



Universidad Nacional de La Matanza
Ingeniería en Informática-Taller de GNU/Linux 2003

TP N° 3
Permisos de archivo

Objetivos:

- Comprender el sistema de permisos para el acceso a archivos y directorios.
- Utilizar los comandos de cambio de permisos: chmod, chown, chgrp.
- Resolver los ejercicios propuestos.

Guía:

Se copia a continuación un resumen de la página de manual de comando chmod. La página completa puede obtenerse con "man chmod". Además se recomienda como lectura el capítulo 3 del documento "**Linux: Instalación y Primeros Pasos**" <http://es.tldp.org/Manuales-LuCAS/LIPP/>

NOMBRE

chmod - cambia los permisos de acceso de ficheros

SINOPSIS

chmod [opciones] modo fichero...

Opciones de POSIX: [-R]

Opciones de GNU (en la forma más corta): [-cfvR] [--reference=rfile] [--help]

[--version] [--]

DESCRIPCIÓN

chmod cambia los permisos de cada fichero dado según modo, que puede ser o bien una representación simbólica de los cambios a realizar, o bien un número octal que representa el patrón de bits de los nuevos permisos.

Modo Simbólico:

El formato de un argumento de cambio de modo simbólico es

[ugoa...][[+|=][rwxXstugo...][...][,...]

Tal argumento es una lista de órdenes de cambio de modo simbólico, separadas por comas. Cada orden de cambio de modo simbólico empieza con cero o más letras del conjunto 'ugoa': éstas controlan a qué usuarios se referirán los nuevos permisos del fichero cuyos permisos se van a cambiar:

- (u) el usuario propietario.
- (g) otros usuarios distintos del propietario pero del mismo grupo que el del fichero.
- (o) otros usuarios que ni son el propietario ni pertenecen al grupo del fichero.
- (a) todos los usuarios. De forma que 'a' equivale aquí a 'ugo'.

Operador de asignación:

- [+] El operador '+' hace que los permisos seleccionados se añadan a los ya existentes en cada fichero;
- [-] hace que se quiten de los que ya había,
- [=] hace que sea los únicos que el fichero va a tener.

Las letras 'rwxXstugo' seleccionan los nuevos permisos para los usuarios afectados:

- (r) lectura.
- (w) escritura o modificación.
- (x) ejecución (para un fichero ejecutable) o permiso de paso (para directorios) ,
- (X) ejecución solamente si el fichero es un directorio o ya tiene permiso de ejecución para algún usuario.
- (s) activar el bit SUID o SGID para ejecución.
- (t) activar el bit "pegajoso" o STI.
- (u) los permisos que el usuario propietario del fichero ya tiene.
- (g) los que otros usuarios del grupo del fichero, pero no el propietario, ya tienen.
- (o) los que otros usuarios distintos de los anteriores ya tienen.

(Así, 'chmod g-s fichero' quita el bit SGID (establecer el identificador de grupo), 'chmod ug+s fichero' activa los bits SUID y SGID, pero en cambio 'chmod o+s fichero' no hace nada.)

Modo Numérico:

Un modo numérico está compuesto de 1 a 4 dígitos octales (0-7), formados mediante adición de los bits de valores 4, 2 y 1. Los dígitos que falten se toman como ceros iniciales. El primer dígito selecciona los atributos SUID (4), SGID (2) y STI ['sticky'] (1). El segundo selecciona los permisos para el usuario propietario del fichero: lectura (4), escritura (2) y ejecución (1); el tercero, los permisos para otros usuarios del mismo grupo que el fichero, pero no el propietario, con los mismos valores que antes; y por último, el cuarto dígito selecciona los permisos para los otros usuarios que no son ni el propietario ni están en el mismo grupo que el fichero, con los mismos valores de nuevo.

chmod nunca cambia los permisos de enlaces simbólicos; la llamada al sistema chmod(2) no puede cambiar sus permisos. Esto no es un problema puesto que los permisos de los enlaces simbólicos nunca se usan. Sin embargo, para cada enlace simbólico puesto en la línea de órdenes, chmod cambia los permisos del fichero al cual apunta. En cambio, chmod hace caso omiso de los enlaces simbólicos que encuentre durante el recorrido recursivo de directorios.

