MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

Santosh,Tangail-1902

LAB REPORT

Lab Report No	: 06
Lab Report name	: Linux command for process
Course Title	: Operating System Lab
Course Code	: ICT-3110
Date of Performance	: 20-08-2020
Date of Submission	: 28-08-2020

Submitted by,

Student Name	: Tanvir Ahmed
Student ID	: IT-18043
Session	: 2017-18

3rd Year 1st semester

Dept. of ICT

Submitted to,

Nazrul Islam

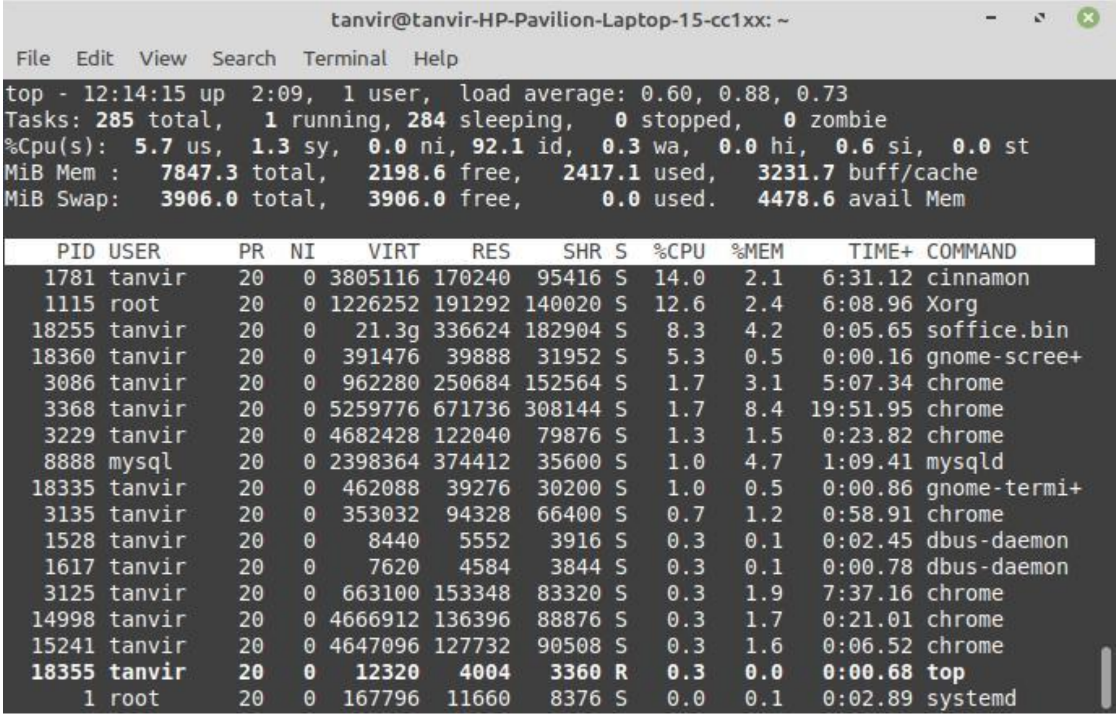
Assistant Professor

Dept. of ICT,

MBSTU.

Lab 06 - Linux command for process

1) **top:** The top command is the traditional way to view your system’s resource usage and see the processes that are taking up the most system resources. Top displays a list of processes, with the ones using the most CPU at the top.


tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

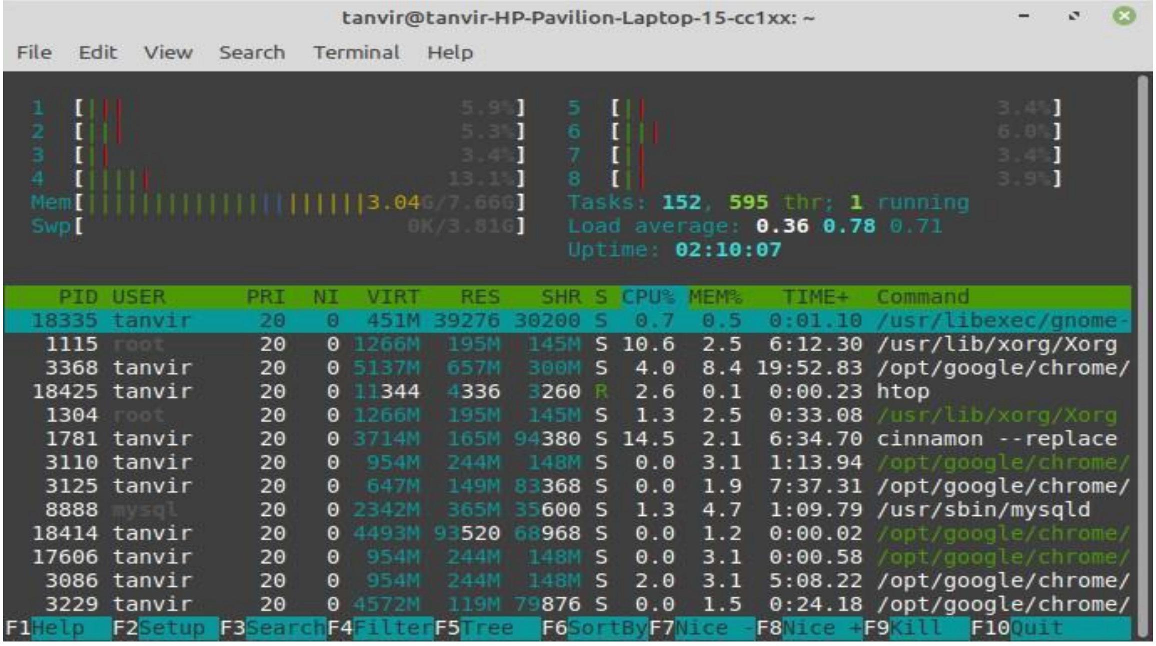
```
top - 12:14:15 up  2:09,  1 user,  load average: 0.60, 0.88, 0.73
Tasks: 285 total,   1 running, 284 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.7 us,  1.3 sy,  0.0 ni, 92.1 id,  0.3 wa,  0.0 hi,  0.6 si,  0.0 st
MiB Mem :   7847.3 total,   2198.6 free,   2417.1 used,   3231.7 buff/cache
MiB Swap:   3906.0 total,   3906.0 free,      0.0 used.   4478.6 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   1781 tanvir    20   0 3805116 170240  95416 S  14.0   2.1   6:31.12 cinnamon
   1115 root      20   0 1226252 191292 140020 S  12.6   2.4   6:08.96 Xorg
  18255 tanvir    20   0   21.3g 336624 182904 S   8.3   4.2   0:05.65 soffice.bin
  18360 tanvir    20   0  391476  39888  31952 S   5.3   0.5   0:00.16 gnome-scree+
   3086 tanvir    20   0  962280 250684 152564 S   1.7   3.1   5:07.34 chrome
   3368 tanvir    20   0 5259776 671736 308144 S   1.7   8.4  19:51.95 chrome
   3229 tanvir    20   0 4682428 122040  79876 S   1.3   1.5   0:23.82 chrome
   8888 mysql     20   0 2398364 374412  35600 S   1.0   4.7   1:09.41 mysqld
  18335 tanvir    20   0  462088  39276  30200 S   1.0   0.5   0:00.86 gnome-termi+
   3135 tanvir    20   0  353032  94328  66400 S   0.7   1.2   0:58.91 chrome
   1528 tanvir    20   0    8440   5552   3916 S   0.3   0.1   0:02.45 dbus-daemon
   1617 tanvir    20   0    7620   4584   3844 S   0.3   0.1   0:00.78 dbus-daemon
   3125 tanvir    20   0  663100 153348  83320 S   0.3   1.9   7:37.16 chrome
  14998 tanvir    20   0 4666912 136396  88876 S   0.3   1.7   0:21.01 chrome
  15241 tanvir    20   0 4647096 127732  90508 S   0.3   1.6   0:06.52 chrome
  18355 tanvir    20   0   12320   4004   3360 R   0.3   0.0   0:00.68 top
      1 root      20   0  167796  11660   8376 S   0.0   0.1   0:02.89 systemd
```

To exit top or htop, use the Ctrl- C keyboard shortcut. This keyboard shortcut usually kills the currently running process in the terminal.

