

# 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

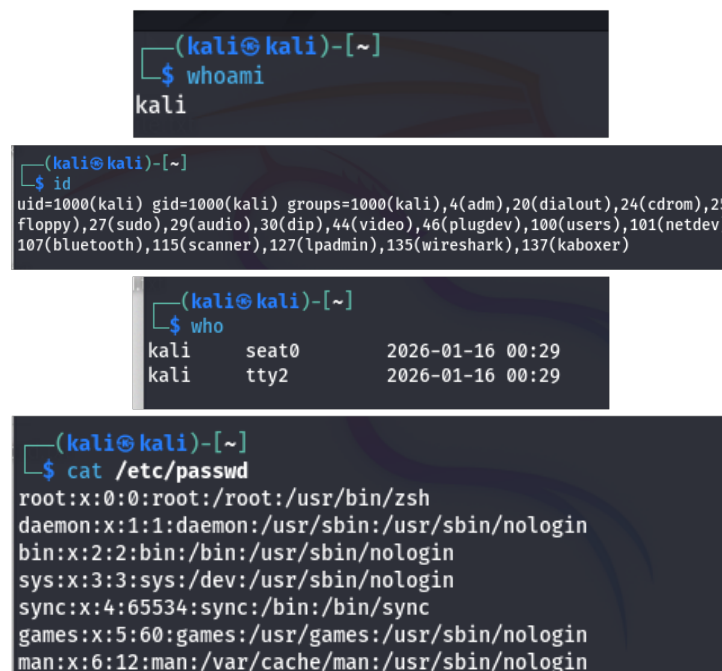
### 1.1.1 Root User

- 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



```
(kali㉿kali)-[~]
$ whoami
kali

(kali㉿kali)-[~]
$ id
uid=1000(kali) gid=1000(kali) groups=1000(kali),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),101(netdev),107(bluetooth),115(scanner),127(lpadmin),135(wireshark),137(kaboxer)

(kali㉿kali)-[~]
$ who
kali    seat0      2026-01-16 00:29
kali    tty2       2026-01-16 00:29

(kali㉿kali)-[~]
$ cat /etc/passwd
root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
```

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

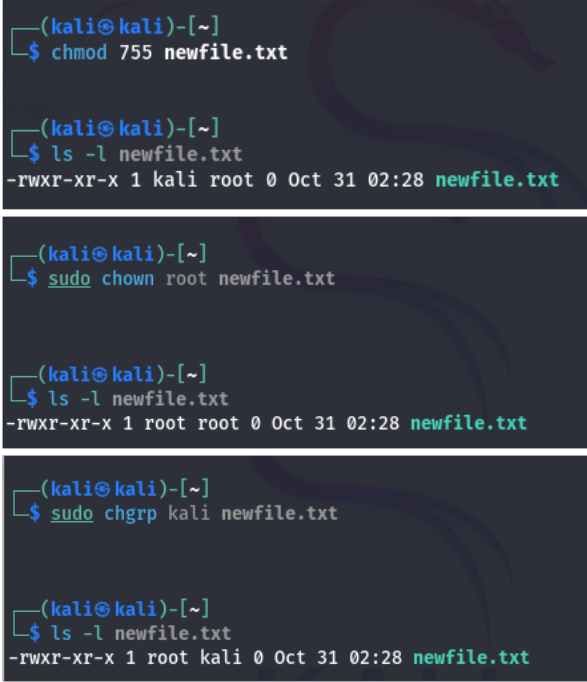
## Permission Meanings

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

## Commands

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

## Screenshot



```
(kali㉿kali)-[~]
$ chmod 755 newfile.txt

(kali㉿kali)-[~]
$ ls -l newfile.txt
-rwxr-xr-x 1 kali root 0 Oct 31 02:28 newfile.txt

(kali㉿kali)-[~]
$ sudo chown root newfile.txt

(kali㉿kali)-[~]
$ ls -l newfile.txt
-rwxr-xr-x 1 root root 0 Oct 31 02:28 newfile.txt

(kali㉿kali)-[~]
$ sudo chgrp kali newfile.txt

(kali㉿kali)-[~]
$ ls -l newfile.txt
-rwxr-xr-x 1 root kali 0 Oct 31 02:28 newfile.txt
```

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

A terminal window with a dark background and light blue text. The prompt is `(kali@kali)-[~]`. The user has entered the following commands: `$ apt update`, `useradd testuser`, and `chmod 777 /etc/passwd`.

```
(kali@kali)-[~]  
$ apt update  
useradd testuser  
chmod 777 /etc/passwd
```

Figure 4: Administrator Privilege using sudo Command

## 2. Standard User Privileges

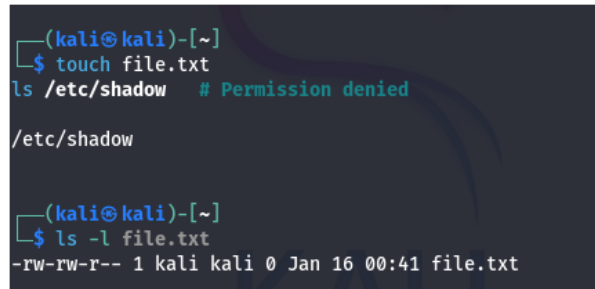
### Who is a Standard User?

