

Permisos de archivos en Linux

Descripción del proyecto

El equipo de investigación de mi organización necesita actualizar los permisos de ciertos archivos y directorios dentro del directorio *projects*. Los permisos actuales no reflejan el nivel de autorización que debería aplicarse. Revisar y actualizar estos permisos ayudará a mantener el sistema seguro. Para completar esta tarea, realicé lo siguiente:

Verificar los detalles de archivos y directorios

El siguiente código demuestra cómo utilicé comandos de Linux para determinar los permisos existentes establecidos para un directorio específico en el sistema de archivos.

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

La primera línea de la captura de pantalla muestra el comando que ingresé, y las demás líneas muestran la salida. El código lista todo el contenido del directorio *projects*. Utilicé el comando **ls** con la opción **-la** para mostrar un listado detallado del contenido de los archivos, lo que también devolvió los archivos ocultos. La salida de mi comando indica que hay un directorio llamado *drafts*, un archivo oculto llamado *.project_x.txt* y otros cinco archivos de proyecto. La cadena de 10 caracteres en la primera columna representa los permisos establecidos para cada archivo o directorio.

Describir la cadena de permisos

La cadena de 10 caracteres puede desglosarse para determinar quién está autorizado a acceder al archivo y qué permisos específicos tiene. Los caracteres y lo que representan son los siguientes:

- **1.er carácter:** Este carácter es una **d** o un guion (-) e indica el tipo de archivo.
 - Si es **d**, es un directorio.
 - Si es -, es un archivo regular.

2.º-4.º caracteres: Indican los permisos de lectura (r), escritura (w) y ejecución (x) para el **usuario**.

- Si alguno aparece como guion (-), significa que ese permiso **no** está concedido al usuario.

5.º-7.º caracteres: Indican los permisos de lectura (r), escritura (w) y ejecución (x) para el **grupo**.

- Si alguno aparece como guion (-), significa que ese permiso **no** está concedido al grupo.

8.º-10.º caracteres: Indican los permisos de lectura (r), escritura (w) y ejecución (x) para **otros**.

Este tipo de propietario incluye a todos los demás usuarios del sistema que no son el usuario ni el grupo.

- Si alguno aparece como guion (-), significa que ese permiso **no** está concedido para otros.

Por ejemplo:

Los permisos del archivo *project_t.txt* son:

Código

`-rw-rw-r--`

- El primer carácter es un guion (-), lo que indica que *project_t.txt* es un archivo, no un directorio.
- El segundo, quinto y octavo caracteres son **r**, lo que indica que el usuario, el grupo y otros tienen permisos de lectura.

- El tercer y sexto caracteres son **w**, lo que indica que solo el usuario y el grupo tienen permisos de escritura.
- Ningún carácter es **x**, por lo que nadie tiene permisos de ejecución sobre *project_t.txt*.

Cambiar permisos de un archivo

La organización determinó que **otros (other)** no deben tener acceso de escritura a ninguno de sus archivos. Para cumplir con esto, revisé los permisos que había obtenido previamente. Determiné que *project_k.txt* debía tener el permiso de escritura eliminado para **otros**.

El siguiente código demuestra cómo utilicé comandos de Linux para hacerlo:

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

Las dos primeras líneas de la captura de pantalla muestran los comandos que ingresé, y las demás líneas muestran la salida del segundo comando. El comando **chmod** cambia los permisos de archivos y directorios. El primer argumento indica qué permisos deben modificarse, y el segundo argumento especifica el archivo o directorio. En este ejemplo, eliminé los permisos de escritura para *other* en el archivo **project_k.txt**. Después de esto, utilicé **ls -la** para revisar las actualizaciones que realicé.

Cambiar los permisos de un archivo oculto

El equipo de investigación de mi organización archivó recientemente el archivo **project_x.txt**. No desean que nadie tenga permisos de escritura sobre este proyecto, pero el usuario y el grupo deben tener permisos de lectura.

El siguiente código demuestra cómo utilicé 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 Dec 20 15:36 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 20 15:36 ..
-r--r----- 1 researcher2 research_team  46 Dec 20 15:36 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 20 15:36 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Dec 20 15:36 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Dec 20 15:36 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Dec 20 15:36 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Dec 20 15:36 project_t.txt
researcher2@3213bbc1d047:~/projects$
```

Las dos primeras líneas de la captura de pantalla muestran los comandos que ingresé, y las demás 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 para el usuario y el grupo, y añadí 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 añadí permisos de lectura al grupo con **g+r**.

Cambiar los permisos de un directorio

Mi organización quiere que únicamente el usuario **researcher2** tenga acceso al directorio *drafts* y a su contenido. Esto significa que nadie más aparte de *researcher2* debe tener permisos de ejecución.

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

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

La salida aquí muestra el listado de permisos para varios archivos y directorios. La línea 1 indica el directorio actual (*projects*), y la línea 2 indica el directorio padre (*home*). La línea 3 indica un archivo regular llamado *.project_x.txt*. La línea 4 corresponde al directorio (*drafts*) con permisos restringidos. Aquí se puede ver que solo **researcher2** tiene permisos de ejecución. Anteriormente se había determinado que el grupo tenía permisos de ejecución, por lo que utilicé el comando **chmod** para eliminarlos. El usuario **researcher2** ya tenía permisos de ejecución, así que no fue necesario añadirlos.

Resumen

Cambié múltiples permisos para que coincidieran con el nivel de autorización que mi organización requería para los archivos y directorios dentro del directorio *projects*. El primer paso fue usar **ls -la** para verificar los permisos del directorio. Esto guio mis decisiones en los pasos siguientes. Luego utilicé el comando **chmod** varias veces para modificar los permisos de archivos y directorios.