

FREE EBOOKS

[LINUX HOWTO](#)[LINUX DISTROS](#)[LINUX COMMANDS](#)[LINUX TOOLS](#)[AUTOMATION](#)16, 2011 | By [Bobbie Zachariah](#) [More](#)[f](#) [t](#) [G+](#) [in](#)

- SHELL SCRIPTS

- DEVOPS

- WEB SERVERS

- UBUNTU HOWTO



Linux based Operating Systems are very powerful but their true power lies in the command line. There is a lot that you can do with the help of commands but can't otherwise (using GUI). In this article, we will find out about basic Linux commands that are used most frequently. Now, let's start from the beginning.



Syntax

The commands in Linux have the following syntax:

\$command options arguments

The command is followed by options (optional of course) and a list of arguments. The options can modify the behavior of a command. The arguments may be files or directories or some other data on which the command acts. Every command might not need arguments. Some commands work with or without them (e.g. 'ls' command). The options can be provided in two ways: full word options with -- (e.g. --help), or single letter options with - (e.g. -a -b -c or multiple options, -abc).

Linux Basic Commands

Let's start with some simple commands.

The pwd command:

'pwd' command prints the absolute path to current working directory.

```
$ pwd
/home/raghu
```

The cal command:

'cal' command displays the calendar of the current month.

```
$ cal
July 2012
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
```

'cal' will display calendar for the specified month and year.

```
$ cal 08 1991
August 1991
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
```

The echo command:

This command will echo whatever you provide it.

```
$ echo "linuxide.com"
linuxide.com
```

The 'echo' command is used to display the values of a variable. One such variable is 'HOME'. To check the value of a variable precede the variable with a \$ sign.

```
$ echo $HOME
/home/raghu
```

The date command:

Displays current time and date.

```
$ date
Fri Jul 6 01:07:09 IST 2012
```

If you are interested only in time, you can use 'date +%T' (in hh:mm:ss):

```
$ date +%T
01:13:14
```

The tty command:

Displays current terminal.

```
$ tty
/dev/pts/0
```

The whoami command:

This command reveals the user who is currently logged in.

```
$ whoami
raghu
```

The id command:

This command prints user and groups (UID and GID) of the current user.

```
$ id
uid=1000(raghu) gid=1000(raghu)
groups=1000(raghu),4(adm),20(dialout),24(cdrom),46(plugdev),112(lpadmin),120(admin),122(sambashare)
```

By default, information about the current user is displayed. If another username is provided as an argument, information about that user will be printed:

```
$ id root
uid=0(root) gid=0(root) groups=0(root)
```

The clear command:

This command clears the screen.

Getting help command

For all of its advantages, a big disadvantage of the command line is that there are a lot of commands with too many options and usage. But nobody can remember all the commands. There are some smart ways of using command line in Linux. Several such resources are discussed here:

› option

almost every command, '--help' option shows usage summary for that command.

```
$ date --help
Usage: date [OPTION]... [+FORMAT]
or: date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]
Display the current time in the given FORMAT, or set the system date.
```

The whatis command

This command gives a one line description about the command. It can be used as a quick reference for any command.

```
$ whatis date
date (1) - print or set the system date and time

$ whatis whatis
whatis (1) - display manual page descriptions
```

Manual Pages

'--help' option and 'whatis' command do not provide thorough information about the command. For more detailed information, Linux provides man pages and info pages. To see a command's manual page, man command is used.

```
$ man date
```

The man pages are properly documented pages. They have following sections:

NAME: The name and one line description of the command.

SYNOPSIS: The command syntax.

DESCRIPTION: Detailed description about what a command does.

OPTIONS: A list and description of all of the command's options.

EXAMPLES: Examples of command usage.

FILES: Any file associated with the command.

AUTHOR: Author of the man page

REPORTING BUGS: Link of website or mail-id where you can report any bug.

SEE ALSO: Any commands related to the command, for further reference.

-k option, a search through man pages can be performed. This searches for a pattern in the name and description of a man page.

```
$ man -k gzip
gzip (1) - compress or expand files
lz (1) - gunzips and shows a listing of a gzip'd tar'd archive
tgz (1) - makes a gzip'd tar archive
uz (1) - gunzips and extracts a gzip'd tar'd archive
zforce (1) - force a '.gz' extension on all gzip files
```

Info pages

Info documents are sometimes more elaborate than the man pages. But for some commands, info pages are just the same as man pages. These are like web pages. Internal links are present within the info pages. These links are called nodes. Info pages can be navigated from one page to another through these nodes.

```
$ info date
```

Linux Filesystem commands

Changing Directories Command

```
$ cd [path-to-directory]
```

Change the current working directory to the directory provided as argument. If no argument is given to 'cd', it changes the directory to the user's home directory. The directory path can be an absolute path or relative to current directory. The absolute path always starts with /. The current directory can be checked with 'pwd' command (remember?):

```
$ pwd
/home/raghu
$ cd /usr/share/
```

```
$ pwd
/usr/share
$ cd doc
$ pwd
/usr/share/doc
```

In the first 'cd' command, absolute path (/usr/share) is used, and with second command, relative path (doc) is used.

