

81 Linux Commands – You must learn

Every Linux administrator should learn these commands

Prepared for:
Public

Prepared by:
Ankam Ravi Kumar

Follow Me on Social Networking Sites

[Facebook](#) | [Twitter](#) | [Reddit](#) | [LinkedIn](#) | [Website](#) | [Blog](#) | [YouTube](#)

Reach me over Email: aravikumar48@gmail.com or aravi@server-computer.com

If you think this document helped a lot [Donate](#) a dollar as complementary

Table of Contents

1. About Author	4
2. cat - command to see file content	5
3. cd – Change directory path	6
4. ls command – listing files and directories	8

How to Use this Guide



Marked as key points, important points and examination points to remember

1. About Author

Ankam Ravi Kumar has more than 10+ years of experience in Information Technology Operations and production support streams. He served more than 5 companies in his career and still continuing.

We provide server and data center related services from purchasing of underlying hardware to provisioning the applications.

Solid industry experience in Infrastructure Management/Customer Support/Operations and Training Domains. I love to help people by sharing my knowledge and skills. I always believe “Power is gained by Sharing Knowledge not hoarding it”.

- Operating System Management Such has Linux Different Flavors, Red hat, Fedora, Ubuntu, AIX, Solaris and Windows
- Enterprise Server Management
- Installing and configuring Blade Servers
- Core Storage Management Dell-EMC, IBM and NetApp
- Database Management MSSQL, POSTGRESQL, MariaDB and MySQL
- Process Management ITIL
- Virtualization management RHEV, vSphere, VMware, KVM, Hyper-V and XEN
- Backup and Recovery Management NetVault, Commvault and Symantec Backup Exec
- Application Server Management and Storage Cluster Management
- Data Center Management and Hosting Solutions
- Programming Languages such as PHP and HTML
- Scripting Languages Shell, Perl and Python
- Cloud administration AWS, Azure and GCP

Specialized in managing and building the Teams for IT services delivery and Service Support, Training and Operations in both smaller and larger companies. Rich experience and strong exposure in IT Infrastructure & Data Center Management.

Implementation of monitoring solutions for Enterprise, Using Tools Nagios, NagiosXI, Cacti, Solarwinds and LogicMonitor.

2. cat - command to see file content

Concatenate files and print on the standard output. Cat command used to see the content of text files. Cat is a core utility, which comes along with Operating System installation no need to install separately.

Syntax: `cat <Options> <File Name>`

```
[user@rhel7 ~]$ cat catfile
This is the first line of the file
second line of the file

after one space this is another line of text

last line in the file
```

Without any options cat can show the content of text file, example shown above screenshot.

-b, --number-nonblank: number nonempty output lines, overrides -n

```
[user@rhel7 ~]$ cat -b catfile
 1 This is the first line of the file
 2 second line of the file

 3 after one space this is another line of text

 4 last line in the file
```

Using -b option cat command will show the content with numbered lines, excluding blank lines.

-E, --show-ends: display \$ at end of each line

```
[user@rhel7 ~]$ cat -E catfile
This is the first line of the file$
second line of the file$
$
after one space this is another line of text$
$
$
last line in the file$
```

Option capital E will add \$ (dollar sign) to end of the line including empty lines

-n, --number: number all output lines

```
[user@rhel7 ~]$ cat -n catfile
 1 This is the first line of the file
 2 second line of the file
 3
 4 after one space this is another line of text
 5
 6
 7 last line in the file
```

Add numbers to all the lines including empty lines using option `-n` cat command

```
[user@rhel7 ~]$ cat --version
cat (GNU coreutils) 8.22
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by Torbjörn Granlund and Richard M. Stallman.
```

Current version 8.22. All above options will work properly as shown in examples. Please check if version change options may change.

There are many options cat command has however above are the options most useful.

3. cd – Change directory path

Change the current directory to dir. The variable HOME is the default dir. By default when you type cd without options, it will go back to HOME path.

Syntax: `cd <Options> <DIRPATH>`

```
[user@rhel7 ~]$ cd /opt/
[user@rhel7 opt]$ cd
[user@rhel7 ~]$ cd -
/opt
[user@rhel7 opt]$ cd ~
[user@rhel7 ~]$ pwd
/home/user
[user@rhel7 ~]$ cd /var/log/
[user@rhel7 log]$ cd ..
[user@rhel7 var]$ cd /etc/httpd/conf.d/
[user@rhel7 conf.d]$ cd ../../
[user@rhel7 etc]$ pwd
/etc
[user@rhel7 etc]$
[user@rhel7 etc]$
[user@rhel7 etc]$ cd
[user@rhel7 ~]$
```


Before jumping into the `cd` command and its options, you have to know about absolute path and relative path.

