

## Linux. Tema 1.

### Historia:

- 1969 La empresa AT&T desarrolla el sistema operativo UNIX y vendido posteriormente a Novell.
- 1983 Richard Stallman comienza el proyecto GNU (GNU is Not Unix) para crear un sistema operativo tipo UNIX pero de software libre.
  - Software libre: Aquel software, que una vez adquirido (no tiene porque ser gratuito), puede ser usado, copiado, modificado y redistribuido.
- 1985 Microsoft publica Windows, un sistema operativo con interfaz gráfica de usuario (GUI) para su propio sistema operativo MS-DOS.
- 1991 Linus Torvald comienza a programar el sistema operativo Linux (Linus + UNIX). El código era totalmente nuevo, pero emulaba el funcionamiento del sistema operativo MINIX, que era un clon de UNIX.
- 1992 Se juntan el proyecto Linux y GNU → GNU/Linux.
- 2001 Se lanza el primer sistema operativo MAC (MAC OS X) con interfaz de escritorio. MAC está basado en UNIX.

### Características:

- Software Libre → Cualquiera puede usarlo o descargarlo.
  - Licencias GPL (General Public License) de GNU
- Código Abierto → Cualquiera puede ver y modificar el código.

### Distribuciones:

- Gratuitas:
  - Ubuntu, CentOS, Mint, Fedora, Knoppix, OpenSUSE.
  - En el caso de Ubuntu sacan distribuciones LTS que tienen mayor tiempo de actualizaciones. Las versiones indican el año y el mes en que se saca dicha versión. Por ejemplo 17.04 (Año 2017, Abril).
- Pago: RedHat o SUSE.
  - Se paga por el soporte, no por el software en sí.
  - Las distribuciones de pago también suelen tener sus versiones gratuitas.

### Usuarios:

- Comunes:
  - Son los usuarios habituales pertenecientes a usuarios del sistema.
  - En ubuntu tienen un UID de 1000 en adelante.
- Super Usuario: ROOT.
  - Es el equivalente al administrador en windows.
  - Para loguearse como root hemos de ejecutar el comando “sudo su”.
  - Tiene permisos de “rwx” de cualquier fichero.
  - En ubuntu tiene el UID 0.
- Especiales:
  - Usuarios relacionados con procesos o hosts, como una impresora.
  - No se puede logear como dicho usuario, ni siquiera siendo root.
  - En ubuntu tienen UID entre 1-999

## SHELL / BASH:

- Es un programa que se ocupa de interpretar y ejecutar las instrucciones a través del terminal o consola. Interprete de línea de comandos.
  - `echo $SHELL` muestra el shell que se está usando en el terminal.

## Rutas:

- Existen dos tipos de rutas:
  - Rutas absolutas: Van desde el nodo raíz (/) hasta el directorio destino:
    - `/home/salvu/Escritorio`
  - Rutas relativas: Son aquellas rutas que parten desde el directorio actual.
- El comando “`pwd`” devuelve la ruta absoluta actual.
- Con “`cd ..`” podemos subir un nivel en el árbol de directorios.

## Directorios:

- `/bin` – Comandos / Programas esenciales del sistema.
- `/dev` – Controladores de dispositivos como discos duros.
- `/etc` – Ficheros de configuración.
- `/home` – Contenido de usuarios (Ficheros, Películas, Música, ...).
- `/usr` – Aplicaciones de usuarios.
- `/media` – Puntos de montaje para dispositivos: (CD, DVD, USB...).
- `/mnt` – Puntos de montaje de carpetas en red o sistemas de ficheros temporales, por ejemplo una carpeta compartida.

## Permisos:

- Podemos visualizar los permisos de un fichero mediante el comando “`ls -l`”
- La primera letra indica el tipo de archivo (-) o directorio (d)
- Los siguientes 9 letras son realmente 3 agrupaciones de 3 tipos de permisos.
  - Usuario Propietario (RWX).
  - Usuarios pertenecientes al mismo grupo que el usuario propietario (RWX).
  - Resto de usuarios (RWX).
- R (Lectura) W(Escritura) X(Ejecución).
- El comando para cambiar los permisos es `chmod`.
- Ejemplo: `-rwxrw-r--`
  - Se trata de un fichero debido al primer -.
  - Usuario Propietario puede leer, escribir y ejecutar.
  - Usuarios del mismo grupo que el usuario propietario pueden leer y escribir pero no ejecutar.
  - El resto de usuarios únicamente pueden leer.
- El usuario root tiene permisos totales.

## Redirección de comandos:

- En linux hay 3 tipos de redirecciones:
  - `stdout` → rutina que redirige la salida por pantalla.
  - `Stdin` → rutina que redirige la entrada de información desde la pantalla.
  - `Stderr` → rutina que redirige la información de error del sistema.
- Estas rutinas se pueden sobrescribir para que no se redirijan a un fichero o variable:
  - `>`, `>>` redirige `stdout`.
    - `>` sobrescribe información.
    - `>>` concatena información.

- Ambos comandos generan el fichero en caso de no existir.
- Ejemplo: `echo Hola Mundo > fichero.txt`.
- `<`, redirige stdin
  - Puede leer un contenido y mostrarlo por pantalla.
  - Ejemplo: `cat < /home/salva/fichero.txt`.
- `2>`, `2>>`, redirige la salida stderr.
  - Podemos volcar en un fichero la información relacionada con errores de sistema en nuestra aplicación.

#### Tuberías:

- Permite concatenar varios comandos, de forma que la salida del primer comando es la entrada del segundo y así sucesivamente.
- Ejemplo: `cat /etc/passwd | wc -l` → Este comando pasa al comando `wc` todo el contenido del fichero `passwd`, y acto seguido cuenta las líneas.

#### Procesos y Aplicaciones:

- Programa es un conjunto de instrucciones almacenado en el disco duro.
- Un proceso es un programa en ejecución. Todo proceso tiene un identificador (PID).
  - Los procesos suelen tener una relación padre-hijo, de tal forma que si un proceso padre muere también lo hacen los procesos hijo.
  - El primer proceso que se crea cuando arrancamos Linux es INIT (PID 1)