8 Parted Commands to Manage Disk Partitions in Linux

Marin Todorov | Last Updated: July 13, 2023 | Read Time: 7 mins | Linux Commands | 19 Comments

Parted is a renowned command-line utility designed to facilitate the management of hard disk partitions in a user-friendly manner.

With **Parted**, you can effortlessly perform tasks such as adding, deleting, shrinking, and extending disk partitions, while also managing the file systems associated with them.

Over time, **Parted** has undergone significant development and evolution, introducing various enhancements and changes to its functionality. Certain features have been retired, while new capabilities have been introduced, rendering it a versatile tool for partition management.

This tutorial aims to provide a comprehensive introduction to Parted, covering its fundamental concepts and demonstrating practical examples. If you are new to Parted, it is essential to note that any modifications made using Parted are immediately written to the disk.

Therefore, it is crucial to exercise caution while attempting to modify your disk partitions to avoid unintended consequences or data loss. Throughout this tutorial, we will guide you step-by-step, ensuring that you grasp the basics of Parted and understand the potential implications of your actions when working with disk partitions.

If you intend to experiment with **Parted**, it is recommended to utilize a virtual machine or an old computer/laptop that does not contain any critical data. When making modifications to a disk partition, it is crucial that the partition is not actively in use.

In the case of primary partitions, you may consider booting into rescue mode, which provides a safe environment for performing partition-related tasks without interference from the running operating system. This precautionary approach ensures the integrity of your valuable data and minimizes the risk of unintended consequences during partition manipulation.

Note: You will need to have root access to the machine you will be working on in order to use parted.

How to Install Parted on Linux

On many Linux distributions, **parted** comes pre-installed. If it is not included in your distro, you can install it with:

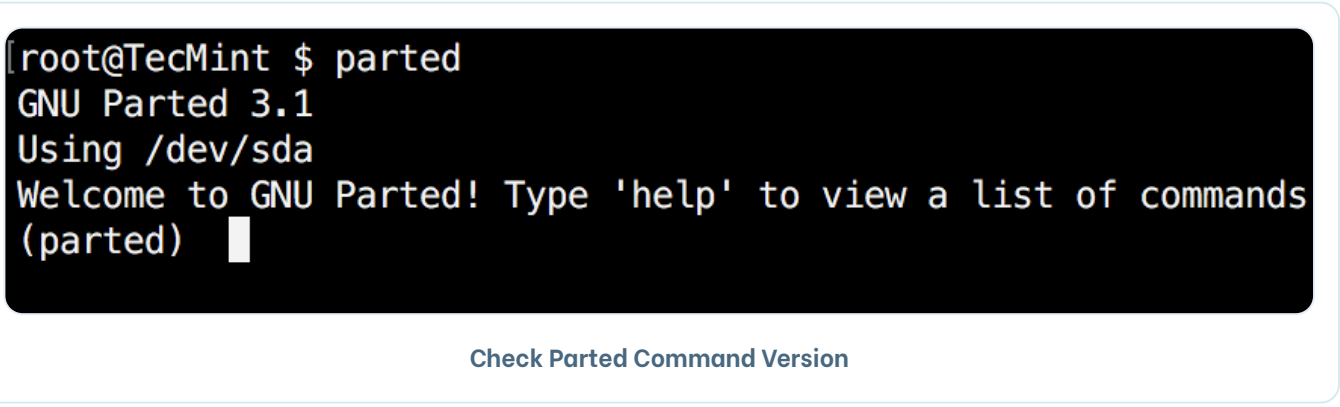
```
$ sudo apt install parted          [On Debian, Ubuntu and Mint]
$ sudo yum install parted          [On RHEL/CentOS/Fedora and Rocky/AlmaLinux]
$ sudo emerge -a sys-apps/parted   [On Gentoo Linux]
$ sudo apk add parted              [On Alpine Linux]
$ sudo pacman -S parted            [On Arch Linux]
$ sudo zypper install parted       [On OpenSUSE]
```

Once you have made sure that **parted** is installed, you can proceed further to check out some real-world examples of parted command in the rest of this article.

1. Check Parted Version

To observe a similar message as depicted in the image below, execute the following command. Please note that the output may differ slightly depending on your specific version of Parted. By default, Parted will operate on your primary drive, typically identified as `/dev/sda`, unless otherwise specified.

```
$ parted
```



Check Parted Command Version

If you want to exit parted, simply type:

```
$ quit
```

2. List Disk Partitions in Linux

Now that **parted** is started, let's list the partitions of the selected hard disk. As mentioned earlier, parted chooses your first drive by default. To see the disk partitions run `print`.

```
(parted) print
```

```
(parted)  print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sda: 53.7GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  525MB   524MB   primary  xfs          boot
 2      525MB   53.7GB  53.2GB  primary               lvm

(parted)
```

