

# Linux System Admin Command

A system administrator manages configuration, upkeep and reliable operations of computer operations. Sysadmin handles servers, has to manage system performance and security without exceeding the budget to meet users need.

## Some important commands for system administrators

Command	Function
man	Display information about all commands
uptime	Show how long system is running
users	Show username who are currently logged in
service	Call and execute script
pkill	Kill a process
pmap	Memory map of a process
wget	Download file from network
ftp or sftp	Connect remote ftp host
free	Show memory status
top	Display processor activity of system
last	Display user's activity in the system
ps	Display about processes running on the system
Shutdown commands	Shutdown and reboot system
info	Display information about given command
env	Display environment variable for currently logged-in user
netstat	Display network status
arp	Check ethernet connectivity and IP address
df	Display filesystem information
du	Display usage
init	Allow to change server bootup
nano	A command line editor
nslookup	Check domain name and IP information
shred	Delete a file by over writing its content
cat	Display, copy or combine text files
pwd>	Print path of current working directory
locate	Finding files by name on system
>alias	To short a command
echo	Display text
cmp	Compare two files byte by byte
mount	Mount a filesystem
ifconfig	Display configuration
traceroute>	Trace existing network
sudo	Run a command as a root user
route	List routing table for your server
ping	Check connection by sending packet test packet

find	Find location of files/directories
users	Show current logged in user
who	Same as w but doesn't show current process
ls	List all the files
tar	Compress directories
grep	Search for a string in a file
su	Switch from one to another user
awk	Search lines for a given pattern

## man Command

man is a built-in manual for utilizing Linux commands. It permits users to access the reference manual of a utility or a command to execute in the terminal.

**For example,**

1. man ls

This command will display all the information about 'ls' command as shown in the screen shot.

```

compaq@compaq-Compaq-620: ~
LS(1)                                User Commands                                LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
    fied.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author

Manual page ls(1) line 1 (press h for help or q to quit)

```

## Uptime Command

The uptime command tells us how long a system has been running.

**"uptime" displays output in one line.**

- **current system time**
- duration for which system has been running (system is running since 18 minutes)
- number of users logged in (2 users are logged in)
- system load average CPU load for past 1, 5 and 15 minutes.

Here, system load averages are the processes which are either in runnable or in uninterruptable state.

```
compaq@compaq-Compaq-620: ~
compaq@compaq-Compaq-620:~$ uptime
19:13:44 up 7:58, 1 user, load average: 0.88, 1.10, 1.05
compaq@compaq-Compaq-620:~$
```

## ps Command in Linux/Unix with Examples

The ps command is used to view currently running processes on the system. It helps us to determine which process is doing what in our system, how much memory it is using, how much CPU space it occupies, user ID, command name, etc .

**The ps command may display different results for different systems because it displays information about the currently running process of a system.**

**Syntax:**

1. ps

```
compaq@compaq-Compaq-620: ~
compaq@compaq-Compaq-620:~$ ps -e
  PID TTY          TIME CMD
    1 ?           00:00:04 systemd
    2 ?           00:00:00 kthreadd
    3 ?           00:00:00 rcu_gp
    4 ?           00:00:00 rcu_par_gp
    8 ?           00:00:00 mm_percpu_wq
    9 ?           00:00:00 ksoftirqd/0
   10 ?           00:00:01 rcu_sched
   11 ?           00:00:00 migration/0
   12 ?           00:00:00 idle_inject/0
   14 ?           00:00:00 cpuhp/0
   15 ?           00:00:00 cpuhp/1
   16 ?           00:00:00 idle_inject/1
   17 ?           00:00:00 migration/1
   18 ?           00:00:00 ksoftirqd/1
   20 ?           00:00:00 kworker/1:0H-kblockd
   21 ?           00:00:00 kdevtmpfs
   22 ?           00:00:00 netns
   23 ?           00:00:00 rcu_tasks_kthre
   24 ?           00:00:00 kauditd
   25 ?           00:00:00 khungtaskd
   26 ?           00:00:00 oom_reaper
   27 ?           00:00:00 writeback
```

