File permissions in Linux

Project description

You are a security professional at a large organization. You mainly work with their research team. Part of your job is to ensure users on this team are authorized with the appropriate permissions. This helps keep the system secure.

Your task is to examine existing permissions on the file system. You'll need to determine if the permissions match the authorization that should be given. If they do not match, you'll need to modify the permissions to authorize the appropriate users and remove any unauthorized access.

Check file and directory details

This document displays the file structure of the /home/researcher2/projects directory and the permissions of the files and subdirectory it contains.
In the /home/researcher2/projects directory, there are five files with the following names and permissions:

- Project_k.txt
  - User = read, write,
  - Group = read, write
  - Other = read, write
- Project_m.txt
  - User = read, write
  - Group = read
  - Other = none
- Project_r.txt
  - User= read, write
  - Group = read, write
  - Other = read
- Project_t.txt
  - User = read, write
  - Group = read, write
  - Other = read
- .project_x.txt
  - User = read, write
  - Group = write

    - Other = none

There is also one subdirectory inside the projects directory named drafts. The permissions on drafts are:

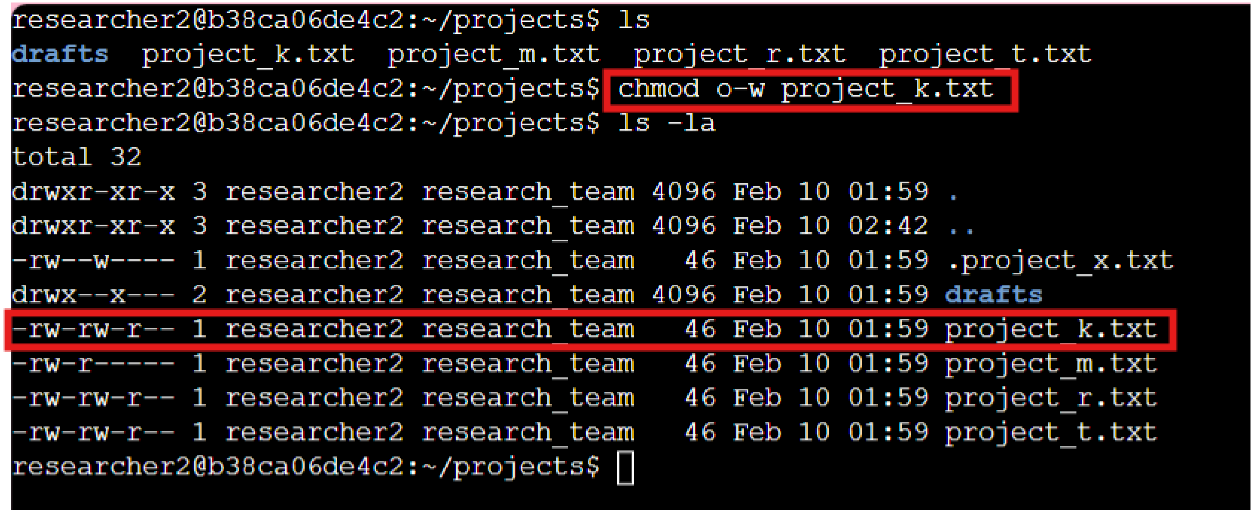
- User = read, write, execute
- Group = execute
- Other = none

Describe the permissions string

A standard 10 character string that records if the object is a file or directory and user, group, and other permissions.
The first character states if the object is a file or directory. - = file, d = directory.
Characters 2-4 are for the user. Character 2 is the read permission [r], 3rd is for write permissions [w], and 4th is for execute permissions [x]. This repeats for 5-7 for Group permissions and 8-10 for Other permissions.
Example: -rw-r-----
This is a file where the user has permissions to read and write but NOT to execute, the group has permission to read but NOT to write or execute, and others have NO permissions.

Change file permissions

The organization does not allow others to have write access to any files. Use a Linux command to modify these permissions.
Based on current file permissions, project_k.txt's permissions need to be changed.

