Práctica clase 2. Tema 1. Gestión de procesos Linux

1. En esta práctica vamos a crear un proceso y vamos a manipularlo para hacerlo pasar a diferentes estados de ejecución.
2. $ nano prueba (creamos un proceso en el sistema operativo)
3. Al ejecutar el programa “nano prueba” estamos creando un nuevo proceso dentro del sistema operativo, por tanto lo primero que realizará el sistema será crear su PCB (bloque de control de proceso), para crear su PCB lo primero que tiene que hacer es cargar en memoria todo el texto del programa y apuntar el PC (Contador de programa a esa dirección).
4. Una vez creado el PCB y asignado el PID (número de identificador de proceso), el proceso podrá pasar a la cola de listos y en cualquier momento pasará a tener ráfagas de ejecución gracias al planificador.
5. Abrimos otra terminal mientras en la terminal anterior tenemos abierto el proceso (nano)
6. En la nueva terminal si lanzamos ps obtenemos solo los procesos asociados a ese usuario dentro de esa terminal.
7. Para obtener los procesos disponibles en todas las terminales abiertas por el usuario lanzamos ps con la opción -a
8. Una vez lanzado “ps -a” podemos ver la lista de todos los procesos con sus PIDs, con la opción “ps -as” podremos ver además su estado, podremos observar que el proceso que tiene el programa nano está bloqueado en espera de E/S.
9. Vamos a manipular el estado del proceso.
10. Los procesos de un sistema operativo pueden recibir determinadas señales, según la señal el proceso actuará de una manera u otra.
11. El programa “kill” nos permite enviar señales a un proceso a través de su PID, por ejemplo la señal 9 hace que un proceso muera en el estado actual sin terminar su ejecución.
12. $ kill -9 PID\_DE\_NANO (matará el proceso asociado al programa nano).
13. Busca en la lista de señales disponibles y lanza la señal SIGSTP al proceso.
14. En la primera terminal verás que el proceso ha quedado en estado detenido, podemos ver su estado actual a través de jobs.
15. Para pasar un proceso a ejecución que ha sido detenido lanzamos “fg 1” pasamos a primer plano el trabajo 1.
16. Para pasar un proceso a segundo plano lanzamos “bg 1” pasamos el proceso en segundo plano.
17. Para ejecutar un proceso directamente en segundo plano, hacemos:
18. $ nano prueba &
19. En cualquier momento podemos pasarlo a primer plano averiguando su número de trabajo o a través del PID enviando una señal.
20. $ fg 1