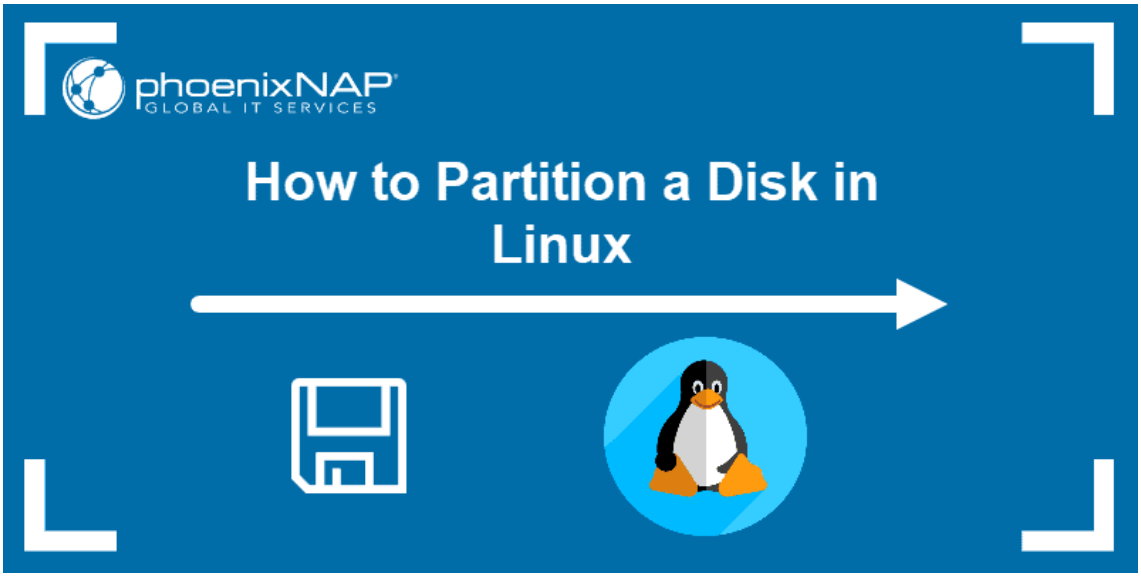


Introduction

Creating disk partitions enables you to split your hard drive into multiple sections that act independently.

In Linux, users must structure storage devices (USB and hard drives) before using them. Partitioning is also useful when you are installing multiple operating systems on a single machine.

In this step-by-step tutorial, you will learn how to create a partition using the Linux `parted` or `fdisk`command.

Prerequisites

- A system running Linux
- A user account with `sudo` or `root` privileges
- Access to a terminal window / command line (**Activities** > **Search** > **Terminal**)

Option 1: Partition a Disk Using parted Command

Follow the steps below to partition a disk in Linux by using the `parted` command.

Step 1: List Partitions

Before making a partition, list available storage devices and partitions. This action helps identify the storage device you want to partition.

Run the following command with `sudo` to list storage devices and partitions:

```
sudo parted -l
```

