

Process Management:

These commands are commonly used in Linux for process management.

1. `ps` Command:

Ps stands for process status

The `ps` command is used to display information about processes running on a system.

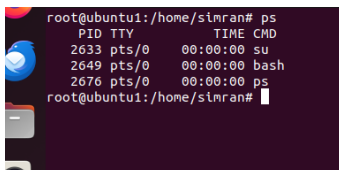
Examples:

- Display a list of all processes running in the current terminal session:

```
```bash
```

```
ps
```

```
```
```



```
root@ubuntu1:/home/simran# ps
  PID TTY          TIME CMD
 2633 pts/0    00:00:00 su
 2649 pts/0    00:00:00 bash
 2676 pts/0    00:00:00 ps
root@ubuntu1:/home/simran#
```

- Display detailed information about all processes running on the system:

```
```bash
```

```
ps aux
```

```
```
```

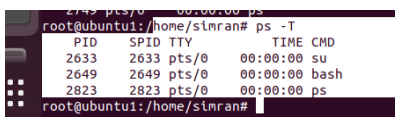
To see all running processes:

```
ps -A
```

```
ps -e
```

To see list of processes associated with the terminal only:

```
ps -T
```



```
2745 pts/0    00:00:00 ps
root@ubuntu1:/home/simran# ps -T
  PID  SPID TTY          TIME CMD
 2633   2633 pts/0    00:00:00 su
 2649   2649 pts/0    00:00:00 bash
 2823   2823 pts/0    00:00:00 ps
root@ubuntu1:/home/simran#
```

```
ps aux
```

```
2829 pts/0 00:00:00 ps
```

| USER | COMMAND | PID | %CPU | %MEM | VSZ | RSS | TTY | STAT | START | TIME |
|------|-------------------|-----|------|------|--------|-------|-----|------|-------|------|
| root | /sbin/init splash | 1 | 0.5 | 0.2 | 101248 | 11884 | ? | Ss | 12:27 | 0:01 |
| root | [kthreadd] | 2 | 0.0 | 0.0 | 0 | 0 | ? | S | 12:27 | 0:00 |
| root | [rcu_gp] | 3 | 0.0 | 0.0 | 0 | 0 | ? | I< | 12:27 | 0:00 |
| root | [rcu_par_gp] | 4 | 0.0 | 0.0 | 0 | 0 | ? | I< | 12:27 | 0:00 |
| root | [slub_flushwq] | 5 | 0.0 | 0.0 | 0 | 0 | ? | I< | 12:27 | 0:00 |
| root | [netns] | 6 | 0.0 | 0.0 | 0 | 0 | ? | I< | 12:27 | 0:00 |

VSZ- virtual memory size of process

RSS – physical memory size of process

TTY- terminal

STAT- Process state (Sleeping running and zombie)

TIME – minutes:seconds

- Show a tree view of processes with parent-child relationships:

```
```bash
```

```
ps auxf
```

```
```
```

Ps aux –sort=- %mem

2. `bg` Command:

The `bg` command is used to move a process to the background, allowing it to continue running while freeing up the terminal.

Example:

- Move a stopped or suspended process to the background:

```
```bash
```

```
Sleep 5
```

Press [Ctrl+Z] to stop the process

```
Jobs
```

It will show you stopped processes

Now we can send this process to background

```
bg %1
```

```
```
```

(Assuming the job ID is 1; replace it with the actual job ID.)

3. `fg` Command:

The `fg` command is used to bring a background process back to the foreground.

Example:

```
Sleep 5
```

Press [Ctrl+Z] to stop the process

```
Jobs
```

It will show you stopped processes

- Bring the most recent background process to the foreground:

```
``bash
```

```
fg %1
```

```
...
```

4. `nice` Command:

The `nice` command is used to launch a command with a specified priority (niceness), affecting its CPU scheduling priority.

Range for nice values : -20 to +19

Examples:

- Start a command with a higher priority (lower niceness value):

```
``bash
```

```
nice -n -10 command
```

```
...
```

- Start a command with a lower priority (higher niceness value):

```
``bash
```

```
nice -n 10 command
```

```
...
```

- Check the current niceness value of a running process:

```bash

nice -n

```

```
:00 ps
root@ubuntu1:/home/simran# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY T
IME CMD
4 S 0 2725 2711 0 80 0 - 3356 do_wai pts/0 00:00
:00 su
4 S 0 2726 2725 0 80 0 - 2623 do_wai pts/0 00:00
:00 bash
0 S 0 3234 2726 0 80 0 - 2092 hrtime pts/0 00:00
:00 sleep
4 R 0 3244 2726 0 80 0 - 3167 - pts/0 00:00
:00 ps
root@ubuntu1:/home/simran# nice -n +10 sleep 5m &
```

```
root@ubuntu1:/home/simran# nice -n +10 sleep 5m &
[2] 3246
root@ubuntu1:/home/simran# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY T
IME CMD
4 S 0 2725 2711 0 80 0 - 3356 do_wai pts/0 00:00
:00 su
4 S 0 2726 2725 0 80 0 - 2623 do_wai pts/0 00:00
:00 bash
4 S 0 3234 2726 0 80 0 - 2092 hrtime pts/0 00:00
:00 sleep
4 S 0 3246 2726 0 90 10 - 2092 hrtime pts/0 00:00
:00 sleep
4 R 0 3247 2726 0 80 0 - 3167 - pts/0 00:00
:00 ps
root@ubuntu1:/home/simran#
```

Lower the nice value higher will be the priority of that process

```
[3] 3267
root@ubuntu1:/home/simran# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY T
IME CMD
4 S 0 2725 2711 0 80 0 - 3356 do_wai pts/0 00:00
:00 su
4 S 0 2726 2725 0 80 0 - 2623 do_wai pts/0 00:00
:00 bash
0 S 0 3234 2726 0 80 0 - 2092 hrtime pts/0 00:00
:00 sleep
4 S 0 3246 2726 0 90 10 - 2092 hrtime pts/0 00:00
:00 sleep
4 S 0 3267 2726 0 61 -19 - 2092 hrtime pts/0 00:00
:00 sleep
4 R 0 3268 2726 0 80 0 - 3167 - pts/0 00:00
:00 ps
root@ubuntu1:/home/simran#
```

Now to change the priority of process which is already running

```
:00 sleep
4 S 0 3267 2726 0 61 -19 - 2092 hrtime pts/0 00:00
:00 sleep
4 R 0 3274 2726 0 80 0 - 3167 - pts/0 00:00
:00 ps
[1] Done sleep 5m
root@ubuntu1:/home/simran# renice +19 3267
3267 (process ID) old priority -19, new priority 19
root@ubuntu1:/home/simran#
```

Changed priority

```
3267 (process ID) old priority -19, new priority 19
root@ubuntu1:/home/simran# ps -l
F S UID PID PPID C PRI NI ADDR SZ WCHAN TTY T
IME CMD
4 S 0 2725 2711 0 80 0 - 3356 do_wai pts/0 00:00
:00 su
4 S 0 2726 2725 0 80 0 - 2623 do_wai pts/0 00:00
:00 bash
4 S 0 3267 2726 0 99 19 - 2092 hrtime pts/0 00:00
:00 sleep
4 R 0 3280 2726 0 80 0 - 3167 - pts/0 00:00
:00 ps
[2] Done nice -n +10 sleep 5m
```