

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from this bar, containing the date.

12-1-2021

EJERCICIOS SOBRE GESTIÓN DE PROCESOS

Módulo Sistemas Informáticos

Several thin, curved lines in shades of blue and grey originate from the bottom left corner and curve upwards and to the right.

Sergio Valle Ocaña

EJERCICIOS SOBRE GESTIÓN DE PROCESOS

1

RESUELVE

Responde a las siguientes cuestiones:

1) ¿Qué son los procesos?

En informática, un **proceso** se trata básicamente de un programa que entra en ejecución. Los procesos son una sucesión de instrucciones que pretenden llegar a un estado final o que persiguen realizar una tarea concreta. Lo más importante de este concepto, es de dónde sale un proceso o qué es realmente un programa y un sistema operativo.

El **sistema operativo** es el software básico de un ordenador, con éste, el usuario es capaz de interactuar a partir de un entorno gráfico o mediante entradas de texto en forma de instrucciones. El sistema operativo es capaz de ejecutar otros procesos dentro de sí mismo e incluso crearlos mediante código de programación y una compilación.

Un proceso también se puede dividir en distintas partes para ver cómo se ejecuta en nuestro ordenador. Dentro de él tenemos lo que llamamos instrucciones, que corresponden a cada uno de los pasos que debemos hacer para completar esa tarea.

Además, para separar cada proceso, el procesador le asigna un contador de programa, para que cada uno este separado y bien diferenciado de otro que incluso puede ser igual, por ejemplo, abrir dos veces el explorador. De esta forma cada proceso se guarda en distintos registros, con distintas variables y por supuesto en distinta región de la memoria RAM.

2) Investiga e indica al menos tres comandos Linux para trabajar con procesos.

Los procesos juegan un papel muy importante en las distribuciones Linux, ya que son los que consumirán estos recursos hardware tan preciados en entornos de producción, administrarlos y gestionarlos correctamente es de vital importancia ya que estos procesos y la gestión que hace el sistema sobre ellos, hacen posible mantener funcionando el servidor sin necesidad de reiniciar después de un cambio o actualización importante. Esto es uno de los puntos más importantes por los que Linux gobierna el 90% de los servidores alrededor del mundo.

Comandos para trabajar procesos:

-PS

-TOP

-KILL

Para ver los procesos en sistemas Linux, contamos con el comando **'ps'**, que listará (de múltiples formas según las opciones que le pasemos) todos los procesos que se encuentran corriendo en nuestro equipo.

ps [opciones]

```
toushiro@toushiro15: ~
toushiro@toushiro15:~$ ps
  PID TTY          TIME CMD
 1865 pts/17    00:00:00 bash
 1938 pts/17    00:00:00 ps
toushiro@toushiro15:~$
```

Top es otro gestor de procesos integrado en la mayoría de sistemas Linux.

Mientras que **ps** nos muestra un listado de procesos estático, es decir, nos informa de los procesos, nombres, usuarios o recursos que se están usando en el momento de la petición; **top** nos da un informe en tiempo real de los mismos.

top

```
Terminal
toushiro@toushiro15: ~
top - 16:22:44 up 23 min, 2 users, load average: 0,10, 0,21, 0,24
Tareas: 161 total, 1 ejecutar, 155 hibernar, 5 detener, 0 zombie
%Cpu(s): 2,7 usuario, 0,3 sist, 0,0 adecuado, 97,0 inact, 0,0 en espera, 0,0 hardw int, 0,0 softw int,
KiB Mem: 2048896 total, 1009848 used, 1039048 free, 89144 buffers
KiB Swap: 2095100 total, 0 used, 2095100 free. 347116 cached Mem

  PID USUARIO PR NI  VIRT  RES  SHR S %CPU %MEM    TIME+  ORDEN
 1443 toushiro 20  0 1212780 186404 67784 S  2,3  9,1   2:26.95 complz
 1050 root      20  0 293316 71612 22484 S  0,7  3,5   0:42.99 Xorg
 1206 toushiro 20  0 50580 3044 2688 S  0,7  0,1   0:05.12 VBoxClient
    1 root     20  0 119444 5696 4020 S  0,0  0,3   0:02.85 systemd
    2 root     20  0 0 0 0 S  0,0  0,0   0:00.00 kthreadd
    3 root     20  0 0 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
    6 root     20  0 0 0 0 S  0,0  0,0   0:00.19 kworker/u2:0
    7 root     20  0 0 0 0 S  0,0  0,0   0:00.46 rcu_sched
    8 root     20  0 0 0 0 S  0,0  0,0   0:00.00 rcu_bh
    9 root     20  0 0 0 0 S  0,0  0,0   0:00.33 rcuos/0
   10 root     20  0 0 0 0 S  0,0  0,0   0:00.00 rcuob/0
   11 root     rt  0 0 0 0 S  0,0  0,0   0:00.00 migration/0
   12 root     rt  0 0 0 0 S  0,0  0,0   0:00.01 watchdog/0
   13 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 khelper
   14 root     20  0 0 0 0 S  0,0  0,0   0:00.00 kdevtmpfs
   15 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 netns
   16 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 perf
   17 root     20  0 0 0 0 S  0,0  0,0   0:00.00 khungtaskd
   18 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 writeback
   19 root     25  5 0 0 0 S  0,0  0,0   0:00.00 ksm
   20 root     39 19 0 0 0 S  0,0  0,0   0:00.37 khugepaged
   21 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 crypto
   22 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 kintegrityd
   23 root     0 -20 0 0 0 S  0,0  0,0   0:00.00 bioset
```

