

Trabajo Práctico 11 - Sistemas Operativos

Integrantes:

- Borgogno, Francisco
- Cañas, Felipe
- Lucero Ruiz, Maximo
- Quesada, Santiago

Profesor: Funes, Gustavo

Ejercicio 1: Listar los Procesos y Subprocesos Activos

Comando:

```
ps -eaf
```

```
(test2@gg)-[~]
$ ps -eaf
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	13:05	?	00:00:00	/sbin/init splash
root	2	0	0	13:05	?	00:00:00	[kthreadd]
root	3	2	0	13:05	?	00:00:00	[pool_workqueue_release]
root	4	2	0	13:05	?	00:00:00	[kworker/R-rcu_g]
root	5	2	0	13:05	?	00:00:00	[kworker/R-rcu_p]
root	6	2	0	13:05	?	00:00:00	[kworker/R-slub_]
root	7	2	0	13:05	?	00:00:00	[kworker/R-netns]
root	9	2	0	13:05	?	00:00:00	[kworker/0:0H-events_highpri]
root	12	2	0	13:05	?	00:00:00	[kworker/R-mm_pe]
root	13	2	0	13:05	?	00:00:00	[rcu_tasks_rude_kthread]
root	14	2	0	13:05	?	00:00:00	[rcu_tasks_trace_kthread]
root	15	2	0	13:05	?	00:00:00	[ksoftirqd/0]
root	16	2	0	13:05	?	00:00:01	[rcu_sched]
root	17	2	0	13:05	?	00:00:00	[migration/0]
root	18	2	0	13:05	?	00:00:00	[cpuhp/0]
root	19	2	0	13:05	?	00:00:00	[cpuhp/1]
root	20	2	0	13:05	?	00:00:00	[migration/1]
root	21	2	0	13:05	?	00:00:00	[ksoftirqd/1]
root	23	2	0	13:05	?	00:00:00	[kworker/1:0H-events_highpri]

Ejercicio 2: Listar Procesos Activos con Todas las Columnas de Información

Comando:

```
ps -eF
```

```
(test2@gg)-[~]
$ ps -eF
```

UID	PID	PPID	C	SZ	RSS	PSR	STIME	TTY	TIME	CMD
root	1	0	0	5526	12076	3	13:05	?	00:00:00	/sbin/init splash
root	2	0	0	0	0	1	13:05	?	00:00:00	[kthreadd]
root	3	2	0	0	0	0	13:05	?	00:00:00	[pool_workqueue_release]
root	4	2	0	0	0	0	13:05	?	00:00:00	[kworker/R-rcu_g]
root	5	2	0	0	0	0	13:05	?	00:00:00	[kworker/R-rcu_p]
root	6	2	0	0	0	0	13:05	?	00:00:00	[kworker/R-slub_]
root	7	2	0	0	0	0	13:05	?	00:00:00	[kworker/R-netns]
root	9	2	0	0	0	0	13:05	?	00:00:00	[kworker/0:0H-events_highpri]
root	12	2	0	0	0	0	13:05	?	00:00:00	[kworker/R-mm_pe]
root	13	2	0	0	0	0	13:05	?	00:00:00	[rcu_tasks_rude_kthread]
root	14	2	0	0	0	0	13:05	?	00:00:00	[rcu_tasks_trace_kthread]
root	15	2	0	0	0	0	13:05	?	00:00:00	[ksoftirqd/0]
root	16	2	0	0	0	0	13:05	?	00:00:01	[rcu_sched]
root	17	2	0	0	0	0	13:05	?	00:00:00	[migration/0]
root	18	2	0	0	0	0	13:05	?	00:00:00	[cpuhp/0]
root	19	2	0	0	0	1	13:05	?	00:00:00	[cpuhp/1]
root	20	2	0	0	0	1	13:05	?	00:00:00	[migration/1]
root	21	2	0	0	0	1	13:05	?	00:00:00	[ksoftirqd/1]

Explicación de las Columnas:

UID: User ID del usuario que posee el proceso.

PID: Process ID.

PPID: Parent Process ID (el ID del proceso padre).

C: Utilización del procesador por el proceso.

STIME: Hora a la que el proceso comenzó.

TTY: Terminal desde la cual el proceso fue iniciado.

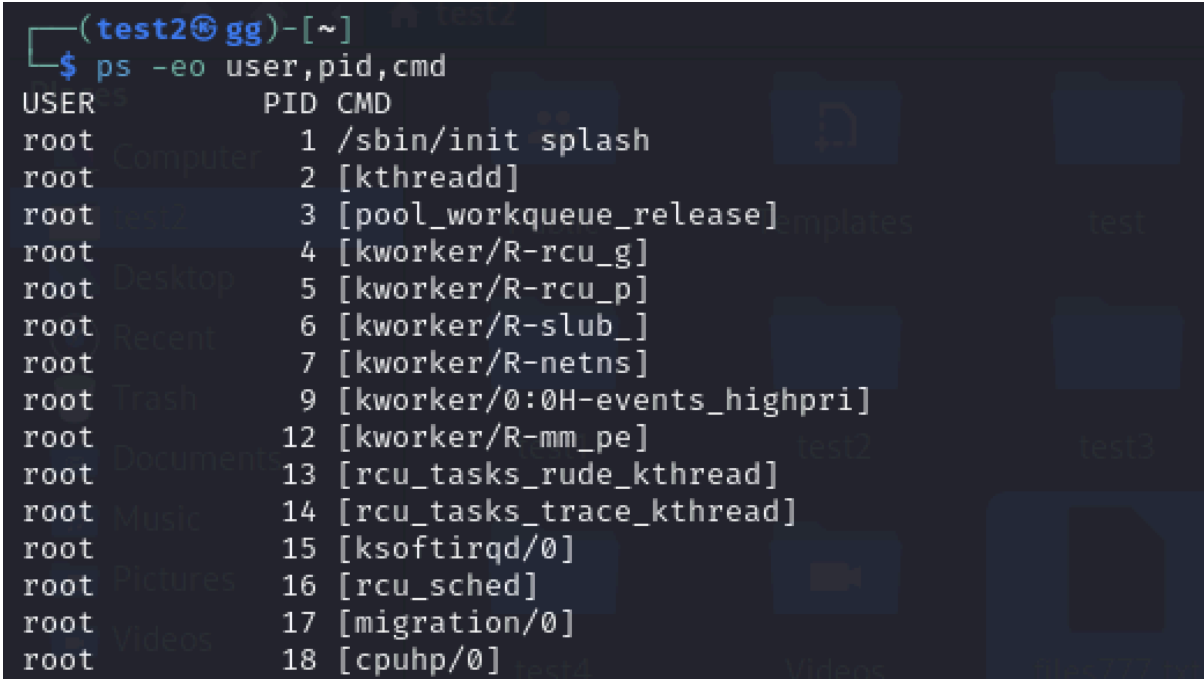
TIME: Tiempo total de CPU que el proceso ha utilizado desde que empezó.

CMD: Comando que inició el proceso.

Ejercicio 3: Listar Procesos de Manera Personalizada

Comando:

```
ps -eo user,pid,cmd
```



```
(test2@gg)-[~]  
$ ps -eo user,pid,cmd  
USER      PID  CMD  
root      1  /sbin/init splash  
root      2  [kthreadd]  
root      3  [pool_workqueue_release]  
root      4  [kworker/R-rcu_g]  
root      5  [kworker/R-rcu_p]  
root      6  [kworker/R-slub_  
root      7  [kworker/R-netns]  
root      9  [kworker/0:0H-events_highpri]  
root     12  [kworker/R-mm_pe]  
root     13  [rcu_tasks_rude_kthread]  
root     14  [rcu_tasks_trace_kthread]  
root     15  [ksoftirqd/0]  
root     16  [rcu_sched]  
root     17  [migration/0]  
root     18  [cpuhp/0]
```

Ejercicio 4: Enviar Señales a los Procesos

Comando para enviar señales:

```
kill [-SIGNAL] PID
```

Comando para listar las señales:

```
kill -l
```

```
(test2@gg)-[~]  
$ kill -l  
HUP INT QUIT ILL TRAP ABRT BUS FPE KILL USR1 SEGV USR2 P  
IPE ALRM TERM STKFLT CHLD CONT STOP TSTP TTIN TTOU URG X  
CPU XFSZ VTALRM PROF WINCH POLL PWR SYS
```

Ejercicio 5: Iniciar y Gestionar un Proceso Largo

Inicia un proceso largo en la primera terminal:

```
sleep 3000
```

En otra terminal, identifica el proceso:

```
ps -ef | grep sleep
```

```
(test2@gg)-[~]  
$ ps -ef | grep sleep  
test2      49123      1411    0 14:39 pts/0    00:00:00 sleep 3000  
test2      49498      49230    0 14:40 pts/1    00:00:00 grep --color=auto sleep
```

Detén el proceso usando su PID:

```
kill -STOP PID
```

```
(test2@gg)-[~]  
$ sudo kill -STOP 49123
```

El proceso se detuvo en la otra terminal:

```
(test2@gg)-[~]  
$ sleep 3000  
zsh: suspended (signal) sleep 3000
```

Para continuar el proceso:

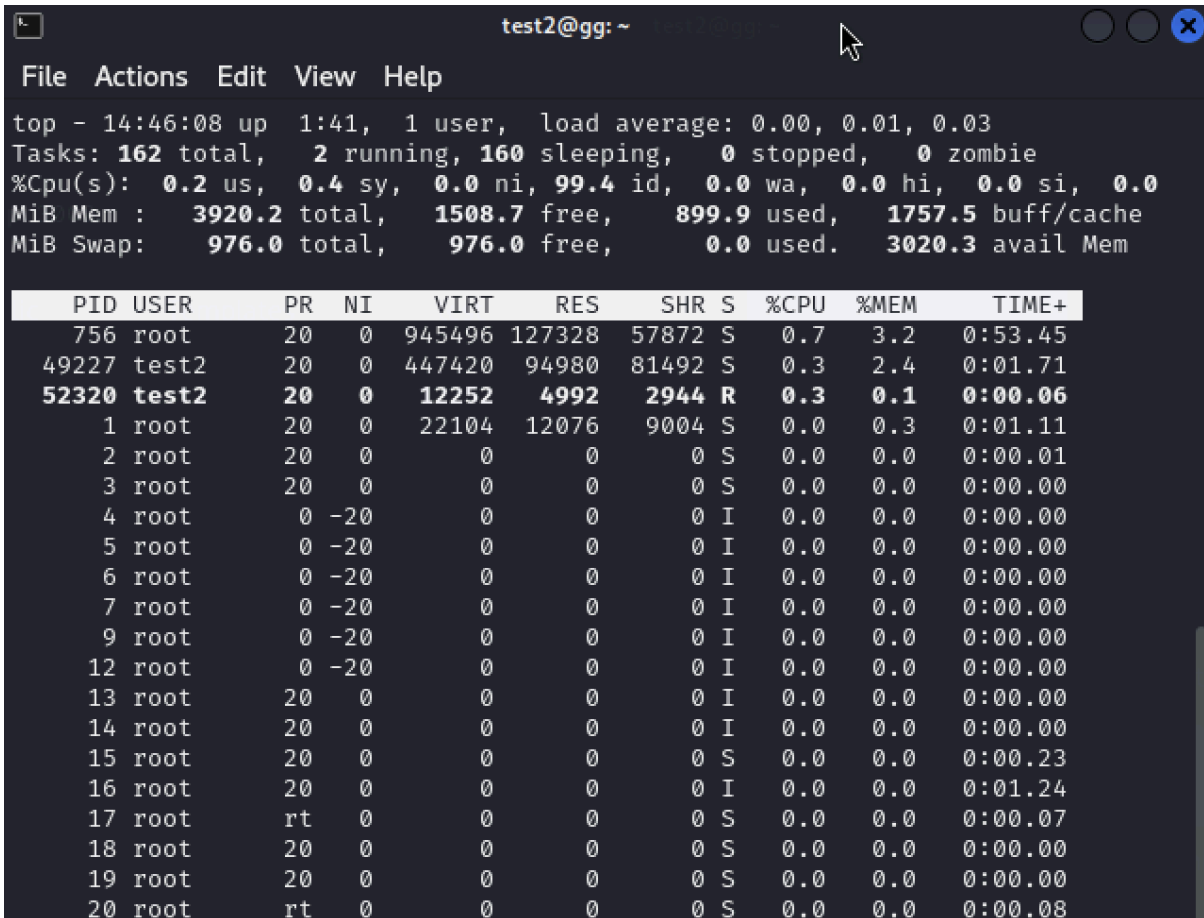
```
kill -CONT PID
```

```
(test2@gg)-[~]  
$ sudo kill -CONT 49123
```

Ejercicio 6: Mostrar el Panel de Procesos con Actualización Permanente

Comando:

top



```
test2@gg: ~  
File Actions Edit View Help  
top - 14:46:08 up 1:41, 1 user, load average: 0.00, 0.01, 0.03  
Tasks: 162 total, 2 running, 160 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 0.2 us, 0.4 sy, 0.0 ni, 99.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0  
MiB Mem : 3920.2 total, 1508.7 free, 899.9 used, 1757.5 buff/cache  
MiB Swap: 976.0 total, 976.0 free, 0.0 used. 3020.3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+
756	root	20	0	945496	127328	57872	S	0.7	3.2	0:53.45
49227	test2	20	0	447420	94980	81492	S	0.3	2.4	0:01.71
52320	test2	20	0	12252	4992	2944	R	0.3	0.1	0:00.06
1	root	20	0	22104	12076	9004	S	0.0	0.3	0:01.11
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
12	root	0	-20	0	0	0	I	0.0	0.0	0:00.00
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00
14	root	20	0	0	0	0	I	0.0	0.0	0:00.00
15	root	20	0	0	0	0	S	0.0	0.0	0:00.23
16	root	20	0	0	0	0	I	0.0	0.0	0:01.24
17	root	rt	0	0	0	0	S	0.0	0.0	0:00.07
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00
20	root	rt	0	0	0	0	S	0.0	0.0	0:00.08