Operating System Security Fundamentals

1 Introduction

Operating System (OS) hardening is the process of securing an operating system by reducing vulnerabilities, minimizing the attack surface, and applying security best practices. This practical demonstrates essential OS hardening techniques using Linux and Windows systems.

Installation of Linux Virtual Machine

A Linux virtual machine can be installed using VirtualBox, VMware, or Windows Subsystem for Linux (WSL). Virtualization allows safe experimentation without affecting the host operating system.

Steps

- Install VirtualBox
- Create a new virtual machine
- Attach Linux ISO file
- Complete installation process

Screenshot

Figure 1: Linux Virtual Machine

User Accounts and Access Control

Linux supports multiple users with different privilege levels. Access control ensures that users only perform authorized actions.

1.1 Types of Users

1.1.1 Root User

- Username: `root`
- User ID (UID): 0
- Has full administrative control over the system
- Can read, write, and execute any file
- Dangerous if misused due to unrestricted privileges

1.1.2 Normal Users

- UID: 1000 and above
- Limited privileges compared to the root user
- Used for daily and routine operations

Screenshot

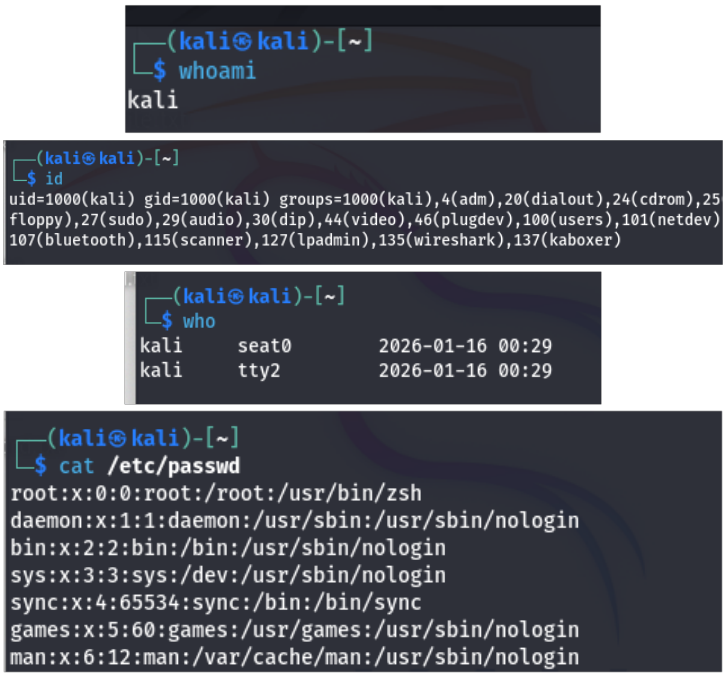


Figure 2: User Accounts and Group Memberships

File Permissions Management

Linux file permissions determine who can read, write, or execute a file.

Breakdown of File Permissions

Part	Meaning
-	File type (- = file, d = directory)
rwx	Owner permissions
r-x	Group permissions
r–	Others permissions

