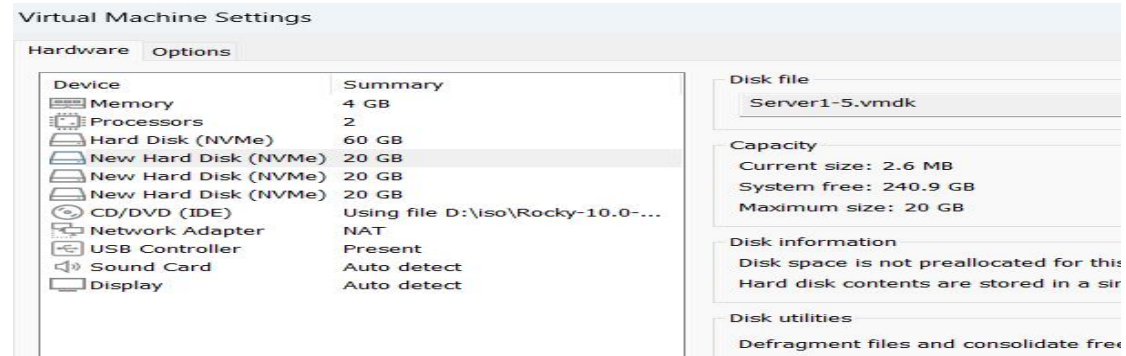**Storage Management**
RAID

In this lab you will configure a RAID 1 and RAID 5 array in Linux.
The mdadm package allows you to create RAID arrays.

First attach 3 hard disk of 20 GB each to the Linux virtual machine as shown below.

Then start the virtual machine and login with a user having sudo permissions.
First you need to install the mdadm package.

sudo yum install mdadm -y

You can verify the successful installation of mdadm package using the following command.

mdadm -h

Verify if there is any RAID array already created in the system using following command.

cat /proc/mdstat

Create a RAID 1 Array

First you will create a RAID 1 array. It is mirroring. It requires 2 hard disks. The RAID device is identified as /dev/md0 or /dev/md1 etc.

Identify the disk names for each 20 GB hard disk. Use lsblk command.

```
[admin@demosrv ~]$ lsblk
NAME            MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sr0              11:0    1   7.1G  0 rom
nvme0n2         259:0    0    20G  0 disk
nvme0n3         259:1    0    20G  0 disk
nvme0n4         259:2    0    60G  0 disk
├─nvme0n4p1     259:4    0     1M  0 part
├─nvme0n4p2     259:5    0     1G  0 part /boot
└─nvme0n4p3     259:6    0    59G  0 part
  ├─rl-root     253:0    0    37G  0 lvm  /
  ├─rl-swap     253:1    0   3.9G  0 lvm  [SWAP]
  └─rl-home     253:2    0  18.1G  0 lvm  /home
nvme0n1         259:3    0    20G  0 disk
[admin@demosrv ~]$ _
```

Here it is /dev/nvme0n1 , /dev/nvme0n2 and /dev/nvme0n3. In your case these names may be different.
Here you will use /dev/nvme0n1 and /dev/nvme0n2 hard disks to create a RAID array by name /dev/md0 .

To create a RAID 1 array give the following command.

sudo   mdadm   --create   --verbose   /dev/md0   --level=1   --raid-devices=2   /dev/nvme0n1 /dev/nvme0n2

```
[admin@demosrv ~]$ sudo mdadm --create --verbose /dev/md0 --level=1 --raid-devices=2 /dev/nvme0n1 /dev/nvm
e0n2
mdadm: Note: this array has metadata at the start and
    may not be suitable as a boot device.  If you plan to
    store '/boot' on this device please ensure that
    your boot-loader understands md/v1.x metadata, or use
    --metadata=0.90
mdadm: size set to 20954112K
Continue creating array [y/N]? y
mdadm: Defaulting to version 1.2 metadata
[ 1097.8621123] md/raid1:md0: not clean -- starting background reconstruction
[ 1097.8621183] md/raid1:md0: active with 2 out of 2 mirrors
[ 1097.8634192] md0: detected capacity change from 0 to 41908224
mdadm: array /dev/md0 started.
[ 1097.8670067] md: resync of RAID array md0
[admin@demosrv ~]$
```

Type y at the prompt to create the RAID array.