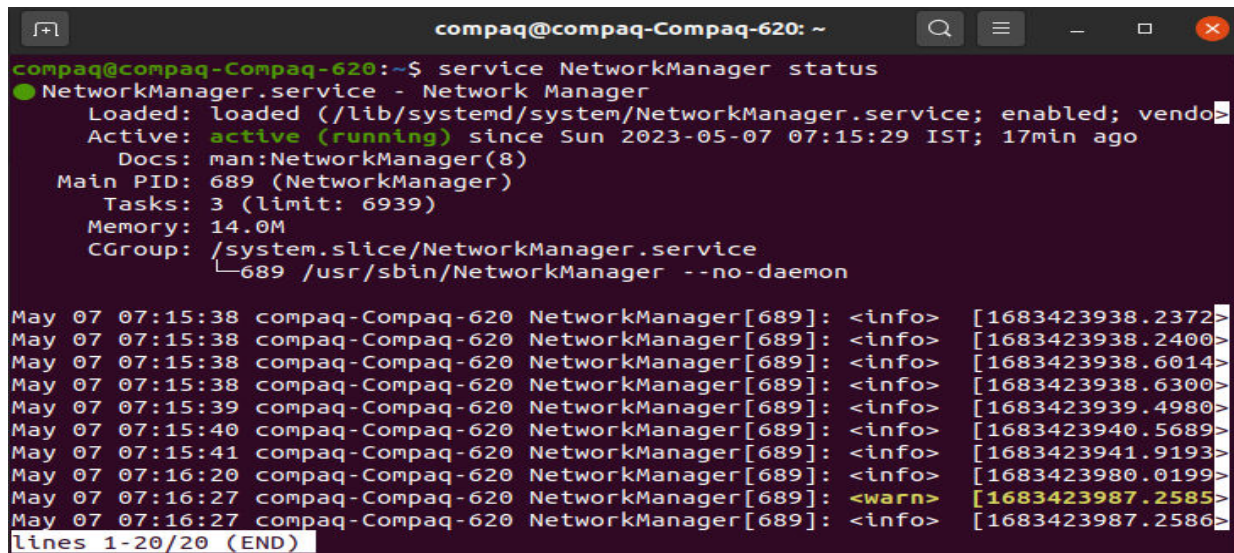
# Linux service

The service command starts, stop and restart a daemon or services by calling the script. Usually all scripts are stored in `/etc/init.d` directory.

It runs a script in as predictable environment as possible.

## Syntax:

1. `service script_name command`



```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ service NetworkManager status  
● NetworkManager.service - Network Manager  
   Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled; vendor preset: enabled)  
   Active: active (running) since Sun 2023-05-07 07:15:29 IST; 17min ago  
     Docs: man:NetworkManager(8)  
    Main PID: 689 (NetworkManager)  
      Tasks: 3 (limit: 6939)  
     Memory: 14.0M  
    CGroup: /system.slice/NetworkManager.service  
            └─689 /usr/sbin/NetworkManager --no-daemon  
  
May 07 07:15:38 compaq-Compaq-620 NetworkManager[689]: <info> [1683423938.2372]>  
May 07 07:15:38 compaq-Compaq-620 NetworkManager[689]: <info> [1683423938.2400]>  
May 07 07:15:38 compaq-Compaq-620 NetworkManager[689]: <info> [1683423938.6014]>  
May 07 07:15:38 compaq-Compaq-620 NetworkManager[689]: <info> [1683423938.6300]>  
May 07 07:15:39 compaq-Compaq-620 NetworkManager[689]: <info> [1683423939.4980]>  
May 07 07:15:40 compaq-Compaq-620 NetworkManager[689]: <info> [1683423940.5689]>  
May 07 07:15:41 compaq-Compaq-620 NetworkManager[689]: <info> [1683423941.9193]>  
May 07 07:16:20 compaq-Compaq-620 NetworkManager[689]: <info> [1683423980.0199]>  
May 07 07:16:27 compaq-Compaq-620 NetworkManager[689]: <warn> [1683423987.2585]>  
May 07 07:16:27 compaq-Compaq-620 NetworkManager[689]: <info> [1683423987.2586]>  
lines 1-20/20 (END)
```