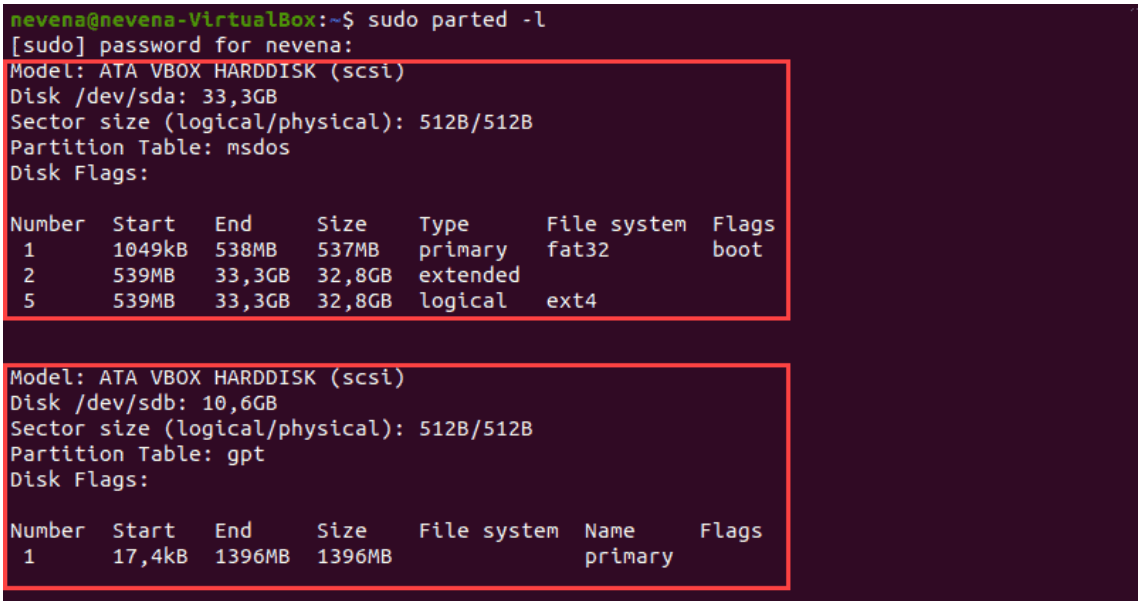
The terminal prints out available storage devices with information about:

- **Model** – Model of the storage device.
- **Disk** – Name and size of the disk.
- **Sector size** – Logical and physical size of the memory. Not to be confused with available disk space.
- **Partition Table** – Partition table type (msdos, gpt, aix, amiga, bsd, dvh, mac, pc98, sun, and loop).
- **Disk Flags** – Partitions with information on size, type, file system, and flags.

Partitions types can be:

- **Primary** – Holds the operating system files. Only four primary partitions can be created.
- **Extended** – Special type of partition in which more than the four primary partitions can be created.
- **Logical** – Partition that has been created inside of an extended partition.

In our example, there are two storage devices (`/dev/sda` and `/dev/sdb`):

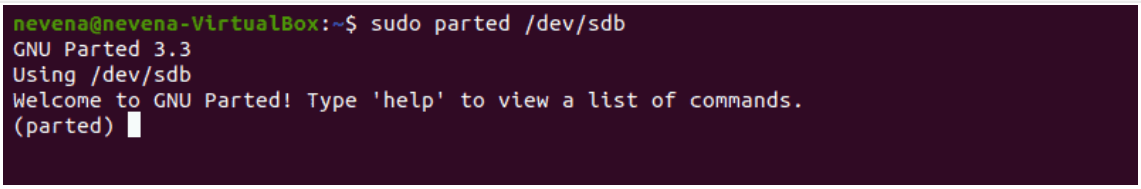


Note: The first storage disk (`dev/sda` or `dev/vda`) contains the operating system. Creating a partition on this disk can make your system unbootable. Only create partitions on secondary disks (`dev/sdb`, `dev/sdc`, `dev/vdb`, or `dev/vdc`).

Step 2: Open Storage Disk

Open the storage disk that you intend to partition by running the following command:

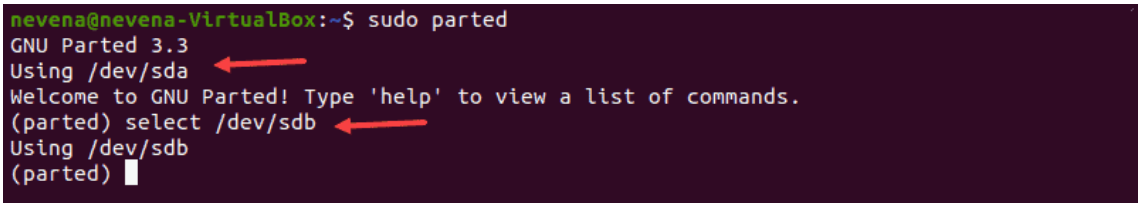
```
sudo parted /dev/sdb
```



Always specify the storage device. If you don't specify a disk name, the disk is randomly selected. To change the disk to `dev/sdb` run:

```
select /dev/sdb
```

The `dev/sdb` disk is open:



Step 3: Make a Partition Table

Create a partition table before partitioning the disk. A partition table is located at the start of a hard drive and it stores data about the size and location of each partition.

