CURSO LINUX ADMINISTRATOR

Comandos de Administración

- INSTRUCTOR: RUDY SALVATIERRA RODRIGUEZ

Administración de Usuarios

Características de usuarios Unix: Los sistemas Unix son sistemas multiusuario. Cada usuario tiene una serie de características propias y asociadas:

- **uid:** identificativo de usuario (debe ser único)
- **gid:** identificativo de grupo
- **home:** carpeta de trabajo o personal
- **shell:** interprete de comandos

Comandos:

- su/sudo (cambia de usuario o privilegios)

Administración de Usuarios

Gestión de usuarios

adduser (crear usuarios)

$ adduser alumno --ingroup nombre_grupo

$ adduser –-home /home/alumno –-shell /bin/sh –uid 5001 –cgroups urso alumno

usermod (modificar usuarios)

$ usermod –-shell /bin/bash alumno

$ usermod -G softwarelibre alumno esto añade a "alumno" al grupo "softwarelibre"

userdel (eliminar usuarios)

$ userdel softwarelibre

$ userdel -r softwarelibre

Administración de Usuarios

Gestión de usuarios

$ deluser nombre_usuario nombre_grupo permite eliminar un usuario de un grupo.

passwd (permite cambiar la contraseña de un usuario)

$ passwd nombre_usuario

Los archivos de configuracion de usuarios son el archivo passwd en el cual se encuentra toda la informacion del usuario, el archivo shadow y el archivo sudoers y los archivos se encuentran en:

/etc/passwd

/etc/shadow

/etc/sudoers

Administración de Usuarios

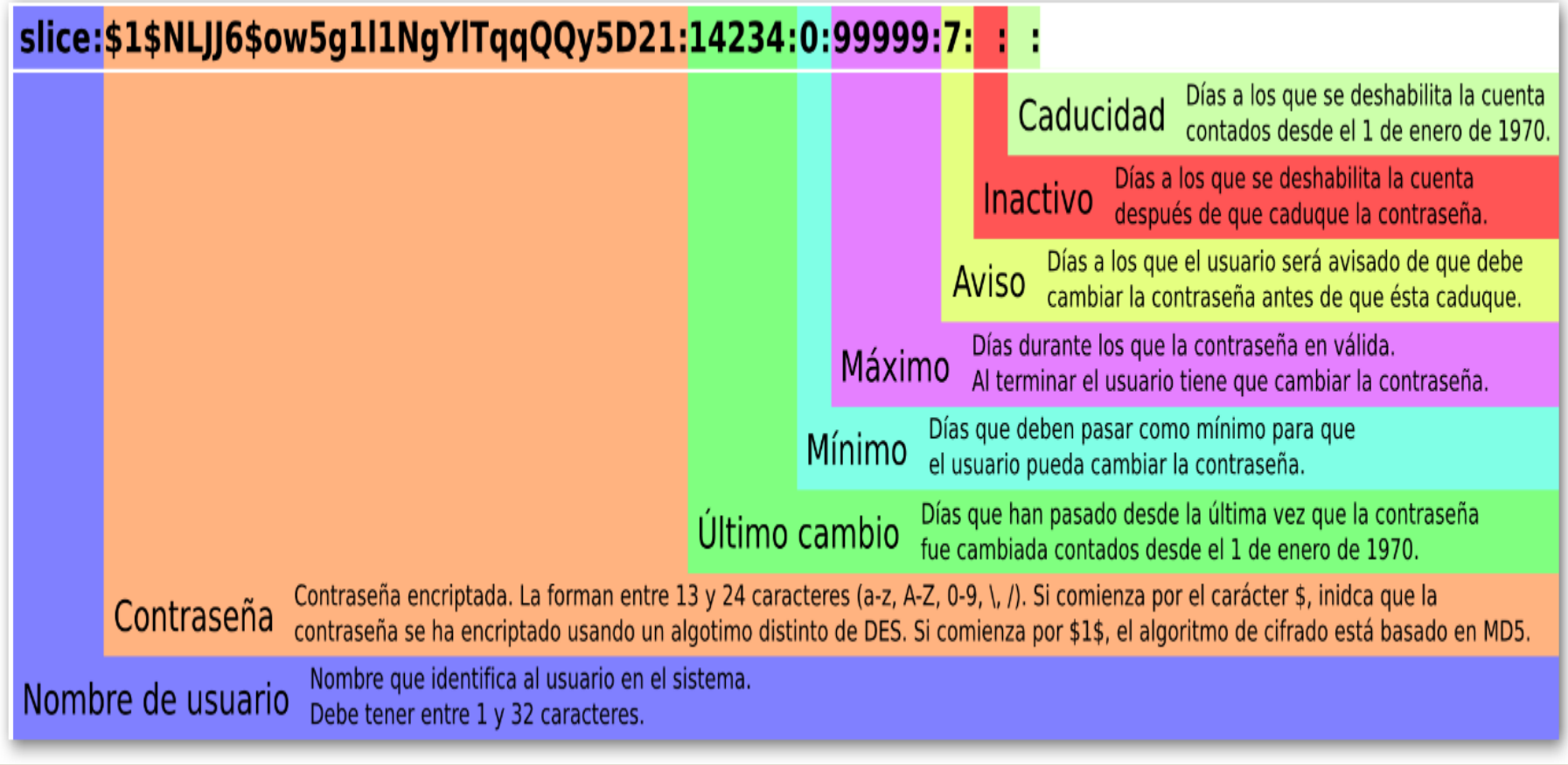
Archivo passwd: Lo primero que se debe hacer es editar el archivo passwd con cualquier editor, se vera que tiene 7 campos separados por :.

```
usuario1:FXWUuZ.vwXttg:500:501:usuario pepito:/home/usuario1:/bin/bash
```

