

# Linux

**Linux:** It is an operating system which provides high speed and security. Linux is wide used among other OS. It is a multi-user OS (Multiple users can connected to the same instance).

**Operating System:** It is an interface between users and computer hardware components. It helps computers to run the programs and applications.

**Kernel:** It is used to manager hardware components - CPU, RAM etc..

**Daemon:** The services which runs in the background.

**Shell:** It is like CLI(Command Line Interface) and GUI(Graphic User Interface). It is used to execute the scripts.

Amazon Linux default user is ec2user but it has limited permissions on the commands

root is the user which has all permissions om the commands. It is the ultimate king of Linux.

**Linux has various types of commands:**

- System commands
- Hardware commands
- Network commands
- File commands
- Search commands
- User commands
- Permission commands

## 1. System commands:

By default, AWS Instance will login with ec2user to change to root user the command is

- **sudo - i (or) sudo su -**

```
[ec2-user@ip-172-31-31-58 ~]$ sudo -i
[root@ec2-gopichand ~]#
```

- **uname:** Displays OS name of instance
- **uname -r:** displays kernel version
- **uname -a:** displays all the details

```
[root@ec2-gopichand ~]# uname
Linux
[root@ec2-gopichand ~]# uname -r
5.10.192-183.736.amzn2.x86_64
[root@ec2-gopichand ~]# uname -a
Linux ec2-gopichand 5.10.192-183.736.amzn2.x86_64 #1 SMP Wed Sep 6 21:15:41 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux
```

- **uptime:** to know the run time of an instance, no.of users connected to the instance
- **uptime -p:** displays only time

- `uptime -s`: displays only date and time

```
[root@ec2-gopichand ~]# uptime
11:41:22 up 1:29, 1 user, load average: 0.00, 0.00, 0.00
[root@ec2-gopichand ~]# uptime -p
up 1 hour, 29 minutes
[root@ec2-gopichand ~]# uptime -s
2023-10-09 10:11:48
```

- `hostname`: to get the hostname details

```
[root@gopi-sys ~]# hostname
gopi-sys
```

- `hostname -i`: to get the private ip details

```
[root@gopi-sys ~]# hostname -i
fe80::89a:18ff:fe7a:fbb9%eth0 172.31.31.58
```

- `ip route`, `ip addr`, `ifconfig` - these commands are also used to get the private ip details
- `hostnamectl set-hostname new-host-name` (ex: `hostnamectl set-hostname gopi-sys`) - This command is used to change the hostname

```
[root@gopi-sys ~]# hostnamectl set-hostname ec2-gopichand
[new-username]
[root@gopi-sys ~]# exit
logout
[ec2-user@ip-172-31-31-58 ~]$ sudo -i
[root@ec2-gopichand ~]#
```

**#once after changing the new username do exit and login back to see the new username**

- `date` -to get today's date and time. By default it will be GMT ex: 10/9/23
- `date+"%d"` - to get only date ex: 9
- `date+"%m"` - to get only month ex: 10
- `date+"%y"` - to get only year ex: 23
- `date+"%H"` - to get only hour
- `date+"%M"` - to get minute
- `date+"%S"` - to get seconds
- `date+"%D"` - to get only date in format (MM/DD/YY)
- `date+"%F"` - to get only date in format (YY-MM-DD)
- `date+"%A"` - to get day ex: Monday
- `date+"%B"` - to get month ex: October

```
[root@ec2-gopichand ~]# date
Mon Oct  9 11:48:58 IST 2023
[root@ec2-gopichand ~]# date +"%d"
09
[root@ec2-gopichand ~]# date +"%m"
10
[root@ec2-gopichand ~]# date +"%y"
23
[root@ec2-gopichand ~]# date +"%D"
10/09/23
[root@ec2-gopichand ~]# date +"%F"
2023-10-09
[root@ec2-gopichand ~]# date +"%A"
Monday
[root@ec2-gopichand ~]# date +"%B"
October
```

- To change the default time zone to IST

`timedatectl set-timezone Asia/Kolkata` (#Hyd using Asia/Kolkata timezone, A & K should be caps)

```
[root@ec2-gopichand ~]# timedatectl set-timezone Asia/Kolkata
[root@ec2-gopichand ~]# date
Mon Oct  9 11:51:49 IST 2023
```

## 2. Hardware Commands:

### To get CPU info:

- `cat /proc/cpuinfo` or `lscpu`

Here `/proc` means processor and `/cpuinfo` is the text file.

