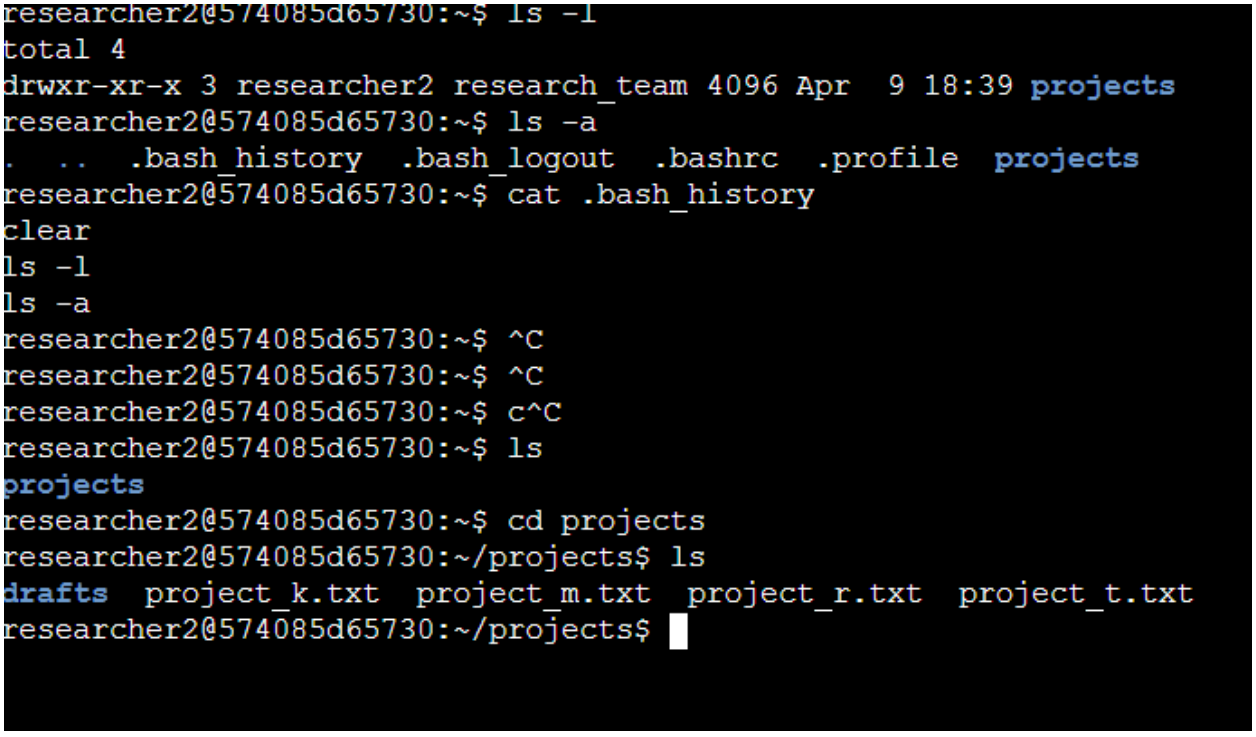
File permissions in Linux

Project description

The research department wants to ensure that every user has the correct permissions set for specific files and directories. Regularly checking to ensure users have correct permissions set can harden our network. Using a virtual machine to locate the files within a directory. Searched the files and the directory to ensure that all groups have the correct permissions set. Removed and granted the correct permissions to files and directories based on access requirements.

