Activity: Manage authorization

Activity overview

In this lab activity, you'll use Linux commands to configure authorization.

Authorization is the concept of granting access to specific resources in a system. It's important because without authorization any user could access and modify all files belonging to other users or system files. This would certainly be a security risk.

In Linux, file and directory permissions are used to specify who has access to specific files and directories. You'll explore file and directory permissions and change the ownership of a file and a directory to limit who can access them.

As a security analyst, setting appropriate access permissions is critical to protecting sensitive information and maintaining the overall security of a system.

Scenario

In this scenario, you must examine and manage the permissions on the files in the /home/researcher2/projects directory for the researcher2 user.

The researcher2 user is part of the research_team group.

You must 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. When it doesn't, you must change the permissions.

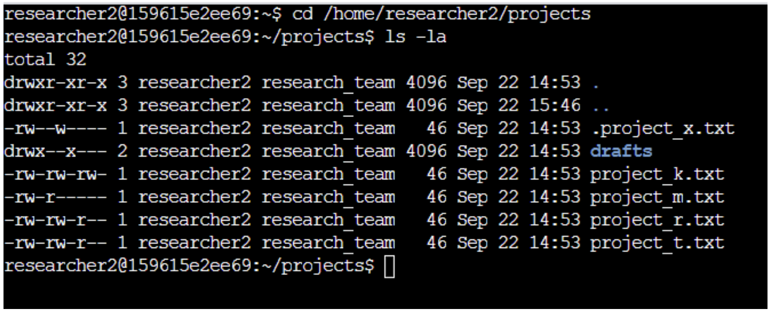
Here's how you'll do this task: **First**, you'll check the user and group permissions for all files in the projects directory. **Next**, you'll check whether any files have incorrect permissions and change the permissions as needed. **Finally**, you'll check the permissions of the /home/researcher2/projects/drafts directory and modify these permissions to remove any unauthorized access.

Task 1. Check file and directory details

In this task, you must explore the permissions of the projects directory and the files it contains. The lab starts with /home/researcher2 as the current working directory. This is because you're changing permissions for files and directories belonging to the researcher2 user.

1. Navigate to the projects directory.
2. List the contents and permissions of the projects directory.

The permissions of the files in the projects directory are as follows:



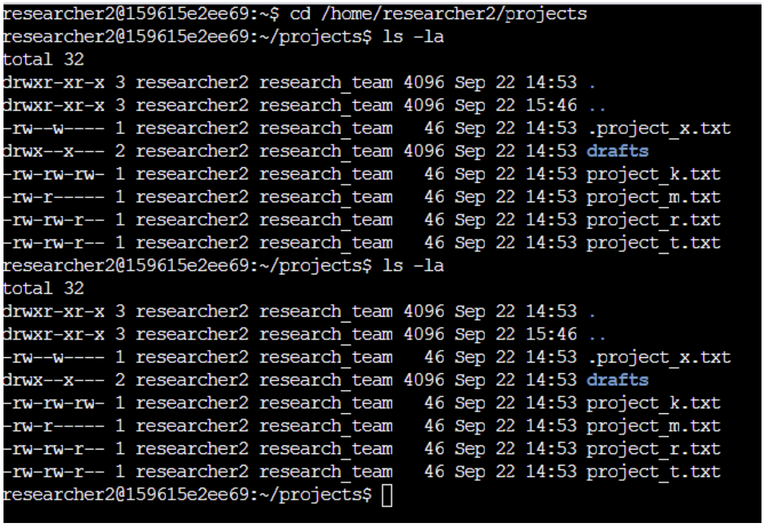
```
researcher2@159615e2ee69:~$ cd /home/researcher2/projects
researcher2@159615e2ee69:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 14:53 .
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 15:46 ..
-rw--w---- 1 researcher2 research_team   46 Sep 22 14:53 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ 
```

As you may recall from the video lesson, a 10-character string begins each entry and indicates how the permissions on the file are set. For instance, a directory with full permissions for all owner types would be drwxrwxrwx:

- The 1st character indicates the file type. The d indicates it's a directory. When this character is a hyphen (-), it's a regular file.
- The 2nd-4th characters indicate the read (r), write (w), and execute (x) permissions for the user. When one of these characters is a hyphen (-) instead, it indicates that this permission is not granted to the user.
- The 5th-7th characters indicate the read (r), write (w), and execute (x) permissions for the group. When one of these characters is a hyphen (-) instead, it indicates that this permission is not granted for the group.
- The 8th-10th characters indicate the read (r), write (w), and execute (x) permissions for the owner type of other. This owner type consists of all other users on the system apart from the user and the group. When one of these characters is a hyphen (-) instead, that indicates that this permission is not granted for other.

The second block of text in the expanded directory listing is the user who owns the file. The third block of text is the group owner of the file.

3. Check whether any hidden files exist in the projects directory.



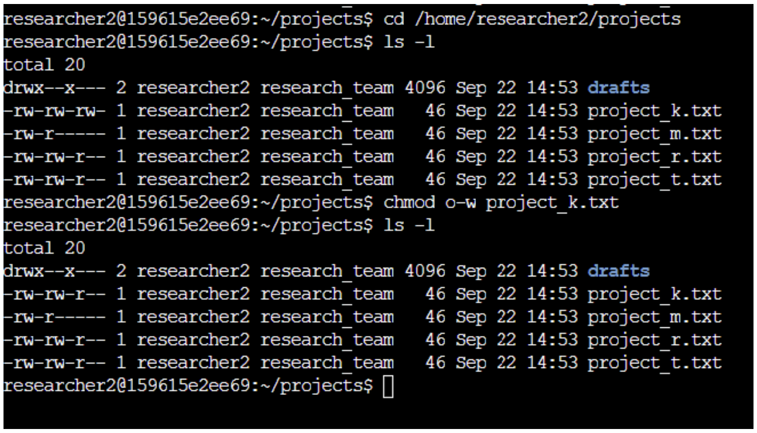
```
researcher2@159615e2ee69:~$ cd /home/researcher2/projects
researcher2@159615e2ee69:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 14:53 .
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 15:46 ..
-rw--w---- 1 researcher2 research_team   46 Sep 22 14:53 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 14:53 .
drwxr-xr-x 3 researcher2 research_team 4096 Sep 22 15:46 ..
-rw--w---- 1 researcher2 research_team   46 Sep 22 14:53 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ 
```

Task 2. Change file permissions

In this task, you must determine whether any files have incorrect permissions and then change the permissions as needed. This action will remove unauthorized access and strengthen security on the system.

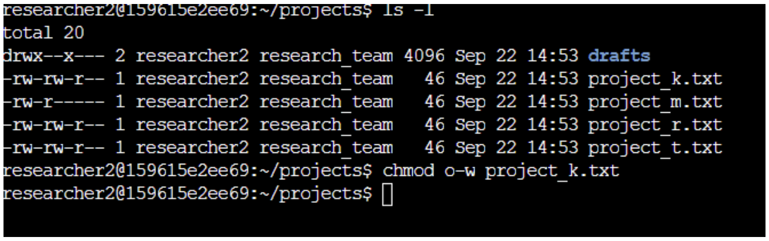
None of the files should allow the other users to write to files.

1. Check whether any files in the projects directory have write permissions for the owner type of other.



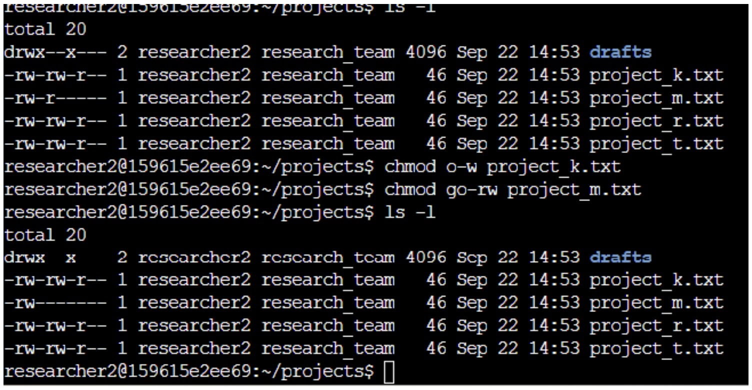
```
researcher2@159615e2ee69:~/projects$ cd /home/researcher2/projects
researcher2@159615e2ee69:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod o-w project_k.txt
researcher2@159615e2ee69:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ 
```

2. Change the permissions of the file identified in the previous step so that the owner type of other doesn’t have write permissions.