usuario1:	Nombre de la cuenta (Login)
FXWUuZ.vwXttg:	Clave de acceso encriptada (password)
500:	UID de esta cuenta
501:	GID del grupo principal al que pertenece la cuenta
usuario pepito:	Nombre del usuario
/home/usuario1:	Directorio de trabajo de usuario1
/bin/bash:	Interprete de comando (shell) de usuario pepito

Administración de Usuarios

Archivo shadow: Lo primero que se debe hacer es editar el archivo passwd con cualquier editor.



Administración de Usuarios

Gestión de grupos

groupadd (añade grupo o usuario a grupo)

$ groupadd curso1

groupmod (modifica grupo)

$ groupmod -n curso curso1

delgroup (elimina grupo o usuario de grupo)

$delgroup curso

groups (muestra a que grupos pertenece un usuario)

$groups nombre_usuario nombre_usuario1

gpasswd (permite cambiar la contraseña de un grupo)

$ gpasswd nombre_usuario

Administración de Usuarios

Los archivos de configuracion de los grupos esta en el archivo group que se encuentra en:

/etc/group

Archivo group: Lo primero que se debe hacer es editar el archivo passwd con cualquier editor.

```
Nombre_grupo:clave:gid:lista_de_usuarios
```

Lista_de_usuarios: colección de usuarios separados por comas que tiene a este grupo como secundario.

Administración de Usuarios

Archivo sudoers: Este archivo tiene reglas que los usuarios tienen que seguir cuando se usa el **comando sudo**.

Lo primero que se debe hacer es editar el archivo passwd con cualquier editor.

root ALL=(ALL) ALL significa que root puede ejecutar desde todas las terminales, en calidad de TODOS los usuarios (cualquiera), y ejecutar todos los comandos (cualquiera).

priv ALL=(ALL) ALL significa que el usuario priv puede ejecutar desde todas las terminales, todos los comandos (cualquiera).

nuevo ALL=/sbin significa que el usuario nuevo puede ejecutar desde cualquier terminal, los comandos q existen en el directorio /sbin.

otro ALL=/sbin/ifconfig significa que el usuario otro puede ejecutar desde cualquier terminal, el comando ifconfig.

Variables de Entorno

Una variable de entorno es un nombre asociado a una cadena de caracteres.

