The Ethical Hacker's Command Handbook

Chapter 2 – Nmap (Network Mapper)

Practical Command Reference

Nmap is one of the first tools used during an authorized security assessment. This chapter focuses on the commands you are most likely to use, when to use them, and what to run next.

nmap

Purpose: Basic scan

What it does: Discover common TCP ports.

When to use: First look at a target.

Avoid when: Need only host discovery.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sn

Purpose: Ping scan

What it does: Check whether hosts are online without scanning ports.

When to use: Finding live hosts on a subnet.

Avoid when: Need open ports.

Next command: -sS

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-Pn

Purpose: Skip ping

What it does: Treat the host as online.

When to use: Target blocks ICMP (common in labs).

Avoid when: Large networks (slower).

Next command: -sS

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sS

Purpose: TCP SYN scan

What it does: Fast half-open TCP scan.

When to use: Default TCP enumeration.

Avoid when: No admin/root privileges.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sT

Purpose: TCP Connect scan

What it does: Full TCP connect.

When to use: When SYN scan isn't available.

Avoid when: Root access is available.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sU

Purpose: UDP scan

What it does: Scan UDP ports.

When to use: DNS, SNMP, NTP checks.

Avoid when: Only TCP services.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sV

Purpose: Version detection

What it does: Identify service versions.

When to use: Research vulnerabilities.

Avoid when: Need only host discovery.

Next command: Searchsploit

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-sC

Purpose: Default NSE scripts

What it does: Run safe default scripts.

When to use: Initial service enumeration.

Avoid when: Need custom scripts only.

Next command: --script

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

--script vuln

Purpose: Vulnerability scripts

What it does: Run vulnerability checks.

When to use: After identifying services.

Avoid when: Very noisy environments.

Next command: Searchsploit

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-O

Purpose: OS detection

What it does: Guess operating system.

When to use: Need OS fingerprint.

Avoid when: Firewall blocks fingerprints.

Next command: -A

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-A

Purpose: Aggressive scan

What it does: Enable OS, version, scripts, traceroute.

When to use: Small lab machines.

Avoid when: Large networks.

Next command: Save output

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-p-

Purpose: All TCP ports

What it does: Scan every TCP port.

When to use: Thorough assessment.

Avoid when: Need quick top-port scan.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

--top-ports 100

Purpose: Top ports

What it does: Scan most common ports.

When to use: Quick reconnaissance.

Avoid when: Need complete coverage.

Next command: -sV

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-T4

Purpose: Faster timing

What it does: Speed up scanning.

When to use: Typical lab environments.

Avoid when: Unstable/slow networks.

Next command: Save output

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

-oA scan

Purpose: Save all formats

What it does: Normal, XML and grepable output.

When to use: Every serious assessment.

Avoid when: One-off experiments.

Next command: Review results

Memory Tip: Learn where this command fits in your workflow, not just its syntax.

Quick Workflow

1. Host discovery (-sn or -Pn if ping is blocked)
2. TCP scan (-sS)
3. Version detection (-sV)
4. Default scripts (-sC)
5. Research findings
6. Save results (-oA)

One-Page Revision

-sn Find hosts
-Pn Ignore ping
-sS Main TCP scan
-sV Versions
-sC Default scripts
-O OS detection
-A Aggressive scan
-p- All ports
-T4 Faster scan
-oA Save results

The Ethical Hacker's Command Handbook

Chapter 2 – Part 2: Port Scanning

Learn what to scan, why to scan it, and when to use each Nmap scan type.

Learning Objectives

Understand ports, TCP vs UDP, the TCP three-way handshake, SYN, Connect and UDP scans, and learn when each scan is appropriate.

What is a Port?

A port is a logical communication endpoint. An IP address identifies the host, while the port identifies the service running on that host (for example SSH on 22 or HTTP on 80).

Why Scan Ports?

Port scanning helps identify exposed services, understand the target's attack surface, and plan the next stage of an authorized security assessment.

TCP vs UDP

TCP is connection-oriented and reliable. **UDP** is connectionless and faster but does not guarantee delivery. Most penetration testing begins with TCP because most common services use it.

