



Penetration Testing & Ethical Hacking

Enumerating Target e Port Scanning

Parte 2

Arcangelo Castiglione
arcastiglione@unisa.it

Nmap

Opzioni per Scansioni TCP – TCP Connect (normal user)

- Nmap utilizza di default una scansione basata su *TCP SYN*
 - Nota come SYN scan, half-open o SYN stealth (poiché essa non completa il three-way handshake)
- Per poter utilizzare la maggior parte delle opzioni di scansione fornite da nmap è necessario essere utenti privilegiati (*root* o *amministratore*)
 - Altrimenti si ottiene un errore

```
kali㉿kali:~$ nmap -ss 10.0.2.10
You requested a scan type which requires root privileges.
QUITTING!
kali㉿kali:~$
```

Errore

- Se invece Nmap è eseguito da **utenti non privilegiati** viene utilizzata di **default una scansione nota come TCP Connect**



Nmap

Opzioni per Scansioni TCP – TCP Connect (normal user)

- Nmap utilizza di default una scansione basata su *TCP SYN*
 - Nota come SYN scan, half-open o SYN stealth (poiché essa non completa il three-way handshake)

N.B. in Kali, da kali-linux-2024.4, la modalità di default è la SYN Scan per tutte le tipologie di utente

- Per poter eseguire un scan da nmap è necessario essere utenti privilegiati (root o amministratore)
 - Altrimenti si ottiene un errore

```
kali㉿kali:~$ nmap -ss 10.0.2.10
You requested a scan type which requires root privileges.
QUITTING!
kali㉿kali:~$
```

Errore

- Se invece Nmap è eseguito da utenti non privilegiati viene utilizzata di default una scansione nota come *TCP Connect*



Nmap

Opzioni per Scansioni TCP – TCP Connect (normal user)

- Nmap tramite il Sistema Operativo su cui è eseguito stabilisce una connessione con la macchina target
 - Invocando la system call «*connect()*»

- Gli **svantaggi** di questa scansione sono che essa richiede
 - Generalmente più tempo per essere completata
 - Di generare più pacchetti per ottenere informazioni

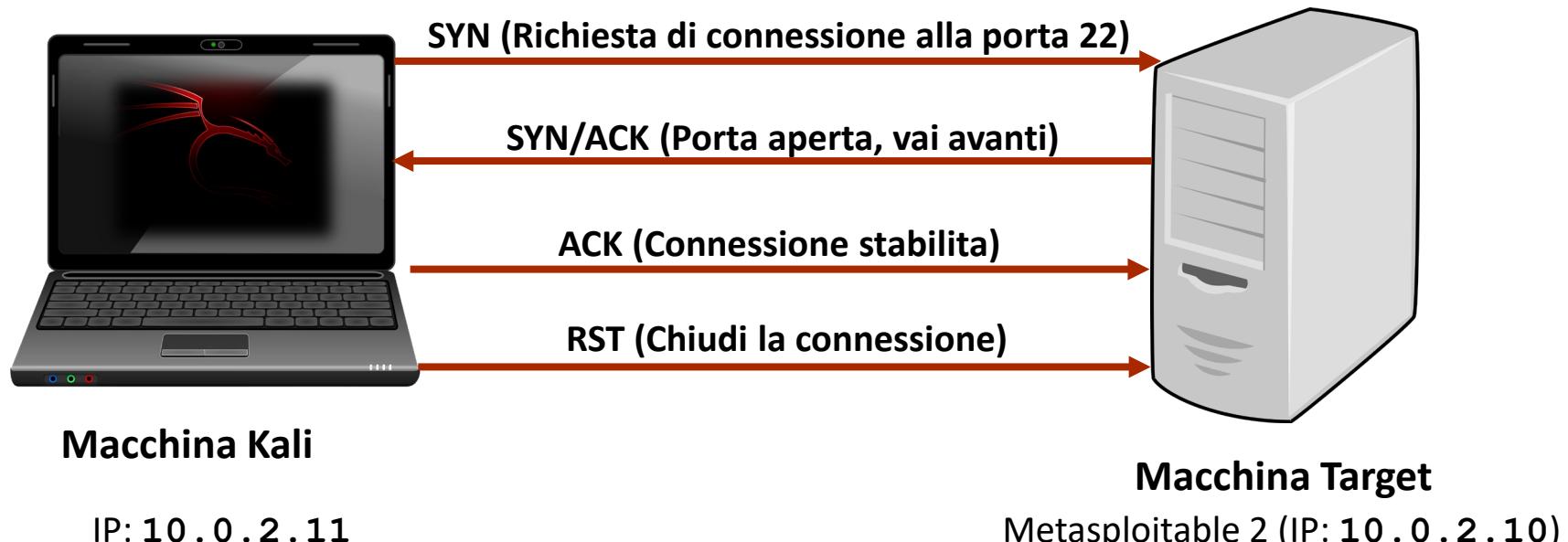
- Il **principale vantaggio** di tale scansione è che essa non «desta sospetti» verso la macchina target
 - Apparirà come una normale connessione *TCP* verso un servizio di rete



Nmap

Opzioni per Scansioni TCP – TCP Connect (normal user)

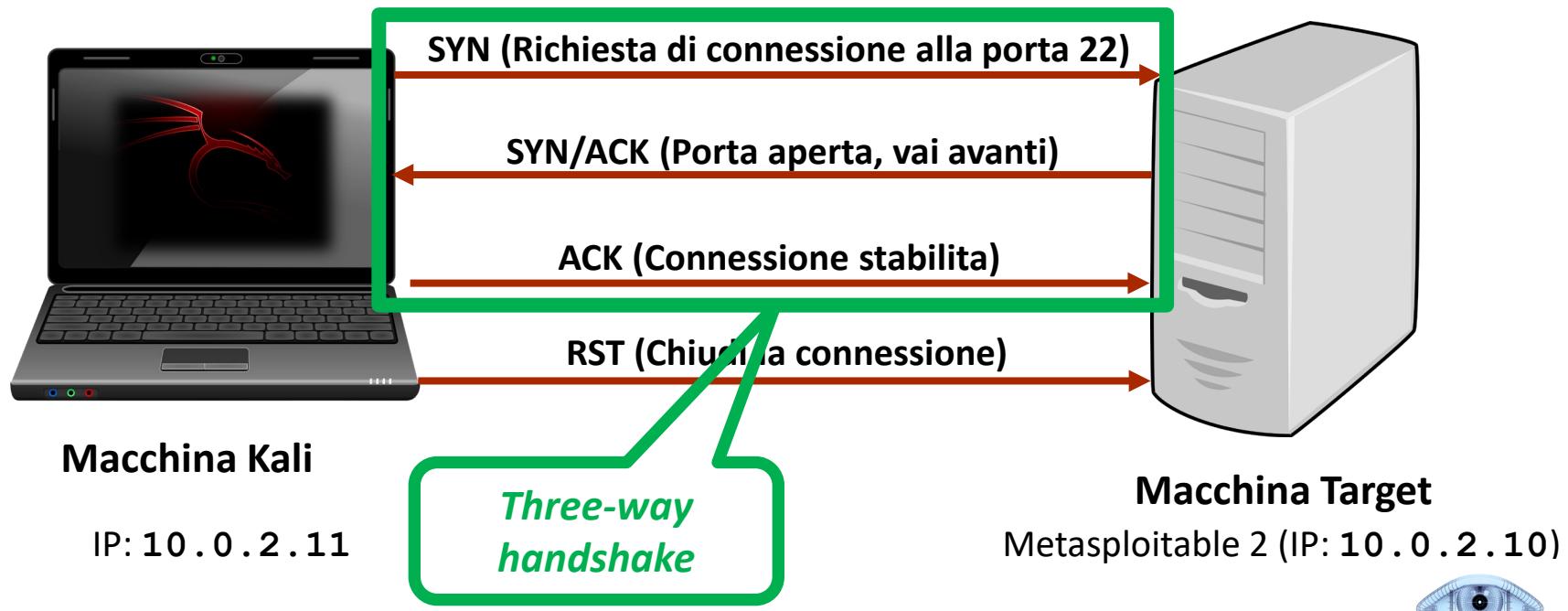
- Caso 1: Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)



Nmap

Opzioni per Scansioni TCP – TCP Connect (normal user)

- Caso 1: Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)



Nmap

TCP Connect Scan – Caratteristiche

- **Opzione -sT**
 - Questa opzione permette di effettuare il *three-way handshake* verso ogni porta da scansionare
 - Se la connessione è stabilita, la porta è considerata «aperta»
 - **N.B.** Poiché deve effettuare il *three-way handshake* verso ogni porta, questo tipo di scansione potrebbe essere lento e molto probabilmente verrà registrato nei log di sistema dalla macchina target
 - Opzione di **scansione predefinita** quando Nmap è eseguito da un **utente che non ha privilegi di root o di amministratore**



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22`

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sT 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sT 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-26 17:37 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0048s latency).
Not shown: 977 closed ports
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
```



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:37:06.005491 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [S], seq 4039503064, wi
n 64240, options [mss 1460,sackOK,TS val 3402587536 ecr 0,nop,wscale 7], length
0
17:37:06.006111 IP 10.0.2.10.22 > 10.0.2.11.54142: Flags [S.], seq 1879068240, a
ck 4039503065, win 5792, options [mss 1460,sackOK,TS val 4294955748 ecr 34025875
36,nop,wscale 5], length 0
17:37:06.006126 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [.], ack 1, win 502, op
tions [nop,nop,TS val 3402587536 ecr 4294955748], length 0
17:37:06.007344 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [R.], seq 1, ack 1, win
502, options [nop,nop,TS val 3402587537 ecr 4294955748], length 0
```



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:37:06.005491 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [S], seq 4039503064, wi
n 64240, options [mss 1460,sackOK,TS val 3402587536 ecr 0,nop,wscale 7], length
0
17:37:06.006111 IP 10.0.2.10.22 > 10.0.2.11.54142: Flags [S.], seq 1879068240, a
ck 4039503065, win 5792, options [mss 1460,sackOK,TS val 4294955748 ecr 34025875
36,nop,wscale 5], length 0
17:37:06.006126 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [.], ack 1, win 502, op
tions [nop,nop,TS val 3402587536 ecr 4294955748], length 0
17:37:06.007344 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [R.], seq 1, ack 1, win
502, options [nop,nop,TS val 3402587537 ecr 4294955748], length 0
```

- La macchina Kali invia
 - Un pacchetto contenente il flag SYN = [S] (Start Connection)
 - Il numero di sequenza (ISN) 4039503064



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:37:06.005491 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [S], seq 4039503064, wi
n 64240, options [mss 1460,sackOK,TS val 3402587536 ecr 0,nop,wscale 7], length
0
17:37:06.006111 IP 10.0.2.10.22 > 10.0.2.11.54142: Flags [S.], seq 1879068240, a
ck 4039503065, win 5792, options [mss 1460,sackOK,TS val 4294955748 ecr 34025875
36,nop,wscale 5], length 0
17:37:06.006126 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [.], ack 1, win 502, op
tions [nop,nop,TS val 3402587536 ecr 4294955748], length 0
17:37:06.007344 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [R.], seq 1, ack 1, win
0, options [nop,nop,TS val 3402587536 ecr 4294955748], length 0
```

- La macchina target risponde con
 - Un pacchetto contenente il flag SYN-ACK = [S.] (SynAck Packet)
 - Il numero di sequenza (ISN) 1879068240
 - Un ACK al numero di sequenza ricevuto dalla macchina Kali
 - $4039503064 + 1 = 4039503065$



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type ENCAPSULATED Ethernet
17:37:06.005491 IP 10.0.2.11.  ➤ La macchina Kali invia un pacchetto contenente il flag ACK = [.] 
n 64240, options [mss 1460,sackOK,TS val 4294955748 ecr 0,nop,wscale 7], length
0
17:37:06.006111 IP 10.0.2.10.22 > 10.0.2.11.54142: Flags [S.], seq 1879068240, a
ck 4039503065, win 5792, options [mss 1460,sackOK,TS val 4294955748 ecr 34025875
36,nop,wscale 5], length 0
17:37:06.006126 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [.], ack 1, win 502, op
tions [nop,nop,TS val 3402587536 ecr 4294955748], length 0
17:37:06.007344 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [R.], seq 1, ack 1, win
502, options [nop,nop,TS val 3402587537 ecr 4294955748], length 0
```

Quindi il three-way handshake è completato...



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:37:06.005491 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [S], seq 4039503064, wi
➤ La macchina Kali invia un pacchetto contenente il flag RST-ACK = [R.] (RstAck Packet)
17:37:06.005491 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [S.], seq 1879068240, ack 4039503065, options [mss 1460,sackOK,TS val 4294955748 ecr 34025875
36,nop,wscale 5], length 0
17:37:06.006126 IP 10.0.2.10.22 > 10.0.2.11.54142: Flags [.], ack 1, win 502, options [nop,nop,TS val 3402587535 ecr 4294955748], length 0
17:37:06.007344 IP 10.0.2.11.54142 > 10.0.2.10.22: Flags [R.], seq 1, ack 1, win 502, options [nop,nop,TS val 3402587537 ecr 4294955748], length 0
```

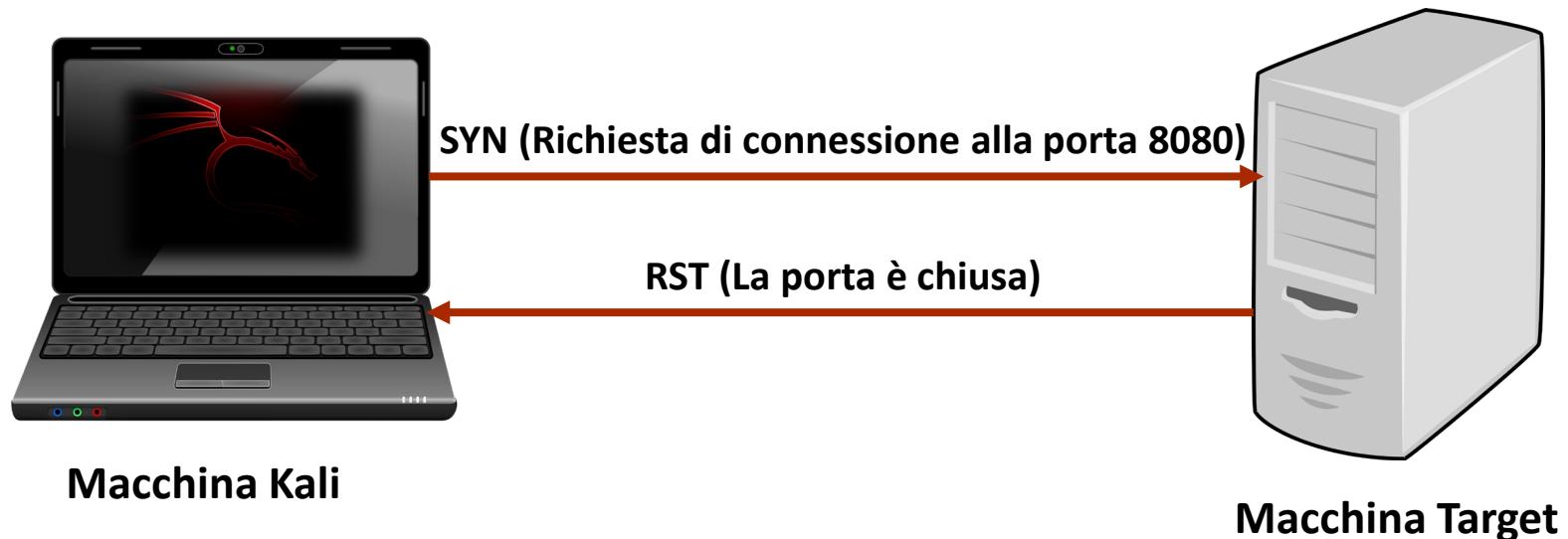
N.B. **seq 1 ed ack 1** subito dopo l'istaurazione della connessione significa che l'handshake iniziale è completo ed entrambe le parti sono pronte per iniziare a trasmettere dati



Nmap

Traffico Generato da una TCP Connect Scan

➤ **Caso 2: Porta Chiusa**



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080`



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sT 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sT 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-26 17:37 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0048s latency).

