



Level Up Linux: 20 Must-Know Commands for Newbies

Ravi Saive | Last Updated: January 29, 2024 | Read Time: 13 mins | [Linux Commands](#) | [78 Comments](#)

So, are you planning to switch from Windows to Linux or have you recently made the switch to Linux? Oops! What am I asking? Why else would you be here?

[You might also like: [How I Switched from Windows to Linux Mint](#)]

In my earlier experience as a newcomer, [Linux commands](#) and [terminals](#) were quite intimidating. I had concerns about the commands and wondered to what extent I needed to [remember and memorize them](#) to become proficient and fully functional with Linux.

Undoubtedly, online documentation, [Linux books](#), [man pages](#), and the user community provided significant assistance.

However, I strongly believe that there should be an article [featuring basic Linux commands](#) in easy-to-learn and understood language.

These motivations inspired me to master Linux and make it more user-friendly. This article is a step towards that goal."

1. Is Command

The command 'ls' stands for 'List Directory Contents', which is used to display the contents of the folder, whether they are files or subfolders, from which the command is executed.

```
ls
```

The 'ls -l' command lists the contents of the folder in a detailed, long listing format.

```
ls -l
```

The 'ls -a' command lists the contents of a folder, including hidden files that start with `'.'`.

```
ls -a
```

In Linux, a file name starting with `'.'` is considered hidden. In Linux, every file, folder, device, or [command is treated as a file](#).

```

[tecmint@TecMint:~]$
[tecmint@TecMint:~]$ ls
AnyDesk                                ExpressVPN-Review
bin                                    Fooocus
ChatGPT-SEO-Prompts                    jesse
Compile-Debian-Kernel.png              Linux-Hard-Disk-Serial-Number.png
Create-Desktop-Shortcut-Launcher-in-Linux.png
DEBIAN                                  ravi.txt
Desktop                                tecmint.info
Downloads                              tecmint.txt
[tecmint@TecMint:~]$ ls -l
total 332
drwx-----  2 tecmint tecmint   4096 Nov 16 12:03 AnyDesk
drwxrwxr-x  2 tecmint tecmint   4096 Oct 10 10:44 bin
-rw-rw-r--  1 tecmint tecmint   2236 Oct 29 00:17 ChatGPT-SEO-Prompts
-rw-rw-r--  1 tecmint tecmint  69925 Jan 19 14:50 Compile-Debian-Kernel.png
-rw-rw-r--  1 tecmint tecmint  33381 Jan 16 11:04 Create-Desktop-Shortcut-Launcher-in-Linux.png
drwxr-xr-x  2 root    root      4096 Nov 24 2021 DEBIAN
drwxrwxr-x  2 tecmint tecmint   4096 Jan 16 11:01 Desktop
drwx--x---+ 7 tecmint tecmint 102400 Jan 29 11:08 Downloads
-rw-rw-r--  1 tecmint tecmint   1490 Oct 12 10:57 ExpressVPN-Review
drwxrwxr-x 17 tecmint tecmint   4096 Dec 20 15:22 Fooocus
-rw-rw-r--  1 tecmint tecmint   1134 Oct 13 10:17 jesse
-rw-rw-r--  1 tecmint tecmint  76207 Dec 26 11:43 Linux-Hard-Disk-Serial-Number.png
-rw-rw-r--  1 tecmint tecmint     44 Dec 19 11:40 ravi.txt
-rw-rw-r--  1 tecmint tecmint     0 Dec 19 11:38 tecmint.info
-rw-rw-r--  1 tecmint tecmint   1323 Oct 24 14:03 tecmint.txt
-rw-rw-r--  1 tecmint tecmint   1001 Nov  1 10:06 Tips-on-How-to-Update-Old-Articles-with-SEO
[tecmint@TecMint:~]$ ls -al
total 760
drwxr-xr-x 38 tecmint tecmint 69632 Jan 29 11:35 .
drwxr-xr-x 3 root    root    4096 Oct  6 2021 ..
drwxr-xr-x 3 tecmint tecmint 4096 Dec  6 2022 .anydesk
drwx----- 2 tecmint tecmint 4096 Nov 16 12:03 AnyDesk
-rw----- 1 tecmint tecmint 48791 Jan 25 14:45 .bash_history
-rw-r--r-- 1 tecmint tecmint  220 Oct  6 2021 .bash_logout
-rw-r--r-- 1 tecmint tecmint 4572 Jun 27 2023 .bashrc
drwxrwxr-x 2 tecmint tecmint 4096 Oct 10 10:44 bin
drwxrwxr-x 47 tecmint tecmint 4096 Dec 25 09:25 .cache
-rw-rw-r-- 1 tecmint tecmint  2236 Oct 29 00:17 ChatGPT-SEO-Prompts
-rw-rw-r-- 1 tecmint tecmint  69925 Jan 19 14:50 Compile-Debian-Kernel.png
drwxrwxr-x 71 tecmint tecmint 4096 Jan 16 10:59 .config
-rw-rw-r-- 1 tecmint tecmint  33381 Jan 16 11:04 Create-Desktop-Shortcut-Launcher-in-Linux.png
drwxr-xr-x 2 root    root    4096 Nov 24 2021 DEBIAN
drwxrwxr-x 2 tecmint tecmint 4096 Jan 16 11:01 Desktop
-rw-r--r-- 1 tecmint tecmint   23 Oct  6 2021 .dmrc

```

Listing Files in Linux

The result of the `ls -l` command is:

- **File Type** – The first character represents the file type (`'-'` for a regular file, `'d'` for a directory, `'l'` for a symbolic link, etc.).
- **Permissions** – The next nine characters represent the file's permissions for the owner, group, and others. These characters can include `'r'` for read, `'w'` for write, and `'x'` for execute permissions.
- **Number of Links** – Indicates the number of hard links pointing to the file or directory.
- **Owner and Group** – Specifies the user (owner) and group associated with the file or directory.

- **File Size** – Shows the size of the file in bytes.
- **Modification Time** – Displays the date and time when the file or directory was last modified.
- **File or Directory Name** – The actual name of the file or directory.

