

## Storage Management Using fdisk

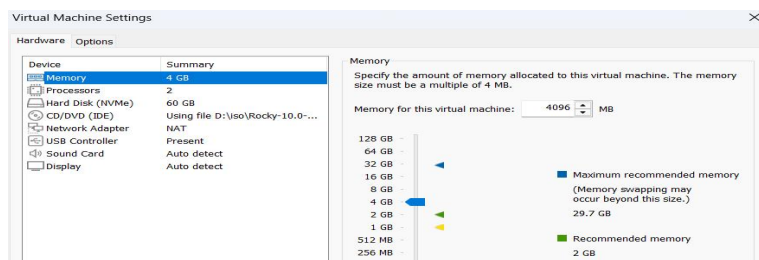
The **fdisk** command in Linux is a menu-driven utility used for creating and manipulating disk partition tables. It is a fundamental tool for managing hard drive structures, allowing users to create, delete, resize, and manage partitions on storage devices.

fdisk enables the creation of primary, extended, and logical partitions on a disk. It also allows for the deletion and modification of existing partitions. It directly interacts with the disk's partition table (**primarily MBR**, but also supports other types like Sun, SGI, and BSD disklabels), allowing for changes to partition types and bootable flags.

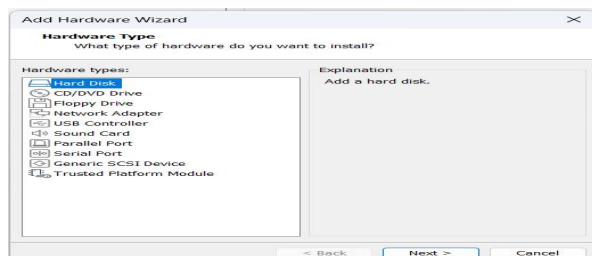
In this document we will create partitions using fdisk. Format them and mount them.

This lab makes use of a virtual machine installed with any Linux operating system.

First go to the settings of the virtual machine.



Click Add button.



Select Hard Disk and click Next. Keep the recommended option as it is. Click Next.

Keep create a New Virtual Hard Disk option selected and click Next.

On the next screen keep the Disk size as 20GB. Make sure this disk size is different that the original hard disk on which Linux OS is installed. Select Store virtual disk as a single file.



Click Next. Click Finish. Click Ok to close the virtual machine settings page.

Now start the virtual machine.

## Creating Partitions on the new hard disk

1. Once the virtual machine starts, log in with a user who has sudo permissions.

First you need to check how Linux OS has detected the new hard disk. For this use following command.

```
sudo fdisk -l
```

The output may show multiple devices. You need to find the device that shows 20 GB capacity as shown below.

```
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 5CF0DD3A-B2AA-4865-A1D1-DFA999718F55

Device            Start      End      Sectors  Size Type
/dev/nvme0n1p1     2048      4095      2048     1M BIOS boot
/dev/nvme0n1p2     4096    2101247  2097152    1G Linux extended boot
/dev/nvme0n1p3    2101248 125827071 123725824  59G Linux LVM

Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Here the disk /dev/nvme0n2 is shown with a capacity of 20 GiB. [In your case the hard disk name may be different. Thus use the device name as per your system.](#) Also this disk is not showing any partitions. This is the hard disk on which we will create partitions.

Here we will create 2 partitions. The first partition will be of 8 GB and primary partition. The second partition will be of 12 GB and it will be of type extended. This extended partition will be then divided into 2 logical drives - one of 5 GB and other of 7 GB.

To create the partitions use following command.

```
sudo fdisk /dev/nvme0n2 # make sure you use your system device name
```

Following prompt is displayed.

```
[admin@demomrv ~]$ sudo fdisk /dev/nvme0n2

Welcome to fdisk (util-linux 2.40.2).
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 (MBR) disklabel with disk identifier 0x6658ae57.

Command (m for help):
```

If you type m at the prompt, it will display the commands that you can execute using this fdisk prompt.

```
a toggle a bootable flag
b edit nested BSD disklabel
c toggle the dos compatibility flag

Generic
d delete a partition
F list free unpartitioned space
l list known partition types
n add a new partition
p print the partition table
t change a partition type
v verify the partition table
i print information about a partition
e resize a partition
```

As displayed in the above help to add/create a new partition type n at the prompt.