Listing File And Directories Command

```
$ ls [files-or-directories]
```

files and/or directories. If no argument is given, the contents of current directory are shown.

```
$ ls
example file1.txt file2.txt file3.txt
```

directory is given as an argument, files and directories in that directory are shown.

```
$ ls /usr
bin games include lib lib64 local sbin share src
```

'ls -l' displays a long listing of the files.

```
$ ls -l
total 4
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 12:52 example
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file1.txt
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file2.txt
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file3.txt
```

In this long listing, the first character is 'd' or '-'. It distinguishes between file types. The entries with a '-' (dash) are regular files, and ones with 'd' are directories. The next 9 characters are permissions ('rwxr-xr-x' in first listing). The number following the permissions is the link count. Link count follows user and group owner. In the above example, the file owner is 'raghu' and group owner is 'raghu' as well. Next is the size of the file. And then time stamp before the name of file (or directory).

By default, hidden files or directories are not shown, to see hidden files as well, -a option is used. Hidden files in Linux start with a period sign (.). Any file that starts with a period is hidden. So, to hide a file, you just need to rename it (and put a period before it).

```
$ ls -la odesk
total 16
drwxr-xr-x 4 raghu raghu 4096 2012-07-06 13:46 .
```

```
drwxr-xr-x 11 raghu raghu 4096 2012-07-06 13:15 ..
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 12:52 example
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file1.txt
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file2.txt
-rw-r--r-- 1 raghu raghu 0 2012-07-06 12:52 file3.txt
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 13:46 .hiddendir
-rw-r--r-- 1 raghu raghu 0 2012-07-06 13:46 .hiddenfile1.txt
-rw-r--r-- 1 raghu raghu 0 2012-07-06 13:46 .hiddenfile2.txt
```

If you want to see the properties of a directory instead of the files contained in it, use -d (with -l) option:

```
$ ls -ld odesk/
drwxr-xr-x 4 raghu raghu 4096 2012-07-06 13:46 odesk/
```

ing files and directories Command

r command

reate a directory, the 'mkdir' command is used.

```
$ mkdir example
$ ls -l
total 4
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 example
```

The touch command

For creating an empty file, use the touch command.

```
$ touch file1 file2 file3
$ ls -l
total 4
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 example
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file2
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
```

If a file already exists, touch will update its time stamp. There are a lot of other methods to create a new file, e.g. using a text editor like vi or gedit, or using redirection. Here is an example of creating a file using redirection:

```
$ ls -l /usr > usrlisting
$ ls -l
total 8
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 example
```

```
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file2
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
-rw-r--r-- 1 raghu raghu 491 2012-07-06 14:23 usrlisting
```

A file named usrlisting is created in this example.

Copy, move and remove commands

Copy command

```
$cp source destination
```

Files and directories. If the source is a file, and the destination (file) name does not exist, then source is copied with new name i.e. with the name provided as the destination.

```
$ cp usrlisting listing_copy.txt
$ ls -l
total 12
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 example
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file2
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:02 listing_copy.txt
-rw-r--r-- 1 raghu raghu 491 2012-07-06 14:23 usrlisting
```

If the destination is a directory, then the file is copied with its original name in that directory.

```
$ cp listing_copy.txt example/
$ ls -l example/
total 4
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:07 listing_copy.txt
```

Multiple files can also be copied, but in that case, the last argument will be expected to be a directory where all the files are to be copied. And the rest of the arguments will be treated as file names.

```
$ cp file1 file2 example/
$ ls -l example/
total 4
-rw-r--r-- 1 raghu raghu 0 2012-07-06 16:10 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 16:10 file2
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:07 listing_copy.txt
```


If a directory is to be copied, then it must be copied recursively with the files contained in it. To copy a directory recursively, use `-r` option with `'cp'` command:

```
$ cp -r example /tmp/expertslogin/
$ ls -l /tmp/expertslogin
total 4
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 16:12 example
```

The move command

```
$ mv source destination
```

» files or directories. The `'mv'` command works like `'cp'` command, except that the original file is removed. the `mv` command can be used to rename the files (or directories).

```
$ mv listing_copy.txt usrcopy
$ ls -l
total 12
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 16:10 example
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file2
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:02 usrcopy
-rw-r--r-- 1 raghu raghu 491 2012-07-06 14:23 usrlisting
```

Here, `'listing_copy.txt'` is moved with the name `'usrcopy'` in the same directory (or you can say that it has been renamed).

To remove or Delete

```
$ rmdir
```

`'rmdir'` command removes any empty directories, but cannot delete a directory if a file is present in it. To use `'rmdir'` command, you must first remove all the files present in the directory you wish to remove (and possibly directories if any).

To remove files and directories

```
$ rm files|directories
```

A directory must be removed recursively with `-r` option.

```
$ rm file2
$ rm -r example/
$ ls -l
```

```
total 8
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:02 usrcopy
-rw-r--r-- 1 raghu raghu 491 2012-07-06 14:23 usrlisting
```

Here, the file named 'file2' is removed first, and then the directory 'example' is removed recursively. This can be seen in the output of 'ls -l' command where these two are no longer present.