For more “ls” command examples read our series of articles:

You might also like:

- [15 Basic ls Command Examples in Linux](#)
- [7 Quirky ‘ls’ Command Tricks Every Linux User Should Know](#)
- [How to Sort Output of ‘ls’ Command By Last Modified Date and Time](#)
- [How to List All Files Ordered by Size in Linux](#)

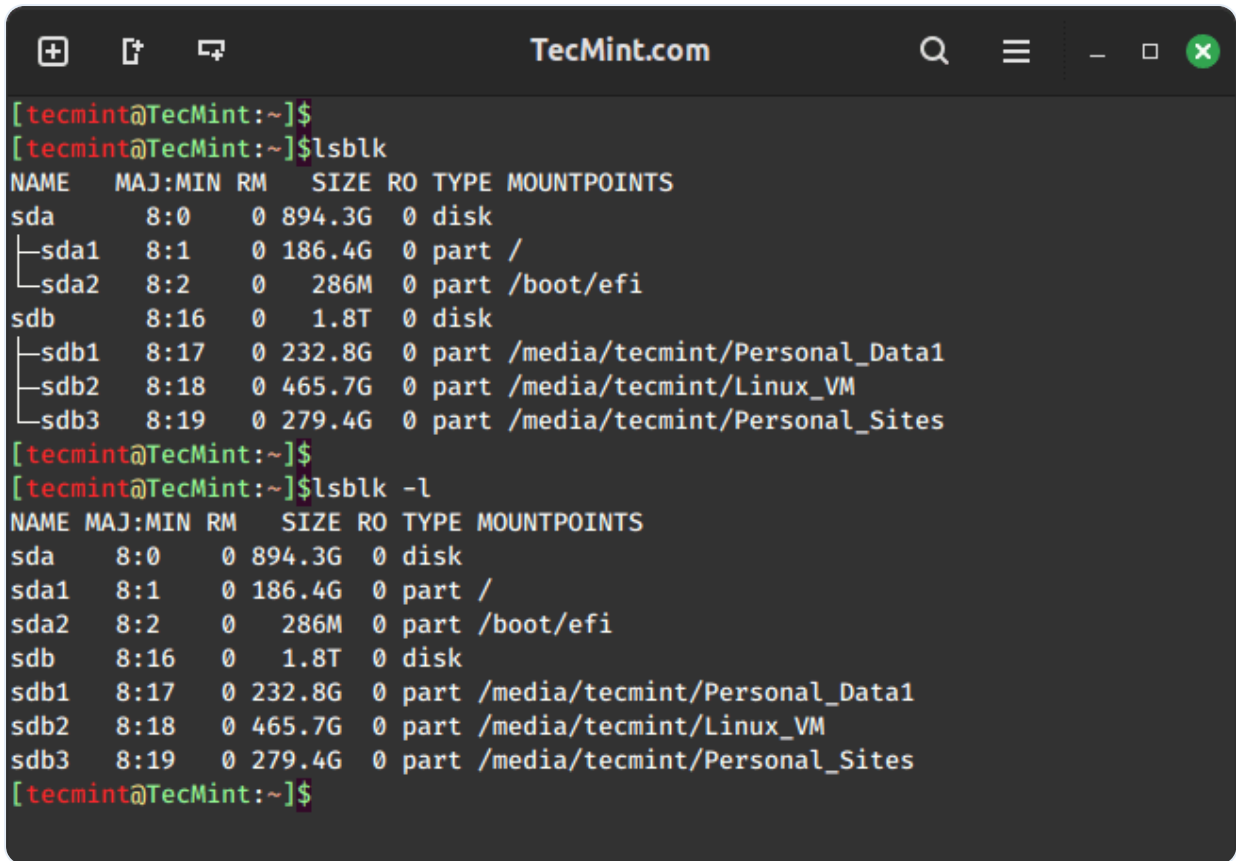
2. lsblk Command

The ‘[lsblk](#)’ command, short for ‘List Block Devices,’ displays block devices by their assigned names (excluding RAM) in a tree-like format on the standard output.

```
lsblk
```

The ‘lsblk -l’ command lists block devices in a ‘list’ structure rather than a tree-like fashion.

```
lsblk -l
```



The screenshot shows a terminal window titled 'TecMint.com' with a dark background. The user is at the prompt [tecmint@TecMint:~]\$ and enters the command lsblk. The output is a table of block devices. The user then enters lsblk -l, and the output is a similar table with more detailed information.

```
[tecmint@TecMint:~]$  
[tecmint@TecMint:~]$lsblk  
NAME      MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS  
sda        8:0    0 894.3G  0 disk  
├─sda1     8:1    0 186.4G  0 part /  
└─sda2     8:2    0   286M  0 part /boot/efi  
sdb        8:16   0    1.8T  0 disk  
├─sdb1     8:17   0 232.8G  0 part /media/tecmint/Personal_Data1  
├─sdb2     8:18   0 465.7G  0 part /media/tecmint/Linux_VM  
└─sdb3     8:19   0 279.4G  0 part /media/tecmint/Personal_Sites  
[tecmint@TecMint:~]$  
[tecmint@TecMint:~]$lsblk -l  
NAME MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS  
sda    8:0    0 894.3G  0 disk  
sda1    8:1    0 186.4G  0 part /  
sda2    8:2    0   286M  0 part /boot/efi  
sdb     8:16   0    1.8T  0 disk  
sdb1    8:17   0 232.8G  0 part /media/tecmint/Personal_Data1  
sdb2    8:18   0 465.7G  0 part /media/tecmint/Linux_VM  
sdb3    8:19   0 279.4G  0 part /media/tecmint/Personal_Sites  
[tecmint@TecMint:~]$
```

List Block Devices

lsblk is a very useful and easy way to [identify the name of the new USB device](#) you just plugged in, especially when you have to work with disks or blocks in the terminal.

[You might also like: [10 Commands to Find Linux Hardware Information](#)]

3. md5sum Command

The 'md5sum' stands for 'Compute and Check MD5 Message-Digest'. MD5 checksum (commonly referred to as a 'hash') is used to match or verify the integrity of files that may have changed due to a faulty file transfer, disk error, or non-malicious interference.

```
md5sum teamviewer_linux.deb  
  
47790ed345a7b7970fc1f2ac50c97002 teamviewer_linux.deb
```

The user can compare the [generated md5sum](#) with the one provided officially. MD5sum is considered less secure than sha1sum, which we will discuss later.

4. dd Command

The [dd command](#) stands for 'Convert and Copy a file' and can be utilized to convert and copy a file. Most often, it is used to copy an ISO file (or any other file) to a USB device (or another location), making it suitable for [creating a bootable USB stick](#).

```
dd if=debian.iso of=/dev/sdb1 bs=512M; sync
```

Note: In the above example the usb device is supposed to be `sdb1` (You should verify it using the [command lsblk](#), otherwise you will overwrite your disk and OS), use the name of the disk very cautiously!

The `dd` command takes some time ranging from a few seconds to several minutes in execution, depending on the size and type of file and read and write speed of the Usb stick.