TCP Three-Way Handshake

Client → SYN → Server → SYN/ACK → Client → ACK. This handshake establishes a TCP connection. Some scan types intentionally stop before the final ACK.

sudo nmap -sS

Purpose: TCP SYN (half-open) scan

What it does: Default TCP scan when you have administrator/root privileges.

When to use: Fast initial TCP enumeration in most labs.

Avoid when: When you lack administrator/root privileges.

Next Step: Next: -sV then -sC

Memory Tip: S = SYN

nmap -sT

Purpose: TCP Connect scan

What it does: Completes the full TCP handshake.

When to use: When running as a normal user without raw packet privileges.

Avoid when: When -sS is available with appropriate privileges.

Next Step: Next: -sV

Memory Tip: T = TCP Connect

sudo nmap -sU

Purpose: UDP scan

What it does: Scans UDP ports such as DNS (53), NTP (123), and SNMP (161).

When to use: After TCP scans or when UDP services are expected.

Avoid when: When only a quick TCP assessment is required.

Next Step: Next: Investigate detected UDP services.

Memory Tip: U = UDP

Comparison Table

Scan	Speed	Privileges	Typical Use
-sS	Fast	Root/Admin	Primary TCP enumeration
-sT	Moderate	None	When -sS isn't available
-sU	Slow	Root/Admin recommended	Discover UDP services

Quick Workflow

Target → **-sS** → **-sV** → **-sC** → Research findings → Document results

Revision Notes

• Port = Service endpoint
• TCP = Reliable
• UDP = Faster, connectionless
• -sS = Main TCP scan
• -sT = Use without admin/root privileges
• -sU = Scan UDP services

Practice Questions

1. Why is -sS called a half-open scan?
2. When would you choose -sT over -sS?
3. Name three common UDP services.
4. What command usually follows a successful -sS scan?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 3: Port Selection & Service Version Detection

Knowing which ports to scan and identifying the services running on them are key steps in authorized network enumeration.

Learning Objectives

Learn how to scan specific ports, multiple ports, ranges, all TCP ports, perform fast scans, use top-port scans, detect service versions, and understand when each option is most appropriate.

Why Port Selection Matters

By default, Nmap scans the 1,000 most common TCP ports. While this is efficient, important services may run on uncommon ports, making targeted or full-port scans necessary in some assessments.

nmap -p 80

Purpose: Scan one specific port

What it does: Checks only the selected port.

When to use: When verifying a known service or troubleshooting.

Avoid when: Initial reconnaissance when you don't know which ports are open.

Memory Tip: Remember: -p = Port.

nmap -p 22,80,443

Purpose: Scan multiple ports

What it does: Scans selected ports only.

When to use: Web server and SSH assessments.

Avoid when: When full enumeration is required.

Memory Tip: Separate ports with commas.

nmap -p 1-1000

Purpose: Scan a port range

What it does: Scans a continuous range of ports.

When to use: Focused assessments of lower-numbered ports.

Avoid when: When uncommon ports may be in use.

Memory Tip: Use ranges to save time.

nmap -p-

Purpose: Scan all TCP ports

What it does: Scans all 65,535 TCP ports.

When to use: CTFs, TryHackMe, thorough assessments.

Avoid when: Quick reconnaissance.

Memory Tip: Hyphen means all ports.

nmap -F

Purpose: Fast scan

What it does: Scans fewer ports than the default.

When to use: Quick overview of a host.

Avoid when: When comprehensive results are required.

Memory Tip: F = Fast.

nmap --top-ports 100

Purpose: Top ports scan

What it does: Scans the most common ports.

When to use: Large environments or time-limited assessments.

Avoid when: When unusual ports are suspected.

Memory Tip: Choose the number based on your goal.

nmap -sV

Purpose: Version detection

What it does: Identifies software and service versions.

When to use: Immediately after discovering open ports.

Avoid when: When only checking host availability.

Memory Tip: V = Version.

nmap -sV --version-light

Purpose: Light version detection

What it does: Uses fewer probes for faster results.

When to use: When speed is more important than detail.

Avoid when: When exact versions are critical.