Verify that the RAID 1 array is created using following command.

cat /proc/mdstat

```
[admin@demosrv ~]$ cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 nvme0n2[1] nvme0n1[0]
      20954112 blocks super 1.2 [2/2] [UU]

unused devices: <none>
[admin@demosrv ~]$
```

Format the RAID array using following command.

sudo mkfs.ext4 /dev/md0

```
[admin@demosrv ~]$ sudo mkfs.ext4 /dev/md0
[sudo] password for admin:
mke2fs 1.47.1 (20-May-2024)
Creating filesystem with 5238528 4k blocks and 1310720 inodes
Filesystem UUID: cbcbc839-b374-4915-ad5d-8cd7e236d441
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[admin@demosrv ~]$ _
```

Mount the array. For this create a directory by name /mnt/raid1.

```
[admin@demosrv ~]$ sudo mkdir /mnt/raid1
```

Now give following command to mount the /dev/md0 device.

sudo mount -t auto /dev/md0 /mnt/raid1

```
[admin@demosrv ~]$ sudo mount -t auto /dev/md0 /mnt/raid1
[ 1898.874800] EXT4-fs (md0): mounted filesystem cbcbc839-b374-4915-ad5d-8cd7e236d441 r/w with ordered dat
a mode. Quota mode: none.
[admin@demosrv ~]$ _
```

Now you can create data into /mnt/raid1 directory. These files will be written on both the hard disks.

To make sure that the array is reassembled automatically at boot, you have to adjust the /etc/mdadm/mdadm.conf file. You can automatically scan the active array and append the file with the following command.

sudo mkdir /etc/mdadm
sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf

```
[admin@demosrv ~]$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf
ARRAY /dev/md0 metadata=1.2 UUID=cff1eba0:480b4c6f:d35d67f7:8db247ef
```

You will need to add an entry in the /etc/fstab file for mounting the /dev/md0 device each time the Linux operating system starts.

This is how you have successfully created and mounted a RAID 1 array.

Removing the RAID 1 array.

Unmount the /dev/md0 device using following command.

sudo umount /dev/md0

```
[admin@demosrv ~]$ sudo umount /dev/md0
[ 2160.214368] EXT4-fs (md0): unmounting filesystem cbcbc839-b374-4915-ad5d-8cd7e236d441.
[admin@demosrv ~]$ _
```

Then stop the RAID device /dev/md0

sudo   mdadm --stop /dev/md0

```
[admin@demosrv ~]$ sudo mdadm --stop /dev/md0
[  269.933584] md0: detected capacity change from 41908224 to 0
[  269.933635] md: md0 stopped.
mdadm: stopped /dev/md0
[admin@demosrv ~]$
```

Verify with

cat /proc/mdstat

```
[admin@demosrv ~]$ cat /proc/mdstat
Personalities : [raid1]
unused devices: <none>
[admin@demosrv ~]$
```

Now give the following command

lsblk -o NAME,SIZE,FSTYPE,TYPE,MOUNTPOINT
```
nvme0n4      259:3    0    20G   0 disk
[admin@demosrv ~]$ lsblk -o NAME,SIZE,FSTYPE,TYPE,MOUNTPOINT
NAME            SIZE FSTYPE              TYPE MOUNTPOINT
sr0             7.1G iso9660             rom
nvme0n1          60G                     disk
├─nvme0n1p1       1M                     part
├─nvme0n1p2       1G xfs                 part /boot
└─nvme0n1p3      59G LVM2_member         part
  ├─rl-root      37G xfs                 lvm  /
  ├─rl-swap     3.9G swap                lvm  [SWAP]
  └─rl-home    18.1G xfs                 lvm  /home
nvme0n2          20G linux_raid_member   disk
nvme0n3          20G linux_raid_member   disk
nvme0n4          20G                     disk
[admin@demosrv ~]$ _
```

