Edversity.

CYBERSECURITY SPECIALIZATION PROGRAM WEEK 4 - DAY 2

Instructor: Muddassir Shafique

TA: Suleiman Taj

Agenda

Week 1: Introduction to Cyber Security and Kali Linux (Completed)
Week 2: Networking and Security Fundamentals
Week 3: Cryptography
Week 4: OS Security
Week 5: Web Security
Week 6: Incident Response and Management
Week 7: Security Policies and Risk Management
Week 8: Emerging Technologies and Future trends
Week 9: CAPSTONE Project and Review
Week 10: CAPSTONE Project Presentations

What We'll Learn Today

- What is OS Hardening?
- Security Features by OS:
- Windows: Windows Defender, Firewall, BitLocker, Security Center
- Kali Linux: AppArmor, unattended upgrades, fail2ban
- macOS: Gatekeeper, FileVault, built-in firewall, XProtect
- Malware examples & defense tools for each OS
- Auditing & Logs: What to look for? Where are they stored?
- Practical (1 Hour):

What is OS Hardening?

Definition

OS Hardening refers to the process of securing an operating system by reducing its attack surface. This means eliminating unnecessary services, tightening configurations, and enforcing strict security controls to prevent unauthorized access or exploitation.

Objective:

- To make the operating system as secure as possible by:
- Minimizing vulnerabilities,
- Reducing potential entry points,
- Limiting what attackers can do even if they gain access.

What is OS Hardening?

Why is OS Hardening Important?

- Prevents exploitation of default settings.
- Makes systems less attractive targets.
- Ensures compliance with security standards (ISO 27001, NIST, etc.).
- Improves resilience against malware, privilege escalation, and remote code execution.

OS Hardening is like locking all the doors and windows of your house, disabling what you don’t need, and adding alarms—before someone tries to break in.

What is OS Hardening?

Key Steps in OS Hardening

Step	Description
Remove Unnecessary Services	Disable unused ports, applications, and services (e.g., FTP, Telnet)
Patch & Update Regularly	Apply the latest OS and software security patches
Strong Authentication	Enforce complex passwords and enable MFA (Multi-Factor Authentication)
Disable Unused User Accounts	Remove or disable guest or default accounts
File Permission Management	Set correct file ownership and restrict access (chmod, NTFS permissions)

What is OS Hardening?

Key Steps in OS Hardening

Step	Description
Logging & Auditing	Enable system logging to monitor and detect suspicious activities
Limit Admin Privileges	Apply the Principle of Least Privilege (PoLP) to all users and services
Secure Boot Settings	Protect BIOS/UEFI with passwords; enable Secure Boot
Application Whitelisting	Only allow approved apps to run (e.g., AppLocker on Windows)

What is OS Hardening?

Examples by Operating System:

OS	Hardening Examples
Windows	Disable SMBv1, enable BitLocker, configure Group Policy, UAC, remove bloatware
Kali/Linux	Harden `ssh` settings, disable root login, configure `iptables`, use `fail2ban`, apply SELinux/AppArmor
macOS	Enable FileVault, configure Gatekeeper, enforce password policy, disable auto-login, apply SIP (System Integrity Protection)

Malware Examples & Defense Tools by OS

Common Malware Examples Windows

Malware	Description
Emotet	Banking Trojan that steals credentials and spreads via phishing
WannaCry	Ransomware that exploited SMBv1 vulnerability to encrypt files
Keyloggers	Record keystrokes to steal passwords and personal data
Trojan Horses	Disguised as legitimate software to gain backdoor access
Rootkits	Hide presence of malware by modifying system files and processes

Malware Examples & Defense Tools by OS

Defense Tools Windows:

Tool	Purpose
Windows Defender	Built-in antivirus, real-time protection
BitLocker	Full disk encryption to protect data at rest
Windows Firewall	Blocks unauthorized inbound/outbound traffic
Microsoft Defender for Endpoint	Enterprise-level malware and threat analytics
Malwarebytes	Popular anti-malware for deeper scans

Malware Examples & Defense Tools by OS

Common Malware Examples Kali Linux / Debian

Malware	Description
Linux.Darlloz	Worm that exploits PHP-CGI vulnerability on Linux systems
EvilGnome	Spyware targeting Linux users via GNOME desktop
Coinminers	Use CPU/GPU resources for crypto mining without user consent
RATs (Remote Access Trojans)	Allow attackers to control system remotely
Rootkits	Alter system binaries to hide malicious processes

Malware Examples & Defense Tools by OS

Defense Tools Linux

Tool	Purpose
ClamAV	Open-source antivirus scanner for Linux
Chkrootkit / Rkhunter	Detect rootkits and hidden processes
AppArmor / SELinux	Mandatory access control systems to limit process behavior
iptables / ufw	Configure firewall rules to control traffic
fail2ban	Blocks IPs after repeated failed login attempts (brute force defense)

