

Permisos de archivo en Linux

Descripción del proyecto

El equipo de investigación de mi organización necesita actualizar los permisos de archivo para ciertos archivos y directorios dentro del directorio `projects`. Actualmente, los permisos no reflejan el nivel de autorización que debe otorgarse. Revisar y actualizar estos permisos ayudará a mantener el sistema seguro. Para completar esta tarea, realicé las siguientes acciones:

Comprobar detalles del archivo y del directorio

El siguiente código muestra cómo utilicé los comandos de Linux para determinar los permisos que se establecieron 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 otras líneas muestran la salida. El código enumera todos los contenidos del directorio `projects`. Utilicé el comando `ls` con la opción `-la` para mostrar una lista detallada de los contenidos de los archivos que también arrojó 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 en cada archivo o directorio.

Describir la cadena de permisos

La cadena de 10 caracteres se puede desglosar para determinar quién tiene autorización para acceder al archivo y qué permisos específicos tiene. Los caracteres y lo que representan son los siguientes:

- **1º carácter:** es una `d` o un guion (`-`) e indica el tipo de archivo. Si es una `d`, se trata de un directorio. Si es un guion (`-`), se trata de un archivo normal.

- **2.º al 4.º carácter:** indican los permisos de lectura (**r**), escritura (**w**) y ejecución (**x**) para el usuario. Cuando uno de estos caracteres es un guion (**-**) en lugar de una letra, indica que no se le ha concedido este permiso al usuario.
- **5.º al 7.º carácter:** indican los permisos de lectura (**r**), escritura (**w**) y ejecución (**x**) para el grupo. Cuando uno de estos caracteres es un guion (**-**) en lugar de una letra, indica que este permiso no está concedido al grupo.
- **8.º al 10.º carácter:** indican los permisos de lectura (**r**), escritura (**w**) y ejecución (**x**) para otros usuarios. Este tipo de propietario está compuesto por todos los demás usuarios del sistema, aparte del usuario y el grupo. Cuando uno de estos caracteres es un guion (**-**) en lugar de una letra, eso indica que este permiso no está concedido a otros usuarios.

Supongamos que los permisos de archivo para `project_t.txt` son `-rw-rw-r--`. Dado que el primer carácter es un guion (**-**), esto indica que `project_t.txt` es un archivo, no un directorio. Tanto el segundo como el quinto y el octavo carácter son **r**. Esto indica que el usuario, el grupo y otros usuarios tienen permisos de lectura. El tercer carácter y el sexto son **w**, lo que indica que solo el usuario y el grupo tienen permisos de escritura. Nadie tiene permisos de ejecución para `project_t.txt`.

Cambiar permisos de archivo

La organización determinó que otros usuarios no deberían tener acceso de escritura a ninguno de sus archivos. Para cumplir con esto, me basé en los permisos de archivo que había obtenido como resultado. Determiné que debía eliminar el permiso de escritura de otros usuarios para `project_k.txt`.

El siguiente código muestra cómo usé los comandos de Linux para hacer esto:

```
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, 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

Recientemente, el equipo de investigación de mi organización archivó `project_x.txt`. No quieren que nadie tenga acceso de escritura a este proyecto, pero el usuario y el grupo deben tener acceso de lectura.

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 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 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

Mi organización solo quiere que el usuario `researcher2` tenga acceso al directorio `drafts` y sus contenidos. Esto significa que nadie más que `researcher2` debe tener permisos de ejecución.

El siguiente código muestra cómo utilicé los 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$
```

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

Cambié varios permisos para que coincidieran con el nivel de autorización que mi organización quería para archivos y directorios en el directorio `projects`. El primer paso en esto fue usar `ls-la` para verificar los permisos para el directorio. En esta información basé mis decisiones para los siguientes pasos. Luego, usé el comando `chmod` varias veces para cambiar los permisos en archivos y directorios.