Linux File Permissions Audit & Remediation (GitHub Portfolio Version)

Objective

This project demonstrates practical Linux system administration and security hardening skills. The objective was to audit file and directory permissions within a research environment and remediate misconfigurations using the principle of least privilege.

Environment

Linux-based system using Bash shell. File structure located in /home/researcher2/projects.

Auditing File Permissions

```
ls -la
```

The -l flag displays detailed file information, including the 10-character permission string. The -a flag ensures hidden files (e.g., .project_x.txt) are included in the audit.

Interpreting the 10-Character Permission String

```
-rw-rw-r--
```

The first character indicates file type (- for file, d for directory). The remaining characters are grouped into user, group, and other permissions. r = read, w = write, x = execute, and - indicates no permission.

Identified Security Risks

During the audit, certain files granted write permissions to unauthorized users, violating least privilege principles. This posed a risk of unauthorized modification.

Remediation Using chmod

Remove write access for others:

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

Secure hidden archived file (read-only for user and group):

```
chmod 440 .project_x.txt
```

Restrict directory access to owner only:

```
chmod 700 drafts
```

Skills Demonstrated

• Linux command-line proficiency • File permission auditing • Access control enforcement • Principle of least privilege implementation • Numeric and symbolic chmod usage

Summary

This project highlights hands-on experience with Linux access control mechanisms. By identifying and remediating permission misconfigurations, I strengthened system integrity and demonstrated applied cybersecurity fundamentals relevant to system administration and incident response.