

UNIDAD DIDACTICA 6

PERMISOS DE LOS ARCHIVOS

1. INTRODUCCIÓN.

SISTEMA DE FICHEROS

- ⌘ El sistema de ficheros es la organización lógica del disco que nos permite almacenar la información en forma de ficheros, de un modo totalmente transparente, sin tener que preocuparnos de pistas, sectores, cilindros.
- ⌘ Cada partición de disco duro, o disquete, dispositivo de almacenamiento USB o cdrom, debe tener un sistema de ficheros si queremos almacenar información.
- ⌘ Una partición sin sistema de ficheros no permite almacenar información.
- ⌘ Cada S.O. posee su propia organización lógica del disco para poder almacenar la información y usarla, y tiene la posibilidad de usar particiones propias de otros sistemas.

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

- ☞ ext2 ➡ Extendido 2. Sistema de ficheros nativo de linux. Soporta características avanzadas: propietarios, permisos, enlaces, etc.
- ☞ ext3 ➡ Extendido 3 (con journaling). Similar a ext2 pero con transacciones para evitar que apagados accidentales puedan deteriorar el sistema de ficheros.
- ☞ ext4 ➡ Extendido 4. Sus principales ventajas radican en su eficiencia (menor uso de CPU, mejoras en la velocidad de lectura y escritura) y en la ampliación de los límites de tamaño de los ficheros, ahora de hasta 16TB, y del sistema de ficheros, que puede llegar a los 1024PB (PetaBytes)
- ☞ ReiserFS ➡ Es el sistema de ficheros de

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

última generación para Linux. Organiza los ficheros de tal modo que se agiliza mucho las operaciones con estos. El problema de ser tan actual es que muchas herramientas (por ejemplo, para recuperar datos) no lo soportan.

↻swap ↗ sistema de ficheros para la partición de intercambio de Linux. Todos los sistemas Linux necesitan una partición de este tipo para cargar los programas y no saturar la memoria RAM cuando se excede su capacidad.

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

↻ vfat ➡ Ampliación del sistema de ficheros msdos, con soporte para nombres largos de ficheros. Los tipos FAT12, FAT16 y FAT32 sólo tienen características monousuario: no admiten propietarios de ficheros y los permisos son muy limitados.

↻ NTFS ➡ sistema de ficheros de Windows. Es un sistema de ficheros con características avanzadas y sí está preparado para utilizarse en entornos multiusuario.

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

Elementos básicos de un sistema de archivos: el fichero y el directorio.

La organización lógica de los ficheros y directorios en disco es similar a las hojas y ramas en un árbol. Las ramas serán los directorios y las hojas los ficheros. Cada rama (directorio) puede contener hojas (archivos) y más ramas.

Un fichero es un conjunto de datos en disco asociado a un i-nodo. En el i-nodo se especifican entre otras cosas los permisos, el propietario y la distribución de los datos del fichero sobre el disco.

Un directorio es un fichero especial donde se asocia un nombre a un número de i-nodo. Un directorio contiene ficheros y otros directorios.

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

☞ Son 3: el superbloque, i-nodos y bloques de datos.

Superbloque

Es el 1º bloque del sistema de ficheros. Contiene la descripción general del sistema de ficheros: Tamaño, bloques libres, tamaño de la lista de i-nodos, i-nodos libres, verificaciones, etc.

i-nodos

Contienen toda la información sobre cada conjunto de datos en disco, que denominamos fichero:

- El propietario de los datos y a qué grupo pertenece
- Tamaño del fichero.
- Permisos del fichero

1. ELEMENTOS DEL SISTEMA DE FICHEROS DE LINUX

- La lista de bloques de datos en disco. Una serie de punteros o direcciones de bloques que indican donde se almacenan los datos en disco.
- Tipo de fichero: regular (fichero de datos), dispositivo (periférico), enlace (apunta a otro fichero), pipe (para intercambiar información entre procesos), directorio (contiene referencias a otros ficheros y directorios)
- Número de enlaces del fichero. El número de nombres distintos que tiene este fichero. El nombre de fichero se asocia a un i-nodo dentro de un directorio. Un mismo i-nodo pueda tener varios nombres si aparece en varios directorios o con distintos nombres.