Not shown: 977 closed ports
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
```



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di `tcpdump` possiamo osservare quanto segue

```
07:04:37.568061 IP 10.0.2.11.51970 > 10.0.2.10.8080: Flags [S], seq 3319017068,  
win 64240, options [mss 1460,sackOK,TS val 449785433 ecr 0,nop,wscale 7], length  
0  
07:04:37.568787 IP 10.0.2.10.8080 > 10.0.2.11.51970: Flags [R.], seq 0, ack 3319  
017069, win 0, length 0
```

- La macchina Kali invia
 - Un pacchetto contenente il flag SYN = [S] (Start Connection)
 - Il numero di sequenza (ISN) 3319017068



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di `tcpdump` possiamo osservare quanto segue

```
07:04:37.568061 IP 10.0.2.11.51970 > 10.0.2.10.8080: Flags [S], seq 3319017068,  
win 64240, options [mss 1460,sackOK,TS val 449785433 ecr 0,nop,wscale 7], length  
0  
07:04:37.568787 IP 10.0.2.10.8080 > 10.0.2.11.51970: Flags [R.], seq 0, ack 3319  
017069, win 0, length 0
```

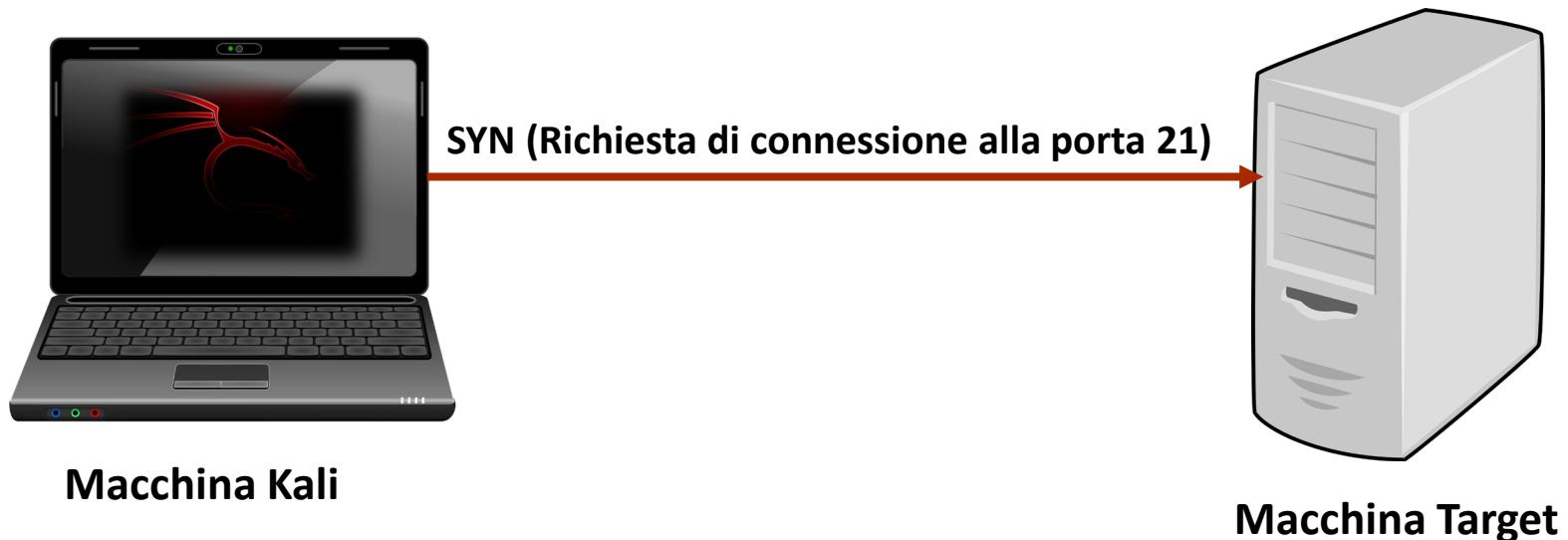
- La macchina target risponde con
 - Un pacchetto contenente il flag RST-ACK = [R.] (RstAcK Packet)
 - Un ACK al numero di sequenza ricevuto dalla macchina Kali
 - $3319017068 + 1 = 3319017069$



Nmap

Traffico Generato da una TCP Connect Scan

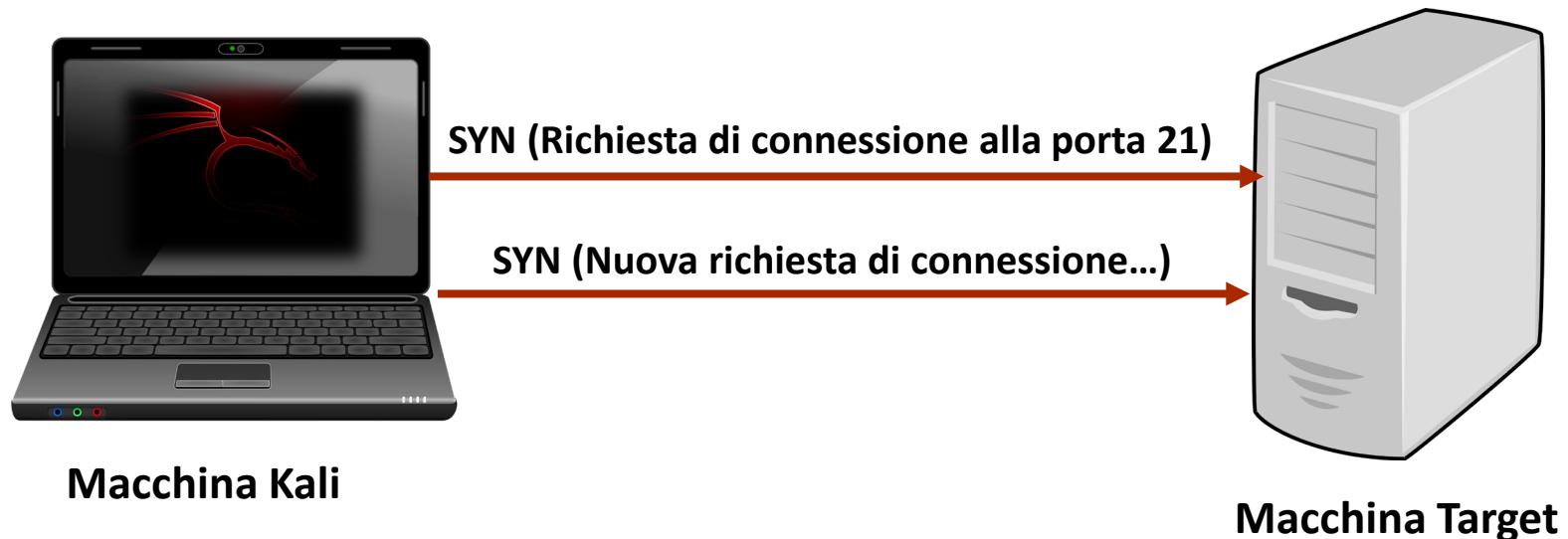
➤ Caso 3: Porta Filtrata



Nmap

Traffico Generato da una TCP Connect Scan

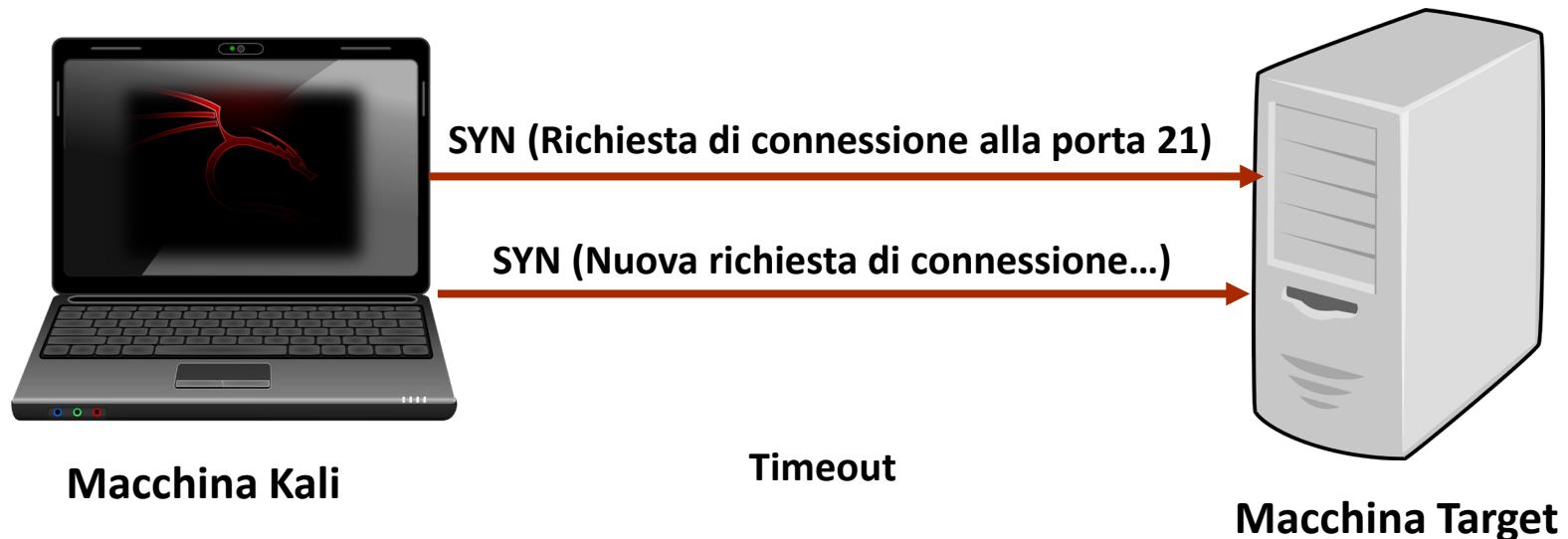
➤ Caso 3: Porta Filtrata



Nmap

Traffico Generato da una TCP Connect Scan

➤ Caso 3: Porta Filtrata



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)

- Tramite il firewall **iptables** filtriamo tutte le porte, consentendo solo traffico in ingresso TCP verso la porta 22 della macchina target
 - Tutto il resto del traffico sarà bloccato dal firewall



Nmap

Traffico Generato da una TCP Connect Scan

- Macchina target
 - Metasploitable 2
 - Indirizzo IP della macchina target: **10.0.2.10**
- Configuriamo il **firewall** (comando **iptables**) sulla **macchina target** affinché esso
 - Cancelli eventuali politiche di filtro definite precedentemente
 - **iptables -F**
 - **iptables -t nat -F**
 - **iptables -X**
 - Accetti tutti i pacchetti relativi a connessioni sulla porta *TCP* 22 e scarti tutti gli altri
 - **iptables -P FORWARD DROP**
 - **iptables -P INPUT DROP**
 - **iptables -P OUTPUT ACCEPT**
 - **iptables -A INPUT -p tcp --dport 22 -j ACCEPT**



Nmap

Traffico Generato da una TCP Connect Scan

- Macchina target
 - Metasploitable 2
 - Indirizzo IP della macchina target: **10.0.2.10**
- Configuriamo il **firewall** (comando **iptables**) sulla **macchina target** affinché esso
 - Cancelli eventuali politiche di filtro definite precedentemente
 - **iptables -F**
 - **iptables -t nat -F**
 - **iptables -X**
 - Accetti tutti i pacchetti relativi a connessioni sulla porta *TCP* 22 e scarti tutti gli altri
 - **iptables -P FORWARD DROP**
 - **iptables -P INPUT DROP**
 - **iptables -P OUTPUT ACCEPT**
 - **iptables -A INPUT -p tcp --dport 22 -j ACCEPT**



Per maggiori informazioni sul comando **iptables**, digitare **man iptables**

Nmap

Traffico Generato da una TCP Connect Scan

- I comandi **iptables** possono essere inseriti in uno script
 - Ad esempio chiamato **iptables.sh**

```
iptables -F
iptables -t nat -F
iptables -X
iptables -P FORWARD DROP
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Contenuto dello script **iptables.sh**

- Vanno impostati i permessi di esecuzione sullo script, prima di eseguirlo (**chmod 755 iptables.sh**)

Eseguiamo lo script: **./iptables.sh**



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.21`



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - `nmap -sT 10.0.2.10`

```
root@kali:~# nmap -sT 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 15:58 EDT
Nmap scan report for 10.0.2.10
Host is up (0.00069s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 08:00:27:80:B2:70 (Oracle VM VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 4.88 seconds
```



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.21
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:01:30.994272 IP 10.0.2.11.57516 > 10.0.2.10.21: Flags [S], seq 3784679982, wi
n 64240, options [mss 1460,sackOK,TS val 3939596135 ecr 0,nop,wscale 7], length
0
16:01:32.095921 IP 10.0.2.11.57530 > 10.0.2.10.21: Flags [S], seq 4034080752, wi
n 64240, options [mss 1460,sackOK,TS val 3939597237 ecr 0,nop,wscale 7], length
0
```

- La macchina Kali invia
 - Un pacchetto contenente il flag SYN = [S] (Start Connection)
 - Il numero di sequenza (ISN) 3784679982



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.21
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:01:30.994272 IP 10.0.2.11.57516 > 10.0.2.10.21: Flags [S], seq 3784679982, wi
n 64240, options [mss 1460,sackOK,TS val 3939596135 ecr 0,nop,wscale 7], length
0
16:01:32.095921 IP 10.0.2.11.57530 > 10.0.2.10.21: Flags [S], seq 4034080752, wi
n 64240, options [mss 1460,sackOK,TS val 3939597237 ecr 0,nop,wscale 7], length
0
```

