Lab3: Advanced File Permissions, Access Controls, and Password Security

Instructor: Dr. Maryam R. Aliabadi
Lab Duration: 2 hours
Objective: Apply advanced file security mechanisms, ACLs, SELinux enforcement, and evaluate account security using password auditing tools.

Learning Outcomes

- Apply standard and special file permissions
- Configure ACLs and SELinux contexts
- Understand password vulnerabilities and importance of complexity
- Implement and enforce a comprehensive account security policy

---

Pre-lab Setup

These labs should be performed on the Kali operating system that you installed in

Lab1.

Create sample users and groups:

sudo useradd alice

sudo useradd bob

sudo useradd charlie

sudo groupadd devteam

sudo usermod -aG devteam alice

sudo usermod -aG devteam bob

Hint: Use id alice to verify group membership.

Task 1: Develop an Account Security Policy

Instructions: Draft a short policy covering:

- Minimum password length/complexity
- File ownership rules
- Access control guidelines for sensitive directories

Deliverable: 3–5 bullet points.

Hint: Think of a policy you could actually apply on the lab VM.

Task 2: Standard File Permissions

Exercise 2.1: Create files and directories

```
mkdir ~/lab_files
cd ~/lab_files
touch report.txt data.csv
mkdir project_docs
```

Exercise 2.2: Inspect permissions

```
ls -l
```

Expected Output:

```
-rw-r--r-- 1 youruser yourgroup 0 Sep 18 12:00 report.txt
-rw-r--r-- 1 youruser yourgroup 0 Sep 18 12:00 data.csv
drwxr-xr-x 2 youruser yourgroup 4096 Sep 18 12:00 project_docs
```

Exercise 2.3: Modify permissions

```
chmod 640 report.txt       # Owner rw, group r, others none
sudo chown alice:devteam data.csv
ls -l
```

Hint: Verify ownership and permissions after each command.

Task 3: Special Permissions

SetUID Example

sudo chmod u+s report.txt

ls -l report.txt

Expected Output:

-rwsr----- 1 alice devteam 0 Sep 18 12:05 report.txt

SetGID Example

sudo chmod g+s project_docs

mkdir project_docs/new_folder

ls -ld project_docs/new_folder

Hint: New folder should inherit the group of project_docs.

Sticky Bit Example

sudo chmod +t project_docs

Scenario Test: Multiple users try deleting files; only file owner can delete.

Task 4: Access Control Lists (ACLs)

Check existing ACLs

getfacl report.txt

Set ACLs

setfacl -m u:bob:r report.txt

setfacl -m u:charlie:--- report.txt

getfacl report.txt

Example Output:

file: report.txt

owner: alice

group: devteam

user::rw-

user:bob:r--

user:charlie:---

group::r--

mask::r--

other::---

Remove ACL

setfacl -b report.txt

getfacl report.txt

Task 5: SELinux Enforcement

Check SELinux status

sestatus

View file context

ls -Z report.txt

Change context

sudo chcon -t httpd_sys_content_t report.txt

ls -Z report.txt

SELinux Boolean example

getsebool -a | grep httpd

sudo setsebool -P httpd_enable_homedirs on

Hint: Test access as a restricted user to see SELinux enforcement in action.

Task 6: File Access Policy Challenge

Scenario:

- Alice: read-only access to project_docs/report.txt

- Bob: full access to project_docs/report.txt
- Charlie: no access

Instructions: Implement policy using standard permissions, ACLs, or SELinux.
Hint: Test each user's access with su username and cat or echo commands.

Task 7: Password Security Scenario – "Password Cracking Challenge"

Scenario Steps:

1. **Create 5 new users**:

sudo useradd user1

sudo useradd user2

sudo useradd user3

sudo useradd user4

sudo useradd user5

2. **Set passwords** (some simple, some complex):

- user1: password
- user2: 12345
- user3: LabFSCT7!
- user4: Cyber2025!
- user5: Qwerty2025

3. **Run John the Ripper**

sudo unshadow /etc/passwd /etc/shadow > mypasswd.txt

john mypasswd.txt

john --show mypasswd.txt

4. Start the johnny program by clicking **Applications**, then **05 - Password Attacks**, then **johnny**. Click **Open password file** and open the file created. Select the accounts to scan and click **Start new attack**.

Reflection Questions:

- Which passwords were compromised using each tool?
- How does password complexity improve security?
- How can password policies complement file access controls?

Task 8: Brute-Force SSH Passwords Using Hydra

Scenario: You are a security analyst tasked with testing password strength on a small network.

- Attacker Machine: Kali Linux
- Victim Machine: Ubuntu Linux
- Goal: Identify weak passwords for user accounts via SSH.

Setup:

1. Victim Machine:
    - Ensure SSH service is running:

sudo systemctl status ssh

sudo systemctl start ssh

    - User accounts for testing: alice, bob, charlie
    - Find the IP address:

ip a

2. Attacker Machine (Kali Linux):
    - Ensure Hydra is installed:

hydra -h

    - Wordlists available: /usr/share/wordlists/rockyou.txt

Step 1 — SSH to Victim:

- Test logging in manually:

ssh <username>@<victim_IP>

- Observe valid accounts.

Step 2 — Prepare Username and Password Lists:

- Create a file with the usernames:

alice bob charlie

- Start with a small password list for testing.

Step 3 — Construct Hydra Command:

hydra -L <userlist> -P <passwordlist> ssh://<target_IP>

Use Hydra to specify:

- *Target host IP*

- *Protocol (ssh)*

- *Username list (-L)*

- *Password list (-P)*

- *Optional verbose flag (-v)*

Step 4 — Run Hydra:

- Execute the command on Kali.
- Record which accounts are successfully cracked.

Step 5 — Analyze Results:

- Which accounts were vulnerable?
- How long did it take?
- Compare small vs large wordlists.
    - **Small wordlist:** much faster, fewer passwords tested, may miss some valid passwords. You can create your own dictionary of passwords and use it to crack the password.
    - **Rockyou.txt (~14 MB, 14 million passwords):** slower but covers a large set of common passwords, more likely to find weak passwords.

Student Input Sections:

1. Victim IP: _____
2. Usernames tested: _____
3. Password list used: _____
4. Accounts cracked: _____
5. Observations/Notes: _____

Lab Submission

1. Screenshots for each task (permissions, ACLs, SELinux, John output)
2. Reflection on file security configuration and password security
3. Reflection on password cracking using John, Johnny and Hydra
4. *You have to upload your work in the following format:*

Filename: Lab3-FirstName-Lastname-StdNo.PDF