Check Linux Partitions

When running `print`, it will also display the hard disk information and model. Here is an example from a real hard disk (not virtual as shown in the image above) :

```
(parted) print

Model: ATA TOSHIBA MQ01ACF0 (scsi)
Disk /dev/sda: 320GB
Sector size (logical/physical): 512B/4096B
Partition Table: msdos

Number  Start   End    Size   Type      File system  Flags

 1      1049kB  256MB  255MB  primary   ext2         boot
 2      257MB   320GB  320GB  extended
 5      257MB   320GB  320GB  logical               lvm
```

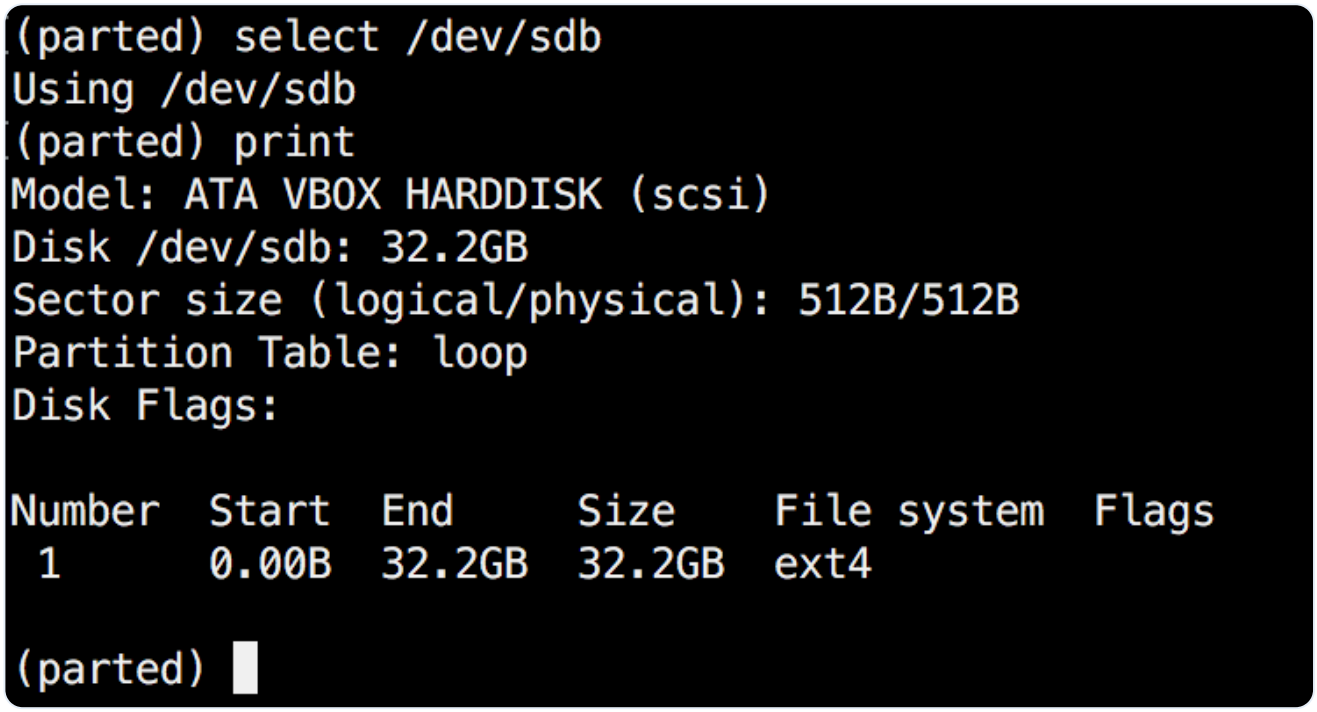
In the example above, you can see the disk model, capacity sector size, and partition table.

3. Change or Switch Partition in Linux

If you have more than one hard disk, you can easily switch between disks, by using the "`select`" command. In the example below, I will switch from `/dev/sda` to `/dev/sdb` which is a secondary drive on my system.

To easily switch between disks you can use:

```
(parted) select /dev/sdX
```



Switch Linux Partitions

Change `"X"` with the letter of the disk to which you wish to switch.