- `cat` is a Linux command used to read the data in a file.

```
[root@gopichand_k739 ~]# cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 79
model name     : Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz
stepping       : 1
microcode      : 0xb0000040
cpu MHz        : 2300.056
cache size     : 46080 KB
physical id    : 0
siblings       : 1
core id        : 0
cpu cores      : 1
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 13
wp             : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_
good nopl xtopology cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand hyp
ervisor lahf_lm abm cpuid_fault invpcid_single pti fsgsbase bmi1 avx2 smep bmi2 erms invpcid xsaveopt
bugs           : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs itlb_multihit mmio_stale_data
bogomips       : 4600.00
clflush size   : 64
cache_alignmen : 64
address sizes   : 46 bits physical, 48 bits virtual
power managemen:
```

```
[root@gopichand_k739 ~]# lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 1
On-line CPU(s) list: 0
Thread(s) per core: 1
Core(s) per socket: 1
Socket(s): 1
NUMA node(s): 1
Vendor ID: GenuineIntel
CPU family: 6
Model: 79
Model name: Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz
Stepping: 1
CPU MHz: 2300.056
BogoMIPS: 4600.00
Hypervisor vendor: Xen
Virtualization type: full
L1d cache: 32K
L1i cache: 32K
L2 cache: 256K
L3 cache: 46080K
NUMA node0 CPU(s): 0
Flags: fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc r
ep_good nopl xtopology cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand
hypervisor lahf_lm abm cpuid_fault invpcid_single pti fsgsbase bmi1 avx2 smep bmi2 erms invpcid xsaveopt
```

## To get RAM info:

- `cat /proc/meminfo`

Here meminfo is the file contains RAM details.

```
[root@gopichand_k739 ~]# cat /proc/meminfo
MemTotal:          975592 kB
MemFree:           267244 kB
MemAvailable:      739852 kB
Buffers:           948 kB
Cached:            581784 kB
SwapCached:        0 kB
Active:            209588 kB
Inactive:          403620 kB
Active(anon):       248 kB
Inactive(anon):     30636 kB
Active(file):       209340 kB
Inactive(file):     372984 kB
Unevictable:        0 kB
Mlocked:           0 kB
SwapTotal:         0 kB
SwapFree:          0 kB
Dirty:             12 kB
Writeback:         0 kB
AnonPages:         30512 kB
Mapped:            34716 kB
Shmem:             408 kB
KReclaimable:      35696 kB
```

- If we use, free -m (gives in MB),
- free -g (gives in GB), free -k (gives in KB)

```
[root@gopichand_k739 ~]# free -g
              total        used         free       shared    buff/cache   available
Mem:           0            0            0            0            0            0
Swap:          0            0            0
[root@gopichand_k739 ~]# free -m
              total        used         free       shared    buff/cache   available
Mem:          952            88          260            0            603          722
Swap:          0            0            0
[root@gopichand_k739 ~]# free -k
              total        used         free       shared    buff/cache   available
Mem:        975592          90148        266992          408        618452        739624
Swap:          0            0            0
[root@gopichand_k739 ~]# free -g
              total        used         free       shared    buff/cache   available
Mem:           0            0            0            0            0            0
Swap:          0            0            0
```

To get Storage details: df -h

To get the list of volumes attached: fdisk -l

```
[root@gopichand_k739 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        468M   0   468M   0% /dev
tmpfs           477M   0   477M   0% /dev/shm
tmpfs           477M  408K   476M   1% /run
tmpfs           477M   0   477M   0% /sys/fs/cgroup
/dev/xvda1      8.0G  2.0G   6.1G  24% /
tmpfs           96M    0    96M   0% /run/user/1000
[root@gopichand_k739 ~]# fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 1F77B33B-A3EA-4061-9406-586ABBBBCBA93

Device            Start      End   Sectors   Size Type
/dev/xvda1        4096 16777182 16773087    8G Linux filesystem
/dev/xvda128      2048    4095     2048    1M BIOS boot
```

### 3. File Commands:

**To create a file:** touch filename

**To view list of files:** ll or ls (ll-long list - displays full details of a file whereas ls -short list displays only file name)