Other file commands

--- command

file command determines the file type of a given file. For example:

```
$ file /etc/passwd
/etc/passwd: ASCII text
```

can provide one or more than one file as an argument to the file command.

```
$ file td.c td.out ARP.java Screenshot.png StringTokenizing.class
idl.rar List.pdf
td.c: ASCII C program text, with CRLF line terminators
td.out: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked (uses shared
libs), for GNU/Linux 2.6.15, not stripped
ARP.java: ASCII Java program text, with CRLF line terminators
Screenshot.png: PNG image data, 1366 x 768, 8-bit/color RGB, non-interlaced
StringTokenizing.class: compiled Java class data, version 50.0 (Java 1.6)
idl.rar: RAR archive data, v1d, os: Win32
List.pdf: PDF document, version 1.4
```

stat command

To check the status of a file. This provides more detailed information about a file than 'ls -l' output.

```
$ stat usrcopy
File: `usrcopy'
Size: 491 Blocks: 8 IO Block: 4096 regular file
Device: 808h/2056d Inode: 149452 Links: 1
Access: (0644/-rw-r--r--) Uid: ( 1000/ raghu) Gid: ( 1000/ raghu)
Access: 2012-07-06 16:07:06.413522009 +0530
Modify: 2012-07-06 16:02:30.204152386 +0530
Change: 2012-07-06 16:17:18.992559654 +0530
```

cat command

The 'cat' command is actually a concatenator, but can be used to view the contents of a file.

```
$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
```

rs

cat command lists file as a whole. But if the file is big enough to fit into one screen, then we will be able to only the last page of the file. The commands 'less' and 'more' display files one page at a time. So they are called pagers. You can navigate through a file using arrow keys. To quit from a pager, hit 'q'.

command

Displays the first few lines of a file. By default, the 'head' command displays the first 10 lines of a file. But with -n option, the number of lines to be viewed can be specified.

```
$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
```

tail command

Similar to 'head'; the 'tail' command shows the last 10 lines by default, and -n option is available as well.

```
$ tail -n 4 /etc/passwd
raghu:x:1000:1000:Raghu Sharma,,,:/home/raghu:/bin/bash
sshd:x:113:65534:./var/run/sshd:/usr/sbin/nologin
dictd:x:114:123:Dictd Server,,,:/var/lib/dictd:/bin/false
mysql:x:115:124:MySQL Server,,,:/nonexistent:/bin/false
```

wc command

Word count

This command counts lines, words and letters of the input given to it.

```
$ wc /etc/passwd
35 57 1698 /etc/passwd
```

The /etc/passwd file has 35 lines, 57 words, and 1698 letters present in it.

grep command

grep' command searches for a pattern in a file (or standard input). It supports regular expressions. It prints a line if it matches the pattern in that line. So, if we wish to find the lines containing the word 'nologin', use 'grep' as follows:

```
$ grep nologin /etc/passwd
sshd:x:113:65534::/var/run/sshd:/usr/sbin/nologin
```

ln command

The ln command is used to create links. Links are a kind of shortcuts to other files. The general form of the command is:

```
$ ln TARGET LINK_NAME
```

There are two types of links, soft links and hard links. By default, hard links are created. If you want to create a soft link, use -s option. In this example, both types of links are created for the file usrlisting.

```
$ ln usrlisting hard_link

$ ln -s usrlisting soft_link

$ ls -l
total 12
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file3
-rw-r--r-- 2 raghu raghu 491 2012-07-06 14:23 hard_link
lrwxrwxrwx 1 raghu raghu 10 2012-07-09 14:00 soft_link -> usrlisting
-rw-r--r-- 1 raghu raghu 491 2012-07-06 16:02 usrcopy
-rw-r--r-- 2 raghu raghu 491 2012-07-06 14:23 usrlisting
```

Text Editors

Pico & Nano

'Pico' is a text editor in Linux. 'Nano' editor is inspired from 'pico'. They work almost the same. If the argument given as filename exists, then that file will be opened for editing in pico/nano. Otherwise, a new file with that name will be created. Let's create a new file named hello.txt:

```
$ pico hello.txt
```

```
GNU nano 2.2.6 File: hello.txt Modified
```

This file is edited with pico editor.

```
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
```

```
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

ng made all the changes to the file, press 'ctrl+o' to write the changes to the file and 'ctrl+x' to exit from the r. There are a lot of functions available with this editor. The help menu can be accessed with 'ctrl+g' trokes.

itor

VI stands for Visual editor; another text editor in Linux. This is a standard editor in many Linux/Unix onments. This is the default editor that comes with many Linux distributions. It might be possible that it is the only text editor available with your distro.

You can open a file with vi for editing using the following:

```
$ vi hello.txt
```

The vi editor has 3 modes in which it performs its functions. The default is COMMAND mode, in which tasks like copy, paste, undo etc can be performed. You can change a mode from command mode only (and come back to it). The second mode is the INSERT mode, in which whatever key you type is treated as a character and will be loaded into the file buffer. To enter this mode, press 'i' when in command mode.

Final mode is EX mode or last line mode. The changes made in the buffer can be saved or discarded in this mode.

