

Unidad 7: GNU/Linux

1. [Ajustes despues de la instalacion de Ubuntu](#)
2. [Sistema de archivos en Linux](#)

Ajustes despues de la instalacion de Ubuntu

Habilitar el usuario root

```
sudo passwd root
```

Cambiar el nombre al equipo

Para cambiar el nombre al equipo debemos editar el archivo `/etc/hostname` y eliminamos el contenido y introducimos el nombre que queramos como host. A continuacion deberemos de cambiar el contenido `/etc/hosts` que contendra algo como lo siguiente:

Iniciar el sistema en manera texto

```
sudo systemctl set-default multi-user.target
```

Y para volver a interfaz grafica:

```
sudo systemctl set-default graphical.target
```

Si queremos iniciar la interfaz grafica desde la manera texto:

```
systemctl start graphical.target  
systemctl start gdm3.service
```

Sistema de ficheros en linux

Caracteristicas generales del sistema de ficheros de linux

Directorio raiz

El sistema de ficheros es la manera en la que el sistema operativo organiza los ficheros en el disco duro, gestionandolo de manera que los datos esten de forma estructurada y sin errores.

El sistema de ficheros de linux tiene una estructura jerarquica, donde el directorio principal o *directorio raiz* es el directorio `/` del que cuelga toda la estructura del sistema.

Por tanto en Linux existe solo un arbol de directorios, no como en Windows que tenemos uno por particion (C:, D:, ...)

Los inodos o nodes-i

En Linux, a cada archivo o directorio se le asigna un numero identificador unico llamado inodo.

De esta manera, existe una tabla de inodos en la cual hay una entrada para cada inodo en la cual se guarda toda la informacion importante del fichero: propietario, grupo, permisos, tipo de fichero, data de creacion, data ultima de

modificación, dirección de los bloques de disco donde se encuentra el fichero, número de enlaces, tamaño, directorio al que pertenece.

Es importante recalcar que no se guarda el contenido del directorio.

Tipos de sistemas de archivos

Linux soporta una gran variedad de sistemas de ficheros, desde sistemas basados en discos, como pueden ser **ext2**, **ext3**, **ext4**, **ReiserFS**, **XFS**, **JFS**, **UFS**, **ISO9660**, **FAT**, **FAT32** o **NTFS** a sistemas de ficheros que sirven para comunicar equipos en la red de diferentes sistemas operativos, como **NFS** (utilizado para compartir recursos entre equipos Linux) o **SMB** (para compartir recursos entre máquinas Linux y Windows)

Directorios importantes de linux

- **/bin** y **/usr/bin**: contienen la mayoría de ficheros ejecutables y comandos comunes de Linux que pueden usar todos los usuarios.
- **/boot**: contiene los ficheros necesarios para el arranque del sistema
- **/dev**: almacena los "ficheros especiales" que representan los dispositivos de E/S. En realidad no son "ficheros" propiamente dichos sino que es la forma en que Linux implementa los controladores de dispositivos.
- **/etc**: contiene los ficheros de configuración del sistema. Solo - **root** puede modificarlos.
- **/home**: donde se almacena el *directorio home* o carpeta personal de cada uno de los usuarios.
- **/lib** y **/usr/lib**: contienen librerías compartidas del sistema.
- **/media**: cuando se monta automáticamente un CD-ROM, o pendrive o disco duro externo, se crea aquí su subdirectorio. PE: */media/usbdisk*
- **/mnt**: es el directorio por defecto para realizar el montaje de otros dispositivos de almacenamiento.
- **/opt**: directorio opcional donde se pueden instalar aplicaciones, además de en */usr*. En algunas distribuciones no existe este directorio.
- **/root**: es el home del usuario root
- **/sbin** y **/usr/sbin**: contienen comandos y ejecutables de tareas de administración que en su mayoría solo puede usar, evidentemente, el usuario root.
- **/usr**: almacena las aplicaciones de uso general para todos los usuarios, por lo cual, si hay muchos paquetes instalados, puede ocupar mucho espacio.
 - **/usr/bin**: contiene programas y comandos para los usuarios
 - **/usr/share**: datos compartidos independientes de la máquina, como documentación de programas, imágenes para el Escritorio de Linux...

