



Actividad 05 – 08: Gestión de procesos



GESTIÓN DE LA MEMORIA Y DE LOS PROCESOS

1. [1] Abre una ventana de terminal y utiliza el comando `top` para observar los procesos en ejecución. A continuación, abre un navegador y reproduce cualquier vídeo de Youtube y observa como consume recursos.

Apunta los tres valores asociados a “load average” y explica qué es cada uno.

```
top - 23:22:11 up 3:18, 2 users, load average: 0,49, 0,22, 0,08
Tasks: 285 total, 1 running, 284 sleeping, 0 stopped, 0 zombie
%Cpu(s): 3,1 us, 1,2 sy, 0,0 ni, 95,7 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 3856,7 total, 408,8 free, 1586,5 used, 1861,5 buff/cache
MiB Swap: 2048,0 total, 2048,0 free, 0,0 used. 1629,7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4096	alberto	20	0	3843684	222364	83764	S	9,0	5,6	0:07.57	gnome-shell
3923	alberto	20	0	323548	57496	30820	S	4,0	1,5	0:04.68	Xorg

Los valores de “load average” se refieren a la carga media del sistema, en el último minuto, los últimos 5 minutos, y los últimos 15 minutos. (Los podemos ver en la primera línea de la captura de pantalla, a la derecha).

YouTube^{ES}

himno del betis

SALVA LA VIDA DE HOSSEIN
Este joven puede ser ejecutado el día 25 y con tu firma, podemos evitarlo

FIRMA

Himno de Real Betis

6,190,093 views...
38K
DISLIKE
SHARE
SAVE
...

```
top - 23:24:38 up 3:20, 2 users, load average: 1,02, 0,74, 0,30
Tasks: 292 total, 2 running, 290 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1,5 us, 0,8 sy, 0,0 ni, 95,6 id, 2,0 wa, 0,0 hi, 0,2 si, 0,0 st
MiB Mem : 3856,7 total, 158,4 free, 1978,6 used, 1719,7 buff/cache
MiB Swap: 2048,0 total, 2045,5 free, 2,5 used. 1070,5 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4890	alberto	20	0	2661380	262528	107816	S	5,3	6,6	0:18.79	Isolated Web Co
4688	alberto	20	0	3482496	328400	172136	S	4,3	8,3	0:24.31	firefox

2. [5] Ejecuta un comando desde la terminal que te permita ver todos los procesos del sistema. Al final de la lista de procesos habrá una línea que tenga en la columna CMD el nombre bash. Este es el nombre del proceso del intérprete de comandos que usamos en la terminal. Apunta su PID. A continuación, utiliza un nuevo comando ps que te permita ver más detalles del proceso con PID que has apuntado y rellena la primera línea de la siguiente tabla:

PID	PPID	Nombre proceso	Usuario dueño del proceso
4500	4492	Bash	alberto
4492	3832	gnome-terminal-server	alberto
3832	1	Systemd --user	alberto
1	0	Init splash	root

A continuación, vuelve a usar el mismo comando, pero usa como PID el PPID del comando anterior. Con los resultados rellena la siguiente línea de la tabla. Repite el proceso anterior hasta que ya no puedas hacerlo más y explica con tus propias palabras qué significado tiene la información que has obtenido.

(Pongo todos los comandos aquí abajo, pero dejo la tabla completada arriba, al ser parte del enunciado):

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -e
  PID TTY          TIME CMD
    1 ?            00:00:01 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 rcu_gp
```

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -e | grep bash
 3426 pts/0        00:00:00 bash
 4500 pts/1        00:00:00 bash
```

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -f -p 4500
UID          PID    PPID  C STIME TTY          TIME CMD
alberto      4500    4492  0 23:21 pts/1        00:00:00 bash
alberto@kadex-HP-15-Notebook-PC:~$
```

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -f -p 4492
UID          PID    PPID  C STIME TTY          TIME CMD
alberto      4492    3832  0 23:21 ?            00:00:03 /usr/libexec/gnome-terminal-server
```

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -f -p 3832
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
alberto	3832	1	0	23:21	?	00:00:00	/lib/systemd/systemd --user

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -f -p 1
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	20:03	?	00:00:01	/sbin/init splash

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -f -p 0
error: process ID out of range

Usage:
ps [options]

Try 'ps --help <simple|list|output|threads|misc|all>'
or 'ps --help <s|l|o|t|m|a>'
for additional help text.

