

Linux para administradores (intermedio)

Manuel Domínguez

Bienvenidos!

Esta sección corresponde con la **Gestión de procesos y servicios**.

Y en esta clase, vamos a explicar cómo se gestionan **los procesos**.

1.- Introducción

Linux es un sistema **multiusuario y multitarea**. Es decir, al mismo tiempo se están ejecutando múltiples procesos de uno o varios usuarios.

Realmente **la CPU sólo puede atender a un proceso** en cada instante. Lo que ocurre es que reparte su tiempo de CPU.

Un **proceso** es la **unidad elemental de ejecución**.

Normalmente se asocia aplicación con proceso, pero no tiene porqué coincidir.

2.- Características de los procesos.

PID: Un número que identifica al proceso.

El **estado** en el que se encuentra:

Ejecución (**R**unning): Está haciendo uso de la CPU.

Listo (**S**leeping): Esperando o preparado.

Bloqueado (**S**Topped): Necesita algún recurso, que está ocupado.

Zombie: Proceso terminado pero sigue apareciendo en la tabla de procesos.

El **usuario** que lo ejecuta.

El **terminal** donde fue invocado.

La **prioridad**.

3.- Monitorización de los procesos.

Existen muchas herramientas para **monitorizar procesos**:

Destacamos:

ps → Nos muestra los procesos del sistema.

pstree → Nos muestra los procesos en forma de árbol.

top → Nos muestra los procesos en forma gráfica y con un menú.

htop → Nos muestra los procesos en forma gráfica y con un menú.

3.- Monitorización de los procesos:ps

Práctica: Preparación del entorno

Heidi (tty3) → Abre un terminal textual y ejecuta **\$nano**

pedro (tty4) → Abre un terminal textual y ejecuta **\$cmatrix**

usuario (pts0) → **root**

3.- Monitorización de los procesos.

ps → Nos muestra los procesos del sistema.

ps #ps u #ps au|less #ps aux|less. #ps -u pedro

a: Procesos asociados a un terminal.

u: Información abundante.

x: Procesos no asociados a un terminal (sistema).

-u Nombre_Usuario: Selecciona los procesos de un usuario

pedro	2698	0.0	0.1	7652	4480	tty4	S	07:40	0:00	-bash
pedro	2702	1.5	0.0	2664	940	tty4	S+	07:40	0:01	cmatrix

¿Sabes decirme qué proceso tiene el PID 1?

3.- Monitorización de los procesos.

pstree → Nos muestra los procesos en forma de árbol.

#pstree → Podemos ver los procesos padres e hijos

#pstree -p → Podemos ver el PID de los procesos.

3.- Monitorización de los procesos.

top → Nos muestra los procesos en forma gráfica y con un menú de opciones:

La lista se actualiza por defecto, cada 3 segundos.

Ordenar los procesos: **M (Memoria) P (CPU)**

Procesos zombies: **i**

Seleccionar los procesos de un usuario: **U** → **pedro**

3.- Monitorización de los procesos.

htop → Nos muestra los procesos en forma gráfica y con un menú.

No viene por defecto instalada, pero merece la pena echarle un vistazo.
Parecida al top, pero con un menú más visual en la parte inferior.

apt install htop

#htop

F6 → Ordenar por PID, USER, STATE, CPU, MEM, etc.

#F3 → Buscar. Ejemplo: cmatrix

4.- Procesos en primer y segundo plano

Primer plano (Foreground):

La consola se queda bloqueada hasta que el proceso termine.
Sólo puede haber un proceso ejecutándose en primer plano.

Segundo plano (Background):

La consola no queda bloqueada.
Se pueden ejecutar varios procesos

Demonio: es un proceso que se ejecuta en segundo plano y es autónomo, de manera que no necesita interacción por parte de un usuario del sistema para arrancar y funcionar.

4.- Procesos en primer y segundo plano

Primer plano:

Proceso: CTRL+C → Se cancela.

Proceso: CTRL+Z → Se manda a 2º plano y se suspende.

Segundo plano:

\$proceso & → El proceso se ejecuta en 2º plano.

\$jobs → Visualizamos los procesos en 2º plano

Proceso está suspendido (CTRL+Z)

\$bg %N → Reinicia el trabajo en 2º plano.

Proceso suspendido o ejecutándose en 2º plano

\$fg %N → Se incorpora a primer plano.

4.- Procesos en primer y segundo plano

Práctica: nano primer y segundo plano

- 1.- **\$ nano prueba &** → Hemos mandado el proceso a 2º plano
- 2.- **\$ps u** → Procesos ejecutándose en el terminal. **T (Stopped): nano prueba**
- 3.- **\$jobs** → Vemos los procesos en segundo plano.
- 4.- **\$fg %N** → Nos traemos el proceso a primer plano.
Si sólo ponemos %, nos traerá el último proceso que ha entrado.

4.- Procesos en primer y segundo plano

Práctica: copia de seguridad en 2º plano.

OPCIÓN 1.- `#tar -zcf /root/home-$(date +%d-%m-%Y).tar.gz /home &`

OPCIÓN 2.- Supongamos que nos ponemos hacer la copia de seguridad y vemos que tarda mucho.

- 1.- La detenemos: CTRL+Z
- 2.- La reanudamos en segundo plano: `bg %N`

4.- Procesos en primer y segundo plano

¿Dónde puede ser interesante ejecutar un proceso en 2º plano?

Imaginaros que os conectáis por ssh a un servidor y tenéis que hacer copias de seguridad.

5.- Matar procesos

Los procesos se comunican entre sí mandando señales.
Ejemplo: las banderas entre los barcos.

\$Kill -I Muestra las señales disponibles.

SIGKILL (9) Termine inmediatamente.

SIGTERM (15) Le da una oportunidad de terminar ordenadamente.
Puede ser ignorada por el proceso.

5.- Matar procesos

Práctica: Matar un proceso determinado.

- 1.- Heidi abre un proceso: \$ nano prueba
- 2.- Buscamos el proceso: # **ps u -u heidi**
- 3.- Lo eliminamos: #**kill -9 1784** (1784 es un ejemplo)

5.- Matar procesos

Práctica: Matar muchos procesos al mismo tiempo.

1.- Usuario, desde su entorno gráfico escribe:

`$xclock &`

2.- Repite el comando 10 veces.

3.- Utilizamos **ps u** o **jobs** para localizar el proceso. Podemos utilizar también: **#pgrep xclock** para localizar los PID's.

4.- Para eliminar todos los procesos, vamos a recurrir a su nombre, en lugar de a su PID: **#killall -9 xclock**

5.- Matar procesos

Práctica: Desconectar a un usuario

- 1.- Supongamos que heidi está conectada en nuestro sistema.
- 2.- Utilizamos: **#who** para averiguar los usuarios conectados.
- 3.- **#pkill -9 -u heidi**

RETO

Práctica:

- 1.- Serías capaz de mostrar en pantalla los 10 procesos que consumen más Memoria RAM, ordenados de mayor a menor.
- 2.- Serías capaz de mostrar en pantalla los 10 procesos que consumen más CPU, ordenados de mayor a menor.

Pista: el comando ps (aux) tiene una **opción sort** para ordenar procesos. También puedes utilizar la opción **width** del comando ps (aux) para establecer la anchura de separación.

Linux para administradores (intermedio)

Manuel Domínguez

Despedida

Hemos llegado al final de este vídeo.

Nos vemos en el siguiente.