File permissions in Linux

Project description

The task is to review the existing permissions on the file system and assess whether they match the intended authorization levels. If the permissions do not align with the appropriate access rights, necessary adjustments should be made to authorize the correct users and remove any unauthorized access. The goal is to ensure that the file system permissions are properly configured to meet the required security measures and access controls.

Check file and directory details

```
researcher2@41f70e6936e2:~$ cd projects
researcher2@41f70e6936e2:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jul 31 17:44 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 31 17:44 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 31 17:44 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_t.txt
```

1. I navigated to the projects directory using the **cd** command.
2. I listed permissions in that projects directory for all files and directories.

Describe the permissions string

```
drwx--x--- 2 researcher2 research_team 4096 Jul 31 17:44 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 31 17:44 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 31 17:44 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_t.txt
```

The permissions string is a set of 10 characters that display read, write, and execute permissions for the user, group, and other. If the string starts with "**d**" this means that the permissions are for a directory. Similarly, a "**-**" would mean the permissions are for a file. The next nine characters, "**rwx**," are permissions assigned to the user, group, and other group. Each groups gets three permissions, and if any permission appears as "**-**" then they do not have that specific authorization.

Change file permissions

```
researcher2@41f70e6936e2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 17:44 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 31 17:44 ..
-rw--w---- 1 researcher2 research_team   46 Jul 31 17:44 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jul 31 17:44 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 31 17:44 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 31 17:44 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 31 17:44 project_t.txt
researcher2@41f70e6936e2:~/projects$ chmod o-w project_k.txt
```

1. Displayed hidden files and their permissions using **ls -la**
2. Changed file permissions for other users to read only on project_k.txt using **chmod o-w project_k.txt**

Change file permissions on a hidden file

```
researcher2@41f70e6936e2:~/projects$ chmod u-w,g+r,g-w .project_x.txt
```

1. Changed user and group permissions to read only for the hidden file .project_x.txt

Change directory permissions

[Add content here.]

```
researcher2@41f70e6936e2:~/projects$ chmod g-x drafts
```

1. Changed group directory permissions so only the user can access the **drafts** directory.

Summary

In this project, I was tasked with modifying permissions on a Linux system using different command-line instructions. The objective was to ensure that the file system access rights aligned with the intended authorization levels. Through the use of various Linux commands, I successfully granted appropriate access to authorized users while removing unauthorized access to enhance the security and integrity of the system.