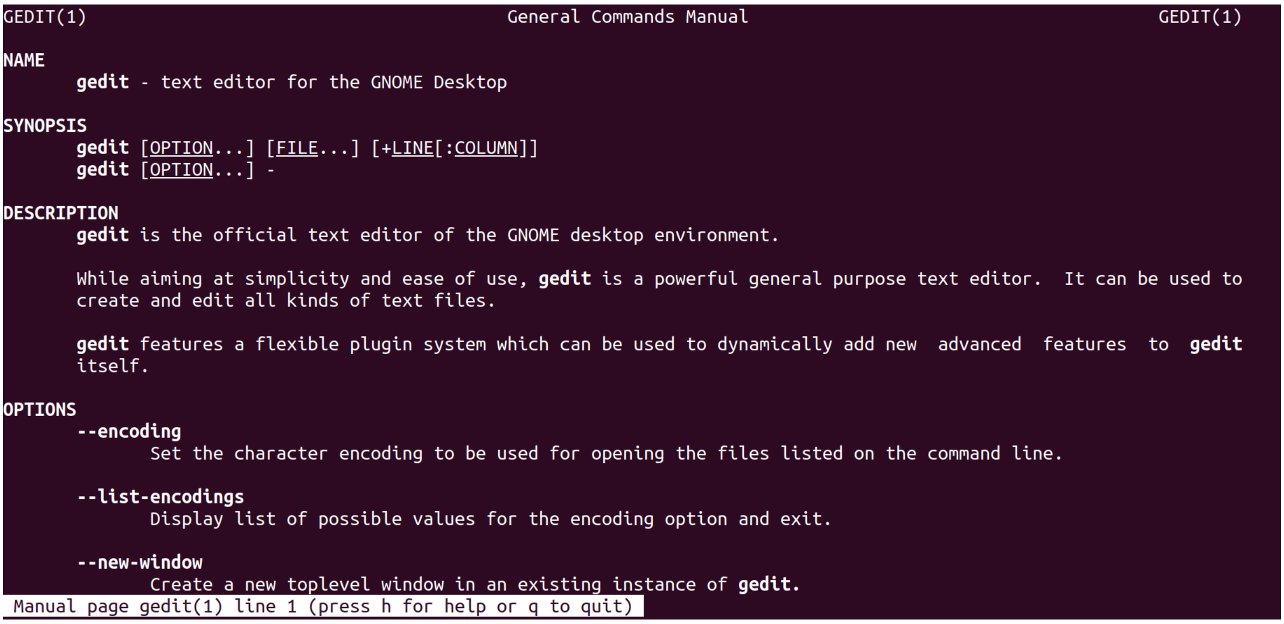
Basic Commands on LINUX Operating System

1. Getting help in Unix

- man – view manual pages for Unix commands

Syntax: man command_name

Ex: man gedit



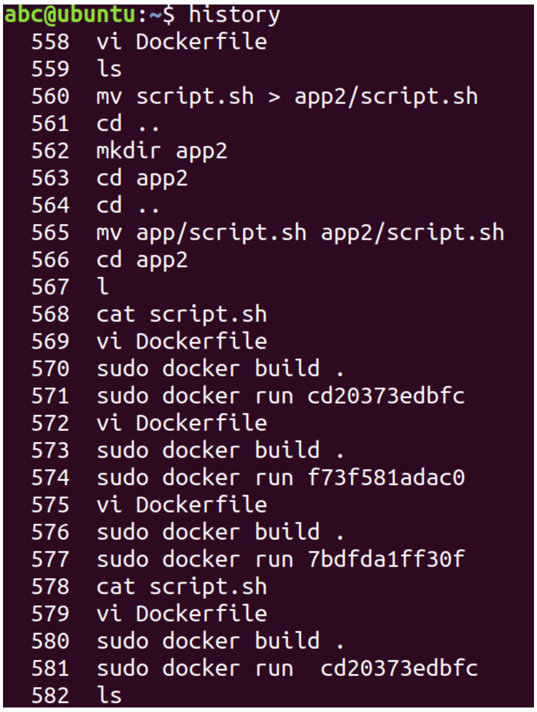
```
GEDIT(1)                         General Commands Manual                         GEDIT(1)

NAME
       gedit - text editor for the GNOME Desktop

SYNOPSIS
       gedit [OPTION...] [FILE...] [+LINE[:COLUMN]]
       gedit [OPTION...] -

DESCRIPTION
       gedit is the official text editor of the GNOME desktop environment.

       While aiming at simplicity and ease of use, gedit is a powerful general purpose text editor.  It can be used to
       create and edit all kinds of text files.

       gedit features a flexible plugin system which can be used to dynamically add new  advanced  features  to  gedit
       itself.

OPTIONS
       --encoding
              Set the character encoding to be used for opening the files listed on the command line.

       --list-encodings
              Display list of possible values for the encoding option and exit.

       --new-window
              Create a new toplevel window in an existing instance of gedit.
 Manual page gedit(1) line 1 (press h for help or q to quit)
```