2) htop: The **htop** command is an improved top. It’s not installed by default on most Linux distributions — here’s the command you’ll need to install it on Ubuntu:

sudo apt-get install htop

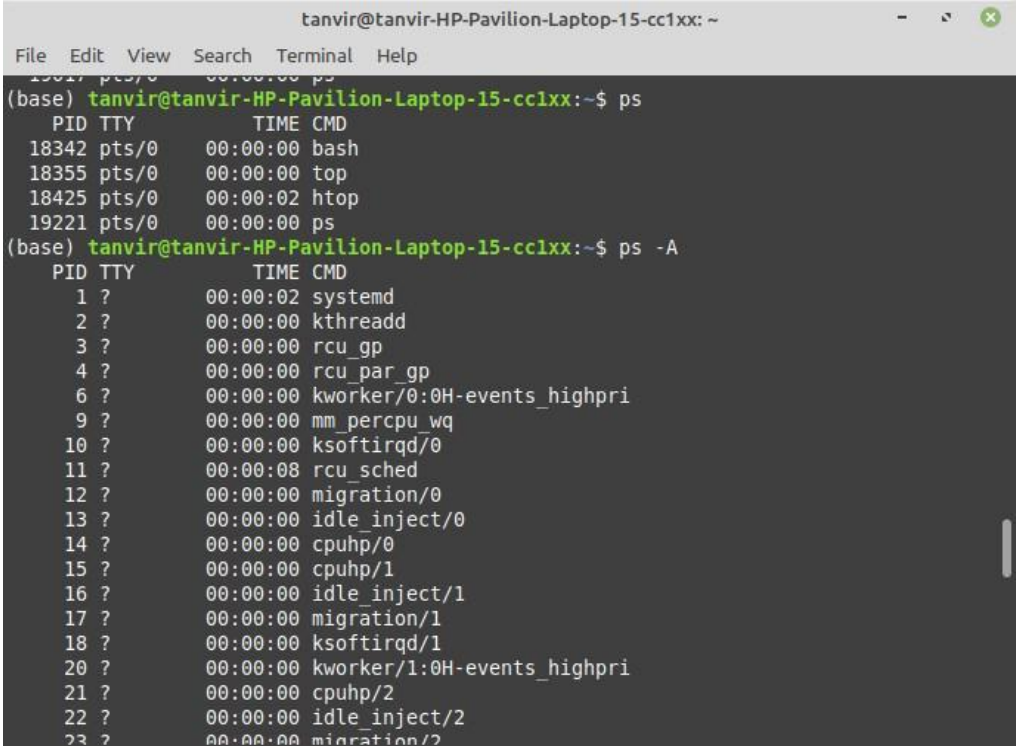

tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

```
  1  [|||                          5.9%]   5  [||                           3.4%]
  2  [|||                          5.3%]   6  [|||                          6.0%]
  3  [||                           3.4%]   7  [||                           3.4%]
  4  [|||||                       13.1%]   8  [||                           3.9%]
  Mem[||||||||||||||||||||||3.04G/7.66G]   Tasks: 152, 595 thr; 1 running
  Swp[                          0K/3.81G]   Load average: 0.36 0.78 0.71
                                            Uptime: 02:10:07

    PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
  18335 tanvir     20   0  451M 39276 30200 S  0.7  0.5  0:01.10 /usr/libexec/gnome-
   1115 root       20   0 1266M  195M  145M S 10.6  2.5  6:12.30 /usr/lib/xorg/Xorg
   3368 tanvir     20   0 5137M  657M  300M S  4.0  8.4 19:52.83 /opt/google/chrome/
