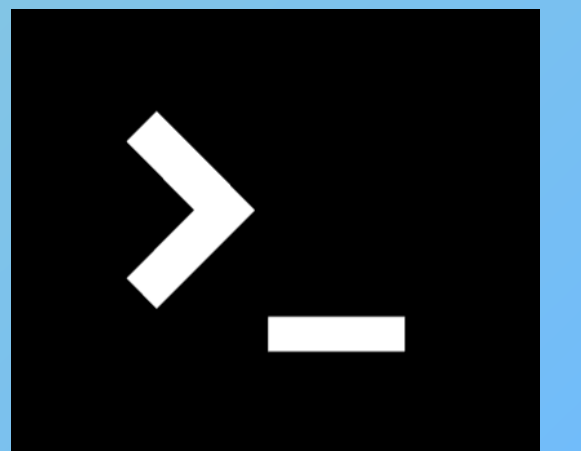


# Linux y Terminal

**Clase 17**



(propietario/owner) y a un grupo al que tiene que pertenecer dicho usuario.

- GNU/Linux nos permite modificar el propietario y el grupo de los ficheros.
- **Comando chown**
  - Permite cambiar el propietario y el grupo al que pertenece un fichero o directorio.
  - Sintaxis:
    - `chown [opciones] <user>:<group> <filename>`
  - Opciones:
    - `-R`
      - Cambia el propietario y el grupo de forma recursiva, es decir, tanto al directorio como a todos sus subdirectorios y ficheros.
  - Ejemplo:
    - `chown alumno:alumno fichero.txt`
    - `chown -R alumno:alumno directorio`
- **Comando chgrp**
  - Cambia el grupo al que pertenece un fichero o directorio
  - Sintaxis:
    - `chgrp <nombre_grupo> <fichero/directorio>`
  - Ejemplos:
    - `sudo chgrp grupo1 texto.txt`
    - `sudo chgrp -R grupo1 directorio`

#### 5.4. Cambiar los permisos a ficheros y directorios

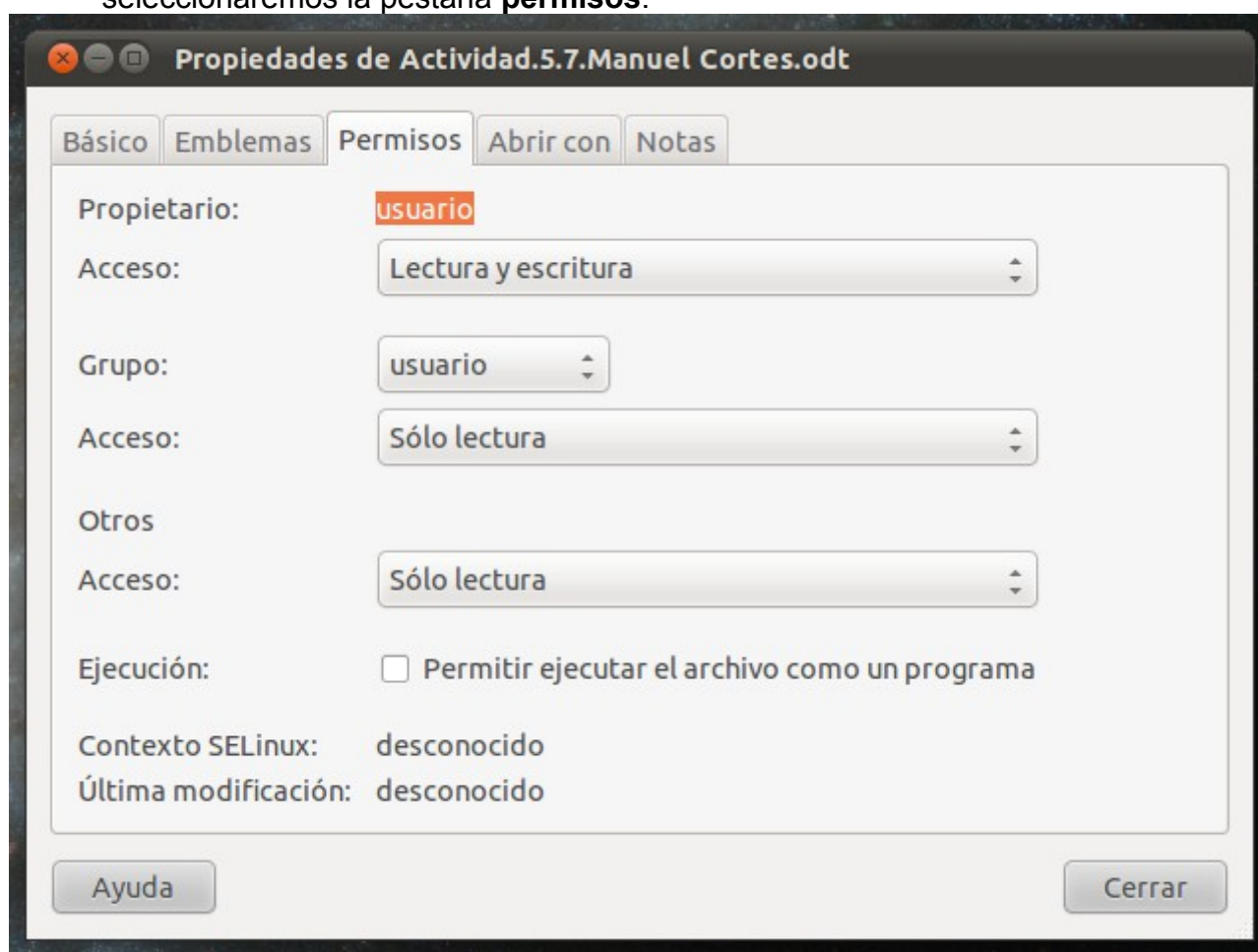
- Todos los archivos y directorios, como pertenecen a unos usuarios y grupos hacen necesario una política de permisos para manipularlos.
- GNU/Linux utiliza 3 grupos de permisos:
  - Los permisos del propietario del archivo (u)
  - Los permisos de los usuarios que pertenecen al mismo grupo del propietario (g)
  - Los permisos del resto de los usuarios (o)