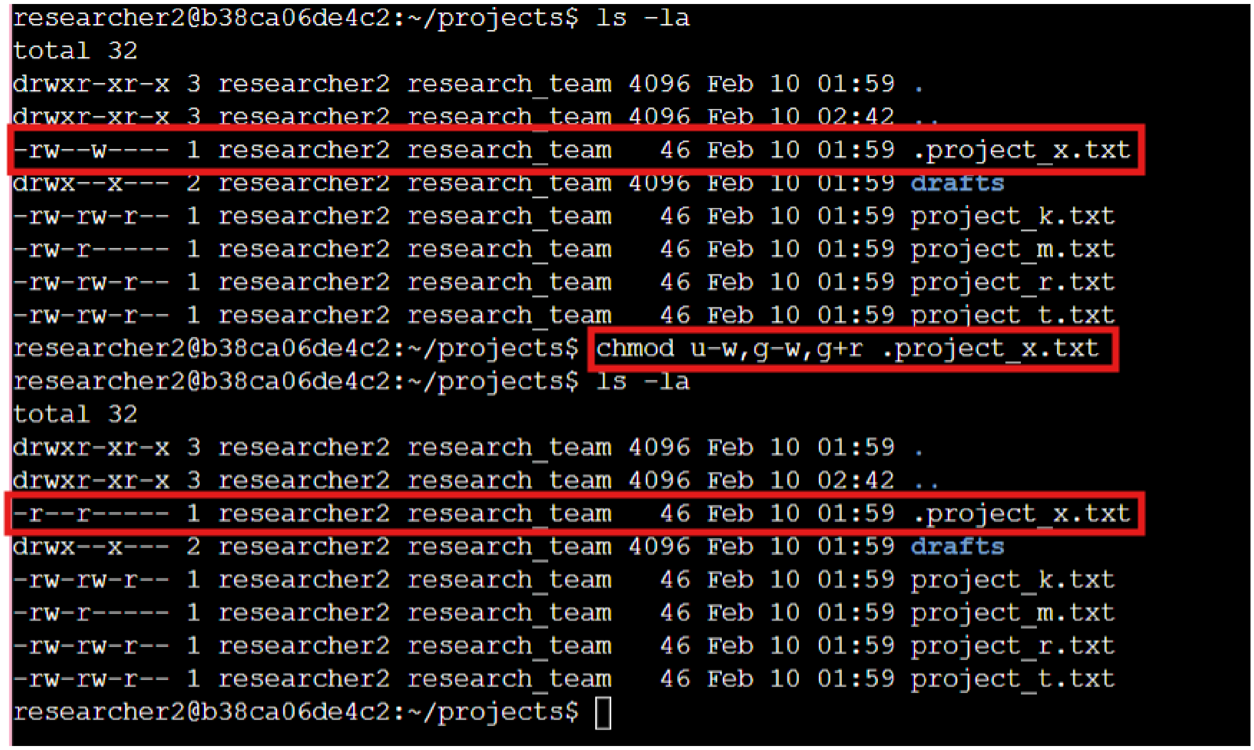


```
researcher2@b38ca06de4c2:~/projects$ ls
drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
researcher2@b38ca06de4c2:~/projects$ chmod o-w project_k.txt
researcher2@b38ca06de4c2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 01:59 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 02:42 ..
-rw--w---- 1 researcher2 research_team   46 Feb 10 01:59 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 10 01:59 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 10 01:59 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_t.txt
researcher2@b38ca06de4c2:~/projects$ 
```

chmod is the function to change modes, the o-w is removing write permissions from the other section (other minus write), and then you need the file/directory that needs changed.

Change file permissions on a hidden file

The research team has archived **.project_x.txt**, which is why it's a hidden file. This file should not have write permissions for anyone, but the user and group should be able to read the file. Use a Linux command to assign **.project_x.txt** the appropriate authorization.



