ps

As we all know Linux is a multitasking and multi-user systems. So, it allows multiple processes to operate simultaneously without interfering with each other. Process is one of the important fundamental concept of the Linux OS. A process is an executing instance of a program and carry out different tasks within the operating system.

Linux provides us a utility called ps for viewing information related with the processes on a system which stands as abbreviation for “Process Status”. ps command is used to list the currently running processes and their PIDs along with some other information depends on different options. It reads the process information from the virtual files in /proc file-system. /proc contains virtual files, this is the reason it’s referred as a virtual file system.

ps provides numerous options for manipulating the output according to our need.

Syntax: ps [Options]

Result contains four columns of information.
Where,
PID – the unique process ID
TTY – terminal type that the user is logged into
TIME – amount of CPU in minutes and seconds that the process has been running
CMD – name of the command that launched the process.
Note: Time is nothing but the total accumulated CPU utilization time for any process and 00:00:00 indicates no CPU time has been given by the kernel till now.

Options:

1) -A / -e → To print system all process
2) -a → View all processes except both session leaders and processes not associated with a terminal.
3) -f → Provides a full-format listing, displaying more details.
4) -aux → Shows a detailed list of all processes in a user-friendly format. (ax for all , u for user friendly)
5) -d → all processes except session leader
6) -N → Negates the selection. Na → print both session leaders and processes not associated with a terminal.
7) -T → all processes associated with this terminal
8) -r → View all running processes
9) -x → Processes i.e same EUID as ps which means runner of the ps command
10) -F →Use -F to view Extra full format.
11) -l → long format

Process Selection:

1) Select the process by the command name. This selects the processes whose executable name is given in cmdlist. There may be a chance you won't know the process ID and with this command it is easier to search.
    a) Syntax : ps -C command_name
2) Select by group ID or name. The group ID identifies the group of the user who created the process. Syntax :
    a) ps -G group_name
    b) ps --Group group_name
3) View by group id. Syntax:
    a) ps -g group_id
    b) ps -group group_id
4) View process by process ID. Syntax :
    a) ps p process_id
    b) ps -p process_id
    c) ps --pid process_id

5) View all the processes belong to any session ID. Syntax:
    a) ps -s session_id
    b) ps --sid session_id
6) Select by tty. This selects the processes associated with the mentioned tty. Syntax :
    a) ps t tty
    b) ps -t tty
    c) ps --tty tty
7) Select by effective user ID or name. Syntax :
    a) ps U user_name/ID
    b) ps -U user_name/ID
    c) ps -u user_name/ID
    d) ps –User user_name/ID
    e) ps –user user_name/ID
8) To view process according to user-defined format. Syntax :
    a) ps --format column_name
    b) ps -o column_name
    c) ps o column_name
    d) Example : ps -aN --format cmd,pid,user,ppid

sleep

the sleep command will tell our computer to don't do anything for a while
and then start
taking commands or run the previous command
ex :- sleep
sleep 5
sleep 5m
sleep 5d

sleep 10 ; pwd

jobs

jobs command will give information about the jobs which are in background or foreground
+ -----> last job
- -----> second last job
& -----> indicates job that is running in background
options with jobs
-p -----> will also give process id of the job
-r -----> only running jobs
-s -----> only stopped jobs
-l -----> long form (it will also display process id)
ctrl+c ----> kill foreground job
ctrl+z -----> suspend foreground job

bg

bg command in linux is used to place foreground jobs in background.
Syntax: bg [job_spec..]

job_spec may be:

%n : Refer to job number n.
%str : Refer to a job which was started by a command beginning with str.
%?str : Refer to a job which was started by a command containing str.
%% or %+ : Refer to the current job. fg and bg will operate on this job if no job_spec is given.
%- : Refer to the previous job.

bg [JOB_SPEC] : This command is used to put the mentioned job in background. We do following :

1. We use jobs command to list all jobs
2. We create a process using sleep command, we get its ID as 1.
3. We put it in the background by providing its ID to bg.

fg

fg command in linux used to put a background job in foreground.
Syntax: fg [job_spec]
job_spec may be:

%n : Refer to job number n.
%str : Refer to a job which was started by a command beginning with str.
%?str : Refer to a job which was started by a command containing str.
%% or %+ : Refer to the current job. fg and bg will operate on this job if no job_spec is given.
%- : Refer to the previous job.

nice

At any given point, there are hundreds of processes running in a system, most of which are created by the Linux operating system and some created by the logged-in user. Each running process has a priority assigned to it that determines how fast it is executed by the system. Higher priority processes are usually carried out earlier than low priority ones.

In Linux, the nice & renice commands are used to change the priority of a process, which, in effect, determines the urgency with which it is executed in the system.

The nice command configures the priority of a Linux process before it is started. Once started, you cannot change the priority using the nice command. This is where the renice command comes in. The renice command sets the priority of an already running process.

The niceness value is denoted by the NI column header. The PRI column header denotes the actual priority of the process.
Syntax: nice -nice_value command-arguments
Example:

1) nice -5 wget https://wordpress.org/latest.zip
2) sudo nice --5 wget https://wordpress.org/latest.zip → Negative Value
3) nice -n 10 command_name
4) sudo renice -n 15 -p 77982 → changing priority of the running process.
5) renice -n 10 -g 4 → To change the priority of all programs of a specific group.
6) sudo renice -n 10 -u 2 → To change the priority of all programs of a specific user.