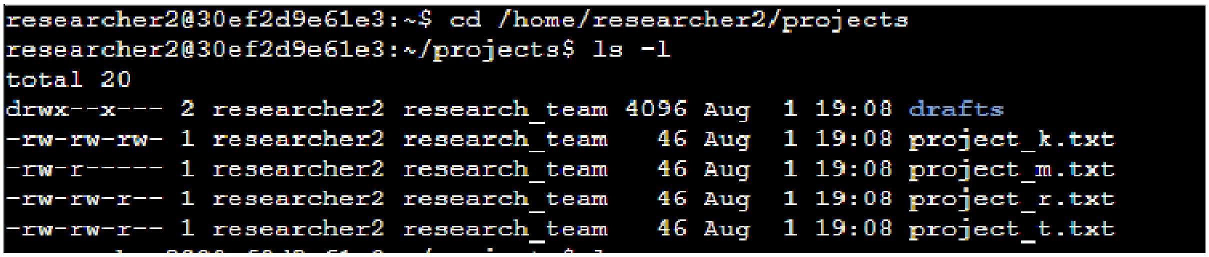
File permissions in Linux

Project description

In this project, I used Linux commands to manage and configure authorization. In this scenario, I examined permissions on files in the `/home/researcher2/projects` directory for the user `researcher2`, part of the group `research_team`.
I checked the permissions on all files in the directory (including hidden files) to ensure permissions aligned with the authorization given, and changed permissions when it didn't align.

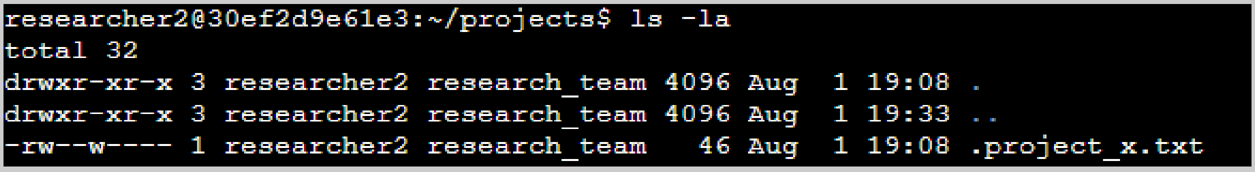
Check file and directory details

The following shows the current permissions for 5 files under the `/home/researcher2/projects` directory, which were found using command `ls -l`



```
researcher2@30ef2d9e61e3:~$ cd /home/researcher2/projects
researcher2@30ef2d9e61e3:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Aug  1 19:08 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Aug  1 19:08 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Aug  1 19:08 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  1 19:08 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  1 19:08 project_t.txt
```

The following shows hidden files and their permissions in the directory, found using commands `ls -a` (show hidden files) and `ls -la` (display permissions of hidden files)



```
researcher2@30ef2d9e61e3:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Aug  1 19:08 .
drwxr-xr-x 3 researcher2 research_team 4096 Aug  1 19:33 ..
-rw--w---- 1 researcher2 research_team   46 Aug  1 19:08 .project_x.txt
```

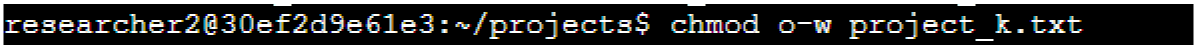
Describe the permissions string

In the above file/directory details, you can see the 10-character string for the file `project_k.txt` : `-rw-rw-rw-`
The initial hyphen indicates that this is a file, not a directory. Characters 2-4 indicate permissions given to the user (`researcher2`) , with "r" and "w" meaning the user currently has read and write permissions respectively. The following "-" indicates the user does not have execute permissions. Characters 5-7 indicate the permissions for the group (`research_team`) . They also have permissions to read and write, but not execute. Characters 8-10 indicate current permissions for other users; read, write, can't execute.

