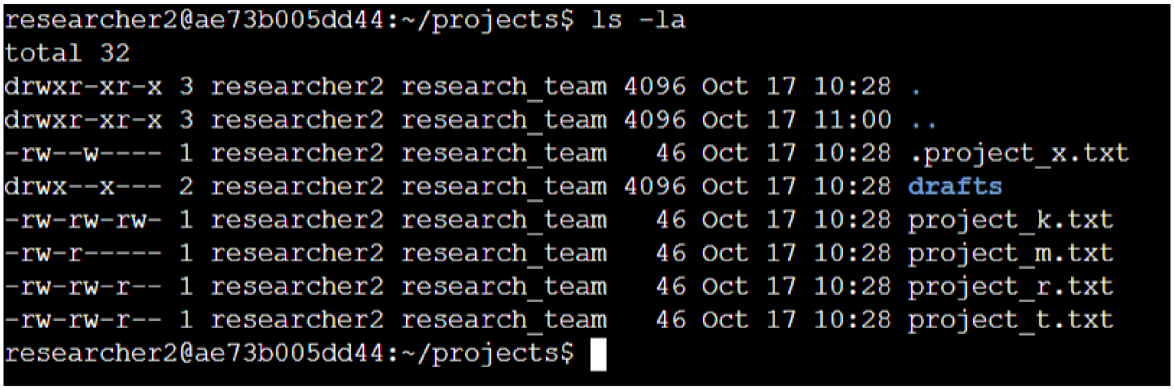
File permissions in Linux

Project description

The research team of my organization it's tasked to examine existing permissions on the file system, specifically, the files in the `projects` directory for the `researcher2` user. The access permissions aren't set appropriately. To correct this, I will check the permissions for all files in the directory, including any hidden files, to make sure that permissions align with the authorization that should be given. To complete this objective, I did the following tasks:

Check file and directory details

In the first place, the permissions of the projects directory and its files are explored. The Linux command I used to display this information is shown in the next screenshot:



```
researcher2@ae73b005dd44:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 10:28 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 11:00 ..
-rw--w---- 1 researcher2 research_team   46 Oct 17 10:28 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Oct 17 10:28 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Oct 17 10:28 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Oct 17 10:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_t.txt
researcher2@ae73b005dd44:~/projects$
```

The first line of the code indicates a command `(ls -la)` that displays all contents of the `projects` directory (including hidden files). The `ls` command displays the contents, and the `-la` command displays the hidden files and all their permissions. The command's output shows that in the `projects` directory there's a directory named `drafts`, a hidden file named `.project_x.txt` , and five other files. The left column displays the 10-character permissions strings on each file.

Describe the permissions string

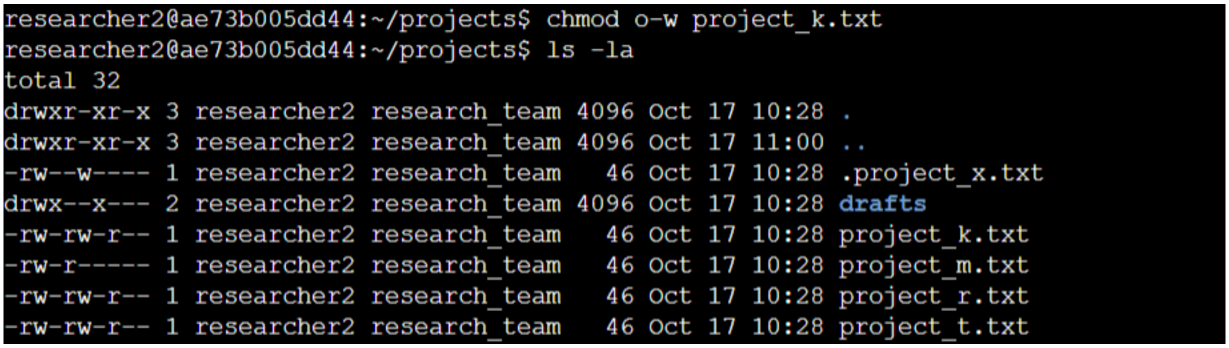
The 10-character permission string indicates who and the type of actions they are permitted to do in the file. It is made up of a first single character and three groups with three characters each.