For more details see ps(1).
```

Lo que está ocurriendo es que, por cada proceso que buscamos usando su PID, nos aparece el PPID, que es el identificador de su proceso 'padre'. Si seguimos subiendo continuamente, llega un punto que alcanzamos el PID 1, perteneciente a root. Este identificador está reservado para el Kernel de Linux, y tiene la primera prioridad de ejecución. Como podemos ver, se denomina init, en este caso, como en versiones anteriores de linux.

Aparece que el PPID de dicho proceso es 0 y, por asegurarnos, lo he buscado para que, efectivamente, constatemos que no existe dicho proceso y que, por ende, es el PID 1 el primero de todos.

3. [3] Lista todos los procesos del sistema y busca alguno que pertenezca a root. Sin usar permisos de sudo, intenta "matar" dicho proceso.

```
alberto@kadex-HP-15-Notebook-PC:~$ ps -ef
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	20:03	?	00:00:01	/sbin/init splash
root	2	0	0	20:03	?	00:00:00	[kthreadd]
root	3	2	0	20:03	?	00:00:00	[rcu_gp]
root	4	2	0	20:03	?	00:00:00	[rcu_par_gp]

```
alberto@kadex-HP-15-Notebook-PC:~$ kill 4
bash: kill: (4) - Operation not permitted
```

4. [5] El comando sleep numero_segundos actúa como una cuenta atrás de numero_segundos antes de devolverte el prompt de la terminal. Por ejemplo, con sleep 60 el terminal quedaría "dormido" durante 60 segundos. Utiliza dos terminales, una para lanzar un sleep 120 y otra terminal para averiguar el PID del proceso anterior y matar el proceso.

```
alberto@kadex-HP-15-Notebook-PC: ~  
alberto@kadex-HP-15-Notebook-PC:~$ ps -ef | grep alberto | grep sleep  
alberto      6657      4500  0 23:34 pts/1    00:00:00 sleep 120  
alberto      6661      6650  0 23:35 pts/2    00:00:00 grep --color=auto sleep  
alberto@kadex-HP-15-Notebook-PC:~$ kill 6657
```

```
alberto@kadex-HP-15-Notebook-PC: ~  
alberto@kadex-HP-15-Notebook-PC:~$ sleep 120  
Terminated
```

(Salte 'terminated' porque he hecho la captura justo después de matarlo en la otra terminal, pero lo había lanzado antes. Si no, no podría verse en la otra captura).

5. El protocolo ntp (network time protocol) permite sincronizar la hora del reloj del sistema conectándose a servidores de hora. Vamos a instalar el cliente de este protocolo del siguiente modo:

`sudo apt install ntp`

De este modo tenemos instalado un servicio llamado ntp que nos permite conectarnos a una máquina remota para sincronizar la hora del sistema.

También vamos a instalar una utilidad llamada ntpstat que nos permite diagnosticar si el servicio está funcionando correctamente. Se instala del siguiente modo:

`sudo apt install ntpstat`

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo apt install ntp  
[sudo] password for alberto:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done
```

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo apt install ntpstat  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done
```

A continuación, debes realizar las siguientes acciones:

a) [1] Arranca el servicio ntp

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo systemctl start ntp  
alberto@kadex-HP-15-Notebook-PC:~$
```

(Es un Daemon, así que utilizo systemctl)

b) [1] Usa el comando `sudo netstat -atunp` para verificar qué puertos está usando ntp tanto en la máquina local como en la máquina remota a la que se conecta. Pista: el puerto viene en la columna "Local Address"

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo apt install net-tools  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done
```

