Listing Processes_ps_and_top

The ps Command (System V Style):

- 1. **ps** provides information about currently running processes keyed by **PID**.
- 2. If you want a repetitive update of this status, you can use top or other commonly installed variants (such as **htop** or **atop**) from the command line.
- 3. **ps** has many options for specifying exactly which tasks to examine. Without options, **ps** will display all processes running under the current shell.
- 4. You can use the **-u** option to display information of processes for a specified username.
- 5. The command **ps -ef** displays all the processes in the system in full detail.
- 6. The command **ps -eLf** goes one step further and displays one line of information for every thread (remember, a process can contain multiple threads).

```
🔵 🗇 🕦 student@ubuntu: ~
student@ubuntu:~$ ps -ef
                      PPID C STIME TTY
0 2 16:07 ?
UID
                                                   00:00:02 /sbin/init auto noprompt
00:00:00 [kthreadd]
00:00:00 [ksoftirqd/0]
00:00:00 [kworker/0:0]
root
                             0 16:07
root
                          2
                             0 16:07
root
root
                              0 16:07
                                                   00:00:00 [kworker/0:0H]
00:00:00 [kworker/u256:0]
root
                              0 16:07
root
                              0
                                 16:07
                              0 16:07
                                                    00:00:00 [rcu_sched]
root
                                                   00:00:00 [rcu_bh]
00:00:00 [migration/0]
00:00:00 [watchdog/0]
00:00:00 [watchdog/1]
root
                 8
                              0 16:07
                 9
                              0 16:07
root
                10
                             0 16:07
root
root
                11
                             0 16:07
                                                    00:00:00 [migration/1]
00:00:00 [ksoftirqd/1]
root
                12
                              0 16:07
root
                13
                              0
                                 16:07
student
              3669
                       3084
                                                    00:00:00 /usr/lib/gnome-terminal/gnome-terminal-server
                              0 16:08 ?
                              0 16:08 pts/18
student
              3675
                       3669
                                                   00:00:00 bash
student
              3701
                      3324
                              0 16:08
                                                    00:00:00 zeitgeist-datahub
              3712
                                                   00:00:00 /usr/bin/zeitgeist-daemon
                      3708
student
                              0 16:08
                                                   00:00:00 /usr/lib/x86_64-linux-gnu/zeitgeist-fts
student
              3719
                      3084
                              0 16:08 ?
                                                   00:00:00 update-notifier
student
              3785
                      3324
                             0 16:09
root
              3792
                          1 22 16:09
                                                    00:00:02 /usr/bin/python3 /usr/sbin/aptd
                       3675 0 16:09 pts/18
student
              4074
                                                   00:00:00 ps -ef
student@ubuntu:~$
```

The ps Command (BSD Style):

- 1. The command **ps aux** displays all processes of all users.
- 2. The command **ps axo** allows you to specify which attributes you want to view.
- 3. The below screenshot shows a sample output of **ps** with the **aux** and **axo** qualifiers.

```
File Edit View Search Terminal Help
                              -10
c7:/home/coop>ps aux | head
USER
            PID %CPU %MEM
                              VSZ
                                     RSS TTY
                                                   STAT START
                                                                  TIME COMMAND
                                                                 0:01 /usr/lib/systemd/sys
                      0.0
                                                   Ss
                           125680
                                    5544
                                                         07:34
root
                 0.0
              1
                 0.0
                      0.0
                                 0
                                       0
                                                         07:34
                                                                 0:00
                                                                      [kthreadd]
root
              3
                                       0 ?
                                                   S
                                                                 0:00
                 0.0
                      0.0
                                0
                                                         07:34
                                                                      [ksoftirqd/0]
root
              5
                 0.0
                       0.0
                                0
                                       0
                                                   S<
                                                         07:34
                                                                 0:00
                                                                      [kworker/0:0H]
root
                                                         07:34
                                                                 0:02 [rcu_preempt]
                                0
                                       0
                                                   S
root
                 0.0
                      0.0
              8
                                                         07:34
                                                                 0:00 [rcu sched]
root
                 0.0
                      0.0
                                0
                                       0
                                                   S
              9
                                                   S
root
                 0.0
                      0.0
                                0
                                       0
                                                         07:34
                                                                 0:00 [rcu bh]
             10
                 0.0
                      0.0
                                0
                                       0
                                                   S
                                                         07:34
                                                                 0:00 [rcuop/0]
root
root
             11
                 0.0
                       0.0
                                0
                                       0
                                                   S
                                                         07:34
                                                                 0:00 [rcuos/0]
c7:/home/coop>ps axo stat,priority,pid,pcpu,comm | head -10
STAT PRI
            PID %CPU COMMAND
Ss
      20
              1
                 0.0 systemd
s
s
      20
              2
                 0.0 kthreadd
      20
              3
                 0.0 ksoftirqd/0
S<
S
S
       0
              5
                 0.0 kworker/0:0H
              7
      20
                 0.0 rcu preempt
      20
              8
                 0.0 rcu_sched
      20
              9
                 0.0 rcu bh
s
      20
             10
                 0.0 rcuop/0
      20
             11
                 0.0 rcuos/0
c7:/home/coop>
```