4. Create Partition in Linux

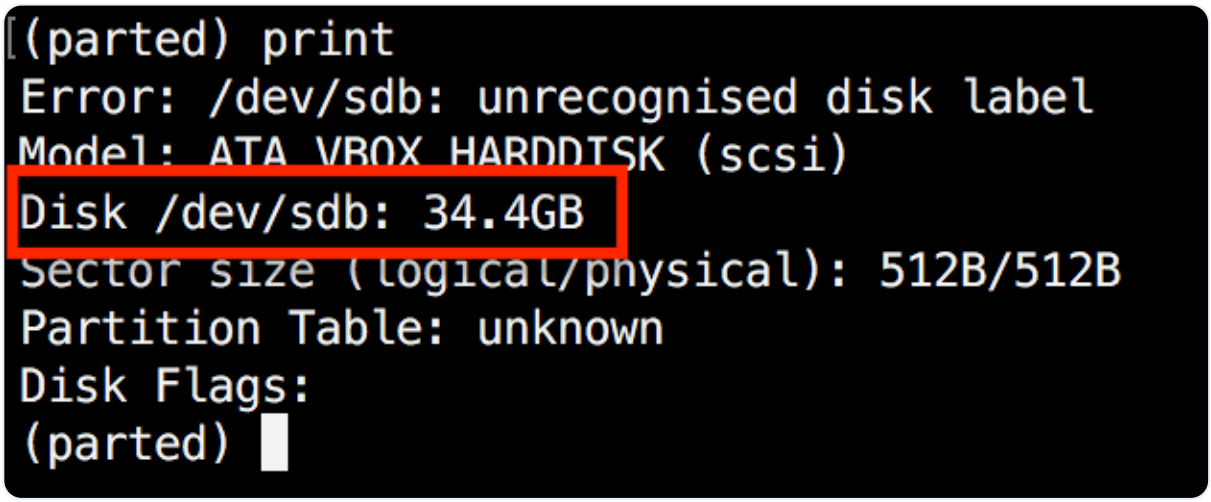
The **parted** can be used to create primary and logical disk partitions. In this example, I will show you how to create a primary partition, but the steps are the same for logical partitions.

To create a new partition, parted uses "`mkpart`". You can give it additional parameters like `"primary"` or `"logical"` depending on the partition type that you wish to create.

Before you start creating partitions, it's important to make sure that you are using (you have selected) the right disk.

Start by using print:

```
(parted) print
```



Show Current Linux Disk

As shown in the above image, we are using a virtual drive of **34 GB**. First, we will give the new disk a label and then create a partition and set a file system on it.

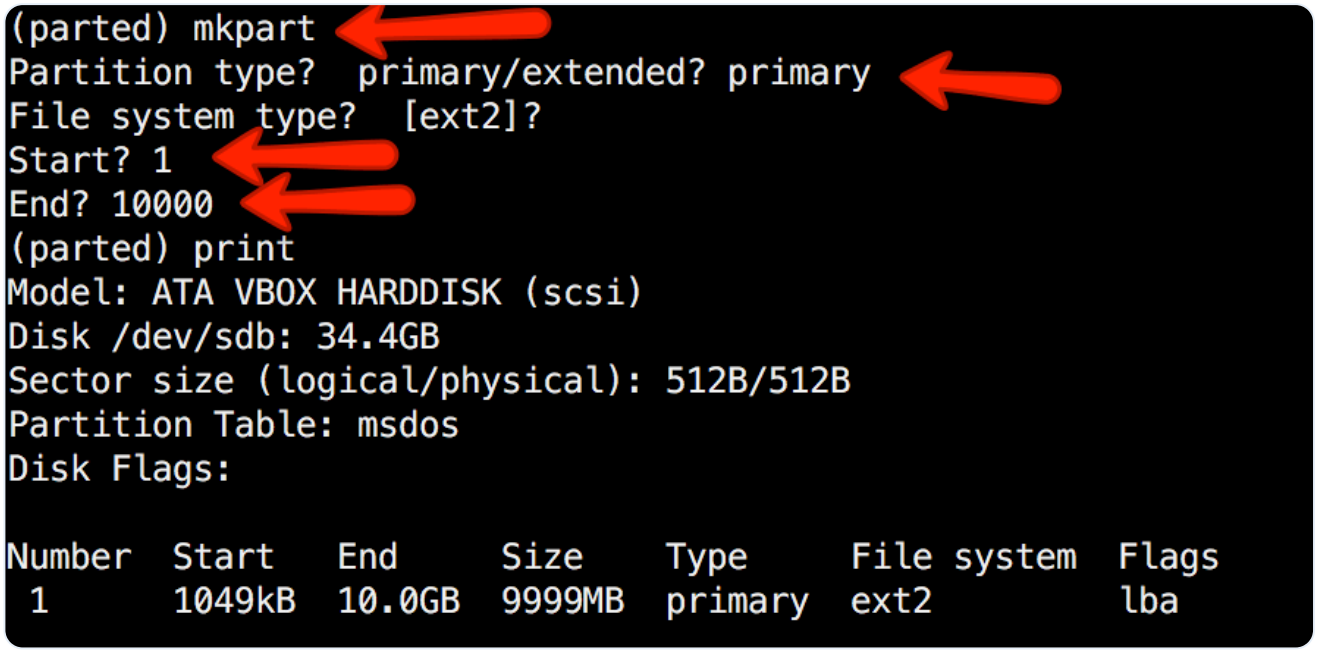
Now the first step is to give the new disk a label name with:

```
(parted) mklabel msdos
```

Now create the new partition with `mkpart`. The listed units are in megabytes (**MB**). We will create a **10 GB** partition starting from **1** to **10000**:

```
(parted) mkpart

Partition type?  primary/extended? primary
File system type?  [ext2]?
Start? 1
End? 10000
(parted) print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 34.4GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:
Number  Start   End     Size    Type     File system  Flags
 1      1049kB  10.0GB  9999MB  primary  ext2         lba
```



Create Primary or Logical Linux Partitions

Next, exit parted with `"quit"` command. We will format our new partition in the ext4 file system using `mkfs`. To make this happen run the following command:

```
# mkfs.ext4 /dev/sdb1
```

Note: It's important to select the right disk and partition when executing the above command!

