

# Linux Basic Command

## Linux Command syntax:

```
command -options argument
```

### 1. pwd command: To see present directory position

```
[root@ns1 ~]# pwd
/root
```

### 2. cd command: to change current directory location.

```
[root@ns1 ~]# cd /home/
[root@ns1 home]# pwd
/home
```

### 3. ls command: to see directory contents.

```
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop  Documents  Downloads  initial-setup-ks.cfg
Music  Pictures  Public  Templates  Videos
```

ls command has some option are -a, -l, -r, -h

Table: Options with the ls command

| Directory | Description  |
|-----------|--|
| -a        | Lists all files, including hidden files  |
| -F        | Appends a special character to each filename to represent the file type, such as * for an executable file and / for a subdirectory |
| -h        | Stands for “human-readable” format, which shows file sizes in megabytes or gigabytes, for example, instead of in bytes             |
| -i        | Displays the inode number for each file  |
| -l        | Changes the display from a column format to a long list  |
| -R        | Contains temporary files that system administrators should delete whenever the system is booted                                    |
| --help    | Lists all options available with a command   |

```
[root@localhost ~]# ls -la
total 52
dr-xr-x---. 14 root root 4096 Jul 24 06:32 .
drwxr-xr-x. 17 root root 4096 Jul 24 12:30 ..
-rw-----.  1 root root 1352 Jul 18 15:08 anaconda-ks.cfg
-rw-r--r--.  1 root root   18 Dec 28 2013 .bash_logout
-rw-r--r--.  1 root root  176 Dec 28 2013 .bash_profile
-rw-r--r--.  1 root root  176 Dec 28 2013 .bashrc
drwx-----.  9 root root 4096 Jul 24 06:38 .cache
drwx-----. 15 root root 4096 Jul 24 06:38 .config
-rw-r--r--.  1 root root  100 Dec 28 2013 .cshrc
drwx-----.  3 root root   24 Jul 18 09:09 .dbus
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Desktop
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Documents
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Downloads
-rw-----.  1 root root   16 Jul 24 06:32 .esd_auth
-rw-----.  1 root root  314 Jul 24 06:32 .ICEauthority
-rw-r--r--.  1 root root 1403 Jul 18 09:10 initial-setup-ks.cfg
drwx-----.  3 root root   18 Jul 24 06:32 .local
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Music
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Pictures
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Public
-rw-r--r--.  1 root root  129 Dec 28 2013 .tcshrc
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Templates
drwxr-xr-x.  2 root root    6 Jul 24 06:32 Videos
```

#### 4. mkdir command: to create directory

```
[root@ns1 ~]# mkdir /class
```

```
[root@ns1 ~]# ls -la /class/
total 4
drwxr-xr-x  2 root root    6 Jul 15 23:57 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
```

```
[root@ns1 ~]# ls -la /class/d1
ls: cannot access /class/d1: No such file or directory
```

```
[root@ns1 ~]# mkdir /class/d1
```

```
[root@ns1 ~]# ls -la /class/
total 4
drwxr-xr-x  3 root root   15 Jul 15 23:57 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
```

```
drwxr-xr-x  2 root root    6 Jul 15 23:57 d1
```

## 5. touch command: to create an empty file

```
[root@ns1 class]# touch file1
```

```
[root@ns1 class]# ls -la
total 4
drwxr-xr-x  3 root root    27 Jul 15 23:58 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
drwxr-xr-x  2 root root    6 Jul 15 23:57 d1
-rw-r--r--  1 root root    0 Jul 15 23:58 file1
```