2. Unix Shell Commands

- clear – clear screen

Syntax: clear or ctrl+l

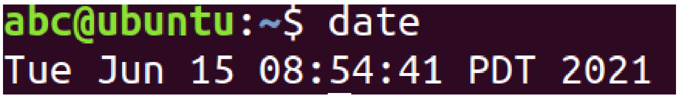
- history – show history of previous commands



```
abc@ubuntu:~$ history
  558  vi Dockerfile
  559  ls
  560  mv script.sh > app2/script.sh
  561  cd ..
  562  mkdir app2
  563  cd app2
  564  cd ..
  565  mv app/script.sh app2/script.sh
  566  cd app2
  567  l
  568  cat script.sh
  569  vi Dockerfile
  570  sudo docker build .
  571  sudo docker run cd20373edbfc
  572  vi Dockerfile
  573  sudo docker build .
  574  sudo docker run f73f581adac0
  575  vi Dockerfile
  576  sudo docker build .
  577  sudo docker run 7bdfda1ff30f
  578  cat script.sh
  579  vi Dockerfile
  580  sudo docker build .
  581  sudo docker run  cd20373edbfc
  582  ls
```

3. Time and Date commands

- date – show current date and time

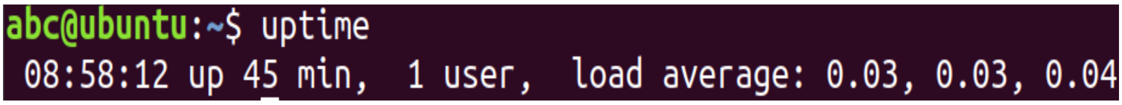


```
abc@ubuntu:~$ date
Tue Jun 15 08:54:41 PDT 2021
```

- sleep – wait for a given number of seconds

  Syntax: sleep 2 (wait for 2 seconds)

- uptime – find out how long the system has been up

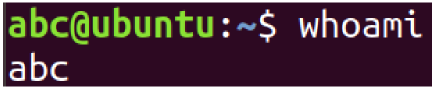


```
abc@ubuntu:~$ uptime
 08:58:12 up 45 min,  1 user,  load average: 0.03, 0.03, 0.04
```

4. Unix users commands

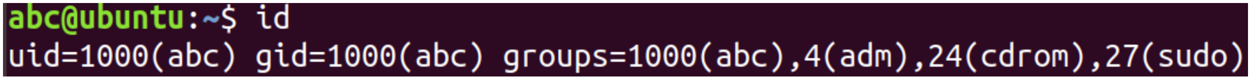
These commands allow you to get basic information about Unix users in your environment

- whoami – show your username



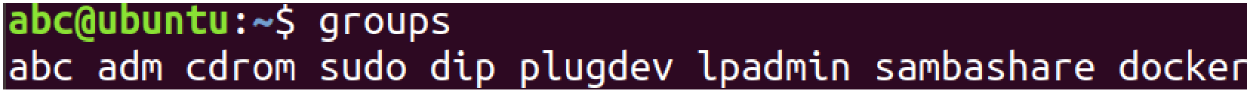
```
abc@ubuntu:~$ whoami
abc
```

- id – print user identity



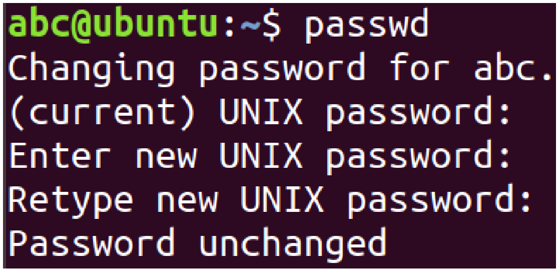
```
abc@ubuntu:~$ id
uid=1000(abc) gid=1000(abc) groups=1000(abc),4(adm),24(cdrom),27(sudo)
```

- groups – show which groups user belongs to



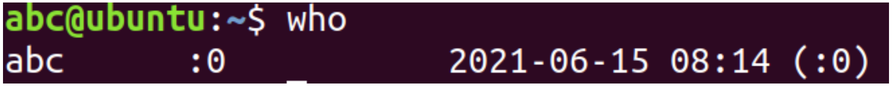
```
abc@ubuntu:~$ groups
abc adm cdrom sudo dip plugdev lpadmin sambashare docker
```