(Primero necesito las net-tools)


```
alberto@kadex-HP-15-Notebook-PC:~$ sudo netstat -atunp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      642/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      682/cupsd
tcp        0      87 192.168.1.138:50638     34.122.121.32:80       ESTABLISHED 684/NetworkManager
tcp6       0      0 :::1:631                :::*                    LISTEN      682/cupsd
udp        0      0 127.0.0.53:53           0.0.0.0:*               642/systemd-resolve
udp        0      0 192.168.1.138:68       192.168.1.1:67        ESTABLISHED 684/NetworkManager
udp        0      0 192.168.1.138:123      0.0.0.0:*               839/ntpd
udp        0      0 127.0.0.1:123          0.0.0.0:*               839/ntpd
udp        0      0 0.0.0.0:123            0.0.0.0:*               839/ntpd
udp        0      0 0.0.0.0:631            0.0.0.0:*               793/cups-browsed
udp        0      0 0.0.0.0:35713          0.0.0.0:*               680/avahi-daemon: r
udp        0      0 0.0.0.0:5353           0.0.0.0:*               680/avahi-daemon: r
udp6       0      0 :::44875                :::*                    680/avahi-daemon: r
udp6       0      0 fe80::1dc8:80a4:f2e:123 :::*                    839/ntpd
udp6       0      0 :::1:123                :::*                    839/ntpd
udp6       0      0 :::123                  :::*                    839/ntpd
udp6       0      0 :::5353                  :::*                    680/avahi-daemon: r
alberto@kadex-HP-15-Notebook-PC:~$
```

c) [1] Usa el comando de diagnóstico para saber si todo funciona correctamente.

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo systemctl status ntp
● ntp.service - Network Time Service
   Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2022-02-19 23:49:09 CET; 5min ago
     Docs: man:ntpd(8)
  Process: 806 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited, status=0/SUCCESS)
    Main PID: 839 (ntpd)
      Tasks: 2 (limit: 4549)
     Memory: 2.0M
    CGroup: /system.slice/ntp.service
            └─839 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 127:134
```

d) [1] Detiene el servicio ntp.

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo systemctl stop ntp
alberto@kadex-HP-15-Notebook-PC:~$
```

e) [2] Vuelve a ejecutar el comando netstat y el comando de diagnóstico y comprueba que realmente se ha detenido el servicio.

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo netstat -atunp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      642/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      682/cupsd
tcp6       0      0 :::1:631                :::*                    LISTEN      682/cupsd
udp        0      0 127.0.0.53:53           0.0.0.0:*               642/systemd-resolve
udp        0      0 192.168.1.138:68       192.168.1.1:67        ESTABLISHED 684/NetworkManager
udp        0      0 0.0.0.0:631            0.0.0.0:*               793/cups-browsed
udp        0      0 0.0.0.0:35713          0.0.0.0:*               680/avahi-daemon: r
udp        0      0 0.0.0.0:5353           0.0.0.0:*               680/avahi-daemon: r
udp6       0      0 :::44875                :::*                    680/avahi-daemon: r
udp6       0      0 :::5353                  :::*                    680/avahi-daemon: r
alberto@kadex-HP-15-Notebook-PC:~$
```

(Como se puede ver, no hay nada de ntp).

(Último comando en la siguiente página)

```
alberto@kadex-HP-15-Notebook-PC:~$ sudo systemctl status ntp
● ntp.service - Network Time Service
   Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Sat 2022-02-19 23:55:38 CET; 1min 50s ago
     Docs: man:ntpd(8)
   Process: 806 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited, status=0/SUCCESS)
   Main PID: 839 (code=exited, status=0/SUCCESS)
```

(Ahora aparece como 'inactive' (dead), hace casi 2 minutos, justo cuando lo he apagado)