2. PROPIETARIOS DE LOS ARCHIVOS

Linux es un sistema multiusuario: Cada i-nodo (fichero o directorio) es propiedad de un usuario y de un grupo.

Los usuarios propietarios de un archivo pueden configurar permisos sobre sus archivos. Estos permisos distinguen tres categorías de usuarios:

- El propietario del archivo. Por defecto es el usuario que crea el archivo o directorio.
- Todos los usuarios que pertenecen al grupo propietario. El grupo propietario por defecto es el grupo principal del usuario que crea el archivo o directorio.
- Los demás usuarios, que son todos los usuarios que no son ni el propietario ni miembros del grupo propietario.

2. PERMISOS DE LOS ARCHIVOS

Para cada uno de estos tres tipos de usuarios podemos definir tres permisos diferentes:

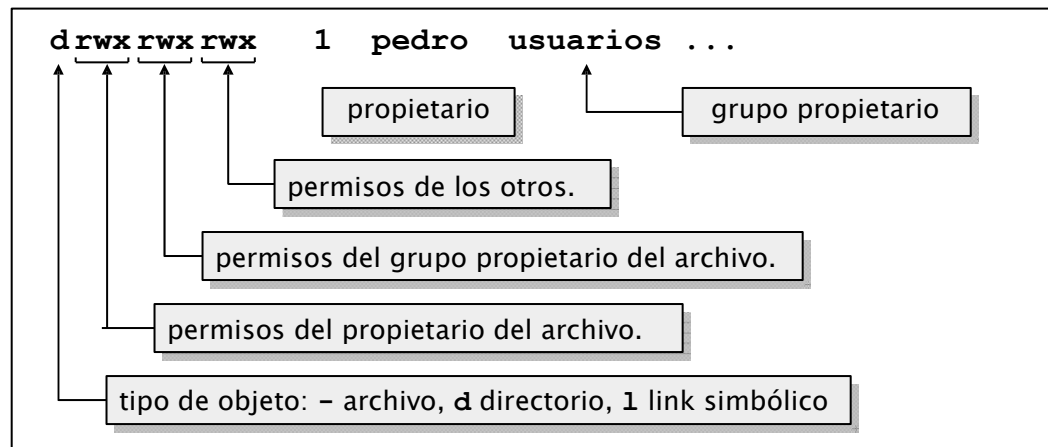
- **Permiso de Lectura** (r por Read) ➡ Para un archivo permite que se lea su contenido. Para un directorio, permite que se muestren los ficheros que contiene.
- **Permiso de Escritura** (w por Write) ➡ Para un archivo permite modificarlo o borrarlo. Para un directorio permite que el usuario añada o elimine archivos.
- **Permiso de Ejecución** (x, por eXecute) ➡ Para un archivo ejecutable permite que sea ejecutado. Para un directorio permite que un usuario lo recorra, es decir, puede entrar leer o escribir en él.

2. PERMISOS DE LOS ARCHIVOS

Cada i-nodo guarda un espacio para almacenar los permisos bajo los cuales se puede acceder a un fichero.

Usuarios	Permisos
Propietario	Lectura
Grupo	Escritura
Resto usuarios	Ejecución

En total tendremos 9 bits que indican los distintos permisos con el siguiente orden: usuario, grupo, otros.



