

Ejercicios Comandos Gestión procesos en Linux -resuelto

Comando ps

1. Muestra por terminal los procesos activos en el sistema.

Ps aux

2. Inicia sesión con el usuario User02. Averigua cuales son los procesos activos de root.

sudo su - User02

ps aux | grep root

exit

3. Vuelve a la sesión de alumno. Mostrar, por pantalla, solo el listado de los procesos o comandos, el pid y el consumo de CPU de cada uno de ellos.

ps -eo cmd,pid,%cpu

4. Idem que el ejercicio anterior pero ordenado por consumo de cpu.

ps -eo cmd,pid,%cpu --sort=-%cpu

5. Idem que el ejercicio anterior pero solo los 3 procesos que más cpu consumen.

ps -eo cmd,pid,%cpu --sort=-%cpu | head -n 4

6. Mostrar, por pantalla, solo el pid , el comando y la cantidad de memoria ram que consumen los procesos.

ps -eo pid,cmd,%mem

7. Idem que el ejercicio anterior pero ordenado por cantidad de memoria consumida de mayor a menor

ps -eo pid,cmd,%mem --sort=-%mem

8. Idem que el anterior pero solo los 5 procesos que más cantidad de memoria consumen.

ps -eo pid,cmd,%mem --sort=-%mem | head -n 6

9. Mostrar el listado de todos los procesos que se están ejecutando incluyendo su jerarquía, es decir, de quien dependen.

ps tree

10. Mostrar los procesos de systemd incluyendo la jerarquía.

ps tree -p | grep systemd

11. Mostrar por pantalla el identificador (pid) , el nombre (comm) y la prioridad (ni) de todos los procesos del sistema.

ps -eo pid,comm,ni

12. Idem pero ordenando por máxima prioridad los procesos.

ps -eo pid,comm,ni --sort=-ni

13. Idem pero mostrando los 5 procesos con máxima prioridad.

ps -eo pid,comm,ni --sort=-ni | head -n 6

14. Mostrar el pid, cantidad de cpu , cantidad de memoria, usuario y comando de todos los procesos secundarios de un proceso systemd.

pgrep -P \$(pgrep -f systemd) -a | awk '{print \$1, \$(NF-4), \$(NF-3), \$(NF-2), \$(NF)}'

Comando pstree

15. Mostrar el árbol de procesos activos en el sistema.

ps tree

16. Mostrar el árbol de procesos activos del sistema deshabilitando los nombres repetidos.

ps tree -U

17. Mostrar el árbol de procesos incluyendo también los argumentos de la línea de comandos en la salida.

ps tree -a

18. Mostrar resaltado el proceso cuyo pid darás como opción.

pstree -p <PID>

19. Mostrar los ID de los grupos de procesos en la salida.

pstree -g

20. Ordenar los procesos en función de los PID

pstree -p --sort=p

21. Mostrar el árbol de procesos del usuario alumno

pstree -u alumno

22. mostrar sólo la información de los padres e hijos del proceso 1701.

pstree -p 1701

Comando jobs

23. ¿Cuándo se debe lanzar un proceso en segundo plano o background?

Se debe lanzar un proceso en segundo plano cuando se desea ejecutar un comando sin bloquear la terminal y permitir que otros comandos se ejecuten mientras el proceso en segundo plano continúa.

24. ¿Cuándo se lanzara un proceso en primer plano o foreground?

Se lanzará un proceso en primer plano cuando se desea ejecutar un comando y bloquear la terminal hasta que ese proceso termine.

25. Ejecuta los siguientes pasos en el terminal de la maquina virtual:

a. Ejecuta el comando Firefox

firefox

b. Pulsa la combinación de teclas Ctrl+Z. ¿Qué ocurre?

Firefox se detiene y vuelve al shell.

c. Ejecuta el comando jobs. ¿Qué muestra?

Muestra el trabajo suspendido, en este caso, el proceso de Firefox.

d. Ejecuta el comando gparted como root.

sudo gparted

e. Pulsa la combinación de teclas Ctrl+Z.

Gparted se detiene y vuelve al shell.

- f. Accede a la pantalla donde esta Firefox iniciado y, ¿Qué ocurre?

No se puede acceder a la pantalla de Firefox, ya que está suspendido.

- g. Vuelve al terminal y ejecuta el comando jobs. ¿Qué observas?

Muestra dos trabajos suspendidos, uno para Firefox y otro para gparted.

- h. Observa que la línea donde aparece Firefox tiene un número. Ejecuta la orden fg numero_linea_firefox. ¿Qué ha ocurrido?

fg 1

Firefox vuelve al primer plano.

- i. Pulsa la combinación de teclas Ctrl+Z.

Firefox se detiene y vuelve al shell.

- j. Ejecuta el comando jobs y observa que aparece.

Muestra el trabajo suspendido de Firefox.

- k. Ejecuta el comando killall con el parámetro -9 y el nombre del proceso firefox. ¿Qué ocurre?

Firefox se cierra al momento

- l. Ejecuta el comando jobs. ¿Qué muestra?

No muestra nada, ya que no hay trabajos en segundo plano.

- m. Ejecuta el comando killall con el parámetro -9 y el nombre del proceso gparted.

sudo killall -9 gparted

- n. Ejecuta el comando jobs.
- o. Ejecuta la siguiente orden: yes > /dev/null. ¿Qué ocurre?

El comando yes se ejecuta en segundo plano y genera una salida infinita.

- p. Pulsa la combinación de teclas Ctrl+Z y ejecuta el comando jobs.

Muestra el trabajo en segundo plano de yes.

- q. Ejecuta la orden : killall -9 yes

Detiene el proceso yes.

- r. Ejecuta la siguiente orden: yes > /dev/null&. ¿Qué ocurre ahora? Ejecuta el comando jobs.

El comando yes se ejecuta en segundo plano.

- s. Ejecuta el comando: fg 1. ¿Qué observas? Pulsa la combinación de teclas Ctrl+Z y ejecuta el comando jobs.

El comando yes vuelve al primer plano. Al pulsar Ctrl+Z, se detiene y vuelve al shell.

- t. Ejecuta el comando: bg 1. Ejecuta el comando jobs.

El comando yes vuelve a ejecutarse en segundo plano.

- u. Ejecuta el comando ps. Observa la línea donde aparece el comando yes. ¿Qué indica?

Muestra información sobre el proceso yes, incluyendo su PID.

- v. Ejecuta el comando kill con la opción -9 y el numero PID que aparece en la línea de yes.

Detiene el proceso yes en el momento