- passwd – change user password



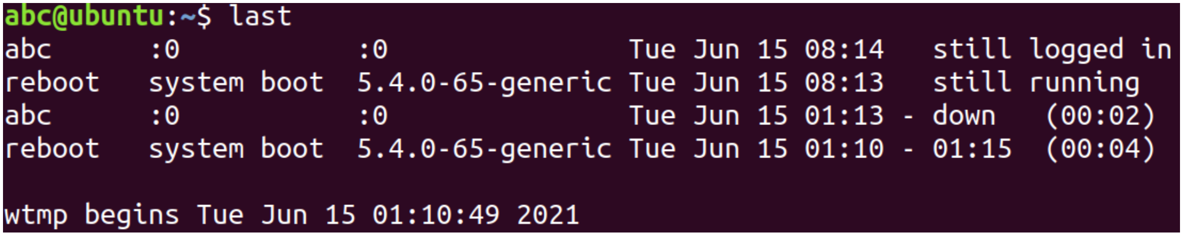
```
abc@ubuntu:~$ passwd
Changing password for abc.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
Password unchanged
```

- who – find out who is logged into the system



```
abc@ubuntu:~$ who
abc      :0           2021-06-15 08:14 (:0)
```

- last – show history of logins into the system



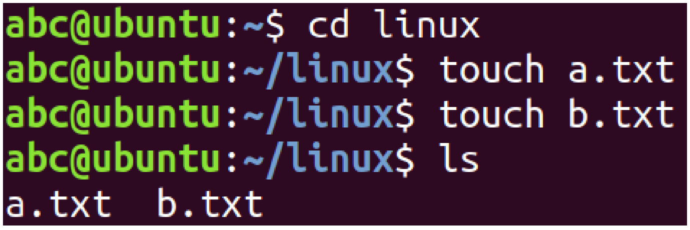
```
abc@ubuntu:~$ last
abc      :0           :0               Tue Jun 15 08:14   still logged in
reboot   system boot  5.4.0-65-generic Tue Jun 15 08:13   still running
abc      :0           :0               Tue Jun 15 01:13 - down   (00:02)
reboot   system boot  5.4.0-65-generic Tue Jun 15 01:10 - 01:15  (00:04)

wtmp begins Tue Jun 15 01:10:49 2021
```

5. Unix file operations

Navigating filesystem and managing files and access permissions:

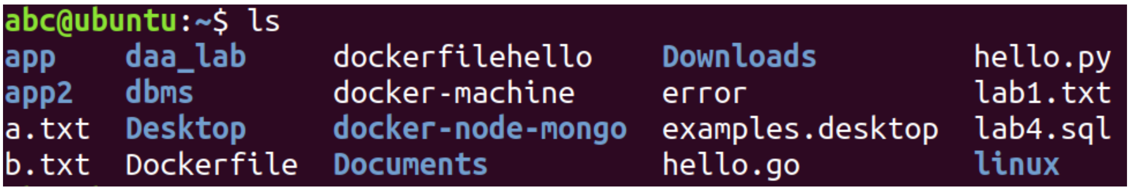
- touch-Used to create, change and modify a file.

  Syntax: touch a.txt (a text file is created in present working directory)



```
abc@ubuntu:~$ cd linux
abc@ubuntu:~/linux$ touch a.txt
abc@ubuntu:~/linux$ touch b.txt
abc@ubuntu:~/linux$ ls
a.txt  b.txt
```

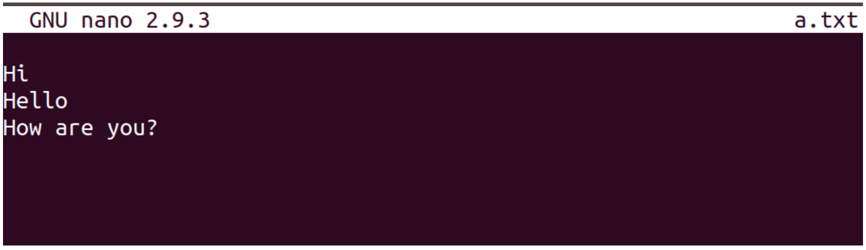
- ls – list files and directories



```
abc@ubuntu:~$ ls
app    daa_lab     dockerfilehello    Downloads         hello.py
app2   dbms        docker-machine     error             lab1.txt
a.txt  Desktop     docker-node-mongo  examples.desktop  lab4.sql
b.txt  Dockerfile  Documents          hello.go          linux
```

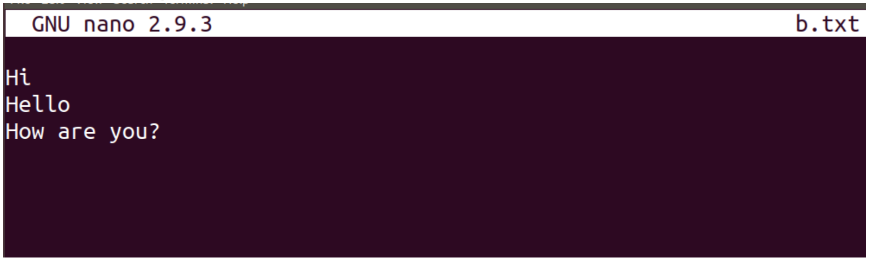
- cp – copy files (work in progress)