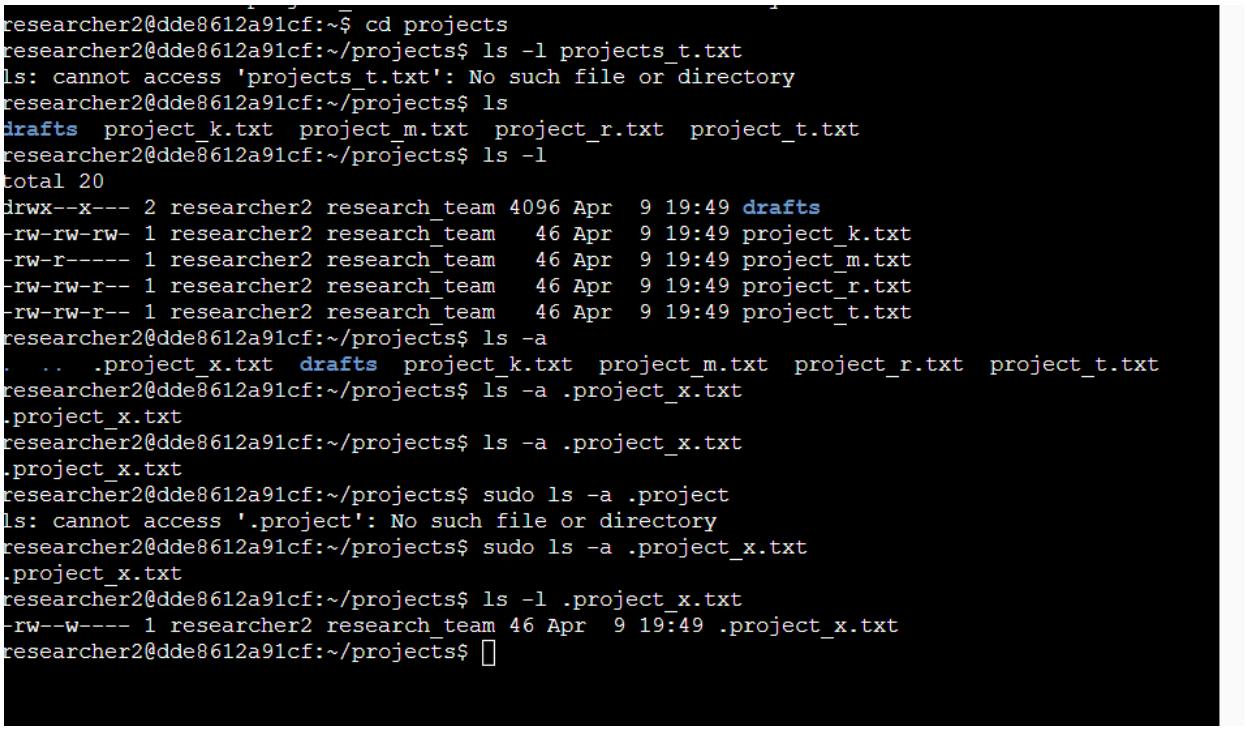
Check file and directory details

I used the command `ls -l` to display the total number of files in the project directory with permissions. I used `ls -a` to display any hidden files permissions. There was a total of four files and one hidden file. After using `cd projects` to ensure I was working in the correct directory, I used `ls` to display all files within the directory. The commands `ls -l project_k.txt`, `ls -l project_m.txt`, `ls -l project_r.txt`, and `ls -l project_t.txt` were run to list the permissions set for each specific file for the user researcher2. To see the permissions set for the hidden file, the command `ls -l .projects_x.txt` was entered in the terminal.



Describe the permissions string

The project_k.txt file showed that all users had the read and write permissions -rw-rw-rw-. The 10-character string represents the permissions set for each of the three groups: user, group, and other. The project_m.txt file showed that the user had read and write permissions, the group had read permissions only, and the others had no access to this file with hyphens in their permission section. The files project_r.txt and project_t.txt showed the user and group permissions were set for read and write, while the others had only read permissions. The hidden file .project_x.txt had permissions set as read and write for the user, write permissions were set for the group, and no permissions were given to the other.