1. Absolute path or Full path points to the location exactly regardless of current directory. In simple terms absolute paths always start with (/) slash
2. Relative path or shortcut path. Relative path starts from some given working directory, avoiding the need to provide the full path.

Example: I am in `/var` directory if I want to change path to `/var/log/` no need to provide full path just `cd log/` do the job.

(Tilde)~ = \$HOME or user home directory path, when you type `cd ~` wherever your jump back to user home path

```
[user@rhel7 ~]$ ls -la
total 1116
drwx----- . 13 user user 4096 May 29 15:06 .
drwxr-xr-x. 7 root root 72 Feb 25 16:44 ..
```



In every directory path there are two hidden directories will be available which are. (Dot) and.. (Dot dot).

Single dot means **current directory**

Two dots means its **parent directory**

```
[user@rhel7 ~]$ cd /var/log/
[user@rhel7 log]$ cd ..
[user@rhel7 var]$ cd .
[user@rhel7 var]$ pwd
/var
[user@rhel7 var]$
```

If you have symbolic links, you want to change to its original path not symbolic links then use `-P` along with `cd` command

```
[user@rhel7 ~]$ cd /
[user@rhel7 /]$ ls
ansible bin boot data dev etc home lib lib64
[user@rhel7 /]$ cd -P bin
[user@rhel7 bin]$ pwd
/usr/bin
[user@rhel7 bin]$ cd /
[user@rhel7 /]$ cd -L bin
[user@rhel7 bin]$ pwd
/bin
[user@rhel7 bin]$
```

cd - You will go back to previous directory path. An argument of `-` is equivalent to `$OLDPWD`

```
[user@rhel7 bin]$ cd /etc/httpd/conf.d/
[user@rhel7 conf.d]$ cd -
/bin
[user@rhel7 bin]$
```

That is about `cd` command

4. ls command – listing files and directories

`ls` command is used to list information about the FILES (the current directory by default). Sort entries alphabetically if none of `-cftuvSUX` nor `--sort` is specified

Syntax: `ls [options] arguments`

Listing current directory content - You can list out all files and directories in current path using `'ls'` without hidden directories


```
# ls
[user@Server-Computer ls]$ ls
Server-Computer1.txt  Server-Computer3.txt  Server-Computer5.txt  Server-Computer7.txt
Server-Computer2.txt  Server-Computer4.txt  Server-Computer6.txt  Server-Computer8.txt
```

Listing including hidden files - default ls command will not list hidden objects, to see the hidden objects you have to use option '-a'

```
# ls -a
[user@Server-Computer ls]$ ls -la
total 4
drwxrwxr-x  2 user user  258 Jun 17 16:37 .
drwx----- 14 user user 4096 Jun 17 16:36 ..
-rw-rw-r--  1 user user   0 Jun 17 16:36 Server-Computer1.txt
-rw-rw-r--  1 user user   0 Jun 17 16:36 Server-Computer2.txt
-rw-rw-r--  1 user user   0 Jun 17 16:36 Server-Computer3.txt
-rw-rw-r--  1 user user   0 Jun 17 16:36 Server-Computer4.txt
-rw-rw-r--  1 user user   0 Jun 17 16:36 Server-Computer5.txt
```

Long list (it display detailed info) - when you use 'ls' command it will list files & directories without their permissions, owner, group and other details, '-l' option will display owner, group, others, Size and time, which is called as long list.

```
# ls -l
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer1.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer2.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer3.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer4.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer5.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer6.txt
```

permissions Owner Group Date & time

List files & directories separated by comma - if there is a requirement that we have to list all the files and directories separated by comma and import to CSV file '-m' is the best option to use

```
# ls -m
[user@Server-Computer ls]$ ls -m
Server-Computer1.txt, Server-Computer2.txt, Server-Computer3.txt, Server-Computer4.txt
```

List remote directory files & directories with & without color – list remote directory content with color and without color below is the example

```
ls --color=always
```

Without color, here all the files and directories will display in block color

```
# ls --color=never
```

```
[user@Server-Computer ls]$ ls --color=always
Server-Computer1.txt  Server-Computer3.txt  Server-Computer5.txt  Server-Computer7.txt  Server-Computer9.txt
Server-Computer2.txt  Server-Computer4.txt  Server-Computer6.txt  Server-Computer8.txt  testing.tar.gz
[user@Server-Computer ls]$ ls --color=never
Server-Computer1.txt  Server-Computer3.txt  Server-Computer5.txt  Server-Computer7.txt  Server-Computer9.txt
Server-Computer2.txt  Server-Computer4.txt  Server-Computer6.txt  Server-Computer8.txt  testing.tar.gz
```

List only directory using option ‘-d’ - option ‘-d’ will display only directory path without its content

```
# ls -d /etc/
```

```
[user@Server-Computer ls]$ ls -ld lsdire/
drwxrwxr-x 2 user user 6 Jun 19 16:56 lsdire/
[user@Server-Computer ls]$ ls -d lsdire/
lsdire/
```