Permisos	Ficheros	Directorios
<b>Lectura (r)</b>	<i>Poder ver su contenido</i>	<i>Poder listar su contenido</i>
<b>Escritura (w)</b>	<i>Poder modificar su contenido y/o eliminar el fichero</i>	<i>Poder crear/eliminar ficheros del directorio y/o el propio directorio</i>
<b>Ejecución (x)</b>	<i>Significa que se trata de un</i>	<i>Poder entrar dentro de él</i>

	<i>ejecutable (fichero binario) o que contiene un grupo de comandos a ejecutar (shell script)</i>	
--	---	--

#### 5.4.1. Permisos en modo gráfico

- Para ver o cambiar los permisos de un fichero en modo gráfico haremos click con el botón derecho sobre el archivo, entraremos en **propiedades** y después seleccionaremos la pestaña **permisos**.



#### 5.4.2. Permisos en modo texto

- Para comprobar los permisos de cualquier archivo en modo texto usaremos el comando `ls -l`
- Comando `chmod`**
  - Este comando cambia los permisos de un archivo o directorio, así como del contenido de un directorio utilizando la opción `-R` (Recursivo)
  - Sintaxis:
    - `chmod [permisos [permisos]...] archivos`

- Tipos de ficheros o directorios:
  - Al listar el contenido de un directorio mediante el comando “ls -l” se puede observar los siguientes campos divididos en columnas:
    - -rw-rw-r-- 1 alumno alumno 11 ene 28 2014 texto.txt
    -
  - d → Indica que es un directorio
  - - → Indica que es un fichero regular
  - l → Indica que es un enlace
- Permisos:
  - Los permisos se pueden añadir o quitar indicando lo siguiente:
    - chmod {a, u, g, o } {+ , - } { r, w, x, X} archivo/s.