```
Hello world.
```

```
This file is edited using vi editor.
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

~
~
~
~
~
~
~

"hello.txt" 2 lines, 50 characters

Some additional useful commands

command

alias' is another name for a command. If no argument is given, it shows current aliases. Aliases can be for short names of commands. For example, you might use the clear command frequently. You can create alias for it:

```
$ alias c="clear"
```

Next time you enter 'c ' on command line, your screen will get clear. Current aliases can be checked with 'alias' command:

```
$ alias
alias alert='notify-send --urgency=low -i "${[ $? = 0 ]}&& echo terminal || echo error" "$(history|tail -n1|sed -e \"s/^s*[0-9]\\+s*//;s/[;&|]s*alert$/\"")'
alias c='clear'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -aF'
alias ls='ls --color=auto'
```

w command

w command is used to check which users are logged in to the system, and what command they are executing at that particular time:

```
$ w
10:06:56 up 57 min, 3 users, load average: 0.04, 0.06, 0.09
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
root tty1 10:06 28.00s 1.02s 0.67s pager -s
```

```
raghu tty7 :0 09:19 57:33 1:22 0.20s gnome-session --session=classic-gnome
raghu pts/0 :0.0 09:34 0.00s 0.78s 0.00s w
```

It also shows the uptime, number of users logged in and load average of the system (in the first line of output above).

last command

Displays information about the users who logged in and out of the system. The output of the last command can be very large, so the following output has been filtered (through head) to display the top 10 lines only:

```
$ last | head
root tty1 Mon Jul 9 10:06 still logged in
root tty1 Mon Jul 9 10:06 - 10:06 (00:00)
raghu pts/1 :0.0 Mon Jul 9 10:05 - 10:06 (00:00)
raghu pts/0 :0.0 Mon Jul 9 09:34 still logged in
raghu tty7 :0 Mon Jul 9 09:19 still logged in
reboot system boot 2.6.38-13-generi Mon Jul 9 09:09 - 10:12 (01:02)
raghu tty7 :0 Sun Jul 8 23:36 - 00:30 (00:54)
reboot system boot 2.6.38-13-generi Sun Jul 8 23:36 - 00:30 (00:54)
raghu tty7 :0 Sun Jul 8 21:07 - down (01:06)
reboot system boot 2.6.38-13-generi Sun Jul 8 21:07 - 22:14 (01:07)
```

A similar command is 'lastb' that shows the last unsuccessful login attempts. But this command must be run as root otherwise you would get an error saying permission denied.

```
$ lastb
raghu tty2 Mon Jul 9 10:16 - 10:16 (00:00)
UNKNOWN tty2 Mon Jul 9 10:15 - 10:15 (00:00)
ubuntu tty8 :1 Mon Jul 2 10:23 - 10:23 (00:00)

btmpt begins Mon Jul 2 10:23:54 2012
```

du command

The du command determines disk usage of a file. If the argument given to it is a directory, then it will list disk usage of all the files and directories recursively under that directory:

```
$ du /etc/passwd
4 /etc/passwd

$ du hello/
52 hello/HelloApp
4 hello/orb.db/logs
```

```
20 hello/orb.db
```

```
108 hello/
```

df command

The df reports file system usage. For example:

```
$ df
```

```
Filesystem 1K-blocks Used Available Use% Mounted on
/dev/sda7 10079084 7372872 2194212 78% /
none 1522384 768 1521616 1% /dev
none 1529012 252 1528760 1% /dev/shm
none 1529012 108 1528904 1% /var/run
none 1529012 4 1529008 1% /var/lock
/dev/sda8 5039616 3758824 1024792 79% /home
/dev/sda2 209715196 196519248 13195948 94% /media/Data
```

command

... fdisk is a tool for getting partition information, and for adding and removing partitions. The fdisk tool requires super user privileges. To list all the partitions of all the hard drives available:

```
$ fdisk -l
```

```
Disk /dev/sda: 320.1 GB, 320072933376 bytes
255 heads, 63 sectors/track, 38913 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x396f396f

Device Boot Start End Blocks Id System
/dev/sda1 1 2611 20971520 7 HPFS/NTFS
/dev/sda2 2611 28720 209715200 7 HPFS/NTFS
/dev/sda3 * 28720 38914 81882113 5 Extended
/dev/sda5 28720 33942 41943040 7 HPFS/NTFS
/dev/sda6 33942 34464 4194304 7 HPFS/NTFS
/dev/sda7 34464 35739 10240000 83 Linux
/dev/sda8 35739 36376 5120000 83 Linux
/dev/sda9 36376 36886 4096000 82 Linux swap / Solaris
/dev/sda10 36887 38276 11164672 83 Linux
/dev/sda11 38277 38914 5117952 83 Linux
```


The fdisk is an interactive tool to edit the partition table. It takes a device (hard disk) as an argument, whose partition table needs to be edited.

```
$ fdisk /dev/sda
```

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to switch off the mode (command 'c') and change display units to sectors (command 'u').

Command (m for help): m

Command action

a toggle a bootable flag

b edit bsd disklabel

c toggle the dos compatibility flag

d delete a partition

l list known partition types

m print this menu

n add a new partition

o create a new empty DOS partition table

p print the partition table

q quit without saving changes

s create a new empty Sun disklabel

t change a partition's system id

u change display/entry units

v verify the partition table

w write table to disk and exit

x extra functionality (experts only)

Pressing 'm' at the fdisk prompt prints out the help shown above that lists all the commands available for fdisk. A new partition can be created with 'n' and an existing partition can be deleted with the 'd' command. When you are done editing the partitions, press 'w' to write the changes to the disk, and finally, hit 'q' to quit from fdisk (q does not save changes).