The output shows that the hard disks /dev/nvme0n2 and /dev/nvme0n3 are showing FSTYPE as linux_raid_member.
You need to zero their superblock which holds metadata for the RAID setup. Zeroing this removes the RAID metadata and resets them to normal. For this give following commands

sudo mdadm --zero-superblock /dev/nvme0n2
sudo mdadm --zero-superblock /dev/nvme0n3

```
[admin@demosrv ~]$ sudo mdadm --zero-superblock /dev/nvme0n2
[admin@demosrv ~]$ sudo mdadm --zero-superblock /dev/nvme0n3
[admin@demosrv ~]$ _
```

Now again confirm with the lsblk command.

```
[admin@demosrv ~]$ lsblk -o NAME,SIZE,FSTYPE,TYPE,MOUNTPOINT
NAME            SIZE FSTYPE              TYPE MOUNTPOINT
sr0             7.1G iso9660             rom
nvme0n1          60G                     disk
├─nvme0n1p1       1M                     part
├─nvme0n1p2       1G xfs                 part /boot
└─nvme0n1p3      59G LVM2_member         part
  ├─rl-root      37G xfs                 lvm  /
  ├─rl-swap     3.9G swap                lvm  [SWAP]
  └─rl-home    18.1G xfs                 lvm  /home
nvme0n2          20G                     disk
nvme0n3          20G                     disk
nvme0n4          20G                     disk
[admin@demosrv ~]$
```

Edit the /etc/mdadm.conf file and remove the RAID1 array entry.
Also if you have an entry in the /etc/fstab file for the /dev/md0 RAID 1 Array.
Restart the virtual machine using   **sudo init 6**

Create a RAID 5 Array

To create a RAID 5 array give the following command

sudo mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/nvme0n2 /dev/nvme0n2 /dev/nvme0n3

make sure you give your system device names.

```
[admin@demosrv ~]$ sudo mdadm --create --verbose /dev/md0 --level=5 --raid-devices=3 /dev/nvme0n2 /dev/nvm
e0n3 /dev/nvme0n4
[sudo] password for admin:
mdadm: layout defaults to left-symmetric
mdadm: layout defaults to left-symmetric
mdadm: chunk size defaults to 512K
mdadm: size set to 20954112K
mdadm: Defaulting to version 1.2 metadata
[ 2828.498116] raid6: skipped pq benchmark and selected avx2x4
[ 2828.498163] raid6: using avx2x2 recovery algorithm
[ 2828.512442] async_tx: api initialized (async)
[ 2828.522650] xor: automatically using best checksumming function    avx
[ 2828.576465] md/raid:md0: device nvme0n3 operational as raid disk 1
[ 2828.576528] md/raid:md0: device nvme0n2 operational as raid disk 0
[ 2828.582706] md/raid:md0: raid level 5 active with 2 out of 3 devices, algorithm 2
[ 2828.5833091] md0: detected capacity change from 0 to 83816448
mdadm: array /dev/md0 started.
[ 2828.591886] md: recover of RAID array md0
[admin@demosrv ~]$
```

Check with

cat /proc/mdstat

```
[admin@demosrv ~]$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 nvme0n4[1] nvme0n2[3] nvme0n3[0]
      41908224 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
[admin@demosrv ~]$
```

Format the device using a file system. Here you will use EXT4 file system.

```
[admin@demosrv ~]$ sudo mkfs.ext4 /dev/md0
[sudo] password for admin:
mke2fs 1.47.1 (20-May-2024)
/dev/md0 contains a ext4 file system
        last mounted on Thu Sep 25 14:24:20 2025
Proceed anyway? (y,N) y
Creating filesystem with 10477056 4k blocks and 2621440 inodes
Filesystem UUID: ef225a1e-24db-4598-9f4a-34dada86eceb
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624

Allocating group tables: done
Writing inode tables: done
Creating journal (65536 blocks): done
Writing superblocks and filesystem accounting information: done

[admin@demosrv ~]$
```

Now create a directory and mount the device.
Add an entry in the /etc/fstab file to mount the device each time the Linux system boots.

Follow the same steps as RAID 1 array to remove the RAID 5 array.