

CURSO FUNDAMENTOS DE LINUX



- INSTRUCTOR: RUDY SALVATIERRA
RODRIGUEZ

Linux
many flavours, many choices. Freedom

REDIRECCIONANDO EN LINUX

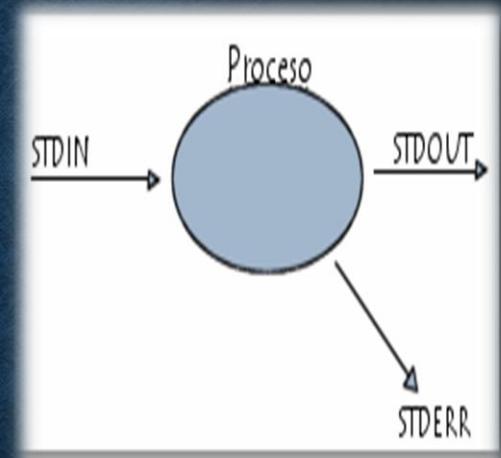
Un proceso en un sistema Linux tiene inicialmente abiertos 3 canales:

- 0: STDIN o entrada estándar
- 1: STDOUT o salida estándar
- 2: STDERR o salida de error

- **Imaginemos una refinería:**

Metes crudo por 0 (STDIN), consigues gasolina por 1 (STDOUT) y bastantes residuos por el “desagüe” 2 (STDERR).

- **Tubería:** También conocidas como pipes relacionan la salida estándar de un comando con la entrada estándar de otro .



REDIRECCIONANDO EN LINUX

Redirigiendo la salida de un comando:

- **>**: redirigir **STDOUT** a un fichero:
 - ls > listado.txt
- **>>**: redirigir **STDOUT** al final de un fichero (añadir):
 - ls >> listados.txt
- **2>**: redirigir **STDERR** a un fichero:
 - ls 2> errores.txt
- **2>>**: redirigir **STDERR** al final de un fichero:
 - ls 2>> errores.txt
- **2>&1**: redirigir **STDOUT** y **STDERR** a un fichero:
 - ls > salida 2>&1

TUBERIAS EN LINUX

Redirigiendo la salida de un comando:

- | **símbolo pipe**: redirigir el **STDOUT** de un comando al **STDIN** de otro : es posible recoger la salida de un desagüe y conducirlo a la entrada de otro comando.
- Ejemplos

history | more

cat fichero.txt | grep nuevo | tail -n 10

cat fichero.txt | sort -u | wc

cat fichero.txt | short > fichero.txt

ps -edf | grep sshd

COMPRESION DE ARCHIVOS EN LINUX

En Linux existen varios formatos de compresión como ser: bzip2, gzip, p7zip, tar, unzip, zip, unrar.

- **Compresión en gzip y bzip2:** para poder comprimir primero tenemos que empaquetar todos los archivos y directorios que queremos comprimir para ello se utiliza el comando **tar**.
- **tar:** permite empaquetar archivos y directorios
tar [opciones] [nombre_archivo.tar] [/origen]

Opciones

c: crear un archivo **x:** extraer de un archivo **t:** listar contenidos del archivo

v: ver un reporte de las acciones a medida que se van realizando.

f: empaquetar contenidos de archivos **z:** para comprimir a la vez que se empaqueta.

COMPRESION DE ARCHIVOS EN LINUX

Ejemplos:

empaquetar: `tar cvf archivo.tar /archivo/carpeta/*`

desempaquetar: `tar xvf archivo.tar`

- **gz (solo comprime archivos, no carpetas)**

Comprimir: `gzip -q archivo`

Descomprimir: `gzip -d archivo.gz`

- **.bz2 y bz**

Comprimir: `bzip2 -q archivo`

Descomprimir: `bzip2 -d archivo.gz`

COMPRESION DE ARCHIVOS EN LINUX

- .tar.gz

Comprimir: tar czvf archivo.tar.gz /archivo/carpeta/*

Descomprimir: tar xzvf archivo.tar.gz

- .tar.bz2

Comprimir: tar cjvf archivo.tar.gz /archivo/carpeta/*

Descomprimir: tar xjvf archivo.tar.bz2

- .zip

Comprimir: zip archivo.zip /carpeta/archivos

Descomprimir: unzip archivo.zip

ADMINISTRACION DE ARCHIVOS

- Al ser **Linux** un sistema multiusuario, para proteger ficheros de usuarios particulares de la manipulación por parte de otros, **Linux** proporciona un mecanismo conocido como permisos de ficheros.
- Este mecanismo permite que ficheros y directorios "pertenezcan" a un usuario y grupo en particular.
- Por ejemplo, como el usuario publica creó ficheros en su directorio "home", el usuario publica es el propietario de esos ficheros y tiene acceso a ellos.
- Cada fichero pertenece a un usuario en particular.
- Por otra parte, los ficheros también pertenecen a un grupo en particular, que es un conjunto de usuarios definido por el sistema.
- Cada usuario pertenece al menos a un grupo cuando es creado.

