JUMP TO ANY SECTION

Basic Syntax & Target Specification

Nmap accepts targets in multiple formats. Understanding these first means you can scope any scan precisely — single hosts, ranges, CIDR blocks, or input files from a recon tool like Subfinder.

SYNTAX & TARGET FORMATS — AUTHORISED TARGETS ONLY

```
$ nmap 192.168.1.1 # Single IP address
$ nmap 192.168.1.1-20 # IP range (1 through 20)
$ nmap 192.168.1.0/24 # Entire /24 subnet (256 hosts)
$ nmap 10.0.0.0/8 # Entire /8 — 16.7M addresses (use carefully)
$ nmap 192.168.1.1 192.168.1.2 # Multiple targets space-separated
$ nmap target.com # Domain name (resolves to IP first)
$ nmap -iL targets.txt # Read targets from file (one per line)
$ nmap -exclude 192.168.1.5 192.168.1.0/24 # Exclude specific host from range
$ nmap -6 2001:db8::1 # IPv6 target scan
```

NMap Scan Types — Every -s Flag Explained

The scan type determines how Nmap probes target ports — the TCP mechanism used, whether the full handshake completes, and therefore how visible the scan is to IDS systems and firewalls. This is the most important flag category to understand.

`-sn`	Ping Sweep (no port scan — host discovery only)	No	N/A

Root = requires sudo/root privileges. Stealth = how likely to be logged by IDS/firewall. All scans require authorisation.

Nmap Scan Types — -sS (SYN) is the default and most widely used — fast, stealthy, requires root. -sT is used when root is unavailable. -sU covers UDP services (DNS, SNMP, NTP). NULL/FIN/Xmas scans bypass some older stateless firewalls but are unreliable on modern systems.

SCAN TYPE COMMANDS WITH REAL EXAMPLES

```
# SYN scan (default, requires root) — fastest and most common
$ sudo nmap -sS 192.168.1.1
# TCP connect (no root needed, more visible)
$ nmap -sT 192.168.1.1
# UDP scan — slow but essential (DNS, SNMP, NTP are UDP)
$ sudo nmap -sU -top-ports 100 192.168.1.1
# Ping sweep — find all live hosts in a subnet (no port scan)
$ nmap -sn 192.168.1.0/24
# Combine TCP + UDP in one command
$ sudo nmap -sS -sU -p T:80,443,22,U:53,161 192.168.1.1
# Treat host as up even if not responding to ping (-Pn)
$ nmap -Pn -sT 192.168.1.1 # Useful when ICMP is blocked
```

Port Specification — Targeting Exactly What You Need

PORT SPECIFICATION FLAGS

```
$ nmap -p 80 192.168.1.1 # Single port
$ nmap -p 80,443,8080 192.168.1.1 # Multiple specific ports
$ nmap -p 1-1024 192.168.1.1 # Port range 1 to 1024
$ nmap -p- 192.168.1.1 # ALL 65535 ports (thorough but slow)
$ nmap -top-ports 100 192.168.1.1 # Top 100 most common ports
$ nmap -top-ports 1000 192.168.1.1 # Top 1000 (default without -p)
$ nmap -p U:53,T:80 192.168.1.1 # UDP port 53 + TCP port 80
$ nmap -F 192.168.1.1 # Fast scan — top 100 ports only
$ nmap -r 192.168.1.1 # Scan ports in consecutive order
```

Service & OS Detection — What Is Actually Running

Knowing a port is open is only the start. Service version detection identifies the exact software and version running — which maps directly to known CVEs. OS detection fingerprints the target operating system. These two flags together are what transforms a port list into a vulnerability assessment surface.

