File permissions in Linux

Project description

- Check permissions for files in a directory
- Check for incorrect file permissions and change permissions as needed
- Remove unauthorized access to a directory

Check file and directory details

The following code demonstrates how I used Linux commands to determine the existing permissions set for files and directories in the file system.

```
researcher2@cd369f6cb68d:~$ pwd
/home/researcher2
researcher2@cd369f6cb68d:~$ ls
projects
researcher2@cd369f6cb68d:~$ cd projects
researcher2@cd369f6cb68d:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jun  8 01:14 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jun  8 01:14 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  8 01:14 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_t.txt
researcher2@cd369f6cb68d:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:14 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:32 ..
-rw--w---- 1 researcher2 research_team   46 Jun  8 01:14 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  8 01:14 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jun  8 01:14 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  8 01:14 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_t.txt
```

Describe the permissions string

A 10-character string indicates how the permissions on the file are set. For instance, a directory with full permissions for all owner types would be `drwxrwxrwx`

For instance, consider `project_k.txt` with permissions `-rw-rw-rw-`. This indicates that it is a regular file with read and write permissions granted to all users, groups, and others, but not execution.

Change file permissions

The organization prohibits others from having write access to any files. However, `project_k.txt` currently permits others to have write access. Therefore, the Linux command `chmod` is used with the following arguments: `o=r project_k.txt`, allowing others to have read permission only within the `project_k.txt file`.

```
researcher2@cd369f6cb68d:~/projects$ chmod o=r project_k.txt
researcher2@cd369f6cb68d:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:14 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:32 ..
-rw--w---- 1 researcher2 research_team   46 Jun  8 01:14 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  8 01:14 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  8 01:14 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_t.txt
```

Change file permissions on a hidden file

The research team within my organization has recently archived `.project_x.txt`. In this archival state, it's essential to restrict write access to all users while allowing both the user and group to retain read access.

```
researcher2@cd369f6cb68d:~/projects$ chmod u=r,g=r .project_x.txt
researcher2@cd369f6cb68d:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:14 .
drwxr-xr-x 3 researcher2 research_team 4096 Jun  8 01:32 ..
-r--r----- 1 researcher2 research_team   46 Jun  8 01:14 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jun  8 01:14 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  8 01:14 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_t.txt
```

The first two lines showcase the commands I inputted, while the subsequent lines present the output generated by the second command. It's evident that `.project_x.txt` is a hidden file due to its commencement with a period (.). In this instance, I executed commands to alter permissions. I allowed only read permissions to both the user and group using `u=r,g=r`.

Change directory permissions

My organization mandates that only the user `researcher2` should have access to the drafts directory and its contents. This entails restricting execution permissions to everyone except `researcher2`. In the provided Linux command sequence, I showcase the modification of permissions:

```
researcher2@cd369f6cb68d:~/projects$ chmod g-x drafts
researcher2@cd369f6cb68d:~/projects$ ls -la
total 32
drwxr-xr-x  3 researcher2 research_team 4096 Jun  8 01:14 .
drwxr-xr-x  3 researcher2 research_team 4096 Jun  8 01:32 ..
-r--r----- 1 researcher2 research_team   46 Jun  8 01:14 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Jun  8 01:14 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jun  8 01:14 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jun  8 01:14 project_t.txt
```

The displayed output illustrates the permissions of various files and directories. Line 1 designates the current directory as 'projects,' while line 2 denotes its parent directory, 'home.' Line 3 represents a regular file named `.project_x.txt`, and line 4 indicates the 'drafts' directory with restricted permissions. As evident, solely `researcher2` possesses execute permissions. Since it was previously determined that the group had execute permissions, I utilized the `chmod` command to revoke them. Given that `researcher2` already had execute permissions, no additional action was necessary.

Summary

I adjusted the permissions of files and directories within the projects directory to align with my organization's authorization standards. Initially, I reviewed the permissions using 'ls -la' to assess the existing settings. This guided my subsequent actions as I utilized the 'chmod' command several times to modify permissions accordingly.