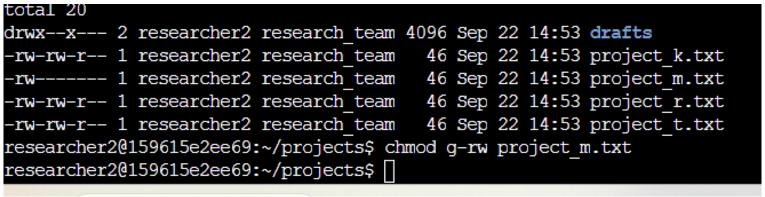
```
researcher2@159615e2ee69:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod o-w project_k.txt
researcher2@159615e2ee69:~/projects$ 
```

3. The file project_m.txt is a restricted file and should not be readable or writable by the group or other; only the user should have these permissions on this file. List the contents and permissions of the current directory and check if the group has read or write permissions.



```
researcher2@159615e2ee69:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod o-w project_k.txt
researcher2@159615e2ee69:~/projects$ chmod go-rw project_m.txt
researcher2@159615e2ee69:~/projects$ ls -l
total 20
drwx  x    2 rcscarchcr2 rcscarch_tcam 4096 Scp 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw------- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ 
```

4. Use the chmod command to change permissions of the project_m.txt file so that the group doesn’t have read or write permissions.



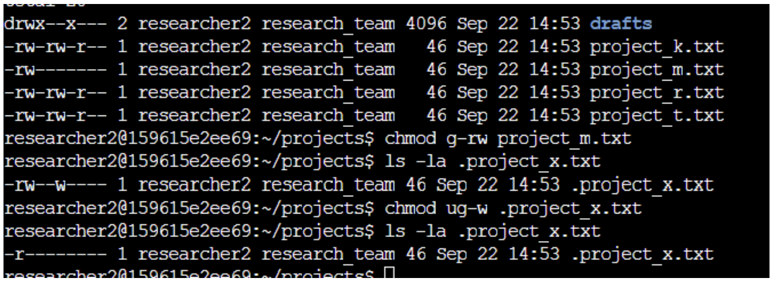
```
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw------- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod g-rw project_m.txt
researcher2@159615e2ee69:~/projects$ 
```

Task 3. Change file permissions on a hidden file

In this task, you must determine if a hidden file has incorrect permissions and then change the permissions as needed. This action will further remove unauthorized access and strengthen security on the system.

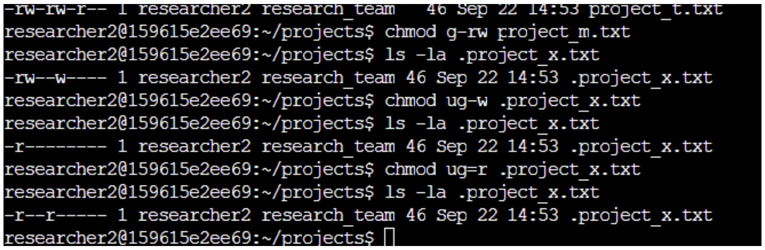
The file .project_x.txt is a hidden file that has been archived and should not be written to by anyone. (The user and group should still be able to read this file.)

1. Check the permissions of the hidden file .project_x.txt and answer the question that follows.



```
total 20
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_k.txt
-rw------- 1 researcher2 research_team   46 Sep 22 14:53 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod g-rw project_m.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-rw--w---- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ chmod ug-w .project_x.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-r-------- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ 
```

2. Change the permissions of the file .project_x.txt so that both the user and the group can read, but not write to, the file.