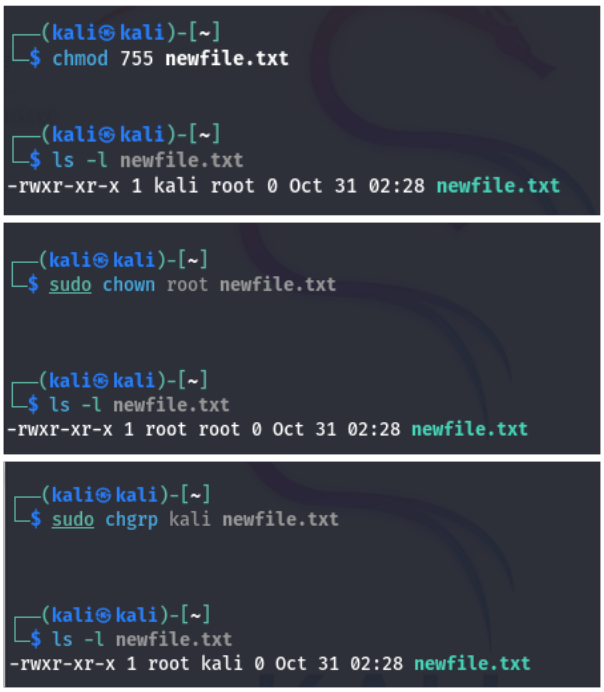
Permission Meanings

- **r** → Read
- **w** → Write
- **x** → Execute

Commands

```
ls -l
chmod 755 filename
chown user:group filename
```

Screenshot



Symbolic	Numeric	Permission
---	0	None
--x	1	Execute
-w-	2	Write
-wx	3	Write + Execute
r--	4	Read
r-x	5	Read + Execute
rw-	6	Read + Write
rwx	7	Read + Write + Execute

Figure 3: File Permission Management using chmod and chown

Administrator vs Standard User Privileges

Administrator users have full control over the system, while standard users have limited access.

1. Administrator (Root) Privileges

Who is the Administrator?

- The **root user** is the system administrator in Linux.
- The root user has a **UID = 0**.
- Root has **unrestricted access** to the entire system.

Capabilities of Root User

- Install and remove software packages.
- Modify critical system files such as `/etc`, `/boot`, and `/usr`.
- Create, modify, and delete users and groups.
- Start, stop, and manage system services.
- Change file permissions and ownership.
- Access and manage all user data on the system.

Screenshot

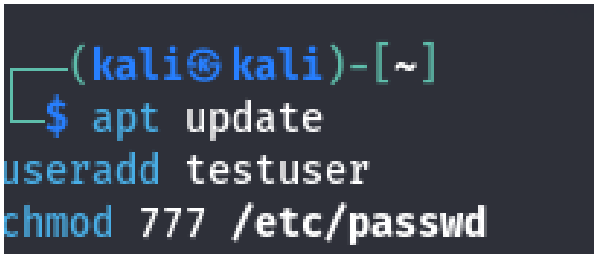


Figure 4: Administrator Privilege using sudo Command

2. Standard User Privileges

Who is a Standard User?

- A standard user is a normal user account (e.g., `kali`).
- Standard users typically have a **UID $\geq$ 1000**.
- They have **limited permissions** by default for system security.

Capabilities of Standard User

- Access their own home directory.
- Run user-level applications.
- Create and modify personal files.
- Cannot modify system files.
- Cannot install system-wide software.
- Cannot manage other users.

Screenshot

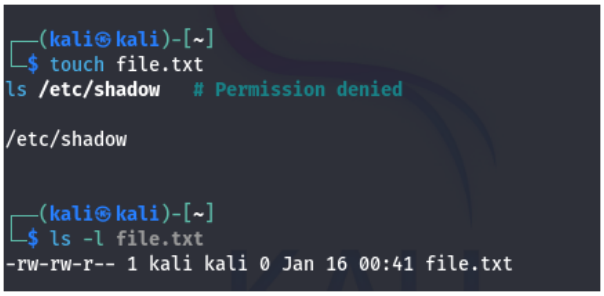


Figure 5: Standard user using sudo Command

Firewall Configuration

Firewalls control incoming and outgoing network traffic.

2 Importance of UFW (Security Perspective)

- Blocks unauthorized network access to the system.
- Reduces the overall attack surface by limiting open ports.
- Protects the system against port scanning and brute-force attempts.
- Essential for securing servers and Kali Linux laboratory environments.