Tipos de archivos

- **Archivos regulares**: Contienen la información con la cual trabaja cada usuario. Son los archivos ordinarios de datos.
- **Directorios**: Son archivos especiales que contienen referencias a otros archivos o directorios.

- **Hard links:** No es específicamente una clase de archivo sino un segundo nombre que se le da a un archivo.
- **Soft links:** También se utilizan para asignar un segundo nombre a un archivo. La diferencia con los enlaces duros es que los simbólicos solo hacen referencia al nombre del archivo original, mientras que los duros hacen referencia al inodo en el cual están situados los datos del archivo original. Sería como los "accesos directos" de Windows.
- **Archivos especiales:** Suelen representar dispositivos físicos, como unidades de almacenamiento, impresoras, terminales, etc. En Linux, todo dispositivo físico que se conecte al ordenador está asociado a un archivo. Pueden ser de dos tipos: de bloques o caracteres. Un disco duro es un dispositivo de bloque y una impresora es un dispositivo de carácter.
- Otro tipo de archivo son las *cañerías (pipes)* o *los sockets*:

Indicador	Descripción
-	Archivo ordinario
d	Directorio
	Enlace simbólico
b / c	Archivo especial (c=caracteres y b=bloques)
p	Tubería
s	Socket

Ficheros y directorios ocultos

Los ficheros ocultos en Linux son aquellos que empiezan por `.`

Permisos y atributos

```
drwxr-xr-- 2 juvndv juvndv 4096 2005-02-16 14:47 tmp
```

En la línea anterior de código identificamos lo siguiente:

- ***drwxr-xr--***: Permisos del archivo
 - ***d***: Tipo de archivo
 - ***rw***: Permisos de propietario
 - ***r-x***: Permisos de grupo
 - ***r--***: Permisos de otros
- ***2***: Enlaces duros
- ***juvndv***: Propietario
- ***juvndv***: Grupo
- ***4096***: Tamaño en bytes
- ***2005-02-16 14:47***: Fecha y hora de la última modificación

- **tmp**: Nombre

Permisos y modificar permisos

En linux los permisos de archivos son los siguientes:

- **r**: lectura
- **w**: escritura
- **x**: ejecucion

Modificar permisos en manera comando

```
chmod o|g|o|a =|+|- r|w|x fichero(s) |directorio(s)

chmod o=rwx, g=rx, o=- ej1.txt
chmod g-wx ej1.txt
```

Modificar permisos en manera octal

Siendo los permisos del propietario **rw-** se puede convertir al octal **110** que en decimal es 6, si aplicamos esto a los tres campos que se le aplican los permisos, es decir, propietario, grupo y otros y queremos aplicar los siguientes permisos:

- **p**: rwx
- **g**: rw-
- **o**: r-x

Convertimos cada uno a valor octal:

- **p**: 7
- **g**: 6
- **o**: 5

y con el siguiente formato

```
chmod XXX fichero(s) |directorio(s)
```

ejecutamos el comando

```
chmod 765 ej1.sh
```

Permisos especiales

- **SUID**: El bit SUID activo en un archivo significa que el que lo ejecute tendra los mismo permisos que el que creo el archivo.

Ejemplo:

```
$ ls -l /bin/el seu
-rwsr-xr-x 1 root root 31012 2016-03-04 07.49 /bin/la sua
```

```
chmod 4775 <fichero>
chmod o+s <fichero>
```

```
chmod o-s <fichero>
```

- **SGID**: Es el mismo que el SUID pero a nivel de grupo. Es decir, todo archivo que tenga activo el SGID, en ser ejecutado, tendra los privilegios del grupo al cual pertenecen

Ejemplo:

```
$ ls -l
drwxrws--- 2 mar mar 4096 2016-03-04 21.27 compartit
```

```
chmod 2775 <directorio>
chmod g+s <directorio>
chmod g-s <directorio>
```

- **Sticky bit**: Se utiliza para permitir que cualquiera pueda escribir y modificar sobre un archivo o directorio, pero que solo su propietario o root puedan eliminarlo.

Ejemplo:

```
drwxr-xr-t 13 root root 4096 2016-04-24 20.55 tmp
```

```
chmod 1755 <directorio>
chmod o+t <directorio>
chmod o-t <directorio>
```