  Syntax: cp a.txt b.txt (Here the content of a.txt is being copied to b.txt)



```
  GNU nano 2.9.3                                                    a.txt

Hi
Hello
How are you?
```

After using the above command



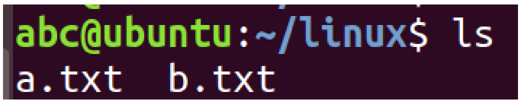
```
  GNU nano 2.9.3                                                    b.txt

Hi
Hello
How are you?
```

- rm – remove files and directories (work in progress)

  Syntax: rm file_name/direcory

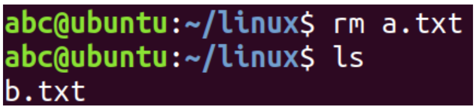
  Ex: rm a.txt

Before



```
abc@ubuntu:~/linux$ ls
a.txt  b.txt
```

After



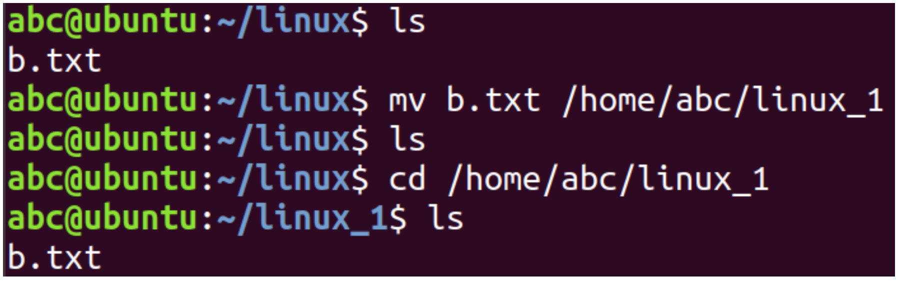
```
abc@ubuntu:~/linux$ rm a.txt
abc@ubuntu:~/linux$ ls
b.txt
```

- mv – rename or move files and directories to another location

  mv used to move a file from one location to another

  Syntax: mv file_name to_location

  Ex: mv b.txt /home/abc/linux_1 (Here b.txt is moved from linux to linux_1)

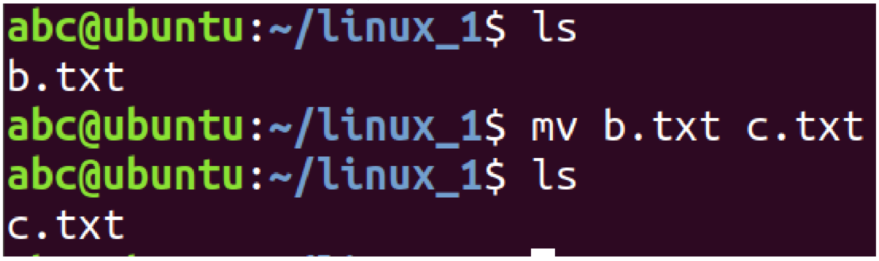


```
abc@ubuntu:~/linux$ ls
b.txt
abc@ubuntu:~/linux$ mv b.txt /home/abc/linux_1
abc@ubuntu:~/linux$ ls
abc@ubuntu:~/linux$ cd /home/abc/linux_1
abc@ubuntu:~/linux_1$ ls
b.txt
```

mv used to rename a existing file

Syntax: mv file_name new_name

Ex: mv b.txt c.txt (Here b.txt is renamed as c.txt)



```
abc@ubuntu:~/linux_1$ ls
b.txt
abc@ubuntu:~/linux_1$ mv b.txt c.txt
abc@ubuntu:~/linux_1$ ls
c.txt
```

