PSize  PFree
  /dev/sda3  rhel lvm2 a--  18.41g     0
  /dev/sdb        lvm2 ---   3.00g 3.00g
  /dev/sdc        lvm2 ---   3.00g 3.00g
[root@localhost ~]#
```

Step 2: Create a Volume Group (VG)

Combine the PVs into a volume group:

Command: vgcreate mirrorvg /dev/sdb /dev/sdc



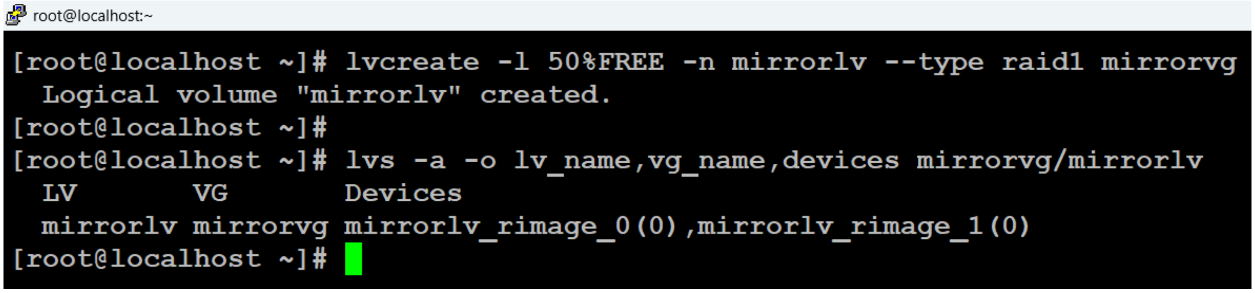
```
[root@localhost ~]# vgcreate mirrorvg /dev/sdb /dev/sdc
  Volume group "mirrorvg" successfully created
[root@localhost ~]#
[root@localhost ~]# vgs mirrorvg
  VG       #PV #LV #SN Attr   VSize VFree
  mirrorvg   2   0   0 wz--n- 5.99g 5.99g
[root@localhost ~]#
```

Combines the PVs into mirrorvg

Step 3: Create a Mirrored (RAID-1) Logical Volume

To stripe across the 3 PVs:

Command: lvcreate -l 50%FREE -n mirrorlv --type raid1 mirrorvg

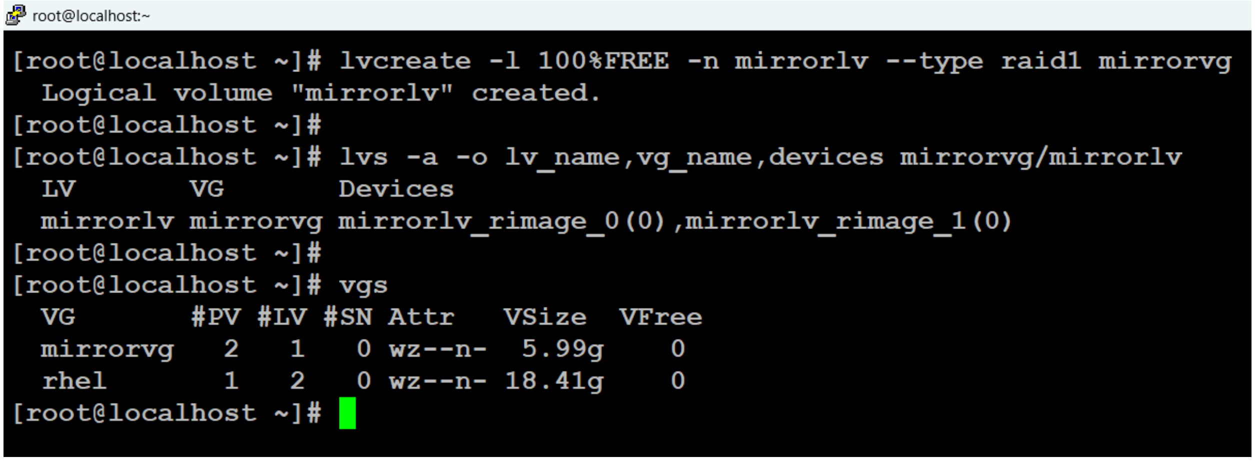