La visualización de los permisos de los archivos y directorios se realiza mediante el comando `ls -l`

2. PERMISOS DE LOS ARCHIVOS

Los primeros diez caracteres del resultado del comando `ls -l` representan el tipo de archivo y los derechos asociados al mismo.

```
elara@elara-desktop:~$ ls -l
total 44
-rw-r--r-- 1 root  root   13 2009-11-27 09:03 edu
-rw-r--r-- 1 elara elara 968 2009-11-05 07:12 error.txt
```

- ⌘ El primer carácter indica el tipo de objeto:
 - si es un archivo, `d` si es un directorio, `l` si es un link simbólico.
- ⌘ Los nueve caracteres que siguen representan los permisos asociados con el archivo:
 - ⌘ los 3 primeros se corresponden al propietario del archivo, los 3 siguientes se aplican a todos los usuarios miembros del grupo propietario, los tres últimos se aplican a los otros.
- ⌘ Finalmente también se informa del dueño del archivo y el nombre del grupo dueño

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

❧ **chmod**

Sintaxis: `chmod [opciones] modo files | directory...`

Permite modificar los permisos del archivo y/o directorio especificado en files y/o directory.

Opciones:

- R Establece los permisos de todos los archivos y subdirectorios incluidos en el directorio especificado.
- v Muestra todas las acciones efectuadas por el comando. Reporta los nuevos permisos, aunque no se haya producido ninguna modificación de los permisos de un archivo o directorio.
- c Como la opción -v, pero únicamente reporta los archivos o directorios cuyos permisos han cambiado debido a la nueva asignación.

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

Los permisos se expresan con una secuencia de expresiones separadas por comas. Una expresión toma la forma:

[categoría] <+|-|=> <permisos>.

Las categorías pueden ser una o más de:

u	user	Hace referencia a los permisos del usuario propietario.
g	group	Hace referencia a los permisos del grupo propietario.
o	others	Hace referencia a los permisos de los otros, los usuarios que no son el usuario propietario y usuarios que no pertenecen al grupo propietario.
a	all	Hace referencia a los permisos de las tres referencias anteriores simultáneamente.

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

Sino se especifica categoría alguna los cambios se aplicarán a todas las categorías. Un + garantiza el permiso y un - lo deniega. El signo = garantiza los permisos que están en la lista y deniega los que no están en la lista.

Los permisos son uno de los siguientes o una combinación de:

r	read	Lectura
w	write	Escritura
x	eXecute	Ejecución
X	eXecute	Activa el permiso de ejecución a cualquier archivo que tenga un permiso de ejecución en cualquiera de las tres categorías de usuarios (owner, group o others)

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

Representación en octal de los permisos

Los bits de permisos de los archivos se pueden agrupar en 3 cifras octales según los 3 propietarios. Por ejemplo el valor 751 indicará:

usuario 7 = 111 = rwx

grupo 5 = 101 = r-x

otros 1 = 001 = --x

} Los permisos quedan rwxr-x--x

Ejemplos chmod

- Deniega al propietario el permiso de lectura del archivo test.

chmod u-r test  **chmod 300 test**

- Proporciona al propietario el permiso de lectura del archivo test.

chmod u+r test  **chmod 700 test**

- Proporciona al grupo propietario los permisos de lectura y escritura del archivo test.

chmod g+rw test  **chmod 060 test**

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

☞ Ejemplos chmod

- `chmod ug-x test` ➡ Deniega al propietario y al grupo propietario el permiso de ejecución del archivo `test`.
- `chmod o-rwx test` ➡ Deniega a los otros los permisos de lectura, escritura y ejecución del archivo `test`.
- `chmod ugo+rw test`
- `chmod a+rwx test` ➡ Proporciona a todos los permisos de lectura, escritura y ejecución del archivo `test`.
- `chmod u-x,g-wx,o-rwx test` ➡ Deniega al propietario el permiso de ejecución, deniega al grupo propietario los permisos de lectura y ejecución, y deniega a los otros los permisos de lectura, escritura y ejecución.
- `chmod -Rc u=rw apuntes/` ➡ Al usuario propietario le proporciona permisos de lectura y escritura y le deniega el permiso de ejecución al directorio `apuntes/` y a todo su contenido.

