

Linux IAM & Hardening Project Report

Student Name: Mohammad Kaif

Date: 04-11-2025

Lab Environment: Kali Linux VM

Table of Contents

1. Baseline Policy
 2. User and Group Creation
 3. Sudoers Configuration
 4. Shared Project Directory with ACLs
 5. Audit Configuration
 6. Vulnerable VM: Misconfigurations & Fixes
 7. Remediation Checklist
 8. Audit Logs & Final State Summary
-

1. Baseline Policy Document

Define user roles, sudo needs, file access levels.

- **Admin**
 - Username: alice
 - Full sudo rights
- **Developer**
 - Username: bob
 - Limited sudo (only apt update, systemctl restart apache2)
- **Auditor**
 - Username: carol

- No sudo; read-only access to /var/log and project file

```
root@kali: /home/kali/policy
File Actions Edit View Help
GNU nano 8.2 readme.txt
A. Admins (Group: admins)
Access Level: Full
Permissions:
-Full sudo (/etc/sudoers.d/admins)
-Read, write, execute all system files
-Manage users and groups
-Configure and restart all services
-Access system and application logs
B. Developers (Group: devs)
Access Level: Limited
Permissions:
-Access project directories: /opt/project/, /var/www/project/
-Restart specific services (e.g. Apache, Nginx, custom app)
-Deploy new code versions
-No user management or system-wide configuration rights
C. Auditors (Group: auditors)
Access Level: Read-Only
Permissions:
[ Read 40 lines ]
^G Help      ^O Write Out ^F Where Is   ^K Cut       ^I Execute   ^C Location
^X Exit      ^R Read File ^N Replace    ^U Paste     ^J Justify   ^_ Go To Line
```

2. User and Group Creation

Commands Used

sudo groupadd devs

sudo groupadd auditors

sudo useradd -m -s /bin/bash alice

sudo usermod -aG sudo alice

sudo useradd -m -s /bin/bash bob

sudo usermod -aG devs bob

sudo useradd -m -s /bin/bash carol

sudo usermod -aG auditors carol

✓ Verification

id alice

id bob

id carol

 *Screenshot: Terminal output showing group memberships.*

```
root@kali: /home
File Actions Edit View Help
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] y
info: Adding new user 'carol' to supplemental / extra groups 'users' ...
info: Adding user 'carol' to group 'users' ...

(root@kali) ~$ sudo useradd -m -G admins alice # Admin user
zsh: bad pattern: ^[[200~sudo

(root@kali) ~$ sudo passwd alice
sudo useradd -m -G devs bob # Developer user
sudo passwd bob
sudo useradd -m -G auditors carol # Auditor user
sudo passwd carol
^[[201~New password:
Retype new password:
No password has been supplied.
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged

(root@kali) ~$ usermod -aG admins alice

(root@kali) ~$ usermod -aG devs bob

(root@kali) ~$ usermod -aG auditors carol

(root@kali) ~$

root@kali: /home
File Actions Edit View Help
(root@kali) ~$ addgroup Admins
err: Please enter a username matching the regular expression
configured via the NAME_REGEX configuration variable. Use the
--allow-bad-names' option to relax this check or reconfigure
NAME_REGEX in configuration.

(root@kali) ~$ addgroup admins
info: Selecting GID from range 1000 to 59999 ...
info: Adding group 'admins' (GID 1003) ...

(root@kali) ~$ addgroup devs
info: Selecting GID from range 1000 to 59999 ...
info: Adding group 'devs' (GID 1004) ...

(root@kali) ~$ addgroup auditors
info: Selecting GID from range 1000 to 59999 ...
info: Adding group 'auditors' (GID 1005) ...

(root@kali) ~$ adduser alice
info: Adding user 'alice' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group 'alice' (1006) ...
info: Adding new user 'alice' (1006) with group 'alice (1006)' ...
info: Creating home directory '/home/alice' ...
info: Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for alice
Enter the new value, or press ENTER for the default
Full Name []:
Room Number []:
```

3. Sudoers Configuration

Files Edited

- /etc/sudoers.d/devs

%devs ALL=(ALL) NOPASSWD: /usr/bin/systemctl restart apache2, /usr/bin/apt update

- /etc/sudoers.d/admins (optional if using sudo group)

 *Screenshot: Content of sudoers files*

```
root@kali: /etc
GNU nano 8.2 sudoers *
# While you shouldn't normally run git as root, you need to with etckeeper
#Defaults:%sudo env_keep += "GIT_AUTHOR_* GIT_COMMITTER_*"
# Per-user preferences; root won't have sensible values for them.
#Defaults:%sudo env_keep += "EMAIL DEBEMAIL DEBFULLNAME"
# "sudo scp" or "sudo rsync" should be able to use your SSH agent.
#Defaults:%sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"
# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"
# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
root    ALL=(ALL:ALL) ALL
kaif    ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
%artist ALL=(ALL) /home/kaif/music
%listener ALL=(ALL) /home/kaif/music
%admins ALL=(ALL:ALL) ALL
%devs ALL=(ALL) NOPASSWD: /usr/bin/systemctl restart apache2, /usr/bin/apt update
# See sudoers(5) for more information on "@include" directives:
@include /etc/sudoers.d
^G Help      ^O Write Out ^F Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

✓ sudo -l output for each user to confirm their permissions.

4. Shared Project Directory with ACLs

Setup Commands