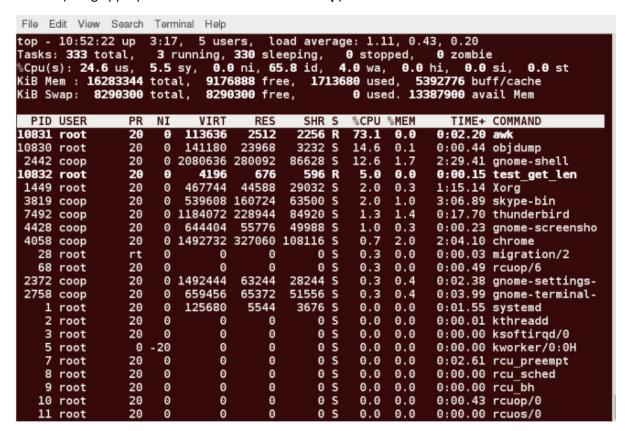
The Process Tree:

- pstree displays the processes running on the system in the form of a tree diagram showing
 the relationship between a process and its parent process and any other processes that it
 created.
- 2. Repeated entries of a process are not displayed, and threads are displayed in curly braces.

```
student@ubuntu:~/Pictures$ pstree
svstemd-
          -ModemManager-
                           {adbus}
                           (gmain)
          -NetworkManager-
                             dhclient
                             dnsmasq
                              {adbus}
                             {qmain}
          -accounts-daemon-
                               {gdbus}
                              {gmain}
          acpid
          agetty
          apache2---2*[apache2---26*[{apache2}]]
          avahi-daemon—avahi-daemon
          cgmanager
          collectl
          -colord-
                    -{gdbus}
                     {gmain}
          CLOU
          cups-browsed-
                          -{gdbus}
                           (qmain)
```

top:

- 1. While a static view of what the system is doing is useful, monitoring the system performance live over time is also valuable. One option would be to run **ps** at regular intervals, say, every few seconds.
- 2. A better alternative is to use **top** to get constant real-time updates (every two seconds by default), until you exit by typing **q**.
- 3. **top** clearly highlights which processes are consuming the most CPU cycles and memory (using appropriate commands from within **top**).



top

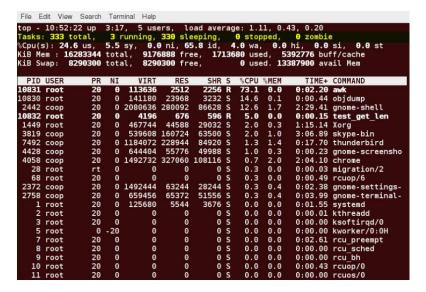
First Line of the top Output:

- 1. The first line of the top output displays a quick summary of what is happening in the system, including:
 - How long the system has been up
 - How many users are logged on
 - What is the load average
- 2. The load average determines how busy the system is.
- 3. A load average of 1.00 per CPU indicates a fully subscribed, but not overloaded, system.
- 4. If the load average goes above this value, it indicates that processes are competing for CPU time.
- 5. If the load average is very high, it might indicate that the system is having a problem, such as a runaway process (a process in a non-responding state).