**To create multiple files:** touch filename1 filename2 filename3 .... filename-n

If we want to create a filename with sequence

Ex: touch gopichand{1..5}

```
[root@gopichand_k739 ~]# touch linux-class-1.txt
[root@gopichand_k739 ~]# ll
total 0
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class-1.txt
```

```
[root@gopichand_k739 ~]# ls
linux-class2  linux-class3  linux-class4  linux-class5
```

ll -a (displays hiddens files along with other files)

ls -a (does same as above)

```
[root@gopichand_k739 ~]# ll -a
total 40
dr-xr-x---  3 root root   248 Oct 13 08:06 .
dr-xr-xr-x 18 root root   257 Oct 11 04:58 ..
-rw-----  1 root root  912 Oct 13 06:15 .bash_history
-rw-r--r--  1 root root   18 Oct 18 2017 .bash_logout
-rw-r--r--  1 root root  176 Oct 18 2017 .bash_profile
-rw-r--r--  1 root root  176 Oct 18 2017 .bashrc
-rw-r--r--  1 root root  100 Oct 18 2017 .cshrc
-rw-r--r--  1 root root    0 Oct 13 08:04 linux-class2
-rw-r--r--  1 root root    0 Oct 13 08:04 linux-class3
-rw-r--r--  1 root root    0 Oct 13 08:04 linux-class4
-rw-r--r--  1 root root    0 Oct 13 08:04 linux-class5
-rw-r--r--  1 root root 12288 Oct 11 06:26 .linux-notes.txt.swp
drwx-----  2 root root    29 Oct 11 04:59 .ssh
-rw-r--r--  1 root root   129 Oct 18 2017 .tcshrc
-rw-----  1 root root  3029 Oct 11 06:26 .viminfo
```

```
[root@gopichand_k739 ~]# ls -a
.  .bash_history  .bash_profile  .cshrc  linux-class3  linux-class5  .ssh  .viminfo
.. .bash_logout  .bashrc       linux-class2 linux-class4  .linux-notes.txt.swp .tcshrc
```

ll -r (or) ls -r (gives files in reverse order).

ll -t (or) ls -t (gives files in top order)

```
[root@gopichand_k739 ~]# ll -r
total 0
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class5
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class4
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class3
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class2
[root@gopichand_k739 ~]# ll -t
total 0
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class2
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class3
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class4
-rw-r--r--  1 root root 0 Oct 13 08:04 linux-class5
[root@gopichand_k739 ~]# ls -r
linux-class5 linux-class4 linux-class3 linux-class2
[root@gopichand_k739 ~]# ls -t
linux-class2 linux-class3 linux-class4 linux-class5
[root@gopichand_k739 ~]#
```

**To delete a file:** rm filename (In case of single file)

```
[root@gopichand_k739 ~]# rm linux-class-1.txt
rm: remove regular empty file 'linux-class-1.txt'? y
```

**To delete multiple files:** rm filename1 filename2...

but it will ask confirmation for each file and it will be difficult in case of numerous files. To overcome it we have other command

`rm -f filename1 filename2....filename-n`

Here -f is forcefully, it will delete the numerous files without asking for the confirmation

`rm -f *` (deletes all the files)

```
[root@gopichand_k739 ~]# ll
total 24
-rw-r--r-- 1 root root 67 Oct 11 06:05 file1
-rw-r--r-- 1 root root 34 Oct 11 06:05 file2
-rw-r--r-- 1 root root 26 Oct 11 06:03 file3
-rw-r--r-- 1 root root 26 Oct 11 06:04 file4
-rw-r--r-- 1 root root 26 Oct 11 06:04 file5
-rw-r--r-- 1 root root 0 Oct 11 05:56 linux-notes.txt
-rw-r--r-- 1 root root 187 Oct 11 06:12 prabhas
[root@gopichand_k739 ~]# rm -f *
[root@gopichand_k739 ~]# ll
total 0
```

`rm -f a*` (deletes files started with a, it is applicable for the other alphabets also)

`rm -f *.ext` (it deletes the files with the mentioned extension)

ex: `rm -f *.java` (it removes all the .java files, it is applicable for the other extensions also)

```
-rw-r--r-- 1 root root 0 Oct 13 08:05 java-class1.txt
-rw-r--r-- 1 root root 0 Oct 13 08:05 java-class2.txt
-rw-r--r-- 1 root root 0 Oct 13 08:05 java-class3.txt
-rw-r--r-- 1 root root 0 Oct 13 08:05 java-class4.txt
-rw-r--r-- 1 root root 0 Oct 13 08:05 java-class5.txt
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class2
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class3
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class4
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class5
[root@gopichand_k739 ~]# rm -f j*
[root@gopichand_k739 ~]# ll
total 0
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class2
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class3
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class4
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class5
```

**Directories (Means Folders):**

**To create new directory:** `mkdir dirname` ex: `mkdir AWS`

**To create multiple directories:** `mkdir dirname1 dirname2....dirname-n`

ex: `mkdir linux aws gcp`

**To create in sequence:** `mkdir aws{1..5}`



**To remove directory:** `rm dirname`

**To remove multiple directories:** `rm dirname1 dirname2...dirname-n` (Here it will not ask for the confirmation, it directly deletes the directory)