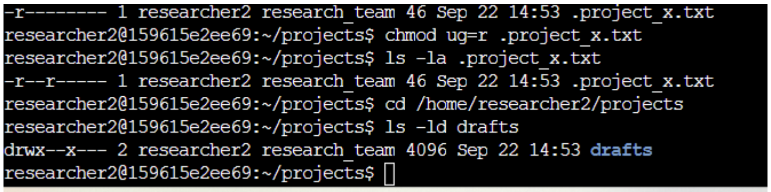
```
-rw-rw-r-- 1 researcher2 research_team   46 Sep 22 14:53 project_t.txt
researcher2@159615e2ee69:~/projects$ chmod g-rw project_m.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-rw--w---- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ chmod ug-w .project_x.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-r-------- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ chmod ug=r .project_x.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-r--r----- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ 
```

Task 4. Change directory permissions

In this task, you must change the permissions of a directory. First, you'll check the group permissions of the /home/researcher2/projects/drafts directory and then modify the permissions as required. (You should be in the projects directory while managing the permissions of its subdirectory drafts.)

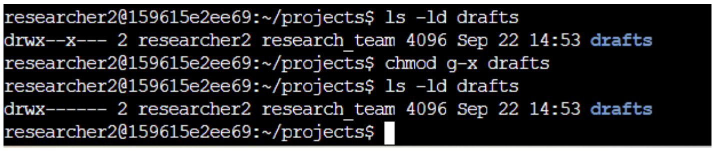
Only the researcher2 user should be allowed to access the drafts directory and its contents. (This means that only researcher2 should have execute privileges.)

1. Check the permissions of the drafts directory and answer the following question.



```
-r-------- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ chmod ug=r .project_x.txt
researcher2@159615e2ee69:~/projects$ ls -la .project_x.txt
-r--r----- 1 researcher2 research_team 46 Sep 22 14:53 .project_x.txt
researcher2@159615e2ee69:~/projects$ cd /home/researcher2/projects
researcher2@159615e2ee69:~/projects$ ls -ld drafts
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
researcher2@159615e2ee69:~/projects$ 
```

2. Remove the execute permission for the group from the drafts directory.



```
researcher2@159615e2ee69:~/projects$ ls -ld drafts
drwx--x--- 2 researcher2 research_team 4096 Sep 22 14:53 drafts
researcher2@159615e2ee69:~/projects$ chmod g-x drafts
researcher2@159615e2ee69:~/projects$ ls -ld drafts
drwx------ 2 researcher2 research_team 4096 Sep 22 14:53 drafts
researcher2@159615e2ee69:~/projects$ 
```

Lab Summary: Configure Authorization with Linux Permissions

Objective

This lab focused on managing file and directory permissions in Linux to enforce proper authorization. By examining existing permissions and applying the correct changes, I ensured that files and directories in the /home/researcher2/projects directory were restricted according to security requirements.

Tasks Completed

Task 1: Check File and Directory Details

- Navigated to /home/researcher2/projects.
- Used ls -la to view all files, including hidden ones, along with their permissions.
- Identified standard files and confirmed the existence of the hidden file .project_x.txt.

Task 2: Change File Permissions

- Verified file permissions with ls -l.
- Identified project_k.txt as incorrectly allowing write permissions to others.
- Removed write access from others using chmod o-w project_k.txt.
- Restricted project_m.txt so that only the user had read and write access by running chmod go-rw project_m.txt.

Task 3: Change File Permissions on a Hidden File

- Examined .project_x.txt and found both the user and group had write permissions, which was not allowed.
- Modified permissions with chmod ug-w .project_x.txt so that both the user and group could only read the file.

Task 4: Change Directory Permissions

- Checked the drafts directory with ls -ld drafts.
- Found that the group and others had execute permissions.
- Removed group execute privileges with chmod g-x drafts.
- Confirmed changes with ls -ld drafts.

Summary

This lab demonstrated how to evaluate and modify Linux file and directory permissions to enforce proper authorization. By removing unauthorized read and write privileges and restricting execute access, I ensured that only the intended user could access sensitive files and directories. These skills are essential for a security analyst to maintain system integrity and protect data from unauthorized access.