# File permissions in Linux

## Project description

What I am accomplishing with the use of the Linux commands is ensuring users on the research team are authorized with appropriate permissions in order to keep the system secure by examining existing permissions on the file system to determine if the permissions match the authorization that should be given. Otherwise, I need to modify the permissions in order to authorize the appropriate users as well as removing any unauthorized access.

## Check file and directory details

## 

The ls -la command displays all the permissions to the files (project\_k.txt, project\_m.txt, project\_r.txt, project\_t.txt) and one directory (drafts) along with a hidden file (.project\_x.txt) which the hidden files always start with a period at the beginning. If I used the ls -l command instead, then it would have shown only the files and directories, and then using only the ls -a command only displays the hidden files. So when I merge l and a together after the minus sign as we display the whole contents, then we get all the files, whether regular or hidden, and directories together, but not being displayed separately like with the minus a or l with the ls command after it.

## Describe the permissions string

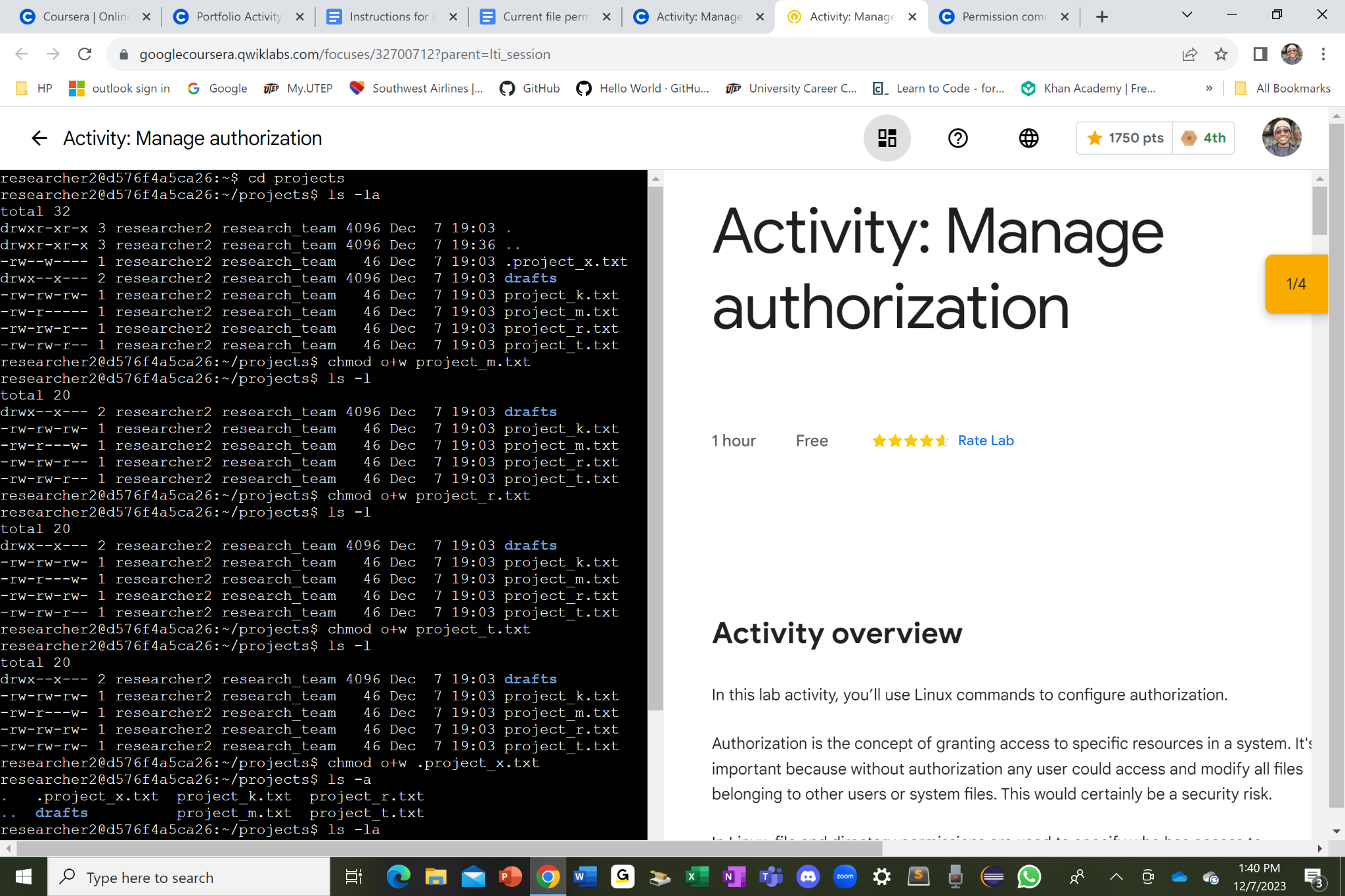
The -rw–-w---- 10 character permission string from the .project\_x.txt, for instance, starts with a hyphen as a first character indicating that it is a file, but if the first character was a d instead, then it would have been a directory.

