🛡️ Project Phoenix – Linux Permissions & Access Control

📘 Scenario Overview

Following the successful investigation of Project Phoenix's critical failures, I was assigned to secure the project's infrastructure. The goal of this lab was to design a **secure and collaborative Linux file system**, ensuring sensitive data protection, correct ownership, and controlled team access.

This lab simulates real-world **least-privilege access control** and **secure DevOps collaboration** practices.

🎯 Security Objectives

- Protect sensitive project files using strict permissions
- Assign correct ownership and group control
- Secure the main project directory from unauthorized access
- Enable safe collaboration through group inheritance

🔐 Step 1: Create a Secure File for Sensitive Keys

Task

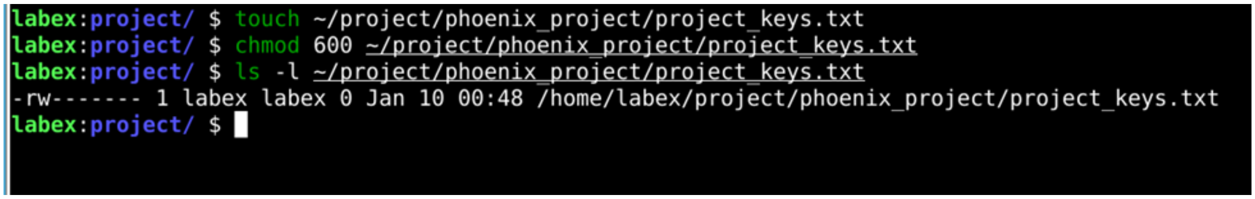
Create a confidential file accessible **only by its owner**.

Command Used

```
touch ~/project/phoenix_project/project_keys.txt
chmod 600 ~/project/phoenix_project/project_keys.txt
```

Result

- File owner: read/write access
- Group & others: no access
- Ensures strict confidentiality for sensitive keys



```
labex:project/ $ touch ~/project/phoenix_project/project_keys.txt
labex:project/ $ chmod 600 ~/project/phoenix_project/project_keys.txt
labex:project/ $ ls -l ~/project/phoenix_project/project_keys.txt
-rw------- 1 labex labex 0 Jan 10 00:48 /home/labex/project/phoenix_project/project_keys.txt
labex:project/ $ 
```

👤 Step 2: Assign Ownership of Project Resources

Objective

Transfer ownership of all project files to the technical lead and development group.

Command Used

```
sudo chown -R dev_lead:developers ~/project/phoenix_project
```

Result

- Owner: `dev_lead`
- Group: `developers`
- Applied recursively to all files and directories

```
labex:project/ $ touch ~/project/phoenix_project/project_keys.txt
labex:project/ $ chmod 600 ~/project/phoenix_project/project_keys.txt
labex:project/ $ ls -l ~/project/phoenix_project/project_keys.txt
-rw------- 1 labex labex 0 Jan 10 00:48 /home/labex/project/phoenix_project/project_keys.txt
labex:project/ $ 
```

🏰 Step 3: Secure the Main Project Directory

Policy

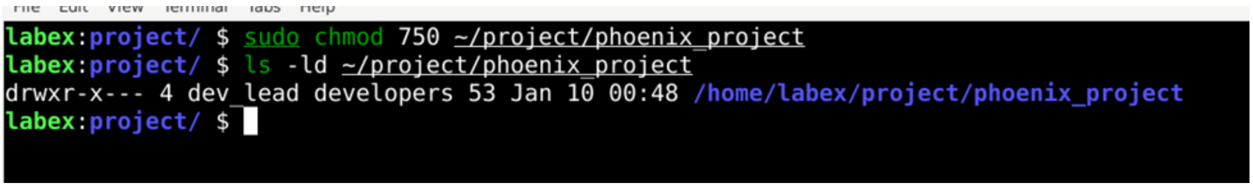
Role	Permissions
Owner (dev_lead)	read, write, execute
Group (developers)	read, execute
Others	no access

Command Used

```
sudo chmod 750 ~/project/phoenix_project
```

Explanation

- Owner has full control
- Group can enter and list directory contents
- Outsiders are fully blocked



```
labex:project/ $ sudo chmod 750 ~/project/phoenix_project
labex:project/ $ ls -ld ~/project/phoenix_project
drwxr-x--- 4 dev_lead developers 53 Jan 10 00:48 /home/labex/project/phoenix_project
labex:project/ $
```

🤝 Step 4: Enable Secure Collaboration in `src`

Objective

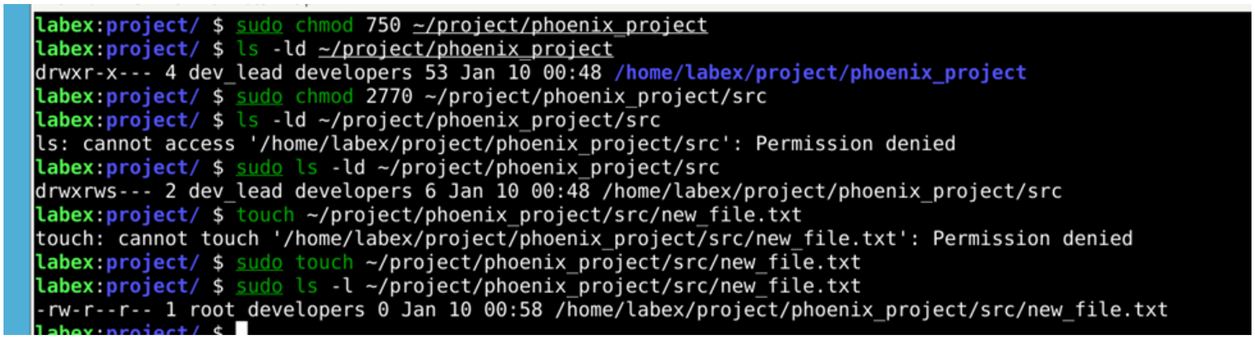
Ensure all new files created in `src` automatically inherit the `developers` group.

Command Used

```
sudo chmod g+s ~/project/phoenix_project/src
```

Why This Matters

- Enforces consistent group ownership
- Prevents access conflicts
- Supports collaborative development securely



```
labex:project/ $ sudo chmod 750 ~/project/phoenix_project
labex:project/ $ ls -ld ~/project/phoenix_project
drwxr-x--- 4 dev_lead developers 53 Jan 10 00:48 /home/labex/project/phoenix_project
labex:project/ $ sudo chmod 2770 ~/project/phoenix_project/src
labex:project/ $ ls -ld ~/project/phoenix_project/src
ls: cannot access '/home/labex/project/phoenix_project/src': Permission denied
labex:project/ $ sudo ls -ld ~/project/phoenix_project/src
drwxrws--- 2 dev_lead developers 6 Jan 10 00:48 /home/labex/project/phoenix_project/src
labex:project/ $ touch ~/project/phoenix_project/src/new_file.txt
touch: cannot touch '/home/labex/project/phoenix_project/src/new_file.txt': Permission denied
labex:project/ $ sudo touch ~/project/phoenix_project/src/new_file.txt
labex:project/ $ sudo ls -l ~/project/phoenix_project/src/new_file.txt
-rw-r--r-- 1 root developers 0 Jan 10 00:58 /home/labex/project/phoenix_project/src/new_file.txt
labex:project/ $
```

🧠 Skills Demonstrated

- Linux file permissions (`chmod`)
- Ownership management (`chown`)
- Least-privilege security design
- Secure collaboration with setgid
- Protecting sensitive configuration data
- Practical DevSecOps access control