```
[root@localhost ~]# lvcreate -l 50%FREE -n mirrorlv --type raid1 mirrorvg
  Logical volume "mirrorlv" created.
[root@localhost ~]#
[root@localhost ~]# lvs -a -o lv_name,vg_name,devices mirrorvg/mirrorlv
  LV       VG       Devices
  mirrorlv mirrorvg mirrorlv_rimage_0(0),mirrorlv_rimage_1(0)
[root@localhost ~]#
```

- **lvcreate**
  This tells LVM you want to **create a new logical disk** inside your existing storage pool (the volume group).
- **-l 50%FREE**
  Use **half of the currently unused space** in the volume group.

If you have 10 GiB free, this uses 5 GiB.

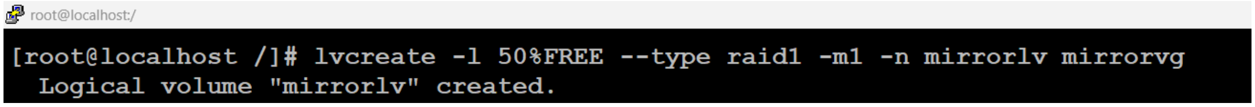
- **-n mirrorlv**
  Name the new logical volume **mirrorlv**—just a friendly label.
- **--type raid1**
  Create a **mirrored disk (RAID 1)**—every bit you write gets copied to two physical disks for safety and redundancy.
- **mirrorvg**
  This is the name of your **volume group**—the storage pool where LVM will create the new logical disk.



```
[root@localhost ~]# lvcreate -l 100%FREE -n mirrorlv --type raid1 mirrorvg
  Logical volume "mirrorlv" created.
[root@localhost ~]#
[root@localhost ~]# lvs -a -o lv_name,vg_name,devices mirrorvg/mirrorlv
  LV       VG       Devices
  mirrorlv mirrorvg mirrorlv_rimage_0(0),mirrorlv_rimage_1(0)
[root@localhost ~]#
[root@localhost ~]# vgs
  VG       #PV #LV #SN Attr   VSize  VFree
  mirrorvg   2   1   0 wz--n-  5.99g    0
  rhel       1   2   0 wz--n- 18.41g    0
[root@localhost ~]#
```

- **-l 100%FREE**
  option **allocates *all* of the unused space** in the volume group (VG) to the new logical volume (mirrorlv)
- If your VG has **10 GiB** free, this will create a **10 GiB** LV

NOTE:



```
[root@localhost /]# lvcreate -l 50%FREE --type raid1 -m1 -n mirrorlv mirrorvg
  Logical volume "mirrorlv" created.
```

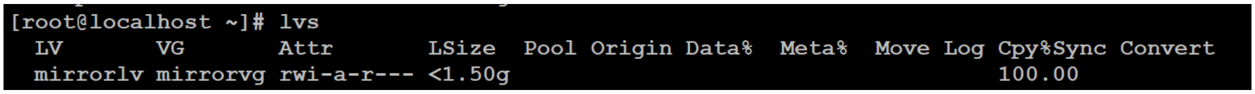
- `-m1` means: **1 mirror copy** = total 2 copies of data (original + mirror)

If have 3 disks,

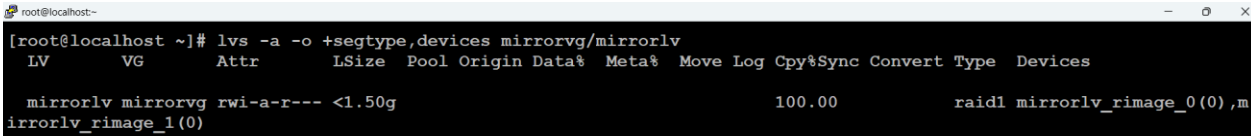
Command: lvcreate -l 50%FREE –type raid1 -m2 -n mirrorlv mirrorvg

- `-m2` = **2 mirror copies**, so **3 disks total** (1 original + 2 mirrors)
- Your data will be written to **all 3 disks simultaneously**
- This setup ensures that even if **2 disks fail**, your data is still safe on the remaining one
- **Result**: Still **only 3 GB usable**, but stored identically on 3 disks(9GB)

Check with:

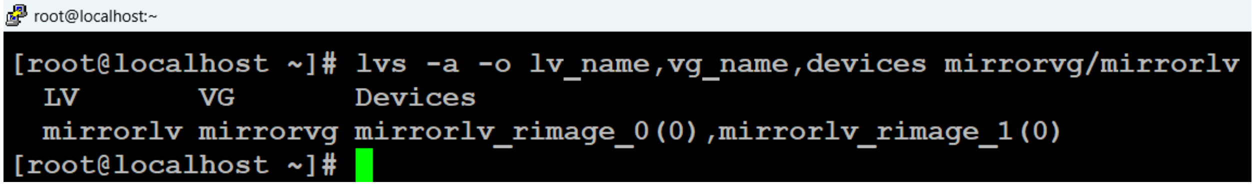