The second to fourth characters represents a user with the permission to read file contents on the second character placement and with the w character string on the third character placement to make modifications on the file contents and with a hyphen on the fourth placement character indicating that the user does not have the ability to execute the file.

The fifth to seventh characters represent a group that the user is a part of which they only have the ability to make modifications to the file since there is only a w character string on the sixth character placement but the fifth and seventh character placement only have hyphens which indicates that the group cannot read the file contents with the hyphen on the fifth character placement as well as not being able to execute the file with the hyphen on the seventh character placement.

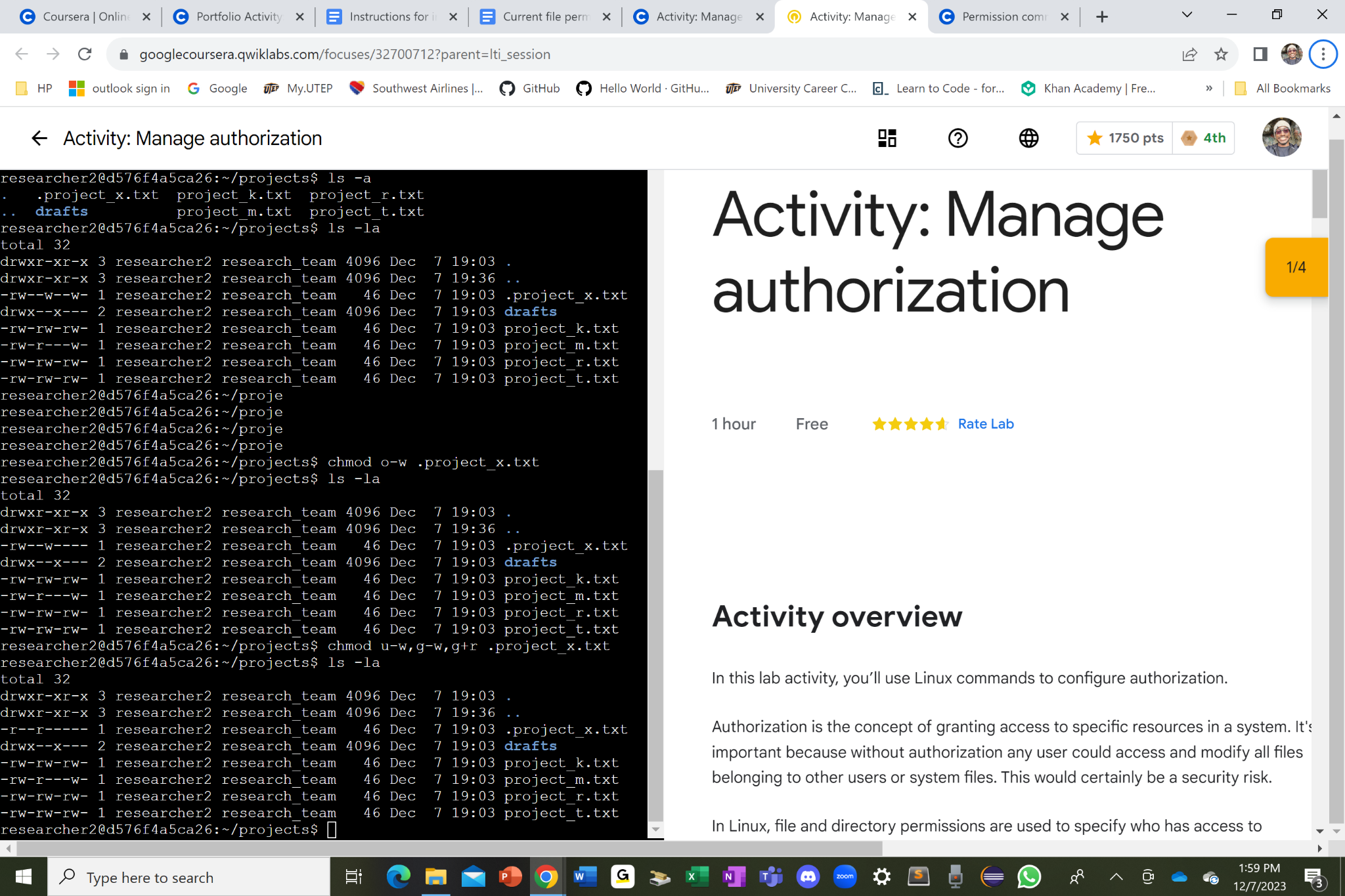
The eighth to tenth characters represent other users on the system who cannot even read the file contents, make modifications on the file contents, or execute the file because the eighth, ninth, and tenth character placement all contain hyphens on them.