```
Command (m for help): n
Partition type
   p primary (0 primary, 0 extended, 4 free)
   e extended (container for logical partitions)
Select (default p):
```

Here we have to create a primary partition first. Thus type p or press Enter as the default option is p.

```
Using default response p.  
Partition number (1-4, default 1): _
```

It will ask for the partition number. As this is the first partition on the hard disk press enter to accept the default number 1.

```
Using default response p.  
Partition number (1-4, default 1):  
First sector (2048-41943039, default 2048):
```

It will ask for the starting sector number from where this partition will start. The default is 2048. Earlier sectors are used by the OS for its own purpose. Thus keep default and press Enter.

```
Using default response p.  
Partition number (1-4, default 1):  
First sector (2048-41943039, default 2048):  
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-41943039, default 41943039): +8G_
```

Next it will ask to either enter the last sector number or the capacity of the partition you desire. It is difficult to calculate the last sector number thus we will specify the desired capacity of the partition. As we have decided to create a 8GB partition, enter +8G as shown above and press Enter.

This is how you have create the first primary partition of 8GB.

Next we have to create an extended partition using remaining space on the hard disk which is 12 GB.

For this on the fdisk prompt enter n again.

```
Command (m for help): n  
Partition type  
p primary (1 primary, 0 extended, 3 free)  
e extended (container for logical partitions)  
Select (default p): e
```

But this time type e to create an extended partition as shown above.

Press Enter to accept the default partition number as 2.

Press Enter to accept the default starting sector.

Press Enter to accept the default last sector. This is because we want to use entire remaining space for this extended partition. Thus now there are 2 partitions created on this disk. Now there is no unallocated space remaining.

Now we have to create 2 logical drives inside the extended partition. For this again type n at the fdisk prompt.

```
Command (m for help): n  
All space for primary partitions is in use.  
Adding logical partition 5  
First sector (16781312-41943039, default 16781312):
```

Now it will not give option to create Primary or Extended partition. It will directly add the logical drive/partition.

Press Enter to select the default starting sector.

Next type the desired capacity of this partition. Here we type +5G as shown below.

```
Command (m for help): n  
All space for primary partitions is in use.  
Adding logical partition 5  
First sector (16781312-41943039, default 16781312):  
Last sector, +/-sectors or +/-size{K,M,G,T,P} (16781312-41943039, default 41943039): +5G
```

Press Enter.

This is how you have created a 5GB logical partition inside the extended partition.

Now to create second logical drive type n at the fdisk prompt. This drive will use all the remaining space. Press Enter at all the prompts to select the default values.

```
Command (m for help): n
All space for primary partitions is in use.
Adding logical partition 6
First sector (27269120-41943039, default 27269120):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (27269120-41943039, default 41943039):

Created a new partition 6 of type 'Linux' and of size 7 GiB.
```

Now verify the partition are properly created. Type p at the fdisk prompt. This will display the partitions on the disk.

```
Command (m for help): p
Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6658ae57

Device            Boot      Start         End Sectors  Size Id Type
/dev/nvme0n2p1                2048    16779263   16777216    8G 83 Linux
/dev/nvme0n2p2    16779264    41943039   25163776   12G  5 Extended
/dev/nvme0n2p5    16781312    27267071   10485760    5G 83 Linux
/dev/nvme0n2p6    27269120    41943039   14673920    7G 83 Linux

Command (m for help): _
```

However these partitions are in the computer memory only. You need to write this partition table to the disk. This will make them permanent on the disk.

For this type w at the fdisk prompt. This will exit the fdisk utility also.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
[ 8408.299286] nvme0n2: p1 p2 < p5 p6 >
Syncing disks.

[admin@demomrv ~]$_
```

To verify that the partitions are recognized by the Linux kernel, type following command.