Los sistemas Linux vienen con la herramienta **KILL** instalada, que usaremos para detener los procesos que necesitemos. Por defecto el comando kill envía una señal denominada TERM a un proceso que le pasaremos mediante su **PID** como argumento. Esta señal TERM pedirá a dicho proceso que termine, permitiéndole gestionar su función de cierre, completando las tareas necesarias y limpiando la información que ha cargado en memoria.

kill [PID del proceso]

```
toushiro@toushiro15: ~
toushiro@toushiro15:~$ ps -aux | grep docker
root      732  0.2  1.6 301664 33432 ?        Ssl  16:32   0:00 /usr/bin/docker daemon -H fd://
toushiro 2433  0.0  0.1 13716  2192 pts/17   S+   16:37   0:00 grep --color=auto docker
toushiro@toushiro15:~$ kill 732
bash: kill: (732) - Operación no permitida
toushiro@toushiro15:~$
```

3) Pon un ejemplo de uso de cada uno de los comandos del punto 2.

A continuación, pongo un ejemplo de cada uno de los comandos expuestos en el punto anterior

ps aux (muestra todos los procesos del sistema)

```
Escritorio de Ubuntu
toushiro@toushiro15: ~
toushiro@toushiro15:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.2  0.2 119444  5696 ?        Ss   15:59   0:02 /sbin/init splash
root         2  0.0  0.0      0     0 ?        S    15:59   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    15:59   0:00 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S<   15:59   0:00 [kworker/0:0H]
root         6  0.0  0.0      0     0 ?        S    15:59   0:00 [kworker/u2:0]
root         7  0.0  0.0      0     0 ?        S    15:59   0:00 [rcu_sched]
root         8  0.0  0.0      0     0 ?        S    15:59   0:00 [rcu_bh]
root         9  0.0  0.0      0     0 ?        S    15:59   0:00 [rcuos/0]
root        10  0.0  0.0      0     0 ?        S    15:59   0:00 [rcuob/0]
root        11  0.0  0.0      0     0 ?        S    15:59   0:00 [migration/0]
root        12  0.0  0.0      0     0 ?        S    15:59   0:00 [watchdog/0]
root        13  0.0  0.0      0     0 ?        S<   15:59   0:00 [khelper]
root        14  0.0  0.0      0     0 ?        S    15:59   0:00 [kdevtmpfs]
root        15  0.0  0.0      0     0 ?        S<   15:59   0:00 [netns]
root        16  0.0  0.0      0     0 ?        S<   15:59   0:00 [perf]
root        17  0.0  0.0      0     0 ?        S    15:59   0:00 [khungtaskd]
root        18  0.0  0.0      0     0 ?        S<   15:59   0:00 [writeback]
root        19  0.0  0.0      0     0 ?        SN   15:59   0:00 [ksmd]
root        20  0.0  0.0      0     0 ?        SN   15:59   0:00 [khugepaged]
root        21  0.0  0.0      0     0 ?        S<   15:59   0:00 [crypto]
root        22  0.0  0.0      0     0 ?        S<   15:59   0:00 [kintegrityd]
root        23  0.0  0.0      0     0 ?        S<   15:59   0:00 [bioset]
root        24  0.0  0.0      0     0 ?        S<   15:59   0:00 [kblockd]
root        25  0.0  0.0      0     0 ?        S<   15:59   0:00 [ata_sff]
root        26  0.0  0.0      0     0 ?        S<   15:59   0:00 [nd]
root        27  0.0  0.0      0     0 ?        S<   15:59   0:00 [devfreq_wq]
root        31  0.0  0.0      0     0 ?        S    15:59   0:00 [kswapd0]
root        32  0.0  0.0      0     0 ?        S    15:59   0:00 [fsnotify_mark]
root        33  0.0  0.0      0     0 ?        S    15:59   0:00 [ecryptfs-kthrea]
root        44  0.0  0.0      0     0 ?        S<   15:59   0:00 [kthrotld]
```