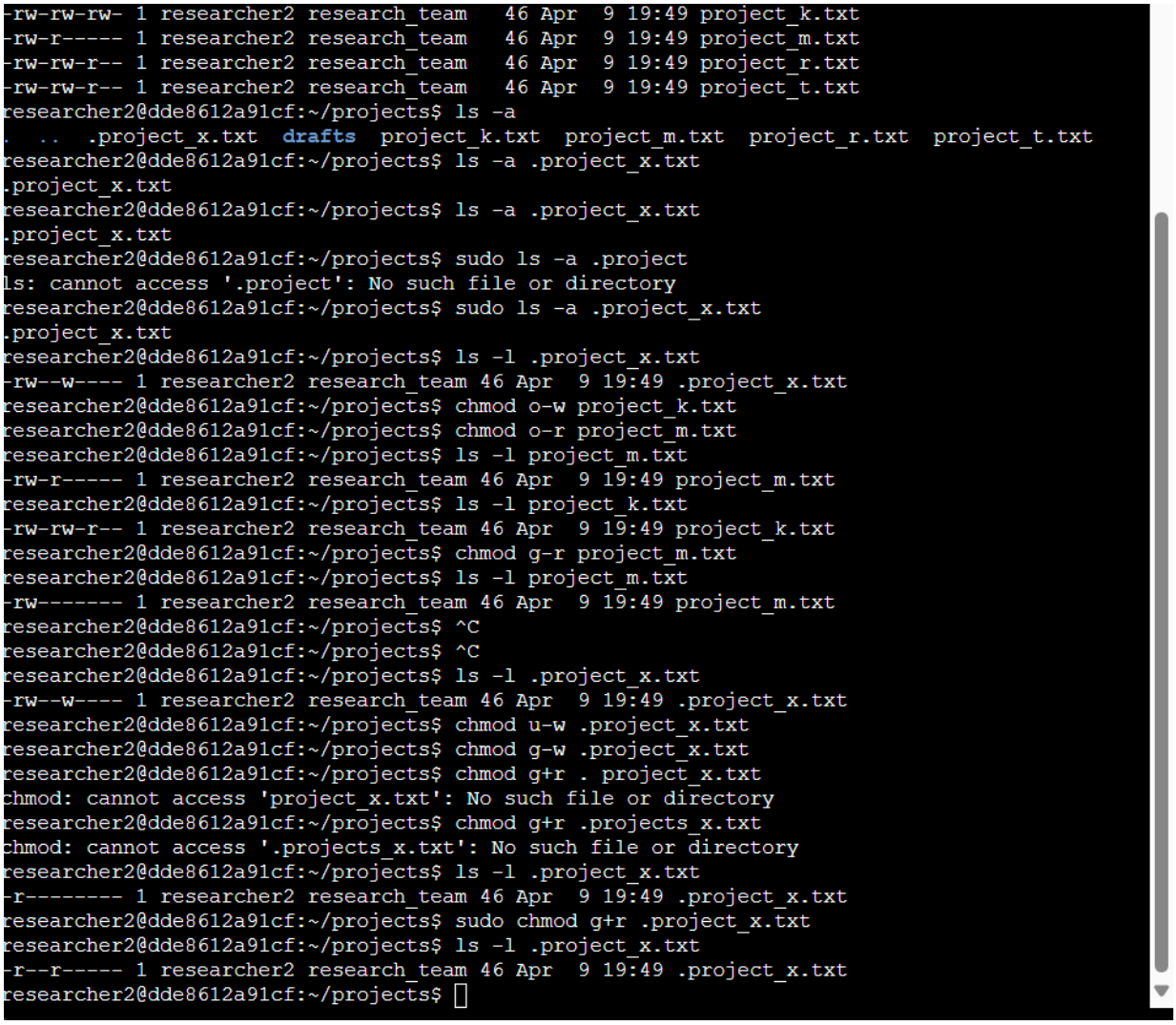
Change file permissions

While checking the permissions of the file project_k.txt, the other had write permission, and only the user should have write permissions. The command `chmod o-w project_k.txt` was run to remove the write permission on the file for the other. File project_m.txt has permissions set to read for the group. This document is a restricted file, and only the user should have access. The command `chmod g-r project_m.txt` was used to secure the document, allowing only the user with permissions.

```
researcher2@dde8612a91cf:~$ ls -l projects_t.txt
ls: cannot access 'projects_t.txt': No such file or directory
researcher2@dde8612a91cf:~$ ls -l project_t.txt
ls: cannot access 'project_t.txt': No such file or directory
researcher2@dde8612a91cf:~$ cd projects
researcher2@dde8612a91cf:~/projects$ ls -l projects_t.txt
ls: cannot access 'projects_t.txt': No such file or directory
researcher2@dde8612a91cf:~/projects$ ls
drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
researcher2@dde8612a91cf:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Apr  9 19:49 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Apr  9 19:49 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Apr  9 19:49 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr  9 19:49 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Apr  9 19:49 project_t.txt
researcher2@dde8612a91cf:~/projects$ ls -a
.  ..  .project_x.txt  drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
researcher2@dde8612a91cf:~/projects$ ls -a .project_x.txt
.project_x.txt
researcher2@dde8612a91cf:~/projects$ ls -a .project_x.txt
.project_x.txt
researcher2@dde8612a91cf:~/projects$ sudo ls -a .project
ls: cannot access '.project': No such file or directory
researcher2@dde8612a91cf:~/projects$ sudo ls -a .project_x.txt
.project_x.txt
researcher2@dde8612a91cf:~/projects$ ls -l .project_x.txt
-rw--w---- 1 researcher2 research_team 46 Apr  9 19:49 .project_x.txt
researcher2@dde8612a91cf:~/projects$ chmod o-w project_k.txt
researcher2@dde8612a91cf:~/projects$ chmod o-r project_m.txt
researcher2@dde8612a91cf:~/projects$ ls -l project_m.txt
-rw-r----- 1 researcher2 research_team 46 Apr  9 19:49 project_m.txt
researcher2@dde8612a91cf:~/projects$ ls -l project_k.txt
-rw-rw-r-- 1 researcher2 research_team 46 Apr  9 19:49 project_k.txt
researcher2@dde8612a91cf:~/projects$ chmod g-r project_m.txt
researcher2@dde8612a91cf:~/projects$ ls -l project_m.txt
-rw------- 1 researcher2 research_team 46 Apr  9 19:49 project_m.txt
researcher2@dde8612a91cf:~/projects$
```