SERVICE VERSION & OS DETECTION

```
$ nmap -sV 192.168.1.1 # Service version detection
$ nmap -sV -version-intensity 9 192.168.1.1 # Max intensity (0-9)
```

```
$ sudo nmap -O 192.168.1.1 # OS detection (requires root)
$ sudo nmap -O -osscan-guess 192.168.1.1 # Aggressive OS guess if uncertain
$ sudo nmap -A 192.168.1.1 # Aggressive: -sV -O -sC + traceroute

# Sample -sV output:
22/tcp open ssh OpenSSH 8.9p1 Ubuntu 3ubuntu0.3
80/tcp open http Apache httpd 2.4.54 ((Ubuntu))
443/tcp open ssl/http nginx 1.18.0
3306/tcp filtered mysql

# Apache 2.4.54 → search CVE database for known vulnerabilities
```

Timing & Performance — Speed vs Accuracy

TIMING TEMPLATES T0–T5

```
$ nmap -T0 192.168.1.1 # Paranoid — 5 min/port, IDS evasion, very slow
$ nmap -T1 192.168.1.1 # Sneaky — 15 sec/probe, low bandwidth
$ nmap -T2 192.168.1.1 # Polite — slows down to avoid congestion
$ nmap -T3 192.168.1.1 # Normal — default timing (no flag = T3)
$ nmap -T4 192.168.1.1 # Aggressive — fast networks, ~4x speed boost ← USE THIS
$ nmap -T5 192.168.1.1 # Insane — very fast, may miss ports on slow targets

# Fine-grained timing control:
$ nmap -min-rate 5000 192.168.1.1 # Min 5000 packets/sec
$ nmap -max-retries 1 192.168.1.1 # Retry failed probes max 1 time (default: 3)
$ nmap -host-timeout 30s 192.168.1.0/24 # Skip host if no response in 30s
```

NSE Scripts — Nmap's Superpower

The Nmap Scripting Engine transforms Nmap from a port scanner into a vulnerability assessment platform. 600+ scripts bundled with Kali Linux cover everything from service banner grabbing to vulnerability detection to brute force authentication testing. Computer Security

NSE SCRIPT COMMANDS

```
$ nmap -sC 192.168.1.1 # Run default scripts (equivalent to -script=default)
$ nmap -script vuln 192.168.1.1 # All vulnerability detection scripts
$ nmap -script auth 192.168.1.1 # Authentication bypass detection
$ nmap -script brute 192.168.1.1 # Brute force scripts (use with caution)
$ nmap -script discovery 192.168.1.1 # Service and host discovery scripts
$ nmap -script http-title 192.168.1.1 # Get HTTP page title (specific script)
$ nmap -script http-headers 192.168.1.1 # Get all HTTP headers
$ nmap -script smb-vuln-ms17-010 192.168.1.1 # Check for EternalBlue
$ nmap -script ssl-cert -p 443 192.168.1.1 # SSL certificate details
$ nmap -script dns-brute target.com # DNS subdomain brute force

# List all scripts matching a keyword:
$ ls /usr/share/nmap/scripts/ | grep http # All HTTP-related scripts
$ nmap -script-help http-title # Documentation for specific script
```

Output Formats — Save and Share Your Results

OUTPUT FLAGS

```
$ nmap -oN scan.txt 192.168.1.1 # Normal output → text file
$ nmap -oX scan.xml 192.168.1.1 # XML output (imports into Metasploit, etc.)
$ nmap -oG scan.gnmap 192.168.1.1 # Grepable format (pipe through grep easily)
$ nmap -oA scan_results 192.168.1.1 # ALL formats simultaneously ← USE THIS
$ nmap -v 192.168.1.1 # Verbose output (show results as discovered)
$ nmap -vv 192.168.1.1 # Very verbose
$ nmap -d 192.168.1.1 # Debug output
$ nmap -reason 192.168.1.1 # Show why each port has its state

# Parse grepable output for open ports:
$ grep "open" scan.gnmap | awk '{print $2}' # Extract IPs with open ports
```

Networking

Firewall Evasion & Spoofing Techniques

In authorised penetration tests, firewalls and IDS systems may filter or block standard scans. These flags allow you to probe for filter rules, fragment packets, and adjust scan behaviour to get more accurate results from filtered environments. All for authorised use on in-scope targets only.