```
[root@localhost ~]# lvs
  LV       VG       Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  mirrorlv mirrorvg rwi-a-r--- <1.50g                                    100.00
```

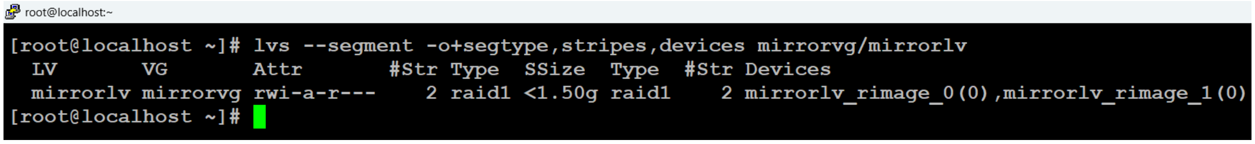


```
[root@localhost ~]# lvs -a -o +segtype,devices mirrorvg/mirrorlv
  LV       VG       Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert Type  Devices

  mirrorlv mirrorvg rwi-a-r--- <1.50g                                    100.00           raid1 mirrorlv_rimage_0(0),m
irrorlv_rimage_1(0)
```



```
[root@localhost ~]# lvs -a -o lv_name,vg_name,devices mirrorvg/mirrorlv
  LV       VG       Devices
  mirrorlv mirrorvg mirrorlv_rimage_0(0),mirrorlv_rimage_1(0)
[root@localhost ~]#
```



