File permissions in Linux

Project description

The purpose of this project was to learn about permissions on Linux, as well as how to add and rescind permissions to/from the different types of users.

Check file and directory details

In the initial steps of this project I first had to navigate to a directory, then list the contents and permissions of said directory (as shown below). To do this I had to first use the *cd* command to navigate, then the *ls- la* command to permissions and hidden files.

```
researcher2@08e54fcf7d20:~$ cd /home/researcher2/projects/drafts
researcher2@08e54fcf7d20:~/projects/drafts$ cd /home/researcher2/projects
researcher2@08e54fcf7d20:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 20 17:57 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 20 18:30 ..
-rw--w---- 1 researcher2 research_team   46 Feb 20 17:57 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 20 17:57 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Feb 20 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 20 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_t.txt
researcher2@08e54fcf7d20:~/projects$
```

Describe the permissions string

The permission string is the string of letters that represent the read, write, and execute permissions for the user, group, and other. In each of the strings the d represents the directory, r represents read privileges, w represents write privileges, and x represents execute privileges. After the d there are three segments of rwx; the first represents the user, the second represents the group, and the third represents other. A dash in the place of any of these letters in any of the three segments expresses the given group not having that kind of privilege.

```
drwxr-xr-x
drwxr-xr-x
-rw--w----
drwx--x---
-rw-rw-rw-
-rw-r-----
-rw-rw-r--
-rw-rw-r--
```

Change file permissions

For this project I was required to adjust permissions for user groups. As shown below in the first line and last, I removed write permissions for researcher2 project_k.txt. This was done using the command *chmod o-w project_k.txt*.

```
-rw-rw-rw- 1 researcher2 research_team   46 Feb 20 17:57 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 20 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_t.txt
researcher2@08e54fcf7d20:~/projects$ ls -a
.   .project_x.txt  project_k.txt  project_r.txt
..  drafts          project_m.txt  project_t.txt
researcher2@08e54fcf7d20:~/projects$ chmod o-w project_k.txt
researcher2@08e54fcf7d20:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Feb 20 17:57 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_k.txt
```

Also I was request to remove read and write privileges for the group in project_m.txt (shown in first and last line). I again used the *chmod* command.

```
-rw-r----- 1 researcher2 research_team   46 Feb 20 17:57 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_t.txt
researcher2@08e54fcf7d20:~/projects$ chmod g-r project_m.txt
researcher2@08e54fcf7d20:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Feb 20 17:57 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 20 17:57 project_k.txt
-rw------- 1 researcher2 research_team   46 Feb 20 17:57 project_m.txt
```

Change file permissions on a hidden file

Changing the file permission on a hidden file is a similar process, the only difference is that to display hidden files in a directory you must use the *ls -a* or *ls -la* Linux command.

Summary

Overall this project was a great opportunity to continue my Linux experience with commands that I will likely use as a Cybersecurity professtional.