

Task 3 Part 1

1. How many states could has a process in Linux?

Created, Ready, Waiting, Running, Terminated

2. Examine the pstree command. Make output (highlight) the chain (ancestors) of the current process.

```
root@master:/home/master# pstree
systemd--ModemManager--2*[{ModemManager}]
      --2*[agetty]
      --cron
      --dbus-daemon
      --login--bash
      --login--bash--sudo--sudo--su--bash
      --multipathd--6*[{multipathd}]
      --networkd-dispat
      --packagekitd--2*[{packagekitd}]
      --polkitd--2*[{polkitd}]
      --rsyslogd--3*[{rsyslogd}]
      --snapd--8*[{snapd}]
      --sshd--sshd--sshd--bash--sudo--sudo--su--bash--pstree
            |      |      |
            |      |      --bash--sleep
            |      --sshd--sshd--sftp-server
      --sudo--sudo--su--bash--more
      --systemd--(sd-pam)
      --systemd-journal
      --systemd-logind
      --systemd-network
      --systemd-resolve
      --systemd-timesyn--{systemd-timesyn}
      --systemd-udev
      --udisksd--4*[{udisksd}]
      --unattended-upgr--{unattended-upgr}
root@master:/home/master#
```

3. What is a proc file system?

/proc filesystem include:

Virtual Files: The files and directories under /proc are not actual files on disk but are rather virtual files that expose information from the kernel.

Process Information: /proc provides information about running processes. Each process has a corresponding directory with its PID (Process ID) as the directory name.

Kernel Information: /proc also exposes various information about the kernel itself. This includes information about system configuration, hardware details, and runtime parameters.

Dynamic Updates: The information exposed by /proc is dynamic and can change as the system and processes run. Reading from /proc files provides a real-time snapshot of the system's state.

4. Print information about the processor (its type, supported technologies, etc.).

```
root@master:/home/master# cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 142
model name     : Intel(R) Core(TM) i7-8650U CPU @ 1.90GHz
stepping       : 10
cpu MHz        : 2112.002
cache size     : 8192 KB
physical id    : 0
siblings       : 1
core id        : 0
cpu cores      : 1
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 22
wp             : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge
rep_good nopl xtopology nonstop_tsc cpuid tsc_known_freq pni pclmulqdq
ervisor lahf_lm abm 3dnowprefetch invpcid_single pti fsgsbase bmi1 avx
bugs           : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass
bogomips       : 4224.00
clflush size   : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

root@master:/home/master#
```


5. Use the `ps` command to get information about the process. The information should be as follows: the owner of the process, the arguments with which the process was launched for execution, the group owner of this process, etc.

```

sion Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
pick connect...
2. 192.168.1.11 (master)
root@master:/home/master# ps -aux | grep 237251
root      237251  0.0  0.1  8836  5728 pts/3    S   07:13   0:00 bash
root      308351  0.0  0.0   6476   2440 pts/3    S+  10:43   0:00 grep --color=auto 237251
```

6. How to define kernel processes and user processes?

Kernel processes:

```

pick connect...
2. 192.168.1.11 (master)
root@master:/home/master# ps aux | grep '\[
root      2  0.0  0.0      0      0 ?        S   Aug16   0:00 [kthreadd]
root      3  0.0  0.0      0      0 ?        I<  Aug16   0:00 [rcu_gp]
root      4  0.0  0.0      0      0 ?        I<  Aug16   0:00 [rcu_par_gp]
root      5  0.0  0.0      0      0 ?        I<  Aug16   0:00 [slub_flushwq]
root      6  0.0  0.0      0      0 ?        I<  Aug16   0:00 [netns]
root      8  0.0  0.0      0      0 ?        I<  Aug16   0:00 [kworker/0:0H-events_highpri]
root     10  0.0  0.0      0      0 ?        I<  Aug16   0:00 [mm_percpu_wq]
root     11  0.0  0.0      0      0 ?        S   Aug16   0:00 [rcu_tasks_rude_]
root     12  0.0  0.0      0      0 ?        S   Aug16   0:00 [rcu_tasks_trace]
root     13  0.0  0.0      0      0 ?        S   Aug16   0:09 [ksoftirqd/0]
root     14  0.0  0.0      0      0 ?        I   Aug16   0:19 [rcu_sched]
root     15  0.0  0.0      0      0 ?        S   Aug16   0:00 [migration/0]
root     16  0.0  0.0      0      0 ?        S   Aug16   0:00 [idle_inject/0]
root     18  0.0  0.0      0      0 ?        S   Aug16   0:00 [cpuhp/0]
```