- La macchina Kali invia
 - Un nuovo pacchetto contenente il flag SYN = [S] (Start Connection)
 - Il numero di sequenza (ISN) 4034080752



Nmap

Traffico Generato da una TCP Connect Scan

- **Caso 3:** Traffico generato tra la macchina Kali e la macchina target sulla porta 21 (*Porta Filtrata*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.21
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:01:30.994272 IP 10.0.2.11.57516 > 10.0.2.10.21: Flags [S], seq 3784679982, wi
n 64240, options [mss 1460,sackOK,TS val 3939596135 ecr 0,nop,wscale 7], length
0
16:01:32.095921 IP 10.0.2.11.57530 > 10.0.2.10.21: Flags [S], seq 4034080752, wi
n 64240, options [mss 1460,sackOK,TS val 3939597237 ecr 0,nop,wscale 7], length
0
```

- Non avendo ricevuto alcuna risposta entro una certa soglia di timeout, nmap passa alla scansione della porta successiva



Nmap

Altre Scansioni Predefinite: TCP NULL, FIN ed XMAS

- **TCP NULL Scan (Opzione `-sN`)**
 - Non imposta alcun bit (*flag*) di controllo
- **FIN Scan (Opzione `-sF`)**
 - Imposta solo il bit (*flag*) **FIN**
- **XMAS Scan (Opzione `-sX`)**
 - Imposta i bit (*flag*) **FIN**, **PSH** e **URG**



Nmap

Altre Scansioni Predefinite : TCP NULL, FIN ed XMAS

- Le tre tipologie di scansione (TCP NULL, FIN ed XMAS)
 - Se non ricevono **risposta** considerano la porta «**aperta**» o «**filtrata**» («**open** | **filtered**»)
 - Se ricevono un pacchetto **RST** come **risposta** considerano la porta «**chiusa**» («**closed**»)



Nmap

TCP NULL Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22`

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```



Nmap

TCP NULL Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sN 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sN 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 06:02 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0016s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
1524/tcp  open|filtered  ingreslock
2049/tcp  open|filtered  nfs
```



Nmap

TCP NULL Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

tcpdump

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10:18:21.040765 IP 10.0.2.11.35807 > 10.0.2.10.22: Flags [none], win 1024, length 0
```



Nmap

TCP NULL Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

tcpdump

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10:18:21.040765 IP 10.0.2.11.35807 > 10.0.2.10.22: Flags [none], win 1024, length 0
```

- Kali invia un pacchetto in cui non è impostato alcun bit di controllo (*NULL Packet*)
- Non viene ricevuta alcuna risposta da parte della macchina target
 - La porta è quindi considerata «aperta» o «filtrata»

nmap

PORT	STATE	SERVICE
21/tcp	open filtered	ftp
22/tcp	open filtered	ssh
23/tcp	open filtered	telnet

Nmap

TCP NULL Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080`



Nmap

TCP NULL Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sN 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sN 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 06:02 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0016s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
1524/tcp  open|filtered  ingreslock
2049/tcp  open|filtered  nfs
```



Nmap

TCP NULL Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di `tcpdump` possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:02:20.883661 IP 10.0.2.11.47858 > 10.0.2.10.8080: Flags [none], win 1024, len
ath 0
06:02:20.884495 IP 10.0.2.10.8080 > 10.0.2.11.47858: Flags [R.], seq 0, ack 1776
932044, win 0, length 0
```

- Kali invia un pacchetto in cui non è impostato alcun bit di controllo



Nmap

TCP NULL Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:02:20.883661 IP 10.0.2.11.47858 > 10.0.2.10.8080: Flags [none], win 1024, length 0
06:02:20.884495 IP 10.0.2.10.8080 > 10.0.2.11.47858: Flags [R.], seq 0, ack 1776
932044, win 0, length 0
```

- La macchina target invia un pacchetto contenente il flag RST-ACK = [R.] (RstAck Packet)



Nmap

TCP FIN Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22`

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```



Nmap

TCP FIN Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sF 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sF 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 06:04 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0011s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
```



Nmap

TCP FIN Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:04:36.497497 IP 10.0.2.11.62500 > 10.0.2.10.22: Flags [F], seq 894489092, win
1024, length 0
```



Nmap

TCP FIN Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:04:36.497497 IP 10.0.2.11.62500 > 10.0.2.10.22: Flags [F], seq 894489092, win
1024, length 0
```

- Kali invia un pacchetto in cui è impostato il flag Fin = [F] (Finish Connection) ed il numero di sequenza (ISN) 894489092
- Non viene ricevuta alcuna risposta da parte della macchina target
 - La porta è considerata «aperta» o «filtrata»



Nmap

TCP FIN Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080`



Nmap

TCP FIN Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sF 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sF 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 06:04 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0011s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
```



Nmap

TCP FIN Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:06:43.368597 IP 10.0.2.11.55832 > 10.0.2.10.8080: Flags [F], seq 3380948498,
win 1024, length 0
06:06:43.369169 IP 10.0.2.10.8080 > 10.0.2.11.55832: Flags [R.], seq 0, ack 3380
948499, win 0, length 0
```

- Kali invia un pacchetto in cui è impostato il flag Fin = [F] (Finish Connection) ed il numero di sequenza (ISN) 3380948498



Nmap

TCP FIN Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di `tcpdump` possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:06:43.368597 IP 10.0.2.11.55832 > 10.0.2.10.8080: Flags [F], seq 3380948498,
win 1024, length 0
06:06:43.369169 IP 10.0.2.10.8080 > 10.0.2.11.55832: Flags [R.], seq 0, ack 3380
948499, win 0, length 0
```

- La macchina target invia un pacchetto contenente il flag RST-ACK = [R.] (RstAck Packet)



Nmap

TCP XMAS Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22`

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```



Nmap

TCP XMAS Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sX 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sX 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 16:40 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0033s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
```



Nmap

TCP XMAS Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:42:41.974141 IP 10.0.2.11.50748 > 10.0.2.10.22: Flags [FPU], seq 2117049082,
win 1024, urg 0, length 0
```



Nmap

TCP XMAS Scan – Esempio

- **Caso 1:** Traffico generato tra la macchina Kali e la macchina target sulla porta 22 (*Porta Aperta*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.22
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0. link-type EN10MB (Ethernet). capture size 262144 bytes
16:42:41.974141 IP 10.0.2.11.50748 > 10.0.2.10.22: Flags [FPU], seq 2117049082,
win 1024, urg 0, length 0
```

- Kali invia un pacchetto in cui sono impostati i flag FIN (F), PSH (P) ed URG (U), oltre al numero di sequenza (ISN) 2117049082
- Non viene ricevuta alcuna risposta da parte della macchina target
 - La porta è considerata «aperta» o «filtrata»



Nmap

TCP XMAS Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo `tcpdump` con gli opportuni parametri
 - `tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080`



Nmap

TCP XMAS Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Avviamo **nmap** usando una nuova finestra (o un nuovo Tab) del Terminale ed attendiamo la fine della scansione
 - **nmap -sX 10.0.2.10**

Output Parziale

```
root@kali:~# nmap -sX 10.0.2.10
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-27 16:40 EDT
Nmap scan report for 10.0.2.10
Host is up (0.0033s latency).
Not shown: 977 closed ports
PORT      STATE      SERVICE
21/tcp    open|filtered  ftp
22/tcp    open|filtered  ssh
23/tcp    open|filtered  telnet
25/tcp    open|filtered  smtp
53/tcp    open|filtered  domain
80/tcp    open|filtered  http
111/tcp   open|filtered  rpcbind
139/tcp   open|filtered  netbios-ssn
445/tcp   open|filtered  microsoft-ds
512/tcp   open|filtered  exec
513/tcp   open|filtered  login
514/tcp   open|filtered  shell
1099/tcp  open|filtered  rmiregistry
```



Nmap

TCP XMAS Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:05:50.735580 IP 10.0.2.11.39988 > 10.0.2.10.8080: Flags [FPU], seq 42137227,
win 1024, urg 0, length 0
17:05:50.736444 IP 10.0.2.10.8080 > 10.0.2.11.39988: Flags [R.], seq 0, ack 4213
7228, win 0, length 0
```

- Kali invia un pacchetto in cui sono impostati i flag FIN (F), PSH (P) ed URG (U), oltre al numero di sequenza (ISN) 42137227



Nmap

TCP XMAS Scan – Esempio

- **Caso 2:** Traffico generato tra la macchina Kali e la macchina target sulla porta 8080 (*Porta Chiusa*)
- Analizzando l'output di **tcpdump** possiamo osservare quanto segue

```
root@kali:~# tcpdump -nnX tcp and host 10.0.2.11 | grep 10.0.2.10.8080
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:05:50.735580 IP 10.0.2.11.39988 > 10.0.2.10.8080: Flags [FPU], seq 42137227,
win 1024. urg 0. length 0
17:05:50.736444 IP 10.0.2.10.8080 > 10.0.2.11.39988: Flags [R.], seq 0, ack 42137228,
win 0, length 0
```

- La macchina target invia un pacchetto contenente il flag RST-ACK = [R.] (RstAck Packet)



Nmap

Altre Scansioni Predefinite – TCP Maimon Scan

- Opzione **-sM**
 - Scansione creata da Uriel Maimon
 - Invia un pacchetto con il bit flag ***FIN/ACK*** impostato
 - I sistemi basati su **BSD (Berkeley Software Distribution)*
 - Scarteranno il pacchetto se la porta è «aperta»
 - Risponderanno con ***RST*** se la porta è «chiusa»
 - Risponderanno con ***ICMP unreachable error*** (type 3, code 1, 2, 3, 9, 10, or 13) se la porta è «filtrata»



Nmap

Altre Scansioni Predefinite – TCP ACK Scan

➤ Opzione **-sA**

- Scansione utilizzata per determinare se un firewall sta filtrando il traffico TCP in ingresso/uscita
- Invia un pacchetto con il solo bit flag **ACK** impostato
 - Se viene ricevuto un **RST** significa che il pacchetto è arrivato, quindi il traffico TCP passa
 - Porta «**unfiltered**»
 - Se non viene ricevuta risposta o viene ricevuto un messaggio **ICMP unreachable error** significa che il firewall ha bloccato il pacchetto
 - Porta «**filtered**»



Nmap

Altre Scansioni Predefinite – TCP Window Scan

➤ Opzione **-sW**

- Nmap invia pacchetti ACK verso il target
- Quando riceve una risposta (tipicamente un **RST**), analizza il valore della *TCP Window Size* nel pacchetto di risposta
 - Se tale campo ha valore positivo, allora la porta è «aperta»
 - Se tale campo ha valore zero, allora la porta è «chiusa»



Nmap

Altre Scansioni Predefinite – TCP Idle Scan

- **Opzione -sI**
 - Scansione che non invia nessun pacchetto alla macchina target
 - I pacchetti relativi alla scansione «rimbalzeranno» su un determinato *host zombie*
 - Un *Intrusion Detection System (IDS)* potrebbe accorgersi dell'host zombie



Nmap

Scansione TCP Personalizzata

- Nmap consente anche di creare scansioni personalizzate
 - Mediante l'opzione **--scanflags**
 - L'argomento di tale opzione può essere un valore numerico o un nome simbolico
 - Come argomento può essere usato
 - Una qualsiasi combinazione (in qualsiasi ordine) degli 8 bit flag **URG**, **ACK**, **PSH**, **RST**, **SYN**, **FIN**, **ECE**, **CWR**
 - Ma anche **ALL** e **NONE**
 - **Esempio:** **--scanflags URGACKPSH**
 - Imposta una scansione che utilizza i bit flag **URG**, **ACK** e **PSH**
 - Le scansioni personalizzate possono essere molto utili
 - Per provare a «bypassare» i controlli effettuati dai firewall
 - Quando è necessario analizzare servizi di rete «poco comuni» o «proprietari»

Nmap

Port Scanning basato su UDP – Logica di Funzionamento

- Viene inviato un pacchetto UDP ad ogni porta da scansionare
- La macchina target può «rispondere» in vari modi
 - Pacchetto *UDP*
 - Denota che la porta è «**aperta**»
 - Pacchetto contenente il messaggio «**ICMP Port Unreachable**»
 - Denota che la porta è «**chiusa**»
 - Altri errori del tipo «**ICMP Unreachable**»
 - Denota che la porta è «**filtrata**»
 - Nessuna risposta
 - Denota che la porta potrebbe essere «**chiusa**» oppure
 - Denota che il pacchetto *UDP* in ingresso sulla macchina target potrebbe essere «**filtrato**» o che la risposta della macchina target potrebbe essere «**filtrata**»

Nmap

Port Scanning basato su UDP

- Alcuni servizi di rete utilizzano *UDP*
- **Nmap** fornisce **una sola tipologia di scansione per *UDP***
 - Opzione **-sU**
- **N.B.** La scansione *UDP* è molto lenta
 - Ciò è dovuto principalmente al fatto che il kernel Linux limita l'invio del messaggio «**ICMP Port Unreachable**»
- La scansione di tutte le porte richiede molto tempo



Nmap

Port Scanning basato su UDP

- Diversi modi per mitigare questo problema
 - Effettuare scansioni *UDP* in parallelo
 - Effettuare prima la scansione delle porte più popolari
 - Utilizzare l'opzione **--host-timeout** per «scartare» gli host «lenti»



Nmap

Port Scanning basato su UDP – Esempio

- Macchina target: Metasploitable 2 (IP: 10.0.2.6)

- Porte da analizzare: 53, 68, 69, 161, 137 e 138



Nmap

Port Scanning basato su UDP – Esempio

- Scansione *UDP* sulle porte **53, 68, 69, 161, 137 e 138**
- **nmap -sU 10.0.2.6 -p 53,68,69,161,137,138**

```
root@kali:~# nmap -sU 10.0.2.6 -p 53,68,69,161,137,138
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-25 22:41 CET
Nmap scan report for 10.0.2.6
Host is up (0.00060s latency).