**To remove the directory and files at a time:** `rm -rf *`

## READ:

`cat filename` ex: `cat linux.txt` (it reads the data in a file)

`cat>filename` (It will look for a file in the directory if present, then allow us to write the data. If not present it will create and allows us to write the data)

Ex:

Linux (Directory)

- Linux-class1.txt (File)

`cat>Linux-class1.txt` (it allow us to write the data directly because Linux-class1.txt already exists)

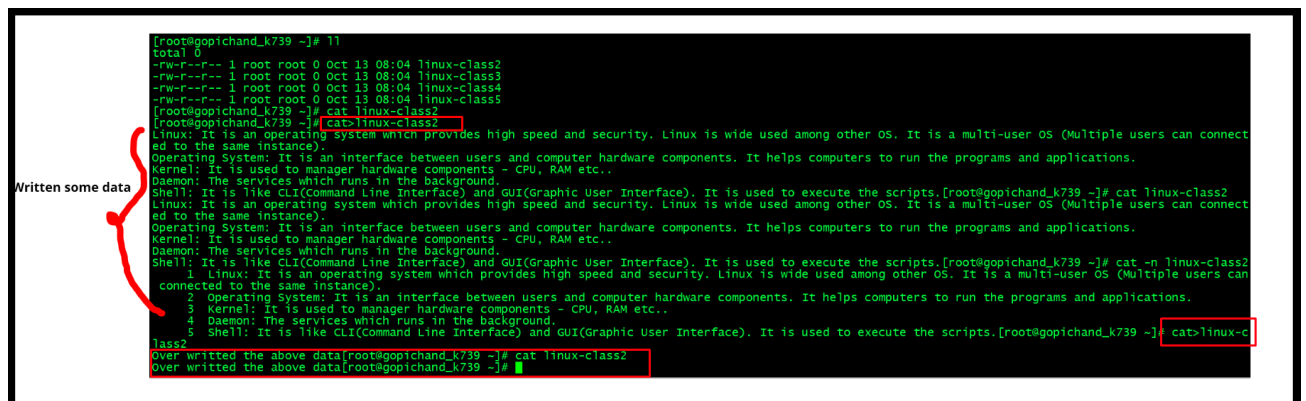
`cat>Linux-class2.txt` (it will create Linux-class2.txt as we don't have that file before and then it allows us to write the data)

`cat>filename` (it overwrites the data if we use `>` for the second time for a same file)

`cat>>filename` (it allows us to append the data to the file)

`cat` commands doesn't support to modify/correct the data in a file. To overcome this situation we use VIM editor

In Linux, CTRL + d - save



```
[root@gopichand_k739 ~]# ll
total 0
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class2
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class3
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class4
-rw-r--r-- 1 root root 0 Oct 13 08:04 linux-class5
[root@gopichand_k739 ~]# cat linux-class2
[redacted]
Linux: It is an operating system which provides high speed and security. Linux is wide used among other OS. It is a multi-user OS (Multiple users can connect to the same instance).
Operating System: It is an interface between users and computer hardware components. It helps computers to run the programs and applications.
Kernel: It is used to manager hardware components - CPU, RAM etc..
Daemon: The services which runs in the background.
Shell: It is like CLI(Command Line Interface) and GUI(Graphic User Interface). It is used to execute the scripts.[root@gopichand_k739 ~]# cat linux-class2
Linux: It is an operating system which provides high speed and security. Linux is wide used among other OS. It is a multi-user OS (Multiple users can connect to the same instance).
Operating System: It is an interface between users and computer hardware components. It helps computers to run the programs and applications.
Kernel: It is used to manager hardware components - CPU, RAM etc..
Daemon: The services which runs in the background.
Shell: It is like CLI(Command Line Interface) and GUI(Graphic User Interface). It is used to execute the scripts.[root@gopichand_k739 ~]# cat -n linux-class2
1 Linux: It is an operating system which provides high speed and security. Linux is wide used among other OS. It is a multi-user OS (Multiple users can connect to the same instance).
2 Operating System: It is an interface between users and computer hardware components. It helps computers to run the programs and applications.
3 Kernel: It is used to manager hardware components - CPU, RAM etc..
4 Daemon: The services which runs in the background.
5 Shell: It is like CLI(Command Line Interface) and GUI(Graphic User Interface). It is used to execute the scripts.[root@gopichand_k739 ~]# cat>linux-c
lass2
Over writted the above data[root@gopichand_k739 ~]# cat linux-class2
Over writted the above data[root@gopichand_k739 ~]#
```