User processes:

```
root@master:/home/master# ps -u
```

| USER | PID | %CPU | %MEM | VSZ | RSS | TTY | STAT | START | TIME | COMMAND |
|------|--------|------|------|-------|------|-------|------|-------|------|---|
| root | 879 | 0.0 | 0.1 | 7836 | 4720 | tty1 | Ss | Aug16 | 0:00 | /bin/login -p -- |
| root | 199739 | 0.0 | 0.1 | 7836 | 4824 | tty6 | Ss | 02:56 | 0:00 | /bin/login -p -- |
| root | 199955 | 0.0 | 0.0 | 6172 | 1092 | tty5 | Ss+ | 02:56 | 0:00 | /sbin/agetty -o -p -- \u --noclear tty5 linux |
| root | 200597 | 0.0 | 0.1 | 11668 | 5468 | tty6 | S+ | 02:58 | 0:00 | sudo su |
| root | 200616 | 0.0 | 0.0 | 11668 | 948 | pts/4 | Ss | 02:58 | 0:00 | sudo su |
| root | 200617 | 0.0 | 0.1 | 10192 | 4424 | pts/4 | S | 02:58 | 0:00 | su |
| root | 200618 | 0.0 | 0.1 | 7632 | 4276 | pts/4 | S+ | 02:58 | 0:00 | bash |
| root | 200674 | 0.0 | 0.0 | 6172 | 1124 | tty4 | Ss+ | 02:58 | 0:00 | /sbin/agetty -o -p -- \u --noclear tty4 linux |
| root | 237233 | 0.0 | 0.1 | 11664 | 5776 | pts/2 | S+ | 07:13 | 0:00 | sudo su |
| root | 237249 | 0.0 | 0.0 | 11664 | 2484 | pts/3 | Ss | 07:13 | 0:00 | sudo su |
| root | 237250 | 0.0 | 0.1 | 10192 | 4232 | pts/3 | S | 07:13 | 0:00 | su |
| root | 237251 | 0.0 | 0.1 | 8836 | 5728 | pts/3 | S | 07:13 | 0:00 | bash |
| root | 314455 | 0.0 | 0.0 | 10068 | 1560 | pts/3 | R+ | 11:01 | 0:00 | ps -u |

```
root@master:/home/master#
```

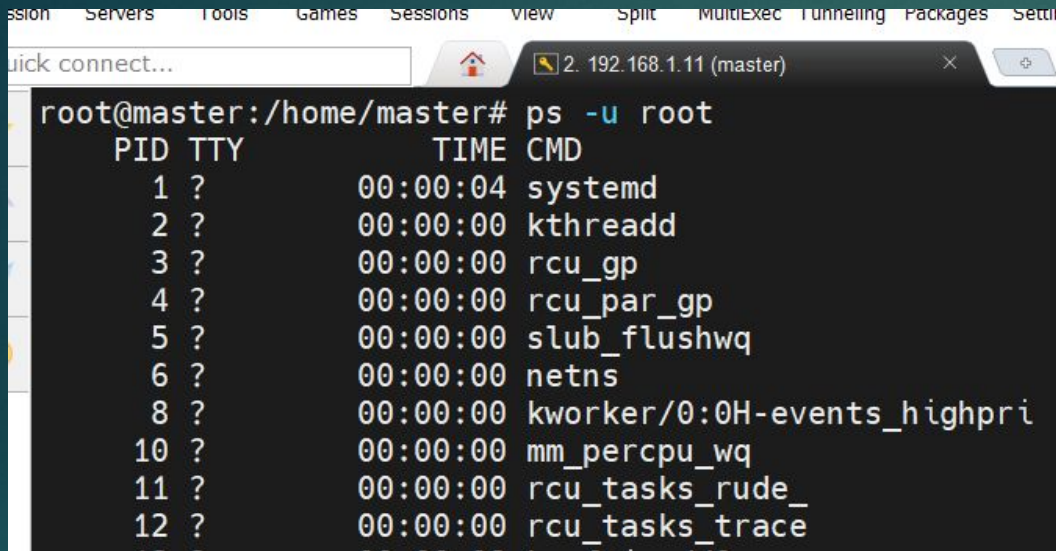

7. Print the list of processes to the terminal. Briefly describe the statuses of the processes. What condition are they in, or can they be arriving in?

```
root@master:/home/master# ps aux
```

