



MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY
Santosh,Tangail-1902

LAB REPORT

Department of : Information & Communication Technology

Lab Report No : 06

Lab Report On : **Linux command for process**

Course Title : Operating Systems Lab

Course Code : ICT - 3110

Submitted by,

Name : Zahid Hasan Chowdhury

Student ID : IT-18017

Session : 2017-18

Year: 3rd Semester : 1st

Dept. of ICT
MBSTU

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.

```
zahid@zahid: ~
File Edit View Search Terminal Help
top - 19:23:15 up 38 min, 1 user, load average: 0.89, 0.71, 0.67
Tasks: 315 total, 4 running, 237 sleeping, 4 stopped, 0 zombie
%Cpu(s): 6.7 us, 2.1 sy, 0.0 ni, 90.5 id, 0.2 wa, 0.0 hi, 0.5 si, 0.0 st
KiB Mem : 8032424 total, 2973724 free, 2299812 used, 2758888 buff/cache
KiB Swap: 4883452 total, 4883452 free, 0 used. 4172856 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1692 zahid 20 0 4222412 233288 106248 R 20.1 2.9 1:49.17 gnome-shell
2794 zahid 20 0 5995856 413292 116956 S 11.2 5.1 3:32.49 chrome
2011 zahid 20 0 4052424 609028 371412 S 10.2 7.6 2:05.35 chrome
2165 zahid 20 0 1482076 231204 137128 R 10.2 2.9 2:02.84 chrome
2377 zahid 20 0 6151348 536028 253812 S 7.6 6.7 5:04.89 chrome
1566 zahid 20 0 909548 62072 42972 R 7.3 0.8 1:12.61 Xorg
1718 zahid 9 -11 2383624 18532 14288 S 5.6 0.2 2:01.78 pulseaudio
3778 zahid 20 0 9850.7m 169964 123164 S 3.6 2.1 0:06.69 chrome
3501 zahid 20 0 5776196 112168 85224 S 3.0 1.4 0:04.67 chrome
2429 zahid 20 0 1482632 55752 46084 S 1.7 0.7 0:42.68 chrome
244 root 20 0 0 0 0 D 1.0 0.0 0:03.26 kworker/u16:5+e
3179 zahid 20 0 791504 35824 26868 S 1.0 0.4 0:14.67 gnome-terminal-
3712 root 20 0 0 0 0 I 1.0 0.0 0:00.11 kworker/0:1-eve
943 root 20 0 559168 16776 13572 S 0.7 0.2 0:01.15 NetworkManager
2171 zahid 20 0 1302916 88780 64152 S 0.7 1.1 0:26.55 chrome
3387 root 20 0 0 0 0 I 0.7 0.0 0:00.30 kworker/6:4-eve
3845 zahid 20 0 44544 4052 3348 R 0.7 0.1 0:00.29 top
1 root 20 0 225664 9484 6796 S 0.3 0.1 0:05.01 systemd
310 root 0 -20 0 0 0 I 0.3 0.0 0:00.08 kworker/2:1H-ev
479 root -51 0 0 0 0 S 0.3 0.0 0:04.87 irq/128-iwlwifi
1105 mysql 20 0 1555616 189856 15684 S 0.3 2.4 0:03.50 mysqld
```

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:

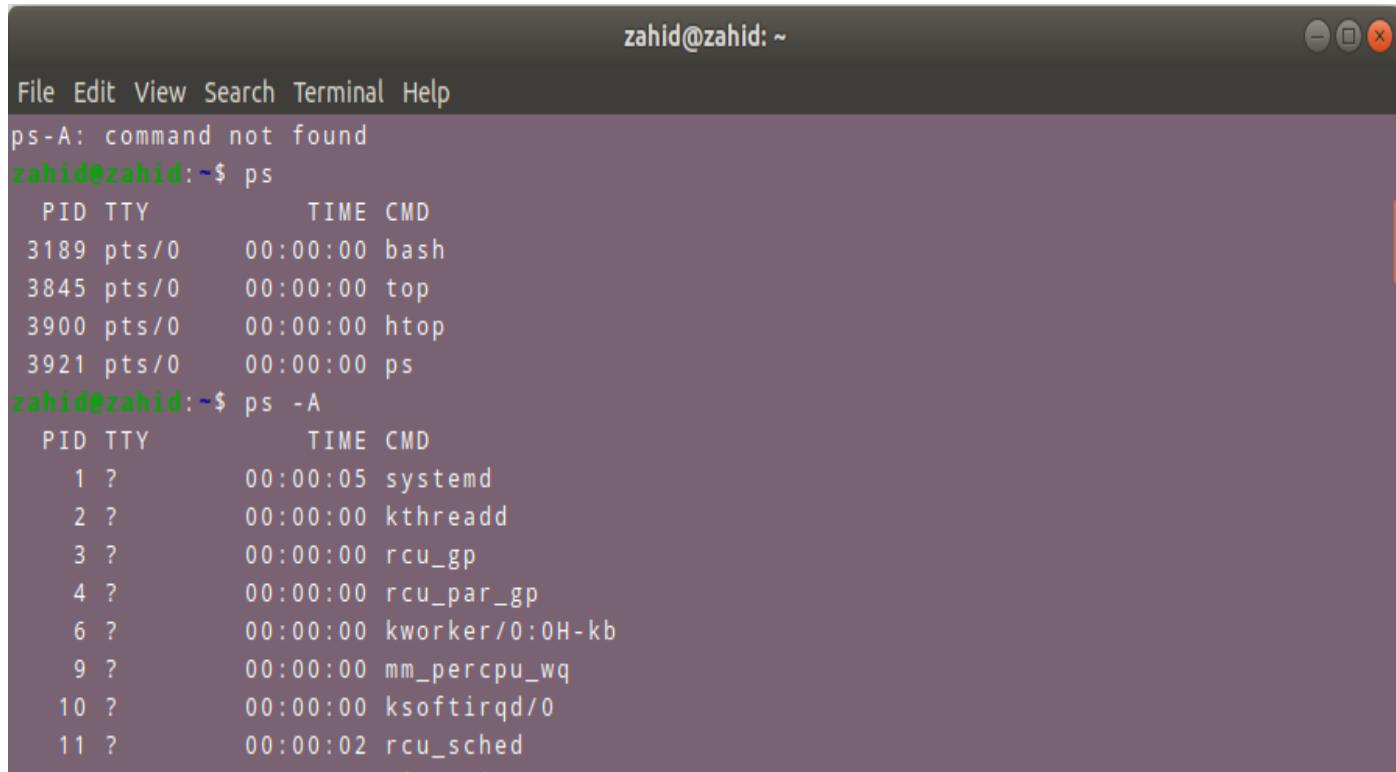
```
sudo apt-get install htop
```

```
zahid@zahid: ~
File Edit View Search Terminal Help
1 [ ] 9.0% 5 [ ] 3.3%
2 [ ] 1.9% 6 [ ] 4.6%
3 [ ] 6.0% 7 [ ] 9.7%
4 [ ] 5.9% 8 [ ] 13.7%
Mem [ ] 8.68G / 7.66G Tasks: 162, 636 thr; 1 running
Swp [ OK / 4.66G Load average: 1.11 0.78 0.69
Uptime: 00:38:24

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
3778 zahid 20 0 9.7G 294M 119M S 13.7 3.8 0:07.66 /snap/chromium/1298/usr/l
1692 zahid 20 0 4123M 228M 103M S 9.1 2.9 1:51.17 /usr/bin/gnome-shell
2377 zahid 20 0 6007M 522M 247M S 7.2 6.7 5:06.38 /snap/chromium/1298/usr/l
1718 zahid 9 -11 2327M 18532 14288 S 5.9 0.2 2:02.82 /usr/bin/pulseaudio --sta
2794 zahid 20 0 5854M 400M 114M S 5.9 5.1 3:32.82 /snap/chromium/1298/usr/l
2165 zahid 20 0 1566M 334M 242M S 4.6 4.3 2:04.27 /snap/chromium/1298/usr/l
1566 zahid 20 0 887M 61152 42052 S 3.9 0.8 1:13.60 /usr/lib/xorg/Xorg vt2 -d
1719 zahid -6 0 2327M 18532 14288 S 3.3 0.2 1:09.68 /usr/bin/pulseaudio --sta
2414 zahid 20 0 6007M 522M 247M S 2.6 6.7 1:08.11 /snap/chromium/1298/usr/l
2699 zahid 20 0 1613M 55308 38756 S 2.6 0.7 0:04.72 /usr/bin/nautilus --gappl
2011 zahid 20 0 3956M 594M 362M S 2.0 7.6 2:06.17 /snap/chromium/1298/usr/l
2429 zahid 20 0 1447M 55752 46084 S 2.0 0.7 0:43.04 /snap/chromium/1298/usr/l
2171 zahid 20 0 1272M 88780 64152 S 1.3 1.1 0:26.67 /snap/chromium/1298/usr/l
3900 zahid 20 0 34152 4940 3796 R 1.3 0.1 0:00.11 htop
2174 zahid 20 0 1272M 88780 64152 S 1.3 1.1 0:20.93 /snap/chromium/1298/usr/l
2436 zahid 20 0 1447M 55752 46084 S 1.3 0.7 0:39.08 /snap/chromium/1298/usr/l
3501 zahid 20 0 5640M 110M 85600 S 0.7 1.4 0:05.17 /snap/chromium/1298/usr/l
F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice -F8Nice +F9Kill F10Quit
```