**sudo fdisk -l**

```
Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6658ae57

Device            Boot      Start         End Sectors  Size Id Type
/dev/nvme0n2p1                2048    16779263   16777216    8G 83 Linux
/dev/nvme0n2p2    16779264    41943039   25163776   12G  5 Extended
/dev/nvme0n2p5    16781312    27267071   10485760    5G 83 Linux
/dev/nvme0n2p6    27269120    41943039   14673920    7G 83 Linux
```

You can use **lsblk** command also.

```
[admin@demomrv ~]$_ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sr0          11:0    1   7.1G  0 rom  /mnt/cdrom
nvme0n1      259:0    0   60G  0 disk
├─nvme0n1p1  259:2    0    1M  0 part
├─nvme0n1p2  259:3    0    1G  0 part /boot
├─nvme0n1p3  259:4    0   59G  0 part
├─r1-root    253:0    0    37G  0 lvm  /
├─r1-swap    253:1    0    3.9G  0 lvm  [SWAP]
└─r1-home    253:2    0   18.1G  0 lvm  /home
nvme0n2      259:1    0   20G  0 disk
├─nvme0n2p1  259:5    0    8G  0 part
├─nvme0n2p2  259:6    0    1K  0 part
├─nvme0n2p5  259:7    0    5G  0 part
└─nvme0n2p6  259:8    0    7G  0 part

[admin@demomrv ~]$_
```

Now this 20 GB hard disk is showing the partitions.

This is how you have successfully created partitions on a new hard disk using fdisk utility.

Format the new partitions.

Linux OS supports various file systems. The most commonly used is the ext4 file systems. The earlier file system was ext3 and ext2 . Both are supported. There are some new filesystems - xfs , reiserfs and btrfs . However by default the Kernel may not support all these new file systems.

To format the partition following commands are available based on the file system that you want.

mkfs.ext3 - Formats the partition using EXT3 file system.

mkfs.ext4 - Formats the partition using EXT4 file system.

mkfs.xfs - Formats the partition using XFS file system.

mkfs.reiserfs - Formats the partition using reiserfs file system.

Here we will format the first primary partition using ext4. Then the first logical partition using ext3 and the second logical partition using xfs file system.

To format the first partition give following command.

**sudo mkfs.ext4 /dev/nvme0n2p1** # Make sure you specify your partition name

```
[admin@demorv ~]$ sudo mkfs.ext4 /dev/nvme0n2p1
[sudo] password for admin:
mke2fs 1.47.1 (20-May-2024)
Creating filesystem with 2097152 4k blocks and 524288 inodes
Filesystem UUID: 30d6b830-2a41-4bc5-a834-741dd3966ff0
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

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

[admin@demorv ~]$ _
```

To format the first logical drive using ext3 file system, give the following command.

**sudo mkfs.ext3 /dev/nvme0n2p5**

```
[admin@demorv ~]$ sudo mkfs.ext3 /dev/nvme0n2p5
mke2fs 1.47.1 (20-May-2024)
Creating filesystem with 1310720 4k blocks and 327680 inodes
Filesystem UUID: 708820a4-fa18-4e7c-8898-b837bf8744e6
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

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

[admin@demorv ~]$ _
```

To format the second logical drive using xfs file system, use following command.

**sudo mkfs.xfs /dev/nvme0n2p6**

```
[admin@demorv ~]$ sudo mkfs.xfs /dev/nvme0n2p6
meta-data=/dev/nvme0n2p6      isize=512    agcount=4, agsize=458560 blks
=                               sectsz=512    attr=2, projid32bit=1
=                               crc=1        finobt=1, sparse=1, rmapbt=1
=                               reflink=1     bigtime=1 inobtcount=1 nrext64=1
=                               exchange=0
data      =                       bsize=4096   blocks=1834240, imaxpct=25
=                               sunit=0       swidth=0 blks
naming    =version 2             bsize=4096   ascii-ci=0, ftype=1, parent=0
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
[admin@demorv ~]$
```

This is how you can format the different partitions on a hard disk using different file systems supported by the local operating system.

You need to mount these partitions, so that users can access them store data (files) on them.

## Mount Partitions

To mount the partitions, first they should be formatted. You can use any file system to format the partitions. The partitions that we want to mount are already formatted.

Now each partition will be mounted in a separate directory. The directory may be below / or below any other directory. This will depend on who (department user) is going to store the data on these partitions.

Here we are keeping the first primary partition for the accounts department data. Thus we will create a /accounts-data directory and mount the partition in it.