- chmod – change file/directory access permissions

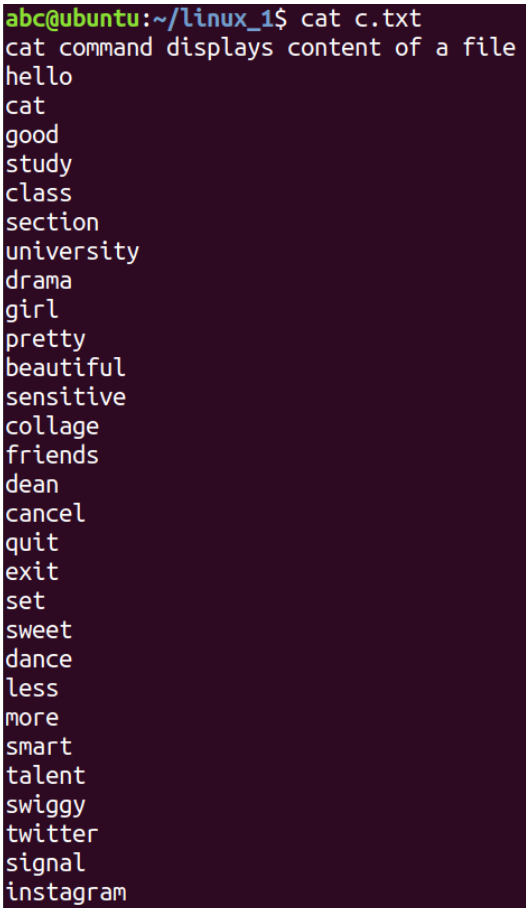
6. Text file operations in Unix

Most of important configuration in Unix is in clear text files, these commands will let you quickly inspect files or view logs:

- cat – concatenate files and show contents to the standard output

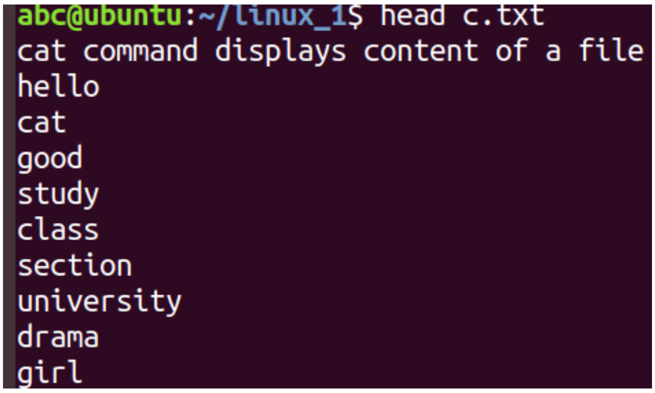
Syntax: cat file_name

Ex: cat c.txt (display content of c.txt)



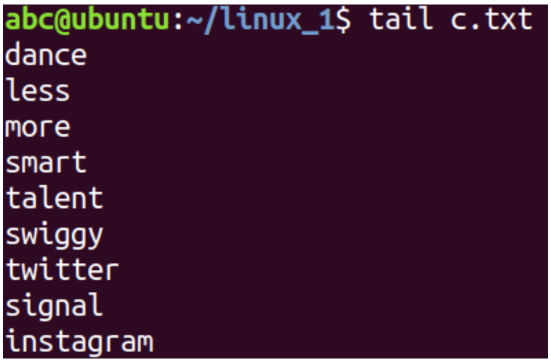
```
abc@ubuntu:~/linux_1$ cat c.txt
cat command displays content of a file
hello
cat
good
study
class
section
university
drama
girl
pretty
beautiful
sensitive
collage
friends
dean
cancel
quit
exit
set
sweet
dance
less
more
smart
talent
swiggy
twitter
signal
instagram
```

- more – basic pagination when viewing text files or parsing Unix commands output
- less – an improved pagination tool for viewing text files(better than more command)
- head – show the first 10 lines of text file (you can specify any number of lines)



```
abc@ubuntu:~/linux_1$ head c.txt
cat command displays content of a file
hello
cat
good
study
class
section
university
drama
girl
```

- tail – show the last 10 lines of text file (any number can be specified)

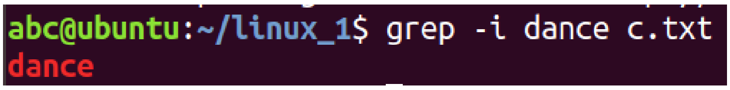


```
abc@ubuntu:~/linux_1$ tail c.txt
dance
less
more
smart
talent
swiggy
twitter
signal
instagram
```

- grep – search for patterns in text files

  Syntax- grep [option] pattern file_name

  Ex- grep -I dance c.txt (displays dance if it exits in c.txt)

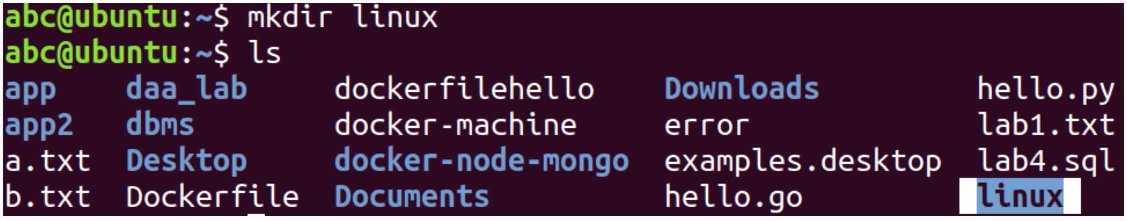


```
abc@ubuntu:~/linux_1$ grep -i dance c.txt
dance
```

