Operating System Security & Hardening

Introduction

Operating System (OS) Security refers to the methods and practices used to protect an operating system from unauthorized access, misuse, malware, and cyber attacks. Since the operating system manages hardware, software, files, users, and network access, compromising it can result in total system control by an attacker.

OS Hardening is the process of strengthening an operating system by reducing vulnerabilities. This is achieved by disabling unnecessary services, enforcing strict access controls, configuring firewalls, and applying secure system configurations. A hardened OS significantly reduces the attack surface and improves overall system security.

This document explains OS Security and Hardening concepts using Linux (Ubuntu) and Windows, written in a beginner-friendly manner suitable for academic and internship evaluation.

Why OS Security and Hardening Is Important

- The operating system is the foundation of system security.
- All applications and users depend on the OS for access control.
- Attackers commonly exploit OS vulnerabilities to gain system access.
- Weak OS security can lead to data theft, malware infection, and system crashes.
- OS hardening minimizes risks by limiting exposure to threats.
- Organizations rely on OS security to protect sensitive and confidential data.

Real-Life Example:
If unnecessary services are running with administrator privileges, an attacker can exploit them to gain complete control of the system.

Tools Used (Linux & Windows)

Linux Tools

- **Ubuntu Linux (Primary Tool)**
- Beginner-friendly and widely adopted

- Strong user and permission-based security model

- **Alternative Linux Distributions**

- Kali Linux – Used for security testing
- Fedora – Enterprise-focused Linux distribution

Windows Tools

- **Windows Defender**
- Built-in antivirus and threat protection
- Provides real-time malware detection
- **Windows Security Settings**
- User Account Control (UAC)
- Windows Firewall

These tools help in understanding OS-level security mechanisms across different platforms.

Virtual Machine Setup Explanation

A Virtual Machine (VM) allows one operating system to run inside another operating system in an isolated environment.

Importance of Virtual Machines

- Provides a safe environment for experimentation
- Prevents damage to the host operating system
- Ideal for learning OS security and hardening

Linux Virtual Machine Setup (Using VirtualBox)

- Install VirtualBox on the host system
- Download Ubuntu Linux ISO file
- Create a new virtual machine
- Allocate memory and storage
- Install Ubuntu inside the VM

Windows Alternative

- If VM is not available, use built-in Windows security features
- Explore user accounts, firewall, and process management

Example:
A virtual machine acts like a practice lab where mistakes do not affect the main system.

User Accounts & Access Control

User accounts determine who can access system resources and what actions they can perform.

Linux User Accounts

- Each user has a unique username and user ID
- Users have individual home directories
- Root user has complete system access
- Access is controlled using file permissions

Windows User Accounts

- Administrator accounts have full system privileges
- Standard user accounts have limited permissions
- Managed using User Account Control (UAC)

Importance of Access Control

- Prevents unauthorized access
- Limits damage caused by compromised accounts
- Enforces the principle of least privilege

Real-Life Example:
A guest user should not be able to install applications or modify system settings.

File Permissions (Linux Commands Explained)

Linux controls access to files and directories using permissions.

Types of Permissions

- **Read (r):** View file contents
- **Write (w):** Modify file contents
- **Execute (x):** Run a file or script

Permission Groups

- Owner
- Group
- Others

---

ls -l Command

```
ls -l
```

Explanation: - Displays file permissions and ownership - Shows read, write, and execute permissions - Helps identify access rights for users

---

chmod Command

```
chmod 755 filename
```

Explanation: - Changes file permissions - Owner gets full access (read, write, execute) - Group and others get read and execute access - Prevents unauthorized file modification

---

chown Command

```
chown user:group filename
```

Explanation: - Changes file ownership - Used by administrators - Ensures correct user control over files

Real-Life Example:
Website files should be owned by the web server user, not the root user.

Administrator vs Standard User

Administrator (Root / Admin)

- Full system privileges
- Can install or remove software
- Can modify system settings

Standard User

- Limited permissions
- Cannot modify system files
- Safer for everyday use

Importance of Separation

- Prevents accidental system damage
- Limits malware impact

- Improves overall system security

Firewall Configuration (Linux & Windows)

Firewalls monitor and control network traffic entering and leaving the system.

Linux Firewall (UFW)

Enable Firewall

```
sudo ufw enable
```

Check Firewall Status

```
sudo ufw status
```

Allow a Service

```
sudo ufw allow ssh
```

Explanation: - Blocks unauthorized network access - Allows only trusted services - Protects the system from external attacks

Windows Firewall

- Open Windows Security
- Navigate to Firewall & Network Protection
- Enable firewall for all network profiles

Real-Life Example:
A firewall blocks hackers attempting to access your system over the internet.

Process & Service Management

Processes and services run applications and background tasks.

Linux

```
ps aux
top
```

Windows

- Use Task Manager
- Monitor applications and background services

Importance

- Unnecessary services increase attack surface
- Malware often runs as hidden processes
- Monitoring helps detect suspicious activity

OS Hardening Best Practices

- Disable unused services
- Apply regular updates and patches
- Use strong passwords
- Limit administrator access
- Enable firewall at all times
- Remove unnecessary software
- Monitor system logs
- Perform regular data backups

OS Security Checklist

- Firewall enabled
- Strong passwords configured
- Limited administrator usage
- Proper file permissions applied
- Unused services disabled
- System updates installed
- Antivirus enabled (Windows)
- Regular system monitoring

Simple Real-World Security Scenario

An employee uses an administrator account for daily activities. A malicious email installs malware silently. Because the account has administrator privileges, the malware gains full system access and spreads across the network.

With OS Hardening: - Standard user account blocks installation - Firewall blocks suspicious connections - System damage is minimized

---

Final Outcome / Learning Summary

By completing this task, we gain a strong understanding of operating system security and hardening concepts. We learn how Linux and Windows manage users, permissions, firewalls, and processes. This knowledge helps in protecting systems from cyber threats and is essential for careers in cybersecurity, system administration, and secure application development.