Now let's verify our results, by printing the partition table on our secondary disk. Under file system column, you should see ext4 or the file system type that you have decided to use for your partition:

```
root@TecMint:~# parted
GNU Parted 3.2
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) select /dev/sdb
sdb   sdb1
(parted) select /dev/sdb
Using /dev/sdb
(parted) print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 34.4GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  10.0GB  9999MB  primary  ext4

(parted)
```

Verify Disk Partition Filesystem

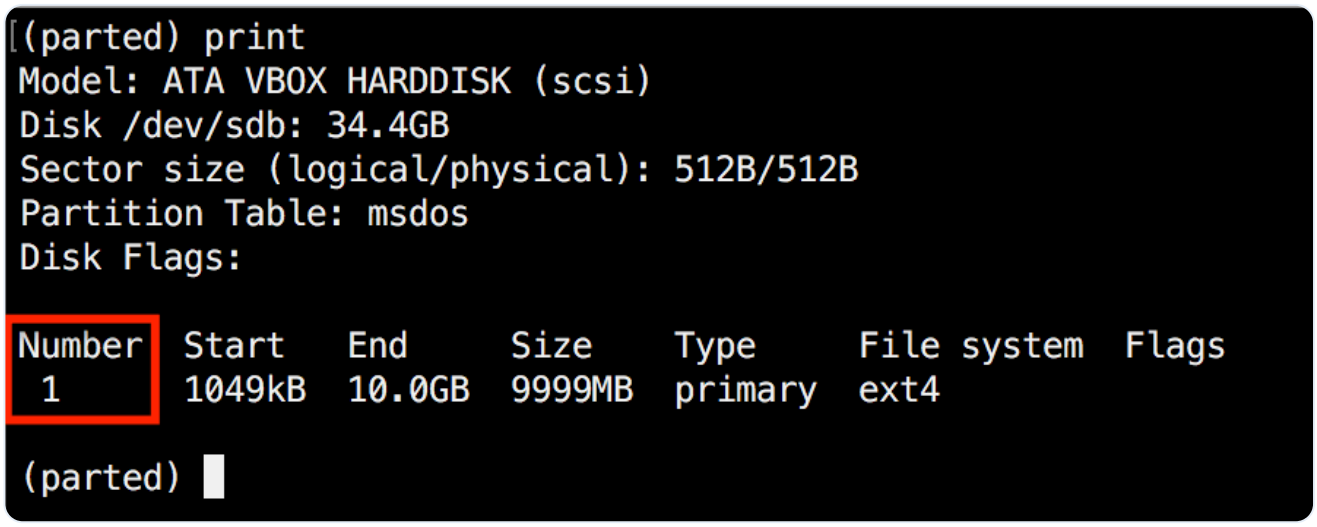
5. Resize Linux Disk Partition

Parted includes multiple useful functions and one of them is `"resizepart"`. As you have probably figured this out by now, `"resizepart"` helps you resize a partition.

In the example below, you will see how to resize an existing partition. For the purpose of this example, we will be using the earlier created partition.

First, you will need to know the number of the partition that you will be resizing. This can be easily found by using `"print"`:

```
(parted) print
```



Find Linux Partition Number

In our example, the partition number is `"1"`. Now run the resizepart command:

```
(parted) resizepart
```

You will be asked for the number of the partition that you will resize. Enter its number. After that, you will be asked to set the new ending point for this partition. Remember that by default the units are in **MB**. In our example, we have set the new partition size to **15 GB**:

```
(parted) resizepart
Partition number? 1
End?  [10.0GB]? 15000
```

Now verify the results with `"print"`:

```
(parted) print
```

```
(parted) resizepart
Partition number? 1
End?  [10.0GB]? 15000
(parted) print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 34.4GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  15.0GB  15.0GB  primary  ext4
```

Verify Linux Resize Partition

6. Delete Linux Partition

The next thing you will learn is how to delete a partition from your hard drive. To do this, you will need to use the rm command within parted. To delete a disk partition you will need to know its number.

As mentioned earlier, you can easily obtain this number by using `"print"`. In our example, we will delete the partition with a number `1` from our secondary drive `/dev/sdb1`:

```
(parted) rm 1
```

Verify the results by printing the partitions table:

```
(parted) print
Model: Unknown (unknown)
Disk /dev/sdb1: 15.0GB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:

Number  Start  End     Size    File system  Flags
 1      0.00B  15.0GB  15.0GB  ext4

(parted) rm 1
(parted) print
Model: Unknown (unknown)
Disk /dev/sdb1: 15.0GB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:

Number  Start  End  Size  File system  Flags

(parted)
```

Delete a Linux Partition

7. Rescue Linux Disk Partition

Parted supports a "`rescue"` utility that helps you recover a lost partition between a starting and ending point. If a partition is found within that range, it will attempt to restore it.