For this use following commands.

```
sudo mkdir /accounts-data
```

```
sudo mount -t auto /dev/nvme0n2p1 /accounts-data
```

```
sudo ls -all /accounts-data
```

```
[admin@demomrv ~]$ sudo mkdir /accounts-data
[admin@demomrv ~]$ sudo mount -t auto /dev/nvme0n2p1 /accounts-data
[11303.462352] EXT4-fs (nvme0n2p1): mounted filesystem 30d6b830-2a41-4bc5-a834-741dd3966ff0 r/w with order
ed data mode. Quota mode: none.
[admin@demomrv ~]$ ls -all /accounts-data/
total 20
drwxr-xr-x. 3 root root 4096 Sep 19 13:09 .
dr-xr-xr-x. 20 root root 272 Sep 19 13:36 ..
drwx-----, 2 root root 16384 Sep 19 13:09 lost+found
[admin@demomrv ~]$ _
```

If you see the lost+found directory, it means it's a partition mounted in that directory.

Similarly we will create /sales directory and mount the first logical partition in it. Create /purchase directory and mount the second logical partition into this directory.

For this use following commands

```
sudo mkdir /sales
```

```
sudo mount -t auto /dev/nvme0n2p5 /sales
```

```
sudo mkdir /purchase
```

```
sudo mount -t auto /dev/nvme0n2p6 /purchase
```

```
[admin@demomrv ~]$ sudo mkdir /sales
[sudo] password for admin:
[admin@demomrv ~]$ sudo mount -t auto /dev/nvme0n2p5 /sales
[11664.412286] EXT4-fs (nvme0n2p5): mounting ext3 file system using the ext4 subsystem
[11664.430171] EXT4-fs (nvme0n2p5): mounted filesystem 708820a4-fa18-4e7c-8898-b837bf8744e6 r/w with order
ed data mode. Quota mode: none.
[admin@demomrv ~]$ sudo mkdir /purchase
[admin@demomrv ~]$ sudo mount -t auto /dev/nvme0n2p6 /purchase
[11697.224683] XFS (nvme0n2p6): Mounting V5 Filesystem 0ea75a86-45b0-47cb-9959-28ed0d7e265b
[11697.247889] XFS (nvme0n2p6): Ending clean mount
[admin@demomrv ~]$
```

This is how now the partitions are available to the users for storing the data.

```
Try :-      cd /sales
           touch file{1..10}
           ls
```

Do you see the 10 files created? Now restart the system using

```
sudo init 6
```

After the system restarts, login as a user with the sudo permissions.  
Now go to the /sales directory and check if the 10 files created earlier are shown?  
No..... Why?

### Making partition mounting permanent

The partitions mounted using the mount command are temporarily mounted. When you restart the Linux OS all these mountings are removed.

Thus to make these mountings permanent i.e. they are available even after the Linux OS restart, you need to add the entries in the /etc/fstab file. The kernel reads this file each time the Linux OS starts. Then as per the configuration it will mount those drives.

Thus perform following steps to make the above partition mounting permanent.

**sudo vi /etc/fstab**

At the end of this file add following lines.

/dev/nvme0n2p1	/accounts-data	ext4	defaults	0	0
/dev/nvme0n2p5	/sales	ext3	defaults	0	0
/dev/nvme0n2p6	/purchase	auto	defaults	0	0

The file will look as shown below.

```
#  
# /etc/fstab  
# Created by anaconda on Fri Aug 8 06:39:06 2025  
#  
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.  
#  
# After editing this file, run 'systemctl daemon-reload' to update systemd  
# units generated from this file.  
#  
UUID=7bc3e44f-b9c8-4e34-9c06-309051f11d67 / xfs defaults 0 0  
UUID=794abff4-f947-42b1-a542-918f0fa78775 /boot xfs defaults 0 0  
UUID=33d8fb65-6515-4879-80cf-1ae7b3ee0a40 /home xfs defaults 0 0  
UUID=5f1a39e5-1529-4f8f-832d-d1d96407bf65 none swap defaults 0 0  
/dev/cdrom /mnt/cdrom auto defaults 0 0  
/dev/nvme0n2 /accounts-data ext4 defaults 0 0  
/dev/nvme0n2p5 /sales ext3 defaults 0 0  
/dev/nvme0n2p6 /purchase auto defaults 0 0
```

Save the file.

Verify if the syntax in the file is correct. If the syntax is not proper and if you restart the system, it will not boot and take you to the recovery mode. Thus first make sure the mounting statements are proper.