ADMINISTRACION DE ARCHIVOS

- Los permisos están divididos en tres tipos:
 - Lectura
 - Escritura
 - Ejecución
- Estos permisos pueden ser fijados para tres clases de usuarios:
 - El propietario del fichero
 - El grupo al que pertenece el fichero y
 - Para todos los usuarios independientemente del grupo

ADMINISTRACION DE ARCHIVOS

Que es el umask ?

umask (abreviatura de user mask, máscara de usuario) es una orden y una función en entornos POSIX que establece los permisos por defecto para los nuevos archivos y directorios creados por el proceso actual. ... Un permiso por defecto, también llamado máscara simbólica. Por ejemplo, u=rwx g=rwx o=

Las máscaras en octal se calculan a través del AND binario del complemento unario del argumento (utilizando el NOT binario) y los permisos completos.

ADMINISTRACION DE ARCHIVOS

Ejemplo de como calcular el umask

$$666_8 \text{ AND NOT}(174_8) = 602_8$$

$$666_8 = (110\ 110\ 110)_2$$

$$174_8 = (001\ 111\ 100)_2$$

$$\text{NOT}(001\ 111\ 100)_2 = (110\ 000\ 011)_2$$

$$(110\ 110\ 110)_2 \text{ AND } (110\ 000\ 011)_2 = (110\ 000\ 010)_2$$

$$666_8 \qquad \qquad \text{NOT } (174)_8 \qquad \qquad (602)_8$$

mientras que

$$777_8 \text{ AND NOT}(174_8) = 603_8$$

$$777_8 = (111\ 111\ 111)_2$$

$$174_8 = (001\ 111\ 100)_2$$

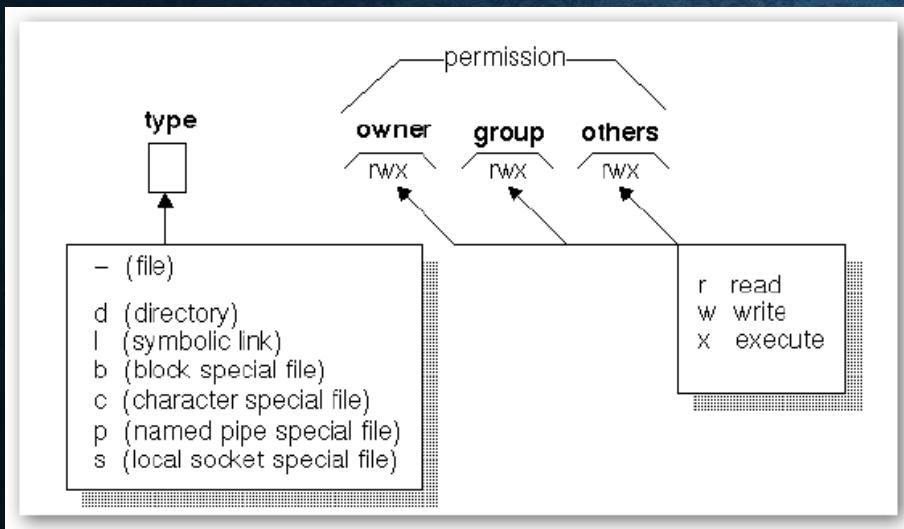
$$\text{NOT}(001\ 111\ 100)_2 = (110\ 000\ 011)_2$$