Here is an example:

```
(parted) rescue
Start? 1
End? 15000
(parted) print
Model: Unknown (unknown)
Disk /dev/sdb1: 15.0GB
Sector size (logical/physical): 512B/512B
Partition Table: loop
Disk Flags:
```

```
Number Start End Size File system Flags
1 0.00B 15.0GB 15.0GB ext4
```

8 Change Linux Partition Flag

Using parted, you can change the state of a flag for disk partitions. The supported flags are:

1. boot
2. root
3. swap
4. hidden
5. raid
6. lvm
7. lba
8. legacy_boot
9. irst
10. esp
11. palo

The states can be either `"on"` or `"off"`. To change a flag simply run `"set"` command within parted:

```
(parted) set 2 lba on
```

The above command sets `lba` flag to on for the second partition. Verify the results with `print`:

```
(parted) set 2 lba on
(parted) print
Model: ATA VBOX HARDDISK (scsi)
Disk /dev/sdb: 34.4GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  15.0GB  15.0GB  primary  ext4
 2      15.0GB  16.0GB  1000MB  primary               lba

(parted)
```

Change Partition Flag

Conclusion

Parted is a useful and powerful utility that can help you manage your disk partitions in Linux systems. As always, when working with disk partitions you need to be extra careful.

It is strongly recommended to go through parted man pages to learn how you can customize its output and find more information about its capabilities.

If you have any questions or comments, please do not hesitate to use the comment section below.

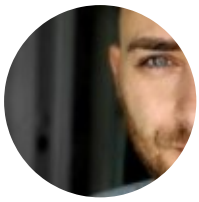
Linux Tricks

PREVIOUS ARTICLE:

TLDR – Simplifying Linux Commands with User-Friendly Man Pages

NEXT ARTICLE:

How to Delete HUGE (100-200GB) Files in Linux



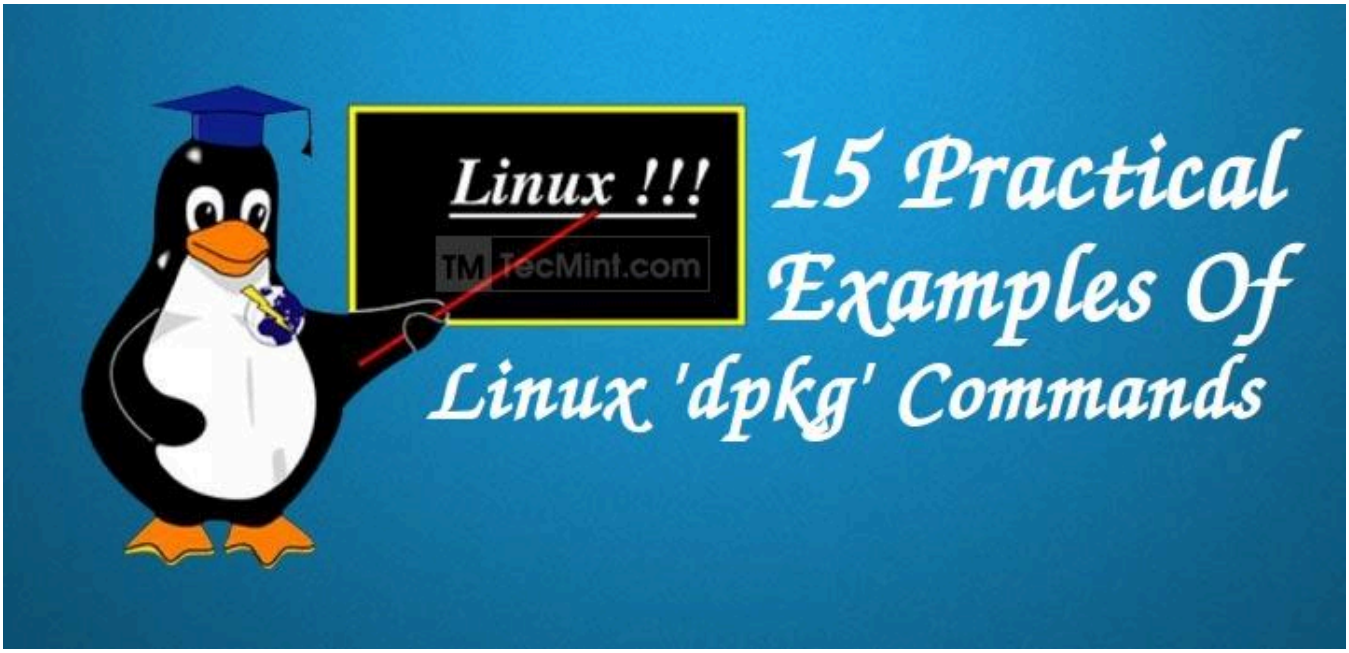
Marin Todorov

I am a bachelor in computer science and a Linux Foundation Certified System Administrator. Currently working as a Senior Technical support in the hosting industry. In my free time I like testing new software and inline skating.