| USER | PID | %CPU | %MEM | VSZ | RSS | TTY | STAT | START | TIME | COMMAND |
|------|-----|------|------|--------|-------|-----|------|-------|------|--------------------|
| root | 1 | 0.0 | 0.3 | 102012 | 13168 | ? | Ss | Aug16 | 0:04 | /sbin/init |
| root | 2 | 0.0 | 0.0 | 0 | 0 | ? | S | Aug16 | 0:00 | [kthreadd] |
| root | 3 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [rcu_gp] |
| root | 4 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [rcu_par_gp] |
| root | 5 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [slub_flushwq] |
| root | 6 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [netns] |
| root | 8 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [kworker/0:0H-eva] |
| root | 10 | 0.0 | 0.0 | 0 | 0 | ? | I< | Aug16 | 0:00 | [mm_percpu_wq] |
| root | 11 | 0.0 | 0.0 | 0 | 0 | ? | S | Aug16 | 0:00 | [rcu_tasks_rude_] |
| root | 12 | 0.0 | 0.0 | 0 | 0 | ? | S | Aug16 | 0:00 | [rcu_tasks_trace] |
| root | 13 | 0.0 | 0.0 | 0 | 0 | ? | S | Aug16 | 0:09 | [ksoftirqd/0] |

R(Running), S(Slipping), Z(Zombie), T(Stopped), W(Waiting), X(Dead)

8. Display only the processes of a specific user.



```
root@master:/home/master# ps -u root
  PID TTY          TIME CMD
    1 ?           00:00:04 systemd
    2 ?           00:00:00 kthreadd
    3 ?           00:00:00 rcu_gp
    4 ?           00:00:00 rcu_par_gp
    5 ?           00:00:00 slub_flushwq
    6 ?           00:00:00 netns
    8 ?           00:00:00 kworker/0:0H-events_highpri
   10 ?           00:00:00 mm_percpu_wq
   11 ?           00:00:00 rcu_tasks_rude_
   12 ?           00:00:00 rcu_tasks_trace
```

9. What utilities can be used to analyze existing running tasks (by analyzing the help for the ps command)?

```
SEE ALSO
    pgrep(1), pstree(1), top(1), proc(5).
```


10. What information does top command display?

```
top - 11:48:47 up 18:50, 6 users, load average: 1.00, 1.04, 1.00
Tasks: 123 total, 3 running, 120 sleeping, 0 stopped, 0 zombie
%Cpu(s): 47.1 us, 52.9 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3912.3 total, 2700.2 free, 242.5 used, 969.7 buff/cache
MiB Swap: 3925.0 total, 3925.0 free, 0.0 used. 3413.1 avail Mem
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ | COMMAND |
|--------|--------|----|-----|--------|-------|------|---|------|------|-----------|----------|
| 117730 | root | 20 | 0 | 6008 | 1320 | 1208 | R | 99.3 | 0.0 | 716:24.16 | more |
| 237157 | master | 20 | 0 | 17460 | 8576 | 5928 | S | 0.3 | 0.2 | 0:14.65 | sshd |
| 237225 | master | 20 | 0 | 7368 | 3580 | 3320 | S | 0.3 | 0.1 | 0:22.85 | bash |
| 1 | root | 20 | 0 | 102012 | 13168 | 8436 | S | 0.0 | 0.3 | 0:04.08 | systemd |
| 2 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.01 | kthreadd |
| 3 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | rcu_gp |

System info: Load average, CPU usage, Memory usage, SWAP usage
Process list: Process ID ,User who owns the process, Command and arguments used to launch the process, Indicates the process status, Priority, Nice value, Threads

Interactive Features: Real-time updates, Interactive commands, Sorting, Total processes, Running processes, Sleeping processes, CPU usage summary

Global Statistics: Total processes, Running processes, Sleeping processes, CPU usage summary

12. Display the processes of the specific user using the top command.

