File Permissions in Linux

Project description
The research team requested that file and directory permissions inside the projects directory be updated to match their required security guidelines. Some files allowed broader access than necessary, which posed a security risk. I reviewed existing permissions, interpreted the permission strings, and used chmod commands to enforce the proper access levels.

Check file and directory details
I navigated to the projects directory and used **ls -lA** to show detailed permission information for all files, including hidden items. This allowed me to confirm which files and folders had incorrect authorization settings. The following screenshot shows the command and its output:

```
cd projects/
ls -lA

drwxr-x---  researcher2  research   4096 drafts
-rw-rw-r--  researcher1  research   1200 project_a.txt
-rw-rw-rw-  researcher1  research   980  project_k.txt
-rw-r-----  researcher1  research   500  .project_x.txt
```

This output shows one directory named drafts, a hidden file named **.project_x.txt**, and several project files. The permission strings help identify who can read, write, or execute each item. For example, **-rw-rw-rw-** shows that user, group, and others all have write access—something the organization wants to restrict.

Change file permissions
The organization stated that no file should allow write access for 'other'. The file **project_k.txt** incorrectly allowed write permissions for other users. I removed this permission using chmod. The updated permissions confirm the change:

```
chmod o-w project_k.txt
ls -l project_k.txt

-rw-rw-r-- researcher1 research 980 project_k.txt
```

Change permissions on a hidden file
The file **.project_x.txt** contains archived research data and should not be modified by anyone. The user and group should only have read access. I used separate chmod commands to remove write permissions from both the user and group, then explicitly added read access to the group.

```
chmod u-w .project_x.txt
chmod g-w .project_x.txt
chmod g+r .project_x.txt
ls -l .project_x.txt

-r--r----- researcher1 research 500 .project_x.txt
```

Change directory permissions
Only **researcher2** should be able to access the drafts directory. This requires removing execute permissions from both group and other users. Execute permission on directories allows entering and navigating the directory, so removing it effectively blocks access.

```
chmod go-x drafts/
ls -ld drafts/

drwxr----- researcher2 research drafts
```

Summary
To secure the projects directory, I reviewed permissions with ls -lA and modified access using chmod. These changes ensured that users, groups, and others only have the access required for their roles. Properly configuring file permissions protects sensitive content and reduces the risk of accidental or unauthorized modification.