

# File permissions in Linux

## Project description

Permissions are needed in the file system for the users, groups and others to read, write or execute. It is important to have correct permissions in the file system. Hence, examining the existing permissions on the file system is mandatory. If the permissions on the file system do not match the authorization then modifying the permissions to authorize the appropriate users and remove any unauthorized access is needed.

## Check file and directory details

This is the file structure of the `/home/researcher2/projects` directory and the permissions of the files and subdirectory it contains.

The existing permissions on the file system are checked by the following command and the output is as follows:

```
researcher2@b3c1b6f505ee:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jun 17 13:44 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Jun 17 13:44 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jun 17 13:44 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jun 17 13:44 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jun 17 13:44 project_t.txt
```

The existing permission of the hidden file can be checked by the following command:

```
ls-la
```

The `ls` command is used to list all the directories and files of the projects directory, `ls -l` is used to check all the files and directories with the access permissions, and `ls-la` is used to list all the directories and files including hidden files with the access permissions which we have used to observe the hidden file and the access permissions of it.

The output is as follows:

```
-rw--w---- 1 researcher2 research_team  46 Jun 17 13:44 .project_x.txt
```

In the `/home/researcher2/projects` directory, the permissions of all the five files are explained in the following:

- `project_k.txt`
  - User = read, write,
  - Group = read, write
  - Other = read, write
- `project_m.txt`
  - User = read, write

- Group = read
  - Other = none
- `project_r.txt`
  - User= read, write
  - Group = read, write
  - Other = read
- `project_t.txt`
  - User = read, write
  - Group = read, write
  - Other = read
- `.project_x.txt`
  - User = read, write
  - Group = write
  - Other = none

There is also one subdirectory inside the `projects` directory named `drafts`. The permissions on `drafts` are explained in the following:

- User = read, write, execute
- Group = execute
- Other = none

## Describe the permissions string

The permissions string is divided into 4 sections. Explaining it with an example from the existing directory's permissions string:

```
drwx--x--- 2 researcher2 research_team 4096 Jun 17 13:44 drafts
```

Here, the permissions string is `drwx--x---`, where 'd' stands for directory, if '-' is present it represents a normal file.

The 2nd section includes 'rwx' which is the permission for 'User' and here the User has read(r), write(w) and execute(x) permissions.

The 3rd section includes '--x' which is for 'Group' and the Group has only execute(x) permission, since it does not have read(r) and write(w) permissions hence they are marked with '-'.

The 4th section is for Other and it is indicated by '---' which means Other has no permission for this directory.

In this way we can identify the permissions for each user.

## Change file permissions

The organization does not allow Others to have write access to any files, so at first we have to locate which files have write permissions for Others and then we need to change that.

The following command shows how to check the file permissions:

```
researcher2@b3c1b6f505ee:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jun 17 13:44 drafts
-rw-rw-rw- 1 researcher2 research_team  46 Jun 17 13:44 project_k.txt
-rw-r----- 1 researcher2 research_team  46 Jun 17 13:44 project_m.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jun 17 13:44 project_r.txt
-rw-rw-r-- 1 researcher2 research_team  46 Jun 17 13:44 project_t.txt
```

Since, we can see the Others have write permission for the project\_k.txt file, so we will remove the write permission from this file using the following command:

```
researcher2@b3c1b6f505ee:~/projects$ chmod o-w project_k.txt
```

The 'chmod' is used to change the access permissions of the files. The 'o-w' is written to remove the write(w) permission from the Other(o), and then the file name is written which is project\_k.txt.

The updated permission of the project\_k.txt is:

```
-rw-rw-r-- 1 researcher2 research_team  46 Jun 17 13:44 project_k.txt
```

We can see, the write permission from the other has been removed.

## Change file permissions on a hidden file

The hidden file is .project\_x.txt, and it should not have write permissions for anyone, but the user and group should be able to read the file.

This is the existing permissions of the hidden:

```
-rw--w---- 1 researcher2 research_team  46 Jun 17 13:44 .project_x.txt
```

This is the command used to remove the permissions for the User(u) and Group(g) to write and give permission for the group to read.

```
researcher2@b3c1b6f505ee:~/projects$ chmod u-w,g-w,g+r .project_x.txt
```

Here we can see, 'chmod' is used to change the access permissions of the file. The 'u-w' is used to remove the permission to write(w) for the 'User(u)', the 'g-w' is used to remove the permission to write(w) for the 'Group(g)' and 'g+r' is used to give permission to the 'Group(g)' to read(r) the file.

The updated permissions for the hidden file will be:

```
-r--r----- 1 reasearcher2 research_team .project_x.txt
```

## Change directory permissions

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. Here, the

researcher2 is the User. But according to the existing permissions, we can observe that the Group (research\_team) can also execute the file meaning it can access the drafts directory as well.

```
drwx--x--- 2 researcher2 research_team 4096 Jun 17 13:44 drafts
```

In order to change this permission we will use the following command:

```
researcher2@b3c1b6f505ee:~/projects$ chmod g-x drafts
```

This command shows 'chmod' is used to change the access permissions, by removing the execute(x) permission of the 'Group(g)' (g-x) of the draft directory.

## Summary

It is important to have the correct access permissions by the User, Group and Other to avoid serious problems that may happen accidentally or intentionally in the Organisation. According to the authorizations access permissions, the permissions of the directories, files, hidden files have been changed and all the commands are written in this report along with proper explanations for an easier understanding. Screenshots have also been given to understand the commands appropriately.