Memory Tip: Faster but less thorough.

nmap -sV --version-all

Purpose: Full version detection

What it does: Uses additional probes for better identification.

When to use: When accurate version information is essential.

Avoid when: Quick reconnaissance.

Memory Tip: More accurate but slower.

Typical Enumeration Workflow

1. Scan ports (-sS)
2. If necessary, scan all ports (-p-)
3. Detect service versions (-sV)
4. Research discovered services
5. Continue with authorized enumeration using appropriate NSE scripts.

Command Summary

Goal	Command
One port	-p 80
Multiple ports	-p 22,80,443
Port range	-p 1-1000
All ports	-p-
Fast scan	-F
Top ports	--top-ports 100
Version detection	-sV
Fast version detection	--version-light
Detailed version detection	--version-all

Revision Notes

• -p → Specific ports
• -p- → All TCP ports
• -F → Fast scan
• --top-ports → Most common ports
• -sV → Version detection
• --version-light → Faster version detection
• --version-all → More detailed version detection

Practice Questions

1. What is the difference between -p 80 and -p-?
2. Why might a default scan miss an important service?
3. When would you use -F?
4. What information does -sV provide?
5. When is --version-all preferable to the default -sV?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 4: Nmap Scripting Engine (NSE)

Automate enumeration with Nmap's built-in Lua scripts.

Learning Objectives

Understand what NSE is, use default and custom scripts, search for scripts, learn common categories, and know when each script is appropriate during authorized security assessments.

What is NSE?

NSE (Nmap Scripting Engine) extends Nmap using Lua scripts to automate information gathering and selected security checks. Instead of only showing open ports, scripts can retrieve banners, web page titles, SMB details, SSL information, and more.

Why Use NSE?

After discovering open ports and services, NSE helps collect additional information that can guide the next stage of enumeration or validation.

nmap -sC

Purpose: Default scripts

What it does: Runs Nmap's default script set.

When to use: After identifying open services.

Avoid when: When only a very quick scan is needed.

Short Note: Next: Review results and research.

nmap --script http-title

Purpose: HTTP title

What it does: Retrieves the title of a web page.

When to use: When HTTP/HTTPS is open.

Avoid when: No web service detected.

Short Note: Useful for identifying applications.

nmap --script ftp-anon

Purpose: FTP anonymous

What it does: Checks for anonymous FTP access.

When to use: When FTP (21) is open.

Avoid when: FTP not present.

Short Note: Verify access is authorized.

nmap --script smb-os-discovery

Purpose: SMB OS discovery

What it does: Collects SMB OS and host information.

When to use: When SMB (445) is open.

Avoid when: SMB unavailable.

Short Note: Useful in Windows environments.

nmap --script ssh-hostkey

Purpose: SSH host key

What it does: Retrieves SSH public host keys.

When to use: When SSH (22) is open.

Avoid when: SSH unavailable.

Short Note: Useful for fingerprinting.

nmap --script vuln

Purpose: Vulnerability category

What it does: Runs scripts in the vuln category.

When to use: After identifying services.

Avoid when: Outside the authorized scope.

Short Note: Validate findings before conclusions.

nmap --script smb-vuln-ms17-010 -p445

Purpose: MS17-010 check

What it does: Checks for indicators of the MS17-010 SMB vulnerability.

When to use: When Windows SMB is exposed.

Avoid when: Port 445 closed.

Short Note: Research results before next steps.

sudo nmap --script-updatedb

Purpose: Update script database

What it does: Rebuilds the local NSE script database.

When to use: After adding or modifying scripts.

Avoid when: Routine scans.

Short Note: Keeps local script index current.

Finding Scripts

List scripts: ls /usr/share/nmap/scripts
Search: ls /usr/share/nmap/scripts | grep smb

NSE Categories

Category	Purpose
default	Common safe scripts
safe	Low-impact checks
discovery	Gather information
auth	Authentication-related checks
brute	Credential guessing scripts
vuln	Vulnerability checks
malware	Malware detection
exploit	Exploit-related scripts

Workflow

Discover host → Scan ports → Detect versions → Run -sC → Run targeted scripts → Research → Document findings

Revision Notes

• -sC = Default scripts
• --script = Specific script
• http-title = Page title
• ftp-anon = Anonymous FTP check
• smb-os-discovery = SMB information
• ssh-hostkey = SSH host keys
• --script vuln = Vulnerability scripts
• --script-updatedb = Refresh script database

Practice Questions

What is the difference between -sC and --script?