`top -u toughiro` (Donde **Toughiro** es el usuario del cual queremos mostrar los procesos)

```

Terminal
toughiro@toughiro15: ~
top - 16:27:02 up 27 min, 2 users, load average: 0.11, 0.16, 0.22
Tareas: 161 total, 1 ejecutar, 155 hibernar, 5 detener, 0 zombie
%Cpu(s): 7.7 usuario, 0.7 sist, 0.0 adecuado, 91.6 inact, 0.0 en espera, 0.0 hardw int, 0.0 softw int,
KiB Mem: 2048896 total, 1009588 used, 1039308 free, 89176 buffers
KiB Swap: 2095100 total, 0 used, 2095100 free, 347220 cached Mem

  PID  USUARIO  PR  NI  VIRT  RES  SHR  S  %CPU  %MEM  HORA+ ORDEN
1443 toughiro 20  0 1212780 186404 67784 S  6.3  9.1  2:40.89 complz
1206 toughiro 20  0 50580 3044 2688 S  0.3  0.1  0:06.15 VBoxClient
6770 toughiro 20  0 29296 3216 2680 R  0.3  0.2  0:00.03 top
1162 toughiro 20  0 45168 5212 4324 S  0.0  0.3  0:00.06 systemd
1163 toughiro 20  0 60968 1756 0 S  0.0  0.1  0:00.00 (sd-pam)
1166 toughiro 20  0 46056 4720 3624 S  0.0  0.2  0:00.68 upstart
1183 toughiro 20  0 49400 292 4 S  0.0  0.0  0:00.00 VBoxClient
1184 toughiro 20  0 117784 4240 3720 S  0.0  0.2  0:00.00 VBoxClient
1192 toughiro 20  0 49400 288 4 S  0.0  0.0  0:00.00 VBoxClient
1193 toughiro 20  0 115600 3136 2776 S  0.0  0.2  0:00.00 VBoxClient
1199 toughiro 20  0 49400 296 4 S  0.0  0.0  0:00.00 VBoxClient
1201 toughiro 20  0 115600 3184 2816 S  0.0  0.2  0:00.00 VBoxClient
1205 toughiro 20  0 49400 292 4 S  0.0  0.0  0:00.00 VBoxClient
1317 toughiro 20  0 32332 256 12 S  0.0  0.0  0:00.05 upstart-udev-br
1321 toughiro 20  0 43672 4024 2732 S  0.0  0.2  0:01.04 dbus-daemon
1333 toughiro 20  0 88172 9868 9224 S  0.0  0.5  0:00.05 window-stack-br
1363 toughiro 20  0 308576 11388 8300 S  0.0  0.6  0:00.06 gnome-keyring-d
1374 toughiro 20  0 271444 6136 5496 S  0.0  0.3  0:00.14 gvfsd
1386 toughiro 20  0 32380 1636 1272 S  0.0  0.1  0:00.04 upstart-dbus-br
1387 toughiro 20  0 32400 1628 1256 S  0.0  0.1  0:00.12 upstart-dbus-br
1389 toughiro 20  0 40876 1712 1280 S  0.0  0.1  0:00.02 upstart-file-br
1390 toughiro 20  0 290620 8504 7516 S  0.0  0.4  0:00.04 tbus-daemon
1398 toughiro 20  0 422132 7444 6620 S  0.0  0.4  0:00.03 gvfsd-fuse
1399 toughiro 20  0 600892 39168 31456 S  0.0  1.9  0:00.99 bamfdemon
1418 toughiro 20  0 566432 34740 29936 S  0.0  1.7  0:00.53 hud-service
  
```

`kill -KILL [PID del proceso]` Con este comando, no estamos mandando al proceso ninguna señal, directamente estamos diciéndole al kernel del sistema que descarte y cierre dicho proceso.

```

root@toughiro15: ~
toughiro@toughiro15:~$ sudo su
root@toughiro15:/home/toughiro# cd
root@toughiro15:~# ps aux | grep docker
root      732  0.2  1.6 301664 33432 ?        Ssl  16:32   0:00 /usr/bin/docker daemon -H fd://
root     2459  0.0  0.1 13716 2220 pts/17    S+   16:38   0:00 grep --color=auto docker
root@toughiro15:~# kill -KILL 732
root@toughiro15:~# ps aux | grep docker
root     2463  0.0  0.1 13712 2244 pts/17    S+   16:38   0:00 grep --color=auto docker
root@toughiro15:~#
  
```

4) ¿Cómo podemos ver los procesos que se están ejecutando en un sistema operativo Windows?