For this try following commands

**sudo mount /accounts-data**

**sudo mount /sales**

**sudo mount /purchase**

If all commands execute without any error then give the following commands

**ls -all /accounts-data**

**ls -all /sales**

**ls -all /purchase**

If you see the 10 files created in the /sales directory, it means the fstab syntax is correct.  
Now restart the system. Login with a user.  
Check with the **mount** command.

Check if the 10 files are visible in the /sales directory.

This is how you have successfully created the partitions on a hard disk. Formatted them and mounted them.

## CleanUP

Now we will learn how to delete the partitions and remove a hard disk.

[In production environment you need to inform users about the downtime. Make sure you have taken the backup of all the required data. Then only proceed with the deletion process.](#)

First unmount all the partitions on the hard disk that you plan to remove.

In our case there are 3 partitions. Thus give following commands. [Make sure you replace partition names with your system partition names.](#)

```
sudo umount /dev/nvme0n2p1
```

```
sudo umount /dev/nvme0n2p5
```

```
sudo umount /dev/nvme0n2p6
```

```
[admin@demossrv ~]$ sudo umount /dev/nvme0n2p1
[ 1065.650931] EXT4-fs (nvme0n2p1): unmounting filesystem 30d6b830-2a41-4bc5-a834-741dd3966ff0.
[admin@demossrv ~]$ sudo umount /dev/nvme0n2p5
[ 1080.313496] EXT4-fs (nvme0n2p5): unmounting filesystem 708820a4-fa18-4e7c-8898-b837bf8744e6.
[admin@demossrv ~]$ sudo umount /dev/nvme0n2p6
[ 1083.508233] XFS (nvme0n2p6): Unmounting Filesystem 0ea75a86-45b0-47cb-9959-28ed0d7e265b
[admin@demossrv ~]$ _
```

Now the data in the /accounts-data, /sales and /purchase directory will not be visible. As these directories are not required, we will delete them.

```
sudo rm -rf /accounts-data
```

```
sudo rm -rf /sales
```

```
sudo rm -rf /purchase
```

```
[admin@demossrv ~]$ sudo rm -rf /accounts-data
[admin@demossrv ~]$ sudo rm -rf /sales
[admin@demossrv ~]$ sudo rm -rf /purchase
```

Now delete partitions using fdisk. [Make sure you have given the correct hard disk name.](#)

```
sudo fdisk /dev/nvme0n2
```

```
[admin@demossrv ~]$ sudo fdisk /dev/nvme0n2
Welcome to fdisk (util-linux 2.40.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): _
```



First type p at the fdisk prompt and verify that the correct hard disk is selected and the correct partitions are shown.

```
Command (m for help): p
Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6658ae57

Device            Boot      Start         End      Sectors  Size Id Type
/dev/nvme0n2p1                2048     16779263    16777216    8G 83 Linux
/dev/nvme0n2p2    16779264    41943039    25163776   12G  5 Extended
/dev/nvme0n2p5    16781312    27267071    10485760    5G 83 Linux
/dev/nvme0n2p6    27269120    41943039    14673920    7G 83 Linux
```

Now type d at the prompt to delete a partition.  
Type 1 to delete partition 1.

```
Command (m for help): d
Partition number (1,2,5,6, default 6): 1
Partition 1 has been deleted.
```

Again type d to delete remaining partition.

Now only 2 logical partitions are remaining and the extended partition. The logical partitions are the parts of the extended partition, thus deleting the extended partition will automatically delete 2 logical partitions. Thus type 2 to delete extended partition.

```
Command (m for help): d
Partition number (2,5,6, default 6): 2
Partition 2 has been deleted.
```

Verify that all the partitions are deleted. Type p at the prompt. No partition should be shown.

```
Command (m for help): p
Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6658ae57
```

Again all these modifications to the partition table are in the computer memory. You need to save the partition table to the hard disk. (Tip: - Still you have a chance. If you selected a wrong hard disk.....Confirm again. If its wrong hard disk just quit the fdisk utility . do not type w. Instead just type q to exit without saving and save your job.)

As the hard disk is the correct one, we will save the partition table to the disk. Type w at the prompt.

At the prompt give following command to verify that all the partitions are deleted.