When would you use http-title?

Why should services be identified before running targeted NSE scripts?

What does smb-os-discovery provide?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 5: OS Detection & Advanced Detection Techniques

Learn how Nmap estimates operating systems using TCP/IP fingerprinting.

Learning Objectives

Understand OS detection, fingerprinting, OS scan options, accuracy, limitations, workflows, and best practices.

What is OS Detection?

OS detection estimates the target operating system by comparing network responses to Nmap's fingerprint database. It helps identify platforms such as Windows, Linux, BSD, or network appliances.

Why It Matters

Knowing the operating system helps guide enumeration, service analysis, and vulnerability research during authorized security assessments.

sudo nmap -O

Purpose: Detect operating system

What it does: Compares probe responses with known fingerprints.

When to use: After discovering open ports.

Avoid when: Only checking host availability.

Memory Tip: O = Operating System.

sudo nmap --osscan-limit

Purpose: Limit OS detection

What it does: Attempts OS detection only on suitable hosts.

When to use: Large network scans.

Avoid when: Small single-host scans where speed is not a concern.

Memory Tip: Reduces unnecessary probes.

sudo nmap --osscan-guess

Purpose: Best OS guess

What it does: Shows closest fingerprint match when confidence is low.

When to use: Lab environments or uncertain results.

Avoid when: When exact identification is required.

Memory Tip: Guess = Best estimate.

sudo nmap -O -sV

Purpose: OS + Version detection

What it does: Combines OS fingerprinting with service version detection.

When to use: Most detailed manual enumeration.

Avoid when: Quick host discovery.

Memory Tip: Versions support OS analysis.

sudo nmap -A

Purpose: Aggressive scan

What it does: Runs OS detection, version detection, default NSE scripts, and traceroute.

When to use: Small lab machines after initial enumeration.

Avoid when: Large networks or quick reconnaissance.

Memory Tip: Use after basic scans.

Why OS Detection Can Fail

Common reasons include firewall filtering, packet loss, VPNs, custom TCP/IP stacks, and the absence of both open and closed ports. Treat results as informed estimates rather than absolute proof.

Comparison Table

Goal	Command
Detect OS	-O
Limit OS detection	--osscan-limit
Best estimate	--osscan-guess
OS + Versions	-O -sV
Comprehensive scan	-A

Typical Workflow

Target → -sS → -sV → -O → Research → Targeted Enumeration → Document Findings

Revision Notes

• -O = Detect OS
• --osscan-limit = Suitable hosts only
• --osscan-guess = Closest match
• -O -sV = OS + Versions
• -A = OS + Versions + Default Scripts + Traceroute

Practice Questions

1. What does -O do?
2. Why can OS detection fail?
3. When should you use --osscan-limit?
4. What is the purpose of --osscan-guess?
5. Why combine -O with -sV?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 6: Timing, Performance & Scan Optimization

Learn how to optimize Nmap scans for speed, reliability, and efficient network enumeration.

Learning Objectives

Understand Nmap timing templates, packet rate controls, scan delays, host timeouts, retry limits, and how to choose the right settings for different authorized environments.

Why Timing Matters

Choosing the correct timing settings balances scan speed and accuracy. Very fast scans may miss responses on slow networks, while very slow scans increase completion time.

Option	Name	Typical Use
-T0	Paranoid	Very cautious, slow scanning
-T1	Sneaky	Slow or sensitive networks
-T2	Polite	Reduce network load
-T3	Normal	Balanced default scanning
-T4	Aggressive	Most lab environments
-T5	Insane	Very fast, reliable local networks only

nmap -T0

Purpose: Paranoid timing

What it does: Slowest scan speed.

When to use: Learning cautious timing or testing slow scans.

Avoid when: Quick assessments.

Memory Tip: 0 = Slowest

nmap -T1

Purpose: Sneaky timing

What it does: Very slow scanning.