7. **Unix directory management commands**

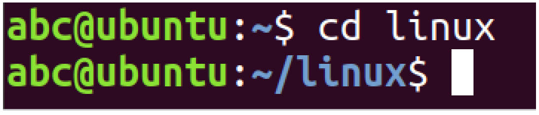
   Navigating filesystems and managing directories:

- mkdir-used to create new directory



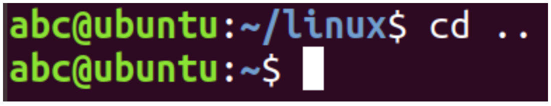
```
abc@ubuntu:~$ mkdir linux
abc@ubuntu:~$ ls
app     daa_lab      dockerfilehello    Downloads          hello.py
app2    dbms         docker-machine     error              lab1.txt
a.txt   Desktop      docker-node-mongo  examples.desktop   lab4.sql
b.txt   Dockerfile   Documents          hello.go           linux
```

- cd- change directory



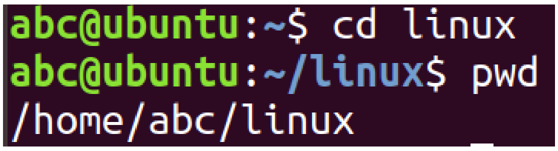
```
abc@ubuntu:~$ cd linux
abc@ubuntu:~/linux$
```

- cd .. – used to come out from the present working directory



```
abc@ubuntu:~/linux$ cd ..
abc@ubuntu:~$
```

- pwd- shows present working directory.



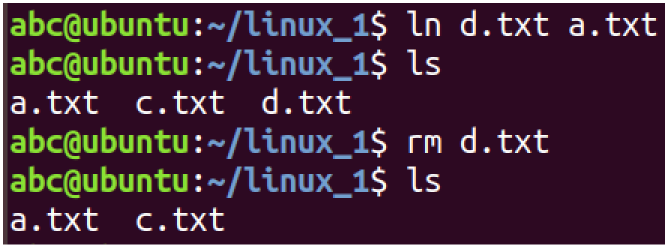
```
abc@ubuntu:~$ cd linux
abc@ubuntu:~/linux$ pwd
/home/abc/linux
```

- ln – make links and symlinks to files and directories
   - <u>Hard link</u>

     Syntax: ln [original filename] [link filename]

     Ex: ln d.txt a.txt

     In the above example hard link is created between d.txt (existing file) and a.txt (link file).If you remove d.txt, a.txt can be still accessed.



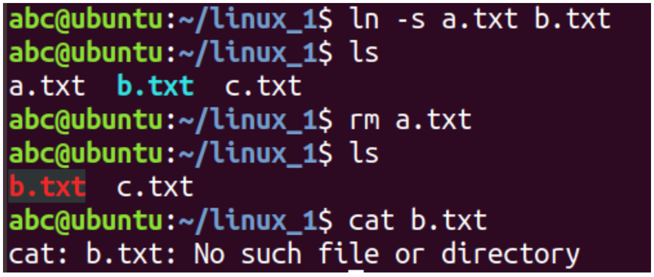
```
abc@ubuntu:~/linux_1$ ln d.txt a.txt
abc@ubuntu:~/linux_1$ ls
a.txt  c.txt  d.txt
abc@ubuntu:~/linux_1$ rm d.txt
abc@ubuntu:~/linux_1$ ls
a.txt  c.txt
```

   - <u>Soft link</u>

     Syntax: ln -s [original filename] [link filename]

     Ex: ln -s a.txt b.txt

     In the above example soft link is created between a.txt (existing file) and b.txt (link file).If you remove a.txt, b.txt cannot be accessed (worthless).

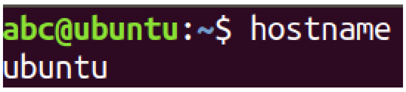


```
abc@ubuntu:~/linux_1$ ln -s a.txt b.txt
abc@ubuntu:~/linux_1$ ls
a.txt  b.txt  c.txt
abc@ubuntu:~/linux_1$ rm a.txt
abc@ubuntu:~/linux_1$ ls
b.txt  c.txt
abc@ubuntu:~/linux_1$ cat b.txt
cat: b.txt: No such file or directory
```