- 1st character: It's either a `d` (directory) or `hyphen` (regular file), indicating the file type.

- 1st group(2nd-4th characters): The user’s permissions. These characters indicate the read `(r)`, write `(w)` and execute `(x)` permissions. If there’s a `hyphen`, it indicates that the permission is not granted.
- 2nd group(5th-7th characters): The group’s permissions. These characters indicate the read `(r)`, write `(w)` and execute `(x)` permissions. If there’s a `hyphen`, it indicates that the permission is not granted.
- 3rd group(8th-10th characters): The other’s permissions. These characters indicate the read `(r)`, write `(w)` and execute `(x)` permissions. If there’s a `hyphen`, it indicates that the permission is not granted.

Change file permissions

The organization decided that other users shouldn’t have write permissions. To comply with this, I checked the files’ permissions. The only one who permits the other users to write is the `project_k.txt file.` To change this I used the following Linux command:

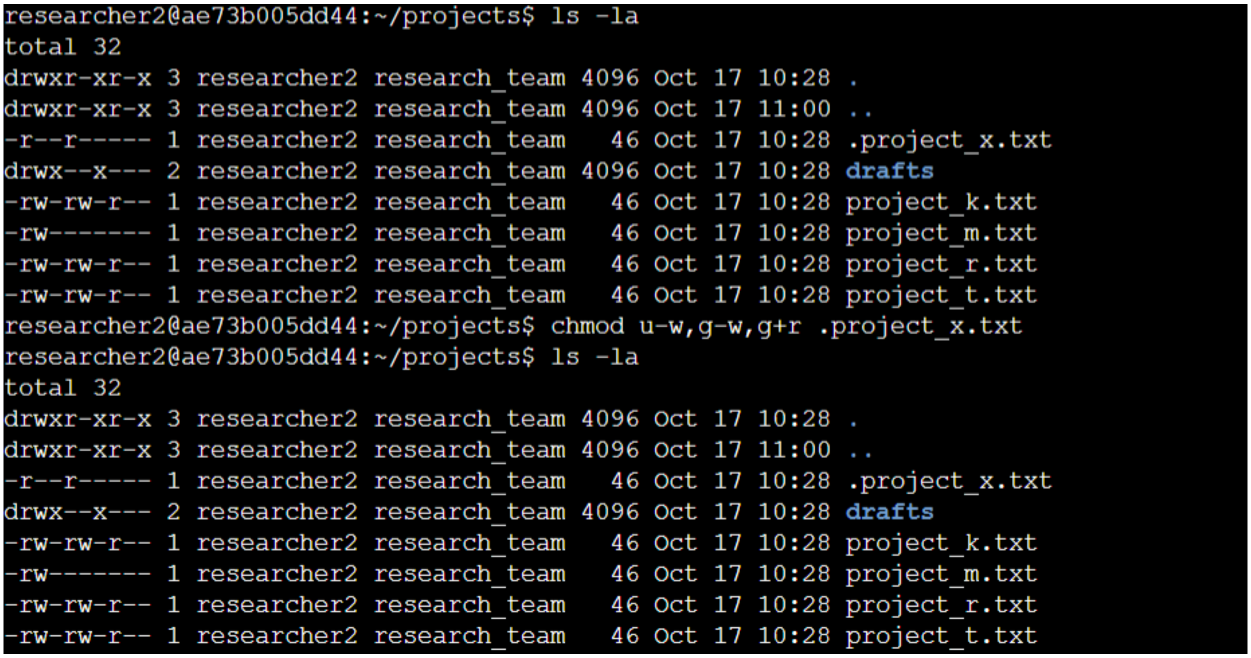


```
researcher2@ae73b005dd44:~/projects$ chmod o-w project_k.txt
researcher2@ae73b005dd44:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 10:28 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 11:00 ..
-rw--w---- 1 researcher2 research_team   46 Oct 17 10:28 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Oct 17 10:28 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Oct 17 10:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_t.txt
```

The first two lines display the commands used, and the rest of the lines are the output. The first line shows the command `chmod o-w project_k.txt`. The `chmod` command changes the indicated permissions. It is followed by the first argument, which indicates what permission should be changed (the first character indicates the type of user, the second indicates if the permission is removed or added and the last indicates the permission). The second argument is the file or directory to change. In this case, I removed write permissions from other users for the `project_k.txt file`. Finally, I used `ls -la` to check the updates I made.