[You might also like: [How to Clone a Partition Using dd Command](#)]

5. uname Command

The [uname command](#) stands for (Unix Name), and prints detailed information about the machine name, operating system, and kernel version.

```
uname -a
```

```
Linux TecMint 6.2.0-39-generic #40~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC  
Thu Nov 16 10:53:04 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
```

The result of the `uname -a` command is:

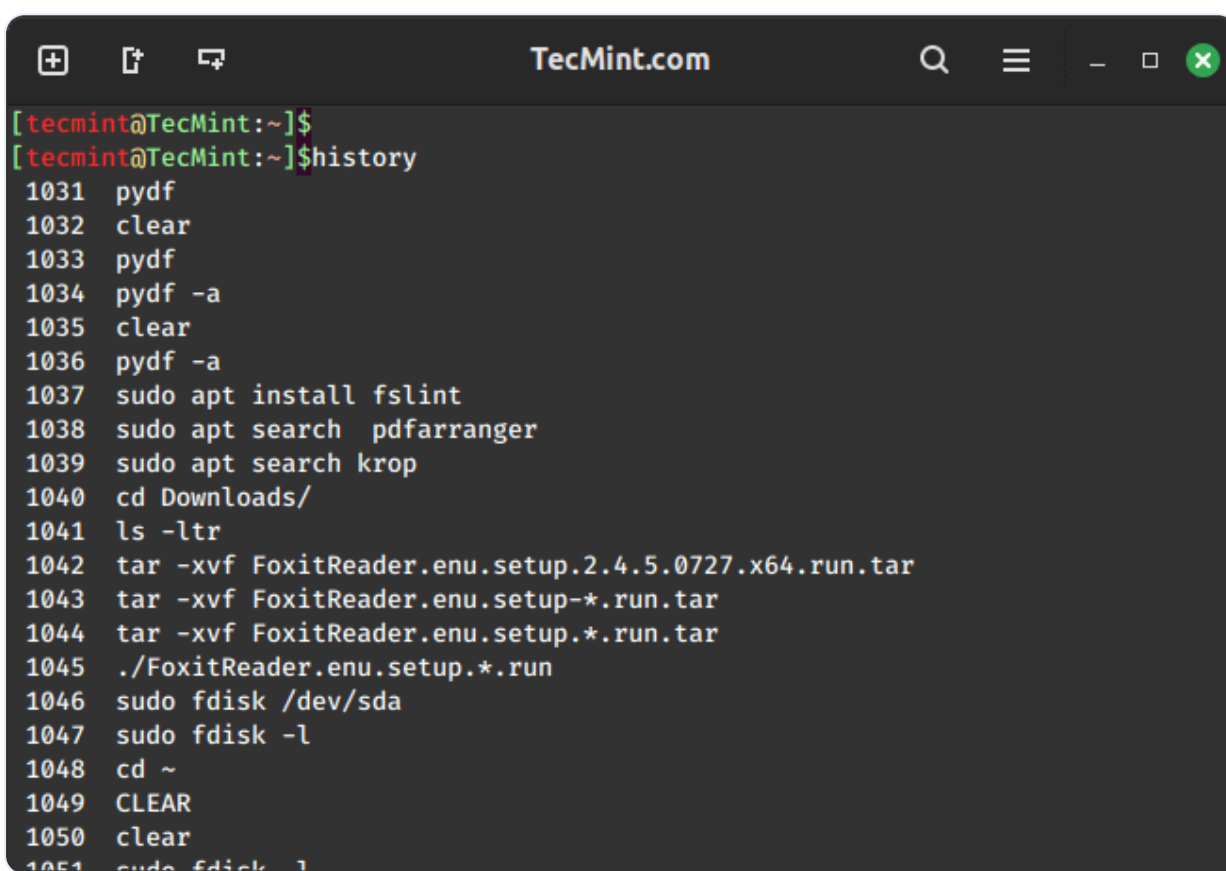
- "Linux": The machine's kernel name.
- "tecmint": The machine's node name.
- "6.2.0-39-generic": The kernel release.
- "22.04.1-Ubuntu SMP": The operating system release version.

- "x86_64": The architecture of the processor.
- "GNU/Linux": The operating system name.

6. history Command

The [history command](#) stands for History (Event) Record, it prints the history of a long list of executed commands in the terminal.

```
history
```



```
[tecmint@TecMint:~]$  
[tecmint@TecMint:~]$history  
1031 pydf  
1032 clear  
1033 pydf  
1034 pydf -a  
1035 clear  
1036 pydf -a  
1037 sudo apt install fslint  
1038 sudo apt search pdfarranger  
1039 sudo apt search krop  
1040 cd Downloads/  
1041 ls -ltr  
1042 tar -xvf FoxitReader.enu.setup.2.4.5.0727.x64.run.tar  
1043 tar -xvf FoxitReader.enu.setup-*.run.tar  
1044 tar -xvf FoxitReader.enu.setup.*.run.tar  
1045 ./FoxitReader.enu.setup.*.run  
1046 sudo fdisk /dev/sda  
1047 sudo fdisk -l  
1048 cd ~  
1049 CLEAR  
1050 clear  
1051 sudo fdisk 1
```

List Previously Executed Commands

Note: Pressing `'Ctrl + R'` allows you to search for [previously executed commands](#), enabling your command to be completed using the auto-completion feature.

```
(reverse-i-search)`if': ifconfig
```

For more examples of history commands, please refer to our series of articles:

You might also like:

- [How to Run a Command Without Saving It in History](#)
- [How to Clear BASH Command Line History in Linux](#)
- [How to Set Date and Time for Command in History](#)

7. sudo Command

The “sudo” (superuser do) command allows a permitted user to execute a command as the superuser or another user, as specified by the security policy in the sudoers list.

```
sudo apt update
```

Note: sudo allows users to borrow superuser privileges, while a similar command ‘su’ allows users to log in as superusers. Sudo is safer than su.

It is not advised to use sudo or su for day-to-day normal use, as it can result in serious errors if accidentally you do something wrong, that’s why a very popular saying in the Linux community is:

```
“To err is human, but to really foul up everything, you need a root password”
```

For more examples of sudo commands, please refer to our series of articles:

You might also like:

- [10 Useful Sudoers Configurations for Setting ‘sudo’ in Linux](#)
- [How to Run ‘sudo’ Without Password in Linux](#)
- [How to Show Asterisks While Typing Sudo Password in Linux](#)

8. mkdir Command

The (make directory) creates a new directory with a name path. However if the directory already exists, it will return an error message “cannot create a folder, folder already exists”.

The [mkdir command](#) (make directory) is used to create a new directory with a specified path. However, if the directory already exists, it will return an error message stating, ‘cannot create a folder, folder already exists’.

```
mkdir tecmint
```

Directories can only be created within folders for which the user has write permissions.

9. touch Command

The [touch command](#) stands for ‘update the access and modification times of each FILE to the current time.’ The ‘touch’ command creates the file only if it doesn’t exist. If the file already exists, it will update the timestamp but not the contents of the file.

```
touch tecmintfile
```

The ‘touch’ command can be used to create a file in a directory for which the user has write permission, only if the file doesn’t already exist there.

10. chmod Command

The “chmod” command stands for “change file mode bits,” which alters the file mode (permissions) of each specified file, folder, script, etc., according to the specified mode.

There exist 3 types of permission on a file (folder or anything but to keep things simple we will be using file).

```
Read (r)=4  
Write(w)=2  
Execute(x)=1
```

So if you want to give only read permission on a file it will be assigned a value of '4', for write permission only, a value of '2' and for execute permission only, a value of '1' is to be given. For read and write permission $4+2 = '6'$ is to be given, and so on.

Now, permission needs to be set for 3 kinds of users and user groups. The first is the owner, then the user group, and finally the world.

```
rwxr-x--x  abc.sh
```

Here the root's permission is rwx (read, write, and execute). usergroup to which it belongs is r-x (read and execute only, no write permission) and for the world is -x (only execute).

