Linux Permission Audit Report

Auditing Linux permissions to enhance security and enforce least-privilege access

By:
Junaid AbdulRahman A
Cybersecurity Analyst

Date:07 December 2025

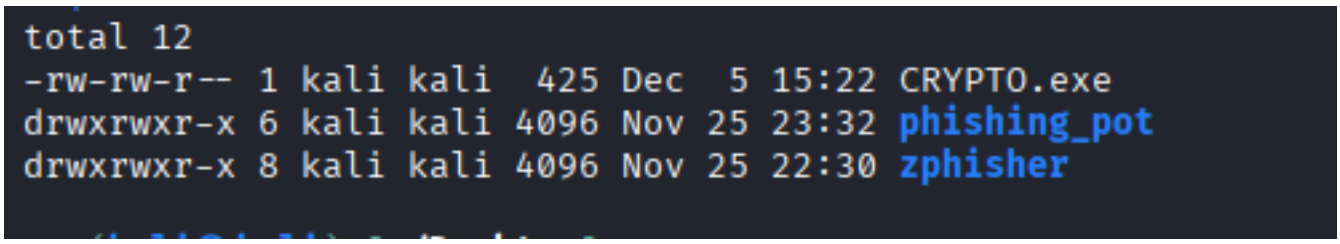
1. Executive Summary

The Linux Permission Audit project was undertaken to evaluate and strengthen access control mechanisms across critical directories. The primary goal was to ensure that file and directories permissions align with organizational security policies, regulatory compliance requirements, and best practices for least-privilege access.

2. Types of Permissions

- Read (r): Allows viewing file contents or listing directory contents.
- Write (w): Allows modifying directory.
- Execute (x): Allows running a file as a program or entering a directory.
Permissions.
whereby this permissions apply to three categories:
- User (u): The file owner.
- Group (g): Users in the file's group.
- Others (o): other users.

3. Viewing Permissions



For CRYPTO.exe
-rw- → user has read and write permissions.
-rw- → group also has read and write permissions.
- r-- → others have read only permission.

4. Adding Permissions

Permissions can be added to any category using the symbolic method (+)
chmod(command) u+x filename Add execute permission for user
chmod(command) g+w filename Add write permission for group
chmod(command) o+r filename Add read peermission for others

```
┌──(kali㉿kali)-[~/Desktop]
└─$ chmod u+x CRYPTO.exe
```

```
total 12
-rwxrw-r-- 1 kali kali  425 Dec  5 15:22 CRYPTO.exe
drwxrwxr-x 6 kali kali 4096 Nov 25 23:32 phishing_pot
drwxrwxr-x 8 kali kali 4096 Nov 25 22:30 zphisher
```

Added execute permission for CRYPTO.exe user using the symbolic mode

Or using numeric method (4, 2, 1):
- r = 4, w = 2, x = 1.
- Combine values for each category.
chmod 755 filename User: rwx (7), Group: r-x (5), Others: r-x (5)

```
┌──(kali㉿kali)-[~/Desktop]
└─$ chmod 755 phishing_pot
```

```
total 12
-rwxrw-r-- 1 kali kali  425 Dec  5 15:22 CRYPTO.exe
drwxr-xr-x 6 kali kali 4096 Nov 25 23:32 phishing_pot
drwxrwxr-x 8 kali kali 4096 Nov 25 22:30 zphisher
```

Added read, write and execute permission for user, read and execute for group, read and execute for others using the 755 symbolic code for the phishing_pot directory

5. Removing Permissions

Permissions can also be removed from any category using the symbolic method (-):

chmod u-x filename Remove execute permission for user
chmod g-w filename Remove write permission for group
chmod o-r filename Remove read permission for others

```
┌──(kali㉿kali)-[~/Desktop]
└─$ chmod u-x CRYPTO.exe
```

```
total 12
-rw-rw-r-- 1 kali kali  425 Dec  5 15:22 CRYPTO.exe
drwxr-xr-x 6 kali kali 4096 Nov 25 23:32 phishing_pot
drwxrwxr-x 8 kali kali 4096 Nov 25 22:30 zphisher
```

Removed execute permission for CRYPTO.exe user using the symbolic mode (u-x)

6. Risks and Best Practices

- Risk of over-permissioning: Granting 777 (full permissions to all categories) can expose files to unauthorized modification.
- Best practice: Grant the minimum required permissions.
- Audit regularly: Use ls -l to check permissions.

7. Conclusion

The Linux Permission Audit successfully achieved its objectives of evaluating, correcting, and strengthening access control across the system directory. Through a systematic review of directory permissions, the audit identified critical misconfigurations and areas of excessive privilege that posed potential security risks.