**sudo fdisk -l**

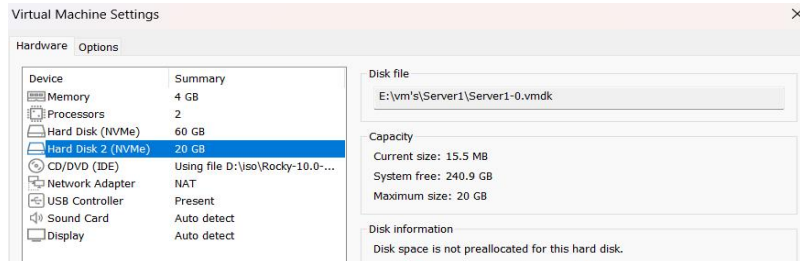
```
Disk /dev/nvme0n2: 20 GiB, 21474836480 bytes, 41943040 sectors
Disk model: VMware Virtual NVMe Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x6658ae57
```

Now shut down the system and remove the hard disk.

For this perform following actions.

```
sudo init 0          # init zero(0) is shutdown
```

Once the system shuts down then go to the settings of the virtual machine.



Select the 20GB hard disk. Click Remove button at the bottom.

Click Ok to close the settings window.

Now power on the virtual machine.



The Linux is taking time to boot and finally comes to the following screen.

```
[ OK ] Started auditd.service - Security Audit Logging Service.
Starting audit-rules.service - Load Audit Rules...
Starting systemd-update-utmp.service - Record System Boot/Shutdown in UTMP...
[ OK ] Finished systemd-update-utmp.service - Record System Boot/Shutdown in UTMP.
Starting systemd-update-utmp-runlevel.service - Record Runlevel Change in UTMP...
[ OK ] Finished audit-rules.service - Load Audit Rules.
[ OK ] Finished systemd-update-utmp-runlevel.service - Record Runlevel Change in UTMP.
[ 105.366554] block dm-0: the capability attribute has been deprecated.
[ 105.521115] block nvme0n1: No UUID available providing old NGUID
[ OK ] Finished kdump.service - Crash recovery kernel arming.
You are in emergency mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, or "exit"
to continue bootup.

Cannot open access to console, the root account is locked.
See slogin(8) man page for more details.

Press Enter to continue.
```

Even if you press Enter. It will take lot of time but fails to start.

```
Reloading system manager configuration.
Starting default.target
You are in emergency mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, or "exit"
to continue bootup.

Cannot open access to console, the root account is locked.
See slogin(8) man page for more details.

Press Enter to continue.
```



This is the common mistake done by the new administrators. While removing a hard disk from a Linux system, all the steps were performed correctly. However you forgot to remove the partition mounting entries from the `/etc/fstab` file.

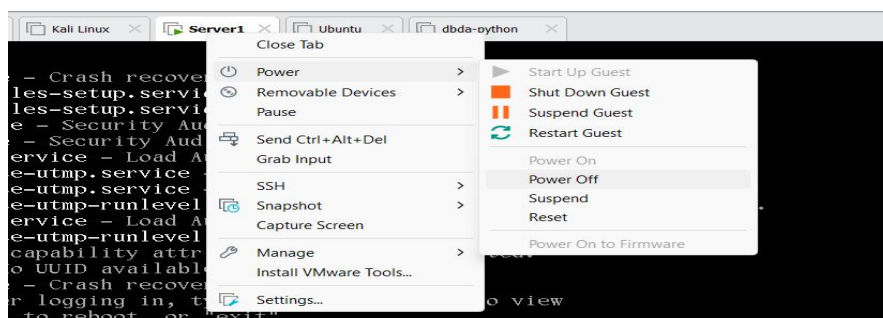
# Who will remove partition mounting entries from the /etc/fstab file.

If there is a problem in the fstab or if Linux kernel is not able to mount a device mentioned in the /etc/fstab file, The system halts. In some versions it may give you a prompt to enter the recovery mode and provide a shell.

But the latest Linux operating systems are more secured. Thus they don't directly provide the shell prompt.

## Recover the system

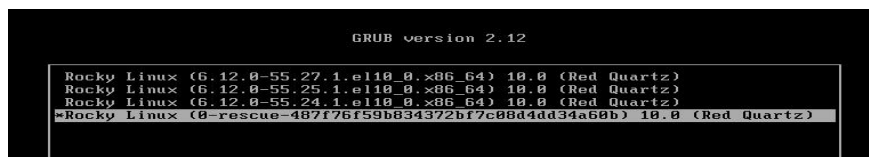
Power Off the virtual machine. Right click the virtual machine name. Select Power. Select Power Off.