## stop command

To stop a service use the following syntax,

## Syntax:

1. `service script_name stop`

## start command

To start a service use the following syntax,

## Syntax:

1. `service script_name start`

## restart command

To restart a service use the following syntax,

## Syntax:

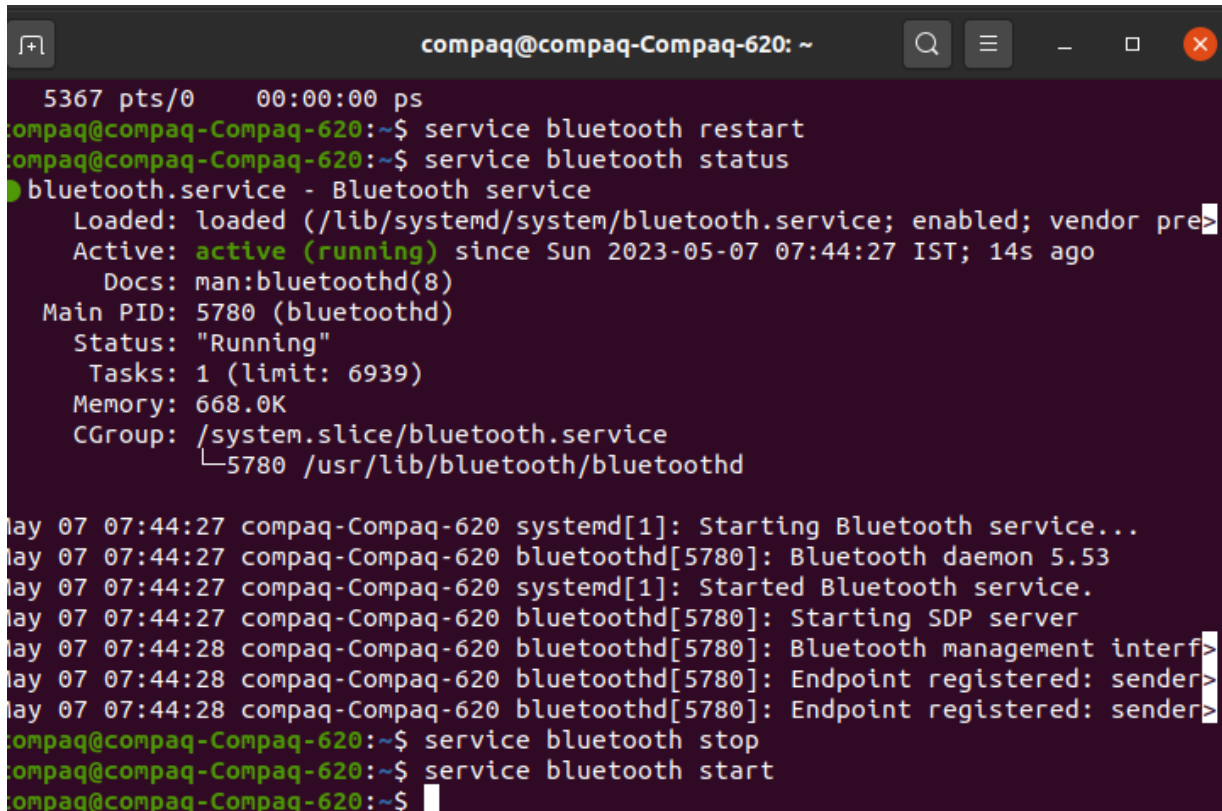
1. `service script_name restart`
-

## status command

To get current status of a service use the following syntax,

### Syntax:

1. service script\_name status

A terminal window titled 'compaq@compaq-Compaq-620: ~' showing the execution of service commands. The user runs 'ps' showing '5367 pts/0 00:00:00 ps'. Then 'service bluetooth restart' and 'service bluetooth status'. The status output for 'bluetooth.service' shows it is 'active (running)' since Sun 2023-05-07 07:44:27 IST. It lists details like 'Loaded: loaded (/lib/systemd/system/bluetooth.service; enabled; vendor pre>', 'Active: active (running) since Sun 2023-05-07 07:44:27 IST; 14s ago', 'Docs: man:bluetoothd(8)', 'Main PID: 5780 (bluetoothd)', 'Status: "Running"', 'Tasks: 1 (limit: 6939)', 'Memory: 668.0K', and 'CGroup: /system.slice/bluetooth.service' with a sub-entry for '└─5780 /usr/lib/bluetooth/bluetoothd'. Below this, a series of log messages from May 07 07:44:27 show the service starting and endpoints registering. The user then runs 'service bluetooth stop' and 'service bluetooth start'.

## Linux Terminating

There are four ways to kill or terminate a process. These commands allow you to run the system uninterrupted after terminating a process without rebooting the system. These commands can be internal or external.

### Command Function

<a href="#">kill</a>	Need to specify PID number
<a href="#">killall</a>	Kill more than one process with a single name
<a href="#">pkill</a>	Need to specify name of the process
<a href="#">xkill</a>	Kill a x server client

## How to know PID

To use terminating commands you need to know different PIDs. PID for a process can be find out with the following command,

### Syntax:

1. ps -A

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ ps -A  
  PID TTY          TIME CMD  
    1 ?           00:00:07 systemd  
    2 ?           00:00:00 kthreadd  
    3 ?           00:00:00 rcu_gp  
    4 ?           00:00:00 rcu_par_gp  
    8 ?           00:00:00 mm_percpu_wq  
    9 ?           00:00:00 ksoftirqd/0  
   10 ?           00:00:04 rcu_sched  
   11 ?           00:00:00 migration/0  
   12 ?           00:00:00 idle_inject/0  
   14 ?           00:00:00 cpuhp/0  
   15 ?           00:00:00 cpuhp/1  
   16 ?           00:00:00 idle_inject/1  
   17 ?           00:00:00 migration/1  
   18 ?           00:00:00 ksoftirqd/1  
   20 ?           00:00:00 kworker/1:0H-kblockd  
   21 ?           00:00:00 kdevtmpfs  
   22 ?           00:00:00 netns  
   23 ?           00:00:00 rcu_tasks_kthre  
   24 ?           00:00:00 kauditd  
   25 ?           00:00:00 khungtaskd  
   26 ?           00:00:00 oom_reaper  
   27 ?           00:00:00 writeback
```

# Linux kill

The most common command to terminate a process is kill command. You need to know the PID of the process you want to terminate.

kill command sends signal to the specified process. For sending signal either signal name or signal number can be used.

kill -SIGNAL PID

Signal Name	Signal Number	Signal Use
SIGNULL	0	NULL, check access to PID
SIGHUP	1	Hangup
SIGINT	2	Interrupt
SIGQUIT	3	Quit
SIGKILL	9	Kill
SIGTERM	15	Terminate
SIGSTOP	24	Stop
SIGTSTP	25	Stop/pause the process
SIGCONT	26	Continue a stopped process

**To list signal names**

To see a list of signal names in your system, following command can be used.

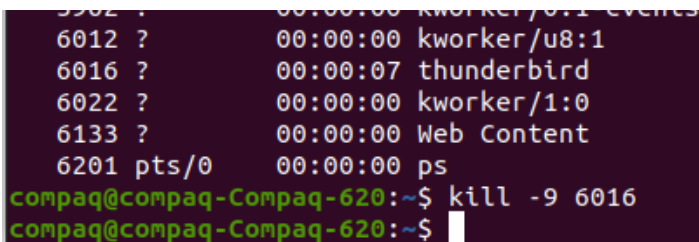
### Syntax:

1. kill -l

### Example:

To kill a process having PID 2408, use following command

kill -9 2408



```
5962 ?        00:00:00 kworker/u8:1 events
6012 ?        00:00:00 kworker/u8:1
6016 ?        00:00:07 thunderbird
6022 ?        00:00:00 kworker/1:0
6133 ?        00:00:00 Web Content
6201 pts/0    00:00:00 ps
compaq@compaq-Compaq-620:~$ kill -9 6016
compaq@compaq-Compaq-620:~$
```

## Linux killall

The killall command need the process name instead of PID. It kills all the processes with the specified name in the system.

### Syntax:

1. killall -<signal name or option> <name>

## Linux pkill

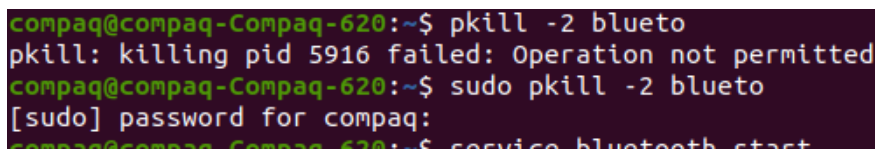
The pkill command uses name of the process instead of PID number. Signal can be send to a process either by typing full name or partial name.

While specifying partial name, the specified name should be within first 15 characters of the process name.

### Example:

pkill -2 sample

With above example, signal will be sent to all the process which has sample in its name.