netstat command

The 'netstat' is a command used to check the network statistics of the system. It will list the current network connections, routing table information, interface statistics, masquerade connections and a lot more information.

```
$ netstat | head
```

```
Active Internet connections (w/o servers)
```

```
Proto Recv-Q Send-Q Local Address Foreign Address State
```

```
Active UNIX domain sockets (w/o servers)
```

```
Proto RefCnt Flags Type State I-Node Path
```

```
unix 13 [ ] DGRAM 8498 /dev/log
unix 2 [ ] DGRAM 6824 @/org/kernel/udev/udev
unix 3 [ ] STREAM CONNECTED 56738 /var/run/dbus/system_bus_socket
unix 3 [ ] STREAM CONNECTED 56113
unix 3 [ ] STREAM CONNECTED 29138
unix 3 [ ] STREAM CONNECTED 29137
```

history command

This command shows the commands you have entered on your terminal so far.

wd command

change your password with 'passwd' command.

own -h now

ly, you can shut down your system using this command.

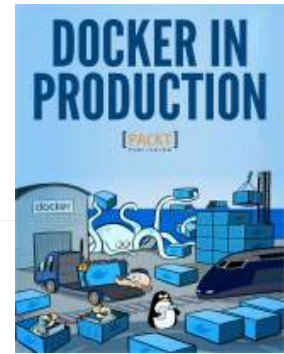
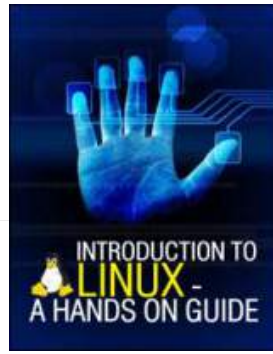
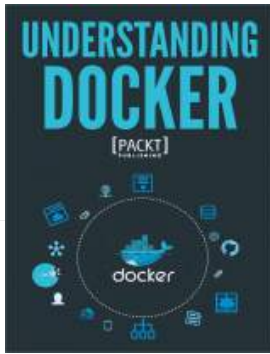
might also like

- [How To Use Linux Kill Command](#)
- [Linux Command To List Directories / Directory Names Only](#)
- [Example – How To Use Tee Command In Linux](#)
- [Linux Hdparm Command: Display Hard Disk Model and Serial Number](#)
- [10 Example Scenarios To Learn Linux Find Command](#)
- [6 Examples Linux Chage Command : Set Password Aging For User](#)
- [Linux Awk Command With Examples To Make It Easy](#)
- [Few Netstat Command Example Useful For Daily Usage](#)
- [Commands Check Linux OS is 64 / 32 bit and CPU 64 / 32 Bit](#)
- [What All Command Used To Find Architecture Of Linux OS and CPU](#)

Filed Under : [LINUX COMMANDS](#)

Tagged With : [Command](#)

Free Linux Ebook to Download



[PREVIOUS ARTICLE](#)

[Rpmforge Repo : How To Install And Set Yum Priority](#)

[CT ARTICLE](#)

[How To Start / Stop Daemons In Linux](#)

Comments (2)

[back URL](#) | [Comments RSS Feed](#)



1. *Mike F* says:

[January 13, 2017 at 8:03 pm](#)

What's with the leading \$ in the commands ?

If I type: \$ whatis date

I get:

-bash: \$: command not found

[Reply](#)



o *Bobbin Zachariah* says:

[January 15, 2017 at 3:22 pm](#)

\$ - Means what kind of user account you are logging in. Dollar (\$) sign means you are a normal user and hash (#) means you are the system administrator. (user root) In some Unix system, the prompt mean ended with percentage (%) sign which mean the user is using C Shell

[Reply](#)

Leave a Reply

Name(required) Email (required; will not be published) Website**menting Policy:**

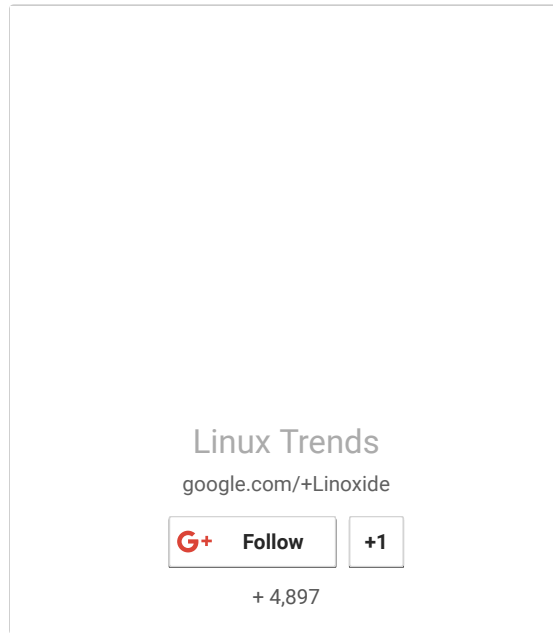
otion of your products ? Comment gets deleted.

omments are subject to moderation.