PORT      STATE            SERVICE
53/udp    open             domain
68/udp    open|filtered   dhcpc
69/udp    open|filtered   tftp
137/udp   open             netbios-ns
138/udp   open|filtered   netbios-dgm
161/udp   closed           snmp
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 7.70 seconds
```



Nmap

TCP vs. UDP Port Scanning

- Il port scanning basato su *UDP* è meno affidabile di quello basato su *TCP*
 - A volte la porta *UDP* è aperta ma il servizio «in ascolto» su tale porta è in attesa di uno specifico payload *UDP*
 - In questo caso, il servizio non invierà alcuna risposta

Nmap

Bypassare l'Host (Target) Discovery

- Se una macchina target blocca le richieste di ping (*ICMP*), Nmap potrebbe considerare tale macchina come non attiva
 - Non effettuando ulteriori analisi, quali
 - Port Scanning
 - Rilevazione delle Versioni dei Servizi
 - Rilevazione del Sistema Operativo
- Mediante l'opzione **-Pn**
 - Nmap assumerà che la macchina target sia disponibile ed eseguirà le scansioni su tale macchina
 - Anche se tale macchina appare come non attiva



Nmap

Opzioni di Temporizzazione (o di Timing)

- Nmap fornisce **6 modalità di timing**, che possono essere impostate mediante l'opzione **-T**
 - **0 (paranoid)**: un pacchetto è inviato circa ogni 5 minuti
 - I pacchetti sono inviati in serie
 - Questa modalità è utile per evitare il rilevamento da parte di IDS
 - **1 (sneaky)**: un pacchetto è inviato circa ogni 15 secondi
 - Non ci sono pacchetti inviati in parallelo
 - **2 (polite)**: un pacchetto è inviato circa ogni 0.4 secondi
 - Non ci sono pacchetti inviati in parallelo



Nmap

Opzioni di Temporizzazione (o di Timing)

- Nmap fornisce **6 modalità di timing**, che possono essere impostate mediante l'opzione **-T**
 - **3 (normal)**: vengono inviati più pacchetti a più destinazioni contemporaneamente
 - Modalità di temporizzazione predefinita utilizzata da Nmap
 - Bilancia il tempo impiegato per la scansione ed il carico di rete
 - Raccomandata per scansioni sulla rete Internet
 - **4 (aggressive)**: Nmap scansiona un determinato host per un «breve lasso di tempo» prima di passare alla scansione della successiva macchina target
 - Raccomandata per scansioni su reti locali



Nmap

Opzioni di Temporizzazione (o di Timing)

- Nmap fornisce **6 modalità di timing**, che possono essere impostate mediante l'opzione **-T**
- **5 (insane):** Nmap scansiona un determinato host per un «brevissimo lasso di tempo» prima di passare alla scansione della successiva macchina target
 - Raccomandata per scansioni su reti definite all'interno di una singola macchina host
 - Ad esempio, la rete definita all'interno di VirtualBox



Nmap

Opzioni di Temporizzazione (o di Timing)

- Nmap fornisce **6 modalità di timing**, che possono essere impostate mediante l'opzione **-T**
- **5 (insane):** Nmap scansiona un determinato host per un «brevissimo lasso di tempo» prima di passare alla scansione della successiva macchina target
 - Raccomandata per scansioni su reti definite all'interno di una singola macchina host
 - Ad esempio, la rete definita all'interno di VirtualBox

N.B. Nella maggior parte dei casi, tali opzioni sono più utili per «rallentare» il processo di scansione piuttosto che per «velocizzarlo»



Nmap

Opzioni di Temporizzazione (o di Timing)

Table 6.3. Timing templates and their effects

	T0	T1	T2	T3	T4	T5
Name	Paranoid	Sneaky	Polite	Normal	Aggressive	Insane
min-rtt-timeout	100 ms	100 ms	100 ms	100 ms	100 ms	50 ms
max-rtt-timeout	5 minutes	15 seconds	10 seconds	10 seconds	1250 ms	300 ms
initial-rtt-timeout	5 minutes	15 seconds	1 second	1 second	500 ms	250 ms
max-retries	10	10	10	10	6	2
Initial (and minimum) scan delay (--scan-delay)	5 minutes	15 seconds	400 ms	0	0	0
Maximum TCP scan delay	5 minutes	15,000	1 second	1 second	10 ms	5 ms
Maximum UDP scan delay	5 minutes	15 seconds	1 second	1 second	1 second	1 second
host-timeout	0	0	0	0	0	15 minutes
script-timeout	0	0	0	0	0	10 minutes
min-parallelism	Dynamic, not affected by timing templates					
max-parallelism	1	1	1	Dynamic	Dynamic	Dynamic
min-hostgroup	Dynamic, not affected by timing templates					
max-hostgroup	Dynamic, not affected by timing templates					
min-rate	No minimum rate limit					
max-rate	No maximum rate limit					
defeat-rst-ratelimit	Not enabled by default					

<https://nmap.org/book/performance-timing-templates.html>



Nmap

Opzioni per Firewall/IDS Evasion

- Alcune macchine target potrebbero essere protette da *firewall* e IDS/IPS

- Utilizzando le impostazioni predefinite di Nmap
 - I *firewall* e gli IDS/IPS potrebbero rilevare e talvolta bloccare la scansione
 - I risultati prodotti da una scansione potrebbero essere poco corretti o non esaustivi



Nmap

Opzioni per Firewall/IDS Evasion

- Nmap fornisce alcune opzioni per provare a «bypassare» i controlli messi in atto da *firewall* o *IDS/IPS*
 - **-f (*fragment packets*):** fa in modo che la scansione utilizzi pacchetti di dimensione più piccola rispetto a quella di default (*pacchetti frammentati*)
 - Specificando questa opzione, Nmap dividerà il pacchetto in 8 byte dopo l'header IP
 - **-mtu <val>:** permette di specificare la dimensione di frammentazione (*Maximum Transmission Unit - MTU*) di ciascun pacchetto
 - *MTU* deve essere un multiplo di 8, altrimenti Nmap restituirà un errore e terminerà la propria esecuzione



<https://nmap.org/man/it/man-bypass-firewalls-ids.html>

Nmap

Opzioni per Firewall/IDS Evasion

- Nmap fornisce alcune opzioni per provare a «bypassare» i controlli messi in atto da *firewall* o *IDS/IPS*
 - **-D (*decoy*)**: permette di utilizzare indirizzi IP «spoofati» per nascondere l'indirizzo IP del mittente
 - **-sI (*idle scan*)**: permette di effettuare la scansione servendosi di una macchina «zombie»
 - **-g <portnumber>**: permette di generare traffico da una porta specifica
 - Utile quando il firewall è impostato per consentire tutto il traffico in entrata proveniente da una porta specifica
 - **-data-length <num>**: indica a Nmap di aggiungere un certo numero di byte casuali a (quasi tutti) i pacchetti che invia e di non usare i valori specifici del protocollo
 - Nmap di default invia un payload casuale di lunghezza fissa

<https://nmap.org/man/it/man-bypass-firewalls-ids.html>



Nmap

Opzioni per Firewall/IDS Evasion

- Nmap fornisce alcune opzioni per provare a «bypassare» i controlli messi in atto da *firewall* o *IDS/IPS*
 - **-max (min) -parallelism <num>**: permette di regolare la parallelizzazione tra i vari *probe* di una scansione
 - **-scan-delay <time>**: permette di regolare la latenza tra i vari *probe* di una scansione
 - Opzione che può essere usata per eludere *IDS/IPS* che usano soglie per rilevare un'attività di port scanning



<https://nmap.org/man/it/man-bypass-firewalls-ids.html>

Nmap

Rilevazione della Versione dei Servizi

- Nmap può essere utilizzato per rilevare la **versione di un servizio di rete** (*Service and Version Detection*) sulla macchina target quando si esegue la scansione delle porte
 - Opzione **-sV**
- Informazione che sarà molto utile nella fase successiva, durante il processo di identificazione delle vulnerabilità (*Vulnerability Mapping*)



Nmap

Rilevazione della Versione dei Servizi – Esempio

➤ **nmap -sV 10.0.2.6 -p 22**

```
root@kali:~# nmap -sV 10.0.2.6 -p 22
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 12:43 CET
Nmap scan report for 10.0.2.6
Host is up (0.00037s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntul (protocol 2.0)
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at http
s://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.10 seconds
```

Sulla porta 22, esiste un servizio SSH, che utilizza la versione del software OpenSSH 4.7p1 ed il protocollo SSH 2.0



Nmap

Rilevazione della Versione dei Servizi – Esempio

➤ **nmap -sV 10.0.2.6 -p 22**

```
root@kali:~# nmap -sV 10.0.2.6 -p 22
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 12:43 CET
Nmap scan report for 10.0.2.6
Host is up (0.00037s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntul (protocol 2.0)
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at http
s://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.10 seconds
```

Service Banner

Sulla porta 22, esiste un servizio SSH, che utilizza la versione del software
OpenSSH 4.7p1 ed il protocollo SSH 2.0



Nmap

Rilevazione della Versione dei Servizi – Esempio

➤ `nmap -sV 10.0.2.6 -p 22`

```
root@kali:~# nmap -sV 10.0.2.6 -p 22
S 2019-03-27 12:43 CET
N
H
  «Common Platform Enumeration (CPE) is a
  structured naming scheme for information
  technology systems, software, and packages»
PORT      STATE SERVICE VERS
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntul (protocol 2.0)
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at http
s://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.10 seconds
```

Sulla porta 22, esiste un servizio SSH, che utilizza la versione del software OpenSSH 4.7p1 ed il protocollo SSH 2.0



Nmap

Rilevazione del Sistema Operativo

- Nmap può essere usato per rilevare la **versione del Sistema Operativo (OS Fingerprinting Attivo)** sulla macchina target quando si esegue la scansione delle porte
 - Tramite l'opzione –O
- Informazione che sarà molto utile durante il processo di identificazione delle vulnerabilità



Nmap

Rilevazione del Sistema Operativo – Esempio

➤ **nmap -O 10.0.2.6**

```
MAC Address: 08:00:27:AE:29:E1 (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 7.32 seconds
```

Output parziale



Nmap

Nmap Scripting Engine (NSE)

- Nmap può essere esteso mediante l'***Nmap Scripting Engine (NSE)***

- L'*Nmap Scripting Engine (NSE)* permette di
 - Automatizzare varie operazioni
 - Ad esempio, verificare la presenza di vulnerabilità all'interno di applicazioni
 - Implementare ed aggiungere nuove funzionalità ad Nmap

- Nmap **fornisce già numerosi script NSE**
 - Circa 600 script NSE
 - Disponibili nella directory `/usr/share/nmap/scripts`
 - Hanno estensione `.nse`



Nmap

Nmap Scripting Engine (NSE)

- Gli script NSE utilizzano il linguaggio di programmazione **Lua** supportato da Nmap
 - <http://www.lua.org>



Nmap

Nmap Scripting Engine (NSE)

- Gli script *NSE* sono classificati in base a diverse categorie
 - ***auth***: script utilizzati per individuare informazioni di autenticazione sulla macchina target
 - ***exploit***: script che forniscono indicazioni su come sfruttare vulnerabilità sulla macchina target
 - ***malware***: script che controllano l'esistenza di *malware* o *backdoor* sulla macchina target
 - ***vuln***: script che verificano l'esistenza di vulnerabilità sulla macchina target



Nmap

Nmap Scripting Engine (NSE)

- Gli script NSE sono classificati in base a diverse categorie
 - **brute**: script che usano «attacchi a forza bruta» per trovare le credenziali di autenticazione a servizi/protocolli
 - Nmap contiene script per il *brute forcing* di numerosi protocolli
 - **dos**: script che possono effettuare attacchi di tipo *Denial of Service (DoS)*
 - **discovery**: script che permettono di ottenere informazioni di rete interrogando registri pubblici
 - **fuzzer**: script progettati per inviare dati inattesi o casuali ad un'applicazione
 - Tipicamente utilizzati per la rilevazione di vulnerabilità 0-day
 - Etc



Nmap

Nmap Scripting Engine (NSE)

- Vari parametri permettono di specificare gli script NSE da utilizzare
 - **-sC o -script=default**: Vengono utilizzati gli **script di default**
 - **-script <nomefile> | <categoria> | <directory>**: Vengono utilizzati gli script definiti in base a: **nome del file, categoria o directory**
 - **-script-args <args>**: Permette di specificare gli argomenti (parametri) per gli script



Nmap

NSE – Esempio Categoria exploit

➤ `nmap -script exploit 10.0.2.6` (Metasploitable 2)

Output parziale

```
PORT      STATE SERVICE
21/tcp    open  ftp
|_ clamav-exec: ERROR: Script execution failed (use -d to debug)
|_ ftp-vsftpd-backdoor:
|   VULNERABLE:
|     vsFTPd version 2.3.4 backdoor
|       State: VULNERABLE (Exploitable)
|       IDs: CVE:CVE-2011-2523  BID:48539
|         vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|       Disclosure date: 2011-07-03
|       Exploit results:
|         Shell command: id
|           Results: uid=0(root) gid=0(root)
|       References:
|         https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|         https://www.securityfocus.com/bid/48539
|         https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp
|         /vsftpd_234_backdoor.rb
|         http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
```

Vulnerabilità del servizio erogato sulla porta 21



Nmap

NSE – Esempio Categoria exploit

➤ **nmap -script exploit 10.0.2.6 (Metasploitable 2)**

Output parziale

```
80/tcp  open  http
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_http-CSRF:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=10.0.2.10
| Found the following possible CSRF vulnerabilities:

  Path: http://10.0.2.10:80/dvwa/
  Form id:
  Form action: login.php

  Path: http://10.0.2.10:80/mutillidae/index.php?page=view-someones-blog.php
  Form id: id-bad-blog-entry-tr
  Form action: index.php?page=view-someones-blog.php

  Path: http://10.0.2.10:80/mutillidae/index.php?page=source-viewer.php
  Form id: id-bad-cred-tr
  Form action: index.php?page=source-viewer.php

  Path: http://10.0.2.10:80/mutillidae/?page=text-file-viewer.php
  Form id: id-bad-cred-tr
  Form action: index.php?page=text-file-viewer.php

  Path: http://10.0.2.10:80/mutillidae/index.php?page=user-info.php
  Form id: id-bad-cred-tr
  Form action: ./index.php?page=user-info.php
```

Vulnerabilità del servizio erogato sulla porta 80



Nmap

NSE – Esempio 1 Categoria brute

- **nmap -script brute 10.0.2.6 (Metasploitable 2)**

```
root@kali:/usr/share/nmap/scripts# nmap --script brute 10.0.2.6
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-31 08:13 EDT
Nmap scan report for 10.0.2.6
Host is up (0.00074s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|  ftp-brute:
|  Accounts:
|    user:user - Valid credentials
|_ Statistics: Performed 3392 guesses in 601 seconds, average tps: 5.1
22/tcp    open  ssh
|  ssh-brute:
|  Accounts:
|    user:user - Valid credentials
|_ Statistics: Performed 181 guesses in 601 seconds, average tps: 0.9
```

Output parziale