```
compaq@compaq-Compaq-620:~$ pkill -2 blueto
pkill: killing pid 5916 failed: Operation not permitted
compaq@compaq-Compaq-620:~$ sudo pkill -2 blueto
[sudo] password for compaq:
compaq@compaq-Compaq-620:~$ service bluetooth start
```

## xkill

Command xkill is used to kill a process on X server without passing process name or PID. It forces the X server to close the communication with its clients, which ultimately kill its clients by its X resource. In short, xkill instructs X server to terminate client.



```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ xlsclients  
compaq-Compaq-620 ibus-ui-gtk3  
compaq-Compaq-620 ibus-extension-gtk3  
compaq-Compaq-620 ibus-x11  
compaq-Compaq-620 xdg-desktop-portal-gtk  
compaq-Compaq-620 gnome-shell  
compaq-Compaq-620 gsd-color  
compaq-Compaq-620 gsd-keyboard  
compaq-Compaq-620 gsd-power  
compaq-Compaq-620 gsd-media-keys  
compaq-Compaq-620 gsd-wacom  
compaq-Compaq-620 gsd-xsettings  
compaq-Compaq-620 evolution-alarm-notify  
compaq-Compaq-620 soffice  
compaq-Compaq-620 update-notifier  
compaq-Compaq-620 gnome-terminal-server  
compaq-Compaq-620 firefox  
compaq-Compaq-620 org.gnome.Nautilus  
compaq-Compaq-620 thunderbird  
compaq@compaq-Compaq-620:~$ xkill  
Select the window whose client you wish to kill with button 1....  
xkill: killing creator of resource 0x2a00039
```

## To use xkill command

Using xkill command when you want to kill a process, type xkill on the terminal. Your cursor will change in the shape of **x**, click on the window which you want to kill using x cursor. You'll get the following message as shown in below snapshot.

## pmap

The command pmap reports memory map of one process or multiple processes. It displays information about memory usage and address space of a process. To check pmap of a process we need PID of the process.

### Syntax:

1. pmap PID

### Example:

```
pmap 2390
```

## Memory map of multiple processes

Memory map of multiple processes can also be seen with the same command.

### Syntax:

1. pmap PID1 PID2 PID3 . . .

### Example:



pmap 1682 2390

# ps Command in Linux/Unix with Examples

The ps command is used to view currently running processes on the system. It helps us to determine which process is doing what in our system, how much memory it is using, how much CPU space it occupies, user ID, command name, etc .

## Introduction to ps Command

The ps command shows details of a selection of the running processes. If we wish repetitive selection updates and displayed information, we can use the top command rather.

This version of the ps command accepts many types of options, which are mentioned below:

ps -ef

```
compaq@compaq-Compaq-620:~$ ps -ef
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	11:15	?	00:00:08	/sbin/init splash
root	2	0	0	11:15	?	00:00:00	[kthreadd]
root	3	2	0	11:15	?	00:00:00	[rcu_gp]
root	4	2	0	11:15	?	00:00:00	[rcu_par_gp]
root	8	2	0	11:15	?	00:00:00	[mm_percpu_wq]
root	9	2	0	11:15	?	00:00:00	[ksoftirqd/0]
root	10	2	0	11:15	?	00:00:06	[rcu_sched]
root	11	2	0	11:15	?	00:00:00	[migration/0]
root	12	2	0	11:15	?	00:00:00	[idle_inject/0]
root	14	2	0	11:15	?	00:00:00	[cpuhp/0]
root	15	2	0	11:15	?	00:00:00	[cpuhp/1]
root	16	2	0	11:15	?	00:00:00	[idle_inject/1]
root	17	2	0	11:15	?	00:00:00	[migration/1]
root	18	2	0	11:15	?	00:00:00	[ksoftirqd/1]
root	20	2	0	11:15	?	00:00:00	[kworker/1:0H-kblockd]
root	21	2	0	11:15	?	00:00:00	[kdevtmpfs]
root	22	2	0	11:15	?	00:00:00	[netns]

## free command

The free command gives information about used and unused memory usage and swap memory of a system. By default, it displays memory in **kb** (kilobytes).

Memory mainly consists of RAM (random access memory) and swap memory.

Swap memory is a part of hard disk drive that acts like a virtual RAM.

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ free  
              total        used        free      shared  buff/cache   available  
Mem:        5982232      2076068      1747764      430552     2158400     3206584  
Swap:        651484           0        651484  
compaq@compaq-Compaq-620:~$
```

**Line1**tells about the memory details like total RAM available in our system, used RAM, free RAM, shared RAM, buffered RAM and cached RAM.

**Line2**indicates used and free buffer/cache memory.

**Line3**indicates total, used and free swap memory.

## Top Command in Linux/Unix with Examples

The top command displays all the running process within the environment of your system. It helps in monitoring system usage and performances. It is mainly used to detect load on the server by system administrators.

The top command stands for table of processes. It is a task manager program, detected in several Unix-like operating systems, that shows information about memory and CPU utilization.

### ftp and sftp

The ftp stands for **F**ile **T**ransfer **P**rotocol. It connect to the remote host to exchange files and directories from one host to another over a network which can be LAN or any other.

The sftp stands for **S**ecure **ftp**.

### ftp prompt

The ftp prompt can be used to perform different ftp functions with ftp commands.

## free command

The free command gives information about used and unused memory usage and swap memory of a system. By default, it displays memory in **kb** (kilobytes).

Memory mainly consists of RAM (random access memory) and swap memory.

Swap memory is a part of hard disk drive that acts like a virtual RAM.

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ free  
              total        used        free      shared  buff/cache   available  
Mem:          5982236      1824136      1192972      358652     2965128     3513980  
Swap:          651484           0        651484  
compaq@compaq-Compaq-620:~$
```

## shutdown

The shutdown command brings down system in a secure way. All the logged-in users are notified about the system shutdown.