## COPY/PASTE:

In Windows:

Terminal:

Copy - select the lines which you want to copy it will copy that data

Paste - Right click in mouse

Console:

Copy - Manual

Paste - CTRL + SHIFT + V (or) CTRL+INSERT (or) SHIFT+INSERT

**To display all the data in a file along with the line numbers:** cat -n filename

**To display first ten lines:** head filename

**To display last ten lines:** tail filename

**To display middle lines:** sed -n 'n,np' filename ex: sed -n '5,15p' linux.txt (it will display 5 to 15 lines of data in a linux.txt file)

**To reverse the order of data:** tac filename (it will reverse the top order to bottom, vice-versa)

**To read data in reverse:** rev filename

**To read multiple files at a time:** cat filename1 filename2 (But it will mess all the data/combines all files data and it is difficult to know the data to which file it belongs too, to overcome this we use

more filename1 filename2 (it will display data file by file)

**To get no.of lines, words, letters in a file:** wc filename

wc -l filename (it gets only no.of lines)

wc -w filename (it gets only no.of words)

wc -c filename (it gets only no.of letters)

## Directories:

cd - change directory

cd foldername → Navigates to a specific folder

default it will be root directory

cd (or) cd~ (If we use this command, wherever we are in, comes back to the root directory)

pwd - present working directory

cd..(one step back)

cd../..(two step back)

## Copy Commands:

**cp filename1 filename2**

It will ask for overwriting the content in file2 then we have to click y (or) n to modify

```

[root@gopichand_k739 ton]# touch file1 file2
[root@gopichand_k739 ton]# cat>file1
file2 is my friend
[root@gopichand_k739 ton]# cat file1
file2 is my friend
[root@gopichand_k739 ton]# cat>>file1
he is not the pen nor pencil to write.
current gone
mad is fan
[root@gopichand_k739 ton]# cat file1
file2 is my friend
he is not the pen nor pencil to write.
current gone
mad is fan
[root@gopichand_k739 ton]# cat -n file1
 1 file2 is my friend
 2 he is not the pen nor pencil to write.
 3 current gone
 4 mad is fan
[root@gopichand_k739 ton]# cp file1 file2
cp: overwrite 'file2'? y
[root@gopichand_k739 ton]# cat file2
file2 is my friend
he is not the pen nor pencil to write.
current gone
mad is fan
[root@gopichand_k739 ton]# cat -n file2
 1 file2 is my friend
 2 he is not the pen nor pencil to write.
 3 current gone
 4 mad is fan
[root@gopichand_k739 ton]# cat file1>>file2
[root@gopichand_k739 ton]# cat -n file2
 1 file2 is my friend
 2 he is not the pen nor pencil to write.
 3 current gone
 4 mad is fan
 5 file2 is my friend
 6 he is not the pen nor pencil to write.
 7 current gone
 8 mad is fan

```

it will overwrite the content in file2 to overcome it cat is placed

**cat file1>>file2**

it will append file1 data to file2

```
[root@gopichand_k739 ton]# cat file2
file2 is my friend
he is not the pen nor pencil to write.
current gone
mad is fan
file2 is my friend
he is not the pen nor pencil to write.
current gone
mad is fan
[root@gopichand_k739 ton]# cat -n file2
 1 file2 is my friend
 2 he is not the pen nor pencil to write.
 3 current gone
 4 mad is fan
 5 file2 is my friend
 6 he is not the pen nor pencil to write.
 7 current gone
 8 mad is fan
[root@gopichand_k739 ton]# cat file1>>file2
[root@gopichand_k739 ton]# cat -n file2
 1 file2 is my friend
 2 he is not the pen nor pencil to write.
 3 current gone
 4 mad is fan
 5 file2 is my friend
 6 he is not the pen nor pencil to write.
 7 current gone
 8 mad is fan
 9 file2 is my friend
10 he is not the pen nor pencil to write.
11 current gone
12 mad is fan
```

