

Realzione sulla scansione delle porte con `nmap` su Metasploitable e Windows10 pro

Introduzione

L'obiettivo dell'esercitazione di oggi, era usare la macchina Kali Linux per eseguire delle scansioni su porte utilizzando il tool `nmap` (già precedentemente installato sulla Kali), avendo come target la Metasploitable e Windows10 pro.

Che tipi di scansioni bisogna eseguire?

- OS fingerprint (per entrambe le macchine)
- Syn Scan
- TCP connect
- Version detection

È importante conoscere gli indirizzi IP di entrambe le macchine per poter effettuare queste scansioni.

Per individuare gli indirizzi IP delle macchine su cui vogliamo fare lo scan delle porte, esistono diversi metodi:

- Aprire dalla Virtualbox entrambe le macchine ed eseguire i seguenti comandi:
 - `ipconfig` su Windows
 - `ifconfig` su Metasploitable
- Dalla Kali eseguire il comando `sudo nmap -sn 192.168.50.0/24` che elenca gli host vivi con gli IP che fanno parte di quella LAN (nel mio caso `192.168.50.x`)

Una volta che abbiamo a disposizione gli indirizzi IP corretti delle due macchine, iniziamo la scansione usando `nmap`.

Scansione Os fingerprint

OS Fingerprinting è una scansione che viene effettuata per identificare il tipo di sistema operativo che è installato sulla macchina target.

Il comando deve essere usato in questo modo:

```
sudo nmap -O <target>
```

Andremo ad eseguire questo comando sulla Kali, e lo faremo per entrambe le macchine, per vedere la distinzione tra i due sistemi operativi installati.

OS Fingerprinting per Metasploitable

```
sudo nmap -O 192.168.50.5
```

```
(kali@kali)-[~]
$ sudo nmap -O 192.168.50.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 19:59 CEST
Nmap scan report for 192.168.50.5
Host is up (0.048s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:2C:82:94 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.92 seconds
```

Lanciando il comando si può vedere come, nelle ultime righe, viene indicato:

- il tipo di device
- il sistema operativo
- versione del kernel
- uptime approssimativo

OS Fingerprinting per Windows10 pro

La stessa cosa accade se lanciamo lo stesso comando, prendendo come target l'indirizzo IP di Windows:

```
sudo nmap -O 192.168.50.6
```

Vediamo l'output con la seguente immagine:

```

(kali@kali)-[~]
$ sudo nmap -O 192.168.50.6
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 20:00 CEST
Nmap scan report for 192.168.50.6
Host is up (0.0010s latency).
Not shown: 981 closed tcp ports (reset)
PORT      STATE SERVICE
7/tcp     open  echo
9/tcp     open  discard
13/tcp    open  daytime
17/tcp    open  qotd
19/tcp    open  chargen
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsdapi
5432/tcp  open  postgresql
8009/tcp  open  ajp13
8080/tcp  open  http-proxy
8443/tcp  open  https-alt
MAC Address: 08:00:27:E1:3A:4B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10
OS details: Microsoft Windows 10 1507 - 1607
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.21 seconds

```

Anche qui vediamo, nelle ultime righe:

- il tipo di device
- il sistema operativo
- versione del kernel
- uptime approssimativo

SYN Scan

Eseguiamo il comando SYN scan solo sulla Metasploitable.

Cosa fa esattamente? Esegue una scansione "half-open", inviando i pacchetti SYN e attendendo risposte SYN/ACK.

- Se riceve SYN/ACK, vuol dire che la porta è aperta
- Se riceve RST, vuol dire che la porta è chiusa

È più veloce e meno rilevabile.

Vediamo come funziona nel concreto:

```
sudo nmap -ss <target>
```

Questo è il comando `nmap` da lanciare dalla Kali.

Nel mio caso, il comando è il seguente:

```
sudo nmap -ss 192.168.50.5
```

Vediamo di seguito l'output e lo analizziamo:

```
(kali@kali)-[~]
$ sudo nmap -sS 192.168.50.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 19:57 CEST
Nmap scan report for 192.168.50.5
Host is up (0.080s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:2C:82:94 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 7.81 seconds
```

Quello che possiamo vedere è una sorta di tabella che raccoglie i dati sulle porte:

- colonna **PORT** → indica il numero della porta e il tipo (tcp, udp ect)
- colonna **STATE** → indica lo stato della porta, se è **open** oppure **closed**
- colonna **SERVICE** → indica il servizio attivo in quella porta (smtp, telnet, http ect)

TCP Connect Scan

La scansione TCP connect, utilizza una chiamata di sistema connect() che completa tutta la 3-way handshake TCP.