Signal SIGTERM notifies all the processes that the system is going down, so that processes can be saved and exit properly.

## info

Command info display information in the document format. It is similar to man command with more robustness for linking pages together.

Info pages are made with texinfo tools, can link with other pages and create menus.

The info document's default location is **/usr/share/info**.

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ info pwd  
compaq@compaq-Compaq-620:~$
```

```
compaq@compaq-Compaq-620: ~  
Next: stty invocation, Up: Working context  
19.1 'pwd': Print working directory  
=====
```

'pwd' prints the name of the current directory. Synopsis:

```
pwd [OPTION]...
```

The program accepts the following options. Also see [\\*note Common options::](#).

```
'-L'  
'--logical'  
    If the contents of the environment variable 'PWD' provide an  
    absolute name of the current directory with no '.' or '..'  
    components, but possibly with symbolic links, then output those  
    contents. Otherwise, fall back to default '-P' handling.  
'-P'  
'--physical'
```

```
-----Info: (coreutils)pwd invocation, 37 lines --Top-----  
Welcome to Info version 6.7. Type H for help, h for tutorial.
```

# env

The env command is a shell command used to display and manipulate environment variables. It is used to either list down environment variables or run a program in a modified environment.

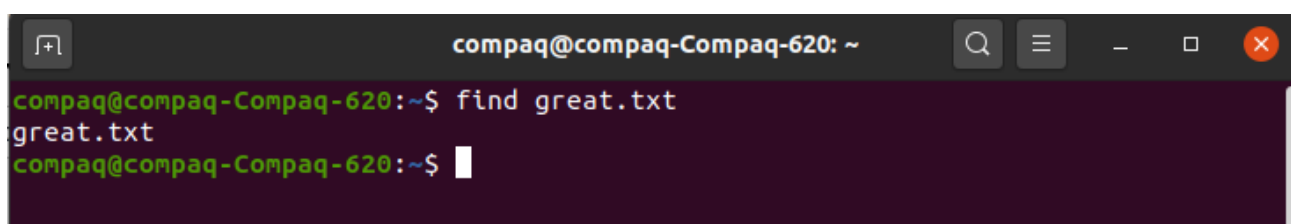
With the help of env, commands can be added or removed, you can assign new values to the existing variables.

A terminal window titled 'compaq@compaq-Compaq-620: ~' with search, menu, and window control icons. The terminal shows the command 'env' and its output, which lists various environment variables and their values.

```
compaq@compaq-Compaq-620:~$ env
SHELL=/bin/bash
SESSION_MANAGER=local/compaq-Compaq-620:~/tmp/.ICE-unix/1660,unix/compaq-Compaq-620:/tmp/.ICE-unix/1660
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GTK_IM_MODULE=ibus
LANGUAGE=en_IN:en
QT4_IM_MODULE=ibus
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
SSH_AGENT_PID=1571
GTK_MODULES=gail:atk-bridge
PWD=/home/compaq
LOGNAME=compaq
```

# Find Command

The find command helps us to find a particular file within a directory.

A terminal window titled 'compaq@compaq-Compaq-620: ~' with search, menu, and window control icons. The terminal shows the command 'find great.txt' and its output, which is 'great.txt'.

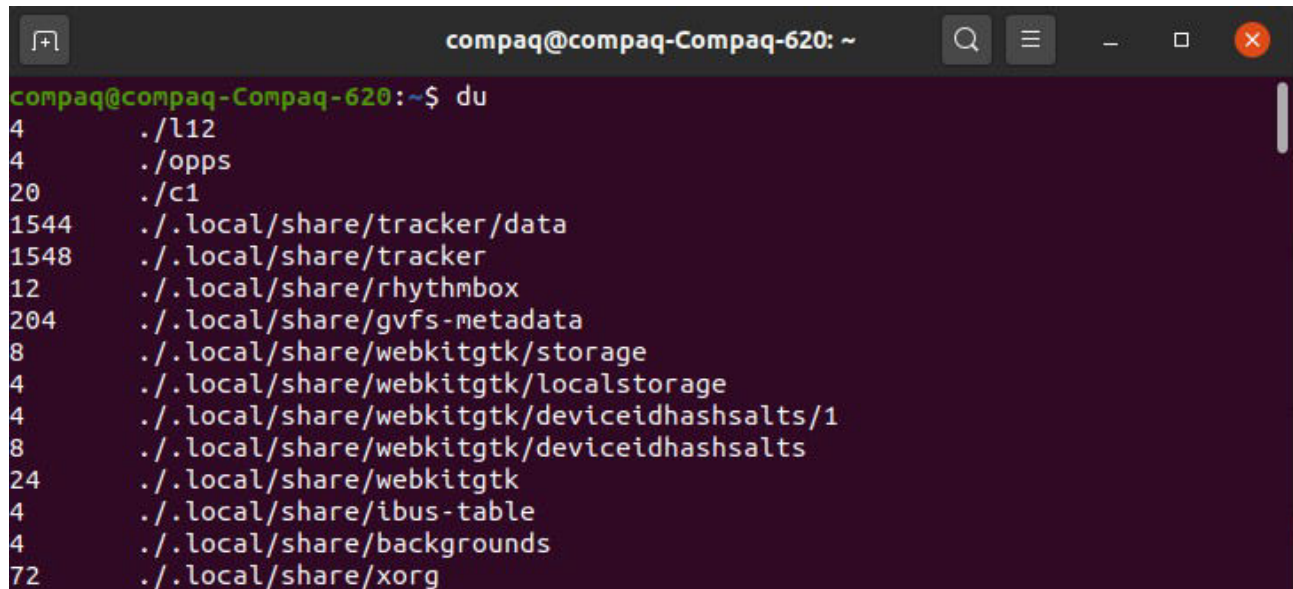
```
compaq@compaq-Compaq-620:~$ find great.txt
great.txt
compaq@compaq-Compaq-620:~$
```

# du Command

Command du stands for **D**isk **U**usage. It is used to check the information of disk usage of files and directories on a system.

Command `du` display a list of all the files along with their respective sizes. By default, size given is in kilobytes.

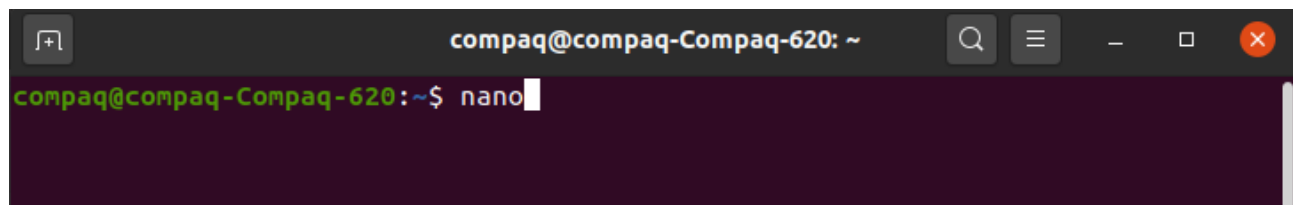
File names are used as arguments to get the file size.

A terminal window titled 'compaq@compaq-Compaq-620: ~' with standard window controls. The command 'du' has been executed, displaying a list of directory sizes in kilobytes. The output is as follows:

```
compaq@compaq-Compaq-620:~$ du
4      ./l12
4      ./opps
20     ./c1
1544   ././local/share/tracker/data
1548   ././local/share/tracker
12     ././local/share/rhythmbox
204    ././local/share/gvfs-metadata
8      ././local/share/webkitgtk/storage
4      ././local/share/webkitgtk/localstorage
4      ././local/share/webkitgtk/deviceidhashsalts/1
8      ././local/share/webkitgtk/deviceidhashsalts
24     ././local/share/webkitgtk
4      ././local/share/ibus-table
4      ././local/share/backgrounds
72     ././local/share/xorg
```

## Nano editor

GNU nano is a friendly and convenient text editor like vi and emacs. It offers many other extra features like word searching, replacing, jump to a line or column, filename tab completion, auto-indentation, etc.

A terminal window titled 'compaq@compaq-Compaq-620: ~' with standard window controls. The command 'nano' has been entered at the prompt, and the cursor is positioned at the end of the command.

```
compaq@compaq-Compaq-620:~$ nano
```

## shred

The `shred` command allows us to delete our files securely making it very much difficult to recover that file by anyone.

Erasing a file with `rm` command only erases the file system entry and keeps the content of the file intact. It is quite easy to recover removed files content using some softwares.

To prevent from data recovering, `shred` **overwrite** the data multiple times by doing maximum destruction of the data.

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ shred fileName
```

## Linux mount

The mount command attaches the filesystem of an external device to the filesystem of a system.

It instructs the operating system that filesystem is ready to use and associate it with a particular point in the system's hierarchy. Mounting will make files, directories and devices available to the users.

It mounts the external storage devices like hard disks, pen drives, USBs etc.

Conversely, **umount** command unmount the mount point and detach the device from the system.

```
compaq@compaq-Compaq-620:~$ mount /dev/sdb1 /mnt  
mount: only root can do that  
compaq@compaq-Compaq-620:~$ sudo mount /dev/sdb1 /mnt  
compaq@compaq-Compaq-620:~$ cd /mnt  
compaq@compaq-Compaq-620:/mnt$ mkdir preeti  
compaq@compaq-Compaq-620:/mnt$ ls -l  
total 33056  
-rw-r--r-- 1 compaq compaq 10563 Mar 20 20:17 'Hardware Specification in MPL  
Jan June 2022.xlsx'  
-rw-r--r-- 1 compaq compaq 11380 Feb 27 19:14 lab_schedule.docx  
drwxr-xr-x 2 compaq compaq 32768 May 7 19:31 preeti  
-rw-r--r-- 1 compaq compaq 33218362 Sep 4 2019 SanDiskMemoryZone_AppInstalle  
r.apk  
-rw-r--r-- 1 compaq compaq 497832 Sep 4 2019 SanDiskMemoryZone_QuickStartG  
uide.pdf  
compaq@compaq-Compaq-620:/mnt$ sudo umount /dev/sdb1  
sudo: umount: command not found  
compaq@compaq-Compaq-620:/mnt$ sudo umount /dev/sdb1  
umount: /mnt: target is busy.  
compaq@compaq-Compaq-620:/mnt$ cd ..  
compaq@compaq-Compaq-620:/$ sudo umount /dev/sdb1  
compaq@compaq-Compaq-620:/$
```

## traceroute

Linux traceroute command is a network troubleshooting utility that helps us determine the number of hops and packets traveling path required to reach a destination. It is used to display how the data transmitted from a local machine to a remote machine. The traceroute can display the routes, [IP](#) addresses, and hostnames of routers over a network. It can be useful for diagnosing network issues.