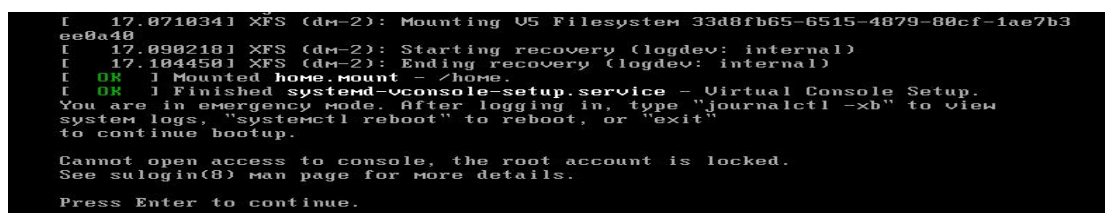
Then Power On the virtual machine.



At this prompt press up arrow key to stop the counter. Else it will automatically start the default operating system. In this select the option with rescue written. In the above screen the the last option is the rescue mode.



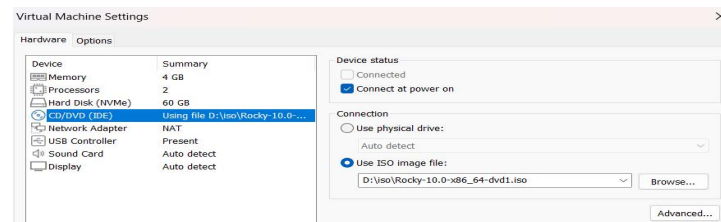
Select it and press Enter. Once the system boots, you will get following prompt.



Press Enter to continue. Again you will get the same prompt. This is because in Rocky Linux 10 the rescue mode is also blocked.

Thus now only option is to boot from the Rocky Linux 10 DVD and access the Linux OS installed.

Power off the virtual machine. Make sure the ISO image is attached to the CD\DVD in the virtual machine. Also the check box for the Connect at Power On is selected.

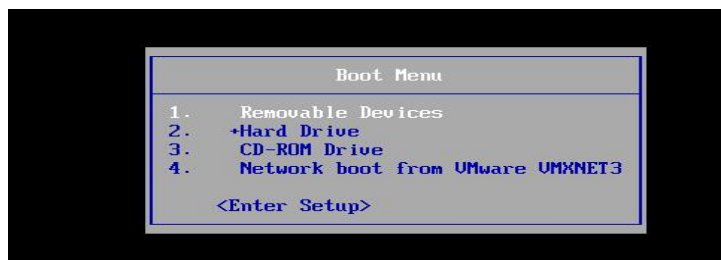


Now close the settings window. Power on the virtual machine.

# It's a test of your reflexes

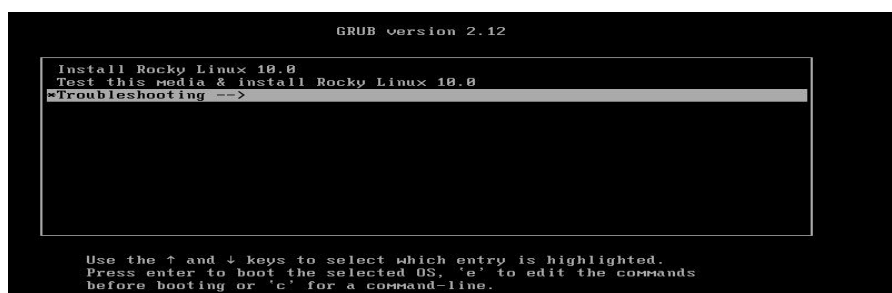
As soon as the virtual machine starts, click in the virtual machine window before the VMWARE logo appears.

As soon as the VMWARE logo appears press Esc key only once. You should get following screen.



Select the CD-ROM Drive option and press Enter.

In the options displayed, Select Troubleshooting.



Press Enter.

On the next Screen, select Rescue a Rocky Linux System.

```
GRUB version 2.12

Install Rocky Linux 10.0 in basic graphics mode
*Rescue a Rocky Linux system
Boot first drive
Boot second drive

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the commands
before booting or 'c' for a command-line. ESC to return
previous menu.
```

Press Enter.