```
top - 11:58:08 up 19:00, 6 users, load average: 1.01, 1.03, 1.00
Tasks: 123 total, 2 running, 121 sleeping, 0 stopped, 0 zombie
%Cpu(s): 52.7 us, 47.3 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3912.3 total, 2700.2 free, 242.5 used, 969.7 buff/cache
MiB Swap: 3925.0 total, 3925.0 free, 0.0 used. 3413.1 avail Mem
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ | COMMAND |
|--------|--------|----|----|--------|------|------|---|------|------|---------|-------------|
| 1534 | master | 20 | 0 | 17032 | 9728 | 8072 | S | 0.0 | 0.2 | 0:00.13 | systemd |
| 1535 | master | 20 | 0 | 104796 | 4864 | 4 | S | 0.0 | 0.1 | 0:00.00 | (sd-pam) |
| 1542 | master | 20 | 0 | 8744 | 5428 | 3672 | S | 0.0 | 0.1 | 0:00.04 | bash |
| 199843 | master | 20 | 0 | 8740 | 5476 | 3692 | S | 0.0 | 0.1 | 0:00.02 | bash |
| 237157 | master | 20 | 0 | 17460 | 8576 | 5928 | S | 0.0 | 0.2 | 0:15.64 | sshd |
| 237194 | master | 20 | 0 | 17308 | 8020 | 5616 | S | 0.0 | 0.2 | 0:00.00 | sshd |
| 237195 | master | 20 | 0 | 7764 | 5392 | 4312 | S | 0.0 | 0.1 | 0:00.00 | sftp-server |
| 237214 | master | 20 | 0 | 8732 | 5420 | 3672 | S | 0.0 | 0.1 | 0:00.07 | bash |
| 237225 | master | 20 | 0 | 7368 | 3580 | 3320 | S | 0.0 | 0.1 | 0:23.53 | bash |
| 333226 | master | 20 | 0 | 5768 | 1008 | 920 | S | 0.0 | 0.0 | 0:00.00 | sleep |

12. What interactive commands can be used to control the top command? Give a couple of examples.

Sorting:

Press M to sort the process list by memory usage.

Press P to sort the process list by CPU usage.

Navigating Process List:

Use the arrow keys (up and down) to navigate through the process list.

Press Home to jump to the top of the process list.

Press End to jump to the bottom of the process list.

Exiting:

Press q to exit the top command.

13. Sort the contents of the processes window using various parameters (for example, the amount of processor time taken up, etc.)

I use P for sort by cpu utilization.

```
ck connect... 2. 192.168.1.11 (master)
top - 12:04:26 up 19:06, 6 users, load average: 1.10, 1.08, 1.02
Tasks: 123 total, 2 running, 121 sleeping, 0 stopped, 0 zombie
%Cpu(s): 52.0 us, 48.0 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3912.3 total, 2700.2 free, 242.5 used, 969.7 buff/cache
MiB Swap: 3925.0 total, 3925.0 free, 0.0 used. 3413.1 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 117730 root        20   0   6008    1320   1208 R   99.0   0.0 731:52.75 more
 237157 master      20   0   17460    8576   5928 S    0.3   0.2  0:16.07 sshd
      1 root        20   0  102012   13168   8436 S    0.0   0.3  0:04.10 systemd
      2 root        20   0         0         0        0 S    0.0   0.0  0:00.01 kthreadd
      3 root         0 -20         0         0        0 I    0.0   0.0  0:00.00 rcu_gp
      4 root         0 -20         0         0        0 I    0.0   0.0  0:00.00 rcu_par_gp
      5 root         0 -20         0         0        0 I    0.0   0.0  0:00.00 slub_flushwq
```


14. Concept of priority, what commands are used to set priority?

Priority levels are usually represented by numeric values, where lower values indicate higher priority.