Partition table types are: **aix**, **amiga**, **bsd**, **dvh**, **gpt**, **mac**, **ms-dos**, **pc98**, **sun**, and **loop**.

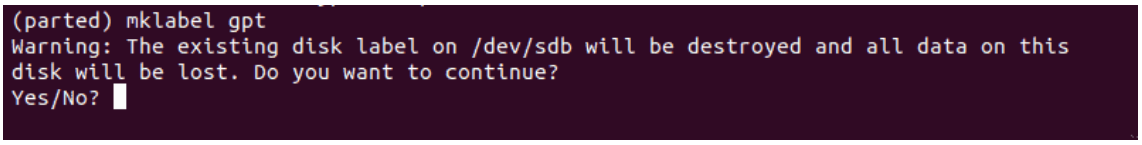
The create a partition table, enter the following:

```
mklabel [partition_table_type]
```

For example, to create a **gpt** partition table, run the following command:

```
mklabel gpt
```

Type **Yes** to execute:



Note: The two most commonly used partition table types are **gpt** and **msdos**. The latter supports up to sixteen partitions and formats up to 16TB of space while gpt formats up to 9.4ZB and supports up to 128 partitions.

Step 4: Check Table

Run the `print` command to review the partition table. The output displays information about the storage device:

```
(parted) print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 10,6GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start  End  Size  File system  Name  Flags

(parted)
```

Note: Run `help mkpart` command to get additional help on how to create a new partition.

Step 5: Create Partition

Let's make a new 1854MB-partition using the ext4 file system. The assigned disk start shall be 1MB and the disk end is at 1855MB.

To create a new partition, enter the following:

```
mkpart primary ext4 1MB 1855MB
```

After that, run the `print` command to review information on the newly created partition. The information is displayed under the *Disk Flags* section:

```
(parted) mkpart primary ext4 1MB 1855MB
(parted) PRINT
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 10,6GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name     Flags
 1      1049kB  1855MB  1854MB  ext4         primary
```

In a gpt partition table, the partition type is the mandatory partition name. In our example, **primary** is the name of the partition, not the partition type.

To save your actions and quit, enter the `quit` command. Changes are saved automatically with this command.

```
(parted) quit
Information: You may need to update /etc/fstab.

nevena@nevena-VirtualBox:~$
```

Note: The *"You may need to update /etc/fstab file"* message signals that the partition can be mounted automatically at boot time.

Option 2: Partition a Disk Using fdisk Command

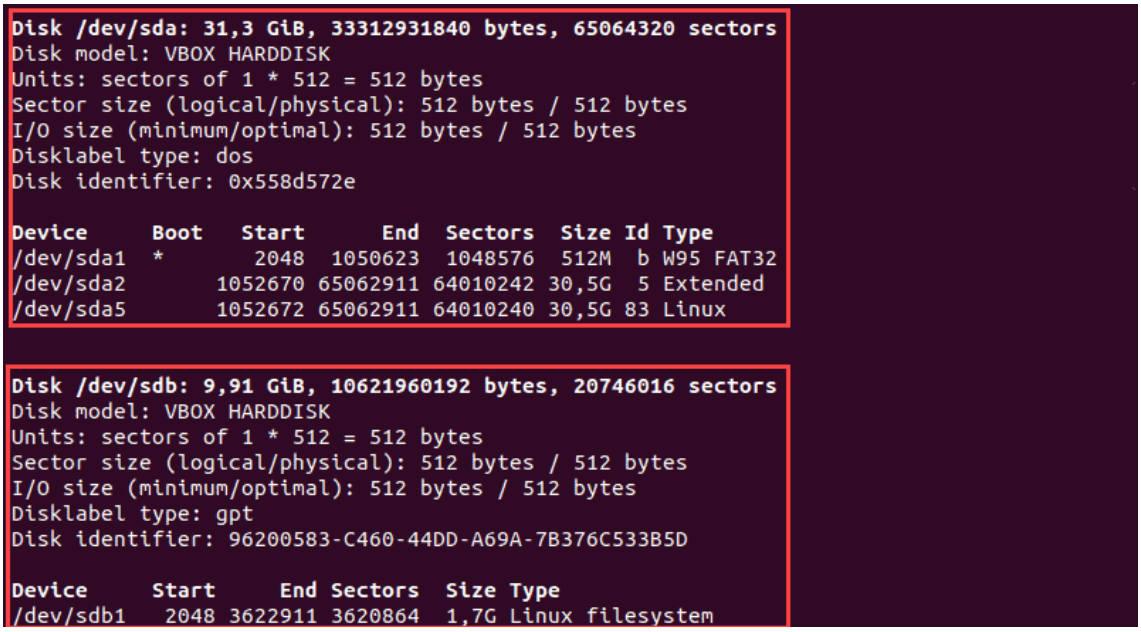
Follow the steps below to partition a disk in Linux by using the `fdisk` command.