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

```
ps -A
```

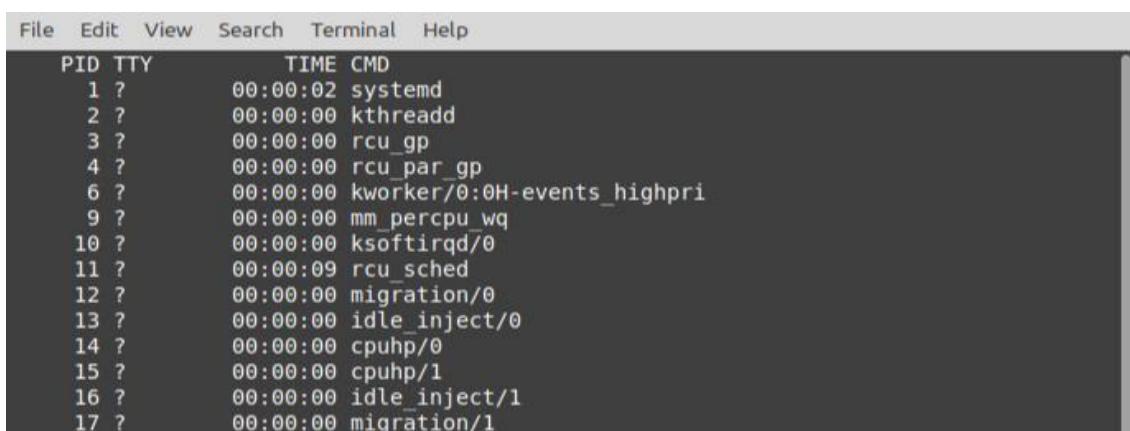


The screenshot shows a terminal window with the title "zahid@zahid: ~". The menu bar includes File, Edit, View, Search, Terminal, and Help. The terminal output is as follows:

```
File Edit View Search Terminal Help
ps-A: command not found
zahid@zahid:~$ ps
  PID TTY      TIME CMD
 3189 pts/0    00:00:00 bash
 3845 pts/0    00:00:00 top
 3900 pts/0    00:00:00 htop
 3921 pts/0    00:00:00 ps
zahid@zahid:~$ ps -A
  PID TTY      TIME CMD
    1 ?        00:00:05 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-kb
    9 ?        00:00:00 mm_percpu_wq
   10 ?       00:00:00 ksoftirqd/0
   11 ?       00:00:02 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
```

4) **ps -A | less**: **ps -A** 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:
```



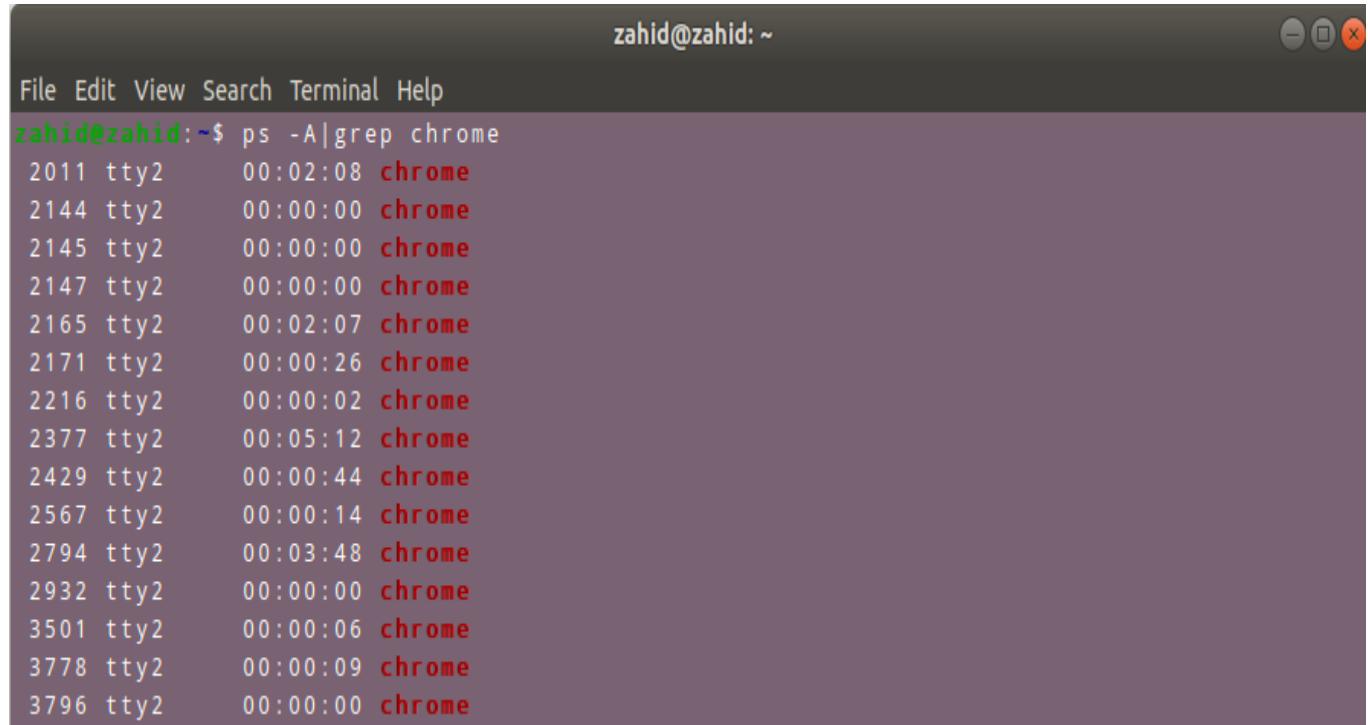
The screenshot shows a terminal window with the title "zahid@zahid: ~". The menu bar includes File, Edit, View, Search, Terminal, and Help. The terminal output is as follows:

```
File Edit View Search Terminal Help
  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
```



A screenshot of a terminal window titled "zahid@zahid: ~". The window has a standard Linux-style title bar with icons for minimize, maximize, and close. The menu bar includes "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal itself shows the command "ps -A | grep chrome" followed by a list of processes. The output is as follows:

```
zahid@zahid:~$ ps -A|grep chrome
2011  tty2      00:02:08  chrome
2144  tty2      00:00:00  chrome
2145  tty2      00:00:00  chrome
2147  tty2      00:00:00  chrome
2165  tty2      00:02:07  chrome
2171  tty2      00:00:26  chrome
2216  tty2      00:00:02  chrome
2377  tty2      00:05:12  chrome
2429  tty2      00:00:44  chrome
2567  tty2      00:00:14  chrome
2794  tty2      00:03:48  chrome
2932  tty2      00:00:00  chrome
3501  tty2      00:00:06  chrome
3778  tty2      00:00:09  chrome
3796  tty2      00:00:00  chrome
```

6) pstree:

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

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

```
zahid@zahid: ~
File Edit View Search Terminal Help
3712 ? 00:00:00 kworker/0:1-eve
3745 ? 00:00:00 kworker/7:1-mm_
3778 tty2 00:00:13 chrome
3796 tty2 00:00:00 chrome
3845 pts/0 00:00:00 top
3846 ? 00:00:00 kworker/u16:0-e
3854 ? 00:00:00 zeitgeist-daemo
3870 ? 00:00:00 zeitgeist-fts
3875 ? 00:00:00 gvfsd-metadata
3896 ? 00:00:00 kworker/2:0-eve
3897 ? 00:00:00 kworker/3:2
3900 pts/0 00:00:00 htop
3905 ? 00:00:00 kworker/6:1-eve
3906 ? 00:00:00 kworker/6:2
3935 ? 00:00:00 kworker/5:1-eve
3936 ? 00:00:00 kworker/0:0-eve
3937 ? 00:00:00 kworker/0:3-eve
3948 ? 00:00:00 kworker/5:3-eve
3998 pts/0 00:00:00 ps
zahid@zahid:~$ kill 3998
bash: kill: (3998) - No such process
zahid@zahid:~$ kill 3948
bash: kill: (3948) - Operation not permitted
zahid@zahid:~$ kill 3269
bash: kill: (3269) - Operation not permitted
zahid@zahid:~$ sudo kill 3269
[sudo] password for zahid:
zahid@zahid:~$ _
```

08) r enice:

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*

```
zahid@zahid:~$ renice 19 2011
2011 (process ID) old priority 0, new priority 19
zahid@zahid:~$ renice 19 $(pgrep chrome)
2011 (process ID) old priority 19, new priority 19
2144 (process ID) old priority 0, new priority 19
2145 (process ID) old priority 0, new priority 19
2147 (process ID) old priority 0, new priority 19
2165 (process ID) old priority 0, new priority 19
2171 (process ID) old priority 0, new priority 19
2216 (process ID) old priority 0, new priority 19
2377 (process ID) old priority 0, new priority 19
2429 (process ID) old priority 0, new priority 19
2567 (process ID) old priority 0, new priority 19
2794 (process ID) old priority 0, new priority 19
2932 (process ID) old priority 0, new priority 19
3501 (process ID) old priority 0, new priority 19
3778 (process ID) old priority 0, new priority 19
3796 (process ID) old priority 0, new priority 19
zahid@zahid:~$
```