## Change file permissions



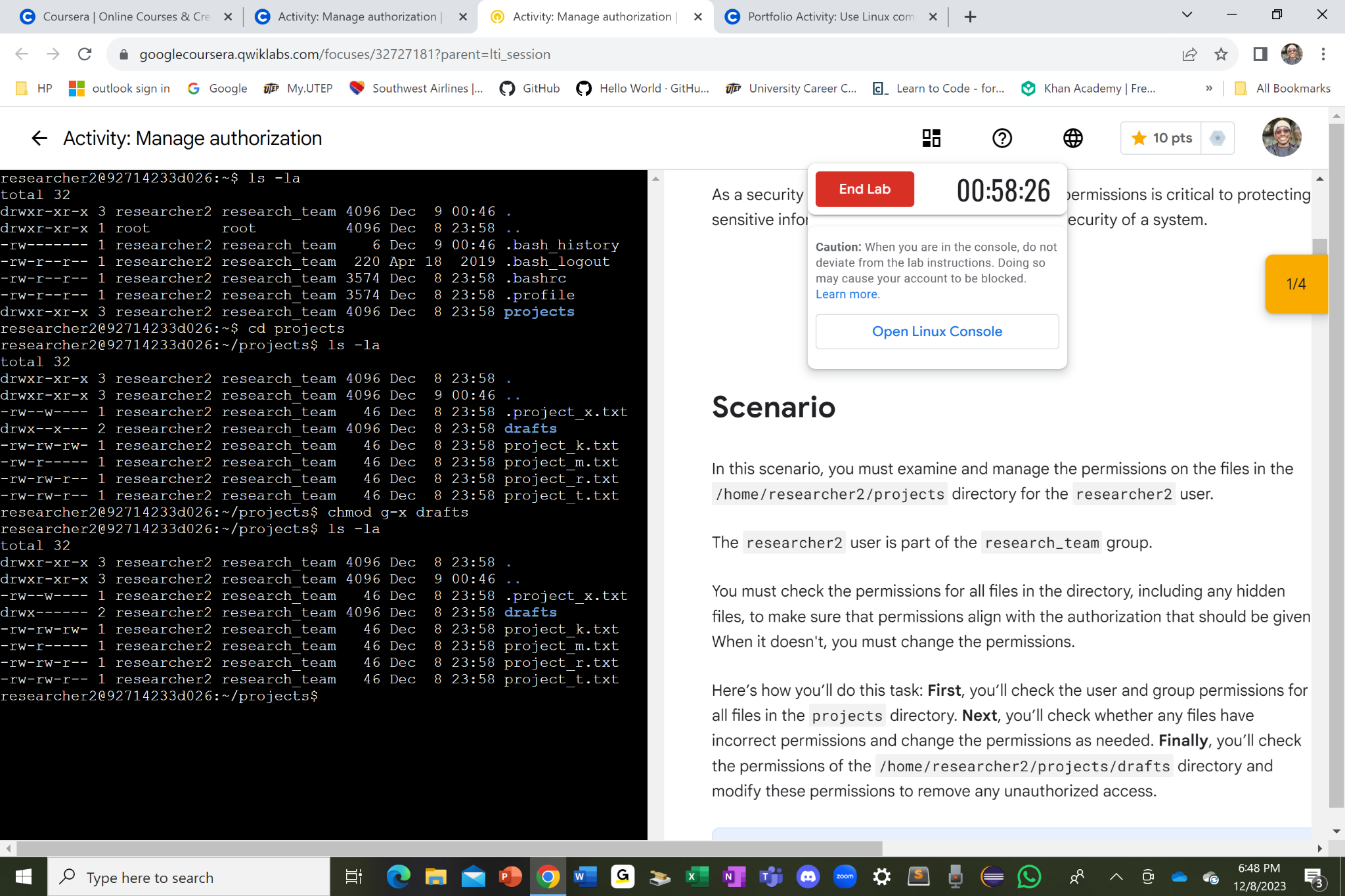
I used the chmod o+w command for the three files, which are not hidden files, to change the permissions in the other sections of the 10 character string, referring to the last three character placement of the string, which the chmod command changes permissions on files and directories, and the o after the chmod command represents the other users who needed to have write access to the three files, and then I plus the o with the w , which the w represents the write command to have permissions to make modifications to the three files.

## Change file permissions on a hidden file



Since the instruction says that the .project\_x.txt should not have write permissions but only the user and group should be able to read the file, I subtracted the w command from the user and group with chmod u-w and chmod g-w commands and then I added the r command to g with the chmod g+r command. But the user already has the r command so I don’t need to add in the r. I can either put the three commands (u-w, g-w,g+r) with the name of the file of each and compile them separately or put them together like this chmod u-w,g-w,g+r .project\_x.txt with no spaces in between the three commands to run the program which should display only the r commands in the user and group section of the 10 character string of the .project\_x.txt file.

## Change directory permissions



Since the instruction says that only researcher2 should be allowed to access the drafts directory and its contents, I decided to remove the x command from the group section by minusing the x from the g which is the seventh character placement of the 10 character string of the drafts directory. Hence why, I put this command chmod g-x drafts just for the researcher2 to only have permissions to access the drafts, which is the user, but not the research\_team, which is the group. Therefore, the user should have access to the drafts but not the group.

## Summary

I successfully made all the appropriate changes to the three regular files, one hidden file, and a directory so that the users can have the necessary proper permissions to either read, write, or execute any files according to their status of permissions with the “r”, “w”, “o”, and hyphens on the 10 character string for each files and directories. As a result, the system is more secure than it ever was better.