

# ENLACES DE FICHEROS

**(RECORDAD QUE SIEMPRE PONEMOS RUTAS ABSOLUTAS SI ESTÁ EN DIRECTORIOS DISTINTOS EL ORIGEN DEL DESTINO)**

**(LOS ENLACES SIEMPRE CON CUALQUIER EDITOR MENOS CON EL VI O VIM)**

## DEFINICIÓN DE ENLACES

- Los enlaces le permiten dar a un único fichero múltiples nombres. Los ficheros son identificados por el sistema por su número de i-nodo, el cual es el único identificador del fichero para el sistema operativo.
- Es decir, los enlaces se utilizan con el fin de poder situar un mismo fichero en diferentes directorios o de poder asignarle diferentes nombres, sin que exista, en realidad, más que un único fichero.

Tenemos dos tipos de enlaces: Duros y Simbólicos

## RESUMEN:

Cuando creamos un **enlace simbólico** significa que **creamos un puntero** a dicho fichero o directorio. Si borrasen dicho fichero o directorio el enlace quedaría apuntando a la nada.

Por otra parte si creas un **enlace duro**, lo que hace es **crear una copia del fichero o directorio**

## ENLACES DUROS (HARD LINKS) COPIA DE FICHEROS

La orden `ln` es usada para crear múltiples enlaces para un fichero. Por ejemplo, supongamos que tenemos un fichero llamado **original** en un directorio. Usando `ls -li` veremos el número de inodo para el fichero:

creamos un fichero:

**touch original**

**ln -i original**

22192 original

Aquí, este fichero tiene el número de inodo 22192 en el sistema de ficheros. Podemos crear otro enlace a **original**, llamado **enlace\_duro**:

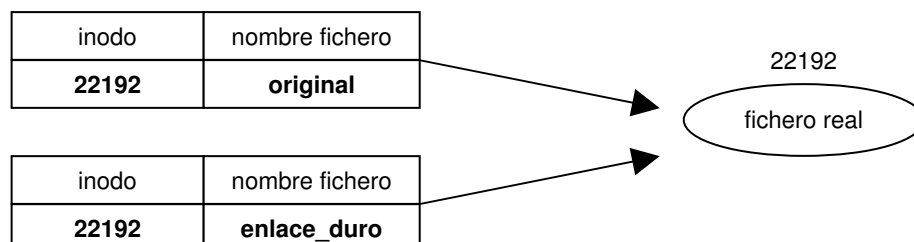
**ln original enlace\_duro**

**ls -li original enlace\_duro**

22192 original 22192 enlace\_duro

Ahora, accediendo a **original** o a **enlace\_duro** accederemos al mismo fichero. Si hacemos cambios en **original**, estos cambios también serán realizados en **enlace\_duro**, y lo mismo al revés. A todos los efectos, **original** y **enlace\_duro** son el mismo fichero.

Estos enlaces se conocen como enlaces duros porque directamente crean el enlace al inodo. Por eso sólo podemos crear enlaces duros entre ficheros del mismo sistema de ficheros; con los enlaces simbólicos no tenemos esta restricción.



Cuando borramos un fichero con **rm**, sólo borramos un enlace al fichero real. Por ejemplo, si borramos **original**:

#### **rm original**

solo estamos borrando el enlace de nombre **original**; el enlace **enlace\_duro** todavía existirá, apuntando al inodo correspondiente al fichero. Un fichero se borra definitivamente del sistema cuando ya no quedan enlaces a él.

Normalmente, un fichero sólo tiene un enlace, por lo que con **rm** lo borramos. Pero si el fichero tiene varios enlaces, con **rm** borramos un único enlace; para borrar el fichero tenemos que borrar todos los enlaces al fichero.

La orden **ls -l** muestra, entre otra información, el número de enlaces a un fichero:

```
ls -l
-rw-r--r--    2    root    root    fecha    original
-rw-r--r--    2    root    root    fecha    enlace_duro
```

La segunda columna nos indica que el fichero asociado a **original** tiene dos enlaces.

De esta manera, un directorio no es más que un fichero que contiene información relacionando enlaces con los inodos correspondientes. Además, cada directorio tiene al menos dos enlaces duros asociados:

- “.” que es un enlace apuntando a si mismo
- “..” que es un enlace apuntando al directorio padre

## **ENLACES SIMBÓLICOS (ACCESO DIRECTO)**

Los enlaces simbólicos son otro tipo de enlace, diferente al enlace duro. Un enlace simbólico permite dar a un fichero el nombre de otro, pero no enlaza el fichero con el inodo.

Se utiliza la orden **ln -s**:

#### **ln -s original enlace\_simb**

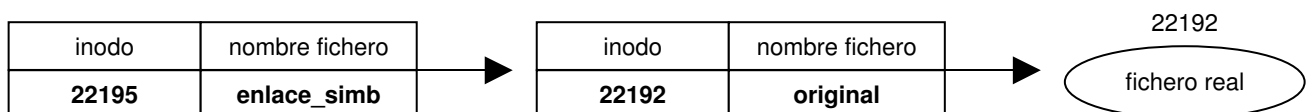
crea un enlace simbólico al fichero **original**. Si utilizamos la orden **ls -i** veremos que los dos ficheros tienen inodos diferentes:

```
ls -i original enlace_simb
22192 original                22195 enlace_simb
```

Además, usando la orden **ls -l** vemos que el fichero **enlace\_simb** es un enlace simbólico apuntando a **original**:

```
ls -l
lrwxrwxrwx    1    root    root    fecha    enlace_simb -> original
-rw-r--r--    2    root    root    fecha    original
```

Podemos observar que el primer bit de la línea nos indica que es un enlace simbólico con la letra **l** (link). Los permisos en un enlace simbólico no se utilizan, y quedan determinados por los permisos del fichero apuntado por el enlace (en este caso, **original**).



Un enlace simbólico se borra también usando el comando **rm**.

Funcionalmente, los enlaces duros y simbólicos son similares, pero hay algunas diferencias. Los enlaces simbólicos son procesados por el núcleo de forma diferente a los duros., y permiten identificar fácilmente el fichero al que apuntan. Con los enlaces duros no hay forma fácil de saber qué fichero está enlazado al mismo inodo.

Además, los enlaces duros sólo pueden crearse dentro de una misma partición, y sólo el usuario root tiene derecho a crear enlaces duros a directorios. Por todo ello, es más recomendable el uso de enlaces simbólicos. Sólo tenemos que tener cuidado que no queden por el sistema enlaces que no apunten a ningún archivo, por haber borrado éste.

HACER EJERCICIO CON ENLACE SIMBÓLICO CAMBIANDO LOS PERMISOS.