  18425 tanvir     20   0 11344  4336  3260 R  2.6  0.1  0:00.23 htop
   1304 root       20   0 1266M  195M  145M S  1.3  2.5  0:33.08 /usr/lib/xorg/Xorg
   1781 tanvir     20   0 3714M  165M 94380 S 14.5  2.1  6:34.70 cinnamon --replace
   3110 tanvir     20   0  954M  244M  148M S  0.0  3.1  1:13.94 /opt/google/chrome/
   3125 tanvir     20   0  647M  149M 83368 S  0.0  1.9  7:37.31 /opt/google/chrome/
   8888 mysql      20   0 2342M  365M 35600 S  1.3  4.7  1:09.79 /usr/sbin/mysqld
  18414 tanvir     20   0 4493M 93520 68968 S  0.0  1.2  0:00.02 /opt/google/chrome/
  17606 tanvir     20   0  954M  244M  148M S  0.0  3.1  0:00.58 /opt/google/chrome/
   3086 tanvir     20   0  954M  244M  148M S  2.0  3.1  5:08.22 /opt/google/chrome/
   3229 tanvir     20   0 4572M  119M 79876 S  0.0  1.5  0:24.18 /opt/google/chrome/
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

3) **<u>ps -A :</u>** The **ps** command lists running processes. The following command lists all processes running on your system:

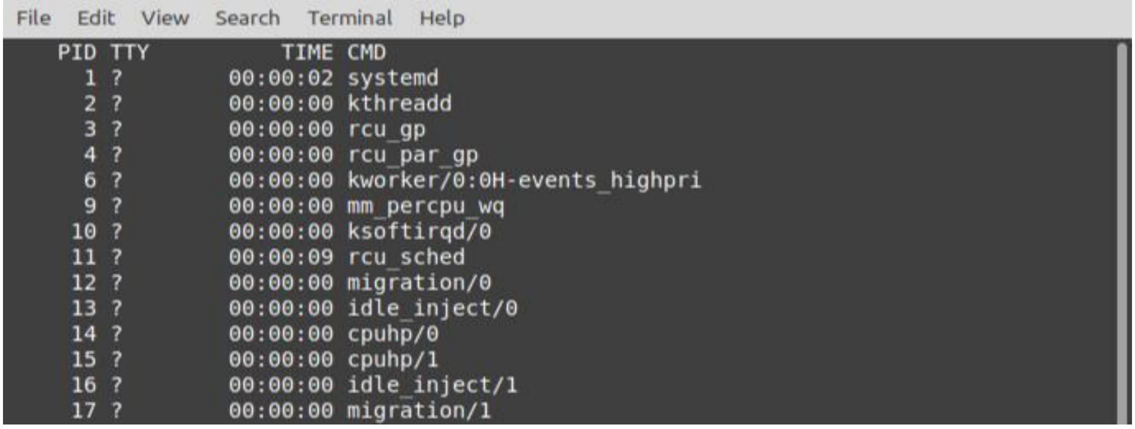
ps -A


tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

