File permissions in Linux

Project description

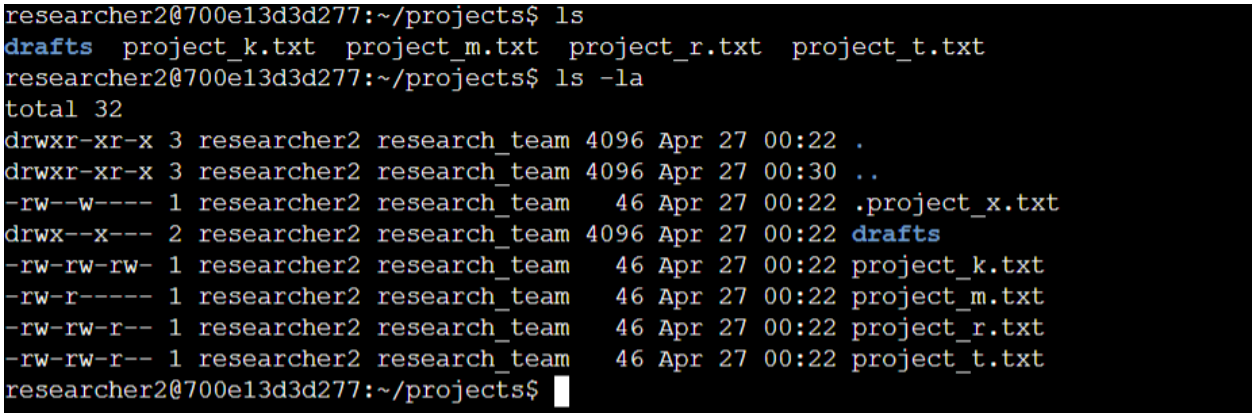
This project included adjusting file permissions for a fictitious research company. The following tasks were required:

1. Check the current file permission details in the “projects” directory.
2. Describe the permissions string.
3. Change the file permissions on all files in this directory so that only the user has writing access. Group and other are not to have writing access.
4. Change the permissions on the secret file so that no one may have writing privileges, and so that both the user and the group have reading privileges.
5. Change the “drafts” directory so that only the user has reading, writing, and execution privileges.
6. Use the command “ls -la” to view all files, including secret files, with their associated permissions to match the above descriptions.

Check file and directory details

In order to check file and directory permissions, I looked up the file’s contents with a listing command “ls”. This revealed line 2 of the screenshot below, where it listed “drafts, project_k.txt, project_m.txt, project_r.txt, project_t.txt”

Then I listed all the files, including hidden files with “ls -la”. This listed all the files and directories, including hidden files and their permissions demonstrated in lines 3-12 of the screenshot below.



Describe the permissions string

Here is a screenshot of the permissions from file "project_k.txt":

```
-rw-rw-rw- 1 researcher2 research team   46 Apr 27 00:22 project_k.txt
```

The first 10 characters of this screenshot indicate the access permissions for the User, Group, and Other and whether this is a file or a directory.

If it were to be a directory, there would be a "d" instead of a "-" for the first character of the 10 letter sequence. Since the screenshot above is a file, it has a "-".

The next nine characters are segmented into three groups. The first three describe the User permissions. The second three describe the Group permissions, and the third three describe the Other permissions. Each group of three has reading (r), writing (w), and execution (x) permissions in that order. In the screenshot above, all the User, Group, and Others have reading and writing access to the file, for example. If the letters are represented, then that User, Group, or Other have permission(s) to read, write, or execute that file/directory. If a "-" is present, then those permissions are not granted.

As seen by the project description, some permissions need to be changed.

Change file permissions

The first file that will have its permission edited, is the project_k.txt file. To do this, a "chmod" command is used where "g-w" indicates that the Group is having writing permission withdrawn and that "o-w" are having writing permissions withdrawn. These are assigned to project_k.txt as the second argument in this screenshot.

```
researcher2@700e13d3d277:~/projects$ chmod g-w,o-w project_k.txt
```

The same process is repeated for the project_r.txt file. "chmod" is used as a command to change the permissions for the file. "g-w" is used to take away writing permissions from the group.

```
researcher2@700e13d3d277:~/projects$ chmod g-w project_r.txt
```

The same process is repeated for project_t.txt.

```
researcher2@700e13d3d277:~/projects$ chmod g-w project_t.txt
```

Change file permissions on a hidden file

Since "ls -la" was used at the start to identify all files (including hidden ones) and their respective permissions, .project_x.txt permissions were already observed in the very first screenshot. Any file with "." at the start will hide the files in the directory. These permissions

need to be changed so that all writing privileges are taken away, and that reading privileges are given to all (User, Group, and Other).
The User and Group permissions have writing access and shouldn't. Therefore, "u-w,g-w" is used with "chmod" to take writing permissions away from the User and the Group. Reading privileges need to be granted to all for this file. To do this, "u+r,g+r,o+r" were used in the first argument as well to add those reading permissions. The second part of the argument, .project_x.txt, indicates that all these changes are assigned to that file.

```
researcher2@700e13d3d277:~/projects$ chmod u-w,g-w,u+r,g+r,o+r .project_x.txt
```

Change directory permissions

The directory "drafts" needs to be changed so that only the User has any permissions related to it. As seen in the first screenshot, the only change needed here is the Group's privilege of execution needing to be withdrawn.

This is done by using "chmod" with "g-x" to take away those permissions. The second part of the argument assigns where this change will occur, or the "drafts" directory.

```
researcher2@700e13d3d277:~/projects$ chmod g-x drafts/
```

Summary

Changing user permissions can significantly influence safety outcomes and honor the principle of least privilege in an organization. This task included key components and analyzing and adjusting these permissions at the User, Group, and Other level. All of these changes can be reviewed by using the same previous command, "ls -la". This screenshot shows all of the files and directories in accordance with the directions given for this assignment by the company, ensuring accuracy in user permissions and ensuring as much safety as possible in the organization of files and directories.

```
researcher2@700e13d3d277:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Apr 27 00:22 .
drwxr-xr-x 3 researcher2 research_team 4096 Apr 27 00:30 ..
-r--r--r-- 1 researcher2 research_team   46 Apr 27 00:22 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Apr 27 00:22 drafts
-rw-r--r-- 1 researcher2 research_team   46 Apr 27 00:22 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr 27 00:22 project_m.txt
-rw-r--r-- 1 researcher2 research_team   46 Apr 27 00:22 project_r.txt
-rw-r--r-- 1 researcher2 research_team   46 Apr 27 00:22 project_t.txt
```