```
512/tcp  open  exec
|  rexec-brute:
|  Accounts:
|    root:root - Valid credentials
|    netadmin:netadmin - Valid credentials
|    guest:guest - Valid credentials
|    user:user - Valid credentials
|    web:web - Valid credentials
|    sysadmin:sysadmin - Valid credentials
|    administrator:administrator - Valid credentials
|    webadmin:webadmin - Valid credentials
|    admin:admin - Valid credentials
|    test:test - Valid credentials
|_ Statistics: Performed 14 guesses in 52 seconds, average tps: 0.3
```



Nmap

NSE – Esempio 1 Categoria brute

- **nmap -script brute 10.0.2.6 (Metasploitable 2)**

```
root@kali:/usr/share/nmap/scripts# nmap --script brute 10.0.2.6
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-31 08:13 EDT
Nmap scan report for 10.0.2.6
Host is up (0.00074s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|_ ftp-brute:
|   Accounts:
|     user:user - Valid credentials
|_ Statistics: Performed 3392 guesses in 601 seconds, average tps: 5.1
22/tcp    open  ssh
|_ ssh-brute:
|   Accounts:
|     user:user - Valid credentials
|_ Statistics: Performed 181 guesses in 601 seconds, average tps: 0.9
```

- La macchina target permette la connessione al servizio **FTP** utilizzando le seguenti credenziali
 - Username: **user**
 - Password: **user**

```
user:user - Valid credentials
tadmin - Valid credentials
c - Valid credentials
user:user - Valid credentials
web:web - Valid credentials
sysadmin:sysadmin - Valid credentials
administrator:administrator - Valid credentials
webadmin:webadmin - Valid credentials
admin:admin - Valid credentials
test:test - Valid credentials
Statistics: Performed 14 guesses in 52 seconds, average tps: 0.3
```

Nmap

NSE – Esempio 1 Categoria brute

- Mediante il seguente comando è possibile connettersi al servizio FTP della macchina target, utilizzando *Username*: **user** e *Password*: **user**

- **ftp 10.0.2.6**

```
root@kali:~# ftp 10.0.2.6
Connected to 10.0.2.6.
220 (vsFTPd 2.3.4)
Name (10.0.2.6:root): user
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> █
```



Nmap

NSE – Esempio 2 Categoria brute

➤ **nmap -script brute 10.0.2.6**

```
root@kali:/usr/share/nmap/scripts# nmap --script brute 10.0.2.6
Starting Nmap 7.80 ( https://nmap.org ) at 2019-10-31 08:13 EDT
Nmap scan report for 10.0.2.6
Host is up (0.00074s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|_ ftp-brute:
|   Accounts:
|     user:user - Valid credentials
|_ Statistics: Performed 3392 guesses in 601 seconds, average tps: 5.1
22/tcp    open  ssh
|_ ssh-brute:
|   Accounts:
|     user:user - Valid credentials
|_ Statistics: Performed 181 guesses in 601 seconds, average tps: 0.9
```

Output parziale

- La macchina target permette la connessione al servizio SSH utilizzando le seguenti credenziali
- Username: **user**
 - Password: **user**

```
512/tcp  open  -
|_ ssh-brute:
|   Accounts:
|     user:user - Valid credentials
|     rootadmin - Valid credentials
|     root - Valid credentials
|     admin - Valid credentials
|     Valid credentials
|     sysadmin - Valid credentials
|     administrator:administrator - Valid credentials
|     wwwadmin:webadmin - Valid credentials
|     admin:admin - Valid credentials
|     test:test - Valid credentials
|_ Statistics: Performed 14 guesses in 52 seconds, average tps: 0.3
```

Nmap

NSE – Esempio 2 Categoria brute

- Mediante il seguente comando è possibile connettersi al servizio *SSH* della macchina target, utilizzando *Username*: **user** e *Password*: **user**

➤ **ssh user@10.0.2.6**

```
root@kali:/usr/share/nmap/scripts# ssh user@10.0.2.6
The authenticity of host '10.0.2.6 (10.0.2.6)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCi0LuVscegPXLQ0suPs+E9d/rrJB84rk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.6' (RSA) to the list of known hosts.
user@10.0.2.6's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Thu Oct 31 09:15:34 2019 from 10.0.2.5
```



Nmap

NSE – Esempio 3 Categoria brute

➤ **nmap -script brute 10.0.2.6 (Metasploitable 3)**

```
|_ Statistics: Performed 6020 guesses in 600 seconds, average tps: 10.0
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3000/tcp open  ppp
3306/tcp open  mysql
| mysql-brute:
|   Accounts:
|     root:<empty> - Valid credentials
|_ Statistics: Performed 45011 guesses in 226 seconds, average tps: 212.3
mysql-enum:
|   Valid usernames:
|     root:<empty> - Valid credentials
|_ Statistics: Performed 10 guesses in 6 seconds, average tps: 1.7
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
7676/tcp  open  imqbrokerd
8009/tcp  open  ajp13
| ajp-brute:
|   URL does not require authentication
```

Output parziale



Nmap

NSE – Esempio 3 Categoria brute

➤ **nmap -script brute 10.0.2.6 (Metasploitable 3)**

```
|_ Statistics: Performed 6020 guesses in 600 seconds, average tps: 10.0
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3000/tcp open  ppp
3306/tcp open  mysql
| mysql-brute:
|   Accounts:
|     root:<empty> - Valid credentials
|_ Statistics: Performed 45011 guesses in 226 seconds, average tps: 212.3
mysql-enum:
  Valid usernames:
    root:<empty> - Valid credentials
|_ Statistics: Performed 10 guesses in 6 seconds, average tps: 1.7
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-h
7676/tcp  open  ?
```

Output parziale

- La macchina target permette la connessione al servizio **MYSQL** utilizzando le seguenti credenziali
 - Username: **root**
 - Password:



Nmap

NSE – Esempio Categoria vuln

➤ **nmap -script vuln 10.0.2.6**

Output parziale

```
root@kali:/usr/share/nmap/scripts# nmap --script vuln 10.0.2.6
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 14:44 CET
Nmap scan report for 10.0.2.6
Host is up (0.00012s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|_  ftp-vsftpd-backdoor:
|   VULNERABLE:
|     vsFTPD version 2.3.4 backdoor
|       State: VULNERABLE (Exploitable)
|       IDs:  CVE:CVE-2011-2523 OSVDB:73573
|         vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
|         Disclosure date: 2011-07-03
|         Exploit results:
|           Shell command: id
|             Results: uid=0(root) gid=0(root)
|             References:
|               http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-
backdoored.html
|               https://github.com/rapid7/metasploit-framework/blob/master/modules/ex-
ploits/unix/ftp/vsftpd_234_backdoor.rb
|               http://osvdb.org/73573
|               https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
```

Nmap

NSE – Esempio Categoria vuln

➤ `nmap -script vuln 10.0.2.6`

Output parziale

```
root@kali:/usr/share/nmap/scripts# nmap --script vuln 10.0.2.6
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 14:44 CET
Nmap scan report for 10.0.2.6
Host is up (0.00012s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
|_ftp-vsftpd-backdoor:
|  VULNERABLE:
|    vsFTPD version 2.3.4 backdoor
|      State: VULNERABLE (Exploitable)
|      IDs: CVE:CVE-2011-2523 OSVDB:73573
|        vsFTPD version 2.3.4 backdoor, this was reported on 2011-07-04.
|        Disclosure date: 2011-07-03
|        Exploit results:
|          Shell command: id
|          Results: user: root
|          References:
|            http://scambackdoored.html
|            https://git/ploits/unix/ftp/vs
|              http://osvdb.org/73573
|              https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
```

“Common Vulnerabilities and Exposures (CVE)
is a dictionary-type list of standardized names
for vulnerabilities and other information related
to security exposures”

Nmap

NSE – Esempio Categoria vuln

➤ <https://cve.mitre.org/>

The screenshot shows the homepage of the CVE.mitre.org website. At the top, there's a navigation bar with links for "CVE List", "CNAs", "Board", "About", "News & Blog", and the "NVD" logo which links to "CVSS Scores", "CPE Info", and "Advanced Search". Below the navigation is a search bar and a menu bar with links for "Search CVE List", "Download CVE", "Data Feeds", "Request CVE IDs", and "Update a CVE Entry". A banner at the bottom of the menu bar displays "TOTAL CVE Entries: 115277".

CVE® is a [list](#) of entries—each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities.

CVE Entries are used in numerous cybersecurity [products and services](#) from around the world, including the U.S. National Vulnerability Database ([NVD](#)).

CNA Participation Growing Worldwide

A world map with various colored dots representing CNA participation. A large orange dot is located in North America, and smaller dots are scattered across Europe, Asia, and Australia.

CVE Numbering Authorities (CNAs)

Latest CVE News

- ◆ [CVE Board Adds a "CNA Coordination Working Group Liaison" Board Member](#)
- ◆ [Minutes from CVE Board Teleconference Meetings on March 20 and April 3 Now Available](#)

[More >>](#)

CVE Blog

Refresher: When to Use the CVE Request Web Form

First introduced in August 2016, the online "[CVE Request Web Form](#)" is the main method for communicating with the [CVE Program Root CNA](#). By using the web form, the CVE Program's ability to receive, manage, track, and respond to user questions and [CVE ID](#) requests has significantly improved.

In this article, inspired by user questions, we will discuss when and how to use the CVE Request Web Form to best assist you.

[More >>](#)

Newest CVE Entries

Tweets by @CVEnew

CVE
@CVEnew
CVE-2019-1841 A vulnerability in the Software Image Management feature of Cisco DNA Center could allow an authenticated, remote attacker to access to internal services without additional authentication. The vulnerability is due to insufficient validation... [cve.mitre.org/cgi-bin/cvenam...](#)

9h

CVE
@CVEnew
CVE-2019-1840 A vulnerability in the DHCPv6 input packet processor of Cisco Prime Network Registrar could allow an unauthenticated, remote

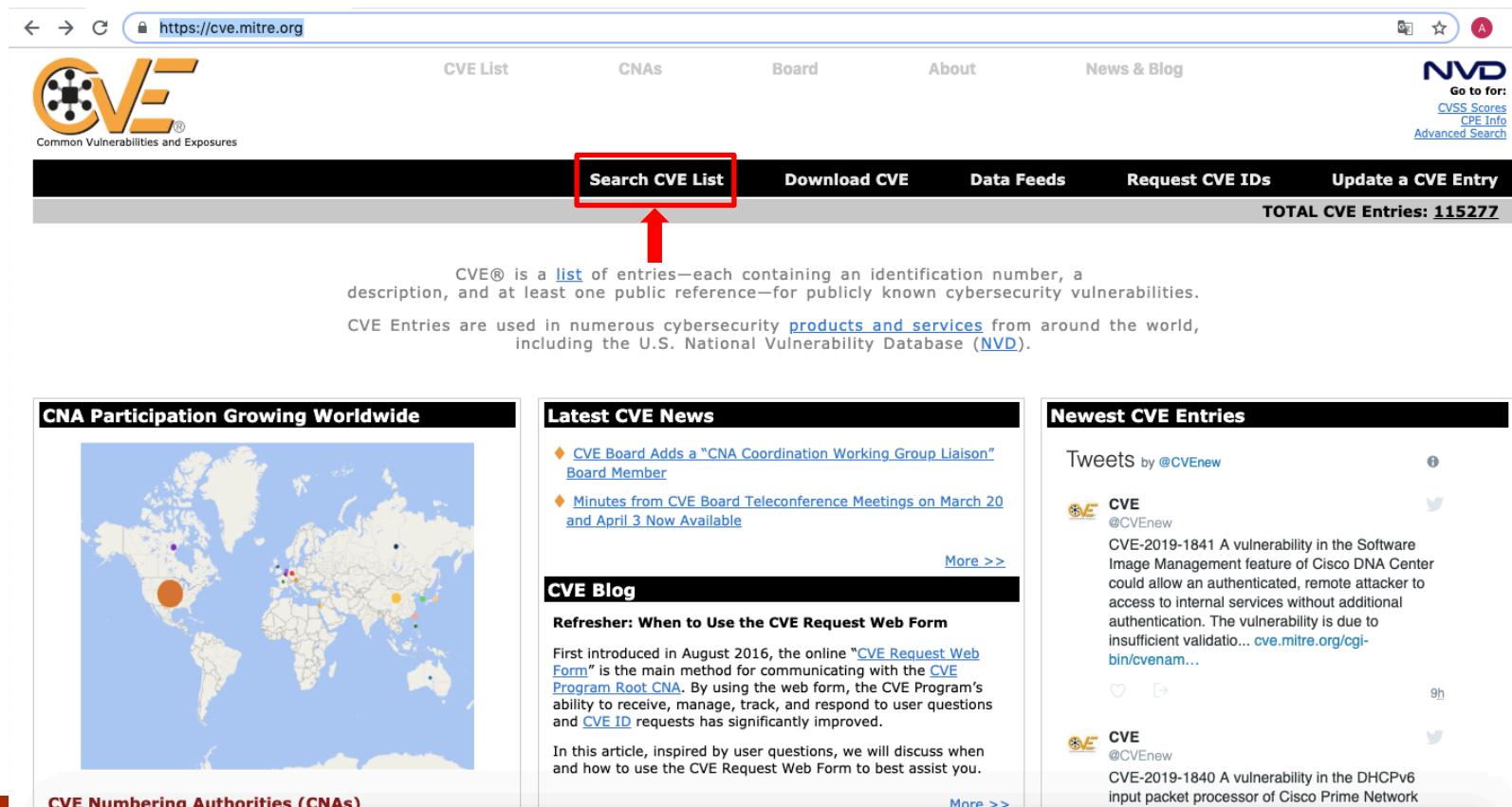
9h

Enumerating Target e Port Scanning

Nmap

NSE – Esempio Categoria vuln

➤ <https://cve.mitre.org/>



The screenshot shows the CVE.mitre.org website. At the top, there is a navigation bar with links for "CVE List", "CNAs", "Board", "About", "News & Blog", and the "NVD" logo which includes links for "CVSS Scores", "CPE Info", and "Advanced Search". Below the navigation bar is a dark header with several buttons: "Search CVE List" (highlighted with a red box and a red arrow pointing to it), "Download CVE", "Data Feeds", "Request CVE IDs", and "Update a CVE Entry". A total count of "TOTAL CVE Entries: 115277" is displayed. The main content area contains a brief introduction to CVE, a map showing CNA participation worldwide, and sections for "Latest CVE News" and "Newest CVE Entries".

CVE® is a [list](#) of entries—each containing an identification number, a description, and at least one public reference—for publicly known cybersecurity vulnerabilities.

CVE Entries are used in numerous cybersecurity [products and services](#) from around the world, including the U.S. National Vulnerability Database ([NVD](#)).

CNA Participation Growing Worldwide



LATEST CVE NEWS

- ◆ [CVE Board Adds a "CNA Coordination Working Group Liaison" Board Member](#)
- ◆ [Minutes from CVE Board Teleconference Meetings on March 20 and April 3 Now Available](#)

[More >>](#)

CVE Blog

Refresher: When to Use the CVE Request Web Form

First introduced in August 2016, the online "[CVE Request Web Form](#)" is the main method for communicating with the [CVE Program Root CNA](#). By using the web form, the CVE Program's ability to receive, manage, track, and respond to user questions and [CVE ID](#) requests has significantly improved.

In this article, inspired by user questions, we will discuss when and how to use the CVE Request Web Form to best assist you.

[More >>](#)

NEWEST CVE ENTRIES

Tweets by @CVEnew

CVE
@CVEnew
CVE-2019-1841 A vulnerability in the Software Image Management feature of Cisco DNA Center could allow an authenticated, remote attacker to access to internal services without additional authentication. The vulnerability is due to insufficient validation... [cve.mitre.org/cgi-bin/cvenam...](#)

9h

CVE
@CVEnew
CVE-2019-1840 A vulnerability in the DHCPv6 input packet processor of Cisco Prime Network Registrar could allow an unauthenticated, remote

9h

Enumerating Target e Port Scanning

Nmap

NSE – Esempio Categoria vuln

➤ <https://cve.mitre.org/>

The screenshot shows the CVE List search interface. At the top, there is a navigation bar with links for 'CVE List', 'CNAs', 'Board', 'About', and 'News & Blog'. On the right, there is a link to 'NVD' which includes 'Go to for:' and links to 'CVSS Scores', 'CPE Info', and 'Advanced Search'. Below the navigation bar is a search bar with the placeholder 'Search CVE List'. To the right of the search bar are buttons for 'Download CVE', 'Data Feeds', 'Request CVE IDs', and 'Update a CVE Entry'. A total count of 'TOTAL CVE Entries: 115277' is displayed. Below the search bar, the URL 'HOME > CVE LIST > SEARCH CVE LIST' is shown. The main content area is titled 'Search CVE List'. It contains instructions: 'You can search the CVE List for a [CVE Entry](#) if the [CVE ID](#) is known. To search by keyword, use a specific term or multiple keywords separated by a space. Your results will be the relevant CVE Entries.' It also provides a link to 'View the [search tips](#)'. A red box highlights the search term '2011-2523' entered into the search bar.



Nmap

NSE – Esempio Categoria vuln

➤ <https://cve.mitre.org/>



CVE List

CNAs

Board

About

News & Blog

NVD

Go to for:

CVSS Scores

CPE Info

[Advanced Search](#)

Search CVE List

Download CVE

Data Feeds

Request CVE IDs

Update a CVE Entry

TOTAL CVE Entries: 115277

HOME > CVE > SEARCH RESULTS

Search Results

There are 1 CVE entries that match your search.



Name	Description
CVE-2011-2523	** RESERVED ** This candidate has been reserved by an organization or individual that will use it when announcing a new security problem. When the candidate has been publicized, the details for this candidate will be provided.

[BACK TO TOP](#)

SEARCH CVE USING KEYWORDS:
You can also search by reference using the [CVE Reference Maps](#).
For More Information: cve@mitre.org



Nmap

NSE – Esempio Categoria vuln

➤ <https://cve.mitre.org/>



CVE List

CNAs

Board

About

News & Blog

NVD

Go to for:

CVSS Scores

CPE Info

[Advanced Search](#)

Search CVE List

Download CVE

Data Feeds

Request CVE IDs

Update a CVE Entry

TOTAL CVE Entries: 115277

HOME > CVE > SEARCH RESULTS

Search Results

There are 1 CVE entries that match your search.



Name	Description
CVE-2011-2523	** RESERVED ** This candidate has been reserved by an organization or individual that will use it when announcing a new security problem. When the candidate has been publicized, the details for this candidate will be provided.

[BACK TO TOP](#)

SEARCH CVE USING KEYWORDS:
You can also search by reference using the [CVE Reference Maps](#).
For More Information: cve@mitre.org

Maggiori dettagli sul CVE verranno mostrati nelle lezioni successive...



Nmap

Aggressive Scan

- Mediante l'opzione **-A** Nmap effettuerà contemporaneamente
 - *Service Version Detection (-sV)*
 - *Operating System Detection (-O)*
 - *Script Scanning (-sC)* [Maggiori dettagli in seguito]
 - *Traceroute (-traceroute)*



Nmap

Aggressive Scan – Esempio

➤ `nmap -A 10.0.2.6`