### Move Commands:

mv folder1 folder2 → it will move folder1 content to folder2

mv filename folder → it will move file to a folder

```

[root@gopichand_k739 ton]# ll
total 8
-rw-r--r-- 1 root root 82 Oct 17 09:32 file1
-rw-r--r-- 1 root root 246 Oct 17 09:36 file2
drwxr-xr-x 2 root root 179 Oct 17 09:39 ton
drwxr-xr-x 4 root root 29 Oct 17 09:39 ton1
[root@gopichand_k739 ton]# mv file1 ton
[root@gopichand_k739 ton]# ll
total 4
-rw-r--r-- 1 root root 246 Oct 17 09:36 file2
drwxr-xr-x 2 root root 192 Oct 17 09:40 ton
drwxr-xr-x 4 root root 29 Oct 17 09:39 ton1
[root@gopichand_k739 ton]# mv ton ton1
mv: overwrite 'ton1/ton'? ll
[root@gopichand_k739 ton]# ll
total 4
-rw-r--r-- 1 root root 246 Oct 17 09:36 file2
drwxr-xr-x 2 root root 192 Oct 17 09:40 ton
drwxr-xr-x 4 root root 29 Oct 17 09:39 ton1
[root@gopichand_k739 ton]# mv ton ton1
mv: overwrite 'ton1/ton'? y
[root@gopichand_k739 ton]# ll
total 4
-rw-r--r-- 1 root root 246 Oct 17 09:36 file2
drwxr-xr-x 4 root root 29 Oct 17 09:40 ton1

```

### VIM Editor:

IT IS USED TO MODIFY THE DATA IN A FILE. IT HAS 3 MODES

1. COMMAND MODE
2. INSERT MODE
3. SAVE & QUIT MODE

### COMMAND MODE:

This is the default mode in vim editor.

This is used to perform some actions like copy the data, delete the data, and undo, redo, search for a word and also we can move to a particular line.

gg : used to go to 1st line of a file

G : used to go to last line of a file

5gg : used to go to 5th line of a file

:19 : we can move to 19th line of a file

:set number : used to set a numbers of a file

yy : copy a line

3yy : used to copy 3 lines from our cursor

p : used to paste the data

5p : used to paste 5 times

dd : delete the entire line

5dd: used to delete 5 lines at a time

u : used for undo

ctrl + r : used for redo

/word : used to search for a word

?word : used to search for a word in a file

**:%s/oldword/new-word/** : used to replace a single occurrence in a line

ex: my name is gopi, chand is a devops engineer, chand is from hyd.

**:%s/chand/gopi/g**

if we use this command the the o/p: my name is gopi, gopi is a devops engineer, gopi is from hyd.

because i used g (global)

### **INSERT MODE:**

It is used to insert the data.

But by default we are in command mode, if you wish to go to command mode to insert mod

you can use small i (i)

To go back to command mode : esc

A : used to go to the end of the line

I : used to go to the starting of the line.

O : used to create a new line (up)

o : used to create a new line (down)

### **SAVE & QUIT MODE:**

To view a file one-by-one: `vim -o file1 file2`





```
~  
managers  
"managers" 3L, 31B
```

## USER COMMANDS:

We can have multiple users in Linux and we can connect to an instance with multiple users.

root is ultimate users in Linux

→ To view list of users available in an instance: [cat /etc/passwd](#)

```
[root@ip-172-31-92-87 home]# cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
bin:x:1:1:bin:/bin:/sbin/nologin  
daemon:x:2:2:daemon:/sbin:/sbin/nologin  
adm:x:3:4:adm:/var/adm:/sbin/nologin  
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin  
sync:x:5:0:sync:/sbin:/bin/sync  
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown  
halt:x:7:0:halt:/sbin:/sbin/halt  
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin  
operator:x:11:0:operator:/root:/sbin/nologin  
games:x:12:100:games:/usr/games:/sbin/nologin  
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin  
nobody:x:99:99:Nobody:/:/sbin/nologin  
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin  
dbus:x:81:81:System message bus:/:/sbin/nologin  
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin  
libstoragemgmt:x:999:997:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin  
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin  
rngd:x:998:996:Random Number Generator Daemon:/var/lib/rngd:/sbin/nologin  
chrony:x:997:995:/:/var/lib/chrony:/sbin/nologin  
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin  
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin  
ec2-instance-connect:x:996:994:/:/home/ec2-instance-connect:/sbin/nologin  
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin  
tcpdump:x:72:72:/:/sbin/nologin  
ec2-user:x:1000:1000:EC2 Default User:/home/ec2-user:/bin/bash  
gopichand:x:1001:1001:/:/home/gopichand:/bin/bash  
linux-flm:x:1002:1002:/:/home/linux-flm:/bin/bash  
git-flm:x:1003:1003:/:/home/git-flm:/bin/bash  
jenkins:x:1004:1004:/:/home/jenkins:/bin/bash  
LEO:x:1005:1005:/:/home/LEO:/bin/bash  
[root@ip-172-31-92-87 home]#
```

→ To create a new user: [useradd username](#)