*Each tutorial at **TecMint** is created by a team of experienced Linux system administrators* so that it meets our high-quality *standards.*

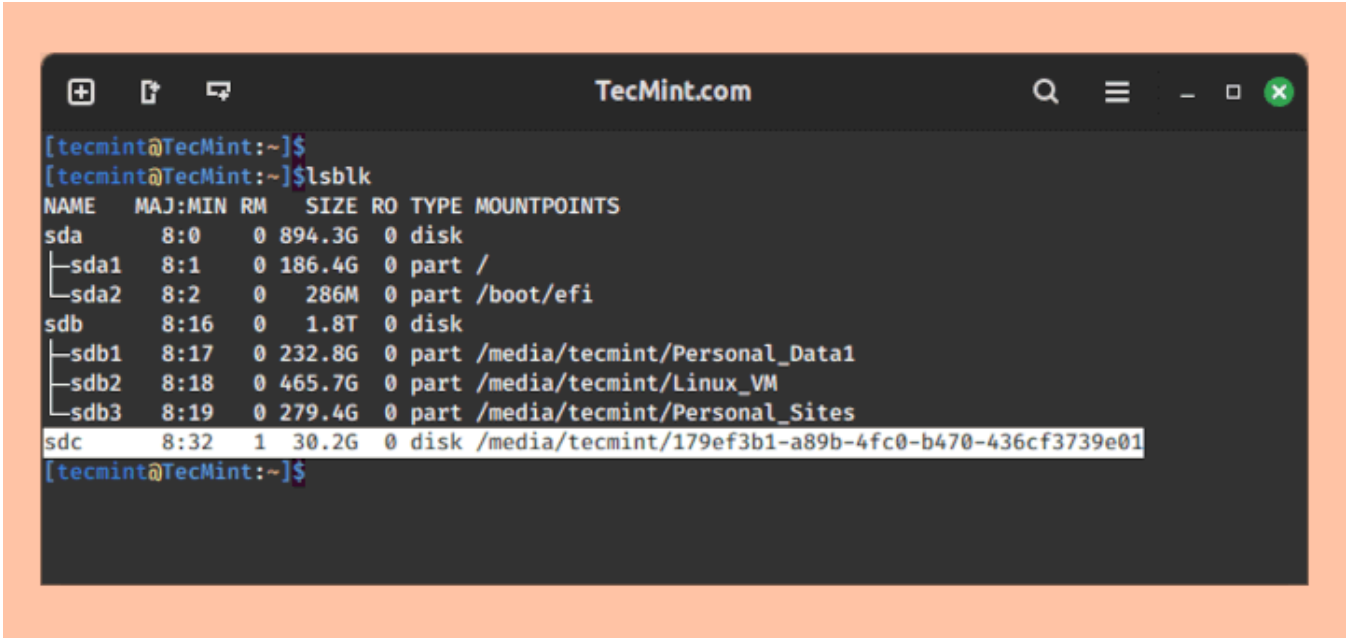
Related Posts

Mutt – A Command Line Email Client to Send Mails from Terminal

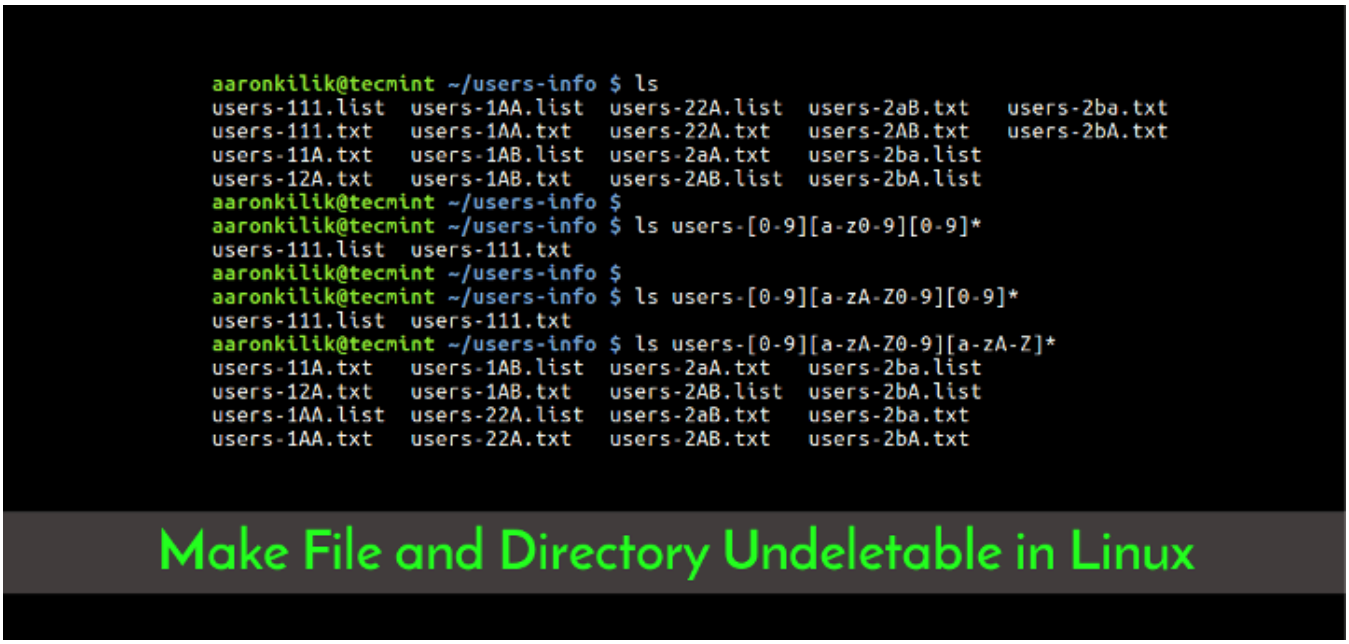


15 Useful ‘dpkg’ Commands for Debian/Ubuntu Users

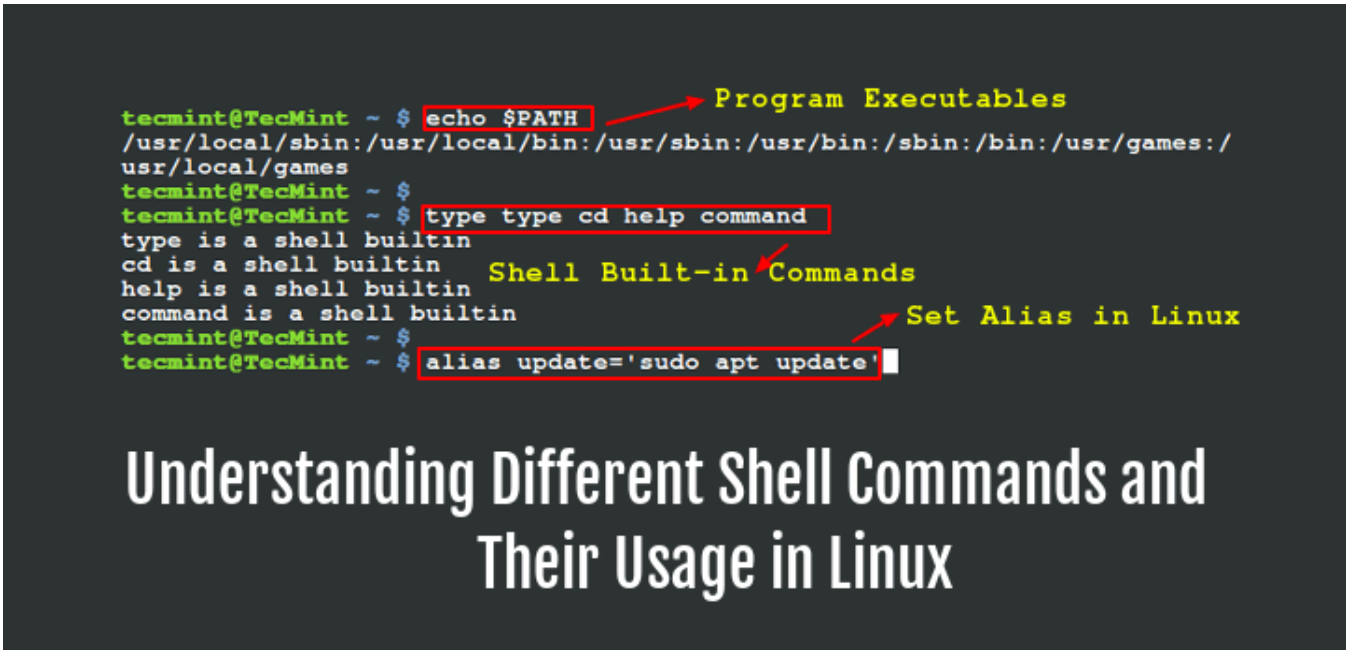
Sysadmins & DevOps: 20 Must-Know Advanced Linux Commands



How to Safely Eject a USB Drive in Linux



How to Make File and Directory Undeletable, Even By Root in Linux



5 Different Types of Shell Commands and Their Usage in Linux

19 Comments

Leave a Reply

mofixx

November 26, 2021 at 12:34 pm

Do I need to unmount the partition when I resize it?

Reply

mofixx
November 26, 2021 at 12:35 pm

*unmount command.

Reply

Kannan Vellaichamy
February 9, 2021 at 5:14 pm

You can update the below point as well.

parted command mainly use to create a partition larger than 2Tb which we can't do with fdisk.

