Linux IAM & Hardening Mini Project

Objective

Design and implement a secure user/group and permission model on an Ubuntu server. Identify and fix misconfigurations in a deliberately vulnerable lab VM, implement auditing, and produce a remediation report.

Tools / Environment

- Ubuntu VM (lab)
- Kali/Attacker VM for testing
- sudo access on lab VM

Baseline IAM Policy

Roles and access model designed for a small 3-member team:

- Admin – full system management, limited sudo.
- Developer (devteam) – project directory write, no system-wide sudo.
- Auditor – read-only access to logs and configurations.

Key Commands

Section	Command / Configuration	Description / Purpose
Group Creation	`sudo groupadd admin`	Create **admin** group for privileged users.
	`sudo groupadd dev`	Create **developer** group for project contributors.

	`sudo groupadd auditor`	Create **auditor** group for read-only monitoring.
User Creation	`sudo useradd -m -s /bin/bash -G admin alice_admin`	Create user **alice_admin** and add to admin group.
	`sudo useradd -m -s /bin/bash -G dev dev1`	Create user **dev1** and assign to dev group.
	`sudo useradd -m -s /bin/bash -G auditor aud1`	Create user **aud1** and assign to auditor group.
Sudoers Configuration	`%admin ALL=(ALL) ALL`	Allow **admin group** full root privileges.
	`%dev ALL=(root) /usr/local/bin/deploy.sh, /bin/systemctl restart project.service`	Allow **dev group** limited sudo for deployment & service restart only.
Project Folder Setup	`sudo mkdir -p /srv/project`	Create project directory.
	`sudo chown root:dev /srv/project`	Set **owner** as root and **group** as dev.

	`sudo chmod 2770 /srv/project`	Give **rwx** to owner & group, restrict others; ensure **setgid** for group inheritance.
ACL Configuration	`sudo setfacl -m g:auditor:rx /srv/project/published`	Grant **auditor group** read/execute on published folder.
AuditD Installation	`sudo apt install -y auditd`	Install **auditd** for monitoring system events.
Audit Watch Rules	`sudo auditctl -w /etc/sudoers -p wa -k sudoers_changes`	Watch for **write/attribute** changes to sudoers file.
	`sudo auditctl -w /etc/passwd -p wa -k passwd_changes`	Watch for **write/attribute** changes to passwd file.

Implementation Steps

1. Created users and groups using useradd and groupadd.
2. Configured sudoers with least-privilege using visudo.
3. Applied POSIX permissions and ACLs on /srv/project_shared.

4. Enabled auditd to track changes in /etc/sudoers and /etc/passwd.
5. Captured evidence before and after fixes.

Identified Misconfigurations

- /etc/cron.d/backup was world-writable (777).
- /etc/sudoers had permissions 666 instead of 440.
- devuser had unrestricted sudo (NOPASSWD:ALL).

Remediation Actions

- chmod 644 /etc/cron.d/backup → Secure permissions.
- chmod 440 /etc/sudoers → Restricted editing.
- Modified /etc/sudoers via visudo to limit devuser commands.
- Applied setfacl on /srv/project_shared to restrict write access.

Evidence & Audit Logs (Final Submission Summary)

All remediation and audit actions were performed on the Ubuntu lab VM (user: ubuntu). The following evidence files correspond to live outputs captured before and after the fixes.

No.	Filename	Description
1️⃣	1_ls_-l_before_fix.txt	Permissions listing of /etc/cron.d, /etc/sudoers, /etc/passwd before remediation.
2️⃣	2_ls_-l_after_fix.txt	Permissions listing after remediation verifying secure chmod 440 /etc/sudoers.
3️⃣	3_getfacl_before_fix.txt	ACL of /srv/project_shared before fix (world-writable).
4️⃣	4_getfacl_after_fix.txt	ACL of /srv/project_shared after fix (restricted).

5️⃣	5_auditd_log_snippet.txt	Audit log entries showing changes to sudoers, passwd, and cron files.

Audit logs confirm that all changes were tracked using keys:

- sudoers_change
- passwd_mod
- cron_perm_fix