```
root@kali:~# nmap -A 10.0.2.6
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 12:56 CET
Nmap scan report for 10.0.2.6
Host is up (0.00045s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
|     FTP server status:
|       Connected to 10.0.2.15
|       Logged in as ftp
|       TYPE: ASCII
|       No session bandwidth limit
|       Session timeout in seconds is 300
|       Control connection is plain text
|       Data connections will be plain text
|       vsFTPD 2.3.4 - secure, fast, stable
| End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntul (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|   2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
```

Output parziale



Nmap

Scansione di Indirizzi IPv6

- Nmap permette di scansionare indirizzi IPv6 (Opzione **-6**)
 - N. B. Può essere specificato un solo indirizzo IPv6 alla volta
- Esempio: **nmap -6 fe80::a00:27ff:feae:29e1**

```
root@kali:~# nmap -6 fe80::a00:27ff:feae:29e1
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-27 14:04 CET
Nmap scan report for fe80::a00:27ff:feae:29e1
Host is up (0.00012s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
53/tcp    open  domain
2121/tcp  open  ccproxy-ftp
5432/tcp  open  postgresql
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 5.98 seconds
Enumerating target's port scanning
```



Nmap

Scansione di Indirizzi IPv6

- Nmap permette di scansionare indirizzi IPv6 (Opzione **-6**)
 - N. B. Può essere specificato un solo indirizzo IPv6 alla volta
- Esempio: nmap -6 **fe80::a00:27ff:feae:29e1**

```
root@kali:~# nmap -6 fe80::a00:27ff:feae:29e1
Starting Nmap 7.70 ( https://nmap.org ) at 2023-11-27 14:04 CET
Nmap scan report for fe80::a00:27ff:feae:29e1
Host is up (0.00012s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
53/tcp    open  domain
2121/tcp  open  ccproxy-ftp
5432/tcp  open  postgresql
MAC Address: 08:00:27:AE:29:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 5.98 seconds
```

Enumerating targets & port scanning

Indirizzo IPv6 di Metasploitable 2



Nmap

Gestione dell'Output

- Il risultato (output) di una scansione Nmap può essere memorizzato in un file esterno

- Questa opzione è utile quando è necessario elaborare il risultato di Nmap mediante altri strumenti

- N.B. Anche se il risultato di una scansione viene memorizzato in un file esterno, Nmap continuerà anche a mostrare tale risultato sullo *Standard Output*



Nmap

Gestione dell'Output

- Nmap supporta diversi formati di output, tra i quali
 - **Interactive output:** formato di output predefinito. Il risultato della scansione viene inviato allo *Standard Output*
 - **Normal output (-oN):** simile all'output interattivo, ma non include informazioni sull'esecuzione ed i *warning*
 - **XML output (-oX):** Genera l'output in formato *XML*
 - Questo formato può essere convertito in formato *HTML*, analizzato tramite altri strumenti o importato in un database
 - **Grepable output (-oG):** formato deprecato. Permette all'output di Nmap di essere meglio usato con strumenti UNIX quali **grep**, **awk**, etc



Nmap

Gestione dell'Output – Esempio

- Salviamo l'output della scansione nmap nel file **myscan.xml**
 - **nmap 10.0.2.6 -oX myscan.xml**
 - Apriamo il file **myscan.xml** mediante **gedit**

Output parziale

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE nmaprun>
<?xml-stylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?>
<!-- Nmap 7.70 scan initiated Wed Mar 27 11:46:40 2019 as: nmap -oX myscan.xml 10.0.2.6 -->
<nmaprun scanner="nmap" args="nmap -oX myscan.xml 10.0.2.6" start="1553683600" startstr="Wed Mar 27 11:46:40 2019"
version="7.70" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000"
services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143
>
<verbose level="0"/>
<debugging level="0"/>
<host starttime="1553683600" endtime="1553683605"><status state="up" reason="arp-response" reason_ttl="0"/>
<address addr="10.0.2.6" addrtype="ipv4"/>
<address addr="08:00:27:AE:29:E1" addrtype="mac" vendor="Oracle VirtualBox virtual NIC"/>
```



Nmap

Gestione dell'Output – Esempio

- Salviamo l'output della scansione nmap nel file **myscan.xml**
 - `nmap 10.0.2.6 -oX myscan.xml`
 - Apriamo il file **myscan.xml** mediante **gedit**

Output parziale

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE nmaprun>
<?xml-stylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?>
<!-- Nmap 7.70 scan initiated Wed Mar 27 11:46:40 2019 as: nmap -oX myscan.xml 10.0.2.6 -->
<nmaprun scanner="nmap" args="nmap -oX myscan.xml 10.0.2.6" start="1553683600" startstr="Wed Mar 27 11:46:40 2019"
version="7.70" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000"
services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143
>
<verbose level="0"/>
<debugging level="0"/>
<host starttime="1553683600" endtime="1553683605"><status state="up" reason="arp-response" reason_ttl="0"/>
<address addr="10.0.2.6" addrtype="ipv4"/>
<address addr="08:00:27:AE:29:E1" addrtype="mac" vendor="Oracle VirtualBox virtual NIC"/>
```



Nmap

Gestione dell'Output – Esempio

- Salviamo l'output della scansione nmap nel file **myscan.xml**
 - `nmap 10.0.2.6 -oX myscan.xml`
 - Apriamo il file **myscan.xml** mediante **gedit**

Output parziale

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE nmaprun>
<?xml-stylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?>
<!-- Nmap 7.70 scan initiated Wed Mar 27 11:46:40 2019 as: nmap -oX my
<nmaprun scanner="nmap" args="nmap -oX myscan.xml 10.0.2.6" start="1553
version="7.70" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000" services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143>
<verbose level="0"/>
<debugging level="0"/>
<host starttime="1553683600" endtime="1553683605"><status state="up" reason="arp-response" reason_ttl="0"/>
<address addr="10.0.2.6" addrtype="ipv4"/>
<address addr="08:00:27:AE:29:E1" addrtype="mac" vendor="Oracle VirtualBox virtual NIC"/>
```

Tipo di scansione e numero
di porte analizzate



Nmap

Gestione dell'Output – Esempio

- Salviamo l'output della scansione nmap nel file **myscan.xml**
 - `nmap 10.0.2.6 -oX myscan.xml`
 - Apriamo il file **myscan.xml** mediante **gedit**

Output parziale

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE nmaprun>
<?xml-stylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?
<!-- Nmap 7.70 scan initiated Wed Mar 27 11:46:40 2019 as: nmap -oX myscan.xml 10.0.2.6 -->
<nmaprun scanner="nmap" args="nmap -oX myscan.xml 10.0.2.6" start="1553683600" end="1553683605" version="7.70" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000" services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143">
</scaninfo>
<verbose level="0"/>
<debugging level="0"/>
<host starttime="1553683600" endtime="1553683605"><status state="up" reason="arp-response" reason_ttl="0"/>
<address addr="10.0.2.6" addrtype="ipv4"/>
<address addr="08:00:27:AE:29:E1" addrtype="mac" vendor="Oracle VirtualBox virtual NIC"/>
```

Porte analizzate (Output Parziale)



Nmap

Gestione dell'Output – Esempio

- Salviamo l'output della scansione nmap nel file **myscan.xml**
 - `nmap 10.0.2.6 -oX myscan.xml`
 - Apriamo il file **myscan.xml** mediante **gedit**

Output parziale

```
<ports><extraports state="closed" count="977">
<extrareasons reason="resets" count="977"/>
</extraports>
<port protocol="tcp" portid="21"><state state="open" reason="syn-ack" reason_ttl="64"/><service name="ftp" method="table" conf="3"/></port>
<port protocol="tcp" portid="22"><state state="open" reason="syn-ack" reason_ttl="64"/><service name="ssh" method="table" conf="3"/></port>
<port protocol="tcp" portid="23"><state state="open" reason="syn-ack" reason_ttl="64"/><service name="telnet" method="table" conf="3"/></port>
<port protocol="tcp" portid="25"><state state="open" reason="syn-ack" reason_ttl="64"/><service name="smtp" method="table" conf="3"/></port>
<port protocol="tcp" portid="53"><state state="open" reason="syn-ack" reason_ttl="64"/><service name="domain" method="table" conf="3"/></port>
```

Osservazione: 1000 porte esaminate

- 23 Aperte
- 977 Chiuse

Porte Aperte



Nmap

Gestione dell'Output – Esempio

- L'output in formato *XML* è scomodo da esaminare per un essere umano
1. Possiamo convertire il file *XML* in un file *HTML*
 - `xsltproc myscan.xml -o myscan.html`
 2. Ed aprire il file `myscan.html` con un Web Browser



Nmap

Gestione dell'Output – Esempio

Nmap Scan Report - Scanned at Wed Mar 27 11:46:40 2019

[Scan Summary](#) | [10.0.2.6](#)

Scan Summary

Nmap 7.70 was initiated at Wed Mar 27 11:46:40 2019 with these arguments:

```
nmap -oX myscan.xml 10.0.2.6
```

Verbosity: 0; Debug level 0

Nmap done at Wed Mar 27 11:46:45 2019; 1 IP address (1 host up) scanned in 5.89 seconds

10.0.2.6

Address

- 10.0.2.6 (ipv4)
- 08:00:27:AE:29:E1 - Oracle VirtualBox virtual NIC (mac)

Ports

The 977 ports scanned but not shown below are in state: **Closed**

Enumerating Target e Port Scanning



Nmap

Gestione dell'Output – Esempio

Ports

The 977 ports scanned but not shown below are in state: **closed**

- 977 ports replied with: **resets**

Port		State (toggle closed [o] filtered [f])	Service	Reason	Product	Version	Extra info
21	tcp	open	ftp	syn-ack			
22	tcp	open	ssh	syn-ack			
23	tcp	open	telnet	syn-ack			
25	tcp	open	smtp	syn-ack			
53	tcp	open	domain	syn-ack			
80	tcp	open	http	syn-ack			
111	tcp	open	rpcbind	syn-ack			
139	tcp	open	netbios-ssn	syn-ack			
445	tcp	open	microsoft-ds	syn-ack			
512	tcp	open	exec	syn-ack			
513	tcp	open	login	syn-ack			
514	tcp	open	shell	syn-ack			
1099	tcp	open	rmiregistry	syn-ack			
1524	tcp	open	ingreslock	syn-ack			
2049	tcp	open	nfs	syn-ack			
2121	tcp	open	coproxy-ftp	syn-ack			
3306	tcp	open	mysql	syn-ack			
5432	tcp	open	postgresql	syn-ack			
5900	tcp	open	vnc	syn-ack			
6000	tcp	open	X11	syn-ack			
6667	tcp	open	irc	syn-ack			
8009	tcp	open	ajp13	syn-ack			
8180	tcp	open	unknown	syn-ack			

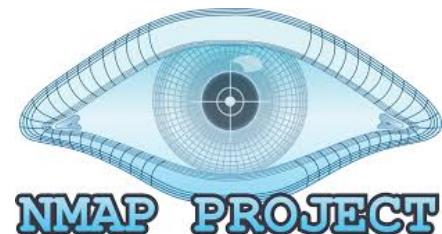


Outline

- Concetti Introduttivi
- Suite Protocollare TCP/IP
- Formato dei Messaggi TCP e UDP
- Active Enumeration