To change its permission and provide read, write, and execute permission to the owner, group, and world.

```
chmod 777 abc.sh
```

only read and write permission to all three.

```
chmod 666 abc.sh
```

The read, write, and execute to the owner and only execute to group and world.

```
chmod 711 abc.sh
```

chmod is one of the most crucial commands that proves beneficial for both system administrators and users. In a multi-user environment or on a server, this command is indispensable, as incorrectly setting permissions can render a file inaccessible or grant unauthorized access to individuals.

11. chown Command

The [chown command](#) stands for “change file owner and group”, which is used to change the owner and/or group of a file or directory.

Below is an example of how the chown command is typically used.

```
chown newowner:newgroup filename
```

For instance, if you want to change the owner of a file named “example.txt” to a user named “tecmint” and a group named “users”, the command would be:

```
chown tecmint:users example.txt
```

12. apt Command

On [Debian-based distributions](#) such as Ubuntu and Linux Mint, the [apt command](#) stands for (Advanced Package Tool), which is used to install, upgrade, and manage software packages on a system from the command line.

```
sudo apt search wget  
sudo apt install wget  
sudo apt update
```

The apt command is considered more advanced and intelligent as compared to [yum](#) or [dnf](#) command.

[You might also like: [Understanding APT and APT-Cache Commands](#)]

13. tar Command

The [tar command](#) is a Tape Archive that is useful in the creation of an archive, in several file format and their extraction.

```
tar -zxvf abc.tar.gz (Remember 'z' for .tar.gz)
```

```
tar -jxvf abc.tar.bz2 (Remember 'j' for .tar.bz2)
```

```
tar -cvf archive.tar.gz(.bz2) /path/to/folder/abc
```

Note: A 'tar.gz' means gzipped. 'tar.bz2' is compressed with bzip which uses a better but slower compression method.

14. cal Command

The "cal" (Calendar), is used to display the calendar of the present month or any other month of any year that is advancing or passed.

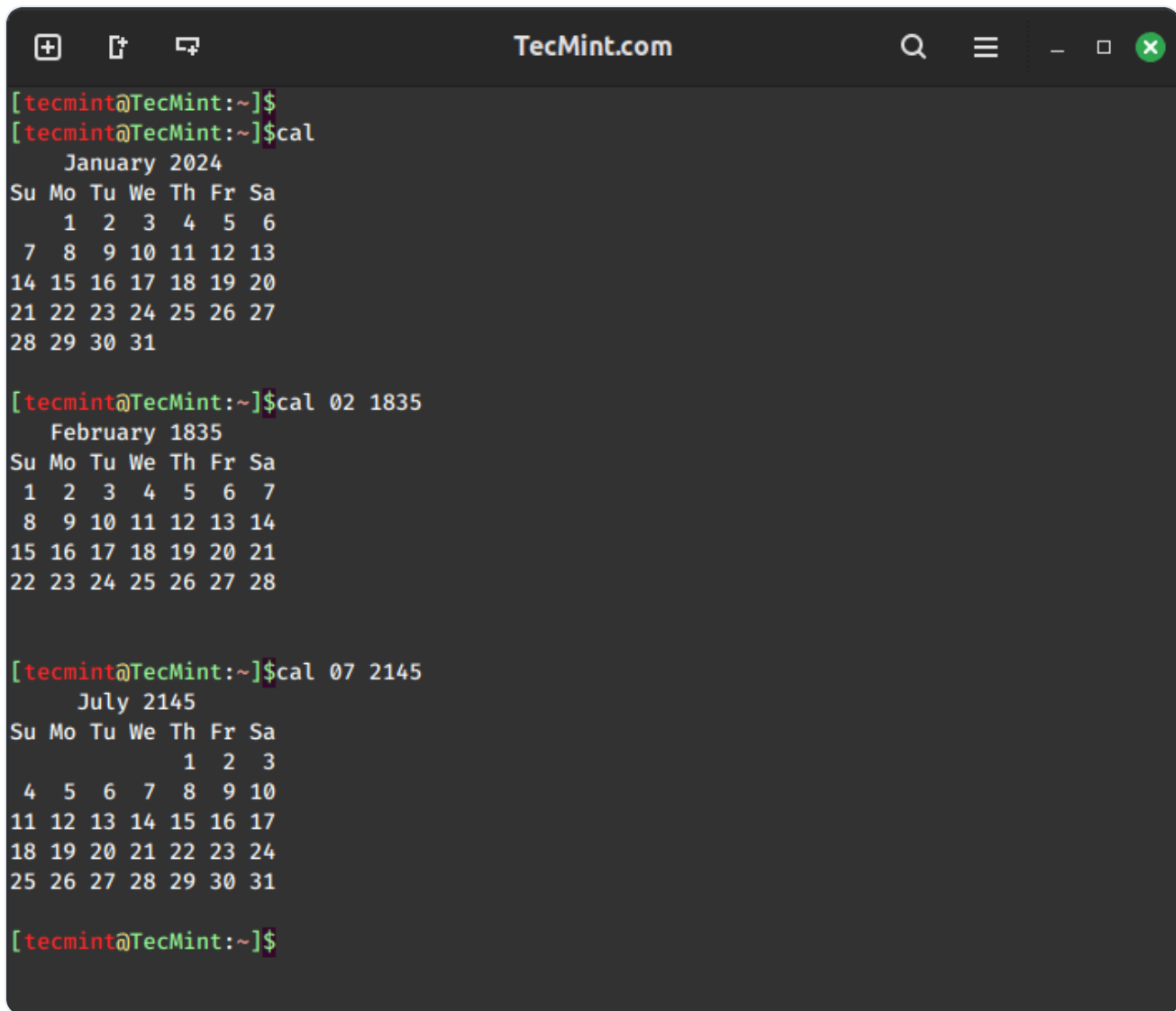
```
cal
```

Show the calendar of the year 1835 for February, which already has passed.

```
cal 02 1835
```

Shows the calendar of the year 2145 for July, which will be advancing

```
cal 07 2145
```



```
[tecmint@TecMint:~]$  
[tecmint@TecMint:~]$cal  
    January 2024  
Su Mo Tu We Th Fr Sa  
    1  2  3  4  5  6  
  7  8  9 10 11 12 13  
14 15 16 17 18 19 20  
21 22 23 24 25 26 27  
28 29 30 31  
  
[tecmint@TecMint:~]$cal 02 1835  
    February 1835  
Su Mo Tu We Th Fr Sa  
  1  2  3  4  5  6  7  
  8  9 10 11 12 13 14  
15 16 17 18 19 20 21  
22 23 24 25 26 27 28  
  
[tecmint@TecMint:~]$cal 07 2145  
    July 2145  
Su Mo Tu We Th Fr Sa  
          1  2  3  
  4  5  6  7  8  9 10  
11 12 13 14 15 16 17  
18 19 20 21 22 23 24  
25 26 27 28 29 30 31  
  
[tecmint@TecMint:~]$
```

Show Calendar in Linux

Note: You don't need to turn the calendar back by 50 years, nor do you need to perform complex mathematical calculations to determine the day you were born or the day your upcoming birthday will fall on.

