

Class activity

A hands-on exercise from the class-activities appendix. The full text also appears in Appendix A of the book.

Accompanies *Computer Security: The Foundations*, Krerk Piromsopa, Ph.D., 2026.
<https://www.cp.eng.chula.ac.th/~krerk/books/ComputerSecurity/>

Authorized use only

These exercises involve real attack techniques. Carry them out only against systems you own, or that you have written permission from the owner to test. Unauthorized access, scanning or interception is a criminal offence under Thailand's Computer Crime Act and its equivalents elsewhere, whether or not any damage results. If you are unsure whether a target is in scope, ask your instructor before you start.

Activity 8: Network Scanner with NMAP

Overview

NMAP ("Network Mapper") is an open-source tool for network exploration and security auditing. It is widely used by system administrators to inventory hosts and services, and by security professionals to identify open ports, detect operating systems, and run diagnostic scripts.

Ethical and legal notice. *Scanning networks or hosts without explicit permission from their owner is illegal in most jurisdictions and may violate institutional policy. Always scan only systems and networks you own or have written authorization to test. For all exercises below, use your own machine (localhost) and the class Wi-Fi segment provided by your instructor.*

Installation. Install nmap from your system package manager or from <https://nmap.org/book/install.html>.

- **Debian/Ubuntu:** `sudo apt install nmap`
- **macOS:** `brew install nmap`
- **Windows:** download the installer from <https://nmap.org/download.html> (Zenmap GUI included)

Verify with `nmap --version`.

Exercises

1. **Host Discovery.** Connect to your university network (e.g. ChulaWIFI or eduroam).

- (a) **List scan** — resolves names but sends *no packets* to the targets:

```
$ nmap -sL 10.201.3.0/24
```

What does `-sL` do? Does it contact any hosts?

- (b) **Ping scan** — ICMP only, no port scan:

```
$ nmap -v -sn 10.201.3.0/24
```

(Replace 10.201.3.0 with your actual subnet.) How many hosts responded? How does -sn differ from a full port scan?

- (c) If a host appears down but you know it is up, try:

```
$ nmap -Pn 10.201.3.250
```

What does -Pn do? When would you use it?

2. Port Scanning Techniques. Use localhost (127.0.0.1) for all scans in this exercise.

- (a) **TCP SYN scan** (“half-open”; requires root):

```
$ sudo nmap -sS 127.0.0.1
```

Why is this called a “half-open” scan? Does it complete the TCP three-way handshake?

- (b) **TCP connect scan** (no root required):

```
$ nmap -sT 127.0.0.1
```

How does this differ from the SYN scan? Which leaves more traces in server logs, and why?

- (c) **UDP scan** on common service ports:

```
$ sudo nmap -sU -p 53,67,68,123,161 127.0.0.1
```

Why is UDP scanning slower than TCP scanning? What services do these five port numbers typically correspond to?

- (d) Compare the scan time using timing templates:

```
$ nmap -T2 127.0.0.1    # Polite
$ nmap -T4 127.0.0.1    # Aggressive
```

What trade-off does the -T template control? When would you deliberately choose a slower template?

3. Service and Version Detection. An open port reveals only that something is listening. NMAP can probe services to identify the software and version.

- (a) Detect service versions on your machine:

```
$ nmap -sV 127.0.0.1
```

List every open port and the software version reported.

- (b) Run an aggressive scan (OS detection + version + scripts + traceroute):

```
$ sudo nmap -A 127.0.0.1
```

What additional information does -A produce compared with -sV alone?

- (c) Scan specific ports only:

```
$ nmap -p 22,80,443,3306,5432 -sV 127.0.0.1
```

Are any database ports (MySQL port 3306, PostgreSQL port 5432) open on your machine? Should they be reachable from outside?

4. OS Fingerprinting. NMAP analyzes TCP/IP stack behavior to infer the remote operating system.

- (a) With a classmate’s permission, scan their machine:

```
$ sudo nmap -O 10.201.3.250
$ sudo nmap -v --osscan-guess 10.201.3.250
```

What OS and version does NMAP report? How confident is it?

- (b) Try at least three machines running different OSes (Windows, Linux, macOS, a smart-phone). Which is easiest to identify?
- (c) Explain the principle behind OS fingerprinting: what characteristics of the TCP/IP stack does NMAP measure to distinguish operating systems?

5. NMAP Scripting Engine (NSE). NSE allows Lua scripts to automate tasks from banner grabbing to vulnerability detection.

- (a) Run the *default* script category:

```
$ nmap -sC 127.0.0.1
```

(-sC is equivalent to --script=default.) What extra information do the scripts provide over a plain port scan?

- (b) Retrieve HTTP server headers from a running web server:

```
$ nmap --script=http-headers 127.0.0.1
```

What headers are returned? Are any security-relevant (e.g. Server, X-Powered-By, Strict-Transport-Security)?

- (c) Run the vulnerability category against *your own machine only*:

```
$ sudo nmap --script=vuln 127.0.0.1
```

Warning: vuln scripts actively probe services and may trigger IDS alerts. Only use on machines you own. What CVEs or potential issues, if any, are reported?

- (d) Browse the NSE script library at <https://nmap.org/nsedoc/>. Find one script relevant to a service open on your machine. Describe what it does and run it.

6. Output Formats and Reporting.

- (a) Save results in all three formats simultaneously:

```
$ sudo nmap -A -oA scan-results 127.0.0.1
```

This creates scan-results.nmap (normal), scan-results.xml (XML), and scan-results.gnmap (grepable).

- (b) Extract only open ports from the grepable output:

```
$ grep "/open/" scan-results.gnmap
```

List all responding IP addresses:

```
$ grep "^Host:" scan-results.gnmap | awk '{print $2}'
```

- (c) Convert the XML output to an HTML report:

```
$ xsltproc scan-results.xml -o scan-results.html
```

Open the HTML file in a browser.

- (d) When is each format most useful: .nmap, .xml, and .gnmap? Give a concrete use case for each.

7. Service Discovery on a Network.

- (a) Find all machines with open HTTP or HTTPS ports:

```
$ nmap -p 80,443 --open 10.201.3.0/24
```

- (b) Find all machines with an open SSH port:

```
$ nmap -p 22 --open 10.201.3.0/24
```

- (c) Write the nmap command to find all open FTP servers on the campus network. Explain each flag you use.
- (d) Run a service-version scan on one of the web servers found in (a). What software and version is it running? Is the version current?

Refer to <https://nmap.org/book/man.html> for the full reference manual.

8. Reflection.

- (a) Run a full aggressive scan on your own machine and review every line of output. Were any open ports or services unexpected? If yes, investigate what they are and whether they should be open.
- (b) From the perspective of a network defender running nmap against your own network:
- What results would be *expected*?
 - What results would be *unexpected* and should trigger immediate investigation?
- (c) A host runs SSH on port 2222 instead of 22. Does moving a service to a non-standard port improve security? Justify your answer.

From *Computer Security: The Foundations*, Kerk Piromsopa, Ph.D.
Reproduce and adapt freely for non-commercial classroom instruction, with attribution. Full terms: LICENSE.txt in the student package.