File permissions in Linux

Project description

In this project, I used Linux commands to manage file permissions in `/home/researcher2/projects` directory. The goal was to verify proper user authorization and update permissions to maintain system security. I made sure all the authorized users have appropriate access and removed any unnecessary users.

Check file and directory details

I used the following command to check the file and directory permissions.

```
researcher2@fd394c85248e:~$ ls -la /home/researcher2/projects
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 17:57 .
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 18:28 ..
-rw--w---- 1 researcher2 research_team   46 Apr 19 17:57 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Apr 19 17:57 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Apr 19 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr 19 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_t.txt
researcher2@fd394c85248e:~$
```

The first line of the screenshot displays the `ls -la` command that I used to display all the following file contents, including the hidden one. The output of my command shows that there is only one hidden file named `.project_x.txt` and a directory named `drafts`.

Describe the permissions string

There are 10 character permission strings used in Linux to determine who is authorized to access what part of the directory.

- 1st character : The 1st character is `(d)` to state a directory file however, hyphen `(-)` is used as the 1st character if it is just a regular file.
- 2-4th character : These 3 characters indicate the read (`r`), write (`w`) and execute (`x`) permissions for the user. If the user does not have one of the permissions then, it is indicated with a hyphen `(-)`.
- 5-7th character : The 3 characters indicate read (`r`), write (`w`) and execute (`x`) permissions for the group. However, if the group is not permitted with one or any of the permissions it is indicated with a hyphen `(-)`.

- 8-10th character: The last 3 characters indicate read (`r`), write (`w`) and execute (`x`) permissions for others and if they are not permitted with one or any of the permissions it is indicated with a hyphen (`-`).

For example, for `project_r.txt`, `-rw-rw-r--`, indicates that it is a file, not a directory because of the 1st character being (`-`). The 2-4th characters indicate that the user only has read (`r`) and write (`w`) permissions and not execute permission, as it is stated by (`-`) and not (`x`). Similarly, the group also only has read (`r`) and write (`w`) permissions, with others only having 1 permission i.e. read (`r`).

Change file permissions

My organization stated that they did not want other to have write access to any of their files. I identified that other directories and files do not have write access to other except for the file `project_k.txt` so, I used the following command to remove write access for other.

```
researcher2@fd394c85248e:~/projects$ chmod o-w project_k.txt
researcher2@fd394c85248e:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 17:57 .
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 18:28 ..
-rw--w---- 1 researcher2 research_team   46 Apr 19 17:57 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Apr 19 17:57 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr 19 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_t.txt
researcher2@fd394c85248e:~/projects$
```

The first line of the code states that I changed the file permission using `chmod o-w` to remove write permission for other. Then, I used the `ls -la` command to show the changes that have been made.

Change file permissions on a hidden file

The research team at my organization archived the folder named `.project_x.txt,` and they did not want anyone to have write and execute permission to the file. However, they did want the user and group to have read permission. So, I used the `chmod` command to change permissions.

```
researcher2@fd394c85248e:~/projects$ chmod g-w,u-w,g+r .project_x.txt
researcher2@fd394c85248e:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 17:57 .
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 18:28 ..
-r--r----- 1 researcher2 research_team   46 Apr 19 17:57 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Apr 19 17:57 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr 19 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_t.txt
researcher2@fd394c85248e:~/projects$
```

At first, I identified the hidden file with `(.)` in front of the file name i.e. `.project_k.txt`. Originally the file had write access to the user and the group and did not have the read access for the group so, I first used (`g-w`) and (`u-w`) to remove write permission for the group and the user respectively then, lastly wrote (`g+r`) to add reading access for the group.

Change directory permissions

My organization only wanted the `researcher2` users to have access to the `drafts` directory and its contents (no one should be able to execute it other than the user) however, initially the group also had access to execute the file.
So, I used the `chmod (g-x)` command to change the file permission to remove the group to have execute permission; however, the user already had the read permission so, I did not have to change any command.

```
researcher2@fd394c85248e:~/projects$ chmod g-x drafts
researcher2@fd394c85248e:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 17:57 .
drwxr-xr-x 3 researcher2 research_team 4096 Apr 19 18:28 ..
-r--r----- 1 researcher2 research_team   46 Apr 19 17:57 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Apr 19 17:57 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr 19 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr 19 17:57 project_t.txt
researcher2@fd394c85248e:~/projects$
```

Summary

I changed permissions for multiple files and directories listed in the `projects` directory to match the level of authorization my organization wanted. First, I used the `ls -la` command to view permissions to the files and directories, then I used the `chmod` command to change permissions on each as mentioned.