t is linuxide based on ? Windows or Linux ? Type your answer into the box

Popular Posts

- 1 [Linux Commands Cheat Sheet in Black & White](#)
- 2 [Shell Script to Check Linux System Health](#)
- 3 [Colourful ! systemd vs sysVinit Linux Cheatsheet](#)
- 4 [Intro to Configure IPsec VPN \(Gateway-to-Gateway \) using Strongswan](#)
- 5 [How to Install CentOS 7 Step by Step with Screenshots](#)
- 6 [How to Install and Setup Asterisk 13 \(PBX\) on Centos 7.x](#)
- 7 [Linux ntopng - Network Monitoring Tool Installation \(Screenshots\)](#)
- 8 [How to Install / Setup rtorrent and rutorrent in CentOS / Ubuntu](#)
- 9 [How to Install GNOME 3.16 On Ubuntu, Mint and Arch Linux](#)
- 10 [How to Setup ClipBucket to Start Video Sharing Website in Linux](#)
- 11 [How to Install Roundcube Webmail on Ubuntu16.04](#)
- 12 [Install Vivaldi in Linux - Awesome Multi Platform Browser](#)
- 13 [70 Shell Scripting Interview Questions & Answers](#)
- 14 [How To Use Ip Command In Linux with Examples](#)
- 15 [25 Linux Interview Questions and Answers \(Basic and Scenarios\)](#)



Your Matched Resources



[Linux
Commands
Cheat Sheet...](#)

[Unbelievable !
30 Linux TOP
Command...](#)

[Important 10
Linux ps
command...](#)

[What is Linux
systemd ? A
Neat Chart of...](#)

[How To Use Ip
Command In
Linux with...](#)

[LINUX
COMMANDS](#)

[20 Linux Is
Command
Examples to...](#)

[15 Linux cp
Command
Examples -...](#)

[Moving or
Rename File /
Directory in...](#)

[Linux whoami
command -
Knowing Who...](#)

[25 Linux](#)

▼ [Interview
Questions](#)

Subscribe To Free Newsletter

Enter email address to subscribe and receive notifications of new posts by email

Join **10400+** Other Subscribers

SUBSCRIBE

100% privacy - We will never spam you

Ad closed by **Google**

Stop seeing this ad

Ads by Google ⓘ

LINUX COMMANDS

All You Need to Know About Linux Kill Command From Scratch

20 SEP, 2016

How to Install perf Tool on CentOS and Ubuntu

19 SEP, 2016

[Unix Z Commands – Zcat, Zless, Zgrep, Zegrep and Zdiff Examples](#)

7 SEP, 2016

[How to Track Progress % of Commands \(cp,dd,tar\) in CentOS 6 / Fedora](#)

5 SEP, 2016

[Run Command Parallel on Multiple Hosts](#)

LINUX TOOLS

[using PDSH Tool](#)
[How to Setup Network Monitoring with OpenNMS on Ubuntu 16.04](#)



[Contact Us](#) | [Privacy](#) | [Terms of Service](#)

[B, 2017](#)
[erful SSH Command Options with Examples on Linux](#)
[v to Make Call using Opensource Voip Phone on Ubuntu](#)

[AN, 2017](#)
[ch - Repeat Linux / Unix Commands at Regular Intervals](#)
[v to Setup Teampass Password Manager on Ubuntu 16.04](#)

[30 JAN, 2017](#)
[A Peep into Process Management Commands in Linux](#)
[How to Setup Foreman to Manage Puppet Nodes on Ubuntu 16.04](#)

[26 JAN, 2017](#)
[How to Manage Network using nmcli Tool in RedHat / CentOS 7.x](#)
[How To Synchronize Directories using Syncthing on Ubuntu 16.04](#)

SUBSCRIBE TO FREE NEWSLETTER

[24 JAN, 2017](#)
[Interface \(NICs\) Bonding in Linux using nmcli](#)

Enter your email address to subscribe to this blog and receive notifications of new posts by email

5 JAN, 2015

Join **10400+** Other Subscribers

Enter Your Email * SUBSCRIBE [Server in CentOS 7.x](#)

28 NOV, 2014

100% privacy - We will never spam you

[A Great Tool To Show Linux Command Progress Like %, ETA](#)

TAGS

[Apache](#) - [Backup](#) - [Clustering](#) - [Command](#) - [Container](#) - [Docker](#) - [Filesystem](#) - [Firewall](#) - [Freebsd](#) - [Monitoring](#) - [Network](#)

[Nginx](#) - [Ssh](#) - [Storage](#) - [Ubuntu 16](#)
[Linux blkid Command to Find Block Devices Details](#)

21 NOV, 2014

[Important 10 Linux ps command Practical Examples](#)

19 NOV, 2014

[timedatectl - Control Linux System Time and Date in Systemd](#)

5 NOV, 2014

[rsync Command to Exclude a List of Files and Directories in Linux](#)

27 OCT, 2014

[iotop - Monitor and Find Statistics for Linux Processes](#)

17 OCT, 2014

[lsof - List Open Files Tool to Identify Sockets / Network Connections with Examples](#)

17 OCT, 2014

[Colourful ! systemd vs sysVinit Linux Cheatsheet](#)

5 SEP, 2014

[Awesome ! systemd Commands to Manage Linux System](#)

29 AUG, 2014

[8 Options to Trace/Debug Programs using Linux strace Command](#)

18 AUG, 2014

[How to Change Hostname in RHEL / CentOS 7.0](#)

16 AUG, 2014

[Linux Systemd - Start/Stop/Restart Services in RHEL / CentOS 7](#)

15 AUG, 2014

[A Pocket Guide for Linux ssh Command with Examples](#)

13 AUG, 2014

[How to Merge Directory Trees in Linux using cp Command](#)

23 JUL, 2014

[Linux slabtop command - Display Kernel Slab Cache Information](#)

