

## Monitorización de procesos existentes en el sistema

### 1. Ejecute el comando xeyes.

Simplemente escribimos el comando **Xeyes** y se nos abrirá una pestaña abajo que al abrirlo nos mostrará los ojos.



### 2. A continuación, ejecute el comando 'ps' con los modificadores -axlf.

'ps'. Con sus distintas opciones, el comando nos muestra la lista de procesos del sistema, y algunas de sus características: hora de inicio, uso de memoria, estado de ejecución, propietario y otros detalles. Si lo ejecutamos sin opciones, nos muestra la lista de los procesos creados por el usuario actual y asociados al terminal de usuario en uso

Aparecen muchos más datos de este tipo.

```
walter.huataydccc54ssoo:~$ ps -axlf
F  UID  PID  PPID  PRI  NI   VSZ  RSS  WCHAN  STAT  TTY      TIME COMMAND
1  0      2      0   20    0     0    0  -  S  ?           0:00 [kthreadd]
1  0      3      2    0 -20    0     0  -  I<  ?           0:00  \ [rcu_gp]
1  0      4      2    0 -20    0     0  -  I<  ?           0:00  \ [rcu_par_gp]
1  0      6      2    0 -20    0     0  -  I<  ?           0:00  \ [kworker/0:0H-kbloc
1  0      8      2    0 -20    0     0  -  I<  ?           0:00  \ [mm_percpu_wq]
1  0      9      2    20    0     0  -  S  ?           0:00  \ [ksortirq/0]
1  0     10      2    20    0     0  -  S  ?           0:00  \ [rcu_sched]
1  0     11      2 -100   -    0     0  -  S  ?           0:00  \ [migration/0]
5  0     12      2 -51   -    0     0  -  S  ?           0:00  \ [idle_inject/0]
1  0     14      2    20    0     0  -  S  ?           0:00  \ [cpuhp/0]
5  0     15      2    20    0     0  -  S  ?           0:00  \ [kdevtmpfs]
1  0     16      2    0 -20    0     0  -  I<  ?           0:00  \ [netns]
1  0     17      2    20    0     0  -  S  ?           0:00  \ [rcu_tasks_kthre]
1  0     18      2    20    0     0  -  S  ?           0:00  \ [kauditd]
1  0     19      2    20    0     0  -  S  ?           0:00  \ [khungtaskd]
1  0     20      2    20    0     0  -  S  ?           0:00  \ [oom_reaper]
1  0     21      2    0 -20    0     0  -  I<  ?           0:00  \ [writeback]
1  0     22      2    20    0     0  -  S  ?           0:00  \ [kcompactd0]
1  0     23      2    25    5     0  -  SN  ?           0:00  \ [ksmd]
1  0     24      2    39   19    0     0  -  SN  ?           0:00  \ [khugepaged]
1  0     70      2    0 -20    0     0  -  I<  ?           0:00  \ [kintegrityd]
1  0     71      2    0 -20    0     0  -  I<  ?           0:00  \ [kblockd]
1  0     72      2    0 -20    0     0  -  I<  ?           0:00  \ [blkcg_punt_bio]
1  0     73      2    0 -20    0     0  -  I<  ?           0:00  \ [tpm_dev_wq]
1  0     74      2    0 -20    0     0  -  I<  ?           0:00  \ [ata_sff]
1  0     75      2    0 -20    0     0  -  I<  ?           0:00  \ [md]
1  0     76      2    0 -20    0     0  -  I<  ?           0:00  \ [edac-poller]
1  0     77      2    0 -20    0     0  -  I<  ?           0:00  \ [devfreq_wq]
```

3. ¿Qué PID tiene el proceso xeyes? ¿Cuál es su proceso padre? ¿Cuál es el PID del proceso padre?

4. ¿Cuántos procesos hay en ejecución en el sistema?

Puede utilizar el comando `top`, o también el comando `ps -axlf`, seguido de una tubería con el comando `wc -l`.