List files & directories detailed time stamp - As we can see above -l option, which will provide long, output, but time stamp is not detail (Example: no year), by following below example it will provide time stamp details including time zone and seconds

```
# ls -l --time-style=full-iso
```

```
[user@Server-Computer ls]$ ls -l --time-style=full-iso
total 4
drwxrwxr-x 2 user user 6 2019-06-19 16:56:38.183590269 +0530 lsdire
-rwxrwxrwx 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer1.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer2.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer3.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer4.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer5.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36:52.590439468 +0530 Server-Computer6.txt
```

Display files and directories its time stamp as “YY-MM-DD HH:MM” - No need to think a lot to list files and directories with Year-Month-Date Hours and Minutes format

```
# ls -l --time-style long-iso
[user@Server-Computer ls]$ ls -l --time-style long-iso
total 4
drwxrwxr-x 2 user user 6 2019-06-19 16:56 lsdire
-rwxrwxrwx 1 user user 0 2019-06-17 16:36 Server-Computer1.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36 Server-Computer2.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36 Server-Computer3.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36 Server-Computer4.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36 Server-Computer5.txt
-rw-rw-r-- 1 user user 0 2019-06-17 16:36 Server-Computer6.txt
```

List objects its time stamp as “MM-DD HH:MM” List all files & directories with date format YY-MM-DD HH:MM

```
# ls -l --time-style iso
```

Display files & directories with customized time stamp format - Here in this option we can mention date command options to customize time stamp as required

```
# ls -l --time-style="+%Y-%m-%d %H:%M:%S"
```

Note: refer [man date](#) command for more options

List Only Group Name without User name (Owner)

```
# ls -lg
```

List Only Owner Name without Group Name - ls command will list only user name (owner name) excluding group details we have to use ‘-G’ option along with option ‘-l’ option

```
# ls -lG
```

List files & directories in human readable format - ls command will give you a nice human readable format of ls command output use option ‘-lh’

```
# ls -lh
```

```
[user@Server-Computer ls]$ ls -lh
total 4.0K
drwxrwxr-x 2 user user 6 Jun 19 16:56 lsdire
-rwxrwxrwx 1 user user 0 Jun 17 16:36 Server-Computer1.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer2.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer3.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer4.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer5.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer6.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer7.txt
```

List assigned inode number of files and directories - we can also list the inode numbers of files and directories using option '-i'

```
# ls -i
```

Files & directories, directories should append with / (slash) - To append the directories with / (slash) we have to use option '-p'. If you observe below example output all the directories ended with / (slash).

```
# ls -p
a anaconda-ks.cfg b c dir1/ dir2/ initial-setup-ks.cfg
```

Print files & directory names in quoted format - As we can list the files and directories using ls command, file names will be printed as in quotes format using '-Q' option

```
# ls -lQ
# ls -Q
```

Sort the list by time stamp- you can sort the files and directories by its time, this option will list old time stamp below newer up, which is most useful option to know new files

```
# ls -lt
```

Print in reverse - as above example prints output old files below and newer files up. using '-r' option print in reverse way, old first and new last

```
# ls -ltr
```

List recursively - using option '-R' you can list files and directories in recursively

```
# ls -R
```

Print the allocated size of each file, in blocks, using option '-S' you can sort by file size

```
# ls -lS
```

Sort by alphabetical order – List files alphabetical order you have to use -X option

```
# ls -lX
```

```
[user@Server-Computer ls]$ ls -lX
total 4
-rw-rw-r-- 1 user user 0 Jun 19 17:43 a
-rw-rw-r-- 1 user user 0 Jun 19 17:43 b
-rw-rw-r-- 1 user user 0 Jun 19 17:43 c
drwxrwxr-x 2 user user 6 Jun 19 16:56 lsdire
-rw-rw-r-- 1 user user 200 Jun 19 16:54 testing.tar.gz
-rwxrwxrwx 1 user user 0 Jun 17 16:36 Server-Computer1.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer2.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer3.txt
-rw-rw-r-- 1 user user 0 Jun 17 16:36 Server-Computer4.txt
```

List with tab space- ls command default will display file space as eight columns, using -T we can print with more tab space in between files

```
# ls -Tl
```

Print file names in one row - using option -1 (numeric number 1)

```
# ls -l
```

```
[user@Server-Computer ls]$ ls -l
a
b
c
lsdire
Server-Computer1.txt
Server-Computer2.txt
Server-Computer3.txt
Server-Computer4.txt
```

Let us know the ls command version details and author details

```
[user@Server-Computer ls]$ ls --version
ls (GNU coreutils) 8.22
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by Richard M. Stallman and David MacKenzie.
```