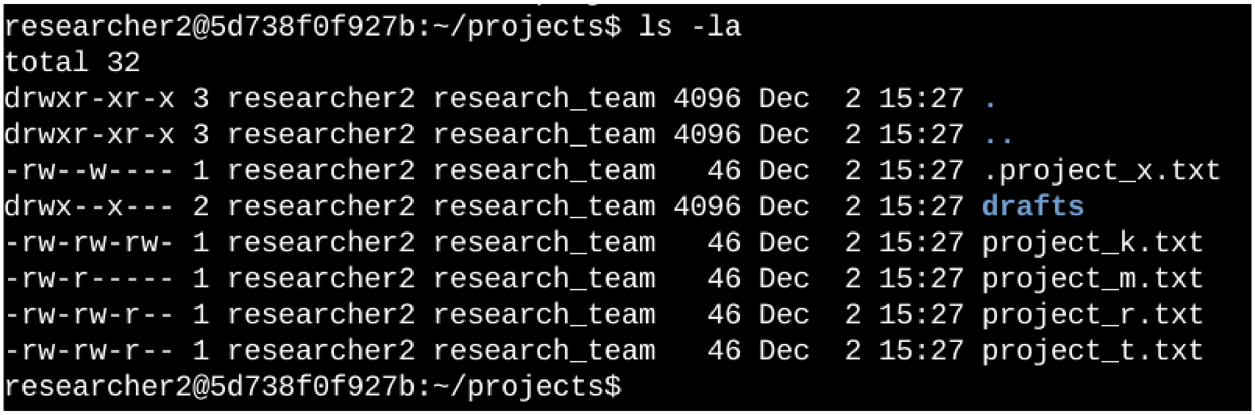
# File permissions in Linux

## Project description

As part of the research team, our job is to ensure that only authorised users with appropriate permissions can access certain files and directories within the project directory. This is to make sure that the system is kept secure.

## Check file and directory details

The following code shows how I used Linux commands to determine the existing permission set for a specific directory in the file system.



```
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 ..
-rw--w---- 1 researcher2 research_team   46 Dec  2 15:27 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec  2 15:27 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec  2 15:27 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec  2 15:27 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_t.txt
researcher2@5d738f0f927b:~/projects$
```

The top line in the screenshot shows the command I executed, while the remaining lines show its output. This command reveals all items within the projects directory. I used ls -la to generate a comprehensive listing that includes hidden files. According to the output, the directory contains one folder named *drafts*, a hidden file called *.project_x.txt*, and five additional project files. The first column, which consists of a 10-character string, indicates the permission settings for each file or folder.

## Describe the permissions string

The 10-character string shows file permissions:

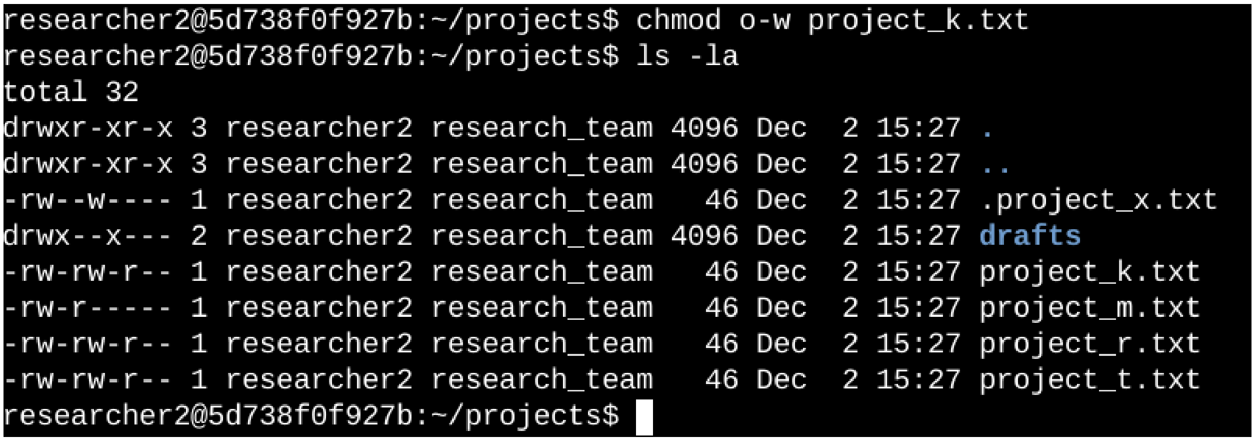
- **1st**: d means directory, - means file
- **2nd–4th**: user permissions (r, w, x)
- **5th–7th**: group permissions
- **8th–10th**: others' permissions

The permissions for project_t.txt are -rw-rw-r--:

- - means it's a file
- r for user, group, and others = read access
- w for user and group = write access
- No x = no execute access for anyone

## Change file permissions

The organization decided that others shouldn't have write access to any files. Based on the permissions I checked earlier, project_k.txt needed that access removed.