When to use: Sensitive or slow networks.

Avoid when: Routine lab work.

Memory Tip: 1 = Very slow

nmap -T2

Purpose: Polite timing

What it does: Reduces network load.

When to use: Shared or bandwidth-limited networks.

Avoid when: Time-limited scans.

Memory Tip: 2 = Polite

nmap -T3

Purpose: Normal timing

What it does: Balanced performance.

When to use: General-purpose scanning.

Avoid when: When another timing template better matches your needs.

Memory Tip: 3 = Balanced

nmap -T4

Purpose: Aggressive timing

What it does: Fast and reliable on most modern networks.

When to use: TryHackMe, Hack The Box, and local labs.

Avoid when: Unstable or high-latency networks.

Memory Tip: 4 = Fast

nmap -T5

Purpose: Insane timing

What it does: Fastest timing template.

When to use: Reliable local networks only.

Avoid when: Internet targets or unstable connections.

Memory Tip: 5 = Fastest

nmap --min-rate 1000

Purpose: Minimum packet rate

What it does: Requests at least the specified packet rate.

When to use: Controlled, fast lab environments.

Avoid when: Congested networks.

Memory Tip: Higher value = faster sending

nmap --max-rate 500

Purpose: Maximum packet rate

What it does: Limits packet sending rate.

When to use: When reducing network load.

Avoid when: When maximum speed is needed.

Memory Tip: Caps sending speed

nmap --scan-delay 500ms

Purpose: Delay between probes

What it does: Waits between packets.

When to use: Devices that respond poorly to rapid requests.

Avoid when: Speed-focused scans.

Memory Tip: Delay improves stability

nmap --host-timeout 5m

Purpose: Host timeout

What it does: Stops scanning a host after a set time.

When to use: Large assessments.

Avoid when: Short scans where timeout isn't an issue.

Memory Tip: Avoids hanging scans

nmap --max-retries 2

Purpose: Retry limit

What it does: Limits retries for unanswered probes.

When to use: Reliable lab environments.

Avoid when: Unstable networks needing higher reliability.

Memory Tip: Fewer retries = faster scans

Typical Workflow

Choose timing (-T3 or -T4) → SYN scan (-sS) → Version detection (-sV) → Default scripts (-sC) → Save results (-oA)

Revision Notes

• -T3 = Balanced
• -T4 = Best for most labs
• -T5 = Fastest
• --min-rate = Minimum packet rate
• --max-rate = Maximum packet rate
• --scan-delay = Delay between probes
• --host-timeout = Stop long scans
• --max-retries = Retry limit

Practice Questions

1. Which timing template is balanced by default?

2. When is -T4 a good choice?

3. What does --scan-delay control?
4. Why use --host-timeout?
5. How does --max-retries affect scanning?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 7: Output Formats, Reporting & Saving Scan Results

A scan that isn't saved is a scan you'll probably have to repeat.

Learning Objectives

Learn how to save Nmap results in different formats, choose the right output type, organize scan files, and build good reporting habits.

Why Save Scan Results?

Saving scan results preserves evidence, supports report writing, enables comparisons over time, and allows results to be imported into other tools.

nmap -oN scan.txt

Purpose: Normal output

What it does: Human-readable text output.

When to use: Most manual scans and lab notes.

Avoid when: Machine parsing.

Memory Tip: N = Normal

nmap -oX scan.xml

Purpose: XML output

What it does: Machine-readable XML.

When to use: Automation and tool integration.

Avoid when: Manual reading only.

Memory Tip: X = XML

nmap -oG scan.grep

Purpose: Grepable output

What it does: Easy to filter with grep and scripts.

When to use: Searching large scan results.

Avoid when: General reading.

Memory Tip: G = Grep

nmap -oA office_scan

Purpose: All formats

What it does: Creates .nmap, .xml, and .gnmap files.

When to use: Recommended for most assessments.

Avoid when: Only one format is needed.

Memory Tip: A = All

nmap -v

Purpose: Verbose mode

What it does: Shows additional scan progress.

When to use: Long scans.

Avoid when: Minimal output preferred.