## 6. cp command: to copy a directory or file

```
[root@ns1 class]# cp file1 file2
```

```
[root@ns1 class]# ls -la
total 4
drwxr-xr-x  3 root root    39 Jul 15 23:59 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
drwxr-xr-x  2 root root    6 Jul 15 23:57 d1
-rw-r--r--  1 root root    0 Jul 15 23:58 file1
-rw-r--r--  1 root root    0 Jul 15 23:59 file2
```

to copy directory you have to add option '-r' with 'cp'

```
[root@ns1 class]# cp d1/ d2
cp: omitting directory 'd1/'
```

```
[root@ns1 class]# cp -r d1 d2
```

```
[root@ns1 class]# ls -la
total 4
drwxr-xr-x  4 root root    48 Jul 16 00:01 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
drwxr-xr-x  2 root root    6 Jul 15 23:57 d1
drwxr-xr-x  2 root root    6 Jul 16 00:01 d2
-rw-r--r--  1 root root    0 Jul 15 23:58 file1
-rw-r--r--  1 root root    0 Jul 15 23:59 file2
```

**7. mv command:** to move or rename directory or file one location to another location.

```
[root@ns1 class]# mv file1 d2/
```

```
[root@ns1 class]# ls -la
total 4
drwxr-xr-x  4 root root   36 Jul 16 00:05 .
drwxr-xr-x. 18 root root 4096 Jul 15 23:57 ..
drwxr-xr-x  2 root root    6 Jul 15 23:57 d1
drwxr-xr-x  2 root root   18 Jul 16 00:05 d2
-rw-r--r--  1 root root    0 Jul 15 23:59 file2
```

```
[root@ns1 class]# ls -la d2
total 0
drwxr-xr-x 2 root root 18 Jul 16 00:05 .
drwxr-xr-x 4 root root 36 Jul 16 00:05 ..
-rw-r--r-- 1 root root  0 Jul 15 23:58 file
```

**8. Cat Command:** cat command used to view multiple file at the same time.

```
# cat /etc/passwd
```

You combine more and less command with cat command to view file contain if that doesn't fit in single screen / page.

```
# cat /etc/passwd | less
# cat /etc/passwd | more
```

**9. Less Command:** less command allows quickly view file. You can page up and down. Press 'q' to quit from less window.

```
# less /etc/passwd
```

**10. More Command:** more command allows quickly view file and shows details in percentage. You can page up and down. Press 'q' to quit out from more window.

```
# more file2
```

**11. Sort command:** Sorting lines of text files in ascending order. with -r options will sort in descending order.

```
#sort /etc/passwd
```

**12. vi or vim Command:** vi and vim are most popular text editor available in most of the UNIX-like OS.

```
# vi /etc/passwd
```

**13. Uptime Command:** In Linux uptime command shows since how long your system is running and the number of users is currently logged in and also displays load average for 1, 5 and 15 minutes intervals.

```
# uptime
15:16:26 up 22 min,  1 user,  load average: 0.00, 0.03, 0.22
```

**14. Users Command:** Users command displays currently logged in users.

```
# users
mahedi
```

**15. Who Command:** who command simply return user name, date, time and host information. who command is similar to w command. Unlike w command who doesn't print what users are doing. Lets illustrate and see the different between who and w commands.

```
[root@localhost ~]# w
16:38:55 up 10:08,  3 users,  load average: 0.01, 0.03, 0.05
USER      TTY      LOGIN@  IDLE   JCPU   PCPU   WHAT
root      :0        06:32   ?xdm?  6:50   0.43s  gdm-session-worker [pam/gdm-password]
root      pts/0    13:50   2:47m  0.28s  0.28s  /bin/bash
root      pts/1    16:01   7.00s  0.43s  0.24s  w
```

```
[root@localhost ~]# who
root      :0                2017-07-24 06:32 (:0)
root      pts/0             2017-07-24 13:50 (:0)
root      pts/1             2017-07-24 16:01 (172.16.186.1)
```

**Who command Options:**

- b : Displays last system reboot date and time.
- r : Shows current run level.
- a, -all : Displays all information in cumulatively.

```
[root@localhost ~]# who -b
system boot 2017-07-24 12:30
```

```
[root@localhost ~]# who -r
run-level 5 2017-07-24 06:31
```

```
[root@localhost ~]# who -a
      system boot   2017-07-24 12:30
root   ? :0         2017-07-24 06:32   ?           12207 (:0)
      run-level 5   2017-07-24 06:31
root   + pts/0     2017-07-24 13:50 02:50       64099 (:0)
root   + pts/1     2017-07-24 16:01 .           5099 (172.16.186.1)
```