<b>A quien se le va a cambiar el permiso</b>	
<b>u</b>	La “u” indica al usuario propietario
<b>g</b>	La “g” indica al grupo
<b>o</b>	La “o” indica a los otros usuarios del sistema
<b>a</b>	La “a” indica a todos tipos de permisos
<b>Indican si queremos añadir o suprimir un permiso</b>	
<b>+</b>	Añadir permisos
<b>-</b>	Eliminar permisos
<b>“=”</b>	se asignan los permisos independientemente de los permisos que tuviera anteriormente
<b>Indican el permiso que queremos añadir o quitar</b>	
<b>r</b>	Permisos de lectura
<b>w</b>	Permisos de escritura o modificación
<b>x</b>	Permisos de ejecución
<b>X</b>	ejecución en caso de que sea un directorio

- Ejemplos en ficheros:
  - -rwx r-x r-- (permisos que hemos sacado de un fichero) f1.txt
  - chmod u-x f1.txt
    - Elimina el permiso de ejecución al usuario.
  - chmod u+r, g+w f1.txt
    - Asigna permiso de lectura al usuario y de escritura al grupo
  - chmod g-rwx f1.txt
    - Elimina todos los permisos al grupo
  - chmod o-r, o+w f1.txt
    - Elimina los permisos de lectura a otros, le añade permiso de escritura
  - chmod ugo-x texto.txt
  - chmod a=r-- texto.txt

- `chmod o=rx- texto.txt`
    - `chmod u+wx,g+w,o+w texto.txt`
  - Ejemplos en directorios:
    - `chmod -R u=rwx,g=rw,o=r prueba`
  - Linux admite otro formato de modificación de permisos mediante el comando `chmod`.
  - Para ello utiliza el sistema octal (0 al 7) para dar o quitar permisos.
  - Con la opción `-R` podemos cambiar los permisos a un directorio y a todos los elementos que hay dentro de él de forma recursiva.
  - Sintaxis:
    - `chmod [0-7][0-7][0-7] <fichero>`
    - `chmod -R [0-7][0-7][0-7] <directorio>`
      - El primer dígito octal indica los permisos del propietario (u)
      - El segundo dígito octal indica los permisos del grupo (g)
      - El tercer dígito octal indica los permisos del resto de usuarios (o)
      - El permiso de lectura (r) tiene un valor de 4
      - El permiso de escritura (w) tiene un valor de 2
      - El permiso de ejecución (x) tiene un valor de 1
  - Ejemplos:
    - `chmod 664 f1.txt`
    - `chmod -R 775 <directorio>`
- **Comando “ls -l”**
    - Al listar el contenido de un directorio mediante el comando “ls -l” se pueden observar los siguientes campos divididos en columnas:
      - `-rw-rw-r-- 1 alumno 2cf 11 ene 28 2014 texto.txt`

Información “ls -l”	Significado
-	Indica el tipo de fichero que es: <ul style="list-style-type: none"> <li>▪ <code>d</code> → Indica que es un directorio</li> <li>▪ <code>-</code> → Indica que es un fichero regular</li> <li>▪ <code>l</code> → Indica que es un enlace</li> </ul>
<code>rw-rw-r--</code>	Indica los permisos del fichero o directorio
<code>1</code>	El número que indica el número de links al fichero
<code>alumno</code>	El propietario
<code>2cf</code>	El grupo al que pertenece
<code>11</code>	El tamaño en bytes
<code>“ene 28 2014”</code>	La fecha de la última modificación
<code>Texto.txt</code>	El nombre del fichero (si es un link simbólico se muestra también el fichero al que hace referencia)

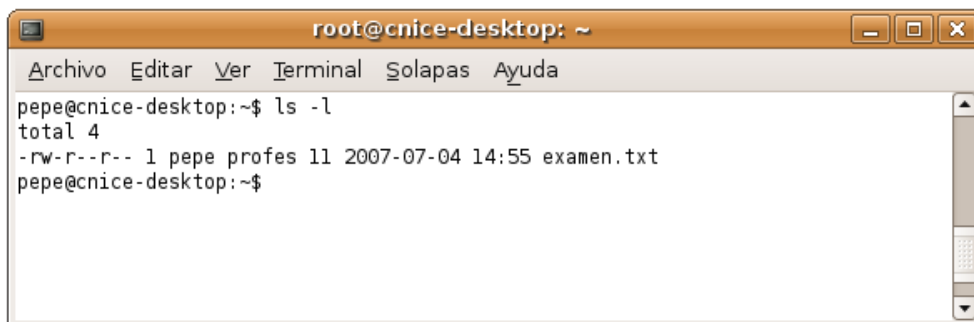
## 1.6 PERMISOS DE ARCHIVOS Y DIRECTORIOS