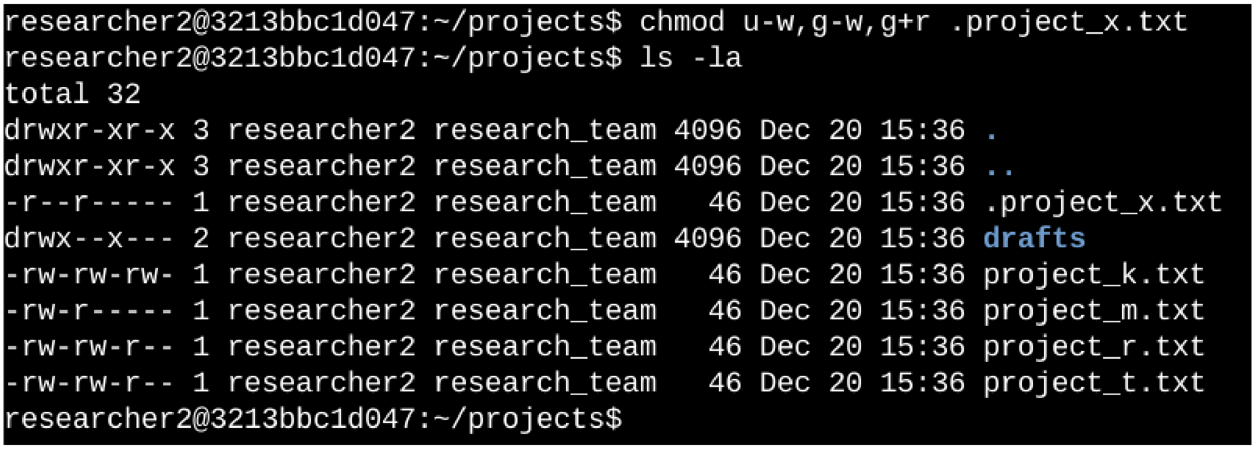
```
researcher2@5d738f0f927b:~/projects$ chmod o-w project_k.txt
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 ..
-rw--w---- 1 researcher2 research_team   46 Dec  2 15:27 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec  2 15:27 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec  2 15:27 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_t.txt
researcher2@5d738f0f927b:~/projects$ 
```

The first two lines in the screenshot show the commands I typed, and the rest show the output from the second one. I used chmod to change file permissions—first to remove write access for others on project_k.txt, then ls -la to confirm the change.

## Change file permissions on a hidden file

The research team archived project_x.txt and wants only read access for the user and group—no write access for anyone.

## Here’s the command I used:



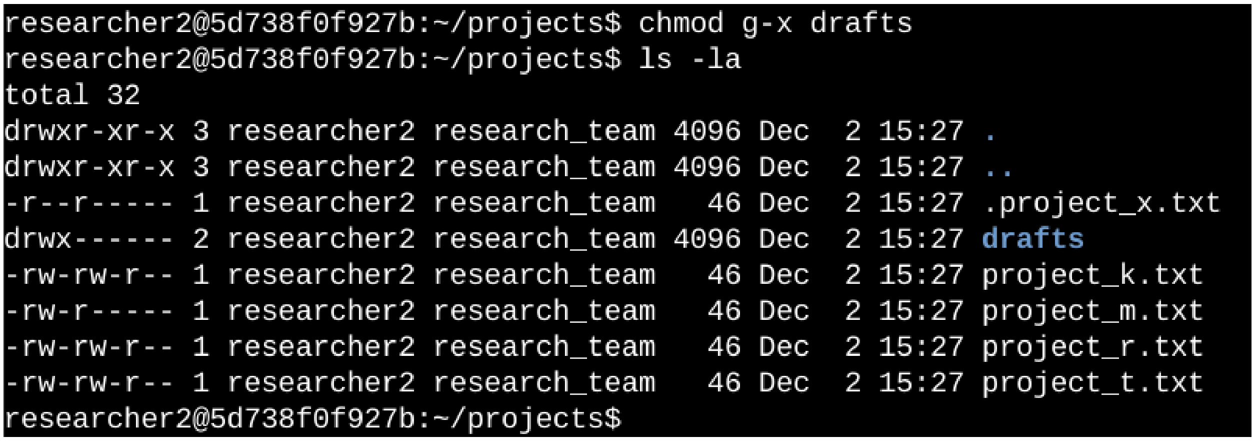
```
researcher2@3213bbc1d047:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@3213bbc1d047:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec 20 15:36 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec 20 15:36 ..
-r--r----- 1 researcher2 research_team   46 Dec 20 15:36 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Dec 20 15:36 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Dec 20 15:36 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec 20 15:36 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 20 15:36 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec 20 15:36 project_t.txt
researcher2@3213bbc1d047:~/projects$
```

The first two lines show the commands I used, and the rest show the output. Since .project_x.txt starts with a dot, it's a hidden file. I removed write access from the user and group using u-w and g-w, then gave the group read access with g+r.

## Change directory permissions

To ensure only researcher2 has access to the drafts directory, I removed execute permissions for everyone else.

Here’s the command I used:



```
researcher2@5d738f0f927b:~/projects$ chmod g-x drafts
researcher2@5d738f0f927b:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 .
drwxr-xr-x 3 researcher2 research_team 4096 Dec  2 15:27 ..
-r--r----- 1 researcher2 research_team   46 Dec  2 15:27 .project_x.txt
drwx------ 2 researcher2 research_team 4096 Dec  2 15:27 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Dec  2 15:27 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Dec  2 15:27 project_t.txt
researcher2@5d738f0f927b:~/projects$
```

The output shows permission details for files and folders:

- Line 1: current directory (projects)
- Line 2: parent directory (home)
- Line 3: hidden file .project_x.txt
- Line 4: drafts directory with limited access—only researcher2 has execute permission

Since the group had execute access before, I used chmod to remove it. researcher2 already had the needed permission, so no changes were made for that user.

## Summary

I updated several permissions to match my organization’s access requirements for the projects directory. First, I ran ls -la to review current settings. Based on that, I used chmod multiple times to adjust permissions on specific files and folders.