$$(111\ 111\ 111)_2 \text{ AND } (110\ 000\ 011)_2 = (110\ 000\ 011)_2$$

$$777_8 \qquad \qquad \text{NOT } (174)_8 \qquad \qquad (603)_8$$

ADMINISTRACION DE ARCHIVOS

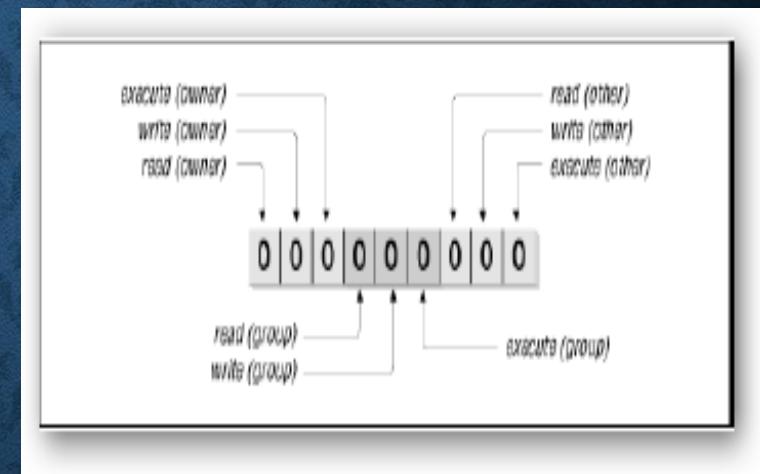
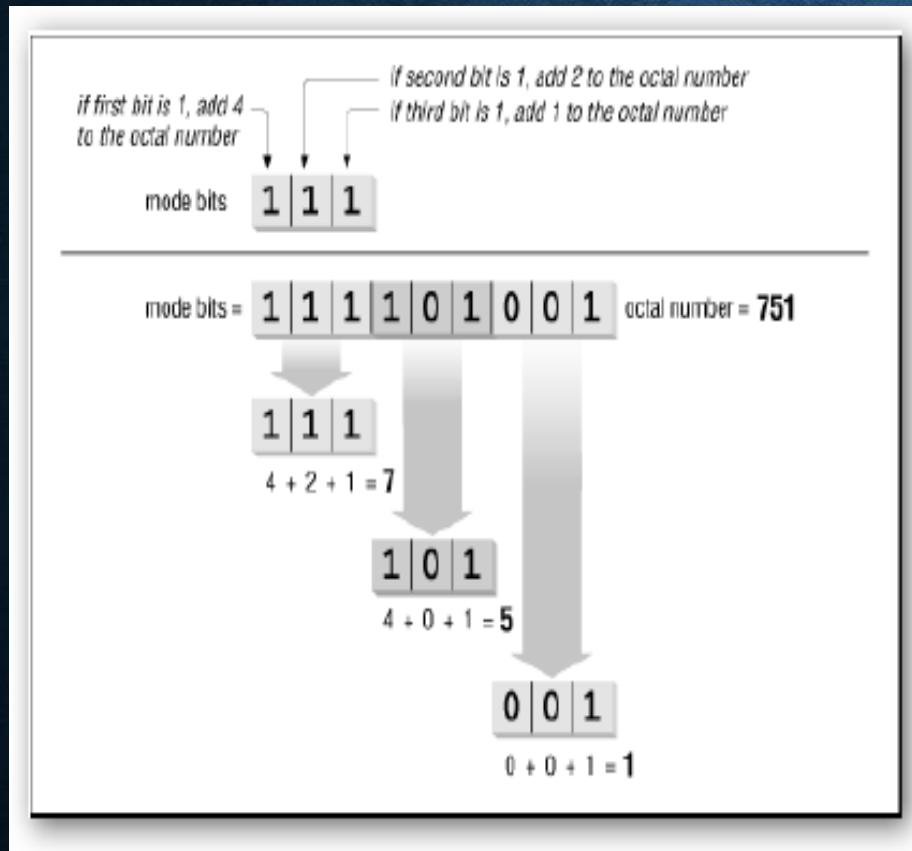
- Todos los archivos en Linux tienen permisos que indican qué y quién puede hacer o no hacer una acción con el archivo.
- Es la base de la seguridad de Linux.
- 2 formas de notación:



Valor	Descripción
-	Denota un fichero regular.
d	Denota un directorio.
b	Denota un fichero especial como dispositivo de bloque.
c	Denota un fichero de carácter especial
l	Denota un enlace simbólico.
p	Denota una tubería nombrada (FIFO)
s	Denota un zócalo de dominio (socket)

ADMINISTRACION DE ARCHIVOS

- Permiso de ficheros en Modo octal:



ADMINISTRACION DE ARCHIVOS

```
/home/publica# ls -l
```

```
-rw-r--r-- 1 publica publica 505 Mar 19 19:05 nuevo
```

La cadena **-rw-r--r--** nos informa, por orden, de los permisos para el **propietario**, el **grupo del fichero** y cualquier **otro usuario**.