### 1.6.1 USUARIO PROPIETARIO Y GRUPO PROPIETARIO DE UN ARCHIVO

Anteriormente se ha comentado que en Unix todos los archivos pertenecen obligatoriamente a un usuario y a un grupo. Cuando un usuario crea un nuevo archivo, el propietario del archivo será el usuario que lo ha creado y el grupo del archivo será el grupo principal de dicho usuario.

Ejemplo, si un usuario llamado 'pepe' cuyo grupo principal es el grupo 'profesores' crea un nuevo archivo, el propietario del archivo será 'pepe' y el grupo propietario del archivo será 'profesores', o lo que es lo mismo, el archivo pertenecerá al usuario pepe y al grupo profesores. Obligatoriamente, todos los archivos del sistema pertenecen a algún usuario y a algún grupo.

Con el **comando ls** añadiendo la opción **-l** (formato largo) podemos visualizar el usuario propietario y el grupo propietario del archivo, ejemplo:



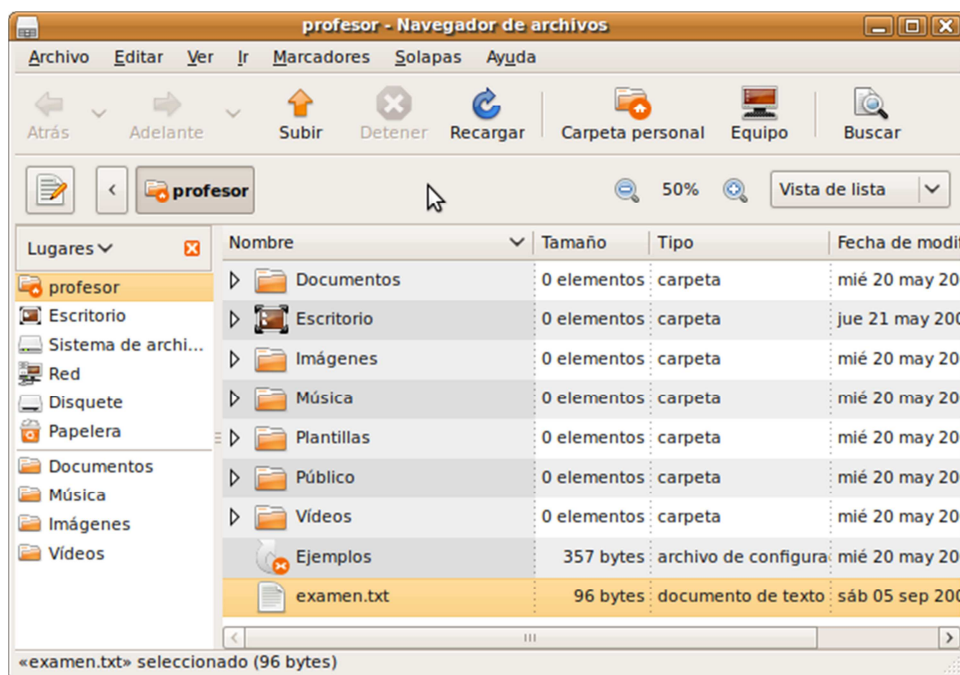
```

root@cnice-desktop: ~
Archivo Editar Ver Terminal Solapas Ayuda
pepe@cnice-desktop:~$ ls -l
total 4
-rw-r--r-- 1 pepe profes 11 2007-07-04 14:55 examen.txt
pepe@cnice-desktop:~$

```

### Mostrar el propietario con el comando ls -l

Comprobamos que el usuario propietario es **pepe** y el grupo propietario es **profesores**. La misma información podemos verla desde el administrador de archivos si vamos a la carpeta `/home/pepe` y mostramos las columnas correspondientes:



### 1.6.2 TIPOS DE PERMISOS

En los Sistemas Unix, la gestión de los permisos que los usuarios y los grupos de usuarios tienen sobre los archivos y las carpetas, se realiza mediante un sencillo esquema de tres tipos de permisos que son:

- Permiso de lectura
- Permiso de escritura
- Permiso de ejecución

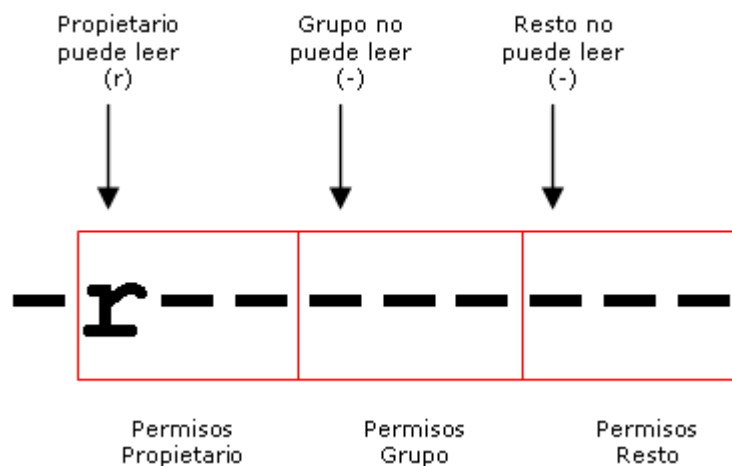
El significado de estos permisos difiere si se tienen sobre archivos o sobre carpetas. A continuación veremos el significado para cada uno de los casos:

#### ✓ Permiso de lectura

Cuando un usuario tiene permiso de lectura de un archivo significa que puede leerlo o visualizarlo, bien sea con una aplicación o mediante comandos. Ejemplo, si tenemos permiso de lectura sobre el archivo examen.txt, significa que podemos ver el contenido del archivo. Si el usuario no tiene permiso de lectura, no podrá ver el contenido del archivo.

Cuando un usuario tiene permiso de lectura de una carpeta, significa que puede visualizar el contenido de la carpeta, es decir, puede ver los archivos y carpetas que contiene, bien sea con el comando 'ls' o con un explorador de archivos como Konqueror. Si el usuario no tiene permiso de lectura sobre la carpeta, no podrá ver lo que contiene.

El permiso de lectura se simboliza con la letra 'r' del inglés 'read'.

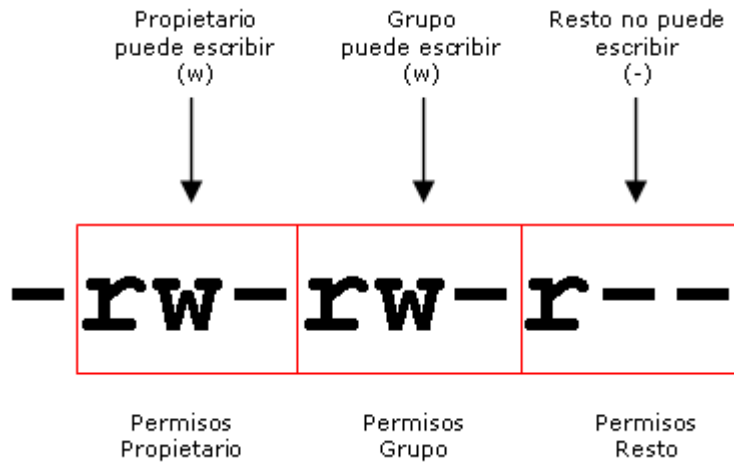


#### ✓ Permiso de escritura

Cuando un usuario tiene permiso de escritura sobre un archivo significa que puede modificar su contenido, e incluso borrarlo. También le da derecho a cambiar los permisos del archivo mediante el comando `chmod` así como cambiar su propietario y el grupo propietario mediante el comando `chown`. Si el usuario no tiene permiso de escritura, no podrá modificar el contenido del archivo.

Cuando un usuario tiene permiso de escritura sobre una carpeta, significa que puede modificar el contenido de la carpeta, es decir, puede crear y eliminar archivos y otras carpetas dentro de ella. Si el usuario no tiene permiso de escritura sobre la carpeta, no podrá crear ni eliminar archivos ni carpetas dentro de ella.