15. date Command

The [date command](#) is used to display the current date and time. It can also be used to set the system date and time.

To display the current date and time.

```
date
```

To display the current date in the format "YYYY-MM-DD".

```
date +%Y-%m-%d
```

To set system date and time.

```
sudo date MMDDhhmm[[CC]YY][.ss]
```

The above command allows you to set the system date and time. Replace MM, DD, hh, mm, CC, YY, and ss with the desired values for month, day, hour, minute, century, year, and second, respectively.

Note: The `date` command is highly useful in scripting, especially for time and date-based operations. Moreover, changing the date and time using the terminal can make you feel like a true GEEK! (Of course, you need to have root permissions to perform this operation, as it involves a system-wide change).

16. cat Command

The [cat command](#) stands for (Concatenation), which means (join) two or more plain files and/or print contents of a file on standard output.

To display the contents of a file.

```
cat filename
```

To concatenate multiple files means the following command concatenates the contents of `file1` and `file2` and displays the result in the terminal.

```
cat file1 file2
```

The content of `a.txt`, `b.txt`, `c.txt`, and `d.txt` will be combined and appended to the end of the `abcd.txt` file.

```
cat a.txt b.txt c.txt d.txt >> abcd.txt
```

cat abcd.txt

Note: The “>>” and “>” are called append symbols. They are used to append the output to a file and not to standard output.

The “>” symbol will delete a file that already existed and create a new file hence for security reasons it is advised to use “>>” that will write the output without overwriting or deleting the file.

Before proceeding further, I must let you know about wildcards (you would be aware of wildcard entries, in most of Television shows) Wildcards are a shell feature that makes the command line much more powerful than any GUI file manager.

You see, if you want to select a big group of files in a [graphical file manager](#), you usually have to select them with your mouse. This may seem simple, but in some cases, it can be very frustrating.

For example, suppose you have a directory with a huge amount of all kinds of files and subdirectories, and you decide to move all the HTML files, that have the word “Linux” somewhere in the middle of their names, from that big directory into another directory.

What’s a simple way to do this? If the directory contains a huge amount of differently named HTML files, your task is everything but simple!

In the Linux command line that task is just as simple to perform as moving only one HTML file, and it’s so easy because of the shell wildcards. These are special characters that allow you to select file names that match certain patterns of characters.

This helps you to select even a big group of files by typing just a few characters, and in most cases, it’s easier than selecting the files with a mouse.

Here’s a list of the most commonly used wildcards :

Wildcard	Matches
*	zero or more characters
?	exactly one character
[abcde]	exactly one character listed
[a-e]	exactly one character in the given range
[!abcde]	any character that is not listed
[!a-e]	any character that is not in the given range
{debian,linux}	exactly one entire word in the options given

The `!` is called not a symbol, and the reverse of the string attached with `'!'` is true.

17. cp Command

The [cp command](#) stands for (copy), it copies a file from one location to another location.

```
cp /home/user/Downloads abc.tar.gz /home/user/Desktop
```

Note: The `cp` is one of the most commonly used commands in shell scripting and it can be used with wildcard characters (describe in the above block), for customized and desired file copying.

18. mv Command

The [mv command](#) moves a file from one location to another location.

```
mv /home/user/Downloads abc.tar.gz /home/user/Desktop
```

Note: The `mv` command can be used with wildcard characters. `mv` should be used with caution, as moving of system/unauthorized files may lead to security as well as a breakdown of the system.

19. pwd Command

The [pwd command](#) (print working directory), prints the current working directory with the full pathname from the terminal.


```
pwd
```

```
/home/user/Desktop
```

Note: The `pwd` command won't be frequently used in scripting but it is an absolute lifesaver for a newbie who gets lost in the terminal in their early connection with Linux. (Linux is most commonly referred to as `nux` or `nix`).

20. `cd` Command

Finally, the frequently used [cd command](#) stands for (change directory), which changes the working directory to execute, copy, move write, read, etc. from the terminal itself.

```
cd /home/user/Desktop
```

```
pwd
```

```
/home/user/Desktop
```

Note: The `cd` command comes to the rescue when switching between directories from the terminal. The `"cd ~"` will change the working directory to the user's home directory, which is very useful if a user finds themselves lost in the terminal. The `"cd .."` will change the working directory to the parent directory of the current working directory.

Conclusion

These commands will undoubtedly enhance your comfort with Linux. However, this is not the conclusion. Soon, I will introduce additional commands that will prove valuable for ['Middle-Level Users'](#). You will observe a promotion in your user-level status, progressing from a newbie to a middle-level user.

In the next article, I will be coming up with commands like ['Kill'](#), ['ps'](#), and ['grep'](#).

Hey TecMint readers,

Exciting news! Every month, our top blog commenters will have the chance to win fantastic rewards, like free Linux eBooks such as RHCE, RHCSA, LFCS, Learn Linux, and Awk, each worth \$20!

Learn [more about the contest](#) and stand a chance to win by [sharing your thoughts below!](#)



PREVIOUS ARTICLE:

Linux is an Art – Driving Force Behind Linux

NEXT ARTICLE:

Understand Linux Shell and Basic Shell Scripting Tips – Part I



Ravi Saive

I am an experienced GNU/Linux expert and a full-stack software developer with over a decade in the field of Linux and Open Source technologies

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

Join the [TecMint Weekly Newsletter](#) (More Than 156,129 Linux Enthusiasts Have Subscribed)

Was this article helpful? Please [add a comment](#) or [buy me a coffee](#) to show your appreciation.

Related Posts

```
tecmin@tecmin ~/testing $ find . -type f \( -name "*.txt" -o -  
name "*.sh" -o -name "*.c" \)  
./emails.txt  
./script-1.sh  
./header.c  
./examples.txt  
./script.sh  
./expenses.txt
```

