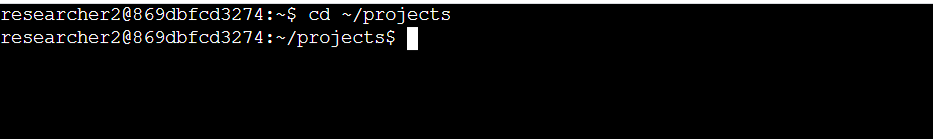
# File permissions in Linux

## Project description

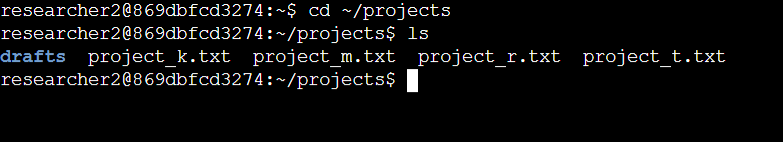
The purpose of this project is to assess the permissions of different users to files contained in the /home/researcher2/projects directory and determine if the permissions match the authorisation that should be given. The goal is to keep the system safe by ensuring that users on the team are authorised with the appropriate permissions. Modifications will be made to non-compliant resources using Linux commands.

## Check file and directory details

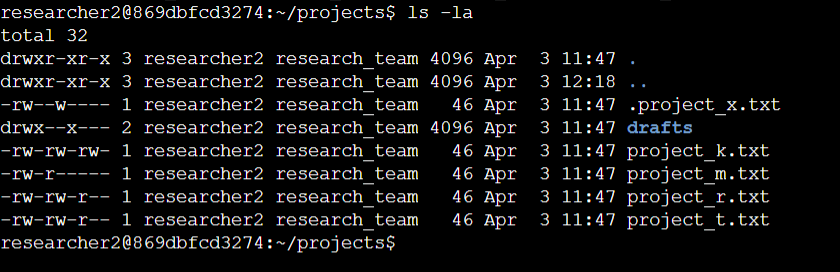
1. To change to the projects directory I used the command: cd ~/projects



1. To list the files in the directory I used the command : ls



1. To list all files along with hidden files and their current permissions, I used the command: ls -la



This command shows that the files and the current users and owners. A 10 character string is used to show the permissions assigned to the different types of users who can have access to a file or a directory. (drwxrwxrwx).

The outcome of the ls -la command showed that the files in the projects directory are owned by the research team. It also shows that 3 files are hidden from public view.

## Describe the permissions string

The 10 character permissions string begins each file entry showing the current permissions on the file or directory.

The first string (d) represents the file type. A (d) string indicates its a directory. When a hyphen (-) is found at the beginning of a file it identifies a file isntead of a dir.

The (r) means the user has read permissions. The (w) represents write permissions whereas the (x) represents execute permissions. This means that the user can have access to the files and run them if they are executables.

When any of the permissions is replaced by a hyphen this means that they user or group lacks a certain permission/ right of access.

The 2nd to 4th strings (rwx) show the permissions of the file owner type - user

The 5th to 7th strings (rwx) show the permissions belonging to the file owner type - group

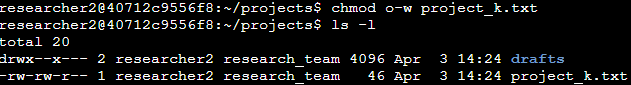
The 8th to 10th strings (rwx) show the permissions belonging to the file owner - other

## Change file permissions

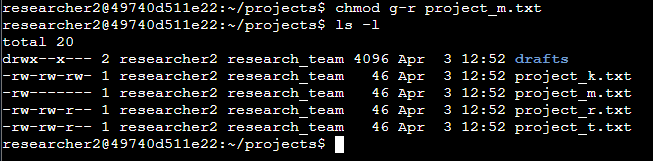
For security reasons the organisation demands that none of the files in the dirrctory should allow the “*other*” user to write access. Seeing that the project\_k file allows the owner type ***other*** to write to the file, the permission must be changed.



To do this, I ran the command chmod o-w project\_k.txt which removed the write permissions from the owner type ***other****.*

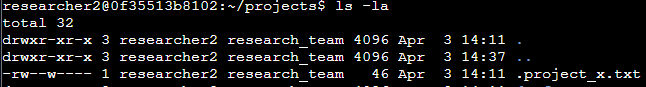
****

The project\_m.txt file which has the current permission -rw-r----- is also a restricted file and should not be read or written to by either the group or other. Only the user should have the ability to write or read the file. To change the permission for the file I ran the command: chmod g-r project\_m.txt which changed the permissions on the file.



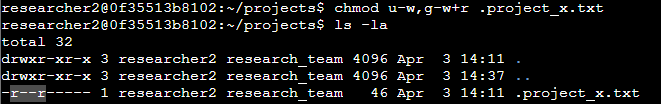
## Change file permissions on a hidden file

The hidden file .project\_x.txt has been archive and should not be written to by anyone. However, the ussr and the group should still be able to read the file. First, the ls -la command was run to show the current permissions on the hidden file and this was the standard output.



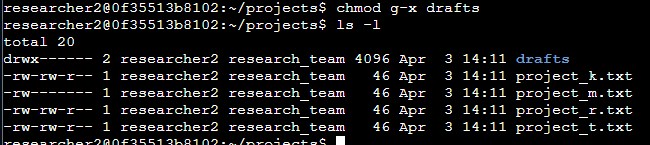
To remove the write permissions from the user and group owners while reservhing their read permissions, I ran the following command: chmod u-w,g-w,g+r .project\_x.txt

but I also realised that the symbols can be combined together to give still the same output as with the following command: chmod u-w,g-w+r .project\_x\_txt . This command removed the write permissions from both the user and the group while reserving their read access.



## Change directory permissions

In this final task, I was directed to change the permissions of the drafts directory to only allow access to the researcher2 user. This user should be the only one having the execute privileges (meaning to access the files in the dir and its contents). To change permissions on the directory I ran the command: chmod g-x drafts. This command removed the current exe permissions of the group allowing the user the sole access to the dir and its contents.



## Summary

This projects demonstrates using Linux commands to access file permissions and modify them to reflect my organisation’s security policy. This is a defense in depth strategy ensuring that employees are not granted permissions beyond what they need to carry out their tasks successfully.

I went through the process of assessing current permissions, determining permissions that need modification and using the command line to change those permissions to reduce the threat surface and bolster my company’s security posture.