El permiso de escritura se simboliza con la letra 'w' del inglés 'write'.



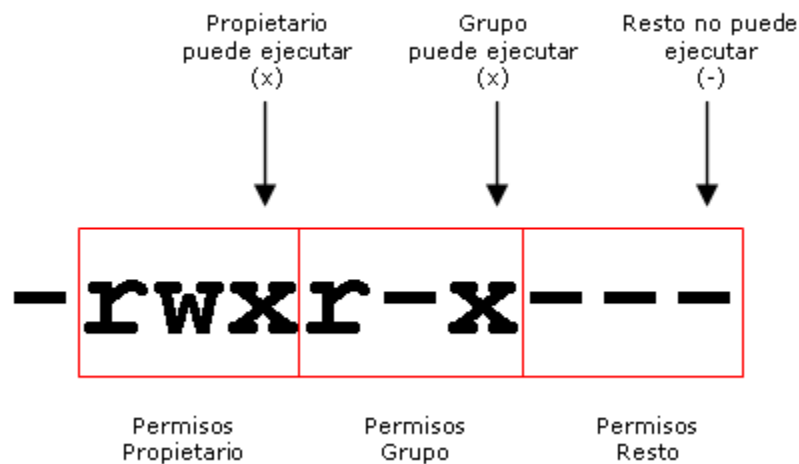
#### ✓ Permiso de ejecución

Cuando un usuario tiene permiso de ejecución de un archivo significa que puede ejecutarlo. Si el usuario no dispone de permiso de ejecución, no podrá ejecutarlo aunque sea una aplicación.

Los únicos archivos ejecutables son las aplicaciones y los archivos de comandos (scripts). Si tratamos de ejecutar un archivo no ejecutable, dará errores.

Cuando un usuario tiene permiso de ejecución sobre una carpeta, significa que puede entrar en ella, bien sea con el comando 'cd' o con un explorador de archivos como Konqueror. Si no dispone del permiso de ejecución significa que no puede ir a dicha carpeta.

El permiso de ejecución se simboliza con la letra 'x' del inglés 'eXecute'.





### 1.6.3 ¿A QUIÉN SE PUEDE OTORGAR PERMISOS?

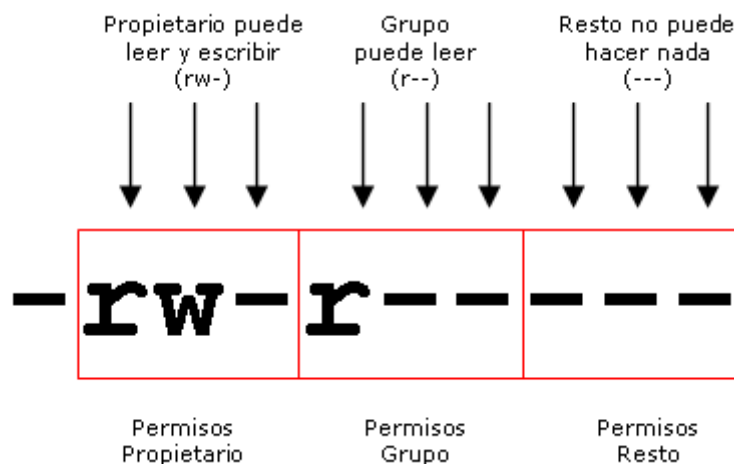
Los permisos solamente pueden ser otorgados a tres tipos o grupos de usuarios:

- Al usuario propietario del archivo
- Al grupo propietario del archivo
- Al resto de usuarios del sistema (todos menos el propietario)

Se pueden dar permisos de lectura, escritura, ejecución ó combinación de ambos al usuario propietario del archivo, al grupo propietario del archivo o al resto de usuarios del sistema. En Unix no existe la posibilidad de asignar permisos a usuarios concretos ni a grupos concretos, tan solo se puede asignar permisos al usuario propietario, al grupo propietario o al resto de usuarios.