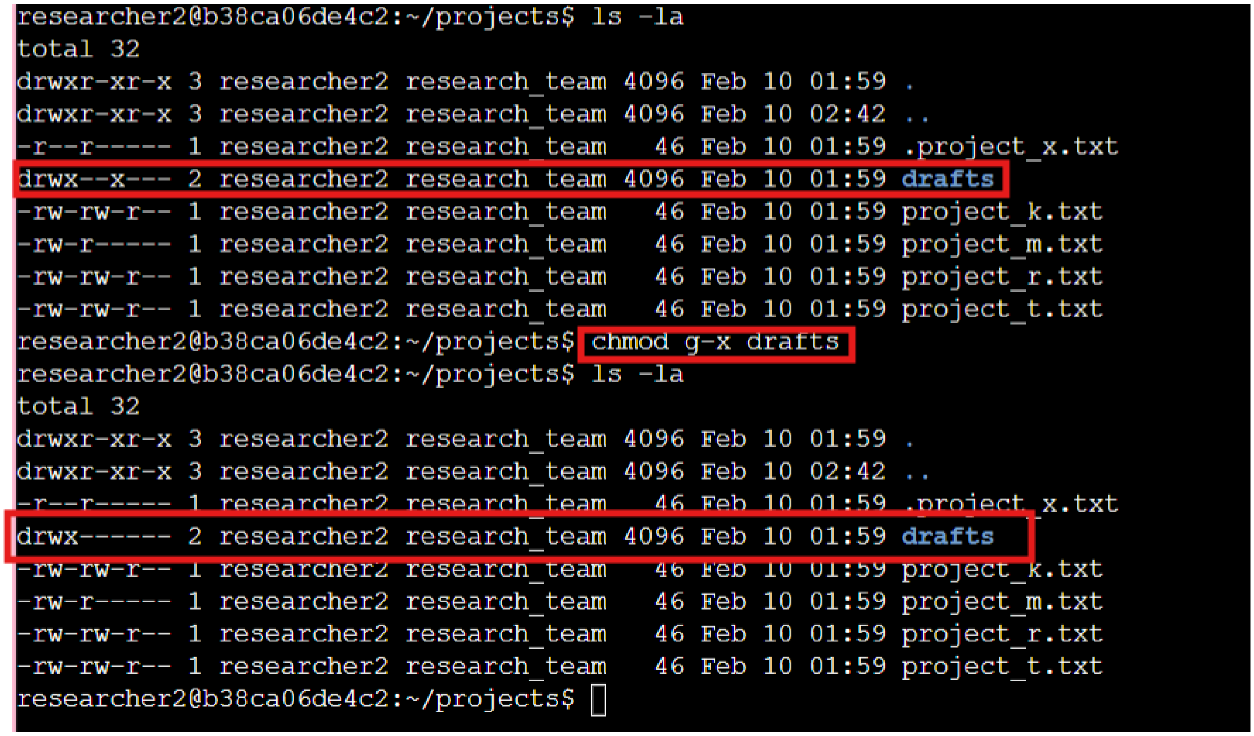
```
researcher2@b38ca06de4c2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 01:59 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 02:42 ..
-rw--w---- 1 researcher2 research_team   46 Feb 10 01:59 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 10 01:59 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 10 01:59 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_t.txt
researcher2@b38ca06de4c2:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@b38ca06de4c2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 01:59 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 02:42 ..
-r--r----- 1 researcher2 research_team   46 Feb 10 01:59 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 10 01:59 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 10 01:59 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_t.txt
researcher2@b38ca06de4c2:~/projects$ 
```

Using ls -la lists all directories and files in the current directory including their permissions and any hidden files/directories.

Once the permissions for the hidden file was found, then we could change the file's permissions via chmod u-w,g-w,g+r .project_x.txt.
Chmod changes modes then u -w removed write from the user (user minus write), g-w removed write from the group (group minus write), and g+r added read to the group (group plus read) then the file/directory that is being edited.

Change directory permissions

The files and directories in the projects directory belong to the **researcher2** user. Only **researcher2** should be allowed to access the **drafts** directory and its contents. Use a Linux command to modify the permissions accordingly.



```
researcher2@b38ca06de4c2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 01:59 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 02:42 ..
-r--r----- 1 researcher2 research_team   46 Feb 10 01:59 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Feb 10 01:59 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 10 01:59 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_t.txt
researcher2@b38ca06de4c2:~/projects$ chmod g-x drafts
researcher2@b38ca06de4c2:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 01:59 .
drwxr-xr-x 3 researcher2 research_team 4096 Feb 10 02:42 ..
-r--r----- 1 researcher2 research_team   46 Feb 10 01:59 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Feb 10 01:59 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Feb 10 01:59 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Feb 10 01:59 project_t.txt
researcher2@b38ca06de4c2:~/projects$ 
```

Ls -la was used to find the current permissions of the draft directory. Chmod g-x drafts were used to change the permissions. Chmod is used to change modes, g-x removed the execute function from the group (group minus execute) and drafts was the directory being edited.

Summary

In this task, I audited the file system of the /home/researcher2/projects directory to align user access with the **Principle of Least Privilege**. By using the chmod command, I neutralized security risks—specifically removing unauthorized write access from "other" users on Project_k.txt and restricting the drafts directory to the owner only. I also reconfigured the permissions on the hidden archive file .project_x.txt to ensure its integrity as a read-only document, effectively reducing the attack surface of the research team's environment.