File E	dit Vi	ew Search	Terr	minal Help						
top -	10:5	2:22 up	3:17	7, 5 use	ers, lo	oad aver	ag	e: 1.1	1, 0.	43, 0.20
		total,								
										hi, 0.0 si, 0.0 st
							L36			392776 buff/cache
KiB S	vap:	8290300	tota	al, 829 0	9300 fre	ee,		0 use	d. 1 3	387900 avail Mem
				"		101				
The second second	USER		NI	VIRT	RES	SHR		%CPU		TIME+ COMMAND
10831			θ	and the last of th	2512	2256		73.1	0.0	0:02.20 awk
10830			0	141180	23968	3232		14.6	0.1	0:00.44 objdump
2442			0			86628		12.6	1.7	2:29.41 gnome-shell
10832			θ		676	596		5.θ	0.0	0:00.15 test_get_len
1449			0	The second second	44588	29032		2.0	0.3	1:15.14 Xorg
3819			0			63500		2.0	1.0	3:06.89 skype-bin
7492			0	1184072		84920		1.3	1.4	0:17.70 thunderbird
4428			0	644404	55776	49988	S	1.0	0.3	0:00.23 gnome-screensho
4058	coop	20	0	1492732	327060	108116	S	0.7	2.0	2:04.10 chrome
28	root	rt	0	Θ	0	0	S	0.3	0.0	0:00.03 migration/2
68	root	20	0	0	0	0	S	0.3	0.0	0:00.49 rcuop/6
2372	coop	20	0	1492444	63244	28244	S	0.3	0.4	0:02.38 gnome-settings-
2758	coop	20	0	659456	65372	51556	S	0.3	0.4	0:03.99 gnome-terminal-
1	root	20	0	125680	5544	3676	S	0.0	0.0	0:01.55 systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01 kthreadd
3	root	20	0	Θ	0	0	S	0.0	0.0	0:00.00 ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00 kworker/0:0H
7	root	20	0	0	0	0	S	0.0	0.0	0:02.61 rcu_preempt
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00 rcu_sched
9	root	20	0	0	0	0	S	0.0	0.0	0:00.00 rcu_bh
10	root	20	0	Θ	0	0	S	0.0	0.0	0:00.43 rcuop/0
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00 rcuos/0

Second Line of the top Output:

- 1. The second line of the top output displays the total number of processes, the number of running, sleeping, stopped, and zombie processes.
- 2. Comparing the number of running processes with the load average helps determine if the system has reached its capacity or perhaps a particular user is running too many processes.
- 3. The stopped processes should be examined to see if everything is running correctly.



Third Line of the top Output:

- 1. The third line of the top output indicates how the CPU time is being divided between the users (us) and the kernel (sy) by displaying the percentage of CPU time used for each.
- 2. The percentage of user jobs running at a lower priority (niceness ni) is then listed.
- 3. Idle mode (id) should be low if the load average is high, and vice versa.
- 4. The percentage of jobs waiting (wa) for I/O is listed.
- 5. Interrupts include the percentage of hardware (hi) vs. software interrupts (si). Steal time (st) is generally used with virtual machines, which has some of its idle CPU time taken for other uses.

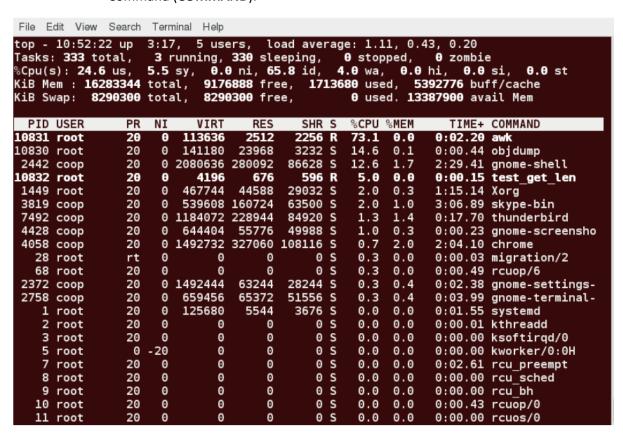
Fourth and Fifth Lines of the top Output:

- 1. The fourth and fifth lines of the top output indicate memory usage, which is divided in two categories:
 - Physical memory (RAM) displayed on line 4.
 - Swap space displayed on line 5.
- 2. Both categories display total memory, used memory, and free space.
- 3. You need to monitor memory usage very carefully to ensure good system performance.
- 4. Once the physical memory is exhausted, the system starts using swap space (temporary storage space on the hard drive) as an extended memory pool, and since accessing disk is much slower than accessing memory, this will negatively affect system performance.

File E	dit Vie	ew Search	Tern	minal Help						
										43, 0.20
		total,						0 stop		0 zombie
				sy, θ.				.θ wa,		hi, 0.0 si, 0.0 st
							.36			392776 buff/cache
KiB S	vap:	8290300	tota	al, 829 0	3500 TF	e,		o use	u. 13	387900 avail Mem
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	6MEM	TIME+ COMMAND
10831	root	20	θ	113636	2512	2256	R	73.1	0.0	0:02.20 awk
10830		20	0	141180	23968	3232	S	14.6	0.1	0:00.44 objdump
2442	соор	20	0	2080636	280092	86628	S	12.6	1.7	2:29.41 gnome-shell
10832	Contract to the Contract of th	20	θ	4196	676	596		5.θ	0.0	θ:θθ.15 test_get_len
1449		20	0	467744	44588	29032		2.0	0.3	1:15.14 Xorg
3819		20	0	539608	160724	63500		2.0	1.0	3:06.89 skype-bin
	coop	20	0	1184072	The second second second	84920	S	1.3	1.4	0:17.70 thunderbird
	соор	20	0	644404	55776	49988		1.0	0.3	0:00.23 gnome-screensho
	coop	20	0	1492732	327060			0.7	2.0	2:04.10 chrome
1,000	root	rt	0	0	0		S	0.3	0.0	0:00.03 migration/2
A STATE OF THE STA	root	20	0	0	0		S	0.3	0.0	0:00.49 rcuop/6
	coop	20	0	1492444	63244	28244		0.3	0.4	0:02.38 gnome-settings-
2758	coop	20	0	659456	65372	51556		0.3	0.4	0:03.99 gnome-terminal-
	root	20	0	125680	5544	3676		0.0	0.0	0:01.55 systemd
	root	20	0	0	0		S	0.0	0.0	0:00.01 kthreadd
=	root	20	0	0	0		S	0.0	0.0	0:00.00 ksoftirqd/0
-	root	0	-20	0	0		S	0.0	0.0	0:00.00 kworker/0:0H
	root	20	0	0	0	0		0.0	0.0	0:02.61 rcu_preempt
	root	20	0	0	0		S	0.0	0.0	0:00.00 rcu_sched
	root	20	0	0	0	0		0.0	0.0	0:00.00 rcu_bh
100	root	20	0	0	0		S	0.0	0.0	0:00.43 rcuop/0
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00 rcuos/0

Process List of the top Output:

- 1. Each line in the process list of the top output displays information about a process.
- 2. By default, processes are ordered by highest CPU usage. The following information about each process is displayed:
 - Process Identification Number (PID)
 - Process owner (USER)
 - Priority (PR) and nice values (NI)
 - Virtual (VIRT), physical (RES), and shared memory (SHR)
 - Status (S)
 - Percentage of CPU (%CPU) and memory (%MEM) used
 - Execution time (TIME+)
 - Command (COMMAND).



Process List of the top Output

Interactive Keys with top:

1. The table lists what happens when pressing various keys when running top:

Command	Output
t	Display or hide summary information (rows 2 and 3)
m	Display or hide memory information (rows 4 and 5)
A	Sort the process list by top resource consumers
r	Renice (change the priority of) a specific processes
k	Kill a specific process
f	Enter the top configuration screen
0	Interactively select a new sort order in the process list