Ejecutamos el comando `top` y nos mostrará la siguiente información, en mi caso hay 1 proceso en ejecución (running).

```
top - 08:32:00 up 29 min, 0 users, load average: 0.04, 0.39, 0.53
Tasks: 266 total, 1 running, 265 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.7 us, 0.3 sy, 0.0 ni, 98.7 id, 0.3 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 4024.4 total, 3555.6 free, 199.0 used, 269.8 buff/cache
MiB Swap: 4096.0 total, 3979.0 free, 117.0 used, 3601.5 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
5175	jesus.a+	20	0	12752	4560	4148	S	0.3	0.1	0:01.37	xeyes
5588	alvaro.+	20	0	6796	3928	2988	S	0.3	0.1	0:00.48	top
1	root	20	0	102220	8232	6168	S	0.0	0.2	0:02.20	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	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
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kblockd
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
9	root	20	0	0	0	0	S	0.0	0.0	0:00.24	ksoftirqd/0
10	root	20	0	0	0	0	I	0.0	0.0	0:00.53	rcu_sched
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
12	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
16	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_kthre
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
20	root	20	0	0	0	0	S	0.0	0.0	0:00.11	oom_reaper
21	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	writeback
22	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kcompactd0
23	root	25	5	0	0	0	S	0.0	0.0	0:00.00	ksmd
24	root	39	19	0	0	0	S	0.0	0.0	0:00.00	khugepaged
70	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kintegrityd
71	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kblockd
72	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	blkcg_punt_bio
73	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	tpm_dev_wq

shaVtem by subscribing to the professional edition here: <https://mashvtem.makatah.net>

5. ¿Cuántos procesos son del usuario root?

Como se observa en la imagen de arriba, tengo muchos root debajo de la captura de pantalla hay muchos más.

6. Descubra la jerarquía de procesos de su usuario. Para ello, utilice el comando `ps tree -up nombreUsuario` y, posteriormente, compare los resultados obtenidos con los que consiga mediante la orden `ps -u nombreUsuario -l`.

Una segunda herramienta de monitorización de los procesos en el sistema es '`ps tree`', la cual muestra la jerarquía de los procesos mediante una estructura de árbol. Si se especifica el PID de un proceso, el árbol empezará desde ese proceso, de lo contrario el árbol empezará por el proceso init (PID=1) y mostrará todos los procesos del sistema. Si se especifica un usuario válido, se mostrará la jerarquía de todos los procesos del usuario.

Ejecutamos el comando `ps tree -up walter.huatay` y aparece la jerarquía de los procesos de mi usuario.

```
walter.huatay@dcc43ssoop3:~$ ps tree -up walter.huatay
sshd(247996) --- bash(247999)

sshd(249190) --- bash(249227) --- pstree(249461)

sshd(249258) --- sftp-server(249259)

systemd(247919) --- (sd-pam) (247922)
walter.huatay@dcc43ssoop3:~$
```

Al ejecutar el comando `ps -u walter.huatay -l` os muestra el PID de los procesos.

```
walter.huatay@dcc43ssoop3:~$ ps -u walter.huatay -l
F S  UID      PID     PPID  C  PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S   1035    247919       1   0   80   0 -  4678 ep_pol  ?           00:00:00 systemd
5 S   1035    247922    247919   0   80   0 -  42707 -      ?           00:00:00 (sd-pam)
5 S   1035    247996    247895   0   80   0 -   3513 -      ?           00:00:00 sshd
4 S   1035    247999    247996   0   80   0 -   1027 poll_s  ?           00:00:00 bash
5 S   1035    249190    249061   0   80   0 -   3514 -      ?           00:00:00 sshd
4 S   1035    249227    249190   0   80   0 -   1027 do_wai  ?           00:00:00 bash
5 S   1035    249258    249123   0   80   0 -   3513 -      ?           00:00:00 sshd
4 S   1035    249259    249258   0   80   0 -   1476 -      ?           00:00:00 sftp-ser
0 R   1035    249623    249227   0   80   0 -   1474 -      ?           00:00:00 ps
walter.huatay@dcc43ssoop3:~$
```

7. Utilizando el comando `top`, averigüe cuál es el proceso que más tiempo de CPU ha consumido. Pista: Tendrá que utilizar alguna tecla que permita la ordenación por esa columna. Para más información, consultar el apartado 4c del man correspondiente.

(Solución del ejercicio 8)

Para filtrar por orden una de las columnas que aparecen mediante el comando `top` se usa `-o` y la columna a ordenar.

Por tanto usaremos el comando `top -o %CPU` que como se observa ordena la columna del CPU de mayor a menor tiempo.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
250886	walter.+	20	0	6608	3548	2824	R	0.7	0.1	0:00.12	top
250827	jesus.a+	20	0	6604	3544	2828	S	0.3	0.1	0:00.51	top
1	root	20	0	169616	8840	6308	S	0.0	0.2	0:13.43	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.09	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
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kb
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
9	root	20	0	0	0	0	S	0.0	0.0	0:00.43	ksoftirqd/0
10	root	20	0	0	0	0	I	0.0	0.0	1:57.87	rcu_sched
11	root	rt	0	0	0	0	S	0.0	0.0	0:01.48	migration/0
12	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
16	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/1
17	root	rt	0	0	0	0	S	0.0	0.0	0:01.58	migration/1
18	root	20	0	0	0	0	S	0.0	0.0	0:00.66	ksoftirqd/1
20	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-kb
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
22	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
23	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tasks_kthre
24	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
25	root	20	0	0	0	0	S	0.0	0.0	0:00.57	khungtaskd
26	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom_reaper

8. Utilizando el comando top, averigüe cuál es el proceso que más espacio de memoria ha consumido.

(Ejercicio 7)

Ejecutamos el comando top -o TIME+ para que nos saque el tiempo que han estado ejecutando esos comandos de mayor a menor tiempo.