```
[root@localhost ~]# lvs --segment -o+segtype,stripes,devices mirrorvg/mirrorlv
  LV       VG       Attr       #Str Type  SSize  Type  #Str Devices
  mirrorlv mirrorvg rwi-a-r---    2 raid1 <1.50g raid1    2 mirrorlv_rimage_0(0),mirrorlv_rimage_1(0)
[root@localhost ~]#
```

- lvs: Shows info about your Logical Volumes (LVs).
- -a: Includes all parts—like hidden mirror segments—not just the main logical volume.
- -o lv_name,vg_name,devices: Only shows these three pieces of info:
  - **lv_name**: The volume's name (e.g., mirrorlv)
  - **vg_name**: The group it belongs to (e.g., mirrorvg)
  - **devices**: The actual disk parts storing the LV (like the mirrored copies)

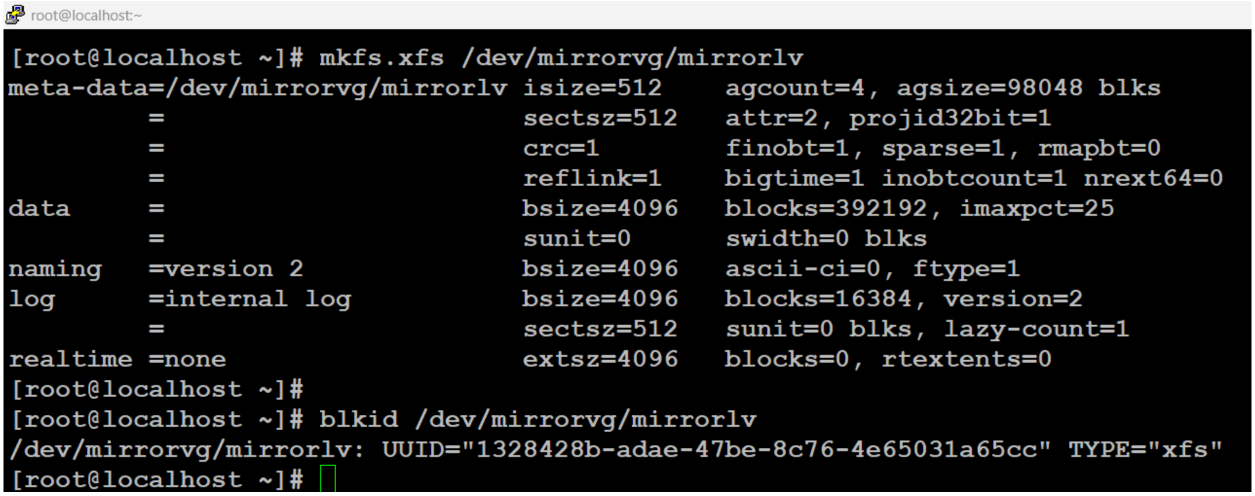
- **mirrorvg/mirrorlv**: Specifically targets the mirrorlv volume in the mirrorvg volume group.
- **mirrorlv_rimage_0** and **mirrorlv_rimage_1** are the two “mirror legs” (replicas) of the volume.
- The (0) indicates each is ACTIVE and online.

Step 4: Format & Mount the LV

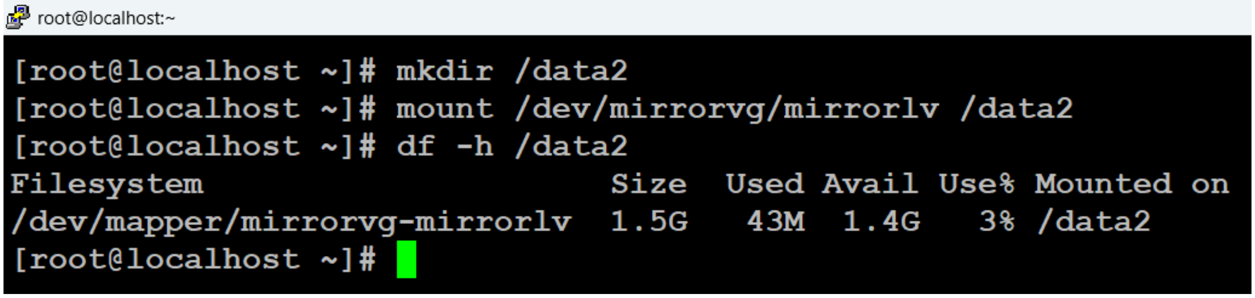
Format the new LV and mount it:

Commands:

mkfs.xfs /dev/mirrorvg/mirrorlv
mkdir /data2
mount /dev/mirrorvg/mirrorlv /data2
df -h /data2



```
[root@localhost ~]# mkfs.xfs /dev/mirrorvg/mirrorlv
meta-data=/dev/mirrorvg/mirrorlv isize=512    agcount=4, agsize=98048 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=392192, 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
[root@localhost ~]#
[root@localhost ~]# blkid /dev/mirrorvg/mirrorlv
/dev/mirrorvg/mirrorlv: UUID="1328428b-adae-47be-8c76-4e65031a65cc" TYPE="xfs"
[root@localhost ~]#
```

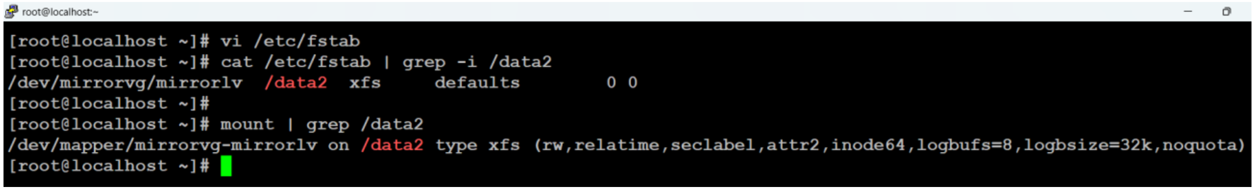


