File permissions in Linux

Project description

This project focuses on using Linux commands to manage file permissions, ensuring that authorization aligns with file access requirements. The primary objective is to check the permissions for all files in a specified directory, including hidden files, to evaluate their compatibility with the desired access restrictions.

This project will utilize various Linux commands to navigate directories, list files, and examine permissions. Specifically, the **ls** and **chmod** commands will be employed to navigate through the directory structure, list all files, and identify files with specific permissions.

The project will conclude with a summary of the permissions for all files in the specified directory, ensuring that the authorization matches the intended access requirements.

Check file and directory details

To examine the directory structure and permissions, I navigated to the projects directory using the **cd projects** command. Then, I employed the **ls -la** command to retrieve a detailed overview of the contents and permissions of the 'projects' directory, including hidden files. This yielded a complete understanding of the directory's organization and access controls.

The permissions of the files in the 'projects' directory are as follows:

```
researcher2@650d4cf73818:~$ cd projects
researcher2@650d4cf73818:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 00:37 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 01:26 ..
-rw--w---- 1 researcher2 research_team   46 Dec 10 00:37 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 10 00:37 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 10 00:37 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 10 00:37 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_t.txt
```

Describe the permissions string

Directory Permissions
In the current directory (.), the owner (researcher2) has read, write, and execute permissions, and members of the group (research team) have read and execute permissions. Everyone else has read and execute permissions. Similarly, in the parent directory (..), the owner (researcher2) has read, write, and execute permissions, and members of the group (research team) have read and execute permissions. Everyone else also has read and execute permissions.

File Permissions

- **.projects_x.txt**: This is a hidden file with read and write permissions for the owner, write permissions for the group (including group members and others), and no permissions for others.

- **drafts**: This is a directory with read, write, and execute permissions for the owner, execute permissions for the group, and no permissions for others.

- **project_k.txt**: This is a file with read and write permissions for everyone. This means that everyone can read and modify the file. However, only the owner of the file can create new files within the directory.

- **project_m.txt**: This file has read and write permissions for the owner, read permissions for the group, and no permissions for others. This means that the owner and members of the group can read and modify the file, but others cannot.

- **project_r.txt**: This file has read and write permissions for the owner and members of the group and read permissions for others. This means that the owner, members of the group, and others can read the file, but only the owner and members of the group can modify the file.

- **project_t.txt**: This file has read, write, and execute permissions for the owner and members of the group, and read permissions for others. This means that the owner, members of the group, and others can read, modify, and execute the file.

Change file permissions

The file project_k.txt initially granted write permissions to other users. To rectify this, I employed the command **chmod o-w project_k.txt** to revoke write permissions from users other than the owner.

The file project_m.txt is intended for restricted access, ensuring that only the owner should possess read or write privileges. I implemented the command **chmod g-r project_m.txt** to eliminate read and write permissions for the group, effectively granting exclusive access to the owner.

```
researcher2@650d4cf73818:~/projects$ chmod o-w project_k.txt
researcher2@650d4cf73818:~/projects$ chmod g-r project_m.txt
researcher2@650d4cf73818:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 00:37 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 01:26 ..
-r--r----- 1 researcher2 research_team   46 Dec 10 00:37 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 10 00:37 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_k.txt
-rw------- 1 researcher2 research_team   46 Dec 10 00:37 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_t.txt
```

Change file permissions on a hidden file

The hidden file .project_x.txt is intended to serve as an archive and should be protected from unauthorized modifications. Initially, both the owner and group had write permissions for the file, which conflicted with its intended purpose.

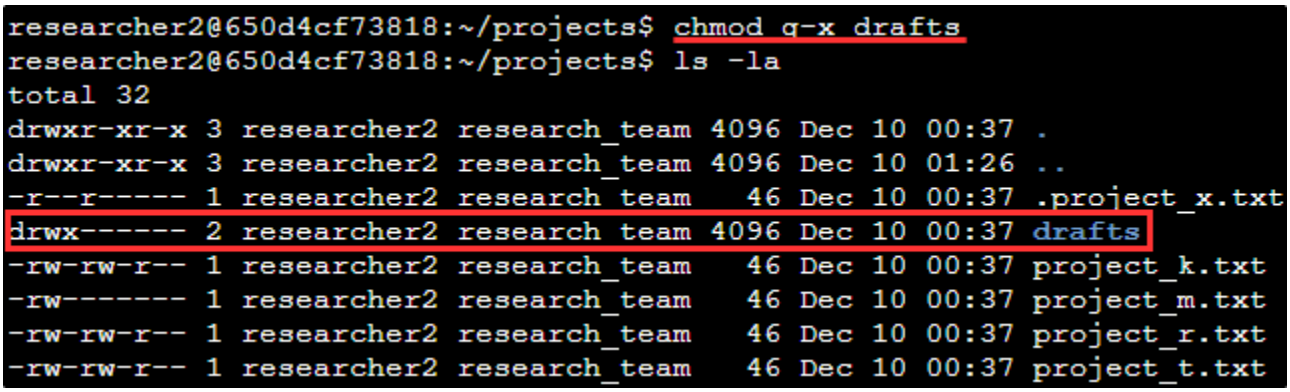
To address this, I executed the command **chmod u-w,g-w,g+r .project_x.txt**. This command effectively revoked write permissions for both the owner and group, ensuring that only read access is granted. The addition of **g+r** ensures that group members can still view the file's contents, while maintaining its archived state.

```
researcher2@650d4cf73818:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@650d4cf73818:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 00:37 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 10 01:26 ..
-r--r----- 1 researcher2 research_team   46 Dec 10 00:37 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 10 00:37 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_k.txt
-rw------- 1 researcher2 research_team   46 Dec 10 00:37 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 10 00:37 project_t.txt
```

Change directory permissions

The drafts directory holds sensitive information and should be accessible only to the owner, researcher2.

To ensure that only the owner could navigate and access the directory's contents, I utilized the command **chmod g-x drafts** to remove the execute permission for the group.



Summary

This project analysed the permissions of files and directories in the 'projects' directory. It found that the permissions were not secure and modified the permissions to ensure that only authorized users could access the files. Here is a summary of the changes made:

- The file project_k.txt initially granted write permissions to everyone. This was changed to only the owner with the command **chmod o-w project_k.txt**.
- The file project_m.txt was intended for restricted access and only the owner should possess read or write privileges. This was changed to exclusive access to the owner by executing the command **chmod g-r project_m.txt**.
- The file .project_x.txt is intended to serve as an archive and should be protected from unauthorized modifications. The initial permissions allowed both the owner and group to write to the file, contradicting its purpose. This was fixed by executing the command **chmod u-w,g-w,g+r .project_x.txt**. This command revoked write permission for both the owner and group and granted read permission to the group.
- The directory drafts hold sensitive information and should be accessible only to the owner. The initial permissions allowed the group to execute the directory, allowing them to access its contents. This was fixed by executing the command **chmod g-x drafts**. This removes the execute permission for the group and ensures that only the owner can navigate and access the directory's contents.