```
walter.huatay@dcc43ssoop3:~$ top -o TIME+
top - 08:13:15 up 5 days, 21:22, 0 users, load average: 0.99, 0.60, 0.31
Tasks: 289 total, 2 running, 287 sleeping, 0 stopped, 0 zombie
%Cpu(s): 50.1 us, 0.3 sy, 0.0 ni, 49.4 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st
MiB Mem : 3936.2 total, 3406.0 free, 254.9 used, 275.3 buff/cache
MiB Swap: 12128.0 total, 12032.5 free, 95.4 used, 3453.5 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
746	root	20	0	161384	2900	2092	S	0.0	0.1	6:13.62	vmtoolsd
251446	jesus.a+	20	0	2356	588	524	R	99.7	0.0	4:25.25	prog1.out
703	root	rt	0	345816	18212	8300	S	0.0	0.5	3:10.06	multipathd
509	root	19	-1	201404	33472	32596	S	0.0	0.8	2:15.45	systemd-journal
10	root	20	0	0	0	0	I	0.0	0.0	1:58.00	rcu_sched
438	root	20	0	0	0	0	S	0.0	0.0	1:04.48	btrfs-transacti
48925	root	20	0	1006664	16940	6220	S	0.0	0.4	0:35.28	snappd
812	104	20	0	224348	2860	2484	S	0.0	0.1	0:23.87	rsyslogd
800	root	20	0	238368	5812	4824	S	0.0	0.1	0:23.70	accounts-daemon
810	root	20	0	81960	2424	2348	S	0.0	0.1	0:14.38	irqbalance
1	root	20	0	169616	8848	6308	S	0.0	0.2	0:13.48	systemd
207	root	-51	0	0	0	0	S	0.0	0.0	0:09.75	irq/16-vmwgfx
88	root	20	0	0	0	0	S	0.0	0.0	0:07.82	kswapd0
334	root	0	-20	0	0	0	I	0.0	0.0	0:05.68	kworker/0:1H-kblock
785	100	20	0	18596	2080	1920	S	0.0	0.1	0:05.35	systemd-network
332	root	0	-20	0	0	0	I	0.0	0.0	0:05.13	kworker/1:1H-kblock
250886	walter.+	20	0	6608	3548	2824	S	0.3	0.1	0:04.72	top
539	root	20	0	22028	3160	2700	S	0.0	0.1	0:03.68	systemd-udev
804	103	20	0	7772	4356	3676	S	0.0	0.1	0:02.10	dbus-daemon
828	root	20	0	17100	5180	4840	S	0.0	0.1	0:01.73	systemd-logind
17	root	rt	0	0	0	0	S	0.0	0.0	0:01.59	migration/1
11	root	rt	0	0	0	0	S	0.0	0.0	0:01.48	migration/0
249190	walter.+	20	0	14056	6416	4844	S	0.0	0.2	0:01.23	sshd
30	root	39	19	0	0	0	S	0.0	0.0	0:01.17	khugepaged

9. ¿En qué fecha y hora arrancó el sistema? Para averiguarlo, debe utilizar algún comando d