**Scenario:**

*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.*

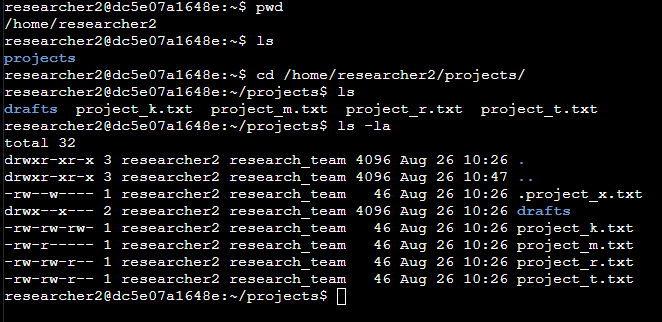
*Your task is to examine existing permissions on the file system. You’ll need to determine if the permissions match the authorization that should be given. If they do not match, you’ll need to modify the permissions to authorize the appropriate users and remove any unauthorized access.*

Use the **pwd** command to identify the current directory.

Use the **ls** command to list the subdirectories in the current directory.

Next type **cd /home/researcher2/projects** to change the current directory to the projects directory.

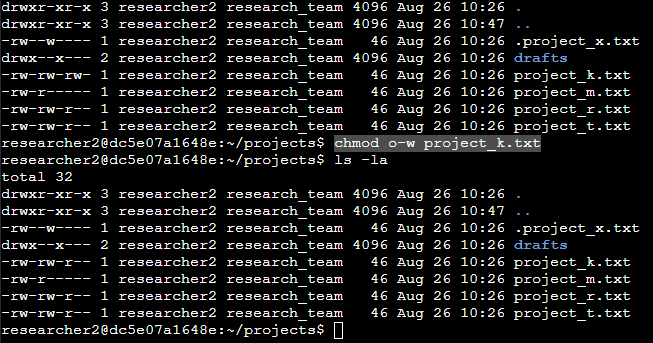
Typing the **ls** command lists the names of files in this directory, using the **-la** lists all files including hidden ones and shows the associated permissions. Hidden files start with “.” at the beginning of the file name.



The beginning of each string has 10 places marked with symbols or letters. This represents the permissions associated with that file or directory. In the first line it begins with “**drwxr-xr-x**”. The first character is the directory position, if this is a directory the position is marked with a “**d**”, if it is a file the first position is marked with “**-**“ The next 3 positions are associated with the user. The following 3 positions are associated with the group. The final 3 positions are associated with all others or simply other. The permissions allowed are: read (denoted by the letter **r**), write (denoted by the letter **w**) and execute (denoted by the letter **x**), if a permission is not allowed a “**-**“ is placed in the associated place.

The organization does not allow other to have write access to any files.

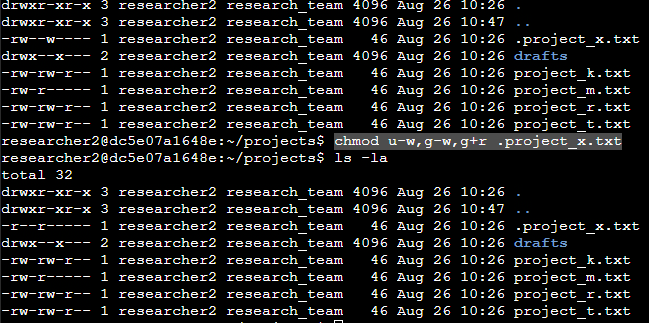
Using **chmod** allows you to change the permissions, in this example the “write” permission needs to be removed from the file **project\_k.txt**.



To remove a permission using the **chmod** command you have to identify which permission is to be changed. This is done by identifying the position, other in this case. By typing **o-w you are telling the system to remove the write permission from the group other concerning file project\_k.txt.**

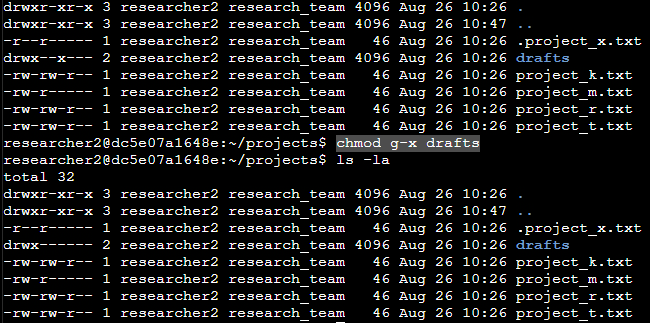
The research team has archived **.project\_x.txt**, which is why it’s a hidden file. This file should not have write permissions for anyone, but the user and group should be able to read the file.

**Using the command chmod u-w,g-w,g+r .project\_x.txt you are telling the system to remove the write permission from the user and group as well as adding the read permission to the group in one combined command for file .project\_x.txt.**



After making the changes type **ls -la** to ensure your changes have been made correctly.

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.



This permission change is done by typing **chmod g-x drafts**. This removes the execute permission from the group leaving the user with the only permissions to the **drafts** directory.

**Summary**

A few simple commands allows you to see all directories and files, to include hidden or archived files and the permissions associated with each. From there you can change permissions as necessary to ensure proper permissions are given to the correct people or groups. Always check after any change to verify that the changes were done correctly.