Reply

sedlav
June 21, 2020 at 11:38 pm

Parted is not a Linux command is a GNU command due it's part of GNU project please be specific about this issue

Reply

Abhishek Padghan

January 29, 2019 at 2:15 am

What if I forgot to mention mklabel in parted will it cause any issues ?

Reply

Kier

June 8, 2018 at 1:16 am

Do you need free space or unallocated space to resize a partition? i want to resize a partition but all space is allocated to some partitions or another

Reply

Ethan

April 28, 2017 at 8:54 am

Uhh, before resizepart, don't you need to run resize2fs first to shrink the data?

Reply

Kiers

June 8, 2018 at 1:18 am

can you provide an example how to embed “resize2fs” in the parted commands shown above? Thanks!

Reply

Wellington Torrejais da Silva
February 13, 2017 at 6:14 pm

Thanks!

Reply

Henry
August 30, 2016 at 3:53 am

GNU Parted 3.1
This version don’t have resizepart command!!

Reply

Admin

Ravi Saive
August 30, 2016 at 11:12 am

@Henry,

Thanks for updating us, but whats the alternative command to resizepart in GNU Parted 3.1?

Reply

Kumar Nikhil

May 18, 2016 at 10:08 am

resizepart not working in 2.1 edition. why it so?

Reply

Marin Todorov

May 18, 2016 at 1:49 pm

Version 2.1 is an older one and the option in there is called “resize”. See below:

$ parted -v
parted (GNU parted) 2.1

$ parted -h | grep resize
resize NUMBER START END resize partition NUMBER and its file

It’s always a good practice to check command’s manual and help list and look for additional options you may need.

Reply

Matt M.

October 15, 2016 at 1:28 am

resize and resizepart aren’t included in the RHEL/CentOS versions of parted, for whatever reason.

Reply

Marcin

August 10, 2018 at 4:41 pm

For RHEL, resize function was added in version 3.1-29

* Thu Aug 10 2017 Brian C. Lane – 3.1-29
– Add support for NVMe devices
Resolves: rhbz#1316239
– Backport partition resize command
Resolves: rhbz#1423357

Reply

Ino Van Winckel

April 7, 2016 at 2:44 pm

Hey

I tried to resize my primary partition and it worked, the only problem i run into now is that linux thinks the whole partition is full but I know it's all free space... How can I allocate this free space on my HD into the primary partition and use it as free space?

Reply

Author

Marin Todorov

April 7, 2016 at 4:26 pm

Hello Ino,

Do you mean that the new partition is showed as allocated? If the whole partition is free, did you try to format it? Alternatively, you may try to run resize2fs on the partition you are resizing and see if this resolves the issue for you.

Reply

Jalal Hajigholamali
February 7, 2016 at 10:43 am

Hi,
Very nice article

Thanks a lot

Reply

K Raghunathan
February 1, 2016 at 8:58 pm

Useful list of parted functions. Resizing, particularly, needed when os has been updated repeatedly. Thanks

Reply

Got Something to Say? Join the Discussion...

Thank you for taking the time to share your thoughts with us. We appreciate your decision to leave a comment and value your contribution to the discussion. It's important to note that we moderate all comments in accordance with our comment policy to ensure a respectful and constructive conversation.

Rest assured that your email address will remain private and will not be published or shared with anyone. We prioritize the privacy and security of our users.

Name *

Email *

Save my name, email, and website in this browser for the next time I comment.

Post Comment

Search...

Linux Commands and Tools

How to Run a Linux Command Without Saving It in History

Pssh – Execute Commands on Multiple Remote Linux Servers

3 Command Line Tools to Install Deb Packages in Ubuntu

How to Run Linux Commands in Background and Detach in Terminal

Learn Why 'less' is Faster Than 'more' Command for Effective File Navigation

Diskonaut – A Terminal Disk Space Navigator for Linux

Linux Server Monitoring Tools

How to Install Zabbix Monitoring Tool on Debian 11/10

LibreNMS – A Fully Featured Network Monitoring Tool for Linux

How to Install Icinga2 Monitoring Tool on OpenSUSE

How to Install vnStat and vnStati to Monitor Network Traffic in Linux

Limit CPU Usage of a Process in Linux with CPULimit Tool

How to Setup and Manage Log Rotation Using Logrotate in Linux

Learn Linux Tricks & Tips

How to Delete Root Mails (Mailbox) File in Linux

2 Ways to Re-run Last Executed Commands in Linux

How to Manage User Password Expiration and Aging in Linux

How to Set and Unset Local, User and System Wide Environment Variables in Linux

10 Practical Examples Using Wildcards to Match Filenames in Linux

12 Useful PHP Commandline Usage Every Linux User Must Know

Best Linux Tools

10 Top Open Source Caching Tools for Linux in 2024

5 Must-Try AI Tools for Linux Users in 2024

16 Best Markdown Editors for Linux

5 Best PDF to Word Converters for Linux

5 Open Source Log Monitoring and Management Tools for Linux

8 Best RDP (Remote Desktop) Clients for Linux in 2024