File permissions in Linux

Project description

The goal of this lab is to ensure the permissions and authorizations are set correctly for all parties. I will go into the according teams directories, files and hidden files ensuring that access is limited for the appropriate owners.

Check file and directory details

```
researcher2@02a5f898899c:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 19:23 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 19:52 ..
-rw--w---- 1 researcher2 research_team   46 Dec 10 19:23 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 10 19:23 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 10 19:23 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 10 19:23 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 19:23 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 19:23 project_t.txt
```

The standard structure file will contain `/home/researcher2/projects` with a subdirectory named `drafts.` Here I use the `ls -la` command to display all permissions, as well as hidden files, that are started off with the `projects` directory.

Displayed above, we can see this directory belongs to the ‘research team’, along with all the current permissions.

Describe the permissions string

There is a 10-character string that indicates each owner and its active permissions: `drwxrwxrwx.`

- D = The directory
- First rwx = The User access and their permissions: read ‘r’, write ‘w’, and execute ‘x’.
- Second rwx = The Group access and their permissions: read ‘r’, write ‘w’, and execute ‘x’.
- Final rwx = The Other access and their permissions: read ‘r’, write ‘w’, and execute ‘x’.

Change file permissions

```
researcher2@f0b562892624:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Dec 10 20:42 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 10 20:42 project_k.txt
```

There should not be ‘write’ permission granted to ‘other’ owners, so authorization must be removed.

```
researcher2@f0b562892624:~/projects$ chmod o-w project_k.txt
researcher2@f0b562892624:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Dec 10 20:42 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 10 20:42 project_m.txt
```

Inputting the command `chmod o-w project_k.txt` has now modified it to remove the ‘other’ access from writing permissions.
We then input the `ls -l` command to confirm that access was removed accordingly.

However, `project_m.txt` has some permissions that also need to be changed since this is a restricted file. Only the ‘user’ should have ‘write’ and ‘read’ permission granted, so we will need to remove permissions from ‘other’ and ‘group’.

```
researcher2@f0b562892624:~/projects$ chmod g-r project_m.txt
```

The only input command that needed to be changed was the ‘group’ readability permission.

After inputting the `ls -l` command, we can then see the removed ‘read’ access from the ‘group’.

```
-rw------- 1 researcher2 research_team   46 Dec 10 20:42 project_m.txt
```

Change file permissions on a hidden file

Now, we need to look into the hidden files on the `project` directory to ensure that the authorizations are correct.

```
researcher2@f0b562892624:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:42 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:56 ..
-rw--w---- 1 researcher2 research_team   46 Dec 10 20:42 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 10 20:42 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_k.txt
-rw------- 1 researcher2 research_team   46 Dec 10 20:42 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_t.txt
researcher2@f0b562892624:~/projects$ chmod u-w,g-w,g+r .project_x.txt
```

Inputting the command ls -la will showcase all regular and hidden files or directories in the working directory and subdirectory. Subdirectory in this case being ‘draft’. You can see there is one hidden file called .project_x.txt and the current permissions granted. You can also tell a hidden file by the ‘.’ symbol prior to the file name.
The current hidden file .project_x.txt should not have ‘write’ access to anyone. As of now, the user and group have that access and should only be able to read that archived file.

```
researcher2@f0b562892624:~/projects$ chmod u-w,g-w,g+r .project_x.txt
```

Here is the command code that was needed. We removed ‘write’ permissions from both the ‘user’ and ‘group’. However, ‘group’ was missing readability access so that had to be granted as well with the command g+r.

```
researcher2@f0b562892624:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:42 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:56 ..
-r--r----- 1 researcher2 research_team   46 Dec 10 20:42 .project_x.txt
```

Now, with the ls -la command, we can see the updated and correct permissions have now been completed.

Change directory permissions

Lastly, only the researcher2 ‘user’ should have access to the drafts directory and ‘execute’ privileges. As of now, the ‘group’ also has that permission. The ‘group’ must be restricted from ‘execute’ access.

```
drwx--x--- 2 researcher2 research_team 4096 Dec 10 20:42 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_k.txt
-rw------- 1 researcher2 research_team   46 Dec 10 20:42 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_t.txt
```

As of now, we will need to input the command to modify the accessibility of the group.

```
researcher2@f0b562892624:~/projects$ chmod g-x drafts
```

This command will remove the ‘execute’ permissions from the ‘group’ owners of the draft directory.

```
researcher2@f0b562892624:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:42 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 20:56 ..
-r--r----- 1 researcher2 research_team   46 Dec 10 20:42 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Dec 10 20:42 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_k.txt
-rw------- 1 researcher2 research_team   46 Dec 10 20:42 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 20:42 project_t.txt
```

To confirm that all authorizations and permission in the directories, subdirectories, files, and hidden files are correct, we use the ls -la command. This displays all updated and correct permissions have now been modified.

Summary

With everything now complete, we have been able to ensure that all the proper permission accessibilities have been granted or restricted. Utilizing concepts like principles of least privilege helps to make sure availability is limited as much as possible to the appropriate owners. Commands like `ls -l` and `ls -la` were highly important to display where we currently stood with the permission and what needed to be changed. As well as confirming after changes were made with the `chmod` to modify the appropriate authorizations, the `ls -la` is very critical to double-check your work.