File permissions in Linux

Project description

Our organization's team is currently in the process of enhancing the file permissions for specific files and directories within the projects directory. The existing permissions do not align with the appropriate level of authorization required. By diligently reviewing and updating these permissions, we aim to fortify the security of our system.

```
researcher2@717624ad2c02:~$ cd /home/researcher2/projects
researcher2@717624ad2c02:~/projects$ ls
drafts  project_k.txt  project_m.txt  project_r.txt  project_t.txt
researcher2@717624ad2c02:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:35 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:44 ..
-rw--w---- 1 researcher2 research_team   46 Jul 11 15:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jul 11 15:35 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 11 15:35 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 11 15:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_t.txt
```

Check file and directory details

The 3rd line of the screenshot displays the command I entered. The code lists all contents of the projects directory and their permissions. I used the `ls -la` option to display a detailed listing of all the file contents, including hidden files. The output of my command indicates that there is one directory named `drafts`, one hidden file named `.project_x.txt`, and 4 other project files. The 10-character string in the first column represents the permissions set on each file or directory.

Describe the permissions string

The 10 characters show the permissions that file has for all 3 categories. User, Group, Other

Character	Example	Meaning
1st	drwxrwxrwx	file type *d* for directory - for a regular file
2nd	drwxrwxrwx	read permissions for the user *r* if the user has read permissions - if the user lacks read permissions
3rd	drwxrwxrwx	write permissions for the user *w* if the user has write permissions

Character	Example	Meaning
		- if the user lacks write permissions
4th	drwxrwxrwx	execute permissions for the user *x* if the user has execute permissions - if the user lacks execute permissions
5th	drwxrwxrwx	read permissions for the group *r* if the group has read permissions - if the group lacks read permissions
6th	drwxrwxrwx	write permissions for the group *w* if the group has write permissions - if the group lacks write permissions
7th	drwxrwxrwx	execute permissions for the group *x* if the group has execute permissions - if the group lacks execute permissions
8th	drwxrwxrwx	read permissions for other *r* if the other owner type has read permissions - if the other owner type lacks read permissions
9th	drwxrwxrwx	write permissions for other *w* if the other owner type has write permissions - if the other owner type lacks write permissions
10th	drwxrwxrwx	execute permissions for other *x* if the other owner type has execute permissions - if the other owner type lacks execute permissions

For example, the file permissions for project_k.txt are -rw-rw-rw-. Since the first character is a hyphen (-), this indicates that project_k.txt is a file, not a directory. The 2nd, 5th, and 8th characters are all r, which indicates that user, group, and other all have read permissions. The 3rd, 6th and 9th characters are w, which indicates that the user, group and other have write permissions. No one has execute permissions for project_k.txt.

Change file permissions

The organization determined that “other” shouldn't have write access to any of the files in the project directory. All of the files and folders were ok except for file “project_k.txt”. I therefore had to remove “write” permissions for this file. Please see the code at the top of the image below that I used in Linux command in order to achieve this.

I then confirmed my changes with the code “ls -la”

```
researcher2@717624ad2c02:~/projects$ chmod o-w project_k.txt
researcher2@717624ad2c02:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:35 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:44 ..
-rw--w---- 1 researcher2 research_team   46 Jul 11 15:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jul 11 15:35 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 11 15:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_t.txt
researcher2@717624ad2c02:~/projects$
```

Previous permission string for “project_k.txt”

- Before -rw-rw-rw-
- After -rw-rw-r--

Change file permissions on a hidden file

My company decided they didn’t want anyone to have write access on this file and determined that only “user” and “group” would have read access

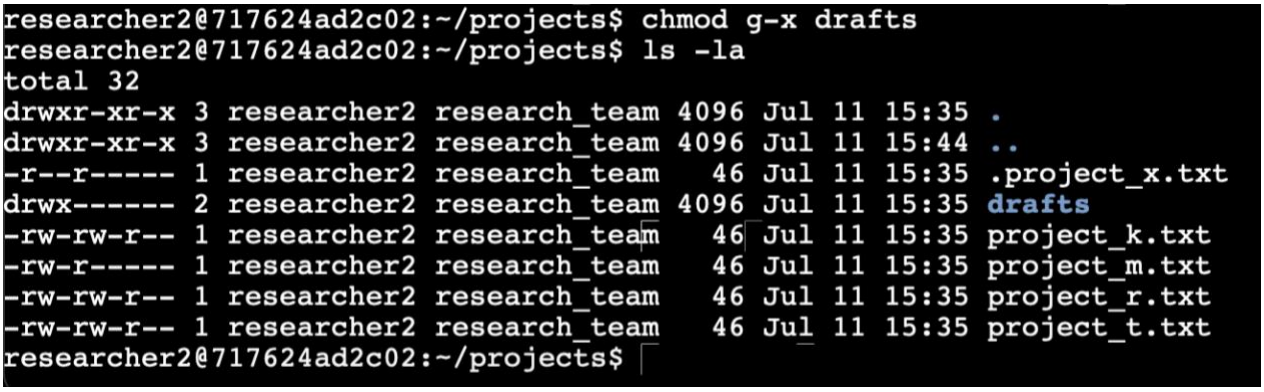
Please see the code I used below in order to achieve this

```
researcher2@717624ad2c02:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@717624ad2c02:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:35 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 15:44 ..
-r--r----- 1 researcher2 research_team   46 Jul 11 15:35 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jul 11 15:35 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 11 15:35 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 15:35 project_t.txt
researcher2@717624ad2c02:~/projects$
```

Change directory permissions

My company then requested that only the researcher2 user should have access to the drafts directory and its contents. Therefore I would need to remove execute permissions for “other”

Please see the code at the top of the image below that I used in order to achieve this. I then used `ls -la` to confirm



Summary

As you can see my company had a list of files and folders that needed their permissions modified, I changed and edited accordingly. I used `chmod` to make these changes and then `ls -la` to confirm my changes.