Chapter: 08 Processes

- Learning objectives

  - Describe what a process is and distinguish between types of process.
  - Enumerate process attributes.
  - manage process using ps and top.
  - understand the use of load averages and other Process metrics.
  - Manipulate process by putting them in background and restoring them to foreground.
  - use at, cron and sleep to schedule process in the future or Pause them.

* What is a process?

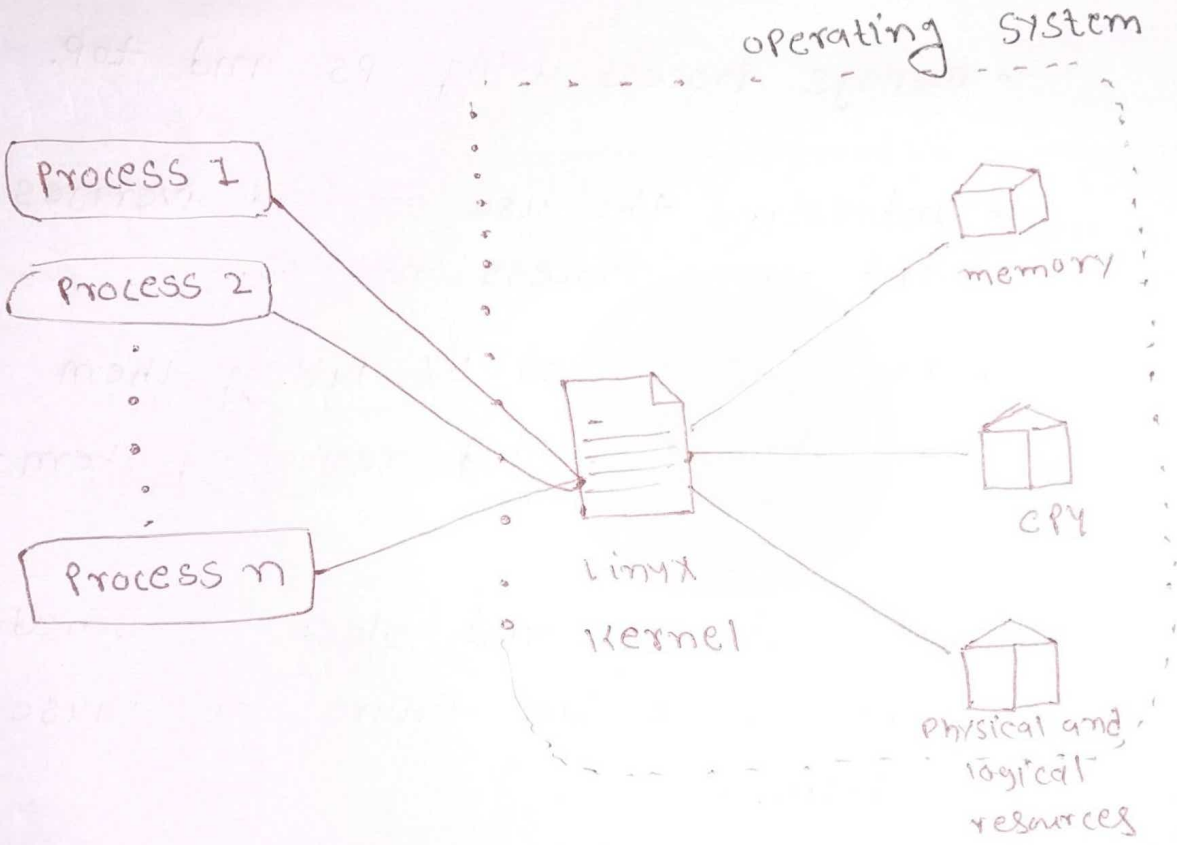
→ A process is simply an instance of one or more related tasks executing on your computer.

→ It is not same as program or command.

→ A single command may actually start several process.

→ Some processes are independent of each others are related.

→ A failure of one process may or may not affect the other running on the system.



Processes

* Process Types

Process Types	Description	Examples
Interactive Process	Need to be started by user, either at a command liner or through a graphical interface such as an icon or menu selection.	bash, firefox, top, Slack, Libreoffice
Batch Process	Automatic process which are scheduled from and then disconnected from the terminal.	Updatedb, idconfig
Daemons	Server process that run continuously. many are lanched during system startup and wait for user system request indicating their service is required	httpd, sshd, libvirtd, cupsd
Threads	Lightweight process. These are tasks that run under the umbrella of main process	dconf-service, gnome-terminal-server
Kernel Threads	Kernel task that users neither start or nor terminate and have little control over.	kthreadd, migration, ksoftirad

* Process and Thread IDs

→ At any given time, there are always multiple process being executed.

↔ The operating system keeps track of them by assigning each unique process ID (PID).

ID Type	Description
Process ID (PID)	Unique process ID number.
Parent process ID (PPID)	Process (Parant) that start this process. If the parant dies, the PPI will refer to an adoptive parant.
Thread ID (TID)	This is same as the PID for single threaded process.

* Terminating a process

→ At some point, one of your application may stop working properly.

→ To eleminate a process, you can use kill -SIGKILL or <Pid>

or kill -9 <pid>

NOTE:- however, you can kill your own processes those belonging to another user are off-limits unless you are root

* User and Group IDs

→ many users can access a system simultaneously, and each can run multiple process.

→ The operating system identifies the user who starts the process by Real User ID (RUID) assigned to the user.

→ The user who determines the access rights for the user is identified by effective UID (EUID).

→ The EUID may or may not be the same as RUID.

→ Users can be organized into enumerated groups. Each group is identified by Real Group ID (RGID). The access rights of group are determined by effective Group ID (EGID).

→ most of time we ignore these details and just talk about the User ID (UID) and Group ID (GID).