

104.5 Gestionar permisos y propiedad de archivos - Ejercicios

Nota: Realiza estos ejercicios en tu directorio personal (~) o en un subdirectorio que crees para la práctica. Necesitarás privilegios de superusuario (sudo) para cambiar la propiedad a otro usuario o establecer bits SUID/SGID.

Ejercicio 4.5.1: Viendo Permisos y Propiedad

- **Objetivo:** Usar `ls -l` para ver los permisos, propietario y grupo de archivos y directorios.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal y asegúrate de estar en tu directorio personal (`cd ~`).
 2. **Crea algunos archivos y directorios de prueba:**
 - `touch archivo1.txt archivo2.txt`
 - `mkdir directorio1 directorio2`
 - `echo "Script de prueba" > script.sh`
 - `chmod +x script.sh`
 3. **Lista el contenido en formato largo:** Ejecuta `ls -l`.
 4. **Analiza la primera columna:**
 - El primer carácter indica el tipo (`-` o `d`).
 - Los siguientes 9 caracteres son los permisos (`rwX`, `r-w`, etc.).
 - Identifica al propietario (tu usuario) y el grupo propietario (tu grupo principal).
 5. **Lista el contenido del directorio padre:** Ejecuta `ls -l ..`. Analiza los permisos de tu directorio personal.
 6. **Lista el contenido de un directorio del sistema:** Ejecuta `ls -l /tmp`. Observa los permisos (probablemente `rwXrwxrwt` - el sticky bit). Identifica al propietario (`root`) y al grupo (`root`). Nota que cualquier usuario puede escribir aquí, pero solo el propietario puede borrar sus propios archivos (debido al sticky bit).
 7. **Lista un archivo ejecutable con SUID (requiere permisos de root para ver algunos):** Ejecuta `ls -l /usr/bin/passwd`. Deberías ver algo como `-rwsr-xr-x`. La `S` en la posición del propietario indica el bit SUID.

Ejercicio 4.5.2: Cambiando Permisos con `chmod` (Simbólico y Numérico)

- **Objetivo:** Modificar permisos usando ambas notaciones.
- **Requisitos:** Usa los archivos y directorios de prueba creados en el Ej. 4.5.1.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal y asegúrate de estar en tu directorio personal (`cd ~`).
 2. **Verifica los permisos iniciales:** Ejecuta `ls -l archivo1.txt`. Probablemente sea algo como `-rw-rw-r--` o `-rw-r--r--`.
 3. **Quita el permiso de escritura para el grupo y otros:** Ejecuta `chmod go-w archivo1.txt`.

4. **Verifica los permisos:** Ejecuta `ls -l archivo1.txt`. Ahora debería ser `-rw-r--r--`.
5. **Añade el permiso de ejecución para el propietario y el grupo:** Ejecuta `chmod ug+x archivo1.txt`.
6. **Verifica:** Ejecuta `ls -l archivo1.txt`. Ahora debería ser `-rwxr-xr--`.
7. **Establece permisos específicos usando notación numérica:** Ejecuta `chmod 640 archivo1.txt`.
8. **Verifica:** Ejecuta `ls -l archivo1.txt`. Ahora debería ser `-rw-r-----`. (Usuario `rw`, Grupo `r`, Otros nada).
9. **Cambia permisos de un directorio:** Ejecuta `ls -l directorio1`. Probablemente sea `drwxrwxr-x` o `drwxr-xr-x`.
10. **Establece permisos 700 para el directorio:** Ejecuta `chmod 700 directorio1`.
11. **Verifica:** Ejecuta `ls -l directorio1`. Debería ser `drwx-----`. Solo el propietario puede acceder a él.
12. **Restablece permisos razonables para el directorio:** Ejecuta `chmod 755 directorio1`.
13. **Cambia permisos recursivamente (ej: solo lectura para otros):** Ejecuta `chmod -R o-w directorio1/`.
14. **Verifica:** Ejecuta `ls -l directorio1` y `ls -l directorio1/*` (si tiene contenido). Los directorios y archivos dentro de `directorio1` ahora tienen quitado el permiso de escritura para "otros".
15. **Restablece permisos recursivamente (ej: 755 para directorios, 644 para archivos - ¡esto es más complejo con chmod estándar, a menudo se hace en dos pasos o con find! Pero para este ejercicio, usa un modo general):** Ejecuta `chmod -R u+rw,go+r,go-w+x directorio1/`. (Esto es un ejemplo de notación simbólica más avanzada, o simplemente usa notación numérica si es apropiado). Un enfoque más simple para muchos casos: `chmod -R a+rX,ug+w,o-w directorio1/`. La 'X' añade ejecución solo si ya la tiene o si es un directorio.

