21/12/22

Processes in Linux

Processes are just programs that are running on our machine

→ These processes are generally managed by Kernel

& each & every process have a ID (PID)

to check processes running on your system
→ use command → ps
output (4 columns)

PID	TTY	Time	CMD
	→ controling terminal	→ CPU usage - time	→ command

(ps aux) → three flags a, u, x

a → all processes
u → more details about process
x → list down all proves even which don't require terminal
→ that don't have TTY associated

top → command to see realtime process, keeps fluctuating

Controlling Terminal

① TTY1 -> regular terminal
② Pts/1 -> pseudo terminal

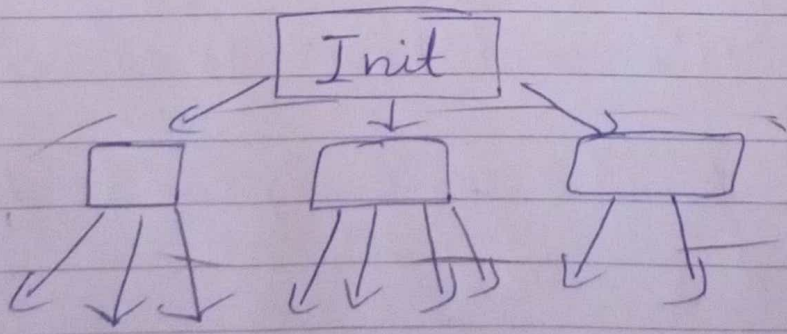
process also closes when terminal associated with it is closed

Who manages these processes
-> Kernel

Init -> It is the first process which is kernel when computer boots up & run until it shutdowns

This has a PID of 1 -> It is also known as mother process

All process are created by forking from init process (or it's children)
other processes are created by forking from parent process



These first children of init processes are known as daemon process

responsible for keeping the system running

These are forks of mother pro

Terminating a process

↓

_exit system call → (responsible for freeing up all resources process was utilizing

when a process terminates it tells kernel why

by termination status

→ most common status → 0 process successful

→ Parent has to acknowledge that their child died

→ by wait system call

Signals

→ It is a notification to the process that something has happened

Some common signals)

SIGHUP/HUP/1 → Hangup
SIGINT/INT/2 → Interupt
SIGKILL/KILL/9 → Kill
SIGSEGV/SEGV/11 → Segmentation Fault

→ Signal Mask is used to block signals but there are some which can't be blocked like KILL

States of Process

R → running
S → interruptable Sleep
D → uninterruptable sleep
Z → Zombie Process
T → Stopped

to run a program/process in background

python3 test.py &

this creates a job

& → used to run a program in bagroun