```
 19217 pts/0    00:00:00 ps
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ ps
    PID TTY          TIME CMD
  18342 pts/0    00:00:00 bash
  18355 pts/0    00:00:00 top
  18425 pts/0    00:00:02 htop
  19221 pts/0    00:00:00 ps
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ ps -A
    PID TTY          TIME CMD
      1 ?        00:00:02 systemd
      2 ?        00:00:00 kthreadd
      3 ?        00:00:00 rcu_gp
      4 ?        00:00:00 rcu_par_gp
      6 ?        00:00:00 kworker/0:0H-events_highpri
      9 ?        00:00:00 mm_percpu_wq
     10 ?        00:00:00 ksoftirqd/0
     11 ?        00:00:08 rcu_sched
     12 ?        00:00:00 migration/0
     13 ?        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-events_highpri
     21 ?        00:00:00 cpuhp/2
     22 ?        00:00:00 idle_inject/2
     23 ?        00:00:00 migration/2
```

4) **<u>ps -A | less:</u>** ***<u>ps -A</u>*** may be too many processes to read at one time, so we can pipe the output through the **less** command to scroll through them at own pace.

ps -A | less:

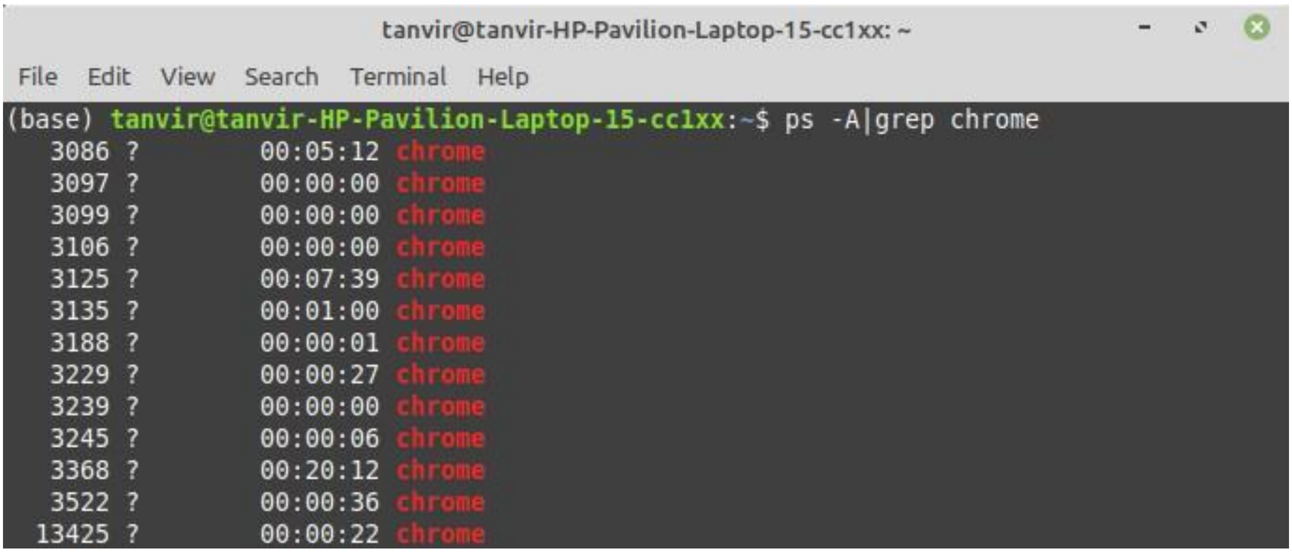


```
    PID TTY          TIME CMD
      1 ?        00:00:02 systemd
      2 ?        00:00:00 kthreadd
      3 ?        00:00:00 rcu_gp
      4 ?        00:00:00 rcu_par_gp
      6 ?        00:00:00 kworker/0:0H-events_highpri
      9 ?        00:00:00 mm_percpu_wq
     10 ?        00:00:00 ksoftirqd/0
     11 ?        00:00:09 rcu_sched
     12 ?        00:00:00 migration/0
     13 ?        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
```

Press q to exit when you’re done.

5) ps -A | grep : We could also pipe the output through **grep** to search for a specific process without using any other commands. The following command would search for the Firefox process:

ps -A | grep firefox

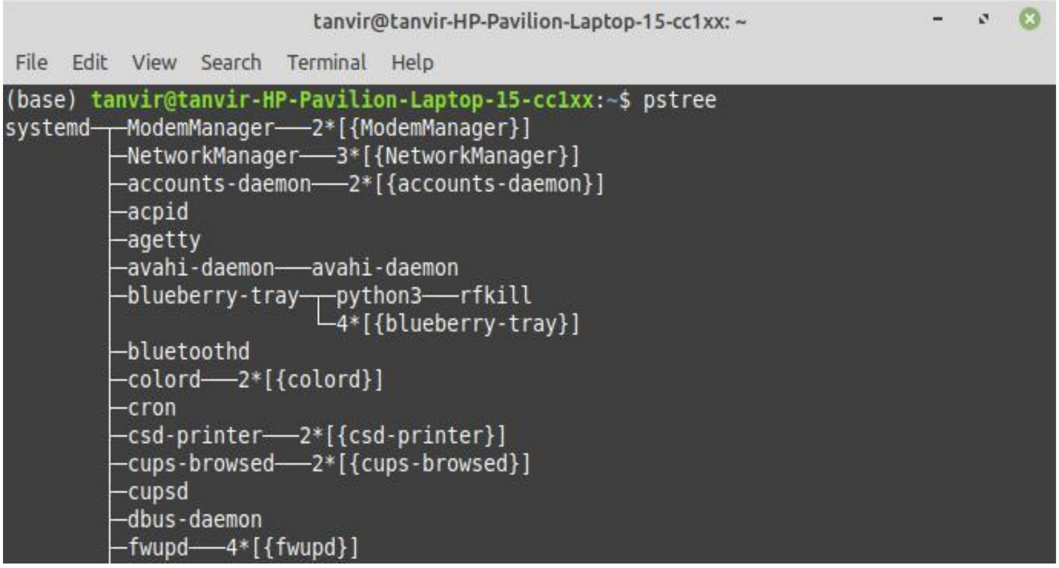


tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