```
sudo mkdir /policy/project
```

```
sudo chown root:devs /policy/project
```

```
sudo chmod 2775 /policy/project
```

```
sudo setfacl -m g:devs:rwX /policy/project
```

```
sudo setfacl -m o:rx /policy/project
```

```
sudo setfacl -d -m g:devs:rwX /policy/project
```

```
sudo setfacl -d -m o:rx /policy/project
```

✓ Verification

```
getfacl /policy/project
```

📸 Screenshot: `getfacl` and `ls -ld /policy/project` output.

```
root@kali: /home/kaif/policy
File Actions Edit View Help
root@kali)~# cd /home/kaif/policy
root@kali)~# ls -l
total 8
drwxrwxr-x 2 kaif devs 4096 Nov  4 07:23 project
-rw-r--r-- 1 root root  719 Nov  4 07:05 readme.txt
root@kali)~# setfacl -m g:devs:rwx project
root@kali)~# setfacl -m o:r-x project
root@kali)~# ls -ld
drwxr-xr-x 3 root root 4096 Nov  4 07:23 .
root@kali)~# getfacl project
# file: project
# owner: kaif
# group: devs
user::rwx
group::rwx
group:devs:rwx
mask::rwx
other::r-x
root@kali)~#
```

5. Audit Configuration

Installation & Service

sudo apt install auditd

sudo systemctl enable --now auditd

Rules Added

sudo auditctl -w /etc/passwd -p wa -k user-modify

sudo auditctl -w /etc/sudoers -p wa -k sudoers-mod

 Screenshot: Output of sudo auditctl -l

6. Vulnerable Snapshot: Misconfigurations Found and Fixed

Issue 1: World-Writable /etc/cron.d

- **Before:**

ls -ld /etc/cron.d

(e.g. drwxrwxrwx)

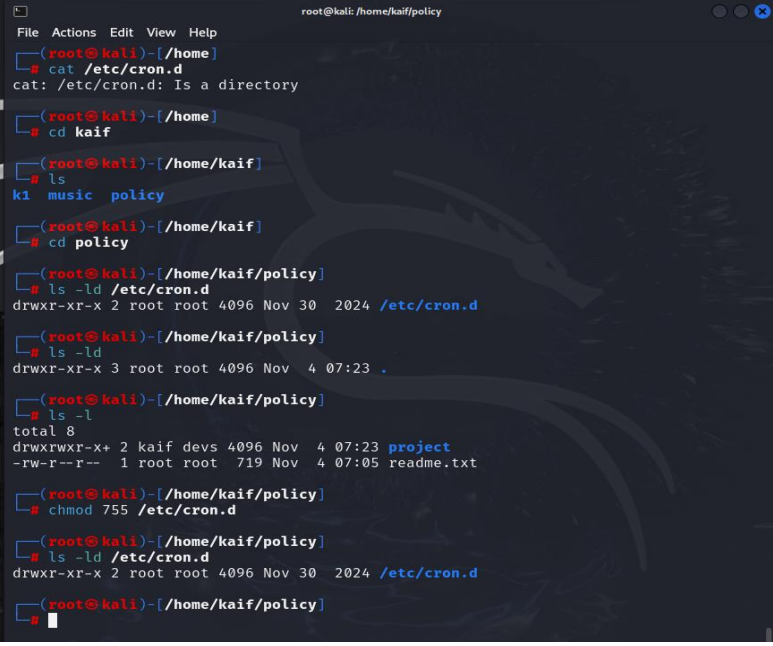
- **Fix:**

sudo chmod 755 /etc/cron.d

- **After:**

ls -ld /etc/cron.d

Screenshots before & after



```
root@kali: /home/kaif/policy
File Actions Edit View Help
(root@kali)-[/home]
# cat /etc/cron.d
cat: /etc/cron.d: Is a directory
(root@kali)-[/home]
# cd kaif
(root@kali)-[/home/kaif]
# ls
k1 music policy
(root@kali)-[/home/kaif]
# cd policy
(root@kali)-[/home/kaif/policy]
# ls -ld /etc/cron.d
drwxr-xr-x 2 root root 4096 Nov 30 2024 /etc/cron.d
(root@kali)-[/home/kaif/policy]
# ls -ld
drwxr-xr-x 3 root root 4096 Nov 4 07:23 .
(root@kali)-[/home/kaif/policy]
# ls -l
total 8
drwxrwxr-x+ 2 kaif devs 4096 Nov 4 07:23 project
-rw-r--r-- 1 root root 719 Nov 4 07:05 readme.txt
(root@kali)-[/home/kaif/policy]
# chmod 755 /etc/cron.d
(root@kali)-[/home/kaif/policy]
# ls -ld /etc/cron.d
drwxr-xr-x 2 root root 4096 Nov 30 2024 /etc/cron.d
(root@kali)-[/home/kaif/policy]
#
```

Issue 2: Unrestricted NOPASSWD in sudoers

- **Before:**