R permiso de lectura. **W** permiso de escritura. **X** permiso de Ejecución.

- El primer carácter de la cadena de permisos.
- Las siguientes tres letras("rw-") representan los permisos para el propietario del fichero, publica.
- Las siguientes tres letras("r--") representan los permisos para el grupo del fichero, publica.
- Las siguientes tres letras("r--") representan los permisos para el cualquier otro usuario del sistema.

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

Comando chmod: Se usa para establecer los permisos de un fichero. Solo el propietario puede cambiar los permisos del fichero.

- La sintaxis de chmod es:

chmod {a,u,g,o}{+,-}{r,w,x} <filenames>

- Brevemente, indicamos a que usuarios afecta all, user, group o other.
- Se especifica si se están añadiendo permisos (+) o quitándolos (-).
- Finalmente se especifica que tipo de permiso read, write o execute.

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

Comando chmod: Se usa para establecer los permisos de un fichero. Solo el propietario puede cambiar los permisos del fichero.

Opciones

Opción	Descripción
-R	Cambia permisos de forma descendente en un directorio dado. Es la única opción de los estándares POSIX .
-c	Muestra que ficheros han cambiado recientemente en una ubicación dada
-f	No muestra errores de ficheros o directorios que no se hayan podido cambiar
-v	Descripción detallada de los mensajes generados por el proceso

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

- El siguiente comando tambien sirve para cambiar los permisos de un fichero

chmod XYZ <filenames>

Donde:

- X: DUEÑO
- Y: GRUPO
- Z: OTROS

Se debe tomar un numero binario representado por 3 variables (1,0) para cada componente X,Y o Z.

- **Ejemplos:**

chmod a+r nuevo Da a todos los usuarios permiso de lectura sobre el archivo nuevo.

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

Ejemplos Modo alfabético:

- Como arriba si no se indica a, u, g,o por defecto se toma a.
chmod +r nuevo
- Quita permisos de ejecución a todos los usuarios excepto al propietario.
chmod og-x nuevo
- Permite al propietario leer, escribir y ejecutar el fichero.
chmod u+rwx nuevo

Modo Octal:

- Cambiar los permisos del archivo file1.txt ls -l file1.txt
-rwxr-x--- 1 luis usuario 587 may 23 17:17 file1.txt

Permisos Actuales **r w x-r- x---** **4+2+1** **4+0+1** **0+0+0**

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

- **Ejemplo:**

- Cambiar los permisos del archivo file1.txt para que el usuario tenga todos los permisos, el grupo tenga permisos de lectura y ejecucion y otros solo lectura.

Permisos actuales -rwxr-x--- 1 luis usuario 587 may 23 17:17 file1.txt

chmod 751 nuevo 7 5 4
 r w x r-x r --

Permisos actuales -rwxr-xr-- 1 luis usuario file1.txt

Numero 4 para lectura r

Numero 2 para escritura w

Numero 1 para ejecución x

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

Valor	Permiso	Descripción
0	-	Nada
1	x	Ejecución
2	w	Escritura
3	wx	Escritura y ejecución
4	r	Lectura
5	rx	Lectura y Ejecución
6	rw	Lectura y Escritura
7	rwx	Lectura, Escritura y Ejecución

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

Permisos	Descripción
<code>drwxr-xr-x</code>	Directorio con permiso 755
<code>crw-rw-r--</code>	Fichero de carácter especial con permiso 664.
<code>srwxrwxr-x</code>	Zócalo con permiso 775.
<code>prw-rw-r--</code>	Tubería (FIFO) con permiso 664.
<code>-rw-r--r--</code>	Fichero regular con permiso 644.

ADMINISTRACION DE ARCHIVOS

(Cambiando permisos)

comando chown: Permite modificar a los usuarios o grupos dueños de un archivo o carpeta en el sistema de archivos.

- formas de utilizar el **comando chown** son las siguientes:

chown usuario archivo o carpeta.

chown -R usuario archivo o carpeta.

chown usuario *

Para poder cambiar el usuario y el grupo .

chown usuario: grupo archivo o carpeta

comando chgrp: Permite modificar al grupo de un archivo.

chgrp [nombre_grupo] [nombre_archivo]

chgrp grupo_nuevo nombre_archivo