8. Unix system status commands

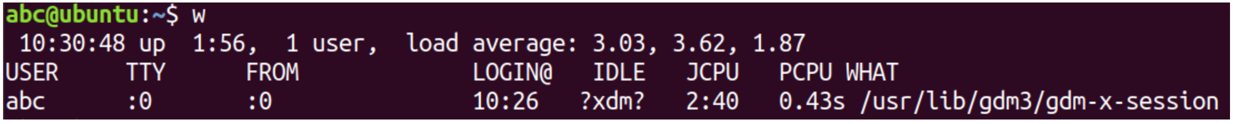
Most useful commands for reviewing hostname configuration and vital stats:

- hostname – show or set server hostname



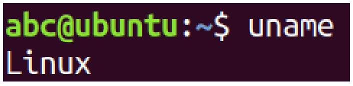
```
abc@ubuntu:~$ hostname
ubuntu
```

- w – display system load, who’s logged in and what they are doing



```
abc@ubuntu:~$ w
 10:30:48 up  1:56,  1 user,  load average: 3.03, 3.62, 1.87
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
abc      :0       :0               10:26   ?xdm?   2:40   0.43s /usr/lib/gdm3/gdm-x-session
```

- uname – print Unix system information



```
abc@ubuntu:~$ uname
Linux
```

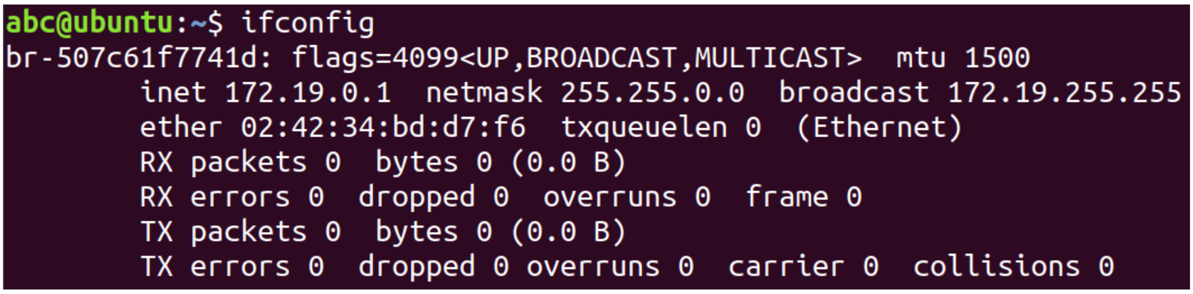
9. Reboot

- shutdown – graceful shutdown and reboot of your system
- reboot – ungraceful reboot (without stopping OS services)

10. Networking commands in Unix

Most useful commands for inspecting network setup and exploring network Connections and ports.