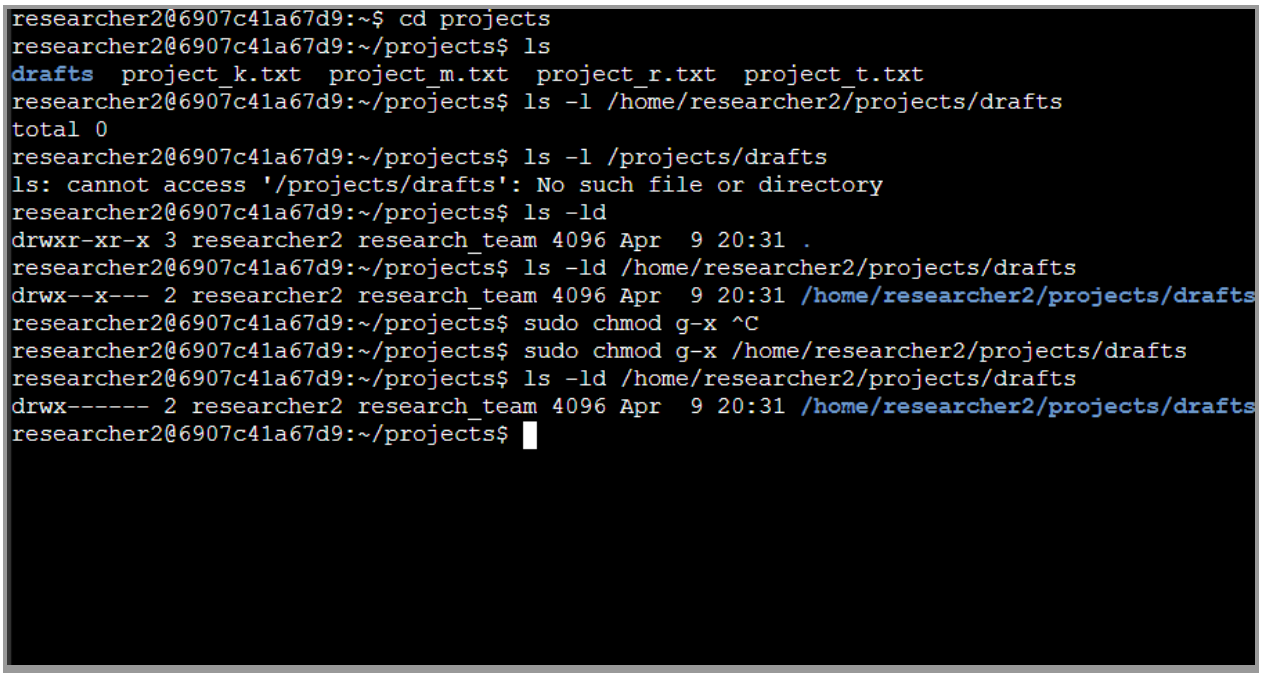
Change file permissions on a hidden file

The hidden file .project_x.txt permissions were set as read and write for the user, and write permissions were set for the group. The restricted file should have only readable permissions; no permissions should be allowed to write. The command `chmod u-w .project_x.txt` was used to remove the permission set for the user to write, the command `chmod g-w .project_x.txt` was used to remove the write permission for the group, and lastly, the group was granted access to read the file by using the command `sudo chmod g+r .project_x.txt`. I used the `ls -l .project_x.txt` command to ensure that I had permissions set correctly for the user and group with read-only access and no access given to others.



Change directory permissions

The group directory permissions were set with executive permissions. The command `ls -ld /home/researcher2/projects/drafts` was used to see which group had permissions. To remove the executive permissions that group had access to, I ran `sudo chmod g-x /home/researcher2/projects/drafts.` The correct permissions was checked using the command `ls -ld /home/researcher2/projects/drafts`



Summary

The directory was hardened by ensuring that all permissions were correctly set. The directory was set for the group to have executive permissions; now only the user has this access, and no unauthorized changes can be made to the files or directory.