Dependiendo de la variable, su utilidad puede ser distinta. Algunas son útiles para no tener que escribir muchas opciones al ejecutar un programa, otras las utiliza el propio shell (PATH, PS1..etc)

Variable	Descripción
DISPLAY	Donde aparecen la salidas de X-Windows.
HOME	Directorio personal.
HOSTNAME	Nombre de la máquina.
MAIL	Archivo de correo.
PATH	Lista de directorios donde buscar los programas.
PS1	Prompt.
SHELL	Intérprete de comandos por defecto.
TERM	Tipo de terminal.
USER	Nombre del usuario.

Variables de Entorno

Los archivos globales del sistema están en:

etc/profile

etc/profile.d/

etc/bashrc o etc/bash.bashrc

y los archivos del espacio de usuario o locales:

~/.bashrc

~/.bash_profile

Variables de Entorno Globales

La forma de definir una variable de entorno cambia con el interprete de comandos, se muestra tcsh y bash siendo los dos mas populares en el ámbito Linux:

bash: export VARIABLE=Valor

tcsh: setenv VARIABLE Valor

Por ejemplo:

bash: export DISPLAY=localhost:0.0

tcsh: setenv DISPLAY localhost:0.0

Para poder ver todas las variables de un usuario se utiliza el siguiente comando

env

Administración de Usuarios

Archivos de configuración de usuario: para una mejor administración Linux contiene archivos y script de ejecución como ser:

skel Contiene todos los archivos **.** (ejemplo: .bashrc, .kde, etc) u otros que serán colocados en el directorio de un usuario al crear el usuario.

.bashrc son las órdenes que se deben ejecutar siempre que entremos en el intérprete.

.bash_profile son las órdenes que se ejecutarán la primera vez que entremos en el intérprete osea cuando iniciemos sesión.

.bash_logout contiene comandos de los usuarios desea ejecutar al cerrar sesión.

Administración de Usuarios

Ejemplo de configuración de .bash_profile:

```
echo “Bienvenido a Linux Centos Señor Usuario”
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi
umask 022
PATH=$PATH:/sbin
JAVADIR=/usr/local/jdk/
COMANDOS=/sbin
export  JAVADIR COMANDOS PATH
```

Administración de Usuarios

Ejemplo de configuración de .bashrc:

```
if [ -f /etc/bashrc ]; then
  . /etc/bashrc
 fi
alias cp=‘cp –i’
alias mv=‘mv –i’
alias ls=‘ls  --color –lF’
Alias escr_grafico=‘startx’
```

Permisos de Ficheros

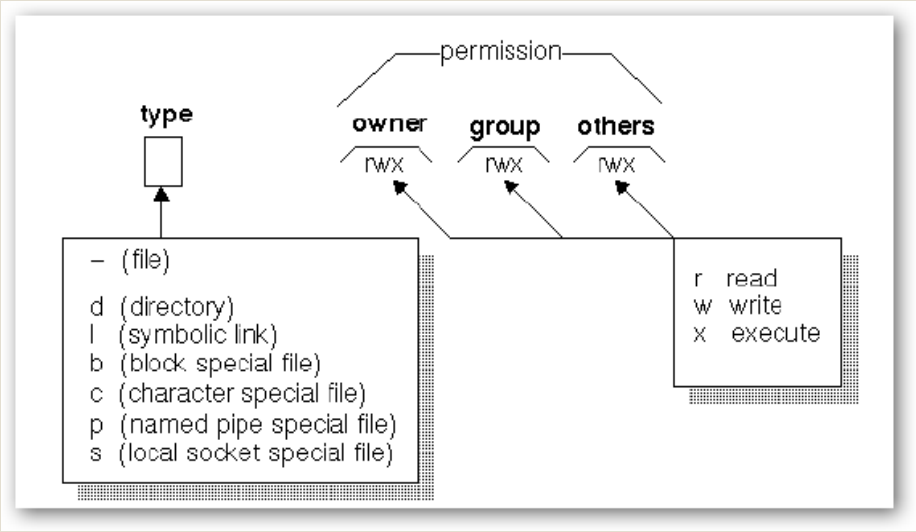
- Al ser **Linux** un sistema multiusuario, para proteger ficheros de usuarios particulares de la manipulación por parte de otros, **Linux** proporciona un mecanismo conocido como permisos de ficheros.
- Este mecanismo permite que ficheros y directorios "pertenezcan" a un usuario y grupo en particular.
- 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.
- El administrador del sistema puede hacer que un usuario tenga acceso a más de un grupo.

