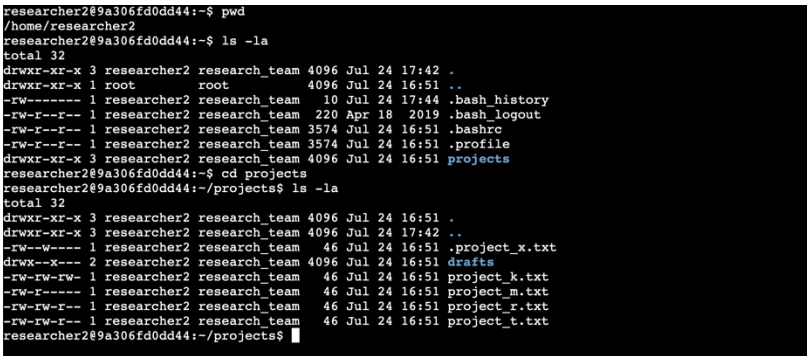
File Permissions in Linux

Project description:

I am a security professional at a large organization mainly working with a research team. Part of my job is to ensure users on this team are authorized with the appropriate permissions.
During this task, I will examine existing permissions on the file system. I will need to determine if the permissions match the authorization that should be given. If they do not match, I will need to modify the permissions to authorize the appropriate users and remove any unauthorized access.

Check file and directory details:

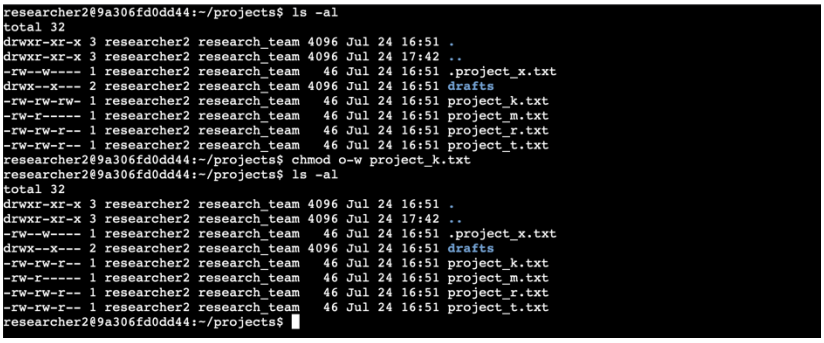


Describe the permissions string:

The Linux permission string is a ten-character string. The first character indicates what type of file it is, `d` is for directory. If it was a file, then it would have a `-` instead of a `d`. The next set of three digits represent the file permissions for the user. So, the second digit in the string would represent if the user had readable permissions. If they do, then there would be a `r`, in the second digit position, if not then there would be a `-` to represent no permissions given. The third digit in the string represents if the user has writable permissions. If they do, then there would be a `w`, in the third digit position, if not then there would be a `-`. The fourth digit in the string represents if the user has executable permissions. If they do, then there would be a `x`, in the third digit position, if not then there would be a `-`. The next set of three digits or the values in 5th-7th digit positions in the 10-digit Linux permissions string represent the group permissions for the file or directory and repeats the same notation from the user permissions. The final three digits or the values in the 8th-10th digit position in the 10-digit Linux permissions string represents the other permissions for the file or directory and repeats the same notation from the user permissions.

Drafts directory has the following Linux permission string – `drwx--x---`
For example, this file is a directory with the user having read, write, & execute permissions, the group only has execute permissions, and other has not permissions.

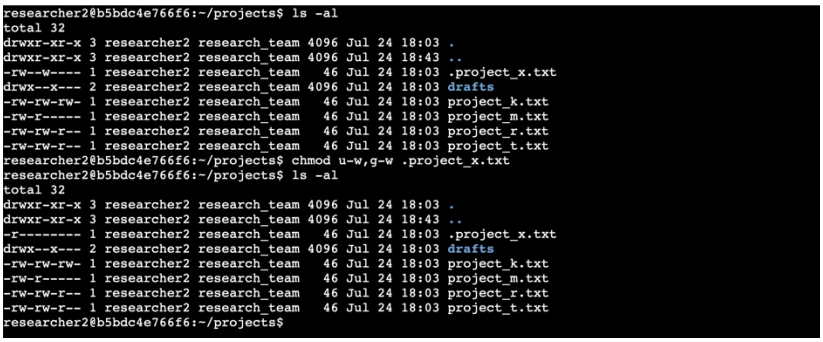
Change file permissions:



To change permissions, you use the `chmod` Linux CLI command. The first argument is the changes you want to make. The second argument is the file or directory you want to make the permission changes too.

For instance, in the screen shot above, I removed the writable permissions (`o-w`) with the first argument and identified the file I wanted to make the changed to (`project_k.txt`) with the second argument.

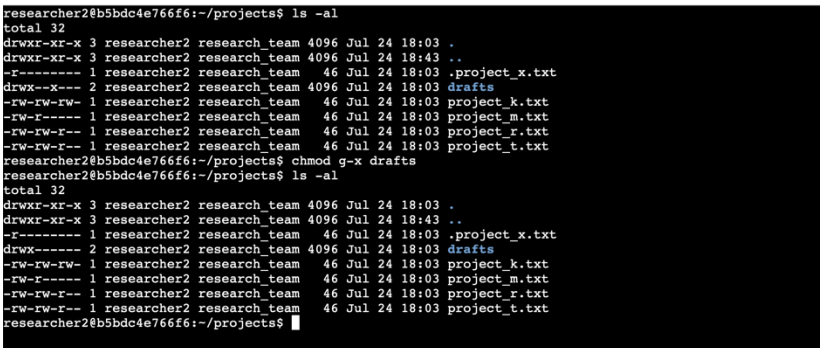
Change file permissions on a hidden file:



To change permissions, you use the `chmod` Linux CLI command. The first argument is the changes you want to make. The second argument is the file or directory you want to make the permission changes too.

For instance, in the screen shot above, I removed the writable permissions (`u-w,g-w`) with the first argument and identified the file I wanted to make the changed to (`.project_x.txt`) with the second argument. Just remember, hidden files have a `.` before the file name to indicate they are a hidden file.

Change directory permissions:



To change permissions, you use the `chmod` Linux CLI command. The first argument is the changes you want to make. The second argument is the file or directory you want to make the permission changes too.

For instance, in the screen shot above, I removed the executable permissions (`u-x`) with the first argument and identified the directory I wanted to make the change to (`drafts`) with the second argument.

Summary:

I work directly with a research team managing the permissions for files on the company server. During this activity, I updated permissions for specific user groups on specific files, including a directory and hidden file. I even provided a breakdown of what I did and how I did it so non-cybersecurity professionals can understand.