Memory Tip: v = Verbose

nmap -vv

Purpose: Very verbose

What it does: Displays even more detail.

When to use: Troubleshooting.

Avoid when: Simple scans.

Memory Tip: vv = More detail

nmap -d

Purpose: Debug mode

What it does: Shows debugging information.

When to use: Learning and troubleshooting.

Avoid when: Normal assessments.

Memory Tip: d = Debug

nmap --append-output -oN scans.txt

Purpose: Append output

What it does: Adds results to an existing file.

When to use: Maintaining one scan log.

Avoid when: Separate files are preferred.

Memory Tip: Append, don't overwrite

nmap --resume scan.xml

Purpose: Resume scan

What it does: Continues an interrupted scan.

When to use: Long-running scans.

Avoid when: Completed scans.

Memory Tip: Resume = Continue

Comparison Table

Goal	Command
Human-readable	-oN
XML	-oX
Grepable	-oG
All formats	-oA
Verbose	-v / -vv
Debug	-d
Append output	--append-output
Resume scan	--resume

Best Practices

Use descriptive filenames (for example, thm_blue_initial.nmap), avoid overwriting previous scans, and prefer -oA for complete documentation.

Practice Questions

1. What is the difference between -oN, -oX, -oG and -oA?
2. When should XML output be used?
3. What does --append-output do?
4. How does --resume help?
5. Why is -oA commonly recommended?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 8: Real-World Nmap Workflows & Enumeration Strategy

Build repeatable, structured enumeration workflows for authorized security assessments.

Learning Objectives

Learn structured enumeration, choose the correct Nmap command for each phase, avoid common mistakes, and document findings effectively.

Recommended Enumeration Workflow

1. Host Discovery (-sn)
2. Port Scan (-sS -p-)
3. Service Version Detection (-sV)
4. Operating System Detection (-O)
5. Default NSE Scripts (-sC)
6. Service-Specific NSE Scripts
7. Research and Documentation

Single Host

Command: sudo nmap -sS -p- -T4

Discover all TCP ports before performing version detection and NSE enumeration.

Web Server

Command: sudo nmap --script http-title,http-enum,http-headers

Gather web server information before researching technologies.

SMB Server

Command: sudo nmap --script smb-os-discovery,smb-enum-shares,smb-vuln-ms17-010

Enumerate SMB safely and identify known issues.

SSH Server

Command: sudo nmap --script ssh-hostkey

Collect SSH host key information and supported algorithms.

FTP Server

Command: sudo nmap --script ftp-anon,ftp-syst

Check anonymous access and identify FTP server details.

UDP Scan

Command: sudo nmap -sU --top-ports 100

Start with common UDP ports before attempting larger UDP scans.

Save Results

Command: sudo nmap -oA initial_scan

Save normal, XML, and grepable outputs for reporting.

Decision Tree

Host Discovery → Port Scan → Service Detection → OS Detection → Default NSE → Service-Specific NSE → Research → Documentation

Comparison Table

Situation	Recommended Command
Initial TCP Scan	-sS -p-
Version Detection	-sV
OS Detection	-O
Default Scripts	-sC
Web Enumeration	http-* NSE
SMB Enumeration	smb-* NSE
FTP Enumeration	ftp-* NSE
SSH Enumeration	ssh-* NSE
Save Results	-oA

Common Mistakes

• Running -A immediately on every target.

• Skipping service version detection.

• Not saving scan results.

• Using every NSE script without a purpose.

• Failing to research discovered services.

Practice Questions

1. Why is phased enumeration preferred?

2. What follows port discovery?

3. Which workflow is suitable for SMB?

4. Why are UDP scans slower?

5. Why should every scan be documented?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 9: Advanced Nmap Features (Defensive Awareness & Network Analysis)

Understand advanced Nmap features for authorized testing, troubleshooting, and network analysis.

Learning Objectives

Understand advanced Nmap options, when to use them, their limitations, and how they support troubleshooting and analysis in authorized environments.

nmap -n

Purpose: Disable DNS

What it does: Skips reverse DNS lookups.

When to use: Faster scans and lab environments.

Avoid when: Hostnames are needed.