Ejemplo, si disponemos de un archivo llamado 'examen.txt' cuyo propietario es 'pepe' y cuyo grupo propietario es 'profesores', se pueden dar permisos de lectura, escritura, ejecución ó combinación de ambos al usuario 'pepe', al grupo 'profesores' y al resto de usuarios, pero no podremos dar permisos a otros usuarios distintos de pepe (juan, luis, pedro,...) ni a otros grupos (alumnos, directivos, personal,...) ya que el esquema Unix no lo permite.

Supongamos que la siguiente figura representa los permisos de examen.txt:



El usuario propietario (pepe) podrá leer y escribir en el documento. Los pertenecientes al grupo profesores podrán leerlo y el resto no podrá hacer nada.

Si deseo que otros usuarios tengan algún permiso sobre el archivo 'examen.txt', no me quedará más remedio que incluirlos en el grupo profesores u otorgar el permiso al resto de usuarios pero si hago esto último, absolutamente todos los usuarios del sistema gozarán del permiso, por eso no se recomienda salvo que eso sea nuestra intención.

Para poder cambiar permisos sobre un archivo, es necesario poseer el permiso de escritura sobre el mismo. El usuario root puede modificar los permisos de cualquier archivo ya que tiene acceso total sin restricciones a la administración del sistema.

### 1.6.4 VISUALIZAR LOS PERMISOS DE UN ARCHIVO O CARPETA

Con el **comando ls -l** podemos visualizar los permisos de los archivos o carpetas. Al ejecutar el comando aparecen todos los archivos, uno por línea. El bloque de 10 caracteres del principio simboliza el tipo de archivo y los permisos.

```

root@cnice-desktop: ~
Archivo  Editar  Ver     Terminal  Solapas  Ayuda
pepe@cnice-desktop:~$ ls -l
total 4
-rw-r--r-- 1 pepe profes 11 2007-07-04 14:55 examen.txt
pepe@cnice-desktop:~$

```

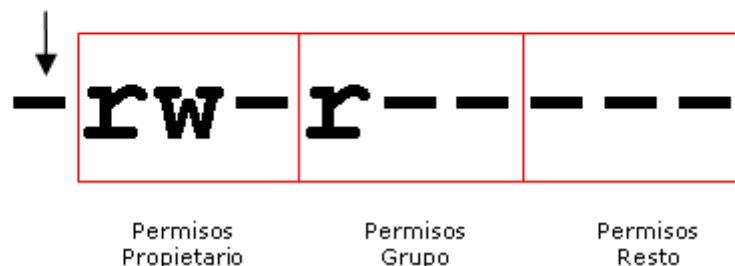
#### Permisos de archivo

El primer carácter indica de qué tipo de archivo se trata. Si es un guión '-' significa que se trata de un archivo normal, la letra 'd' significa que se trata de una carpeta (directory), la letra 'l' significa que se trata de un enlace (link). Otros valores son s, p, b que se refieren a sockets, tuberías (pipe) y dispositivos de bloque respectivamente.

Los 9 caracteres siguientes simbolizan los permisos del usuario propietario (3 caracteres), los permisos del grupo propietario (3 caracteres) y los permisos del resto de usuarios (3 caracteres). Vienen codificados con las letras r, w y x que se refieren a los permisos de lectura, escritura y ejecución. Si en lugar de aparecer dichas letras aparecen guiones significa que se carece de dicho permiso. Ejemplo, si los diez primeros caracteres son -rw-r----- significa que es un archivo normal, que el usuario propietario dispone de permisos de lectura y escritura pero no de ejecución, que el grupo propietario dispone tan solo de permiso de lectura y el resto de usuarios no dispone de ningún permiso. Veámoslo en la siguiente imagen:

#### Tipo de archivo:

- (-) para archivos normales
- (d) para carpetas (directory)
- (l) para enlaces (link)
- (s)=socket, (p)=tubería (pipe), (b)=dispositivo de bloque.



