Nmap Tutorial

Einführung

Nmap ("Network Mapper") ist ein Open-Source-Werkzeug für die Netzwerkanalyse und Sicherheitsüberprüfung. Es wurde entworfen, um große Netzwerke schnell zu scannen, obwohl es auch bei einzelnen Hosts gut funktioniert. Nmap benutzt rohe IP-Pakete, um festzustellen, welche Hosts im Netzwerk verfügbar sind, welche Dienste (Anwendungsname und -version) diese Hosts bieten, welche Betriebssysteme (und Versionen davon) darauf laufen, welche Art von Paketfiltern/-Firewalls benutzt werden sowie Dutzende anderer Eigenschaften. Auch wenn Nmap üblicherweise für Sicherheitsüberprüfungen verwendet wird, wird es von vielen Systemen und Netzwerkadministratoren für Routineaufgaben benutzt, z.B. Netzwerkinventarisierung, Verwaltung von Ablaufplänen für Dienstaktualisierungen und die Überwachung von Betriebszeiten von Hosts oder Diensten.

Obwohl ein ausspähen eines Netzwerks mit Nmap auf gewisse Weise verschleiert werden kann, so erzeugt es dennoch eine hohe Netzwerklast und wird als Angriff auf das Netzwerk angesehen und von den meisten Administratoren verboten. Es gilt daher sicherzustellen, dass Nmap nicht auf das Hochschulnetz oder sonstige Netzwerke ohne ausdrückliche Erlaubnis ausgeführt wird.

Die Ausgabe von Nmap ist eine Liste gescannter Ziele mit zusätzlicher Information zu jedem, abhängig von den benutzten Optionen. Die entscheidende Information dabei steht in der "Tabelle der interessanten Ports". Diese Tabelle listet die Portnummer und das -protokoll sowie den Dienstnamen und -zustand auf. Der Zustand ist entweder *open*, *filtered*, *closed* oder *unfiltered*. *Offen* bedeutet, dass auf diesem Port des Zielrechners eine Anwendung auf eingehende Verbindungen/Pakete lauscht. *Filtered* bedeutet, dass eine Firewall, ein Filter oder ein anderes Netzwerkhindernis den Port blockiert, so dass Nmap nicht wissen kann, ob er *open* oder *closed* ist. Für geschlossene Ports gibt es keine Anwendung, die auf ihnen lauscht, auch wenn sie jederzeit geöffnet werden könnten. Als *unfiltered* werden Ports dann klassifiziert, wenn sie auf Nmaps Testpakete antworten, Nmap aber nicht feststellen kann, ob sie *open* oder *closed* sind. Nmap gibt die Zustandskombinationen *open*|*filtered* und *closed*|*filtered* an, wenn es nicht feststellen kann, welcher der beiden Zustände für einen Port zutrifft. Die Port-Tabelle enthält eventuell auch Details zur Softwareversion, sofern eine Versionserkennung verlangt wurde. Wurde ein IP-Protokoll-Scan verlangt (-sO), dann bietet Nmap Angaben über die unterstützten IP-Protokolle statt über lauschende Ports.

Zusätzlich zur Tabelle der interessanten Ports kann Nmap weitere Angaben über Ziele bieten, darunter Reverse-DNS-Namen, Mutmaßungen über das benutzte Betriebssystem, Gerätearten und MAC-Adressen.

Einen typischen Nmap-Scan sehen Sie in Example 1. Die einzigen in diesem Beispiel benutzten Nmap-Argumente sind -A für die Betriebssystem- und Versionserkennung, Script-Scanning und Traceroute und -T4 für eine schnellere Ausführung. Danach kommen die Namen der Zielhosts.

Beispiel 1. Ein repräsentativer Nmap-Scan

```
# nmap -A -T4 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Interesting ports on scanme.nmap.org (64.13.134.52):
Not shown: 994 filtered ports
PORT    STATE  SERVICE VERSION
22/tcp  open   ssh     OpenSSH 4.3 (protocol 2.0)
25/tcp  closed smtp
53/tcp  open   domain  ISC BIND 9.3.4
70/tcp  closed gopher
80/tcp  open   http    Apache httpd 2.2.2 ((Fedora))
|_ HTML title: Go ahead and ScanMe!
113/tcp closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.20-1 (Fedora Core 5)

TRACEROUTE (using port 80/tcp)
HOP RTT   ADDRESS
[Cut first seven hops for brevity]
8   10.59 so-4-2-0.mpr3.pao1.us.above.net (64.125.28.142)
9   11.00 metro0.sv.svcolo.com (208.185.168.173)
10  9.93  scanme.nmap.org (64.13.134.52)

Nmap done: 1 IP address (1 host up) scanned in 17.00 seconds
```