```
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ ps -A|grep chrome
   3086 ?        00:05:12 chrome
   3097 ?        00:00:00 chrome
   3099 ?        00:00:00 chrome
   3106 ?        00:00:00 chrome
   3125 ?        00:07:39 chrome
   3135 ?        00:01:00 chrome
   3188 ?        00:00:01 chrome
   3229 ?        00:00:27 chrome
   3239 ?        00:00:00 chrome
   3245 ?        00:00:06 chrome
   3368 ?        00:20:12 chrome
   3522 ?        00:00:36 chrome
  13425 ?        00:00:22 chrome
```

6) pstree:

The **pstree** command is another way of visualizing processes. It displays them in tree format.



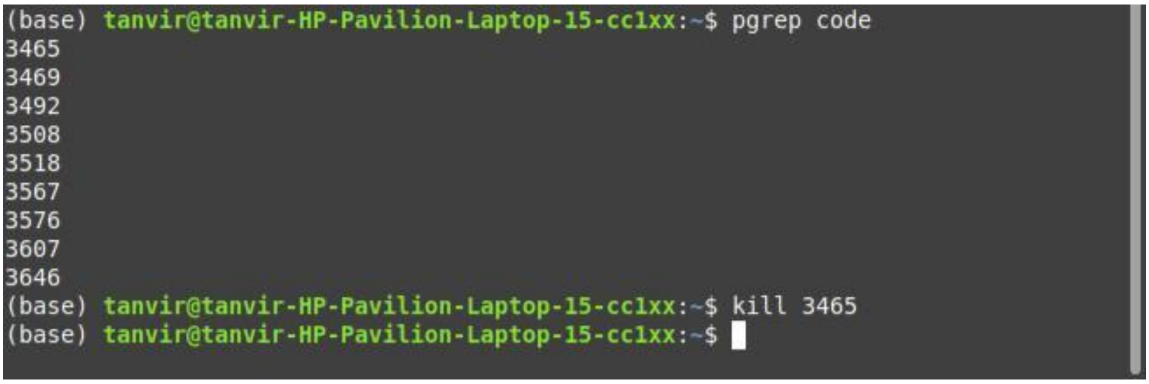
tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

```
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ pstree
systemd─┬─ModemManager───2*[{ModemManager}]
        ├─NetworkManager───3*[{NetworkManager}]
        ├─accounts-daemon───2*[{accounts-daemon}]
        ├─acpid
        ├─agetty
        ├─avahi-daemon───avahi-daemon
        ├─blueberry-tray─┬─python3───rfkill
        │                └─4*[{blueberry-tray}]
        ├─bluetoothd
        ├─colord───2*[{colord}]
        ├─cron
        ├─csd-printer───2*[{csd-printer}]
        ├─cups-browsed───2*[{cups-browsed}]
        ├─cupsd
        ├─dbus-daemon
        ├─fwupd───4*[{fwupd}]
```

7) kill :

The **kill** command can kill a process, given its process ID. You can get this information from the **ps -A**, **top** or **pgrep** commands.

kill PID

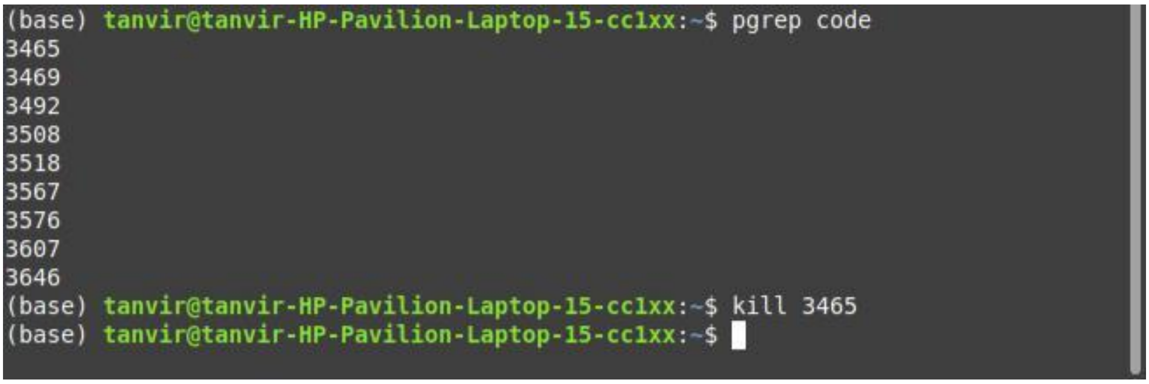