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ traceroute google.com  
traceroute to google.com (142.250.66.14), 64 hops max  
 1  192.168.0.1  1.045ms  0.741ms  0.667ms  
 2  100.95.0.1  7.512ms  5.593ms  16.841ms  
 3  113.193.214.101  7.418ms  8.780ms  7.797ms  
 4  122.187.36.73  8.982ms  4.163ms  11.858ms  
 5  116.119.106.206  23.912ms  25.602ms  21.605ms  
 6  72.14.212.48  27.316ms  20.523ms  23.980ms  
 7  * * *  
 8  108.170.248.177  21.738ms  22.553ms  20.173ms  
 9  72.14.236.219  16.917ms  18.349ms  20.558ms  
10  108.170.248.209  19.568ms  19.519ms  22.221ms  
11  142.250.66.14  21.986ms  19.172ms  19.791ms  
compaq@compaq-Compaq-620:~$
```

## Linux sudo

The Linux sudo command stands for **Super User Do**. Generally, it is applied as a prefix of a few commands that superuser is allowed to execute.

If we prefix the command along with other commands, it would execute that command with high privileges. In other words, it will permit user along with proper authorization eating a command as other users like the superuser.

It is equal to the option "**run as administrator**" in Windows.

```
compaq@compaq-Compaq-620:~$ sudo  
usage: sudo -h | -K | -k | -V  
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]  
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]  
[command]  
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p  
prompt] [-T timeout] [-u user] [VAR=value] [-i|-s] [<command>]  
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p  
prompt] [-T timeout] [-u user] file ...  
compaq@compaq-Compaq-620:~$ sudo -V  
Sudo version 1.8.31  
Sudoers policy plugin version 1.8.31  
Sudoers file grammar version 46  
Sudoers I/O plugin version 1.8.31  
compaq@compaq-Compaq-620:~$
```

## Tar Command

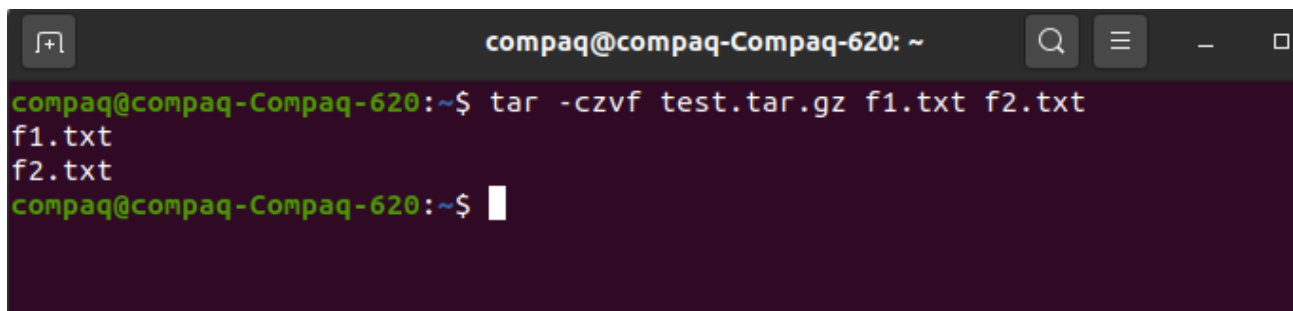
It is short for Tape Archive and is used to create and extract archive files. An archive file is a compressed file that contains one or more files bundled together for more accessible storage and portability.



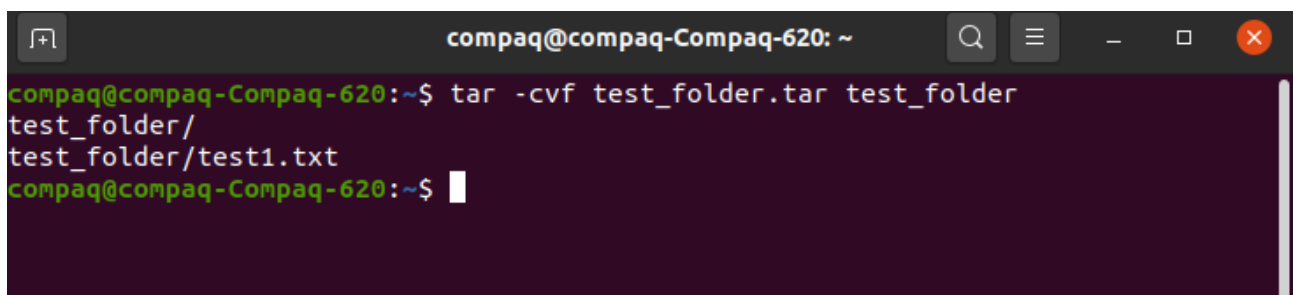
## tar command options

The **tar** command provides the following options –

- -c – This creates an archive file.
- -x – The option extracts the archive file.
- -f – Specifies the filename of the archive file.
- -v – This prints verbose information for any tar operation on the terminal.
- -t – This lists all the files inside an archive file.
- -u – This archives a file and then adds it to an existing archive file.
- -r – This updates a file or directory located inside a .tar file
- -z – Creates a tar file using gzip compression
- -j – Create an archive file using the bzip2 compression
- -W – The -w option verifies an archive file.