Linux (UFW)

```
sudo ufw enable
sudo ufw status
```

Screenshot

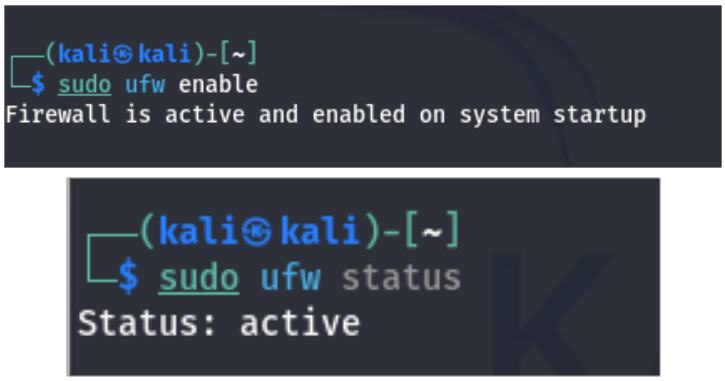


Figure 6: Firewall Enabled using UFW

3 Identifying Running Processes and Services

Monitoring active processes helps detect malicious or unnecessary applications.

Commands

```
ps aux
top
systemctl list-units --type=service
```

Screenshot

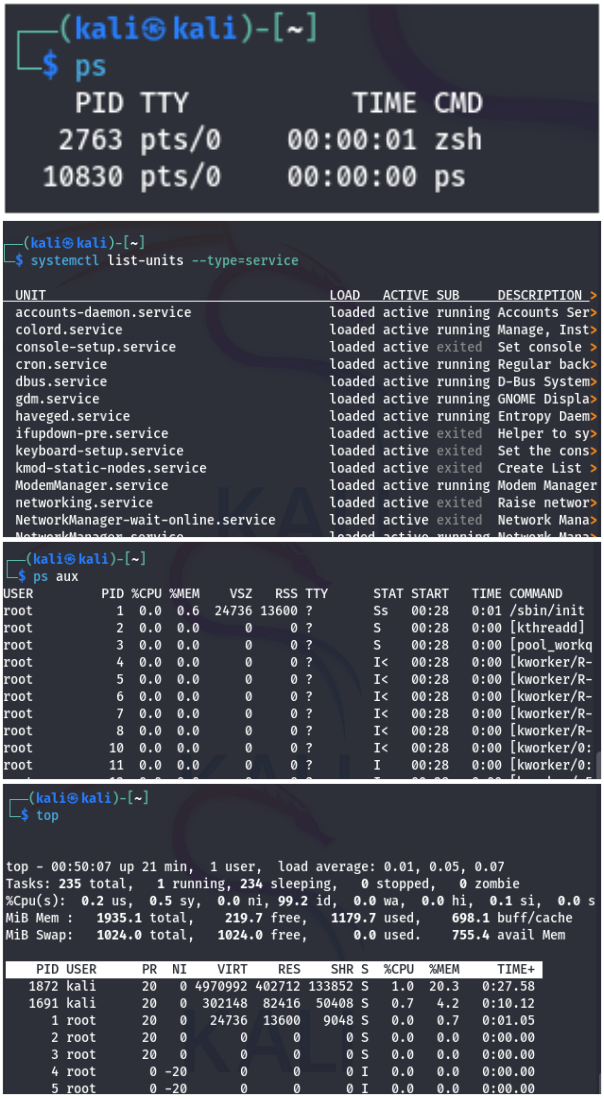


Figure 7: Viewing Running Processes and Services

4 Disabling Unnecessary Services

Unused services increase the attack surface and should be disabled.

Commands

```
sudo systemctl stop servicename
sudo systemctl disable servicename
```

Screenshot

Figure 8: Disabling Unnecessary Services

5 Best Practices for OS Hardening

Applying best practices improves overall system security.

Best Practices

- Keep the system updated
- Use strong passwords
- Disable unused services
- Configure firewalls
- Apply least privilege principle

6 Summary

Operating system hardening is a crucial security practice that helps protect systems from cyber threats. By implementing user control, file permissions, firewall rules, and service management, system security can be significantly improved.