Find Multiple Filenames (File Extensions) Using 'find' Command in Linux

How to Search Files by Name or Extension Using find Command



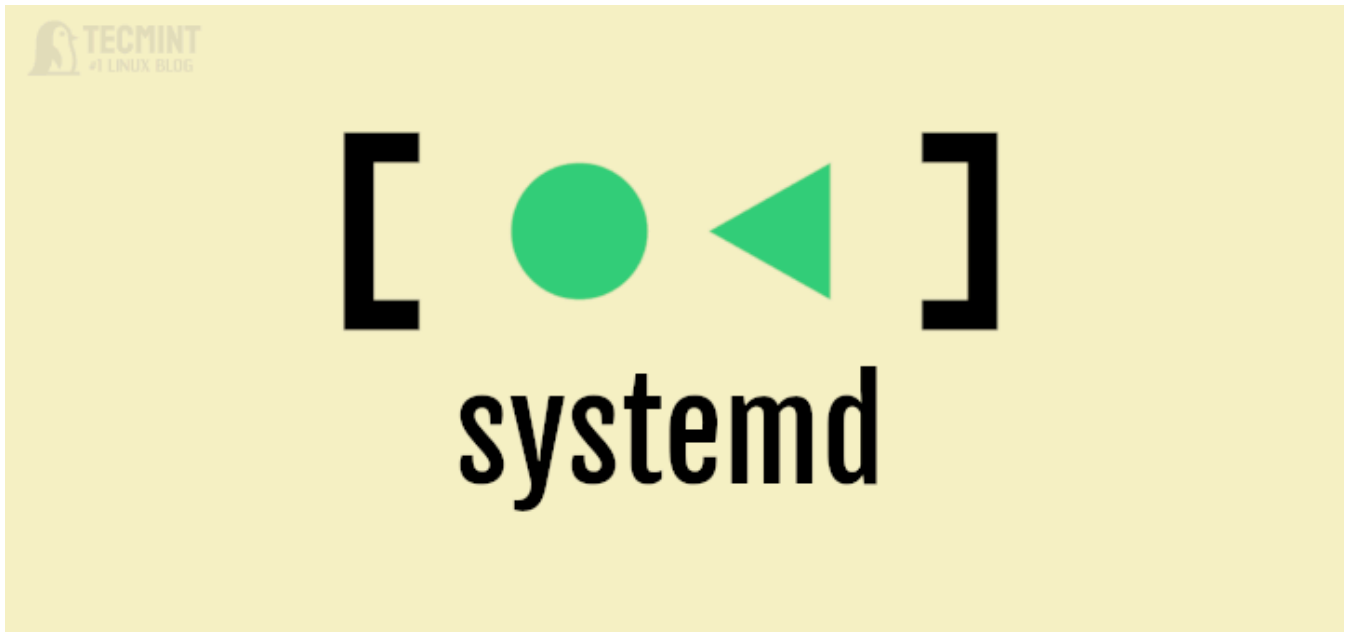
10 Lesser Known Linux Commands – Part 2



11 Lesser Known Useful Linux Commands



26 Security Hardening Tips for Modern Linux Servers



How to Remove Systemd Services on Linux

```
TecMint.com
ravi@TecMint:~/glibc-2.39/build$
ravi@TecMint:~/glibc-2.39/build$ ../configure --prefix=/usr/local/glibc-2.39
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for gcc... gcc
checking for suffix of object files... o
checking whether the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking for g++... g++
checking whether the compiler supports GNU C++... yes
checking whether g++ accepts -g... yes
checking for g++ option to enable C++11 features... none needed
checking whether g++ can link programs... yes
checking for sysdeps preconfigure fragments... aarch64 alpha arc arm csky hppa i386 loong
arch m68k microblaze checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
mips nios2 orlk powerpc riscv s390 sh checking for grep that handles long lines and -e...
(cached) /usr/bin/grep
```

How to Install and Run Multiple glibc Libraries in Linux

 **78 Comments**

[Leave a Reply](#)

Robyn Willison

June 17, 2020 at 10:25 am

dd is Disk Dump. I use it to back up a hard drive to a `.iso` file before I do any work on it ie recover files, upgrade os, etc.

Nicely paced article

[Reply](#)

madhav

October 22, 2018 at 9:17 pm

As i know; is a meta character which is used to separate two commands then what is the use of command after ; here in this dd command and what bs=512M used for?

```
root@tecmint:~# dd if=/home/user/Downloads/debian.iso of=/dev/sdb1  
bs=512M; sync
```

[Reply](#)

jebin

May 10, 2018 at 11:21 am

why it comes permission denied when we access some folder?

[Reply](#)

praveen kumar

May 15, 2017 at 10:45 am

nice one

[Reply](#)

Jamshock

March 27, 2017 at 7:02 pm

Include the command vi

[Reply](#)



Ararsa Hiko

November 19, 2016 at 12:33 pm

I really liked it, please add more such topics, thanks.

[Reply](#)



Rod

August 12, 2016 at 5:29 pm

You forgot to tell the readers the "root@tecmint:" refers to your computer and will be different for each owner.... Otherwise this is a useful article.

Although it is directed to power users, most Linux users will find Ubuntu or Mint, along with the many other GUI, will be just fine! Operating from the terminal is complex and fun at the same time just remember be careful with commands.

[Reply](#)



erm3nda

March 28, 2016 at 5:23 pm

I think this guide is so nice, well fitted. Just missing a command, which is really useful for new people coming, looking at logs :) to solve problems.

I'm talking about head and tail and grep. Those commands are necessary to inspect logs in a better/faster/easier way and can be "added" aside the "cat" section you made.

Also, an Alias section would be mega useful to don't force users to use ctrl+r to remember their commands.

I'm one of those people hating viruses along Windows, so I've switched to Debian and I know what I say.

Nice work.

[Reply](#)



VIKASH KUMAR GAUTAM

February 24, 2016 at 3:18 pm

Very good tutorial and very useful

[Reply](#)



Aniruddha

February 2, 2016 at 4:50 pm

This is very useful and very well explained.

Thanks !

[Reply](#)

Mick

November 29, 2015 at 11:17 pm

Very good introduction from Windows to Linux. It should be placed at first position of Section 2, before detailed explanations or examples given for each of the commands. Thank you from a windows defector!

[Reply](#)

jay nath jha

August 23, 2015 at 3:27 pm

I want to learn LINUX please teach me.

[Reply](#)

Author

**Ravi Saive**

August 24, 2015 at 12:01 pm

@Jay,

Here is the exclusive course module from TecMint, compiled for newbies only to learn and improve their Linux skills..

<https://www.tecmint.com/free-online-linux-learning-guide-for-beginners/>

[Reply](#)**Swapnil Nimje**

July 16, 2015 at 10:53 am

If i done Linux (RHCE).....is jobs in market that i can get easily....

[Reply](#)**Swapnil Nimje**

July 16, 2015 at 10:50 am

Dear Sir,

What is the future in Linux ?... I want to know....is there any growth?...is it good field in Linux (RHCE)?...please help me...

[Reply](#)**ravendhran**

June 19, 2015 at 2:04 pm

ia verry happy for the section

[Reply](#)**Avishek Kumar**

June 29, 2015 at 3:32 pm

Welcome ravendhran,
Keep connected for more such posts.

[Reply](#)**Vatsal**

April 1, 2015 at 6:18 am

Hi avishek, this website is very useful for me. Can you tell me what other stuff as a newbie i need to learn to give an interview for position of sys admin intern?

[Reply](#)**Avishek Kumar**

April 1, 2015 at 12:02 pm

Welcome Vatsal. Have good command over Scripting (Shell/Python/perl) and general service which includes but not limited to Apache, MySQL,... Also have sound knowledge of networking and network Protocols and what ever you can learn just keep doing...