Memory Tip: n = No DNS

nmap -R

Purpose: Force DNS

What it does: Always resolve hostnames.

When to use: Asset inventory and documentation.

Avoid when: Scan speed is more important.

Memory Tip: R = Resolve

nmap -6

Purpose: IPv6 Scan

What it does: Scans IPv6 targets.

When to use: IPv6-enabled networks.

Avoid when: IPv4-only targets.

Memory Tip: 6 = IPv6

nmap --packet-trace

Purpose: Packet Trace

What it does: Displays packets sent and received.

When to use: Troubleshooting and learning.

Avoid when: Routine scans.

Memory Tip: Trace = Watch packets

nmap -e eth0

Purpose: Select Interface

What it does: Uses a chosen network interface.

When to use: Systems with multiple interfaces.

Avoid when: Only one interface exists.

Memory Tip: e = Interface

nmap --source-port 53

Purpose: Source Port

What it does: Uses a specified source port.

When to use: Testing firewall policies in labs.

Avoid when: Default behavior is sufficient.

Memory Tip: Specify packet source port

nmap -f

Purpose: Fragment Packets

What it does: Sends fragmented probe packets.

When to use: Studying fragmentation in labs.

Avoid when: General scanning.

Memory Tip: f = Fragment

nmap --mtu 24

Purpose: Custom MTU

What it does: Sets fragment size.

When to use: Controlled networking experiments.

Avoid when: No fragmentation required.

Memory Tip: MTU = Maximum Transmission Unit

nmap --spoof-mac 00:11:22:33:44:55

Purpose: Specify MAC

What it does: Uses a chosen MAC address on local networks.

When to use: Testing local network behavior.

Avoid when: Internet scanning.

Memory Tip: MAC = Layer 2 identity

Comparison Table

Feature	Command

Disable DNS	-n
Force DNS	-R
IPv6	-6
Packet Trace	--packet-trace
Interface	-e
Source Port	--source-port
Fragmentation	-f
Custom MTU	--mtu
MAC Address	--spoof-mac

Practice Questions

1. What is the difference between -n and -R?
2. When is --packet-trace useful?
3. Why choose an interface with -e?
4. When should you use -6?
5. Why do MAC address options matter only on local networks?

The Ethical Hacker's Command Handbook

Chapter 2 – Part 10: Nmap Master Cheat Sheet, Common Mistakes & Interview Preparation

Your quick revision guide for labs, interviews, and certification preparation.

20 Essential Nmap Commands

Purpose	Command	Memory Tip
Host Discovery	nmap -sn	Scan No Ports
SYN Scan	nmap -sS	S = SYN
TCP Connect	nmap -sT	T = TCP
UDP Scan	nmap -sU	U = UDP
Version Detection	nmap -sV	V = Version
OS Detection	nmap -O	O = Operating System
Default Scripts	nmap -sC	C = Common Scripts
Aggressive	nmap -A	A = Aggressive
All Ports	nmap -p-	Every Port
Timing	nmap -T4	Fast Labs
Save Results	nmap -oA	All Outputs

Recommended Workflow

1. Host Discovery (-sn)
2. SYN Scan (-sS -p-)
3. Version Detection (-sV)
4. OS Detection (-O)
5. Default NSE Scripts (-sC)
6. Service-Specific NSE Scripts
7. Save Results (-oA)
8. Research and Documentation

Common Beginner Mistakes

• Running -A before basic enumeration.

• Scanning only common ports.

• Ignoring service version detection.

• Not saving scan results.

• Running unnecessary NSE scripts.

Interview Questions

Q: Difference between -sS and -sT?

A: -sS performs a SYN scan, while -sT completes a full TCP connection.

Q: What does -A include?

A: OS detection, version detection, default NSE scripts, and traceroute.

Q: Why use -sV?

A: To identify software versions for further research.

Q: What does -oA do?

A: Saves results in .nmap, .xml, and .gnmap formats.

Q: Why is -T4 popular?

A: It provides a good balance of speed and reliability on most lab networks.

Quick Revision

Host Discovery → Port Scan → Version Detection → OS Detection → Default Scripts → Service-Specific Scripts → Save Results → Research → Documentation