11 JUL, 2014

[Linux chpasswd Command - Change Passwords in Bulk Mode](#)

7 JUL, 2014

[Advanced Directory Navigations Tips and Tricks in Linux](#)

JN, 2014

[How to use Linux lsblk Command to List Block Device Information](#)

JN, 2014

[Linux script command - A recorder inside your Terminal](#)

30 MAY, 2014

[How To Use Ip Command In Linux with Examples](#)

2 MAY, 2014

[Why htop Command Compete Linux top Command](#)

28 APR, 2014

[Linux findmnt Command To Find Mounted Filesystems](#)

28 MAR, 2014

[Linux Commands Cheat Sheet in Black & White](#)

21 MAR, 2014

[What Options you use for Linux killall Command ? I have 5 Options](#)

14 MAR, 2014

[10 ssh options for a Secure shell for Safe Data Communication](#)

2 MAR, 2014

[Some powerful options of Linux Dig command to Query DNS](#)

26 FEB, 2014

[Power of Linux wget Command to Download Files from Internet](#)

24 FEB, 2014

[10 all Options for DNS lookup using Linux nslookup command](#)

26 FEB, 2014

[10 head Command Options with Examples - Display First Part of Files](#)

26 FEB, 2014

[Linux more Command - Display Long Text File Per page at a Time](#)

10 FEB, 2014

[Linux touch command - Why do we need to change Timestamp?](#)

5 FEB, 2014

[Moving or Rename File / Directory in Linux - 10 Practical mv Command Examples](#)

3 FEB, 2014

[Linux rm Command Examples - Should be Aware and Cautious](#)

27 JAN, 2014

[Create Directory - subdirectory, other than that What mkdir command do in Linux](#)

23 JAN, 2014

[15 Linux cp Command Examples - Create a Copy of Files and Directories](#)

14 JAN, 2014

[Linux pwd command - Know Your Current Working Directory](#)

10 JAN, 2014

[20 Linux ls Command Examples to Display the Entries of Directory](#)

9 JAN, 2014

[userdel Command - Delete User Account from Linux system](#)

1 JAN, 2014

[ix free Command - Display Free and used nory in the System](#)

EC, 2013

[Understanding Linux cd Command with mples](#)

EC, 2013

[Linux who command - Displays who is on the system](#)

22 DEC, 2013

[Linux id Command - Print user ID and group ID information](#)

21 DEC, 2013

[Linux iostat Command to Report CPU Statistics and I/O Statistics](#)

20 DEC, 2013

[Linux vmstat Command - Tool to Report Virtual Memory Statistics](#)

17 DEC, 2013

[Linux lsusb Command to Print information about USB on System](#)

14 DEC, 2013

[Linux mpstat Command - Reports Processors Related Statistics](#)

13 DEC, 2013

[Linux whoami command - Knowing Who is Logged In](#)

12 DEC, 2013

[Linux dmesg Command to Print or Control the Kernel Ring Buffer](#)

10 DEC, 2013

[How to use Linux Netcat Command as Port Scanner](#)

10 DEC, 2013

[ix w Command to See Who is Logged In What Doing](#)

EC, 2013

[ix cal and ncal Commands to Display Under](#)

C, 2013

[Linux date command - Display and Set System Date and Time](#)

7 DEC, 2013

[Using Password Aging to Secure your Linux Access](#)

6 DEC, 2013

[Built in Audit Trail Tool - Last Command in Linux](#)

3 DEC, 2013

[How To Display And Set Hostname in Linux](#)

1 DEC, 2013

[Unbelievable ! 30 Linux TOP Command Examples With Screenshots](#)

28 NOV, 2013

[Linux Uptime Command - Find How Long Your System Been Running](#)

24 NOV, 2013

[20 Linux Cat Command Examples For File Management](#)

23 NOV, 2013

[Linux diff Command Explained With Examples](#)

20 NOV, 2013

[9 Linux Uname Command Examples To Get Operating System Details](#)

17 NOV, 2013

[Package Management Using YUM In Red Hat IX](#)

14 NOV, 2013

[Queries in Linux RPM Command to Query Packages](#)

14 SEP, 2013

[Linux Pmap Command - Find How Much Memory Process Use](#)

11 SEP, 2013

[Linux Change File / Directory Ownership - Chown Command](#)

5 SEP, 2013

[Linux Change File Permission - Chmod Command](#)

2 SEP, 2013

[Power Of Linux Sort Command](#)

26 AUG, 2013

[Power Of Linux sed Command](#)

19 AUG, 2013

[Fg/Bg Command - Linux Manage Process Foreground / Background](#)

12 AUG, 2013

[Linux wc Command - File Count Options](#)

5 AUG, 2013

[Linux Groupadd Command - Adding New Groups](#)

29 JUL, 2013

[Managing Linux Services at Startup With Chkconfig Command](#)

22 JUL, 2013

[Linux User Add Command And Its All Options](#)

8 JUL, 2013

[Linux Tail Command Options With Examples](#)

1 JUL, 2013

[Linux Process And Directory Structure Tree Commands](#)

1 JUN, 2013

[Linux du Command All Options To Study](#)

20 JUN, 2013

[Linux - Finger Command To Find User Details](#)

29 MAY, 2013

[Linux df command - Disk Space Usage For Filesystems](#)

