File permissions in Linux

Author: Mudapaka Sailaxman
Email : sailaxmanpersonal@gmail.com
Date : 11 June 2025

Project description

This project focuses on managing file and directory permissions in a Linux environment using command-line tools. The goal is to secure sensitive files by appropriately assigning read, write, and execute permissions to users, groups, and others.

Through the use of commands like `chmod`, I:

- Adjust file and directory access rights.
- Remove unnecessary write or execute permissions.
- Limit access to only specific users or groups.
- Apply the principle of least privilege to reduce risk.

These tasks help enhance system security and ensure that only authorized users can interact with critical data.

Check file and directory details

```
ls -la
```

The `ls -la` command is used to **list all files and directories**, **including hidden ones**, in a **detailed format** showing file **permissions, ownership, size, and timestamps**.

The `-l` flag shows permissions and details, but **without -a, hidden files are not shown**.

```
researcher2@8ab058f1eb64:~/projects$ ls -la
total 32
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 08:51 .
drwxr-xr-x 3 researcher2 research_team 4096 Jul 11 09:21 ..
-rw--w---- 1 researcher2 research_team   46 Jul 11 08:51 .project_x.txt
drwx--x--- 2 researcher2 research_team 4096 Jul 11 08:51 drafts
-rw-rw-rw- 1 researcher2 research_team   46 Jul 11 08:51 project_k.txt
-rw-r----- 1 researcher2 research_team   46 Jul 11 08:51 project_m.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 08:51 project_r.txt
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 08:51 project_t.txt
researcher2@8ab058f1eb64:~/projects$
```

Describe the permissions string

The first 10-character string (`-rw-rw-rw-`) represents the **file type and permissions** for the file `project_k.txt`.

- The **1st character** indicates the **type** of file:
  `-` for a **regular file**
  `d` for a **directory**
  `l` for a **symbolic link**

- **Characters 2–4** (`rw-`) show the **user/owner permissions**.
  In this case, the user `researcher2` has **read** and **write** permissions.

- **Characters 5–7** (`rw-`) show the **group permissions**.
  The group `research_team` also has **read** and **write** permissions.

- **Characters 8–10** (`rw-`) show the **permissions for others** (anyone else).
  Others also have **read** and **write** permissions.

Each permission triplet can include:

- `r` = read
- `w` = write
- `x` = execute

- `-` = no permission

Change file permissions

The file `project_k.txt` currently has permissions that allow **others (everyone else)** to **read and write** to the file. This is indicated by the last part of the permission string: `rw-` for others.

To **remove the write permission** for others and make the file more secure, you can use the following command:

```
chmod o-w project_k.txt
```

- `chmod` is used to change file permissions.
- `'o'` stands for **others**.
- `'-w'` means you are **removing write permission**.

After running this command, others will only have **read** access to `project_k.txt`, and can no longer modify its contents.

```
-rw-rw-r-- 1 researcher2 research_team   46 Jul 11 08:51 project_k.txt
```

Change file permissions on a hidden file

The file `.project_x.txt` has the following permissions:

```
-rw--w---- 1 researcher2 research_team   46 Jul 11 08:51 .project_x.txt
```

- User has **read and write** permissions.
- Group has **write** permission.
- Others have **no permissions**.

To remove write permission from user and group, and add read permission to group, the command used is:

`chmod u-w,g-w+r .project_x.txt`

- `chmod` is used to change file permissions.

- ‘o’ stands for **others**.
- ‘g’ sands for **group.**
- ‘-w’ means you are **removing write permission**.
- ‘-w+r’ means you are **removing write permission and adding permission**.

After running the command, the permissions are:

```
-r--r----- 1 researcher2 research_team   46 Jul 11 08:51 .project_x.txt
```

Change directory permissions

The directory `drafts` has the following permissions:

```
drwx--x--- 2 researcher2 research_team 4096 Jul 11 08:51 drafts
```

- `d` indicates it is a **directory**.

- User has **read, write, and execute** permissions.

- Group has **execute** permission.

- Others have **no permissions**.

Only `researcher2` needs access to the directory and its contents. To remove execute permission from the group, the command used is:

`chmod g-x drafts`

After running the command, the permissions are:

```
drwx------ 2 researcher2 research_team 4096 Jul 11 08:51 drafts
```

- Only the user has **read, write, and execute** permissions.
- Group and others have **no permissions**.

Summary

This report outlines the application of permission changes to secure files and directories by limiting access to specific users and groups:

- **File `project_k.txt`**:
  Removed write permission for others using `chmod o-w`.
- **File `.project_x.txt`**:
  Removed write permissions from user and group, added read permission for group using `chmod u-w g-w+r`.
- **Directory `drafts`**:
  Removed execute permission from group to restrict access using `chmod g-x`.

These permission changes ensure that only authorized users can access or modify sensitive files and directories, improving overall system security.