Permisos de Ficheros

- Los permisos están divididos en tres tipos con sus siguientes valores:
  - Lectura **r** **4**
  - Escritura **w** **2**
  - Ejecución **x** **1**
- 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

Permisos de Ficheros

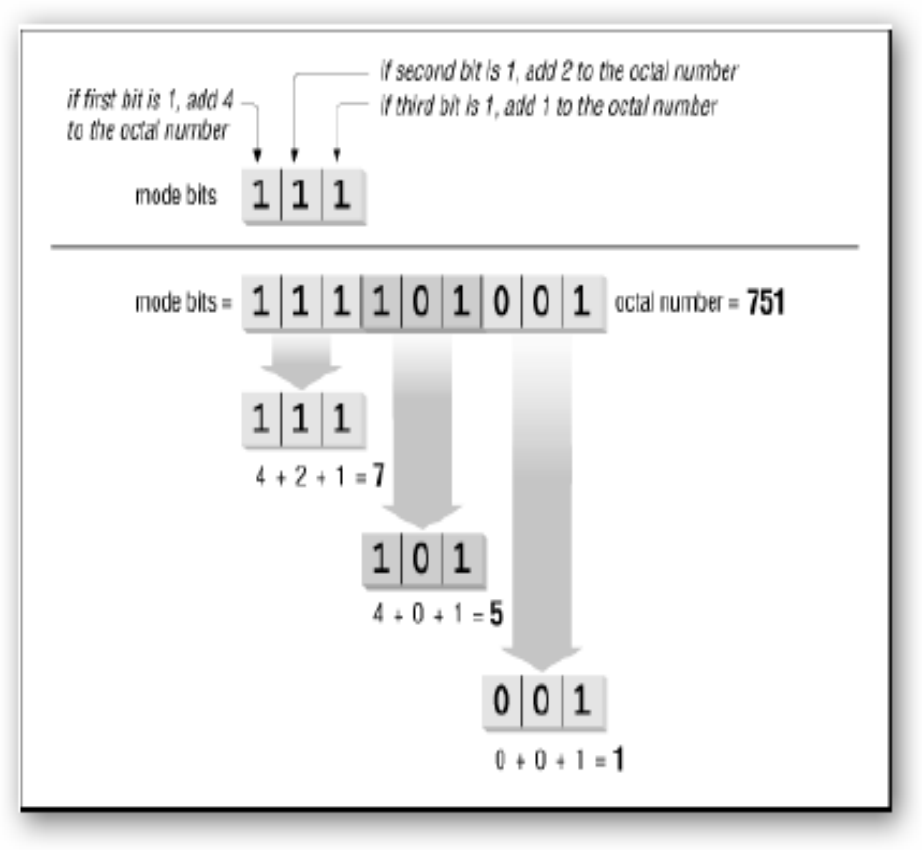
- Todos los archivos en Linux tienen permisos que indican que y quien puede hacer o no hacer una acción con el archivo.
- Es la base de la seguridad de Linux.
- 2 formas de notación:
  - **Modo alfabético**

