

# nmap 快速参考

端口扫描、主机发现、服务检测与 NSE 脚本

## 基础扫描

### 扫描目标

```
nmap 192.168.1.1 # single host
nmap 192.168.1.0/24 # entire subnet
nmap 192.168.1.1-50 # IP range
nmap -iL targets.txt # hosts from file
```

### 目标规格

```
192.168.1.1 单个 IP 地址
192.168.1.0/24 CIDR 表示法 (256 台主机)
192.168.1.1-254 IP 范围
example.com 主机名 (解析为 IP)
-iL file.txt 从文件读取目标
--exclude 192.168.1.1 排除特定主机
--excludefile skip.txt 从文件排除主机
```

## 端口扫描

### 扫描类型

```
-sS TCP SYN 扫描 (默认, 隐蔽, 需 root)
-sT TCP 全连接扫描 (完整握手, 不需 root)
-sU UDP 扫描 (速度慢, 常被过滤)
-sA TCP ACK 扫描 (检测防火墙)
-sN TCP NULL 扫描 (无标志位)
-sF TCP FIN 扫描 (仅 FIN 标志)
-sX Xmas 扫描 (FIN+PSH+URG 标志)
```

### 端口选择

```
nmap -p 80,443 target # specific ports
nmap -p 1-1000 target # port range
nmap -p- target # all 65535 ports
nmap --top-ports 100 target # most common 100 ports
```

### 端口状态

```
open 应用程序正在接受连接
closed 端口可达但无服务监听
filtered 防火墙拦截, 无法确定状态
unfiltered 端口可访问, 但开/关未知
open|filtered 无法区分开放还是被过滤
```

## 主机发现

### 发现方法

```
-sn 仅 Ping 扫描 (不扫描端口)
-Pn 跳过主机发现 (全部视为在线)
-PS 80,443 在指定端口进行 TCP SYN 发现
-PA 80 TCP ACK 发现
-PU 53 UDP 发现
-PE ICMP 回显请求
-PR ARP 发现 (本地网络)
```

### 网络扫描

```
nmap -sn 192.168.1.0/24 # ping sweep subnet
nmap -sn -n 10.0.0.0/24 # sweep, skip DNS
nmap -sn -PR 192.168.1.0/24 # ARP scan (fastest)
```

## 服务检测

### 版本检测

```
nmap -sV target # detect service versions
nmap -sV --version-intensity 5 target # deeper probing
nmap -sV --version-all target # try every probe (slow)
nmap -A target # OS + version + scripts + traceroute
```

## 服务检测标志

```
-sV 对开放端口探测服务/版本
--version-intensity 0-9 探测强度 (默认 7)
--version-light 轻量探测 (强度 2)
--version-all 尝试所有探测 (强度 9)
-A 激进模式: -sV -O --script=default-traceroute
-sC 运行默认 NSE 脚本
```

## 操作系统检测

### OS 指纹识别

```
nmap -O target # OS detection (needs root)
nmap -O --osscan-limit target # only scan promising hosts
nmap -O --osscan-guess target # aggressive OS guessing
nmap -A target # includes OS detection
```

### OS 检测标志

```
-O 启用 OS 检测
--osscan-limit 跳过没有开放+关闭 TCP 端口的主机
--osscan-guess 更激进地猜测 OS
--max-os-tries N 每台主机最大 OS 检测尝试次数
```

## 脚本 (NSE)

### 脚本用法

```
nmap --script=default target # default category
nmap --script=vuln target # vulnerability scripts
nmap --script=http-headers target
nmap --script="http-*" target # wildcard match
```

### 脚本类别

```
default 安全实用的脚本 (-sC 简写)
vuln 检查已知漏洞
safe 非侵入性脚本
intrusive 可能导致目标崩溃或触发 IDS
discovery 网络与服务发现
auth 认证相关检查
brute 暴力破解凭据测试
exploit 主动漏洞利用
```

### 常用脚本

```
http-title 获取网页标题
ssl-cert 显示 SSL 证书详情
ssh-hostkey 显示 SSH 主机密钥指纹
dns-brute 枚举 DNS 子域名
smb-os-discovery 通过 SMB 检测 Windows OS
vuln 运行所有漏洞检查
```

## 输出格式

### 输出选项

```
nmap -oN scan.txt target # normal text output
nmap -oX scan.xml target # XML output
nmap -oG scan.gnmap target # grepable output
nmap -oA scan_all target # all formats at once
```

## 输出标志

```
-oN file 普通文本输出到文件
-oX file XML 输出 (适合工具解析)
-oG file 可 grep 格式 (每行一台主机)
-oA basename 三种格式同时输出
-v 增加详细程度 (-vv 更多)
-d 调试输出 (-dd 更多)
--open 仅显示开放端口
--reason 显示端口状态的原因
```

## 速度与性能

### 时序模板

```
-T0 (偏执) 极慢, IDS 规避 (探测间隔 5 分钟)
-T1 (鬼鬼祟祟) 慢速, IDS 规避 (探测间隔 15 秒)
-T2 (礼貌) 降低速度, 减少带宽占用
-T3 (正常) 默认时序
-T4 (激进) 快速, 假设网络可靠
-T5 (疯狂) 最快, 可能遗漏结果
```

### 精细调优

```
--min-rate 1000 每秒至少发送 1000 个数据包
--max-rate 500 每秒最多 500 个数据包
--max-retries 2 最大重试次数
--host-timeout 30m 扫描超过 30 分钟则跳过
--scan-delay 1s 探测间隔
--min-parallelism 10 最小并行探测组数
```

## 防火墙规避

### 规避技术

```
-f 分片数据包 (8 字节块)
-D RND:5 使用 5 个随机 IP 进行诱骗扫描
-S spoof_ip 伪造源 IP (需要原始数据包)
-e eth0 使用指定网络接口
--source-port 53 使用指定源端口 (如 DNS)
--data-length 25 在数据包中追加随机数据
--spoof-mac 0 随机化 MAC 地址
```

### 规避示例

```
nmap -f -D RND:3 target # fragments + decoys
nmap --source-port 53 target # DNS port (often allowed)
nmap -T1 --scan-delay 5s target # slow to evade IDS
```

## 常见模式

### 快速侦察

```
nmap -T4 -F target # fast common ports
nmap -T4 -A -v target # OS + service detection
nmap -sV --top-ports 1000 target # top 1000 + versions
```

### 全面扫描

```
# Full TCP + service + OS + scripts
nmap -sS -sV -O -sC -p- -T4 -oA full target
# UDP scan on common ports
nmap -sU --top-ports 50 target
```

### Web 服务器审计

```
nmap -p 80,443 --script=http-title,http-headers,\
ssl-cert,http-methods target
# Check for open proxies and vulns
nmap -p 80,443,8080 --script=http-open-proxy,vuln target
```

# nmap 快速参考

---

## 网络资产清单

```
# Discover all live hosts with OS info
nmap -sn 192.168.1.0/24 -oG - | grep "Up"
# Service inventory for subnet
nmap -sV -T4 192.168.1.0/24 -oX inventory.xml
```