```
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ pgrep code
3465
3469
3492
3508
3518
3567
3576
3607
3646
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ kill 3465
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ 
```

8) pgrep :

Given a search term, **pgrep** returns the process IDs that match it. For example, you could use the following command to find Firefox’s PID:

pgrep firefox

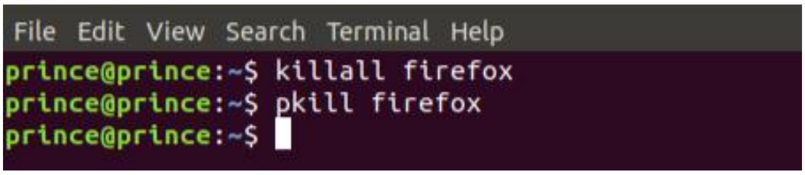


```
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ pgrep code
3465
3469
3492
3508
3518
3567
3576
3607
3646
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ kill 3465
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ 
```

9) pkill & killall :

The **pkill** and **killall** commands can kill a process, given its name. Use either command to kill Firefox:

pkill firefox
killall firefox



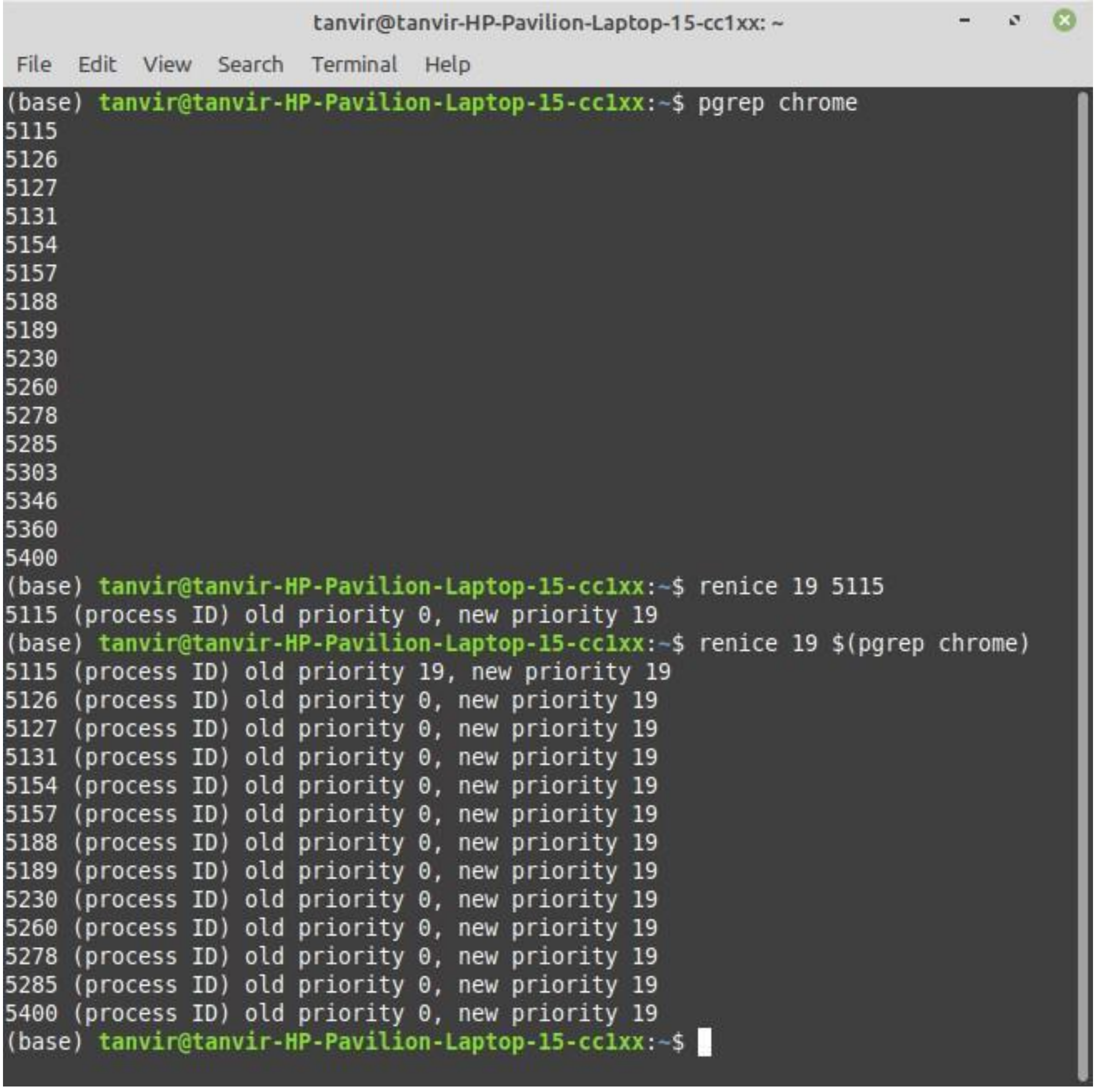
```
prince@prince:~$ killall firefox
prince@prince:~$ pkill firefox
prince@prince:~$ 
```

10) renice:

The **renice** command changes the nice value of an already running process. The nice value determines what priority the process runs with. A value of **-19** is very high priority, while a value of **19** is very low priority. A value of **0** is the default priority.

The renice command requires a process's PID. The following command makes a process run with very low priority:

renice 19 *PID*



tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx: ~
File Edit View Search Terminal Help

```
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ pgrep chrome
5115
5126
5127
5131
5154
5157
5188
5189
5230
5260
5278
5285
5303
5346
5360
5400
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ renice 19 5115