FIREWALL EVASION FLAGS (AUTHORISED TARGETS ONLY)

```
$ nmap -f 192.168.1.1 # Fragment packets (8-byte fragments)
```

```
$ nmap -f -f 192.168.1.1 # 16-byte fragments (smaller = harder to reassemble)
$ nmap -mtu 24 192.168.1.1 # Custom MTU offset (must be multiple of 8)
$ nmap -D 192.168.1.100,192.168.1.101,ME 192.168.1.1 # Decoy scan
$ nmap -source-port 53 192.168.1.1 # Spoof source port (firewalls allow port 53)
$ nmap -data-length 25 192.168.1.1 # Append random data to packets
$ nmap -badsum 192.168.1.1 # Send packets with bad checksums (probe firewalls)
$ nmap -scan-delay 1s 192.168.1.1 # 1 second between probes (IDS threshold evasion)
```

Professional Scan Combinations — Copy-Paste Ready

These are the actual Nmap command combinations used in professional penetration testing engagements. Each serves a specific purpose — from fast initial triage to comprehensive vulnerability baseline. Run these on your home lab or authorised targets.

Kali Linux — Professional Nmap Scan Sequences

```
# ── SCAN 1: RAPID HOST DISCOVERY (large networks) ────────
$ sudo nmap -sn -PE -PP -PM 192.168.1.0/24
Uses ICMP echo, timestamp, and netmask requests for host discovery

# ── SCAN 2: QUICK TRIAGE (first look at a host) ──────────
$ sudo nmap -sS -sV -T4 -top-ports 1000 192.168.1.1
Fast SYN scan, top 1000 ports, service versions, aggressive timing

# ── SCAN 3: COMPLETE BASELINE (standard pentest) ─────────
```

```
$ sudo nmap -sV -sC -O -p- -T4 -oA full_scan 192.168.1.1
All ports, service versions, default scripts, OS detect, save all formats

# ── SCAN 4: VULNERABILITY SCAN (after baseline) ──────────
$ sudo nmap -sV -script vuln -p 22,80,443,3306 192.168.1.1
Run all vuln scripts against specific open ports found in previous scan

# ── SCAN 5: UDP KEY SERVICES ─────────────────────────────
$ sudo nmap -sU -sV -p U:53,161,123,137,500,1194 192.168.1.1
DNS(53), SNMP(161), NTP(123), NetBIOS(137), IKE(500), OpenVPN(1194)

# ── SCAN 6: WEB SURFACE (web app targets) ────────────────
$ nmap -sV -script http-title,http-headers,http-methods -p 80,443,8080,8443 192.168.1.1

✓ These 6 scans constitute a complete initial reconnaissance on any authorised target
```

Professional Nmap Scan Sequences — These six scans are used in real penetration testing engagements. Run them in order on any authorised target: (1) host discovery, (2) quick triage, (3) complete baseline, (4) vuln scan on open ports, (5) UDP key services, (6) web surface. Save all output with -oA for the engagement report.

Nmap Quick Reference Card — Screenshot This

securityelites.com

NMAP QUICK REFERENCE CARD 2026 — SCREENSHOT & KEEP

SCAN TYPES

-sS SYN (default, root)
-sT TCP connect
-sU UDP scan

PORTS

-p 80 Single port
-p 1-1024 Range
-p- All 65535

DETECTION

-sV Service versions
-O OS detection
-A Aggressive all

-sn Ping sweep only
-Pn Skip ping, treat as up

-F Top 100 fast
-top-ports 1000

-sC Default scripts
-script vuln

TIMING

-T0 Paranoid (slow)
-T3 Normal (default)
-T4 Aggressive ←USE
-T5 Insane
-min-rate 5000

OUTPUT

-oN Normal text
-oX XML
-oG Grepable
-oA ALL formats ←USE
-v Verbose

PRO COMBO

sudo nmap
-sV -sC -O
-p- -T4
-oA results
[TARGET]

Full PDF: securityelites.com/nmap-commands-cheat-sheet.pdf

📸 Nmap Quick Reference Card 2026 — Screenshot this card for your second monitor or desktop wallpaper. Every category you will use daily: scan types, port specification, detection, timing, output formats, and the professional combo command. PDF version available at the link.