

Introducción a Linux

Módulo 2



Objetos del sistema de archivos

Objetos del sistema de archivos

Los sistemas de archivos que se van a explicar en este apartado, poseen objetos que se agrupan en un **árbol jerárquico**. Esta organización hace que puedan contener diversos archivos; algunos de ellos con el mismo nombre. Sin embargo, al estar organizados en forma de árbol jerárquico, es posible realizar diversas tareas; entre ellas, “ver” nombres de archivos iguales, (¿pero cada objeto es único?). Estos objetos se organizan mediante tablas propias del sistema de archivos, que tienen diferentes propiedades. Ahora se hará foco en los directorios y los archivos.



Directorios y archivos

Los **directorios** son contenedores que van a incluir la información de los objetos, que pueden ser otros directorios u otros archivos.

En Linux, el directorio de mayor jerarquía es la raíz “/”, a partir de ahí surgen todos los demás directorios y archivos.

Un archivo, por otro lado, es el que va a contener información; en cambio, un directorio, nos da un orden de cómo vamos a poder ver todos los archivos y/o directorios.

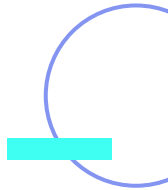
Para poder especificar una ruta se antepone “/” porque todo nace de ahí; en cambio, si quisiéramos nombrar un archivo de nuestro **home**, la ruta completa del archivo sería la siguiente:
`/home/nombre_usuario/test.sh`

No debemos confundir `root filesystem` con `root home directory`.



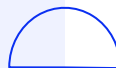
Inodos

El **sistema de archivos identifica a los objetos mediante inodos** que contienen información sobre ellos. Cada uno tiene distintas propiedades que dependiendo de su uso, va a cambiar el valor que contenga. Por ejemplo: datos que posee, ubicación, fecha de modificación, tamaño, datos seguridad, etc.



El comando `stat` es de gran utilidad para mostrar esta información, por ejemplo:

```
# stat /etc/resolv.conf
  Fichero: /etc/resolv.conf
    Tamaño: 125          Bloques: 8          Bloque E/S: 4096    fichero regular
Dispositivo: fd00h/64768d      Nodo-i: 1050548      Enlaces: 1
Acceso: (0644/-rw-r--r--)  Uid: (      0/      root)  Gid: (      0/      root)
    Acceso: 2019-03-21 14:32:29.532283100 -0300
Modificación: 2019-03-21 14:32:29.432282543 -0300
    Cambio: 2019-03-21 14:32:29.439282582 -0300
  Creación: -
```



Es decir, este comando nos **informa sobre las propiedades de un archivo**:

- Nombre.
 - Tamaño.
 - Tipo de archivo.
 - El número de i-nodo.
 - La cantidad de enlaces, que en archivos regulares refiere a cuántos archivos tienen el mismo número de i-nodos.
 - Permisos y propietarios.
- Marcas de tiempo.
 - Acceso (**atime**): la última vez que se accedió al archivo.
 - Modificación (**mtime**): la última vez que se modificó el contenido de un archivo.
 - Cambio (**ctime**): la última vez que se cambiaron metadatos del archivo, por ejemplo, los permisos.
 - Creación: la fecha de creación tradicionalmente no se usa en los sistemas de archivos en Unix.

Comandos ls y cd

El comando ls

El comando `ls` se utiliza para **mostrar el contenido de directorios**, como también para **ver propiedades** de los distintos archivos.

Sintaxis

```
ls [ opciones ] [archivo1] [archivo2] [archivoN]  
ls [opciones] [directorio1] [directorio2]
```

Las opciones más utilizadas son:

- a** Muestra todos los archivos, incluso los ocultos.
- l** Muestra el tipo de archivo, permisos, propietarios, tamaño, fecha de modificación.
- h** Muestra el tamaño de los archivos en una unidad fácil de leer.
- r** Ordena los archivos de manera inversa.
- t** Ordena los archivos por fecha de modificación.



Ejemplo

```
# ls -ltr /etc/X11/  
total 24  
drwxr-xr-x. 2 root root 4096 feb  3  2016 applnk  
drwxr-xr-x. 2 root root 4096 ago 11  2016 fontpath.d  
-rw-r--r--  1 root root  493 dic 14 08:27 Xresources  
-rw-r--r--  1 root root  547 dic 14 08:27 Xmodmap  
drwxr-xr-x. 2 root root 4096 mar 15 15:38 xorg.conf.d  
drwxr-xr-x. 5 root root 4096 abr  8 20:51 xinit
```

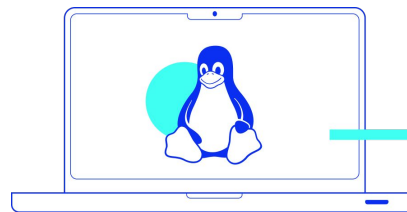


El comando cd

Este comando se utiliza para **avanzar** o **retroceder** niveles de directorios.

Ejemplo

```
# cd dir # (cambio a un directorio hijo)
# cd .. # (retrocedo)
# cd # (me lleva a mi home)
```



**¡Sigamos
trabajando!**