```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ tar -czvf test.tar.gz f1.txt f2.txt  
f1.txt  
f2.txt  
compaq@compaq-Compaq-620:~$
```

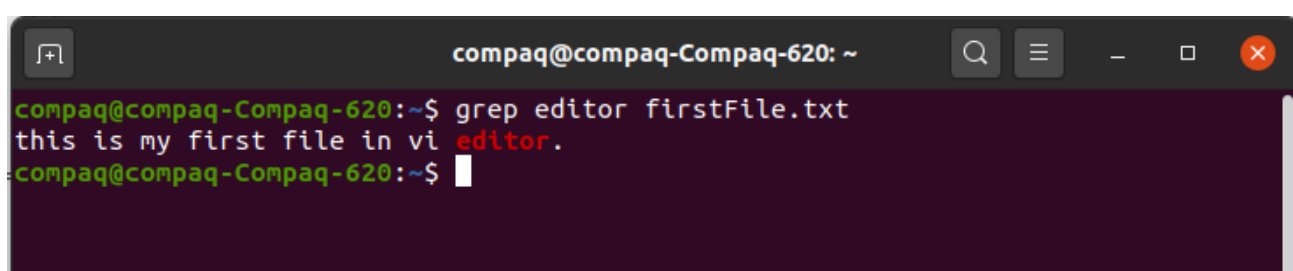


```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ tar -cvf test_folder.tar test_folder  
test_folder/  
test_folder/test1.txt  
compaq@compaq-Compaq-620:~$
```

## Grep Command

The 'grep' command stands for "**global regular expression print**". grep command filters the content of a file which makes our search easy.

It is a command-line utility to search plain-text data groups for lines that are the same as a regular expression.



```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ grep editor firstFile.txt  
this is my first file in vi editor.  
compaq@compaq-Compaq-620:~$
```

# AWK Command

The **awk command** is used for **text processing** in Linux. Although, the sed command is also used for text processing, but it has some limitations, so the awk command becomes a handy option for text processing. It provides powerful control to the data.

The Awk is a powerful scripting language used for **text scripting**. It searches and replaces the texts and sorts, validates, and indexes the database.

```
compaq@compaq-Compaq-620:~$ cat record.txt
Karan CS
Ashish CS
Roshni IT
Sandeep IT
Kirti ECE
Rahul ECE
compaq@compaq-Compaq-620:~$ awk '/CS/{print}' record.txt
Karan CS
Ashish CS
compaq@compaq-Compaq-620:~$
```

```
compaq@compaq-Compaq-620: ~
compaq@compaq-Compaq-620:~$ cat record.txt
Karan CS
Ashish CS
Roshni IT
Sandeep IT
Kirti ECE
Rahul ECE
compaq@compaq-Compaq-620:~$ awk '{print $1}' record.txt
Karan
Ashish
Roshni
Sandeep
Kirti
Rahul
compaq@compaq-Compaq-620:~$
```

# Linux aliases

Linux 'alias' command replaces one string from the shell with another string. It is a shell built-in command. It converts a complicated command into a simpler command or in other words, it creates a shortcut by replacing it with the simpler one.

```
compaq@compaq-Compaq-620: ~
compaq@compaq-Compaq-620:~$ alias ll='ls -l'
compaq@compaq-Compaq-620:~$ ll
total 976
-rwxrwxr-x 1 compaq compaq 379 Apr 28 15:21 add.sh
-rw-rw-r-- 1 compaq compaq 2 Apr 12 09:56 A.sh
-rw-rw-r-- 1 compaq compaq 0 Apr 26 09:45 a.ts
-rw-rw-r-- 1 compaq compaq 0 Apr 27 09:38 A.ts
-rw-rw-r-- 1 compaq compaq 9 Apr 23 06:37 bar.txt
drwxrwxr-x 2 compaq compaq 4096 Apr 26 10:58 bb
drwxrwxr-x 2 compaq compaq 4096 Apr 21 13:54 c1
-rw-r--r-- 1 compaq compaq 19362 Apr 9 06:46 Class_1.odt
-rwxrwxr-x 1 compaq compaq 41 Apr 30 22:32 cmd.sh
-rw-r--r-- 1 compaq compaq 87 Apr 9 08:15 createFile.txt
-rwxrwxr-x 1 compaq compaq 16 Apr 23 20:15 dddd.sh
drwxr-xr-x 2 compaq compaq 4096 May 7 19:54 Desktop
drwxr-xr-x 2 compaq compaq 4096 May 7 20:57 Documents
```

```
compaq@compaq-Compaq-620: ~  
compaq@compaq-Compaq-620:~$ alias p='pwd'  
compaq@compaq-Compaq-620:~$ p  
/home/compaq  
compaq@compaq-Compaq-620:~$ pwd  
/home/compaq  
compaq@compaq-Compaq-620:~$
```

## Linux route

The route command displays and manipulate IP routing table for your system.

A router is a device which is basically used to determine the best way to route packets to a destination.

```
compaq@compaq-Compaq-620: /  
compaq@compaq-Compaq-620:/$ route  
Kernel IP routing table  
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface  
default          _gateway        0.0.0.0          UG    600    0      0 wls1  
link-local       0.0.0.0         255.255.0.0      U     1000   0      0 wls1  
192.168.0.0      0.0.0.0         255.255.255.0    U     600    0      0 wls1  
compaq@compaq-Compaq-620:/$
```

```
compaq@compaq-Compaq-620:~$ service --status-all
```

<https://file-examples.com/storage/fe0d875dfd645260e96b346/2017/10/>

file\_example\_PNG\_500kB.png