Step 1: List Existing Partitions

Run the following command to list all existing partitions:

```
sudo fdisk -l
```

The output contains information about storage disks and partitions:



Step 2: Select Storage Disk

Select the storage disk you want to create partitions on by running the following command:

```
sudo fdisk /dev/sdb
```

The `/dev/sdb`storage disk is open:

```
nevena@nevena-VirtualBox:~$ sudo fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
```

Step 3: Create a New Partition

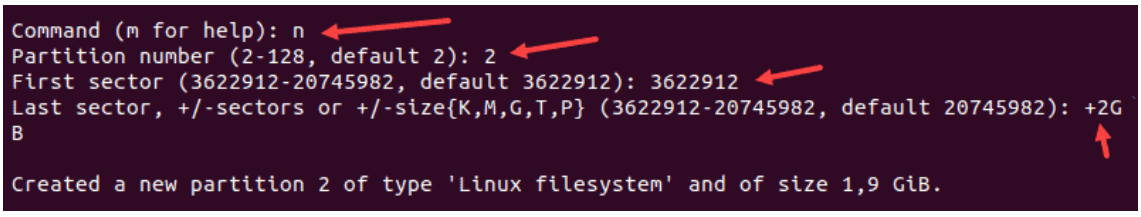
1. Run the `n` command to create a new partition.

2. Select the partition number by typing the default number (2).

3. After that, you are asked for the starting and ending sector of your hard drive. It is best to type the default number in this section (3622912).

4. The last prompt is related to the size of the partition. You can choose to have several sectors or to set the size in megabytes or gigabytes. Type `+2GB` to set the size of the partition to 2GB.

A message appears confirming that the partition is created.



Step 4: Write on Disk

The system created the partition, but the changes are not written on the disk.

1. To write the changes on disk, run the w command:

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

2. Verify that the partition is created by running the following command:

```
sudo fdisk -l
```

As you can see, the partition `/dev/sdb2` has been created.

