File permissions in Linux

Project description

Using Linux commands, this project demonstrates how to inspect and modify file and directory permissions to ensure authorized access for the research team. It includes checking current permissions, interpreting them, and updating access levels.

Check file and directory details

Use the `ls -l` command to view file and directory details, including the owner, group, and permission settings (read, write, execute). For example, `ls -l /path/to/directory`.

Describe the permissions string

The permission string shows file access levels: `r` for read, `w` for write, and `x` for execute. The first character indicates the file type (`-` for file, `d` for directory), followed by three sets of permissions: owner, group, and others. For example: `-rwxr-xr--` means the owner has full access, the group can read and execute, and others can only read.

Change file permissions

The `chmod` command is used to modify permissions. For example, `chmod 755 filename` gives the owner read, write, and execute permissions, while the group and others get read and execute access.

Change file permissions on a hidden file

To change permissions for hidden files (those starting with a dot, such as `.hiddenfile`), use `chmod` like you would with any other file: `chmod 700 .hiddenfile` grants the owner full access and removes all access for group and others.

Change directory permissions

To modify permissions for a directory, use `chmod` with the directory path. For instance, `chmod 750 /path/to/directory`allows the owner to read, write, and execute, the group to read and execute, and blocks others.

Summary

This project focused on managing file and directory permissions to ensure secure and authorized access to resources. By using commands like `ls -l` to inspect permissions and `chmod` to modify them, you can maintain system security and proper access control in a Linux environment.