[Reply](#)

Romanator

March 31, 2015 at 2:45 am

Amazing website. I am a newbie, however I have a quick question in regards to "cat" command and the example above:

```
oot@tecmint:~# cat a.txt b.txt c.txt d.txt abcd.txt
```

Are you trying to concatenate 5 files (a, b, c, d, abcd.txt) and print the content, or are you trying to concatenate a, b, c, d to abcd.txt? If the latter, should not this command be:

```
oot@tecmint:~# cat a.txt b.txt c.txt d.txt >>abcd.txt
```

Again – I have learnt a lot, and I am trying to follow every example. It took me a while to understand what was going on with the first command. Thank you!

[Reply](#)



Avishek Kumar

June 29, 2015 at 3:47 pm

Dear Romanator,

You are right. It is a mistake on my part.

The command should be

```
# cat a.txt b.txt c.txt d.txt >> abcd.txt
```

or

```
# cat a.txt b.txt c.txt d.txt > abcd.txt
```

Asking admin to amend it, please.

[Reply](#)



Dinesh Yadav

March 25, 2015 at 3:12 am

nice post with good explanation...

thank you

[Reply](#)



Avishek Kumar

March 25, 2015 at 1:57 pm

Welcome @ Dinesh

[Reply](#)

wahidullah

November 13, 2014 at 10:49 am

Thanks a loot

[Reply](#)**Avishek Kumar**

March 25, 2015 at 1:57 pm

Welcome @ wahidullah

[Reply](#)**Pritam Saha**

October 14, 2014 at 9:30 am

These are the very Basic commands in LINUX...Interested to know some more commands which will be useful too..

Thanking you...

[Reply](#)**venkatesh**

October 27, 2014 at 1:24 pm

how to add a user and how to delete a user plz put some example here

[Reply](#)

Author

**Ravi Saive**

October 27, 2014 at 5:09 pm

Here is the complete guide on how to add users and delete users in Linux.

<https://www.tecmint.com/add-users-in-linux/>

[Reply](#)

Amol Bhosale

September 18, 2014 at 9:44 pm

Dear Sir,

I am looking for lab session like

- 1) creating local repository by copying packages from DVD.
- 2) Installing & Configuring SAMBA Server step by step.
- 3) Creating Backup Server.

Thanks & Regards,

Amol Bhosale

[Reply](#)

Author



Ravi Saive

September 20, 2014 at 11:55 am

Dear Amol,

The things you looking for, we already covered, juse use our search form to search for articles.

[Reply](#)**vinayak**

November 5, 2014 at 2:36 pm

I also looking for installation and configuration on different servers like samba,dhcp,mailing servers etc.

[Reply](#)**Alex**

August 28, 2014 at 2:37 am

PS – This page has now been honoured by being bookmarked on the Tool-bar!!!

:)

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:25 pm

we are honoured @ Alex

[Reply](#)

Alex

August 28, 2014 at 2:35 am

Very nice work. As another ancient but new to Linux user: I Thank you!
Linux Lite suits my venerably old machine just fine – she hums along happily without the screaming and screeching with Windows...we are both happier :)

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:25 pm

:)

[Reply](#)**Narasinhareddy**

August 27, 2014 at 4:12 pm

Nice notes for beginner....thank you sir

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:26 pm

Welcome @ Narasinhareddy

[Reply](#)**Late P.E.**

June 30, 2014 at 3:31 pm

commands explained easily to understand...thanks

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:26 pm

Welcome Late P.E.

[Reply](#)**Suresh**

June 5, 2014 at 12:37 pm

The commands are explained nicely with good examples as well. Good job.

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:27 pm

Oh! Thankyou Suresh.

[Reply](#)

Crazy Bani

May 9, 2014 at 1:28 am

It helps me out....thnx

[Reply](#)



Avishek Kumar

September 2, 2014 at 3:29 pm

Great to know it @ crazy Bani

[Reply](#)

crustyasp

April 19, 2014 at 6:04 am

As a noob, moving over to Linux, on an old PIII, Dell laptop, I appreciate your effort to introduce me to the command line. I also appreciate the links to find more command line usage.

My motivation to move to Linux was the age of my computer, the lack of support for the computer by Microsoft, (why throw away that which is usefull, and be forced to buy something new just to have to repeat it again, because Microsoft

forces you to) and also to disprove the saying that an old dog (68) can't be taught new tricks. The distro that I found that seems best suited to my needs and computer was Slitaz. TThank you Linux world.

[Reply](#)

N0d3_z3r0

May 19, 2014 at 8:38 pm

Welcome. We are glad to have you apart of our world!!

[Reply](#)



Avishek Kumar

September 2, 2014 at 3:29 pm

Difficult to understand but thanks for your quote.

[Reply](#)



Avishek Kumar

September 2, 2014 at 3:30 pm

Welcome SIR,
we really feels inspired.

[Reply](#)

rps

February 2, 2014 at 8:49 pm

'lsblk' does not work on Redhat?

[Reply](#)**Amol**

March 30, 2014 at 1:50 am

It works on RedHat. Btw... keep it up ABhishek Kumar.... !!

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:31 pm

@ Amol

it works on all Distros.

[Reply](#)**minhaj**

October 29, 2013 at 1:46 pm

Nice one

[Reply](#)**Avishek Kumar**

September 2, 2014 at 3:32 pm

thanks

[Reply](#)**John Susa**

August 9, 2013 at 6:11 am

Apart of having a TERRIBLE English, your post is good. You're a experienced computer user, you SHOULD have a good command of English please!!!!

[Reply](#)**Gabbleduck**

February 18, 2014 at 4:19 am

John Susa, you might want to make sure that your command of english is 100% accurate before you start saying things like that.

"Apart of having...." ????

[Reply](#)

N0d3_z3r0

May 19, 2014 at 8:36 pm

Lol. Well played sir.

[Reply](#)**Lestrade**

June 3, 2014 at 8:34 pm

The point is, it's not English anymore. It's Earthish.

[Reply](#)**Jason Bourne**

August 9, 2013 at 2:43 am

What the hell is this????

The 1st ravisaive in the above example means that file is owned by user ravisaive.

The 2nd ravisaive in the above example means file belongs to user group ravisaive.

What is a "ravissaive"????

[Reply](#)

Author

**Ravi Saive**

August 9, 2013 at 12:02 pm

It's a system USER.

[Reply](#)

MightyMoo

May 20, 2013 at 3:50 pm

This is a nice guide for someone who does more then check email and browse websites on a machine. The basic email/web guys would like more graphical style stuff though I think. It'd be neat if there was a side by side comparison of the command line and graphical tool (if available).

[Reply](#)

mankou

May 20, 2013 at 1:22 pm

Nix?

Nux?

[Reply](#)



Avishek

May 20, 2013 at 10:51 am

@ ken

sudo apt-get remove ubuntu is ok.

but if you mean sudo apt-get remove linux or yum remove linux
then dear i have to ask you, what the hell are you doing here ?

[Reply](#)



Avishek

May 20, 2013 at 10:48 am

@ Cat and Scott no Linux user will be afraid of terminal. it is the curocity, the adventure, the fun that drives them to the world of Linux, they know that terminal (shell) is the soul of Linux and is much powerful that powershell or any other shell implemented in other OSes. if this is not true, most of us would have still be sticks to walls, gates and windows.