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS

❧ **chown**

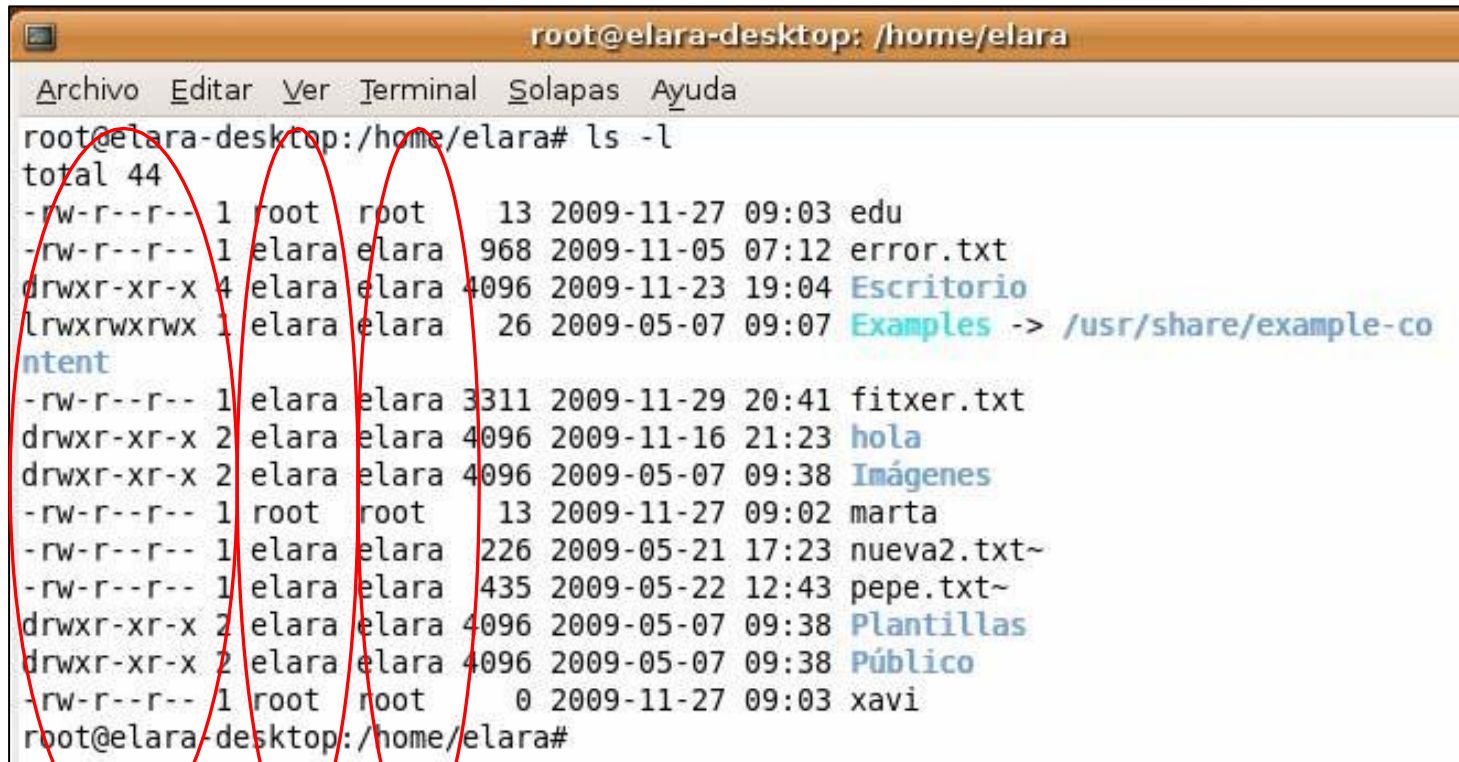
Sintaxis: `chown [opciones] usuario files | directory...`

Permite cambiar el usuario propietario de uno o más archivos o directorios especificados en files y/o directory. Únicamente el usuario root puede modificar el propietario de un archivo.

❧ **chgrp**

Sintaxis: `chgrp [opciones] grupo files | directory...` Permite cambiar el grupo propietario de uno o más archivos o directorios especificados en files y/o directorios. Únicamente el usuario root y los usuarios que pertenecen al nuevo grupo pueden cambiar el grupo propietario.