Change file permissions on a hidden file

The organization doesn’t want anyone to have written permissions to a recently archived project, `.project_x.txt.` The user and group should still be able to read this file.
The following screenshot demonstrates the Linux commands used:



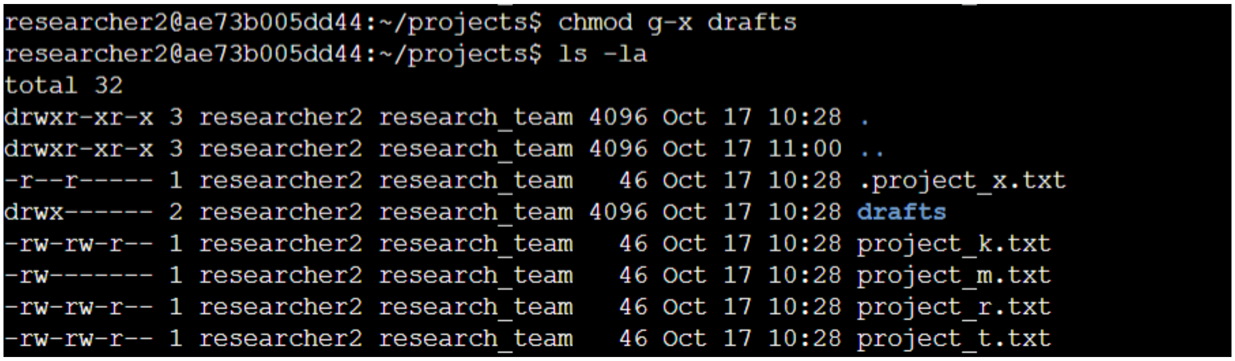
```
researcher2@ae73b005dd44:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 10:28 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 11:00 ..
-r--r----- 1 researcher2 research_team   46 Oct 17 10:28 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Oct 17 10:28 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_k.txt
-rw------- 1 researcher2 research_team   46 Oct 17 10:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_t.txt
researcher2@ae73b005dd44:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@ae73b005dd44:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 10:28 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 11:00 ..
-r--r----- 1 researcher2 research_team   46 Oct 17 10:28 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Oct 17 10:28 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_k.txt
-rw------- 1 researcher2 research_team   46 Oct 17 10:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_t.txt
```

The first two lines display the commands I used, and the other lines display the output. Thanks to the `-la` command, the hidden files are shown. I know `.project_x.txt` is a hidden file due to the initial period. With the `chmod` command I removed write permissions from the user and group `(u-w, g-w)`, and I added read permissions to the group `(g+r)`.

Change directory permissions

My organization only wants the `researcher2` to access (execute privileges) the `drafts` directory.

To accomplish this, I used the Linux commands shown in the screenshot:



```
researcher2@ae73b005dd44:~/projects$ chmod g-x drafts
researcher2@ae73b005dd44:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 10:28 .
drwxr-xr-x 3 researcher2 research_team 4096 Oct 17 11:00 ..
-r--r----- 1 researcher2 research_team   46 Oct 17 10:28 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Oct 17 10:28 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_k.txt
-rw------- 1 researcher2 research_team   46 Oct 17 10:28 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Oct 17 10:28 project_t.txt
```

In the output, the line 4 is the directory `(drafts)`, which indicates that the user and the group have execute permissions. To correct this, I used the `chmod` command to remove them `(g-x)`. The researcher2 user already had execute permissions, so I didn't have to add them. Finally, I used the `ls -la` command to check the changes.

Summary

Using multiple Linux commands, I changed files and directories permissions in the `projects` directory, to make sure that permissions aligned with the authorization my organization wanted. The first used command is the `ls -la` command, to check the permissions of all the files and directories, including hidden ones. Secondly, I used the `chmod` command multiple times to change the permissions of files and directories, as the company tasked.