[Reply](#)

Gabbleduck

February 18, 2014 at 4:17 am

I agree, I am very new but it was the excitement of being able to control things more, to be able to play around with a command line and have the excitement of the risk of potentially being able to break things while playing, that is what drew me to Linux. You are dead right, someone who is afraid of the command line would not be trying to play with it, and would continue to point-and-click :-)

[Reply](#)

**Avishek**

May 20, 2013 at 10:44 am

@ Rambo Tribble sure df is one of the important command but for sure a newbie in linux don't have to bother much about his disk usages, and when you have to provide only 20 commands, whom will you select and whom to discard is a matter of debate, moreover df will surely be available in the next part of this tutorial.

[Reply](#)**Curtis**

May 20, 2013 at 10:42 am

I think that part about apt being superior to yum is pretty infamatory. It would be one thing if you actually had some kind of included reference to back it up. Or at the very least you could have included a brief explanation.

Honestly, I think they both have their strong points over one another. You can't say that one is unequivocally better than the other, as they are both very mature.

I am a pacman user though. And just ask Allan, pacman f-ing rules! (At least is is certainly far faster than yum or apt)

[Reply](#)**Avishek**

May 20, 2013 at 12:53 pm

yum :

yum Comes with Fedora and fedora like OS by default and is Widely supported

by many RPM repositories which is Fairly easy to roll your own repository. However, No “pinning” or other mechanisms for backing out updates, Somewhat confusing to create a hierarchy of “trust,” in terms of which repositories to use for which packages and No graphical update/management tool are the drawbacks of yum.

apt :

apt Somewhat has wider support across distributions (since its a Debian tool), which include Synaptic, a graphical GUI for doing updates and “Pinning” support, and uninstall support.

However, apt is Not a base part of the FC package set. Perhaps a slightly higher learning curve to use effectively.

Harder to use as a verb in jargon filled conversations. “I apted GIMP2 last night” as opposed to “Yum up that new GIMP2 package”

(apt is used by debian and debian like distro viz., ubuntu, kubuntu, fuduntu, kali,, etc and yum is used by fedora and fedora like os viz., redhat, centos, etc. The number of distros using apt is several times more than distros using yum as package manager.)

None of the two is perfect, but surely apt has an edge over yum. You better compare yourself.

[Reply](#)

ken

May 19, 2013 at 11:28 pm

sudo apt-get remove ubuntu

[Reply](#)

dolan

October 7, 2014 at 3:56 pm

that is the evilest thing I can imagine

[Reply](#)



Avishek Kumar

October 9, 2014 at 10:30 pm

lol :)

[Reply](#)

Cat

May 19, 2013 at 7:44 pm

Sure way to scare linux newbies away is starting to tell them about command line. First they must master normal use in windowed environment. When they have those skills and are interested in deeper dive in linux, then it's time to start learning command line.

[Reply](#)



Scott

May 20, 2013 at 1:36 am

I agree. This leaves the impression that to use Linux one must learn the command line. For desktop use the command line is completely unnecessary.

[Reply](#)**Lestrad**

June 3, 2014 at 8:26 pm

I disagree. When I got my first computer, Windows 3.0 had just come out. But instead of installing it, I sat down and learned to work with DOS, and to this day I have never regretted it. It STILL comes in handy and has saved my yass a few times. After all, the operations you perform in a graphical environment are the same ones you perform from the command line. Knowing how to do them in their original, historical form is not going to hurt anybody and will do people a lot of good. It's a little like learning Latin – once you study it a bit you start to understand English with a lot more depth.

[Reply](#)**psyuk**

October 3, 2014 at 9:27 pm

I disagree.

This was exactly what I needed in coming to run Linux in an embedded environment with only a shell for access.

[Reply](#)**Rambo Tribble**

May 19, 2013 at 7:41 pm

I would suggest adding "df" to your list. Keeping track of disk resources is always a good idea.

[Reply](#)



Bob

May 18, 2013 at 12:25 am

#20 has some duplicate code from #19. Otherwise, it's a great tutorial :)

[Reply](#)



Avishek

May 18, 2013 at 3:07 pm

#20 has some duplicate code from #19 ???
of which kind and where?

[Reply](#)

kiran

November 14, 2014 at 12:00 pm

Hi Sir,

I am into windows side,working as system admin,need to switch over to linux..so
plz suggest what all the basic things need to learn in linux..tell me how to
prepare myself..waiting for ur reply...

[Reply](#)

Author

**Ravi Saive**

November 14, 2014 at 9:09 pm

@kiran,

We always suggest newbies to start with basic Linux commands, we've a huge collection of basic Linux commands with examples to learn at

<https://www.tecmint.com/category/linux-commands/>.

[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.

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

Post Comment

Do You Enjoy My Blog?

Support from readers like YOU keeps this blog running. Buying me a cup of coffee is a simple and affordable way to show your appreciation and help keep the posts coming!

Buy Me a Coffee

Linux Commands and Tools

[How to Fix “-bash: sudo: command not found” in Linux](#)

[10 Lesser Known Commands for Linux – Part 3](#)

[5 ‘chattr’ Commands to Make Important Files IMMUTABLE \(Unchangeable\) in Linux](#)

[How To Assign Output of a Linux Command to a Variable](#)

[Sysadmins & DevOps: 20 Must-Know Advanced Linux Commands](#)

[45 Zypper Commands to Manage ‘Suse’ Linux Package Management](#)

Linux Server Monitoring Tools

[Monitor Server Logs in Real-Time with “Log.io” Tool on RHEL/CentOS 7/6](#)

[How to Create a Centralized Log Server with Rsyslog in CentOS/RHEL 7](#)

[vtop – A Linux Process and Memory Activity Monitoring Tool](#)

[Conky – A System Monitor Tool for Linux Desktop](#)

[Psensor – Monitor Linux Hardware Temperature \[Motherboard and CPU\]](#)

[iPerf3 – Test Network Speed/Throughput in Linux](#)

Learn Linux Tricks & Tips

[Assign Read/Write Access to a User on Specific Directory in Linux](#)

[Show a Custom Message to Users Before Linux Server Shutdown](#)

[2 Ways to Re-run Last Executed Commands in Linux](#)

[How To Assign Output of a Linux Command to a Variable](#)

[How to Recover a Deleted File in Linux](#)

[How to Check Remote Ports are Reachable Using ‘nc’ Command](#)

Best Linux Tools

[5 Best Open-Source School Management Software for Linux](#)

[15 Best Kali Linux Web Penetration Testing Tools](#)

[8 Best MySQL/MariaDB GUI Tools for Linux in 2024](#)

[3 Tools to Monitor and Debug Disk I/O Performance in Linux](#)

[11 Best IP Address Management Tools for Linux Network](#)

[6 Best Linux Boot Loaders](#)

Tecmint: Linux Howtos, Tutorials & Guides © 2024. All Rights Reserved.

The material in this site cannot be republished either online or offline, without our permission.

Hosting Sponsored by : [Linode Cloud Hosting](#)