En el siguiente ejemplo vemos que pepe tiene permiso de lectura y escritura y que el resto solo tiene permiso de lectura tanto sobre el archivo 'apuntes.doc' como sobre el archivo 'examen.txt'.

```
// Visualización de permisos
ls -l
total 8
-rw-r--r-- 1 pepe profesores 359 2011-09-28 18:02 apuntes.doc
-rw-r--r-- 1 pepe profesores 11 2011-09-27 19:26 examen.txt
```

### 1.6.5 CAMBIO DE PERMISOS

Para cambiar los permisos de un archivo o una carpeta es necesario disponer del permiso de escritura (w) sobre dicho archivo o carpeta. Para hacerlo, se utiliza el comando `chmod`. La sintaxis del comando `chmod` es la siguiente:

```
chmod [opciones] permiso nombre_archivo_o_carpeta
```

Los permisos se pueden representar de dos formas.

**La primera** es mediante las iniciales de a quién va dirigido el permiso (usuario=u, grupo=g, resto=o (other)), seguido de un signo + si se quiere añadir permiso o un signo - si se quiere quitar y seguido del tipo de permiso (lectura=r, escritura=w y ejecución=x). Ejemplos:

```
// Dar permiso de escritura al usuario propietario sobre el archivo 'examen.txt'
# chmod u+w examen.txt
```

```
// Quitar permiso de escritura al resto de usuarios sobre el archivo 'examen.txt'
# chmod o-w examen.txt
```

```
// Dar permiso de ejecución al grupo propietario sobre el archivo '/usr/bin/games/tetris'
# chmod g+x /usr/bin/games/tetris
```

```
// Dar permiso de lectura al grupo propietario sobre el archivo 'examen.txt'
# chmod g+r examen.txt
```

```
// Se pueden poner varios permisos juntos separados por comas
# chmod u+w,g-r,o-r examen.txt
```

```
// Se pueden poner varios usuarios juntos
# chmod ug+w examen.txt
```

**La segunda** forma de representar los permisos es mediante un código numérico cuya transformación al binario representaría la activación o desactivación de los permisos. El código numérico está compuesto por tres cifras entre 0 y 7. La primera de ellas representaría los permisos del usuario propietario, la segunda los del grupo propietario y la tercera los del resto de usuarios.

En binario, las combinaciones representan el tipo de permisos. El bit más a la derecha (menos significativo) se refiere al permiso de ejecución (1=activar y 0=desactivar). El bit central se refiere al permiso de escritura y el bit más a la izquierda se refiere al permiso de lectura. La siguiente tabla muestra las 8 combinaciones posibles:

Cód Binario	Permisos efectivos
0	0 0 0 ---
1	0 0 1 --x
2	0 1 0 -w-
3	0 1 1 -wx
4	1 0 0 r--
5	1 0 1 r-x
6	1 1 0 rw-
7	1 1 1 rwx

Si deseamos otorgar sólo permiso de lectura, el código a utilizar es el 4. Si deseamos otorgar sólo permiso de lectura y ejecución, el código es el 5. Si deseamos otorgar sólo permiso de lectura y escritura, el código es el 6. Si deseamos otorgar todos los permisos, el código es el 7. Si deseamos quitar todos los permisos, el código es el 0. Ejemplos:

```
// Dar todos los permisos al usuario y ninguno ni al grupo ni al resto
chmod 700 examen.txt
```

```
// Dar al usuario y al grupo permisos de lectura y ejecución y ninguno al resto
chmod 550 examen.txt
```

```
// Dar todos los permisos al usuario y lectura y ejecución al grupo y al resto
chmod 755 /usr/bin/games/tetris
```

```
// Dar todos los permisos al usuario y de lectura al resto, sobre todos los archivos
chmod 744 *
```

```
// Cambiar permisos a todos los archivos incluyendo subcarpetas
chmod -R 744 *
```

Existe la posibilidad de cambiar los permisos utilizando el explorador de archivos. Para ello tan solo hay que seleccionar los archivos o carpetas y haciendo clic sobre la selección con el botón derecho del ratón > Propiedades, nos aparecerá la ventana de propiedades. Haciendo clic en la pestaña Permisos podremos establecer los permisos de una forma sencilla y haciendo clic en 'Permisos avanzados' de una forma avanzada.



**<Despedida>**

Email

**bienvenidosaez@gmail.com**

Instagram

**@bienvenidosaez**

Youtube

**youtube.com/bienvenidosaez**