Ejercicio 4.5.3: Cambiando Propiedad y Grupo

- **Objetivo:** Cambiar el propietario y/o grupo de archivos y directorios.
- **Requisitos:** Archivos y directorios de prueba. Privilegios de superusuario (`sudo`). Un usuario de prueba (`testuser`) y un grupo de prueba (`testgroup`, puedes crearlo con `sudo groupadd testgroup`) si no tienes otros.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal y asegúrate de estar en tu directorio personal (`cd ~`).
 2. **Verifica la propiedad actual:** Ejecuta `ls -l archivo1.txt`.

3. **Cambia el grupo propietario:** Ejecuta `sudo chgrp users archivo1.txt` (reemplaza `users` por un grupo existente en tu sistema, ej: `users`, `staff`, `testgroup`).
4. **Verifica el cambio de grupo:** Ejecuta `ls -l archivo1.txt`. El grupo propietario debería haber cambiado.
5. **Cambia el propietario (requiere sudo):** Ejecuta `sudo chown testuser archivo1.txt` (reemplaza `testuser` si usas otro usuario de prueba).
6. **Verifica el cambio de propietario:** Ejecuta `ls -l archivo1.txt`. Ahora el propietario es `testuser`.
7. **Cambia propietario y grupo a la vez (requiere sudo):** Ejecuta `sudo chown tu_usuario:tu_grupo_principal archivo1.txt` (reemplaza `tu_usuario` y `tu_grupo_principal`).
8. **Verifica:** Ejecuta `ls -l archivo1.txt`. La propiedad debería haber regresado a ti.
9. **Cambia propietario y grupo recursivamente en un directorio:** Ejecuta `sudo chown -R testuser:testgroup directorio1`.
10. **Verifica el cambio recursivo:** Ejecuta `ls -l directorio1` y `ls -l directorio1/*`.

Ejercicio 4.5.4: Entendiendo la Máscara de Creación (umask)

- **Objetivo:** Ver el valor de `umask` y entender cómo afecta los permisos por defecto.
- **Desarrollo Paso a Paso:**
 1. Abre una terminal.
 2. **Muestra el valor de umask en octal:** Ejecuta `umask`. Probablemente verás `0002` o `0022`. (El primer 0 es para permisos especiales).
 3. **Muestra el valor de umask en formato simbólico:** Ejecuta `umask -S`. Verás algo como `u=rwx, g=rwx, o=rx` (para `umask 002`) o `u=rwx, g=rx, o=rx` (para `umask 022`). Esto muestra los permisos que *no* se quitarán.
 4. **Crea un nuevo archivo y directorio:** Ejecuta `touch nuevo_archivo_umask.txt` y `mkdir nuevo_directorio_umask`.
 5. **Lista sus permisos:** Ejecuta `ls -l nuevo_archivo_umask.txt` y `ls -l nuevo_directorio_umask`.
 6. **Calcula los permisos esperados:**
 - Permisos máximos para archivo: `666 (-rw-rw-rw-)`. Resta tu `umask` (ej: `002`) octalmente: $666 - 002 = 664$. Deberías ver `-rw-rw-r--`.
 - Permisos máximos para directorio: `777 (drwxrwxrwx)`. Resta tu `umask` (ej: `002`) octalmente: $777 - 002 = 775$. Deberías ver `drwxrwxr-x`.
 - Si tu `umask` es `022`, espera `644 (-rw-r--r--)` para archivos y `755 (drwxr-xr-x)` para directorios.

7. **Establece un nuevo valor de umask para la shell actual (solo temporalmente):**
Ejecuta `umask 007`. Esto quitará todos los permisos a "otros".
8. **Verifica el nuevo umask:** Ejecuta `umask` y `umask -S`.
9. **Crea otro archivo y directorio:** Ejecuta `touch otro_archivo_umask.txt` y `mkdir otro_directorio_umask`.
10. **Lista sus permisos:** Ejecuta `ls -l otro_archivo_umask.txt`
`otro_directorio_umask`. Deberías ver permisos como `660 (-rw-rw----`) y `770 (drwxrwx---`).
11. **Sal de la terminal o abre una nueva para volver al umask por defecto.**
12. **Limpia:** Ejecuta `rm -r archivo1.txt archivo2.txt directorio1`
`directorio2 script.sh nuevo_archivo_umask.txt`
`nuevo_directorio_umask otro_archivo_umask.txt`
`otro_directorio_umask`.