La tarea de visualizar los procesos en el sistema operativo Windows. Esto es tarea sencilla, ya que solo debemos de pulsar con *botón derecho* sobre la barra de tareas y elegir la opción "Administrador de tareas". Una segunda forma de hacer esto será

pulsar la combinación de teclas "*Ctrl+ Mayus + Esc*". Y una tercera será pulsar la combinación de teclas "*Ctrl + Alt + Supr*".

Qué es un proceso informático y x +

Administrador de tareas

Archivo Opciones Vista

Procesos Rendimiento Historial de aplicaciones Inicio Usuarios Detalles Servicios

Nombre	Estado	9% CPU	30% Memoria	0% Disco	0% Red	Consumo de ...	Tendencia de ...
Aplicaciones (4)							
> Administrador de tareas		0,4%	19,8 MB	0 MB/s	0 Mbps	Muy baja	
> Explorador de Windows		0,1%	46,0 MB	0 MB/s	0 Mbps	Muy baja	
> Google Chrome (9)		1,1%	169,9 MB	0 MB/s	0 Mbps	Muy baja	
> Microsoft Word (2)		0%	69,9 MB	0 MB/s	0 Mbps	Muy baja	
Procesos en segundo plano (59)							
> Adobe Acrobat Update Service (...)		0%	0,1 MB	0 MB/s	0 Mbps	Muy baja	
> Adobe Genuine Software Servic...		0%	0,8 MB	0 MB/s	0 Mbps	Muy baja	
Adobe IPC Broker (32 bits)		0%	0,9 MB	0 MB/s	0 Mbps	Muy baja	
> Adobe Update Service (32 bits)		0%	0,3 MB	0 MB/s	0 Mbps	Muy baja	
AMD External Events Client Mo...		0%	1,0 MB	0 MB/s	0 Mbps	Muy baja	
> AMD External Events Service Mo...		0%	0,3 MB	0 MB/s	0 Mbps	Muy baja	
> Antimalware Service Executable		0,1%	144,1 MB	0 MB/s	0 Mbps	Muy baja	
> AnyDesk (32 bits)		0%	1,1 MB	0 MB/s	0 Mbps	Muy baja	
> AnyDesk (32 bits)		0%	3,9 MB	0 MB/s	0 Mbps	Muy baja	
> Aplicación de subsistema de cola		0%	7,5 MB	0 MB/s	0 Mbps	Muy baja	
Application Frame Host		0%	2,3 MB	0 MB/s	0 Mbps	Muy baja	
> Búsqueda (2)		0%	2,3 MB	0 MB/s	0 Mbps	Muy baja	
Cargador de CTF		0%	2,4 MB	0 MB/s	0 Mbps	Muy baja	
CCXProcess		0%	0,1 MB	0 MB/s	0 Mbps	Muy baja	

Menos detalles Finalizar tarea

5) Indica cómo proceder en Windows para matar, reanudar, etc procesos.

Matar procesos en Windows

Forzar el cierre desde el Administrador de tareas de Windows

En la mayoría de casos, se puede conseguir cerrar el programa desde el Administrador de tareas de Windows:

- Presiona simultáneamente las teclas CTRL+ALT+SUPR/DEL y selecciona "Administrador de tareas".
- Haz clic derecho sobre el proceso en cuestión y selecciona "Finalizar tarea".

Lamentablemente, este método no funciona siempre.

El comando Taskkill

Cuando utilizamos el Administrador de Windows, este utilitario también es llamado internamente. Sin embargo, en este caso no podemos utilizar las opciones del comando kill.

Para hacer un Taskkill manualmente, abre un prompt ("Windows" + R, escribe cmd y acepta).

Si has identificado un proceso que deseas terminar, tendrás que conocer su nombre y en ocasiones su PID (como hemos mencionado antes, puedes saberlo haciendo uso del comando Tasklist).

Procedimiento:

- Con el comando *"taskkill /IM NombreDeimagen.exe /F"*
- Con el comando *"taskkill /PID xxxx /F" (donde xxxx es el PID)*

A continuación, veamos el significado de las opciones de TaskKill:

- /F ? Fuerza el cierre de un proceso
- /IM ? Nombre de la imagen
- /PID ? Nombre del Proceso

Si todo ha funcionado correctamente, debería aparecer un mensaje indicando que el proceso ha sido eliminado.

Reanudar o pausar procesos en Windows

Desde el Administrador de tareas de Windows:

- Presiona simultáneamente las teclas CTRL+ALT+SUPR/DEL y selecciona "Administrador de tareas".
- Haz clic derecho sobre el proceso en cuestión y selecciona "Pausar tarea" o "Reanudar tarea" según queramos una opción u otra.

Conclusión sobre un proceso informático.

Cada proceso tiene una finalidad y según que sistema operativo se pueden interpretar de una manera u otro, pero la finalidad es la misma.

Se puede decir que un proceso tiene un punto de partida, en el cual se ejecuta, para obtener un resultado.