File permissions in Linux

Project description

In this scenario, Linux commands are used to manage file and directory permissions within a project folder structure. The objective is to ensure that access rights are aligned with organizational policies by granting or restricting read, write, and execute permissions to the user, group, and others.

Check file and directory details

- Checked the permissions of all the files & sub-directories under the ‘projects’ directory using the `ls -l` command.
- Permissions for hidden files & sub-directories can be viewed by using `ls -la` command.

```
researcher2@d33958f59fbc:~$ cd /home/researcher2/projects
researcher2@d33958f59fbc:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jul 30 08:13 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 30 08:13 project_k.tx
t
-rw-r----- 1 researcher2 research_team   46 Jul 30 08:13 project_m.tx
t
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_r.tx
t
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_t.tx
t
```

Describe the permissions string

For example, when analyzing the permissions for `project_t.txt:`

```
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_t.tx
t
```

- The permission string contains 10 characters.
- The first character (`-`) signifies that `project_t.txt` is a regular file, not a directory.

- The next nine characters are divided into three triplets, each representing the permissions for the user (owner), group, and others, respectively.
- Within each triplet:
    - The first character indicates read permission (`r`),
    - The second character indicates write permission (`w`),
    - The third character indicates execute permission (`x`).
- If "`r`", "`w`", or "`x`" is shown, the associated permission is granted to the respective user class (owner, group, or others).
- If a letter is replaced by a dash ("`-`"), it indicates that the corresponding permission is denied to that entity.

File permissions assigned in the exemplar file:

1. User permissions - has read and write permissions only.
2. Group permissions - has read and write permissions only.
3. Others' permissions - has read permissions only.

Change file permissions

1. The organization does not allow others to have write access to any files.
2. The file `project_m.txt` is a restricted file and should not be readable or writable by the group or other; only the user should have these permissions on this file.

In order to change the necessary permissions, the `chmod` command was used, as shown in the screenshot below.

- `o-w` : removes write permissions(`w`) for others(`o`)
- `g-r` : removes read permissions(`r`) for group(`g`)

```
researcher2@d33958f59fbc:~/projects$ chmod o-w project_k.txt
researcher2@d33958f59fbc:~/projects$ chmod g-r project_m.txt
researcher2@d33958f59fbc:~/projects$ ls -l
total 20
drwx--x--- 2 researcher2 research_team 4096 Jul 30 08:13 drafts
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_k.tx
t
-rw------- 1 researcher2 research_team   46 Jul 30 08:13 project_m.tx
t
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_r.tx
t
-rw-rw-r-- 1 researcher2 research_team   46 Jul 30 08:13 project_t.tx
t
```

The output confirms that 'others' lack write access to all files, while the group has no read or write permissions for `project_m.txt`.

Change file permissions on a hidden file

The research team archived `.project_x.txt`, making it a hidden file that should be readable by the user and group, but not writable by anyone.

The file permissions for the hidden file were changed using the `chmod` command.

- `u-w` : removes write permissions(`w`) for the user (`u`)
- `g-w` : removes write permissions(`w`) for group (`g`)
- `g+r` : adds read permissions(`r`) for group(`g`)

```
researcher2@d33958f59fbc:~/projects$ chmod u-w,g-w,g+r .project_x.txt
researcher2@d33958f59fbc:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 30 08:13 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 30 08:36 ..
-r--r----- 1 researcher2 research_team   46 Jul 30 08:13 .project_x.t
xt
```

The output confirms that the `.project_x.txt` file is only readable by the user and the group.

Change directory permissions

- The permissions of the `drafts` directory had to be changed so that only `researcher2` can access it, since they own all the files and folders in the `projects` directory.
- After checking the permissions in the drafts directory, it was found that the group also had executable permissions to the directory.

To remove them, `g-x` command was used.

```
researcher2@d33958f59fbc:~/projects/drafts$ chmod g-x /home/researche
r2/projects/drafts
researcher2@d33958f59fbc:~/projects/drafts$ ls -l
total 0
```

The output confirms that the user now has exclusive access to the `drafts` directory.

Summary

Throughout the project, I checked file and directory permissions using commands like `ls -l` and `ls -la`, then managed access using `chmod`. I ensured that sensitive files were restricted to the owner, hidden files had appropriate read-only access, and only the intended user—`researcher2`—could access the `drafts` directory.
These tasks highlighted how access control helps keep file systems secure.