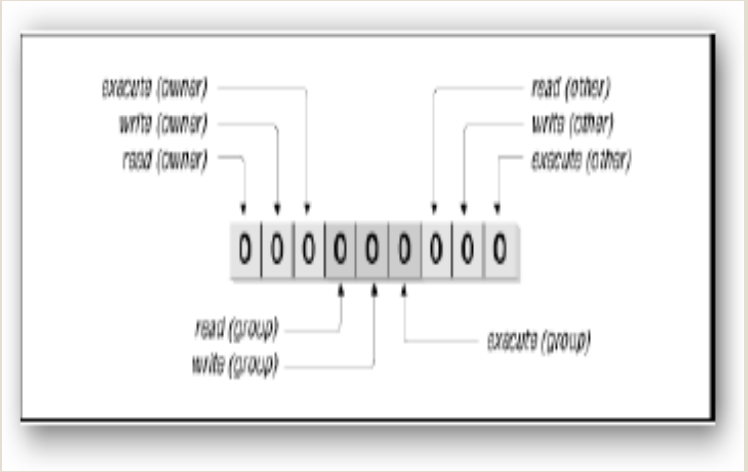


Valor	Descripción
-	Denota un fichero regular.
d	Denota un directorio.
b	Denota un fichero especial como dispositivo de bloque.
c	Denota un fichero de carácter especial
l	Denota un enlace simbólico.
p	Denota una tubería nombrada (FIFO)
s	Denota un zócalo de dominio (socket)

Permisos de Ficheros

- **Modo octal:**





Permisos de Ficheros

/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.

Permisos de Ficheros

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.

Permisos de Ficheros

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

Permisos de Ficheros

- 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.

Permisos de Ficheros

Ejemplos Modo alfabetico:

- 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

Permisos de Ficheros

- **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**

Numeo 1 para ejecucion **x**

Permisos de Ficheros

EJEMPLOS

Valor	Permiso	Decripción
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

Permisos de Ficheros

EJEMPLOS

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

Permisos de Ficheros

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

Permisos de Especiales

Permisos especiales: En los sistemas de archivos Linux se encuentran disponibles los siguientes permisos especiales o bit especiales

SUID (Set User ID) – Permite a los usuarios ejecutar un programa como si ellos fueran el usuario propietario del programa. En la mayoría de los casos el usuario propietario es el usuario root. El valor numérico para este permiso es 4

chmod **4**750 nombrearchivo.txt

Permisos de Especiales

SGID (Set Group ID) – Cuando es establecido en un directorio para cada nuevo archivo creado dentro de ese directorio, le asigna de forma automática el grupo propietario del directorio. El valor numérico es 2. En cambio, cuando es establecido en un archivo, SGID permite a los usuarios ejecutar un programa como si ellos fueran el grupo propietario del archivo.

chmod 2750 nombrearchivo.txt

chmod 2750 nombredir

Permisos de Especiales

Sticky bit – Este permiso es usado para evitar que los usuarios que no sean propietarios puedan borrar archivos en un directorio común o compartido. El valor numérico para este permiso es 1. En un directorio con el Sticky bit activo, solo el propietario del archivo o del directorio puede borrar el archivo. El usuario root siempre puede borrar los archivos

chmod **1**750 nombrearchivo.txt

Permisos de Especiales

EJEMPLOS:

Pueden establecerse varios bits de forma simultánea. El siguiente ejemplo establece los bits SUID (2) y SGID (4) para un determinado programa (lo cual 2 + 4 = 6):

chmod 6750 /some/archivo

La segunda forma de establecer permisos especiales es a través del modo simbólico. Por ejemplo, para agregar el SUID a un archivo:

chmod u+**s** /some/archivo

Para el caso de un directorio mediante el método simbólico:

chmod g+**s** /home/archivo

Permisos de Especiales

comando chattr: Este comando crea la máxima protección a archivos y/o carpetas en Linux mediante atributos o flags.

Si quisiéramos bloquear y proteger el archivo passwd, para establecer un atributo de solo lectura se coloca el siguiente comando.

chattr +i passwd

Para listar o ver los atributos que tenga un archivo podemos usar el comando lsattr, por ejemplo

lsattr passwd

para poder quitar este permiso se realiza el siguiente comando:

chattr -i passwd

Permisos de Especiales

comando chattr: Si quisiéramos en ocasiones que un determinado archivo pueda ser modificado, **PERO** sin alterar su contenido original.

Utilizamos el siguiente comando

chattr +a passwd

para poder quitar este permiso se realiza el siguiente comando:

chattr -a passwd

Para trabajar con carpetas es idéntico, la única diferencia es que si queremos cambiar atributos en los archivos de forma recurrente (los archivos que contiene la carpeta) utilizaremos el atributo –r.

chattr -r +i directorio1