Lab 6: System Hardening and Reducing Attack Surface

Instructor: Dr. Maryam R. Aliabadi
Lab Duration: 2 hours
Date: Oct 16th, 2025

Overview

In this lab, you will apply system hardening techniques to secure a Linux environment. You'll disable unused services, apply patches, enforce access control, and test the system for vulnerabilities. By the end, you should have a more secure and efficient system with minimized attack surface.

Learning Objectives

- Understand and apply system hardening principles.
- Identify and disable unnecessary or insecure services.
- Apply security patches and updates using software management tools.
- Configure user permissions and security policies to minimize risk.
- Verify and validate security configurations.
- Perform footprinting and network monitoring using nmap and tcpdump.

Lab Setup

1. Start your Linux virtual machine (Ubuntu or CentOS preferred).

2. Ensure you have administrative (sudo) privileges.

4. Ensure tools installed: nmap, tcpdump, tshark, ss, netstat, whois, dig, ufw, fail2ban, lynis.

Install if missing (Ubuntu example):

```
sudo apt install nmap tcpdump tshark ufw fail2ban lynis -y
```

Lab Instructions

Part A: Baseline Assessment (Before Hardening)

• List all active services and open ports:

```
sudo systemctl list-units --type=service --state=running

ss -tuln
```

• Note any unexpected services (e.g., avahi-daemon, cups, bluetoothd).

Part B: Footprinting (Local and External)

• Local footprinting with nmap (scan localhost):

```
nmap -sS -Pn -p- localhost

nmap -sV -O localhost
```

• Targeted scans for specific ports and service detection:

```
nmap -p 22,80,443 --script=banner -sV <target-ip>
```

• Remote footprinting (if allowed): use whois, dig, nslookup to gather public info:

```
whois example.com

dig +short example.com ANY
```

• Interpreting nmap results:
- Identify open ports and associated services.
- Note service versions; outdated versions may indicate vulnerabilities.
- OS detection (-O) can reveal exposure from banner information.

Challenge 1: From the nmap output, list three findings that suggest a remediation action (e.g., close port, update service).

Part C: Network Monitoring with tcpdump and tshark

• Basic tcpdump live capture (capture 1000 packets to file):

```
sudo tcpdump -i any -c 1000 -w /tmp/capture.pcap
```

• Capture only traffic to/from SSH (port 22):

```
sudo tcpdump -i any port 22 -w /tmp/ssh_capture.pcap
```

• Capture HTTP traffic (port 80) and print readable output:

```
sudo tcpdump -i any tcp port 80 -A
```

• Read pcap with tshark (CLI Wireshark):

```
tshark -r /tmp/capture.pcap -q -z io,stat,0,COUNT,PACKETS
```

• Filter examples with tcpdump:

```
sudo tcpdump -i any 'tcp and (src host 10.0.0.5) and (dst port 80)' -w
/tmp/filtered.pcap
```

• Extract specific fields with tshark:

```
tshark -r /tmp/capture.pcap -T fields -e ip.src -e ip.dst -e tcp.srcport -e
tcp.dstport
```

Challenge 2: Use tcpdump/tshark to capture a short login attempt to SSH; identify client IP, timestamps, and any failed authentication packet sequences.

Part D: Hardening Steps (Disable services, Patch, Firewall)

• Stop and disable unnecessary services:

```
sudo systemctl stop <service>

sudo systemctl disable <service>

systemctl is-enabled <service>
```

• Apply patches / updates:

```
sudo apt update && sudo apt upgrade -y

sudo yum update -y
```

• Check kernel version:

```
uname -r
```

• Configure firewall (example ufw):

```
sudo ufw enable

sudo ufw allow ssh

sudo ufw status verbose
```

Part E: Post-hardening Validation (Repeat Scans and Captures)

• Repeat nmap scans used in Part B and compare results.

```
nmap -sS -Pn -p- localhost

nmap -sV -O localhost
```

• Re-run tcpdump capture while re-testing connectivity; compare packet traces to identify removed services.

Challenge 3: Show a before-and-after comparison (nmap output and pcap snippets) that proves the attack surface reduction.

Part F: Automation and Monitoring Script

• Create a bash script hardening_check.sh that performs:

```
#!/bin/bash

echo "Hardening check - $(date)" > /var/log/hardening_check.log

ss -tuln >> /var/log/hardening_check.log

sudo ufw status >> /var/log/hardening_check.log

sudo apt list --upgradable 2>/dev/null >> /var/log/hardening_check.log
```

• Schedule it daily with cron: `sudo crontab -e` and add:

```
0 2 * * * /path/to/hardening_check.sh
```

Part G: Analysis and Reporting

• Prepare a short report (2-3 pages) including:
- Baseline nmap output and identified risks
- tcpdump/tshark captures demonstrating exposed services or attack patterns
- Detailed list of disabled services and rationale
- Before-and-after comparisons (nmap and pcap evidence)
- Automation script and cron entry

Deliverables & Submission

Submit the following:
- Lab report (PDF or DOCX) with findings and screenshots
- Baseline and post-hardening nmap outputs
- One or two representative pcap files (or excerpts) with explanation
- hardening_check.sh script and cron line
- List of disabled services and justification

- submit your report with the following name format through the Learning Hub.

Filename: Lab6-FirstName-Lastname-StdNo.PDF

Good luck!