Modern operating systems are usually multitasking, meaning that they create the illusion of doing more than one thing at once by rapidly switching from one executing program to another. The Linux kernel manages this through the use of processes. Processes are how Linux organizes the different programs waiting for their turn at the CPU.

A process can launch another process - the new process is said to be a *child* process of the parent.

Each process is assigned an ID number so that the kernel can keep track of the process. (PID or Process ID). Init always get PID of 1. PIDs also have an associated UID.

- **ps** - list processes for the current user on the current terminal (columns)
- **ps tree** - lists the processes in a tree structure identifying parent processes
- **top** - dynamically lists the top processes running
- **jobs** - lists the jobs run from this terminal
- **bg** - put a program in the background
- **fg** - put a program in the foreground
- **kill** - kills a process based on the PID
- **killall** - kills a process based on the name

A process can be in one of several states:

- \( R \) = running
- \( S \) = Sleeping; process is waiting for an event to occur (keystroke)
- \( D \) = Uninterruptible cleep; process waiting for i/o
- \( T \) = stopped
- \( Z \) = zombie; child has been terminated but not cleaned up by parent

- **ctrl-c** kills a process
- **ctrl-z** puts a process in the background without killing it

**ps options**
- a - list all processes
- x - lift the BSD style
- u - list processes for all users
- o - customize the process list

![ps xao pid,ppid,comm]

**Killing**

- kill -9 pid # will kill a process ID
- killall xclock # will kill the xclock process

**shutdown options**

System Shutdown and Restart commands

```
shutdown -r time message
```

Example: sudo shutdown -r +5 “Server will restart in 5 minutes. Please save your work.” Example: sudo shutdown -h now

**shutdown options**

- halt - leaves the machine powered on

Example: sudo halt Example: sudo reboot