**Ex: useradd leo**

```
[root@ip-172-31-92-87 home]# useradd tnr
[root@ip-172-31-92-87 home]# ll
total 0
drwx----- 3 ec2-user  ec2-user  95 Oct 18 08:07 ec2-user
drwx----- 2 git-flm   git-flm   83 Oct 18 08:45 git-flm
drwx----- 2 gopichand gopichand 83 Oct 18 08:10 gopichand
drwx----- 2 LEO      LEO       62 Oct 18 09:33 LEO
drwx----- 2 linux-flm linux-flm 83 Oct 18 08:12 linux-flm
drwx----- 2 LEO      LEO       62 Oct 18 09:24 pipelines
drwx----- 2 tnr      tnr        62 Oct 18 09:41 tnr
```

→ **Important Note:** Whenever we create a new user, it will create a folder with that user name in /home path

```
[root@ip-172-31-92-87 ~]# cd /home
[root@ip-172-31-92-87 home]# ll
total 0
drwx----- 3 ec2-user  ec2-user  95 Oct 18 08:07 ec2-user
drwx----- 2 git-flm   git-flm   83 Oct 18 08:45 git-flm
drwx----- 2 gopichand gopichand 83 Oct 18 08:10 gopichand
drwx----- 2 LEO      LEO       62 Oct 18 09:33 LEO
drwx----- 2 linux-flm linux-flm 83 Oct 18 08:12 linux-flm
drwx----- 2 LEO      LEO       62 Oct 18 09:24 pipelines
drwx----- 2 tnr      tnr        62 Oct 18 09:41 tnr
[root@ip-172-31-92-87 home]#
```

→ To create a new user without creating folder: useradd -M username

```
[root@ip-172-31-92-87 /]# useradd -M dunki
[root@ip-172-31-92-87 /]# cat /etc/passwd
```

```
HiNanna:x:1009:1009::/home/HiNanna:/bin/bash
dunki:x:1009:1009::/home/dunki:/bin/bash
[root@ip-172-31-92-87 /]# cd /home
[root@ip-172-31-92-87 home]# ll
total 0
drwx----- 3 ec2-user  ec2-user  95 Oct 18 08:07 ec2-user
drwx----- 2 git-flm   git-flm   83 Oct 18 08:45 git-flm
drwx----- 2 gopichand gopichand 83 Oct 18 08:10 gopichand
drwx----- 2 HiNanna  HiNanna  62 Oct 27 09:03 HiNanna
drwx----- 2 LEO      LEO       62 Oct 18 09:33 LEO
drwx----- 2 linux-flm linux-flm 83 Oct 18 08:12 linux-flm
drwx----- 2 pepsico  pepsico  62 Oct 27 06:45 pepsico
drwx----- 2 LEO      LEO       62 Oct 18 09:24 pipelines
drwx----- 2          1006    1006  62 Oct 18 09:41 tnr
```

→ To set the password to a user: passwd username

```
[root@ip-172-31-92-87 ~]# passwd HiNanna
Changing password for user HiNanna.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ip-172-31-92-87 ~]# █
```

provide new password and re-confirm the new password

If we forgot password, repeat the same above steps

**Important Note: Only root user can specify a password for the user.**

```
[root@ip-172-31-92-87 ~]# su - HiNanna
[HiNanna@ip-172-31-92-87 ~]$ uptime
 09:18:54 up 9 days,  1:43,  1 user,  load average: 0.00, 0.00, 0.00
[HiNanna@ip-172-31-92-87 ~]$ passwd HiNanna
passwd: Only root can specify a user name.
[HiNanna@ip-172-31-92-87 ~]$ █
```

From root user to any user we can switch without any password.

From Normal user to switch another user it requires password

```
[root@ip-172-31-92-87 ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
libstoragemgmt:x:999:997:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rngd:x:998:996:Random Number Generator Daemon:/var/lib/rngd:/sbin/nologin
chrony:x:997:995:/:/var/lib/chrony:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
ec2-instance-connect:x:996:994:/:/home/ec2-instance-connect:/sbin/nologin
postfix:x:89:89:/:/var/spool/postfix:/sbin/nologin
tcpdump:x:72:72:/:/sbin/nologin
ec2-user:x:1000:1000:EC2 Default User:/home/ec2-user:/bin/bash
gopichand:x:1001:1001:/:/home/gopichand:/bin/bash
linux-flm:x:1002:1002:/:/home/linux-flm:/bin/bash
git-flm:x:1003:1003:/:/home/git-flm:/bin/bash
jenkins:x:1004:1004:/:/home/jenkins:/bin/bash
LEO:x:1005:1005:/:/home/LEO:/bin/bash
pepsico:x:1007:1007:/:/home/pepsico:/bin/bash
HiNanna:x:1008:1008:/:/home/HiNanna:/bin/bash
dunki:x:1009:1009:/:/home/dunki:/bin/bash
[root@ip-172-31-92-87 ~]# su - pepsico
[pepsico@ip-172-31-92-87 ~]$ su - HiNanna
Password:
Last login: Fri Oct 27 09:22:34 UTC 2023 on pts/0
[HiNanna@ip-172-31-92-87 ~]$
```