```
sudo grep -R NOPASSWD /etc/sudoers*
```

(e.g. `user1 ALL=(ALL) NOPASSWD: ALL`)

- **Fix:**

Edit file via `sudo visudo` or `sudo nano /etc/sudoers.d/user1`, replace with:

```
user1 ALL=(ALL) /usr/bin/apt
```

- **After:**

Confirm with:

```
sudo -l -U user1
```

Screenshots before & after

```
root@kali:/home/kaif/policy
File Actions Edit View Help
# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root  ALL=(ALL:ALL) ALL
kaif  ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL
%artist ALL=(ALL) /home/kaif/music
%listener ALL=(ALL) /home/kaif/music
%admins ALL=(ALL:ALL) ALL
%devs ALL=(ALL) NOPASSWD: /usr/bin/systemctl restart apache2, /usr/bin/apt update

# See sudoers(5) for more information on "@include" directives:
@include_dir /etc/sudoers.d

(root@kali) ~/home/kaif/policy
$ nano /etc/sudoers
(root@kali) ~/home/kaif/policy
$ grep -R "NOPASSWD" /etc/sudoers /etc/sudoers.d/
/etc/sudoers:charlie ALL=(ALL) NOPASSWD: ALL
/etc/sudoers:%devs ALL=(ALL) NOPASSWD: /usr/bin/systemctl restart apache2, /usr/bin
/apt update
/etc/sudoers.d/ospd-openvas:gvm ALL = NOPASSWD: /usr/sbin/openvas
/etc/sudoers.d/kali-grant-root:kali-trusted ALL=(ALL:ALL) NOPASSWD: ALL
(root@kali) ~/home/kaif/policy
$
```

🔑 Issue 3: Weak Permissions on /etc/shadow

- **Before:**

ls -l /etc/shadow

(e.g. -rw-r--r--)

- **Fix:**

sudo chown root:shadow /etc/shadow

sudo chmod 640 /etc/shadow

- **After:**

ls -l /etc/shadow

📷 Screenshots before & after

```
(root@kali) ~/home/kaif
$ ls -l /etc/shadow
-rw-r--r-- 1 root shadow 1947 Nov  4 06:21 /etc/shadow

(root@kali) ~/home/kaif
$ chmod 640 /etc/shadow

(root@kali) ~/home/kaif
$ ls -l /etc/shadow
-rw-r----- 1 root shadow 1947 Nov  4 06:21 /etc/shadow

(root@kali) ~/home/kaif
$
```

7. Remediation Checklist ☒

Checkpoint	Status
All users in correct groups	✓
Sudo access limited via /etc/sudoers.d/	✓
/etc/sudoers protected and audited	✓
/policy/project writeable only by devs	✓
Audit rules log changes to passwd/sudoers	✓
Cron directories not world-writable	✓
No NOPASSWD:ALL in sudoers	✓
Sensitive files (e.g. /etc/shadow) secured	✓

8. Audit Logs and Final State Summary

Audit Log Examples

```
sudo ausearch -k user-modify
```

```
sudo ausearch -k sudoers-mod
```

 *ausearch logs showing modifications*

Final System State

Users created with appropriate permissions

- Project directory ACL verified
- Audit logs enabled and working

- All found vulnerabilities remediated
-

Major Learnings

Through this project, I gained practical insights into **Linux Identity and Access Management (IAM)** and **system hardening** practices that form the foundation of cybersecurity and ethical hacking. Key learnings include:

1. **Role-Based Access Control (RBAC):**

I learned how to design and enforce a structured access hierarchy using groups such as admins, devs, and auditors — ensuring each role has only the privileges needed to perform their tasks.

2. **Principle of Least Privilege:**

Configuring limited sudo permissions (via `/etc/sudoers.d/`) helped me understand how privilege minimization can significantly reduce attack surfaces and prevent privilege escalation.

3. **Secure File Access Management:**

Implementing **POSIX permissions** and **Access Control Lists (ACLs)** for shared directories (`/policy/project`) showed how to control read/write access granularly across teams.

4. **System Auditing and Monitoring:**

Setting up **auditd** rules provided hands-on experience in tracking changes to critical files like `/etc/passwd` and `/etc/sudoers`, strengthening accountability and traceability in system operations.

5. **Vulnerability Identification & Remediation:**

Discovering real-world misconfigurations (e.g., world-writable cron directories, weak permissions on `/etc/shadow`, and unrestricted NOPASSWD rules) reinforced how small oversights can lead to significant security risks — and how to remediate them properly.

6. **Hardening Mindset:**

The project cultivated a security-oriented mindset, emphasizing continuous verification (`sudo -l`, `getfacl`, `ausearch`) and post-fix validation to maintain a hardened environment.