MONITOR AND MANAGE LINUX PROCESSES

DESCRIPTION	COMMANDS / OPTIONS
Process-related information including:	**Process-related information including:** PID: Process ID USER: Owner of the process PR: Priority NI: Nice value VIRT: Virtual memory usage RES: Resident set size (non-swapped physical memory used) SHR: Shared memory S: Process status (S: Sleeping, R: Running, I: Idle) %CPU: Percentage of CPU usage %MEM: Percentage of memory usage TIME+: Total CPU time COMMAND: Command or process name
Shows the state of each process	**ps [options]** -a List all ruining processes for all users -u Username -x View All Processes Owned By You -e Lists all processes on the entire system -O, --list-fields -ao list of fields to view **To see every process on the system** -ef To display all processe -eF -ely **To see every process on the system using BSD** ax aux **To print a process tree** -ejH -axjf
To find out the process IDs	**pidof [options] program1..** -s To display only one process ID of the program we have to pass -x To find the process ids of shells running the named script called bash -c pidof returns only process ids that are running with the same root directory,

DESCRIPTION	COMMANDS / OPTIONS
continuously refreshes at a configurable interval	**top[options]** -n exit top command after N th time -u Specific user process -d It tells delay time between screen updates -b Send output from top to file or any other programs -s Use top in Secure mode -c starts top with last closed state
To bring a background job to the foreground	Fg %<jobs num>
background jobs	bg %<jobs num>
To stop running process	Ctrl + z
To close running process	Ctrl + c
To display the list of jobs for the shell's session	**jobs[options]** -l Lists process IDs in addition to the normal information. -n List only processes that have changed status since the last notification. -p Lists process IDs only -r Restrict output to running jobs -s Restrict output to stopped jobs
Running in the foreground on the controlling Terminal	**sleep NUMBER[SUFFIX]** s This is used for specifying seconds. h This is used for specifying hours d This is used for specifying days
To send any signal, not just those signals for terminating programs.	**kill[options] PID** -l to list the names and numbers of all available signals. -9 Kill jobs by force
To signal one or more processes that match selection criteria.	pkill [options] pattern
To find the PID of a process	**pgrep [options] pattern** -l --To list the process names and IDs. -u --To specify the ID of the user who owns the processes
To view a process tree for the system or a single user	**pstree [options] [pid or username]** -a To include command line arguments in output -p To display PIDs -c To force pstree to expand identical subtrees in output -n To sort processes -u To see who is the owner/user of a process -h To highlight the current process or any other process -g To show process group IDs in output
This command sends a process/script/command to the background.	[commands] &
To display the current load average.	uptime

DESCRIPTION	COMMANDS / OPTIONS
To trace running process	**strace** -c To count number of system calls -e To trace particular or specific system calls -r To print timestamp of each call -T To print time spent on system calls -t To print wall clock time of each system call -i To print instruction pointer -o To print output to a file
Execution of a program/process with modified scheduling priority	**nice[options]** -n To set the priority(n) of a process
To change and modify the scheduling priority of an already running process	**renice[options]** -p Changing priority of the running process -g To change the priority of all programs of a specific group -u To change the priority of all programs of a specific user