È più facile da eseguire, in quanto non richiede privilegi speciali.

La eseguiremo sempre sulla Metasploitable

Viene lanciato con il seguente comando, da Kali:

```
sudo nmap -sT <target>
```

Nel mio caso:

```
sudo nmap -sT 192.168.50.5
```

Vediamo l'output:

```

(kali@kali)-[~]
$ sudo nmap -sT 192.168.50.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 20:18 CEST
Nmap scan report for 192.168.50.5
Host is up (0.0082s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 08:00:27:2C:82:94 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 7.04 seconds

```

Come possiamo vedere, l'output è praticamente identico a quello della SYN Scan. Come mai?

Riassumiamo nella seguente tabella le differenze:

Caratteristiche	SYN Scan	TCP Connect Scan
Tipo di connessione	parziale (half-open)	Completa (full 3-way handshake)
Rumorosità	Basso profilo (meno rilevabile)	Più evidente nei log
Richiede permessi root?	Sì	No
Velocità	Più veloce	Più lento
Visibilità sui log del target	Difficile da rilevare	facile da rilevare

Quindi cosa vediamo?

- I risultati sulle porte aperte/chiusure sono identici
- la differenza principale è nel modo in cui si ottengono, non nel risultato finale:
 - SYN è più stealth e rapido
 - TCP Connect è più semplice ma facilmente loggabile e meno efficiente

Version detection

Version Detection viene usato per identificare i servizi e le versioni su porte aperte.

Dopo aver identificato le porte aperte, invia richieste mirate per capire:

- Quale servizio è in esecuzione (es. http, ssh, ftp ect)
- La versione esatta del software (es. Apache 2.4.58, OpenSSH 8.2p1)

Usa un database di firme e risposte tipiche per confrontare con i dati ricevuti. Può anche riconoscere banner applicativi, protocolli particolari e, a volte, il nome del dispositivo.

Ecco come deve essere lanciato il comando su Linux: ù

```
sudo nmap -sV <target>
```

Nel mio caso:

```
sudo nmap 192.168.50.5
```

Di seguito analizziamo l'output:

```
(kali@kali)-[~]
└─$ nmap -sV 192.168.50.5
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 20:07 CEST
Nmap scan report for 192.168.50.5
Host is up (0.056s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login        OpenBSD or Solaris rlogind
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)
6667/tcp  open  irc          UnrealIRCd
8009/tcp  open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:2C:82:94 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.39 seconds
```

Come ci aspettavamo, l'output mostra chiaramente:

- **PORT** → le porte scansionate
- **STATE** → lo stato della porta
- **SERVICE** → il servizio attivo sulla porta
- **VERSION** → la versione del servizio attivo sulla porta

Conclusioni

La campagna di scansione ha permesso di identificare le porte aperte, i servizi esposti e le versioni software presenti sui sistemi analizzati, oltre ad ottenere indicazioni utili sul sistema operativo tramite fingerprinting.

Le scansioni SYN (**-ss**) si sono rivelate più rapide e meno evidenti nei log rispetto alle scansioni TCP connect (**-sT**), mentre i risultati sullo stato delle porte (aperte/chiuso/filtrate) sono risultate sostanzialmente coerenti tra le due metodologie; le differenze principali risiedono nella visibilità dell'attività nei log e nel requisito di privilegi.

L'identificazione delle versioni (`-sv`) ha confermato versioni datate di alcuni servizi che rappresentano potenziali rischi di vulnerabilità se non aggiornati; dove possibile, si raccomanda un controllo approfondito tramite tool e script NSE mirati.

L'OS fingerprinting (`-o`) ha fornito utili indicazioni sul tipo di sistema e sulla sua exposability, ma i risultati vanno interpretati con cautela quando sono presenti firewall o meccanismi di evasione.