

## PERMISOS DE ARCHIVO EN LINUX

### Descripción del proyecto

El equipo de investigación de la organización identificó la necesidad de actualizar los permisos de archivo para ciertos archivos y directorios dentro del directorio "*projects*". En su estado actual, los permisos no reflejan el nivel de autorización, lo cual es crucial para mantener la seguridad del sistema. Para abordar esta cuestión, se realizaron las siguientes acciones:

### Comprobar detalles del archivo y del directorio

```
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:27 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:27 ..
-rw--w---- 1 researcher2 research_team  46 Jul 31 15:27 .project_x.txt
drwx--x--- 2 researcher2 research_team  46 Jul 31 15:27 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Jul 31 15:27 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jul 31 15:27 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:27 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:27 project_t.txt
researcher2@5d738f0f927b:~/projects$
```

---

Se utilizó un conjunto de comandos en Linux para determinar los permisos asignados a un directorio específico en el sistema de archivos. En la primera línea de una captura de pantalla se muestra el comando introducido, mientras que las líneas subsiguientes reflejaron la salida generada. El comando *ls* con la opción *-la* fue empleado para listar detalladamente todos los contenidos del directorio "*projects*", incluyendo archivos ocultos. Los resultados indicaron la existencia de un directorio denominado "drafts", un archivo oculto llamado ".*project\_x.txt*", y otros cinco archivos de proyecto. La primera columna de la salida mostraba una cadena de 10 caracteres que representa los permisos asignados a cada archivo o directorio.

## Descripción de la cadena de permisos

La cadena de 10 caracteres permite identificar quién tiene acceso a los archivos y cuáles son los permisos específicos otorgados. Esta cadena se descompone de la siguiente manera:

1. El primer carácter indica el tipo de archivo: una "d" para directorios y un guion (-) para archivos normales.
2. Los caracteres del 2º al 4º representan los permisos de lectura (r), escritura (w), y ejecución (x) para el usuario. Un guion (-) indica la ausencia de dicho permiso.
3. Los caracteres del 5º al 7º detallan los permisos de lectura, escritura y ejecución para el grupo.
4. Los caracteres del 8º al 10º reflejan los permisos para otros usuarios del sistema.

Por ejemplo, para el archivo "project\_t.txt" con permisos -rw-rw-r--, se determina que es un archivo normal (primer carácter es un guion), y que tanto el usuario, el grupo, como otros usuarios tienen permisos de lectura. Los permisos de escritura son exclusivos para el usuario y el grupo, mientras que no se otorgan permisos de ejecución.

## Cambios en los permisos de archivo

La organización Julidió que otros usuarios no deberían tener permisos de escritura en ninguno de los archivos. Para implementar esta política, se eliminó el permiso de escritura para otros usuarios en el archivo "project\_k.txt" utilizando el comando `chmod`. Posteriormente, se verificaron las modificaciones con el comando `ls -la`.

```
researcher2@5d738f0f927b:~/projects$ chmod o-w project_k.txt
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:30 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:30 ..
-rw--w---- 1 researcher2 research_team  46 Jul 31 15:30 .project_x.txt
drwxr-x--- 2 researcher2 research_team 4096 Jul 31 15:30 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:30 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jul 31 15:30 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:30 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:30 project_t.txt
researcher2@5d738f0f927b:~/projects$
```

---

Las dos primeras líneas de la captura de pantalla muestran los comandos que ingresé, mientras que las demás líneas muestran la salida del segundo comando. El comando `chmod` cambia los permisos en archivos y directorios. El primer argumento indica qué permisos se deben cambiar y el segundo argumento especifica el archivo o directorio. En este ejemplo, eliminé los permisos de escritura de otros para el archivo `project_k.txt`. Luego, utilicé `ls-la` para revisar las actualizaciones que había hecho.

### **Cambiar permisos de archivo en un archivo oculto**

El archivo oculto `".project_x.txt"` fue recientemente archivado por el equipo de investigación, y se Julidió que nadie debería tener permisos de escritura, aunque el usuario y el grupo debían mantener permisos de lectura. Se aplicaron comandos específicos para ajustar estos permisos, incluyendo la eliminación de permisos de escritura para el usuario y el grupo, y la adición de permisos de lectura para el grupo.

El siguiente código muestra cómo utilicé los comandos de Linux para cambiar los permisos:

```
researcher2@3213bbc1d047:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@3213bbc1d047:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:32 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:32 ..
-r--r----- 1 researcher2 research_team  46 Jul 31 15:32 .project_x.txt
drwxr-x--- 2 researcher2 research_team 4096 Jul 31 15:32 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jul 31 15:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_t.txt
researcher2@3213bbc1d047:~/projects$
```

---

Las dos primeras líneas de la captura de pantalla muestran los comandos que ingresé, y las otras líneas muestran la salida del segundo comando. Sé que `.project_x.txt` es un archivo oculto porque comienza con un punto (.). En este ejemplo, eliminé los permisos de escritura del usuario y el grupo, y agregué permisos de lectura para el grupo. Eliminé los permisos de escritura del usuario con `u-w`. Luego, eliminé los permisos de escritura del grupo con `g-w` y agregué permisos de lectura para el grupo con `g+r`.

## Cambiar permisos de directorio

Para el directorio "*drafts*", se estableció que solo el usuario "*researcher2*" debería tener acceso, lo que implica que nadie más debería tener permisos de ejecución. Esto se logró eliminando los permisos de ejecución para el grupo mientras se mantenían para "*researcher2*".

```
researcher2@5d738f0f927b:~/projects$ chmod g-x drafts
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:32 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 15:32 ..
-r--r----- 1 researcher2 research_team  46 Jul 31 15:32 .project_x.txt
drwxr-x--- 2 researcher2 research_team 4096 Jul 31 15:32 drafts
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jul 31 15:32 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jul 31 15:32 project_t.txt
researcher2@5d738f0f927b:~/projects$
```

---

Las dos primeras líneas de la captura de pantalla muestran los comandos que ingresé. Las otras líneas muestran la salida del segundo comando. Antes había determinado que el grupo tenía permisos de ejecución, así que utilicé el comando *chmod* para eliminar estos permisos. El usuario *researcher2* ya tenía permisos de ejecución, por lo que no era necesario agregarlos.

## Resumen

Se ajustaron varios permisos para alinearlos con el nivel de autorización requerido por la organización para los archivos y directorios dentro del directorio "*projects*". El proceso incluyó la verificación inicial de permisos usando *ls -la*, seguido de la implementación de cambios con el comando *chmod*.