You can use nice command:

`nice -n -5 my_command` or `renice -n priority -p process_id`

15. Can I change the priority of a process using the top command? If so, how?

```
top - 12:09:39 up 19:11, 6 users, load average: 1.03, 1.05, 1.00
Tasks: 123 total, 2 running, 121 sleeping, 0 stopped, 0 zombie
%Cpu(s): 54.8 us, 45.2 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3912.3 total, 2700.2 free, 242.5 used, 969.7 buff/cache
MiB Swap: 3925.0 total, 3925.0 free, 0.0 used, 3413.1 avail Mem
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ | COMMAND |
|--------|------|----|-----|--------|-------|------|---|------|------|-----------|---------------|
| 117730 | root | 20 | 0 | 6008 | 1320 | 1208 | R | 98.7 | 0.0 | 737:02.48 | more |
| 237233 | root | 20 | 0 | 11664 | 5776 | 4920 | S | 0.3 | 0.1 | 0:00.96 | sudo |
| 335261 | root | 20 | 0 | 10488 | 3968 | 3392 | R | 0.3 | 0.1 | 0:00.38 | top |
| 1 | root | 20 | 0 | 102012 | 13168 | 8436 | S | 0.0 | 0.3 | 0:04.10 | systemd |
| 2 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.01 | kthreadd |
| 3 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | rcu_gp |
| 4 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | rcu_par_gp |
| 5 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | slub_flushwq |
| 6 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | netns |
| 8 | root | 0 | -20 | 0 | 0 | 0 | I | 0.0 | 0.0 | 0:00.00 | kworker/0:0H- |
| 10 | root | 0 | -20 | 0 | 0 | 0 | T | 0.0 | 0.0 | 0:00.00 | mm_percpu_wq |

I use mobaXterm for this lab.

Run the top command in your terminal:

In the top interface, find the process for which you want to change the priority.

Select the process by moving the cursor to it using the arrow keys.

Once the process is selected, press the r key. This will prompt you to enter the new "renice" value (priority) for the process.

Enter the new priority value. Remember that lower values indicate higher priority.

Press Enter to confirm.

16. Examine the kill command. How to send with the kill command process control signal? Give an example of commonly used signals.

```
ick connect... 2. 192.168.1.11 (master) X
root@master:/home/master# vimtutor
[1]+  Stopped                  vimtutor
root@master:/home/master# ps -aux | grep vimtutor
root      338574  0.0  0.0   2888   944 pts/3    T   12:14   0:00 /bin/sh /usr/bin/vimtutor
root      338663  0.0  0.0   6608  2232 pts/3    S+  12:14   0:00 grep --color=auto vimtutor
root@master:/home/master# kill -SIGKILL 338574
Vim: Caught deadly signal HUP
Vim: Finished.
```


17. Commands jobs, fg, bg, nohup. What are they for? Use the sleep, yes command to demonstrate the process control mechanism with fg, bg.

jobs - This command lists the currently running and suspended background jobs associated with the current shell session.

fg - The fg command brings a background job to the foreground, allowing you to interact with it in the terminal.

bg - The bg command resumes a suspended background job, allowing it to continue running in the background.

nohup - The nohup command is used to run a command immune to hangups, meaning the process will continue to run even if the terminal is closed.

```
Quick connect... 2. 192.168.1.11 (master)
root@master:/home/master# yes > /dev/null &
[1] 341587
root@master:/home/master# ps -axu | grep 341587
root    341587  50.5  0.0   5764   1052 pts/3    R   12:22   0:06 yes
root    341655   0.0  0.0   6476   2208 pts/3    S+  12:23   0:00 grep --color=auto 341587
root@master:/home/master#
```

```
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages
Quick connect... 2. 192.168.1.11 (master)
root@master:/home/master# jobs
[1]+  Running                  yes > /dev/null &
root@master:/home/master#
```

```
Session Servers Tools Games Sessions View Split MultiExec T
Quick connect... 2. 192.168.1.11 (master)
root@master:/home/master# fg %1
yes > /dev/null
```

```
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 2. 192.168.1.11 (master)
root@master:/home/master# nohup sleep 300 &
[1] 342963
root@master:/home/master# nohup: ignoring input and appending output to 'nohup.out'

root@master:/home/master# ps -axu | grep 342963
root    342963   0.0  0.0   5768   1044 pts/3    S   12:27   0:00 sleep 300
root    343133   0.0  0.0   6476   2292 pts/3    S+  12:27   0:00 grep --color=auto 342963
root@master:/home/master#
```