(Quelle: https://nmap.org/book/man.html)

Optionen

Sämtliche Optionen von Nmap werden in der man page aufgelistet oder auch durch Eingabe von nmap ohne Parameter in die Konsole. Die häufigsten Parameter werden im Folgenden beschrieben.

-sT

TCP Connect() scan

Versucht sich auf Ports zu verbinden um sie zu prüfen.

-sS

TCP stealth SYN scan

Verwendet nur SYN requests statt Connect um TCP Ports zu erkennen und ist daher etwas leiser

-sn

No Port Scan

Findet heraus welche Hosts im Netzwerk online sind ohne die Ports zu scanen

-sU

UDP Scan

Scannt sämtliche UDP Ports

-O

OS detection

Versucht herauszufinden welches Betriebssystem zu jedem Host gehört

-6

Enable IPv6 scanning

Um Nmap mit IPv6 zu verwenden muss sowohl die Quelle als auch das Zielsystem IPv6 fähig sein.

In Version 7.01 ermöglicht diese Option auf Windows (ab Vista) raw-socket IPv6 scans nur auf Ethernet-Geräten und nicht auf Tunneln.

-v/-V

verbosity

Erhöhen/Verringern der Ausführlichkeit des Berichts

-F

Fast mode

Scannt nur die 100 häufigsten TCP ports

-A

Agressive scan

Diese Option stellt zusätzliche erweiterte und aggressive Optionen zur Verfügung. In Version 7.01 hat es dieselbe Bedeutung wie OS detection (`-O`), version scanning (`-sV`), script scanning (`-sC`) und traceroute (`--traceroute`).

-Pn

No Ping

Schaltet die Nmap Erkennung aus und führt damit sämtliche Scans auch gegen Hosts aus, die keinen Ping zurückliefern.

Aufgaben

A1

Führen sie einen Nmap scan auf ihren Rechner durch und analysieren sie die Rückgabe.

A2

Welche Hosts sind in ihrem Netzwerk online?

A3

Wählen sie einen Host im Netzwerk (z.B. den ihres Sitznachbarn) und finden sie über nmap heraus, welches Betriebssystem er benutzt, welche Ports offen sind und welche Services darauf laufen.

Lösung

A1

```
nmap -A localhost
```

Hierbei sollte ein Ergebnis ähnlich des folgenden herauskommen:

```
Interesting ports on 10.0.0.4:
The 1668 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 2.0.3
22/tcp   open  ssh     OpenSSH 4.2 (protocol 1.99)
631/tcp  open  ipp     CUPS 1.1
6000/tcp open  X11      (access denied)
Device type: general purpose
Running: Linux 2.4.X|2.5.X|2.6.X
OS details: Linux 2.4.0 - 2.5.20, Linux 2.5.25 - 2.6.8 or
        Gentoo 1.2 Linux 2.4.19 rc1-rc7
```

In diesem Beispiel sind die TCP Ports 21, 22, 631 und 6000 offen und alle außer 6000 geben ihre Version bekannt. Zudem Erkennen wir das 10.0.0.4 ein Linux System mit einem Kernel zwischen Version 2.4 und 2.6 ist.

A2

```
nmap –sn <routerIP>/24
```

Hierbei sollten alle Hosts im Netzwerk, die online sind mit Netzwerknamen, Macadresse, IP und evtl. dem Hersteller aufgelistet werden

A3

`nmap -sS -sU -T4 -A -v <target>` oder `nmap -sS -sU -T4 –O -sV -v <target>`

Die Option T4 wählt ein schnelles Timing template und beschleunigt den Vorgang, ist jedoch nicht nötig. Die Optionen –sS und –sU werden benötigt um sowohl TCP als auch UDP Ports zu scannen. Mit –A bekommen wir die Servicenamen, die Version und das Betriebssystem heraus. –v hilft uns dabei Informationen über den Fortschritt des Scans zu erhalten, da vor allem der UDP Scan einige Zeit in Anspruch nimmt.