```
[root@localhost ~]# mkdir /data2
[root@localhost ~]# mount /dev/mirrorvg/mirrorlv /data2
[root@localhost ~]# df -h /data2
Filesystem                    Size  Used Avail Use% Mounted on
/dev/mapper/mirrorvg-mirrorlv 1.5G   43M  1.4G   3% /data2
[root@localhost ~]#
```

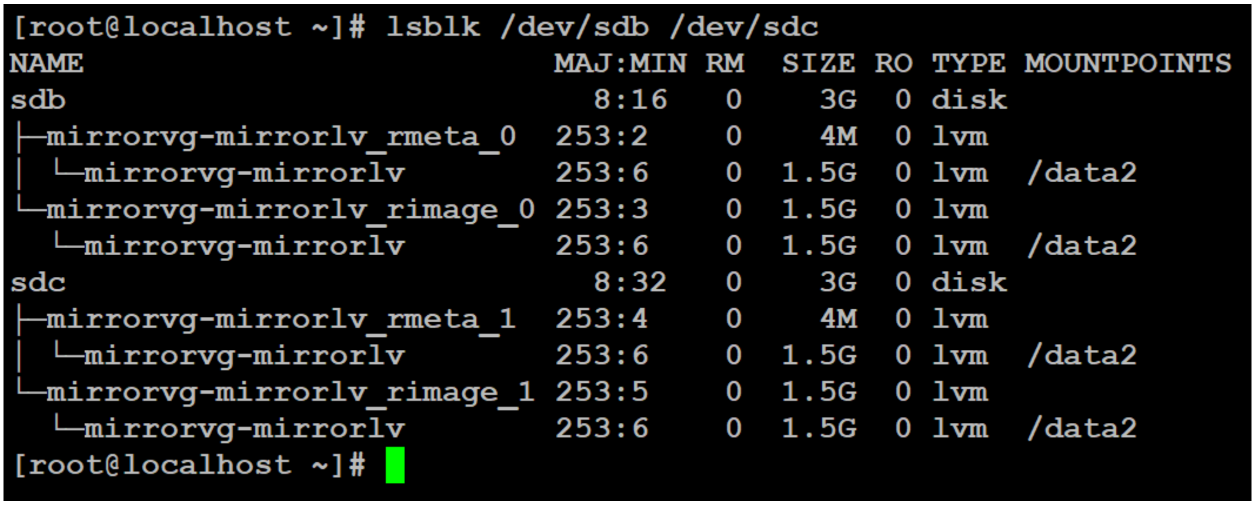
Step 5: Add entry to /etc/fstab for persistent mount:

Create a stable mount entry so that **/data2** mounts automatically at every boot.

Command: vi /etc/fstab



```
[root@localhost ~]# vi /etc/fstab
[root@localhost ~]# cat /etc/fstab | grep -i /data2
/dev/mirrorvg/mirrorlv  /data2  xfs     defaults        0 0
[root@localhost ~]#
[root@localhost ~]# mount | grep /data2
/dev/mapper/mirrorvg-mirrorlv on /data2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
[root@localhost ~]#
```



```
[root@localhost ~]# lsblk /dev/sdb /dev/sdc
NAME                            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sdb                               8:16   0    3G  0 disk
├─mirrorvg-mirrorlv_rmeta_0     253:2    0    4M  0 lvm
│ └─mirrorvg-mirrorlv           253:6    0  1.5G  0 lvm  /data2
└─mirrorvg-mirrorlv_rimage_0 253:3    0  1.5G  0 lvm
  └─mirrorvg-mirrorlv           253:6    0  1.5G  0 lvm  /data2
sdc                               8:32   0    3G  0 disk
├─mirrorvg-mirrorlv_rmeta_1     253:4    0    4M  0 lvm
│ └─mirrorvg-mirrorlv           253:6    0  1.5G  0 lvm  /data2
└─mirrorvg-mirrorlv_rimage_1 253:5    0  1.5G  0 lvm
  └─mirrorvg-mirrorlv           253:6    0  1.5G  0 lvm  /data2
[root@localhost ~]#
```

Final Takeaway for RAID 1

RAID 1 ensures full data redundancy by mirroring—every write is copied identically to another disk. While usable capacity is halved, it provides high availability even if a disk fails, making it ideal for protecting critical data. When paired with backups, it offers both performance and reliability. It ensures **high availability and redundancy**—even if one drive fails, your data remains safe and the filesystem remains online.