3. ORDENES PARA MANIPULAR LOS PERMISOS DE LOS ARCHIVOS



```
root@elara-desktop: /home/elara
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
root@elara-desktop:/home/elara# ls -l
total 44
-rw-r--r-- 1 root root 13 2009-11-27 09:03 edu
-rw-r--r-- 1 elara elara 968 2009-11-05 07:12 error.txt
drwxr-xr-x 4 elara elara 4096 2009-11-23 19:04 Escritorio
lrwxrwxrwx 1 elara elara 26 2009-05-07 09:07 Examples -> /usr/share/example-co
ntent
-rw-r--r-- 1 elara elara 3311 2009-11-29 20:41 fitxer.txt
drwxr-xr-x 2 elara elara 4096 2009-11-16 21:23 hola
drwxr-xr-x 2 elara elara 4096 2009-05-07 09:38 Imágenes
-rw-r--r-- 1 root root 13 2009-11-27 09:02 marta
-rw-r--r-- 1 elara elara 226 2009-05-21 17:23 nueva2.txt~
-rw-r--r-- 1 elara elara 435 2009-05-22 12:43 pepe.txt~
drwxr-xr-x 2 elara elara 4096 2009-05-07 09:38 Plantillas
drwxr-xr-x 2 elara elara 4096 2009-05-07 09:38 Público
-rw-r--r-- 1 root root 0 2009-11-27 09:03 xavi
root@elara-desktop:/home/elara#
```

The image shows a terminal window with the command `ls -l` executed. The output lists files and directories with their permissions, owner, group, size, date, and name. Three red circles are drawn around the first three columns of the output: permissions (e.g., `-rw-r--r--`), owner (e.g., `root`), and group (e.g., `root`).

chmod chown chgrp

EJERCICIO PRACTICO DE GESTIÓN DE USUARIOS Y PERMISOS

Paso 1. Desde nuestro usuario, creamos 3 ficheros:

```
echo fichero1.txt > fichero1.txt
```

```
echo fichero2.txt > fichero2.txt
```

```
echo fichero3.txt > fichero3.txt
```

```
elara@elara-desktop:~$ ls -l f*
-rw-r--r-- 1 elara elara 13 2010-05-06 15:00 fichero1.txt
-rw-r--r-- 1 elara elara 13 2010-05-06 15:00 fichero2.txt
-rw-r--r-- 1 elara elara 13 2010-05-06 15:01 fichero3.txt
elara@elara-desktop:~$ █
```

Paso 2. Modificamos los permisos de los ficheros mediante chmod:

```
chmod 700 fichero1.txt
```

```
chmod 770 fichero2.txt
```

```
chmod 777 fichero3.txt
```

```
elara@elara-desktop:~$ ls -l f*
-rwx----- 1 elara elara 13 2010-05-06 15:00 fichero1.txt
-rwxrwx--- 1 elara elara 13 2010-05-06 15:00 fichero2.txt
-rwxrwxrwx 1 elara elara 13 2010-05-06 15:01 fichero3.txt
elara@elara-desktop:~$
```

EJERCICIO PRACTICO DE GESTIÓN DE USUARIOS Y PERMISOS

Paso 3. Contamos con los siguientes usuarios:

Usuario	Grupos
elara	elara
usu01	elara
usu02	usu02 y elara (como secundario)

```
usermod -g elara usu01
```

```
usermod -G elara usu02
```

Paso 4. Indica que ficheros pueden visualizar los 3 usuarios:

```
elara@elara-desktop:~$ ls -l f*
-rwx----- 1 elara elara 13 2010-05-06 15:00 fichero1.txt
-rwxrwx--- 1 elara elara 13 2010-05-06 15:00 fichero2.txt
-rwxrwxrwx 1 elara elara 13 2010-05-06 15:01 fichero3.txt
elara@elara-desktop:~$
```

