Linux Hardening Audit Tool

Objective

The objective of this project is to develop a simple Python script that performs a basic security audit on a Linux system. The script checks common hardening settings to help identify weak configurations and improve the overall security posture.

Abstract

This project introduces a Linux Hardening Audit Tool written in Python. The tool checks three essential configurations:

- If root login via SSH is disabled
- If the file permissions for /etc/passwd and /etc/shadow are correct
- If the UFW firewall is installed and enabled

The purpose of the tool is to assist learners and system users in identifying basic vulnerabilities and applying best practices. It is lightweight, easy to understand, and serves as a foundation for more advanced system hardening scripts.

Tools and Technologies Used

- Operating System: Kali Linux
- Programming Language: Python 3
- Python Modules: os, subprocess
- Files Accessed: /etc/passwd, /etc/shadow, /etc/ssh/sshd_config
- Firewall Tool: ufw (Uncomplicated Firewall)

Implementation

1. **Created the audit script (audit_tool.py)**

    - Used Python to read system files and check configurations.

2. **SSH Root Login Check**

    - Read /etc/ssh/sshd_config to verify if PermitRootLogin is set to no.

3. **File Permission Check**

    - Checked the file permissions for /etc/passwd (should be 644) and /etc/shadow (should be 640).

4. **Firewall Status Check**

    - Used UFW to check if the firewall is active using:
      sudo ufw status

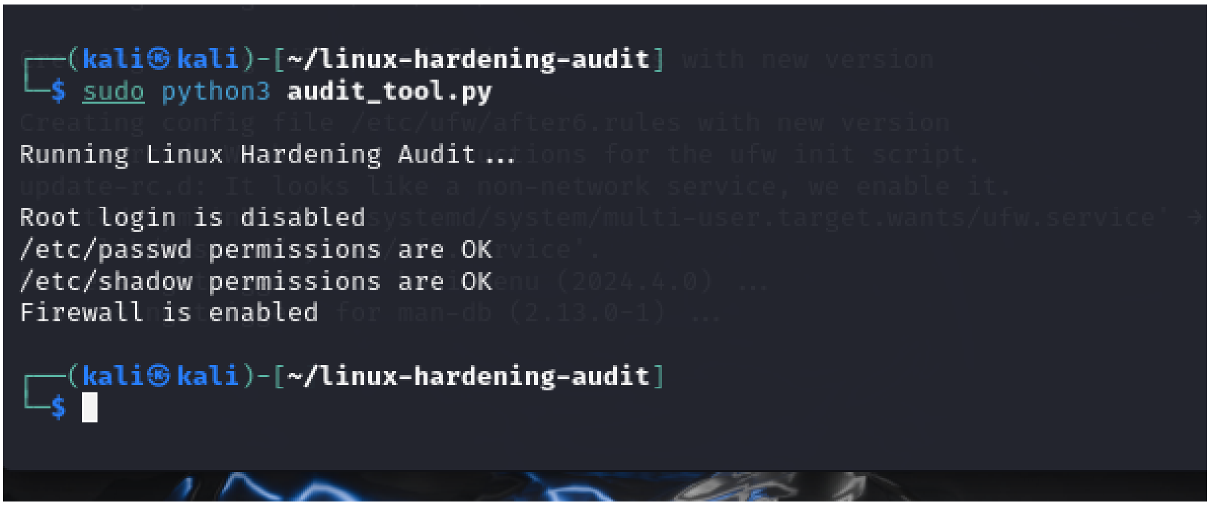
5. **Testing**

    - The script was tested on Kali Linux.

Commands Used

mkdir linux-hardening-audit
cd linux-hardening-audit
nano audit_tool.py
sudo python3 audit_tool.py
sudo apt install ufw
sudo ufw enable
sudo nano /etc/ssh/sshd_config (to set PermitRootLogin no)

Output



```
┌──(kali㉿kali)-[~/linux-hardening-audit]
└─$ sudo python3 audit_tool.py

Running Linux Hardening Audit ...

Root login is disabled
/etc/passwd permissions are OK
/etc/shadow permissions are OK
Firewall is enabled

┌──(kali㉿kali)-[~/linux-hardening-audit]
└─$ ▮
```

Conclusion

This project demonstrates how basic system security checks can be automated using Python. The tool helps identify whether important hardening practices like SSH restrictions, file permission settings, and firewall usage are properly configured. It is suitable for learning, demonstration, and basic auditing purposes. The project also provides a good starting point for extending the script to include more advanced checks.

NAME: SAUMYA MAHESHWARI