Change file permissions

The organization in this scenario does not allow other users to have write access to any files. The following file needs modifying:
`project_k.txt`

I used the following command to change the file permissions



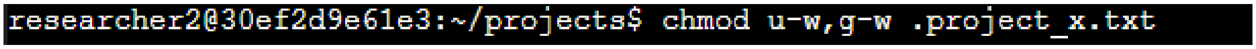
```
researcher2@30ef2d9e61e3:~/projects$ chmod o-w project_k.txt
```

The `chmod` “change mode” command instructs Linux to change permissions on the file. `o-w` instructs it to remove write access for other users. `Project_k.txt` directs it to what file I want to make the change to.

Change file permissions on a hidden file

In this scenario, `project_x.txt` is a hidden file because the research team archived it. Although the user and group are allowed to read the file, no one should have write permissions.

I used the following command to change the file permissions



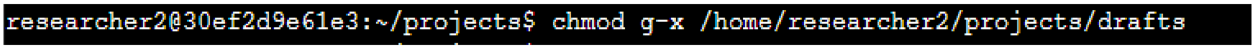
```
researcher2@30ef2d9e61e3:~/projects$ chmod u-w,g-w .project_x.txt
```

Using the same `chmod` command, I entered `u-w` and `g-w` to remove write access to the user and group respectively. The comma “,” in between distinguishes the different file owners. The period “.” in front of `project_x.txt` distinguishes it as a hidden file in Linux.

Change directory permissions

Current permissions allow the group to have execute access to the drafts directory, as indicated by the character string `drwx--x---`. Only the user (`researcher2`) should be allowed this access.

I used the following command to change the file permissions



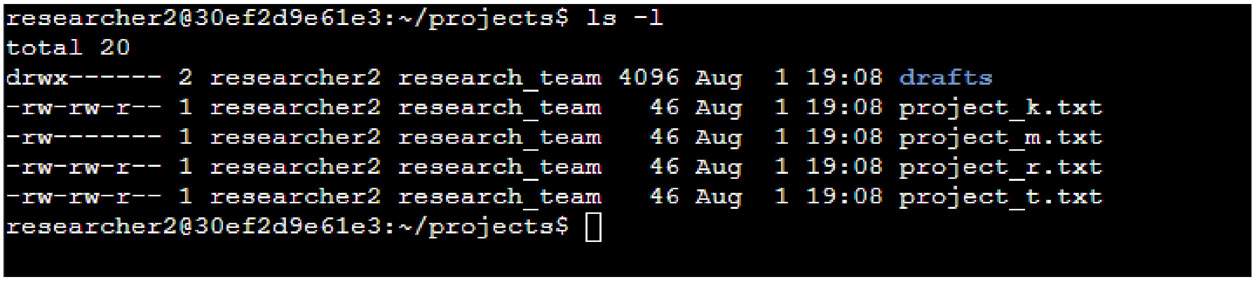
```
researcher2@30ef2d9e61e3:~/projects$ chmod g-x /home/researcher2/projects/drafts
```

As before, I use the `chmod` command to tell Linux I want to modify permissions. `g-x` indicates I wish to remove execute access from the group. The file path `/home/researcher2/projects/drafts` identifies the correct directory.

Summary

The permissions for the files within the `/home/researcher2/projects` directory have been modified to align with the authorization compliance of the organization in this scenario. Only the user now has execute permissions to the directory itself. Write permissions to directory files have been removed for other users, as well as to everyone for the hidden files. By applying the concept of least privilege, I have limited security risk to these files by modifying file/directory access based on the minimum working needs for each owner group.

The following shows correct permissions for the directory



```
researcher2@30ef2d9e61e3:~/projects$ ls -l
total 20
drwx------ 2 researcher2 research_team 4096 Aug  1 19:08 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Aug  1 19:08 project_k.txt
-rw------- 1 researcher2 research_team   46 Aug  1 19:08 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  1 19:08 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Aug  1 19:08 project_t.txt
researcher2@30ef2d9e61e3:~/projects$ 
```