- 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



```
(kali㉿kali)-[~]  
$ touch file.txt  
ls /etc/shadow # Permission denied  
  
/etc/shadow  
  
(kali㉿kali)-[~]  
$ ls -l file.txt  
-rw-rw-r-- 1 kali kali 0 Jan 16 00:41 file.txt
```

Figure 5: Standard user using sudo Command

## Firewall Configuration

Firewalls control incoming and outgoing network traffic.

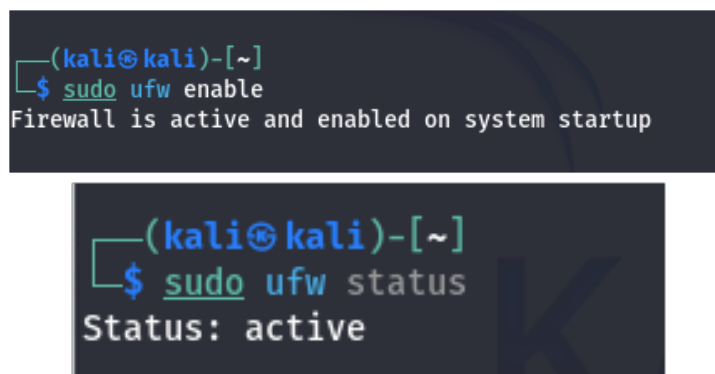
## 2 Importance of UFW (Security Perspective)

- 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



```
(kali㉿kali)-[~]  
$ sudo ufw enable  
Firewall is active and enabled on system startup  
  
(kali㉿kali)-[~]  
$ sudo ufw status  
Status: active
```

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

The figure consists of three terminal screenshots from a Kali Linux machine. The first screenshot shows the output of the `ps` command, displaying a table with columns PID, TTY, TIME, and CMD. The second screenshot shows the output of `systemctl list-units --type=service`, displaying a table with columns UNIT, LOAD, ACTIVE, SUB, and DESCRIPTION. The third screenshot shows the output of the `ps aux` command, displaying a table with columns USER, PID, %CPU, %MEM, VSZ, RSS, TTY, STAT, START, TIME, and COMMAND.

```
(kali@kali)-[~]
$ ps
  PID TTY          TIME CMD
 2763 pts/0        00:00:01 zsh
 10830 pts/0        00:00:00 ps

(kali@kali)-[~]
$ systemctl list-units --type=service
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
accounts-daemon.service            loaded active running Accounts Ser
colord.service                     loaded active running Manage, Inst
console-setup.service              loaded active exited Set console
cron.service                       loaded active running Regular back
dbus.service                       loaded active running D-Bus System
gdm.service                        loaded active running GNOME Displa
haveged.service                   loaded active running Entropy Daem
ifupdown-pre.service               loaded active exited Helper to sy
keyboard-setup.service             loaded active exited Set the cons
kmod-static-nodes.service          loaded active exited Create List
ModemManager.service              loaded active running Modem Manager
networking.service                loaded active exited Raise networ
NetworkManager-wait-online.service loaded active exited Network Mana
NetworkManager.service            loaded active running Network Mana

(kali@kali)-[~]
$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.6 24736 13600 ?        Ss   00:28   0:01 /sbin/init
root         2  0.0  0.0      0     0 ?        S    00:28   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    00:28   0:00 [pool_workq
root         4  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/R-
root         5  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/R-
root         6  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/R-
root         7  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/R-
root         8  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/R-
root        10  0.0  0.0      0     0 ?        I<   00:28   0:00 [kworker/0:
root        11  0.0  0.0      0     0 ?        I    00:28   0:00 [kworker/0:

(kali@kali)-[~]
$ top
top - 00:50:07 up 21 min,  1 user, load average: 0.01, 0.05, 0.07
Tasks: 235 total,  1 running, 234 sleeping,  0 stopped,  0 zombie
%Cpu(s):  0.2 us,  0.5 sy,  0.0 ni, 99.2 id,  0.0 wa,  0.0 hi,  0.1 si,  0.0 st
MiB Mem : 1935.1 total,  219.7 free, 1179.7 used,  698.1 buff/cache
MiB Swap: 1024.0 total, 1024.0 free,  0.0 used,  755.4 avail Mem

   PID USER      PR  NI  VIRT  RES  SHR S  %CPU  %MEM    TIME+
1872  kali     20   0 4970992 402712 133852 S   1.0  20.3   0:27.58
1691  kali     20   0 302148  82416  50408 S   0.7   4.2   0:10.12
   1 root      20   0  24736 13600  9048 S   0.0   0.7   0:01.05
   2 root      20   0      0      0      0 S   0.0   0.0   0:00.00
   3 root      20   0      0      0      0 S   0.0   0.0   0:00.00
   4 root      0 -20      0      0      0 I   0.0   0.0   0:00.00
   5 root      0 -20      0      0      0 I   0.0   0.0   0:00.00
```

Figure 7: Viewing Running Processes and Services

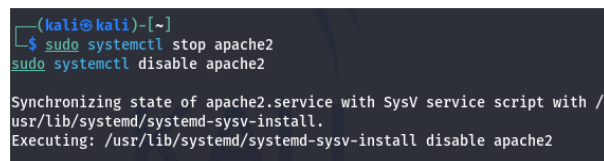
## 4 Disabling Unnecessary Services

Unused services increase the attack surface and should be disabled.

### Commands

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

### Screenshot



```
(kali@kali)-[~]  
└─$ sudo systemctl stop apache2  
sudo systemctl disable apache2  
  
Synchronizing state of apache2.service with SysV service script with /  
usr/lib/systemd/systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install disable apache2
```

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.