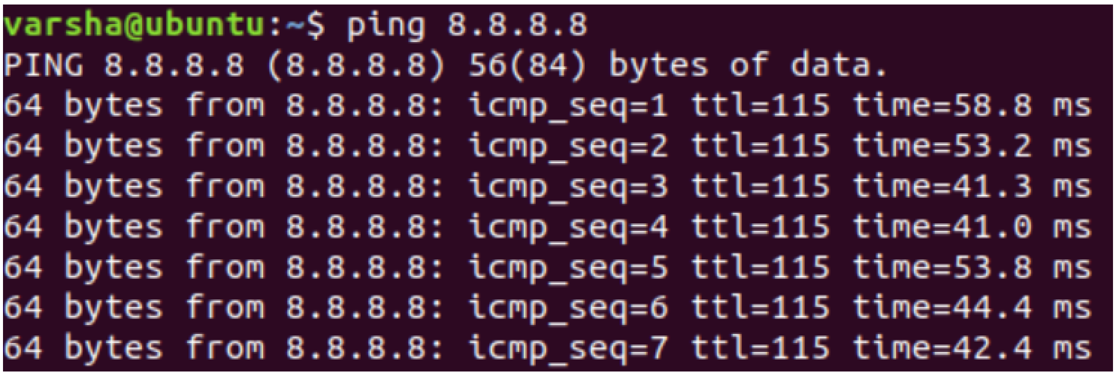
- ifconfig – show and set IP addresses (found almost everywhere)



```
abc@ubuntu:~$ ifconfig
br-507c61f7741d: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.19.0.1  netmask 255.255.0.0  broadcast 172.19.255.255
        ether 02:42:34:bd:d7:f6  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- ping – check if remote host is reachable via ICMP ping

  Its usually used as a simple way to verify that a computer can communicate over the network another network



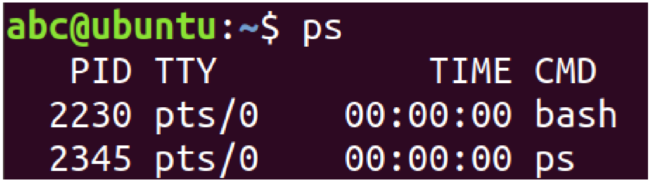
```
varsha@ubuntu:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=58.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=53.2 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=41.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=41.0 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=115 time=53.8 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=115 time=44.4 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=115 time=42.4 ms
```

(DNS server address of Google-8.8.8.8)

11.Process management

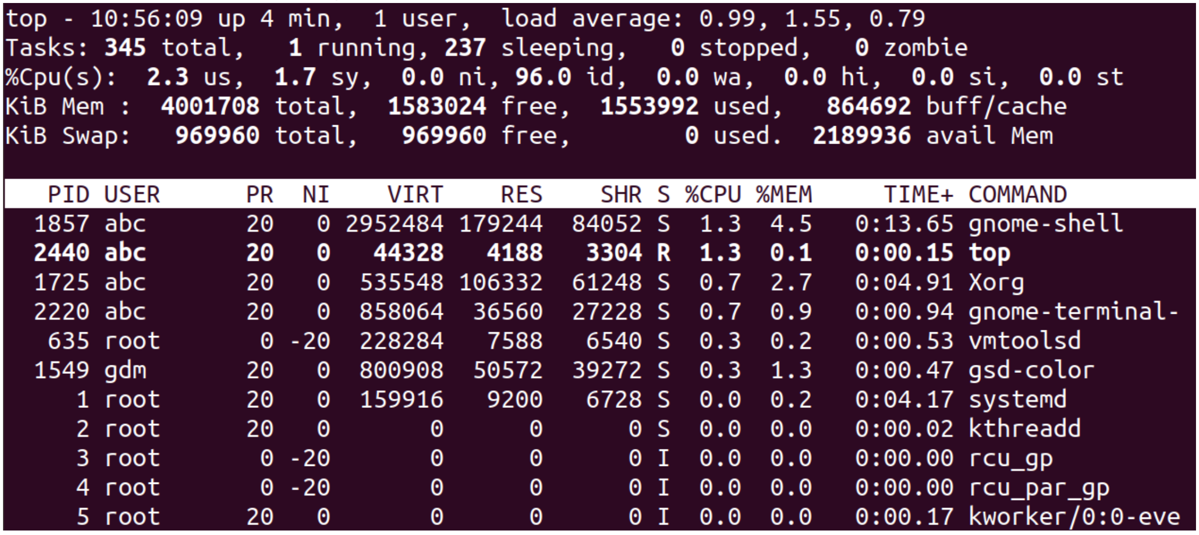
Listing processes and confirming their status, and stopping processes if needed:

- ps – list processes



```
abc@ubuntu:~$ ps
   PID TTY          TIME CMD
  2230 pts/0    00:00:00 bash
  2345 pts/0    00:00:00 ps
```

- top – show tasks and system status



```
top - 10:56:09 up 4 min,  1 user,  load average: 0.99, 1.55, 0.79
Tasks: 345 total,   1 running, 237 sleeping,   0 stopped,   0 zombie
%Cpu(s):  2.3 us,  1.7 sy,  0.0 ni, 96.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  4001708 total,  1583024 free,  1553992 used,   864692 buff/cache
KiB Swap:   969960 total,   969960 free,        0 used.  2189936 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 1857 abc       20   0 2952484 179244  84052 S   1.3  4.5   0:13.65 gnome-shell
 2440 abc       20   0   44328   4188   3304 R   1.3  0.1   0:00.15 top
 1725 abc       20   0  535548 106332  61248 S   0.7  2.7   0:04.91 Xorg
 2220 abc       20   0  858064  36560  27228 S   0.7  0.9   0:00.94 gnome-terminal-
  635 root       0 -20  228284   7588   6540 S   0.3  0.2   0:00.53 vmtoolsd
 1549 gdm       20   0  800908  50572  39272 S   0.3  1.3   0:00.47 gsd-color
    1 root      20   0  159916   9200   6728 S   0.0  0.2   0:04.17 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.02 kthreadd
    3 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 rcu_gp
    4 root       0 -20       0      0      0 I   0.0  0.0   0:00.00 rcu_par_gp
    5 root      20   0       0      0      0 I   0.0  0.0   0:00.17 kworker/0:0-eve
```

- kill – kill a process (stop application running)

  Syntax: kill PID (Process id) or kill -9 PID

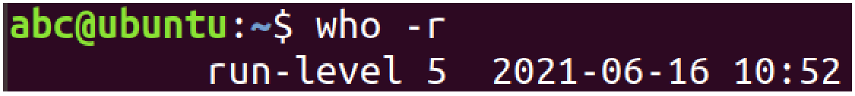
12.Privileged Access

- su – switch user (commonly used to become root)

- sudo – run commands with elevated (usually root-like) privileges o be sure to check out sudo reference

13.Unix system status commands

- who -r – confirm current run-level of your Unix/Linux OS



```
abc@ubuntu:~$ who -r
         run-level 5  2021-06-16 10:52
```