Malware Examples & Defense Tools by OS

Common Malware Examples MaC

Malware	Description
Shlayer	Most widespread macOS malware, installs adware via fake Flash updates
Flashback	Trojan horse that infected 600,000 Macs via Java exploit
MacStealer	Harvests iCloud Keychain, files, and browser cookies
XCSSET	Injects malicious code into Xcode projects to infect developers
KeRanger	One of the first ransomware found on macOS

Malware Examples & Defense Tools by OS

Defense Tools MaC

Tool	Purpose
XProtect	Built-in malware signature scanner (auto-updated by Apple)
Gatekeeper	Prevents unverified apps from running
System Integrity Protection (SIP)	Blocks modification of system-critical files
FileVault	Encrypts the entire drive to protect user data
Little Snitch	Third-party firewall to monitor outgoing network connections

Malware Examples & Defense Tools by OS

Summary

OS	Malware Example	Defense Tool
Windows	Emotet (banking trojan)	Windows Defender, BitLocker
Linux	EvilGnome (spyware)	ClamAV, AppArmor, fail2ban
macOS	Shlayer (adware downloader)	XProtect, Gatekeeper, SIP

Auditing & Logs

What to Look for and Where They’re Stored

What is Auditing?

Auditing is the process of tracking and recording events or activities on an operating system to detect suspicious behavior, troubleshoot problems, or ensure policy compliance.

What Are Logs?

Logs are time-stamped records of system events, user actions, and application behaviors. They are essential for monitoring, incident response, and forensics.

Auditing & Logs

What Should You Look For in Logs?

🔍 Event Type	⚠️ What to Look For
Logins/Logouts	Unusual login times, failed login attempts, multiple logins from the same IP
Privilege Escalation	A normal user suddenly running admin/root commands
New User/Group Creation	Suspicious user additions or changes to admin groups
Service Changes	Disabled/enabled firewall, antivirus, SSH, etc.
File Access/Modifications	Access to critical system files or logs being cleared
Remote Access Attempts	SSH, RDP, or VPN usage outside approved times/IPs
Software Installations	Unexpected applications or tools installed

Auditing & Logs

Where Are Logs Stored?

Location:

Logs are stored in the Event Viewer under various categories:

path: Control Panel > Administrative Tools > Event Viewer

Log Types

Log Name	Purpose
Application	Errors or logs from installed apps
Security	Login attempts, privilege use, account creation
System	OS events, driver failures, service starts/stops
Setup	System installation logs
Forwarded Events	Logs collected from other machines

Auditing & Logs

Where Are Logs Stored?

Log Directory in Kali: /var/log/

Commands:

- sudo cat /var/log/auth.log | grep "sudo"
- sudo tail -f /var/log/syslog

Imp Log Files

File	Purpose
`/var/log/auth.log`	Logins, sudo usage, SSH access attempts
`/var/log/syslog` or `/var/log/messages`	General system activity
`/var/log/boot.log`	Boot process logs
`/var/log/kern.log`	Kernel-related issues
`/var/log/faillog`	Failed login attempts

Auditing & Logs

Where Are Logs Stored?

Location in macOS:

- Log Viewer App: Use Console.app: Applications > Utilities > Console

Log Files (Also stored in /var/log/):

File	Purpose
`/var/log/system.log`	System-level events
`/var/log/install.log`	App installations
`/var/log/asl/`	Apple System Logs
`~/Library/Logs/`	User-specific app logs

Commands

- log show --predicate 'eventMessage contains "login"' --info
- tail -f /var/log/system.log

Auditing & Logs

Enable Advanced Auditing

OS	How
Windows	Use `secpol.msc` → Advanced Audit Policy Configuration
Linux	Install and configure `auditd` (`/etc/audit/audit.rules`)
macOS	Use `OpenBSM` (Basic Security Module auditing) via `/etc/security/audit_control`

Why Logs Matter

- Detect Attacks: Failed login attempts, privilege misuse.
- Ensure Compliance: Who accessed what and when?
- Troubleshooting: Identify cause of system/application failure.
- Digital Forensics: Logs can be used as legal evidence in breaches.

Practical

What We Learned Today

- What is OS Hardening?
- Security Features by OS:
- Windows: Windows Defender, Firewall, BitLocker, Security Center
- Kali Linux: AppArmor, unattended upgrades, fail2ban
- macOS: Gatekeeper, FileVault, built-in firewall, XProtect
- Malware examples & defense tools for each OS
- Auditing & Logs: What to look for? Where are they stored?
- Practical (1 Hour):