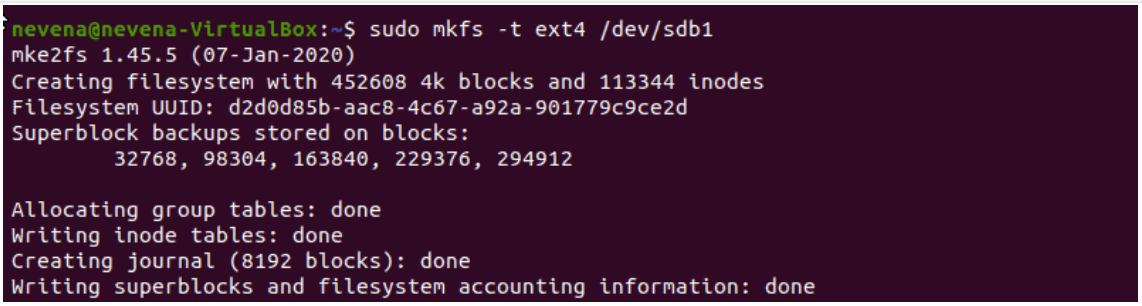
```
Disk /dev/sda: 31,3 GiB, 33312931840 bytes, 65064320 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x558d572e

Device     Boot   Start      End  Sectors  Size Id Type
/dev/sda1  *       2048  1050623  1048576  512M  b W95 FAT32
/dev/sda2       1052670 65062911 64010242 30,5G  5 Extended
/dev/sda5       1052672 65062911 64010240 30,5G 83 Linux


Disk /dev/sdb: 9,91 GiB, 10621960192 bytes, 20746016 sectors
Disk model: VBOX HARDDISK
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: 96200583-C460-44DD-A69A-7B376C533B5D

Device       Start     End Sectors  Size Type
/dev/sdb1     2048 3622911 3620864  1,7G Linux filesystem
/dev/sdb2  3622912 7528447 3905536  1,9G Linux filesystem
```

Format the Partition

Once a partition has been created with the `parted` of `fdisk` command, format it before using it.

Format the partition by running the following command:

```
sudo mkfs -t ext4 /dev/sdb1
```



Note: Check out our guide and learn how to format and mount disk partitions in Linux using ext4, FAT32, or NTFS file system!

Mount the Partition

To begin interacting with the disk, create a **mount point** and **mount the partition** to it.

1. Create a mount point by running the following command:

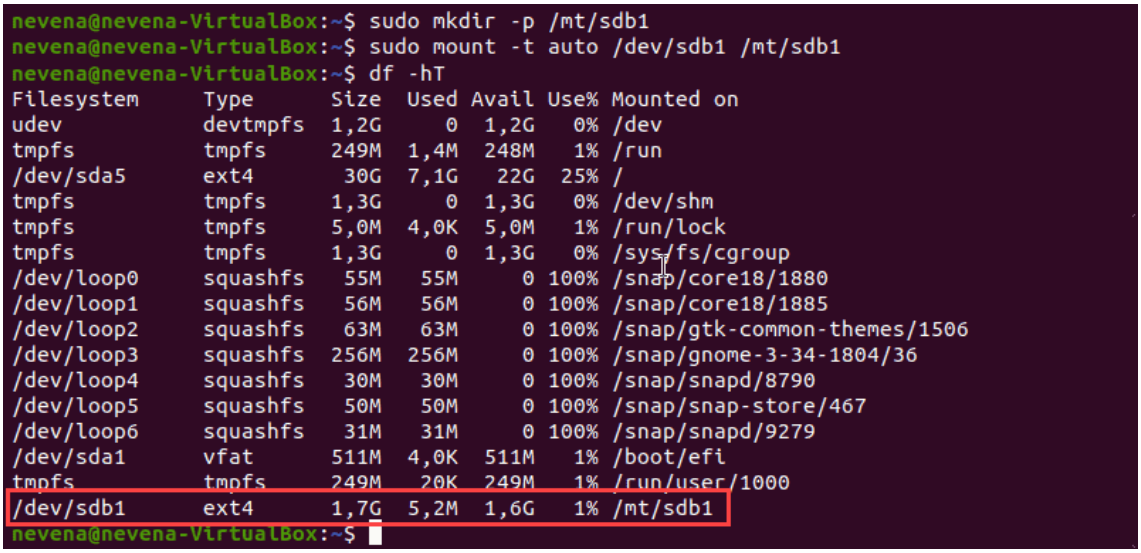
```
sudo mkdir -p /mt/sdb1
```

2. After that, mount the partition by entering:

```
sudo mount -t auto /dev/sbd1 /mt/sdb1
```

The terminal does not print out an output if the commands are executed successfully.

3. Verify if partition is mounted by using the `df hT` command:



Note: If you have NTFS partitions on your hard drive, check out our article on how to mount NTFS partitions in Linux.

Conclusion

After following this step-by-step tutorial, you should have a better understanding on how to partition a disk in Linux by using the `parted` or `fdisk` command.