File permissions in Linux

Project description

In this project directory, I have demonstrated the ability to manage file permissions & viewing permissions using commands and describing permissions string. I work to ensure that proper permissions were granted to specific types of owners and specific files and directories to meet the organizational needs. This configuration represents separation of duties and principle of least privileges.

Check file and directory details

In order to check file and directory details, I would have to type in the following command, `ls - l`***. This command displays details of all the files and directories in this current directory such as permission strings, username, group name, and current date/time that I have run this command.***

To check any hidden files in the current directory, I would have to type in the following command `ls -a`**.**

Screenshot displaying `ls -l` ***and*** `ls -a` ***command:***

```
researcher2@57fe32306bd4:~$ cd projects
researcher2@57fe32306bd4:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  3 17:45 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug  3 17:45 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 17:45 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 17:45 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 17:45 project_t.txt
researcher2@57fe32306bd4:~/projects$ ls -a
.  ..  .project_x.txt  drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
researcher2@57fe32306bd4:~/projects$
```

Describe the permissions string

Suppose for example, I choose the file name "project_m.txt". The permission string is on the very left of the screenshot that displayed "-rw-r-----". I will explain what every character in this string represents.

String characters
1: "-" represents that this is a file.
2-4: "rw-" represents the 'User' permission to read and write, but not execute.

5-7: "r--" represents the 'Group' permission to read, but not write nor execute.
8-10: "---" represents the 'Other' that <u>does not have</u> the permission to read, write, or to execute.

Change file permissions

The organization does not allow 'other' to be given write access to any files. Based on the previous screenshots, the file name 'project_k.txt' needed to have permissions modified.

The following command `chmod o-w project_k.txt` ***represents the command line to change the permissions of file 'project_k.txt'.***

<u>Here's what they represent</u>

chmod: *Command to change permission types to who*
o-w: *Revoked write access to other*
project_k.txt: *File name*

Screenshot displaying that project_k.txt that revoked access to write in

```
researcher2@57fe32306bd4:~/projects$ chmod o-w project_k.txt
researcher2@57fe32306bd4:~/projects$ ls-l
-bash: ls-l: command not found
researcher2@57fe32306bd4:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  3 17:45 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 17:45 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 17:45 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 17:45 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 17:45 project_t.txt
```

Change file permissions on a hidden file

The hidden file '.project_x.txt' should not have write permissions for anyone, but the user and group should be allowed to read the file.

Typing in the following command `chmod g+r,u-w,g-w .project_txt` ***modifies the permissions of '.project_txt' by giving 'group' permission to read and revoking access to both the 'user' and the 'group' to write.***

Screenshot displaying change of file permissions on a hidden file (next page)

```
researcher2@39c040ee4c42:~/projects$ ls -al
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug  3 19:15 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug  3 20:16 ..
-rw--w---- 1 researcher2 research_team   46 Aug  3 19:15 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug  3 19:15 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 19:15 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_t.txt
researcher2@39c040ee4c42:~/projects$ chmod g+r,u-w,g-w .project_x.txt
researcher2@39c040ee4c42:~/projects$ ls -al
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug  3 19:15 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug  3 20:16 ..
-r--r----- 1 researcher2 research_team   46 Aug  3 19:15 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Aug  3 19:15 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 19:15 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_t.txt
researcher2@39c040ee4c42:~/projects$
```

Change directory permissions

It was given that only 'researcher2' should be allowed to access the drafts directory and its contents.

Thus, the command `chmod g-x drafts` ***allows to remove the 'group' permission to access the drafts directory and its contents (also known as executable files/directories).***

Screenshot of changing directory permissions of commands and output

```
researcher2@39c040ee4c42:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  3 19:15 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 19:15 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_t.txt
researcher2@39c040ee4c42:~/projects$ chmod g-x drafts
researcher2@39c040ee4c42:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research_team 4096 Aug  3 19:15 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  3 19:15 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  3 19:15 project_t.txt
researcher2@39c040ee4c42:~/projects$
```

Summary

Overall, I was able to view permission strings and hidden files using the following commands: `ls -a`, `ls -l`, `ls -la` . The purpose was to see the general overview permissions on directories, files, and hidden files to ensure the right owners were granted which type of access.

Using chmod allows modifying any changes of ownerships to the files, hidden files, directories based on whether or not to revoke or grant permission to read, write, or execute.