OPCIONES DE POSIX

- R Cambia recursivamente los permisos de directorios y sus contenidos.

OPCIONES DE GNU

- c, --changes
Muestra un mensaje mencionando solamente aquellos ficheros cuyos permisos cambian realmente.
- f, --silent, --quiet
No muestra mensajes de error sobre ficheros cuyos permisos no se pueden cambiar.
- v, --verbose
Describe prolijamente la acción efectuada o no para cada fichero.

-R, --recursive

Cambia recursivamente los permisos de directorios y sus contenidos.

OPCIONES ESTÁNDARES DE GNU

--help Muestra un mensaje en la salida estándar sobre el modo de empleo y acaba con código de éxito.

--version

Muestra en la salida estándar información sobre la versión y luego acaba con código de éxito.

-- Termina la lista de opciones.

--reference=rfile

(Nueva en fileutils 4.0.) Cambia el modo de file al del fichero rfile.

Los comandos para cambiar propietario y grupo de un archivo son:

chown - cambia el usuario y grupo propietarios de ficheros

chgrp - cambia el grupo al que pertenecen los ficheros

Consultar las páginas de manual para mas información.

Ejercicios:

1)En un directorio vacío (nuevo), crear 9 archivos (archiv1,archiv2,etc.) utilizando el comando touch. Quitarle todos los permisos con el comando "chmod a-rwx archiv*"

2)Modificar los permisos usando el operador '=' del 'chmod', para que queden de la siguiente manera:

archiv1 -rwx-----	\$chmod u=rwx,go= archiv1
archiv2 -rw-----	\$
archiv3 -rwxrwxrwx	\$chmod a=rwx archiv3
archiv4 -rwxrw-r--	\$
archiv5 -rwxr-----	\$
archiv6 -r-xrw-r--	\$
archiv7 -r-----x	\$
archiv8 -rw-r--r--	\$
archiv9 -rw-rw-r--	\$

3)Modificar los permisos de los archivos anteriores utilizando los operadores + y - del 'chmod' para que queden de la siguiente manera (Los cambios son relativos a los archivos del ejercicio anterior):

archiv1 -rwx---r-- (agrega lectura para otros)	\$chmod o+r archiv1
archiv2 -r----- (quita escritura para propietario)	\$
archiv3 -rw-rw-rw- (quita ejecución para todos)	\$chmod a-x archiv3
archiv4 -rwx-w---- (quita lectura para grupo y otros)	\$
archiv5 -rwx----wx (quita lectura al grupo, agrega esc. y ejec para otros)	\$
archiv6 -rwxrw---- (agrega escritura al propietario, quita lectura a otros)	\$
archiv7 -rw---x-w-	\$
archiv8 -----r--	\$
archiv9 -rwx-----	\$

4) Crear 9 archivos (num1,num2,etc.) utilizando el comando touch.

5) Sobreescribir los permisos utilizando el comando chmod con argumento numérico (octal) para que queden de la siguiente manera:

num1 -r---w---x	\$chmod 0421 num1
num2 -----	
num3 -rwxrwxrwx	\$chmod 0777 num3
num4 -r-xrw-r--	
num5 -rwxr-----	
num6 -rw-r--r--	
num7 -rw-r--r-x	
num8 -rwxrw-r--	
num9 -rwx-----	

6) Con una sola instrucción, quitar permisos de lectura, escritura y ejecución para "otros" a todos los archivos utilizados en el último ejercicio.

\$

7) Crear un directorio y quitarle todos los permisos de ejecución. Explicar qué pasa al intentar entrar al directorio con el comando cd. Explicar el significado de los permisos r,w y x para directorios.

8) Informarse sobre los grupos a los que pertenece su usuario.

9) Utilizando los comandos chown y chgrp, intentar cambiar el propietario y el grupo del archivo "num3". Cual es el problema?