nmap Network scanner

START HERE

`nmap <host>`
Scan top 1000 ports

`nmap -A <host>`
OS, versions, scripts

`nmap -sn <network>/24`
Find live hosts

`nmap -Pn <host>`
Skip host discovery

`nmap -p- <host>`
Scan all 65535 ports

`nmap -oA <name> <host>`
Save all formats

`nmap -sV <host>`
Identify services

`sudo nmap -sS <host>`
Fast SYN scan

Targets hosts, CIDR → `discovery` → **Live hosts** ping probes → `port scan` → **Port states** open/closed/filtered → `-sV -O -sC` → **Details** service, OS, scripts

CONCEPTS

Only scan hosts and networks you own or are explicitly authorised to test. Unauthorised scans can be illegal and set off alarms.

Without root, nmap falls back to a TCP connect scan (-sT); raw-packet scans like -sS, -sU and -O need privileges.

Port states: open, closed, filtered (no reply, probably a firewall), unfiltered, plus open|filtered and closed|filtered when nmap cannot tell.

TARGETS

`nmap <host>`
Scan one host
Scans the 1000 most common ports with the default scan.

`nmap <host1> <host2>`
Scan several hosts

`nmap <net>/24`
Scan a whole subnet (CIDR)

`nmap 192.168.0-255.1-254`
Scan with octet ranges

`nmap -iL <file>`
Read targets from a file
Use - to read targets from standard input.

`nmap --exclude <hosts>`
Skip hosts or networks

`nmap --excludefile <file>`
Skip hosts listed in a file

`nmap -sL <net>/24`
List targets, send no probes

`nmap -6 <host>`
Scan IPv6 addresses

`nmap -iR <n>`
Pick n random targets

HOST DISCOVERY

`nmap -sn <net>/24`
Ping scan only, no port scan
Formerly -sP. Lists the hosts that answered.

`nmap -Pn <host>`
Skip discovery, assume host is up
Use when hosts block ping and show as down.

`nmap -PS<ports> <host>`
Discover with TCP SYN probes
Default port is 80.

`nmap -PA<ports> <host>`
Discover with TCP ACK probes

`nmap -PU<ports> <host>`
Discover with UDP probes

HOST DISCOVERY (cont.)

`nmap -PE <host>`
Discover with ICMP echo

`nmap -PR <net>/24`
ARP discovery on local LAN
Used automatically for hosts on the local Ethernet.

`nmap -n <host>`
Never resolve DNS names
Faster; -R forces resolution for every host.

`nmap --traceroute <host>`
Trace route after the scan

PORTS

`nmap -p <port> <host>`
Scan one port

`nmap -p 22,80,443 <host>`
Scan a port list

`nmap -p 1-1023 <host>`
Scan a port range

`nmap -p- <host>`
Scan all 65535 ports

`nmap -F <host>`
Fast: top 100 ports only

`nmap --top-ports <n> <host>`
Scan the n most common ports

`nmap -p T:80,U:53 <host>`
Mix TCP and UDP ports
UDP ports also need -sU alongside a TCP scan type.

`nmap -p http* <host>`
Select ports by service name
Quote the argument so the shell does not expand it.

`nmap --exclude-ports <ports> <host>`
Leave ports out

SCAN TYPES

`nmap -sS <host>`
TCP SYN (half-open) scan
Default when run as root: fast and rarely logged by apps.

`nmap -sT <host>`
TCP connect scan
Default without root; completes every handshake.

`nmap -sU <host>`
UDP scan
Slow; combine with --top-ports to keep it practical.

`nmap -sA <host>`
ACK scan to map firewall rules
Reports only filtered or unfiltered.

`nmap -sN <host>`
TCP NULL scan (no flags)

`nmap -sF <host>`
TCP FIN scan

SCAN TYPES (cont.)

`nmap -sX <host>`
TCP Xmas scan (FIN, PSH, URG)

`nmap -sW <host>`
TCP window scan

`nmap -sO <host>`
Scan IP protocols, not ports

DETECTION

`nmap -sV <host>`
Detect service versions

`nmap -O <host>`
Detect the operating system
Needs root; works best with an open and a closed port.

`nmap -A <host>`
OS, versions, scripts, traceroute
Noisy: it is the aggressive all-in-one option.

`nmap -sV --version-intensity <0-9> <host>`
Set version probe depth
Default is 7.

`nmap -sV --version-light <host>`
Quick version detection
Same as intensity 2.

`nmap -sV --version-all <host>`
Try every version probe

`nmap -O --osscan-guess <host>`
Guess OS more aggressively

SCRIPTS (NSE)

`nmap -sC <host>`
Run default scripts
Equals --script=default.

`nmap --script <name> <host>`
Run a named script

`nmap --script <category> <host>`
Run a category (safe, vuln...)

`nmap --script "http-*" <host>`
Run scripts by wildcard

`nmap --script-args <k>=<v> <host>`
Pass arguments to scripts

`nmap --script-help <script>`
Describe a script or category

NSE scripts are not sandboxed and categories like intrusive, brute, exploit and dos can disrupt services. Review scripts before running them and prefer safe.

TIMING & PERFORMANCE

`nmap -T4 <host>`
Faster scan on a reliable network
Templates run -T0 paranoid to -T5 insane; default is -T3.

`nmap -T2 <host>`
Polite: slow, light on the target

`nmap --min-rate <pps> <host>`
Send at least this many packets/s

`nmap --max-rate <pps> <host>`
Cap the send rate

OUTPUT

`nmap -v <host>`
More verbose, repeat for more

`nmap --open <host>`
Show only open ports

`nmap -oN <file> <host>`
Save normal output

`nmap -oX <file> <host>`
Save XML output

`nmap -oA <base> <host>`
Save normal, XML and grepable

`nmap -oG <file> <host>`
Save grepable output
Deprecated; prefer XML for parsing.

`nmap --reason <host>`
Explain each port state

`nmap -oX - <host>`
Write XML to stdout

MISC

`nmap -h`
Short help screen

`nmap --version`
Print nmap version

RECIPES

Map a LAN, then dig in

```
# 1. who is up?
nmap -sn 192.168.1.0/24 -oG - | grep Up
# 2. details on one host
sudo nmap -sS -sV -O -p- 192.168.1.10
```

Thorough scan, saved

```
sudo nmap -p- -sV -sC -T4 -oA scan <host>
# review later
less scan.nmap
```

Quick web service check

```
nmap -p 80,443,8080 --open -sV \
 "--script=http-title" <net>/24
```

Top UDP services

```
sudo nmap -sU --top-ports 50 -sV <host>
```