Under root, checked the present users

Switched to pepsico from root without any password

From pepsico to HiNanna, it asked for the password

To create a user with the limited time:

**useradd -e date username**

Here date should be in the format YYYY-MM-DD

ex: **useradd -e 2024-01-01 leo**

To check expiry date of user: **chage -l username**

```
[root@ip-172-31-92-87 ~]# useradd -e 2023-11-23 leo
[root@ip-172-31-92-87 ~]# chage -l leo
Last password change                : Oct 27, 2023
Password expires                    : never
Password inactive                   : never
Account expires                     : Nov 23, 2023
Minimum number of days between password change : 0
Maximum number of days between password change : 99999
Number of days of warning before password expires : 7
[root@ip-172-31-92-87 ~]#
```

To delete a user: **userdel username**

Note: It will delete only user not the folder associated with the user

To delete a user and user folder at a time: **userdel -r username**

```
[root@ip-172-31-92-87 ~]# cd /home/
[root@ip-172-31-92-87 home]# ll
total 0
drwx----- 3 ec2-user  ec2-user  95 Oct 18 08:07 ec2-user
drwx----- 2 git-flm   git-flm   83 Oct 18 08:45 git-flm
drwx----- 2 gopichand gopichand 83 Oct 18 08:10 gopichand
drwx----- 2 HiNanna  HiNanna  83 Oct 27 09:21 HiNanna
drwx----- 2 leo      leo      62 Oct 27 09:35 leo
drwx----- 2 LEO      LEO      62 Oct 18 09:33 LEO
drwx----- 2 linux-flm linux-flm 83 Oct 18 08:12 linux-flm
drwx----- 2 pepsico   pepsico   83 Oct 27 09:31 pepsico
drwx----- 2 LEO      LEO      62 Oct 18 09:24 pipelines
drwx----- 2          1006      1006 62 Oct 18 09:41 tnr
[root@ip-172-31-92-87 home]# userdel -r LEO
[root@ip-172-31-92-87 home]# ll
total 0
drwx----- 3 ec2-user  ec2-user  95 Oct 18 08:07 ec2-user
drwx----- 2 git-flm   git-flm   83 Oct 18 08:45 git-flm
drwx----- 2 gopichand gopichand 83 Oct 18 08:10 gopichand
drwx----- 2 HiNanna  HiNanna  83 Oct 27 09:21 HiNanna
drwx----- 2 leo      leo      62 Oct 27 09:35 leo
drwx----- 2 linux-flm linux-flm 83 Oct 18 08:12 linux-flm
drwx----- 2 pepsico   pepsico   83 Oct 27 09:31 pepsico
drwx----- 2          1005      1005 62 Oct 18 09:24 pipelines
drwx----- 2          1006      1006 62 Oct 18 09:41 tnr
```

Group Commands:

To see the list of groups in a system

Syntax: `cat /etc/group`

```
[root@ip-172-31-92-87 /]# cat /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
sys:x:3:
adm:x:4:ec2-user
tty:x:5:
disk:x:6:
lp:x:7:
mem:x:8:
kmem:x:9:
wheel:x:10:ec2-user
cdrom:x:11:
mail:x:12:postfix
man:x:15:
```

Note: Whenever we create a new user it will create a new group with the same name automatically also deletes the group whenever we deletes the user.

To create only group without user:

**Syntax:** groupadd groupname

**Ex:** groupadd devara

```
[root@ip-172-31-92-87 /]# groupadd devara
```

cat /etc/group

```
devara:x:1011:
```

To delete a group: groupdel groupname

```
[root@ip-172-31-92-87 /]# groupdel devara
[root@ip-172-31-92-87 /]# cat /etc/group
```

To add a user into a group

Syntax: `usermod -a -G groupname username`

```
[root@ip-172-31-92-87 ~]# usermod -a -G pepsico leo
[root@ip-172-31-92-87 ~]# id leo
uid=1010(leo) gid=1010(leo) groups=1010(leo),1007(pepsico)
[root@ip-172-31-92-87 ~]#
```