

## Parte 1, Clase 12

### 1 ¿Qué es un usuario root en Linux?

R//El usuario root en GNU/Linux es el usuario que tiene acceso administrativo al sistema. Los usuarios normales no tienen este acceso por razones de seguridad.

### 2. ¿Por qué ubuntu no me deja establecer la contraseña durante la instalación?

R// Debido a que Ubuntu es una versión de Linux, que es una versión específica de procesador de Unix (que estaba algo basada en Multics). Unix fue diseñado desde el principio como un sistema multitarea multiusuario, por lo que las contraseñas se usaron para proteger diferentes cuentas de usuario en la misma máquina. Esto también incluía cuentas del sistema como root, que también tenían una contraseña para evitar que usuarios desconocidos causen daños al sistema.

### 3. ¿Cuáles son los procesos típicos de Linux? ¿Cómo identificarlos?

Linux, como ya hemos visto, es un sistema operativo multitarea y multiusuario. Esto quiere decir que múltiples procesos pueden operar simultáneamente sin interferir unos con otros. Por ejemplo, si cinco usuarios desde equipos diferentes ejecutan el mismo programa al mismo tiempo, habría cinco instancias del mismo programa, es decir, cinco procesos distintos.

Cada proceso que se inicia es identificado con un número de identificación único conocido como Process ID (PID), que es siempre un número natural.

Haciendo análisis muchas veces ejecutaremos programas, crearemos procesos, que duren un tiempo considerable. Es interesante que durante el tiempo que dure el proceso podamos consultar su estado. Los entornos UNIX tienen una serie de herramientas para poder conocer el estado de los procesos y del sistema en general.

Para identificar los procesos debemos usar el comando: `ps -eH | less`.

El parámetro `-e` muestra todos los procesos ejecutados en el sistema por parte de todos los usuarios. Mientras que el parámetro `-H` muestra la jerarquía de los procesos (hierarchy).

```
$ ps
```

Si queremos obtener la lista completa de procesos podemos usar las opciones `-ef`:

```
$ ps -ef
```

En este caso la segunda columna nos indicará el PID o identificador único del proceso.

**kill**

El comando kill, a pesar de su nombre, no sólo sirve para matar o terminar procesos sino también para enviar señales a los procesos. La señal por defecto (cuando no se indica ninguna es terminar o matar el proceso), y la sintaxis es kill PID, siendo PID el número de ID del proceso. Pero hay otras señales que podemos enviar. Así, por ejemplo, es posible enviar una señal de STOP al proceso y se detendrá su ejecución, después cuando se quiera reanudar su ejecución podemos enviar la señal CONTInuar y el proceso continuará desde donde se quedó detenido. Con kill -l podemos acceder a una lista de todas las señales que podemos mandar a un proceso:

```
$ kill -l
```

El modo más convencional de matar un proceso es intentar primero que muera ordenadamente con un -15 (Termination signal) y sino lo conseguimos matarlo con un -9 (Kill signal):

```
$ kill -15 4719
```

```
$ kill -9 4719
```

### **free**

free nos muestra información sobre el uso y disponibilidad de la memoria. Es aconsejable usar la opción -h ya que así generará la información en una forma más fácil de leer para los seres humanos.

```
$ free -h
```

### **top**

Una herramienta muy usada y muy útil para el monitoreo en tiempo real del estado de los procesos y de otras variantes del sistema es el programa llamado top, se ejecuta desde la línea de comandos, es interactivo y por defecto se actualiza cada 3 segundos.

Estando dentro de la aplicación, presionando 'h' se accede a una ayuda de los posibles comandos que permiten configurar top, por ejemplo, al presionar 's' pregunta por el tiempo en segundos de actualización, etc.

Una alternativa más moderna al comando top, que normalmente no es instalada por defecto, es htop.

## **screen y byobu**

screen permite tener varios shells a la vez en la misma terminal y permite cerrar la terminal sin que los shells sean destruidos. screen tiene dos usos principales:

- Tener varios shells abiertos bajo una sola sesión SSH
- Lanzar procesos que queremos que se mantengan entre distintas sesiones.

byobu es una modificación de screen que lo hace más fácil de utilizar. La distribución Ubuntu mantiene un manual de ayuda para \*byobu\*

### **4. ¿Cómo establecer la contraseña root?**

Necesitamos abrir una terminal; para establecer una clave o password a root, y escribir el siguiente comandó: `sudo passwd root`.

Tendrás que escribir tu password actual de tu cuenta con acceso sudo.

Después te pedirá crear la clave UNIX, la cual será la clave de root. a la hora de escribir los passwords o claves no verás ningún carácter escribiéndose, por seguridad. De [Enter] para finalizar cada clave.

Para hacer login con la nueva clave de root, usa el siguiente comandó.

`su`

para salirte del login como super usuario, escribe:

`exit`

Ten mucho cuidado cuando haces login como root, pues puedes eliminar archivos importantes del sistema que podrían dejarlo inutilizable etc. Una de las razones de ponerle una clave a root es por la razón de que tu computadora tenga que ser usada por más usuarios que deban tener permisos SE súper usuario root para modificar el sistema, o razones administrativas que eviten usar repetidamente el comando sudo.