**16. Whoami Command:** whoami command prints the name of current user. You can also use “who am i” command to display the current user. If you are logged in as a root using sudo command “whoami” command return root as current user. Use “who am i” command if you want to know the exact user logged in.

```
[root@localhost ~]# whoami
root
```

```
[root@localhost ~]# who am i
root pts/1 2017-07-24 16:01 (172.16.186.1)
```

**17. Free command:** Free command shows free, total and swap memory information in bytes.

```
[root@localhost ~]# free
      total          used          free        shared    buffers         cached
Mem:   1878192      1811712         66480           9868         3984        608392
-/+ buffers/cache:    1199336         678856
Swap:   2097148           220       2096928
```

Free with -h options shows total memory used and available to use in human readable.

```
[root@localhost ~]# free -h
      total          used          free        shared    buffers         cached
Mem:   1.8G          1.7G          65M          9.6M         3.9M         594M
-/+ buffers/cache:    1.1G          663M
Swap:   2.0G           220K          2.0G
```

**18. Top Command:** top command displays processor activity of your system and also displays tasks managed by kernel in real-time. It'll show processor and memory are being used. Use top command with 'u' option this will display specific User process details as shown below. Press 'O' (uppercase letter) to sort as per desired by you. Press 'q' to quit from top screen.

```
Tasks: 288 total, 5 running, 283 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.1 us, 0.8 sy, 1.1 ni, 96.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 1878192 total, 1812252 used, 65940 free, 3984 buffers
KiB Swap: 2097148 total, 220 used, 2096928 free. 608568 cached Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+  COMMAND

```

```

12585 root      20    0 494336 15276  4728 R  7.9  0.8  0:01.37 tracker-store
12614 root      39   19 816544 10736  7348 S  6.2  0.6  0:00.39 tracker-miner-f
12535 root      20    0 1054444 22512 15712 S  0.7  1.2  0:01.16 nautilus
 6507 root      20    0 123792  1788  1152 R  0.3  0.1  0:00.03 top
    1 root      20    0  53680  7640  2532 S  0.0  0.4  0:06.18 systemd
    2 root      20    0     0     0     0 S  0.0  0.0  0:00.04 kthreadd
    3 root      20    0     0     0     0 R  0.0  0.0  0:04.72 ksoftirqd/0
    5 root       0  -20     0     0     0 S  0.0  0.0  0:00.00 kworker/0:0H
    7 root      rt    0     0     0     0 S  0.0  0.0  0:00.00 migration/0
    8 root      20    0     0     0     0 S  0.0  0.0  0:00.00 rcu_bh
    9 root      20    0     0     0     0 S  0.0  0.0  0:00.00 rcuob/0
   10 root      20    0     0     0     0 S  0.0  0.0  0:00.00 rcuob/1

```

**19. Tar Command:** tar command is used to compress files and folders in Linux. For example the below command will create an archive for /home directory with file name as archive-name.tar.

```
# tar -cvf archive-name.tar /home
```

To extract tar archive file use the option as follows.

```
# tar -xvf archive-name.tar
```

**20. grep Command:** grep search for a given string in a file. Only “mahedi” user displays from /etc/passwd file. we can use -i option for ignoring case sensitive.

```
[root@localhost ~]# grep mahedi /etc/passwd
mahedi:x:1000:1000:Md. Mahedi Hasan:/home/mahedi:/bin/bash
```

**21. Find Command:** Find command used to search files, strings and directories. The below example of find command search “passwd” word in ‘/’ partition and return the output.

```
[root@localhost ~]# find / -name passwd
/sys/fs/selinux/class/passwd
/sys/fs/selinux/class/passwd/perms/passwd
/etc/passwd
/etc/pam.d/passwd
/usr/bin/passwd
/usr/share/bash-completion/completions/passwd
```