Elara ➡ Todos

Usu01 ➡ 2 y 3

Usu02 ➡ 2 y 3

EJERCICIO PRACTICO DE GESTIÓN DE USUARIOS Y PERMISOS

Paso 5. Cambiamos la pertenencia de los ficheros:

```
sudo chown usu01 fichero1.txt
```

```
sudo chown usu01 fichero2.txt
```

```
sudo chgrp usu02 fichero3.txt
```

```
root@elara-desktop:/home/elara# ls -l fi*  
-rwx----- 1 usu01 elara 13 2010-05-06 15:00 fichero1.txt  
-rwxrwx--- 1 usu01 elara 13 2010-05-06 15:00 fichero2.txt  
-rwxrwxrwx 1 elara usu02 13 2010-05-06 15:01 fichero3.txt
```

Paso 6. Indica que ficheros pueden visualizar los 3 usuarios:

Elara ➡ 2 y 3

Usu01 ➡ Todos

Usu02 ➡ 2 y 3

Ejercicios. GESTIÓN DE USUARIOS Y PERMISOS

Paso 1. Crear los siguientes grupos de usuarios:

profesoresm

asi2m

sti2m

todosm

Paso 2. Editar el archivo que contiene la información de los grupos del sistema y pegar en el dossier de la práctica las líneas asociadas a los nuevos grupos creados. ¿Qué es el GID de un grupo? Indica el GID del grupo profesores.

Ejercicios. GESTIÓN DE USUARIOS Y PERMISOS

Paso 3. Crear el usuario pgarciam con las siguientes características:

- login: pgarciam
- Contraseña: UPC
- Nombre Completo: Pedro Garcia
- Grupo Base: profesoresm
- Otros Grupos: asi2m, sti2m, todosm

Paso 6. Indica el UID del usuario pgarciam. ¿Qué es el UID?

Paso 7. Crear el usuario lcasasm con las siguientes características:

- Login: lcasasm
 - Contraseña: UPC
 - Nombre Completo: Luis Casas
 - Grupo Base: asi2m
 - Otros Grupos: todosm
-

Ejercicios. GESTIÓN DE USUARIOS Y PERMISOS

Paso 8. Crear el usuario todosm con las siguientes características:

- Login: todosm
- Contraseña: UPC
- grupo base: todosm
- Impedir que el usuario pueda iniciar sesión en el ordenador.

Paso 9. Crear el usuario oramirezm con las siguientes características:

- Login: oramirezm
- Contraseña: UPC
- Nombre Completo: Oscar Ramirez
- Grupo Base: stim
- Otros Grupos: todosm

Ejercicios. GESTIÓN DE USUARIOS Y PERMISOS

Paso 10. Mostrar las líneas del archivo que contiene la información de los usuarios del sistema referentes a los usuarios creados en la práctica.

Paso 11. Mediante la orden `groups` comprobar a que grupos pertenecen los cuatro usuarios creados.

Paso 12. Editar otra vez el archivo que contiene la información de los grupos y comentar la información añadida después de crear los usuarios.

Paso 13. Modificar los permisos del directorio personal del usuario `pgarciam` para que únicamente él puede leer y escribir el contenido del directorio.

Ejercicios. GESTIÓN DE USUARIOS Y PERMISOS

Paso 14. Modificar los permisos del directorio personal del usuario lcasasm (para que se cumplan las siguientes condiciones:

- El usuario lcasasm puede leer y escribir en el directorio.
- Los usuarios del grupo profesoresm también pueden leer y escribir en el directorio.
- Los otros usuarios del grupo asi2m no pueden leer ni escribir.
- Los otros usuarios no pueden leer ni escribir.

Paso 15. Modificar los permisos del directorio personal del usuario oramirezsm para que se cumplan las siguientes condiciones:

- El usuario oramirezsm pueden leer y escribir en el directorio.
- Los usuarios del grupo sti2m pueden leer el contenido del directorio.
- Los demás usuarios pueden leer el contenido del directorio.