Once the system boots from the DVD, you will get following screen.

```
Starting installer, one moment...
libreport is not available in this environment - bug reporting disabled
anaconda 40.22.3.26-1.el10.rocky.0.2 for Rocky Linux 10.0 started.
* installation log files are stored in /tmp during the installation
* shell is available on TTY2 and in second TUIX pane (ctrl+b, then press 2)
* when reporting a bug add logs from /tmp as separate text/plain attachments
=====
Rescue

The rescue environment will now attempt to find your Linux installation and
mount it under the directory : /mnt/sysroot. You can then make any changes
required to your system. Choose '1' to proceed with this step.
You can choose to mount your file systems read-only instead of read-write by
choosing '2'.
If for some reason this process does not work choose '3' to skip directly to a
shell.

1) Continue
2) Read-only mount
3) Skip to shell
4) Quit (Reboot)

Please make a selection from the above: _
```

Type 1 to continue. After some time following message will be displayed.

```
Rescue Shell

Your system has been mounted under /mnt/sysroot.

If you would like to make the root of your system the root of the active system,
run the command:

    chroot /mnt/sysroot

Warning: The rescue shell will trigger SELinux autorelabel on the subsequent
boot. Add "enforcing=0" on the kernel command line for autorelabel to work
properly.
When finished, please exit from the shell and your system will reboot.

Please press ENTER to get a shell: _
```

Press Enter to get the shell.

```
Please press ENTER to get a shell:
bash-5.2#
```

At this shell prompt give the following command.

**mount | grep sysroot**

```
bash-5.2# mount | grep sysroot
/dev/mapper/r1-root on /mnt/sysroot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
```

In the output look for the entry /mnt/sysroot only. Make sure in the bracket rw is present. It means the hard disk file system is mounted as read write.

Only if it shows ro instead of rw, then give the following command

**mount -o remount,rw /mnt/sysroot**

Our hard disk partition is mounted under /mnt/sysroot directory. Thus the fstab file will be below /mnt/sysroot/etc/fstab. Instead of specifying such a long path we change the root to /mnt/sysroot. Thus when we say / in any filename, the system will replace / with /mnt/sysroot automatically.

**chroot /mnt/sysroot**

```
bash-5.2# chroot /mnt/sysroot
bash-5.2# _
```

Now edit the fstab file and remove all unwanted entries or correct the wrong entries.

**vi /etc/fstab**

```
#
# /etc/fstab
# Created by anaconda on Fri Aug 8 06:39:06 2025
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=7bc3e44f-b9c8-4e34-9c06-309051f11d67 /                xfs     defaults        0 0
UUID=794abff4-f947-42b1-a542-918f0fa78775 /boot           xfs     defaults        0 0
UUID=33d8fb65-6515-4879-00cf-1ae7b3ee0a40 /home           xfs     defaults        0 0
UUID=5f1a39e5-1529-4f0f-032d-d1d96407bf65 none            swap    defaults        0 0
/dev/cdrom      /mnt/cdrom     auto    defaults        0 0
/dev/nvme0n2p1  /accounts-data ext4    defaults        0 0
/dev/nvme0n2p5  /sales         ext3    defaults        0 0
/dev/nvme0n2p6  /purchase      auto    defaults        0 0
```

In our file we do not want the last 3 entries. Thus delete them. And save the file. The final file will look like as shown below.

```
#
# /etc/fstab
# Created by anaconda on Fri Aug 8 06:39:06 2025
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=7bc3e44f-b9c8-4e34-9c06-309051f11d67 /                xfs     defaults        0 0
UUID=794abff4-f947-42b1-a542-918f0fa78775 /boot           xfs     defaults        0 0
UUID=33d8fb65-6515-4879-00cf-1ae7b3ee0a40 /home           xfs     defaults        0 0
UUID=5f1a39e5-1529-4f0f-032d-d1d96407bf65 none            swap    defaults        0 0
```

In RedHat based systems sometimes the SELinux creates the problem. To relable the modified file use following command.

**touch /.autorelabel**

Now type exit till the system restarts.

```
Rocky Linux 10.0 (Red Quartz)
Kernel 6.12.0-55.27.1.el10_0.x86_64 on x86_64

demosrv login: [ 52.331591] perf: interrupt took too long (2966 > 2500), lowering kernel.perf_event_max_
sample_rate to 67000

demosrv login: _
```

# Finally... It's Working !!!