9 MAY, 2013

[Chattr Command or Change Attributes of File](#)

5 MAY, 2013

[Linux ifconfig : How To Set Ip Address / Configure Interface](#)

17 APR, 2013

[Linux Umask : Permission Set When New File / Folder Created](#)

15 APR, 2013

[5 Ways To Find Linux Kernel Version](#)

11 APR, 2013

[How To Tar Linux Files Into .tgz Compressed Extension](#)

3 APR, 2013

[Linux Word Count wc Command With Example](#)

18 MAR, 2013

[Manage Linux Group Using gpasswd Command](#)

14 MAR, 2013

[ailed Understanding Of Linux Inodes With mple](#)

B, 2013

[ix - Limit Process At User Level](#)

\N, 2013

[ix - List / Show Users Thats Belongs Corresponding Groups](#)

28 JAN, 2013

[Fdisk Commands - Manage Partitions in Linux](#)

26 JAN, 2013

[Commands To Understand Page Faults Stats In Linux](#)

10 JAN, 2013

[Learn Linux Nmap Command With Examples](#)

29 DEC, 2012

[Linux - How To Safely Remove External Drives](#)

11 DEC, 2012

[Linux - Memory Analysis With Free and Pmap Command](#)

8 DEC, 2012

[Examples - Linux Shutdown Commands](#)

5 DEC, 2012

[Example : Linux Command To Add User To A Group](#)

30 NOV, 2012

[Create User In Linux From Command Line And Its Options](#)

28 NOV, 2012

[Linux Command To Change Password For Root / User / Group](#)

25 NOV, 2012

[Example How To Use Linux nohup Command](#)

OV, 2012

[Linux Command To Change Runlevels](#)

OV, 2012

[How To : Reboot Linux Server Command Line And Remotely](#)

10 NOV, 2012

[Rsync Over Ssh On Different Port In Linux](#)

3 NOV, 2012

[Linux Command To List Directories / Directory Names Only](#)

2 NOV, 2012

[Linux Usermod Command To Modify User Details](#)

20 OCT, 2012

[Linux Command To Show / List All Users In the System](#)

18 OCT, 2012

[Run Command In Linux - Run Shell With Specified User / Group ID](#)

2 OCT, 2012

[Linux System Performance Monitoring Using Sar Command](#)

12 SEP, 2012

[How To Mount & Umount Different Devices In Linux](#)

1 SEP, 2012

[How To Use Linux Grep Command To Find Strings](#)

6 MAY, 2012

[What Is Linux RPM And Installing, Uninstalling, Updating, Querying, Verifying RPM](#)

CT, 2011

[Examples How To Use Linux Tar Command All Options](#)

P, 2011

[Learn How To Use Linux vi Editor And Its Commands](#)

7 SEP, 2011

[Linux Commands To Manage Shared Libraries](#)

6 SEP, 2011

[8 Examples - Linux du Command With All Its Options](#)

6 SEP, 2011

[15 Difference Between Linux Yum and Up2date Command](#)

6 SEP, 2011

[Example - How To Use Tee Command In Linux](#)

22 AUG, 2011

[13 Examples To Explain Linux Netstat Command](#)

22 AUG, 2011

[Learn Linux DD Command - 15 Examples With All Options](#)

24 JUN, 2011

[Linux Watch Command - Repeat A Command At Regular Intervals](#)

21 JUN, 2011

[7 Examples Linux Dmidecode Command - Display Hardware Details In BIOS](#)

27 MAY, 2011

[8 Examples For Linux Zip And UnZip Command](#)

18 MAY, 2011

[Basic Linux Commands To Learn With Examples](#)

AY, 2011

[Commands Check Linux OS is 64 / 32 bit and 64 / 32 Bit](#)

AY, 2011

[Linux Print Command : Usage Of Linux lp Command With Examples](#)

10 MAY, 2011

[6 Examples Linux Chage Command : Set Password Aging For User](#)

2 MAY, 2011

[10 Example Scenarios To Learn Linux Find Command](#)

30 APR, 2011

[Linux Awk Command With Examples To Make It Easy](#)

30 APR, 2011

[What All Command Used To Find Architecture Of Linux OS and CPU](#)

30 APR, 2011

[Linux Hdparm Command: Display Hard Disk Model and Serial Number](#)

24 APR, 2011

[Everyday Linux ps command Examples for System Admin](#)

24 APR, 2011

[How To List Linux Pci Devices Using lspci Command](#)

23 APR, 2011

[How To Use Screen And Script Command In Linux](#)

23 APR, 2011

[Different Options of Rsync Command in Linux With Examples](#)

PR, 2011

[How To Use Linux Route Add Command](#)

PR, 2011

[Few Netstat Command Example Useful For Daily Usage](#)

23 APR, 2011

[Can Linux sfdisk Tool Used Instead Of fdisk](#)

20 MAR, 2011

[Can dd Command Used To Wipe Data? Examples](#)

20 MAR, 2011

[How To Use Linux Kill Command](#)

20 MAR, 2011

[How To Backup LVM Configuration On Linux \(vgcfgbackup & vgcfgrestore\)](#)

25 FEB, 2011

[How to Create Partition in Linux using fdisk Tool](#)

25 FEB, 2011

[How To Use Linux Tr Command With Examples](#)

22 FEB, 2011