5115 (process ID) old priority 0, new priority 19
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$ renice 19 $(pgrep chrome)
5115 (process ID) old priority 19, new priority 19
5126 (process ID) old priority 0, new priority 19
5127 (process ID) old priority 0, new priority 19
5131 (process ID) old priority 0, new priority 19
5154 (process ID) old priority 0, new priority 19
5157 (process ID) old priority 0, new priority 19
5188 (process ID) old priority 0, new priority 19
5189 (process ID) old priority 0, new priority 19
5230 (process ID) old priority 0, new priority 19
5260 (process ID) old priority 0, new priority 19
5278 (process ID) old priority 0, new priority 19
5285 (process ID) old priority 0, new priority 19
5400 (process ID) old priority 0, new priority 19
(base) tanvir@tanvir-HP-Pavilion-Laptop-15-cc1xx:~$
```

Conclusion : Process is the most common phenomenon of any operating system . This is the very basic part of execution of program.
In this lab experiment we learned how to work with processes. Here we learned some basic linux command like – top , htop , ps , pstree , kill , pkill , killall , pgrep , renice etc. These are very helpful for process management.
Here we saw how process are identified.The PID or process id number refers to the identity of process . top and htop commands are for showing the detail of processes.

Pstree shows how processes are connected by sup-processes. pgrep finds the pid of any process by name. Kill commands terminates the processes by pid or by name.
However we learned process management in detail in this lab experiment . It was not very difficult to learn because in the earlier lab experiment we have learned to deal with thread.