 - Network Scanner Nmap
 - Zenmap
 - Unicornscan
- Passive Enumeration
 - Shodan
 - ZoomEye
 - FOFA
 - Censys

Zenmap

- Utile interfaccia grafica (*GUI*) per Nmap
- Fornisce numerosi vantaggi rispetto all'utilizzo testuale di Nmap
- Non presente di default in Kali
 - `apt-get install zenmap-kde`

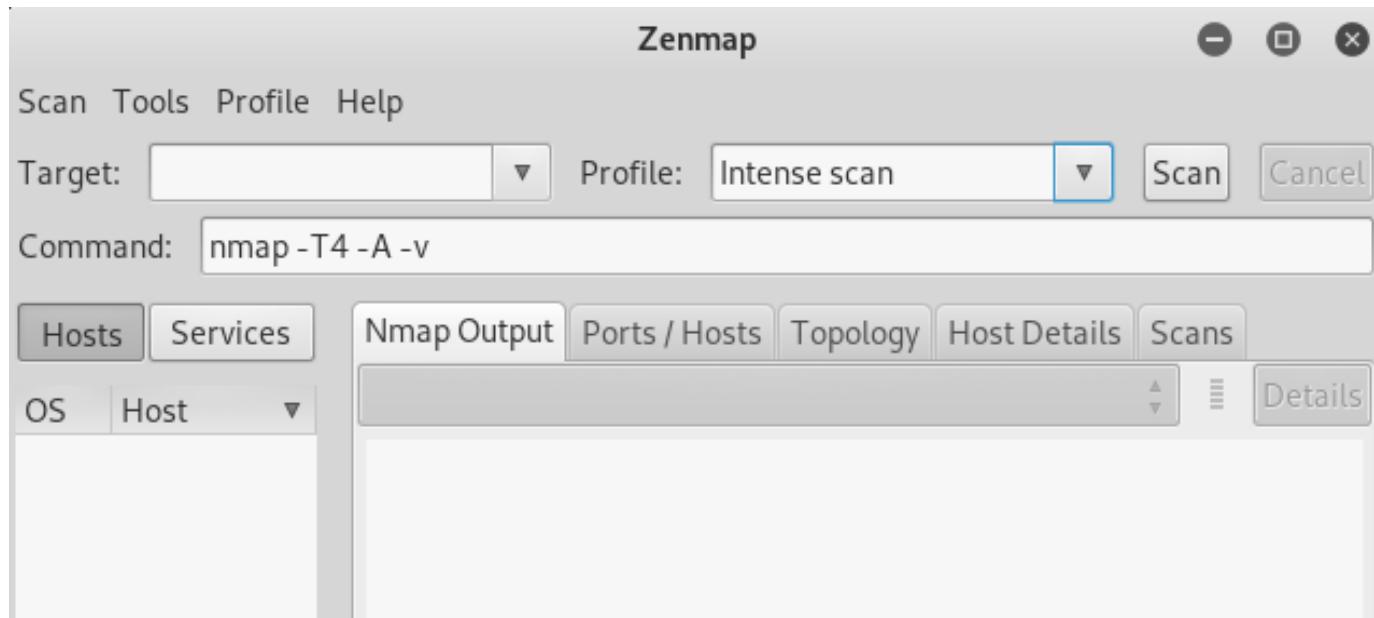


Zenmap

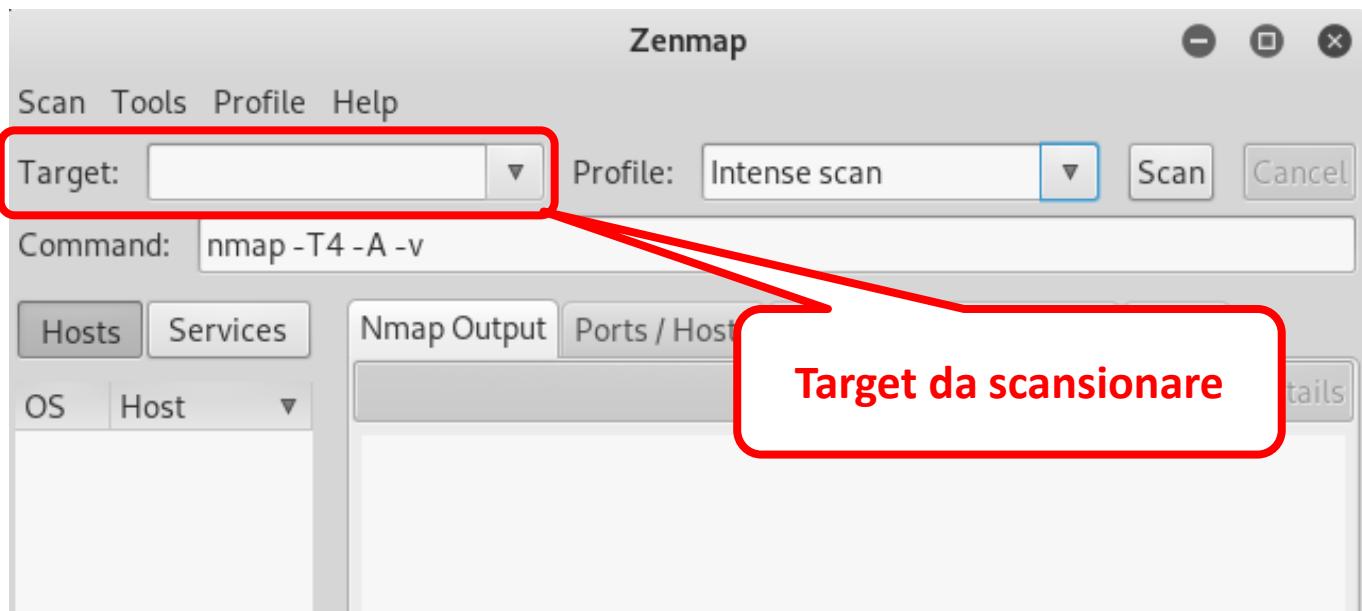
- Zenmap
 - Strumento interattivo che permette di analizzare comodamente i risultati di una scansione
 - Ad esempio, mostrando anche una *mappa topologica* della rete analizzata
 - Permette di effettuare confronti tra due scansioni
 - Tiene traccia dei risultati della scansione
 - Permette di creare «**profili di scansione**»
 - Per eseguire più volte la stessa scansione il pentester può utilizzare un determinato **profilo** Zenmap
 - Mostra sempre il comando che viene eseguito, così che il pentester possa verificare manualmente tale comando

Zenmap

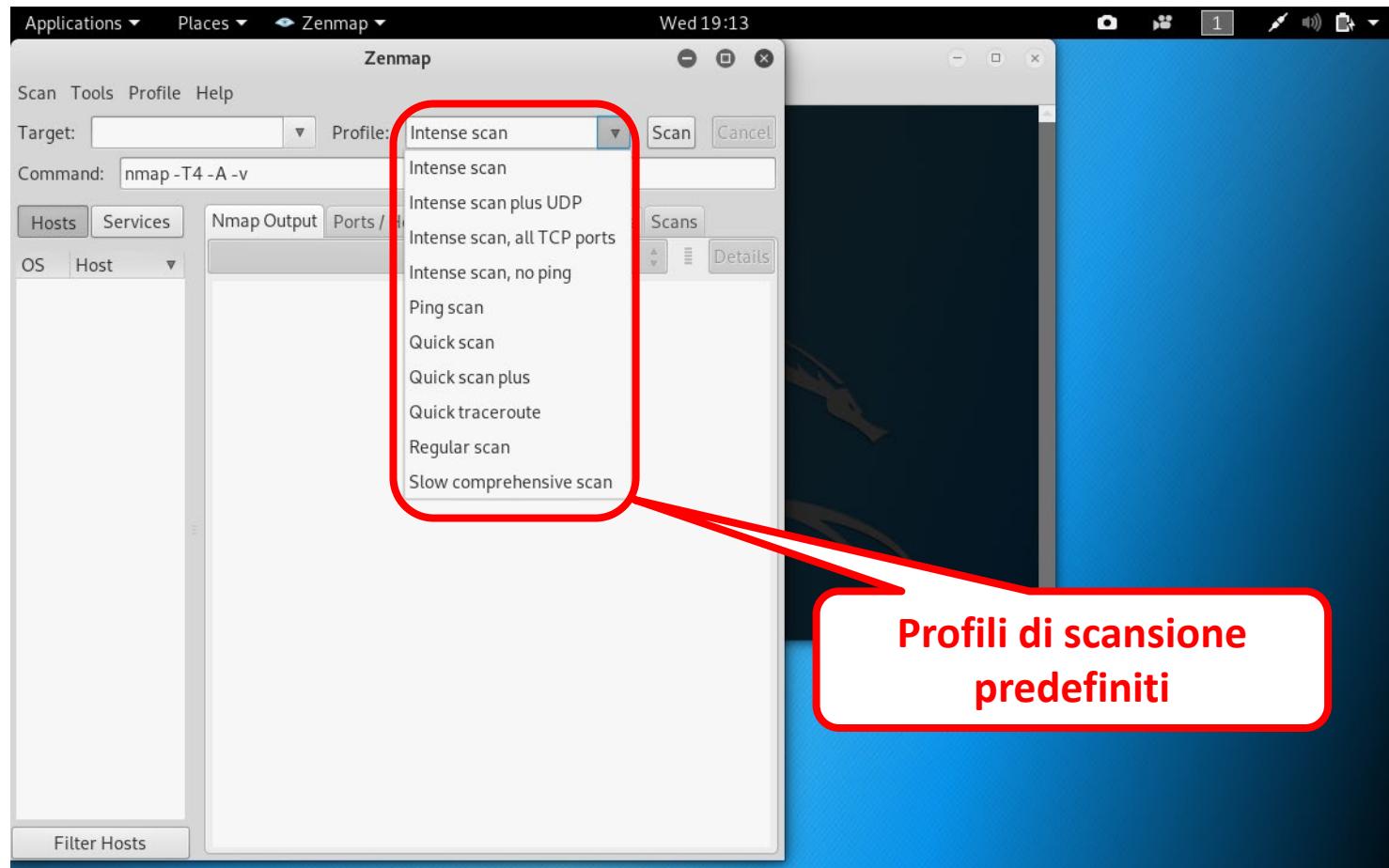
- Zenmap può essere avviato da Terminale digitando il comando **zenmap-kbx**



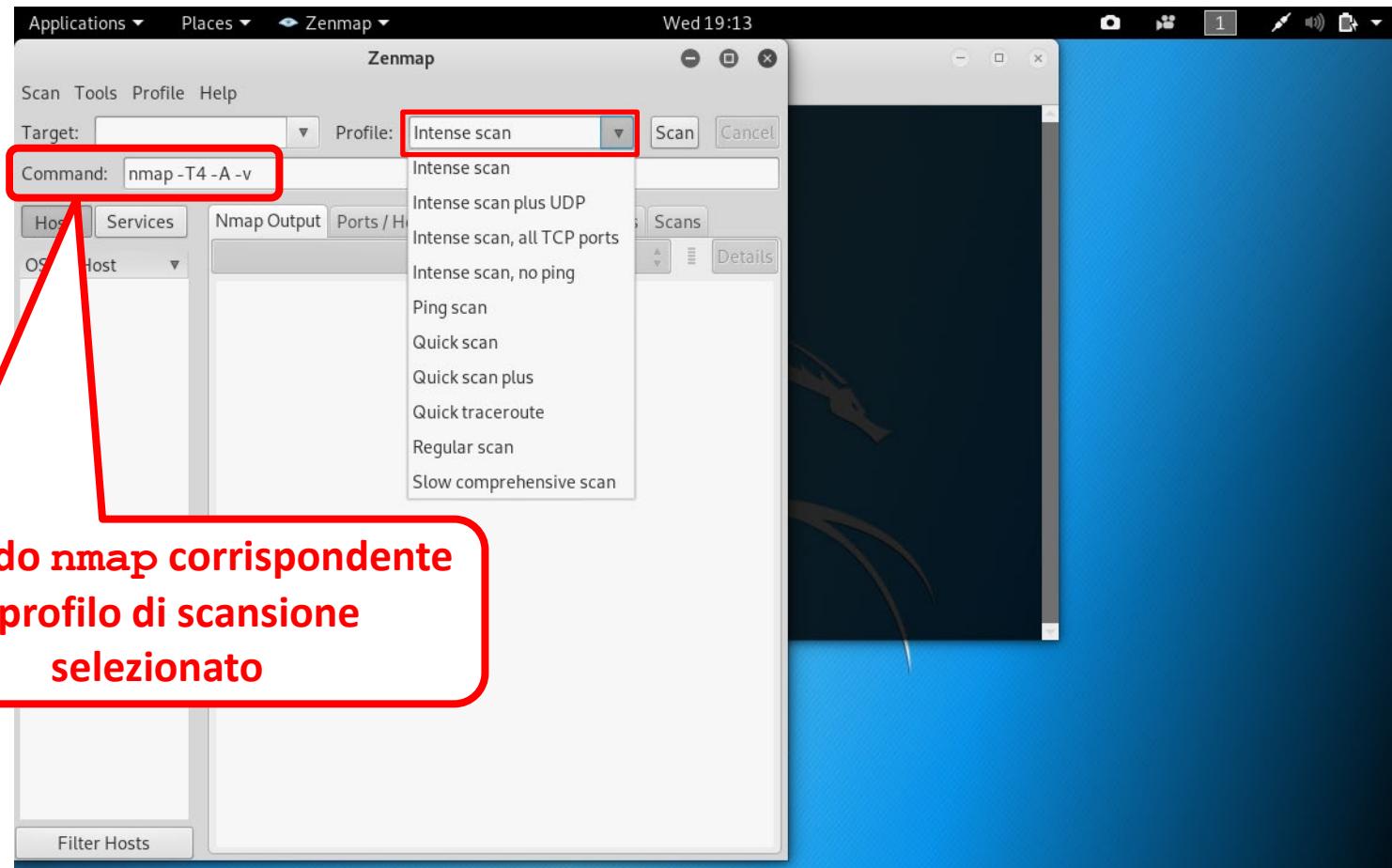
Zenmap



Zenmap

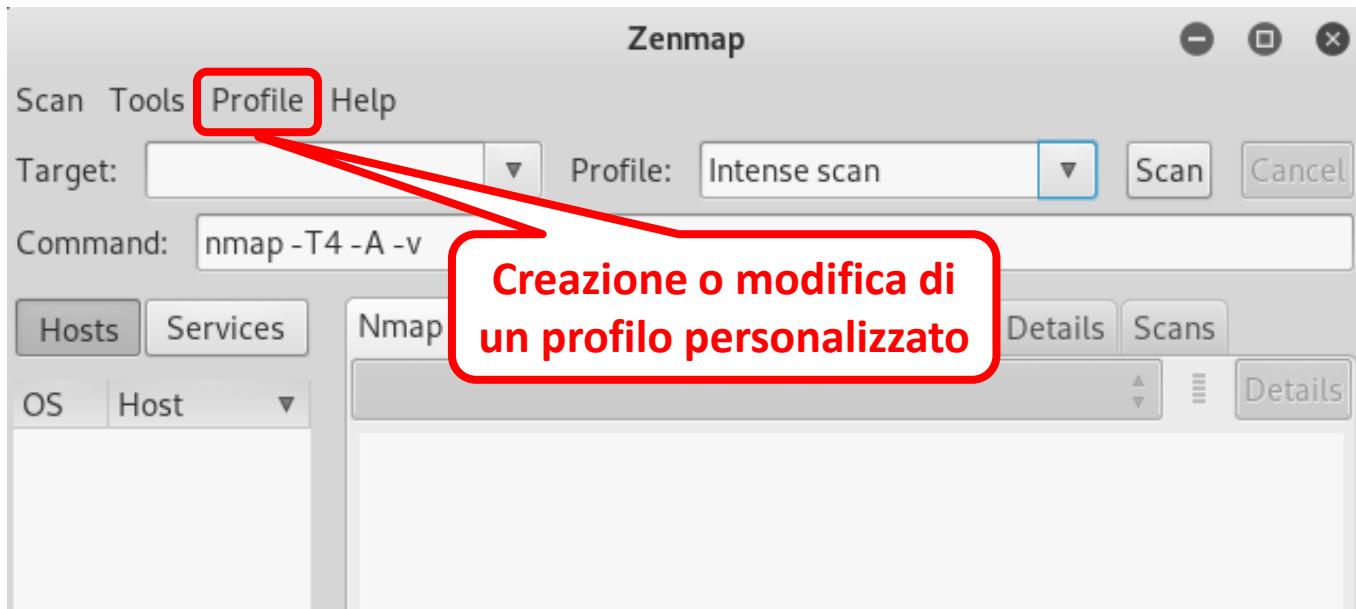


Zenmap



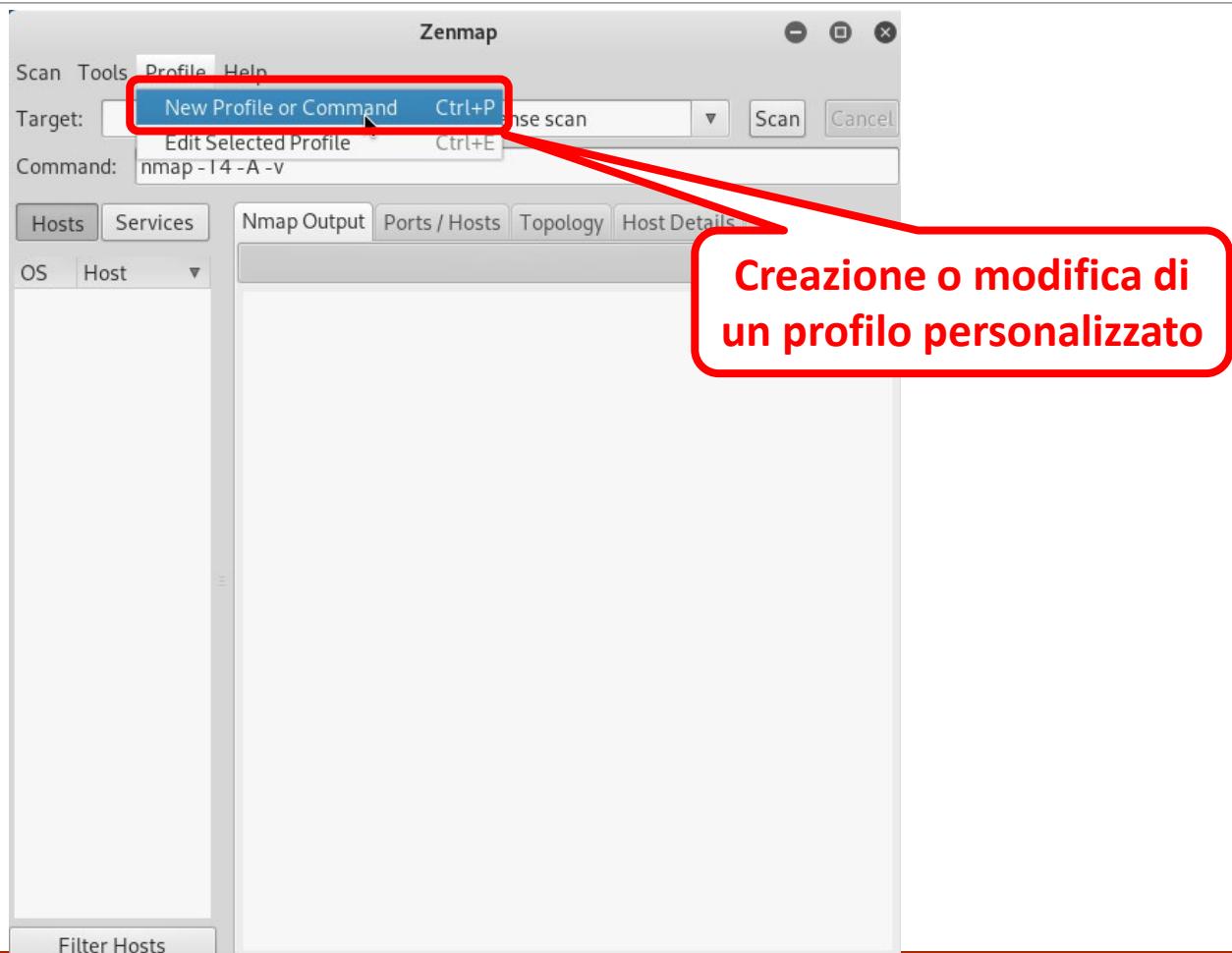
Zenmap

Creazione Profili di Scansione Personalizzati



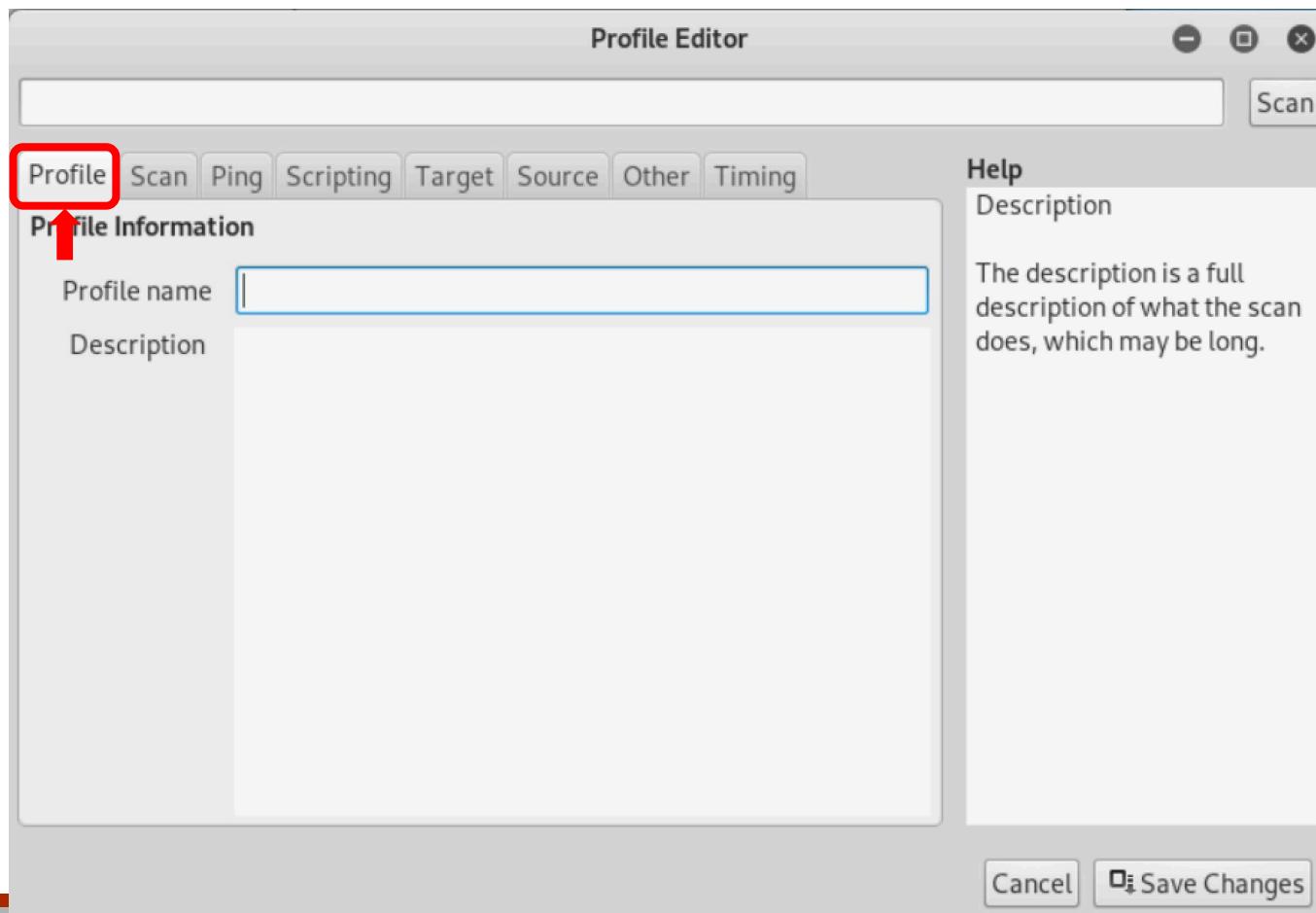
Zenmap

Creazione Profili di Scansione Personalizzati



Zenmap

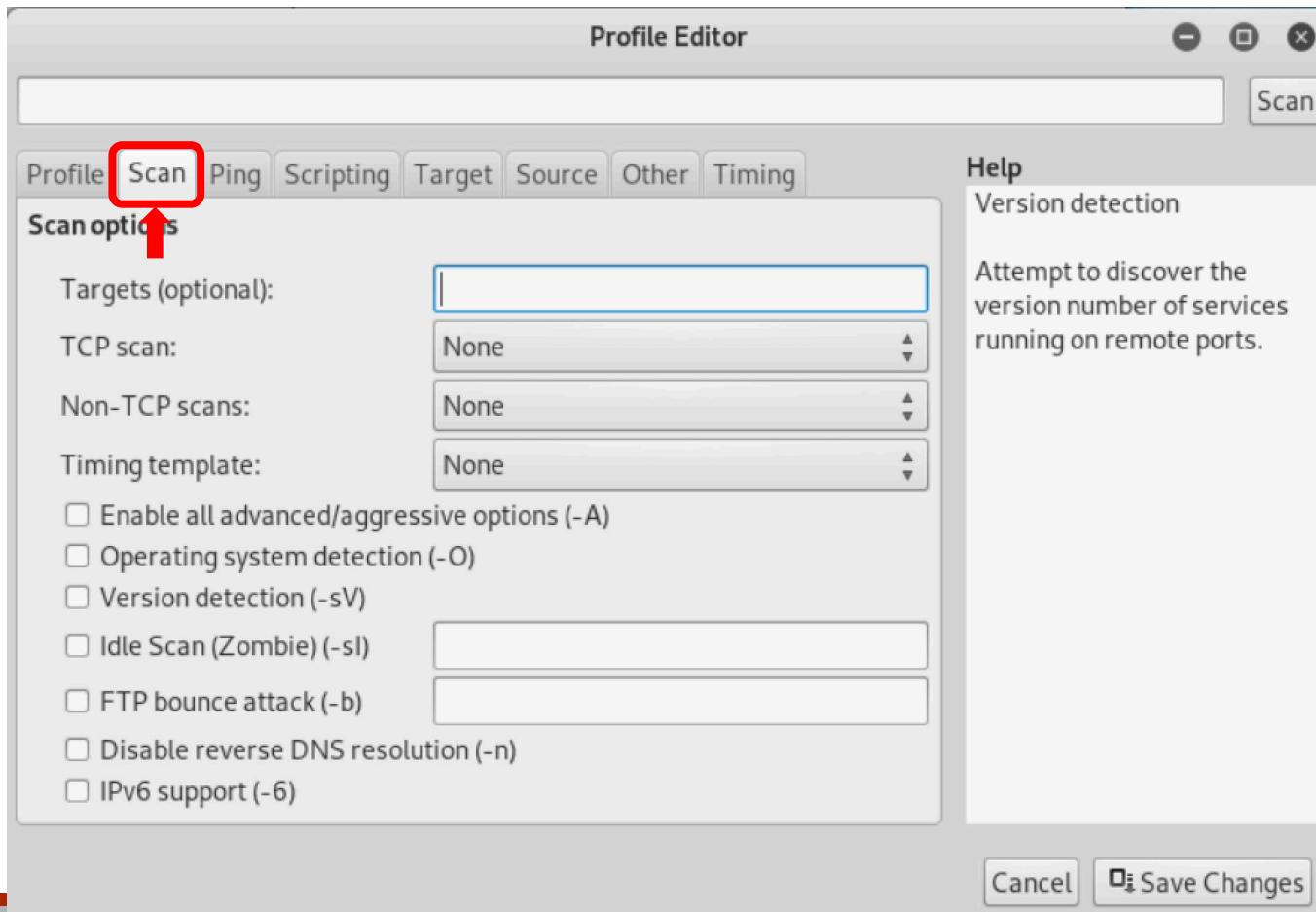
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

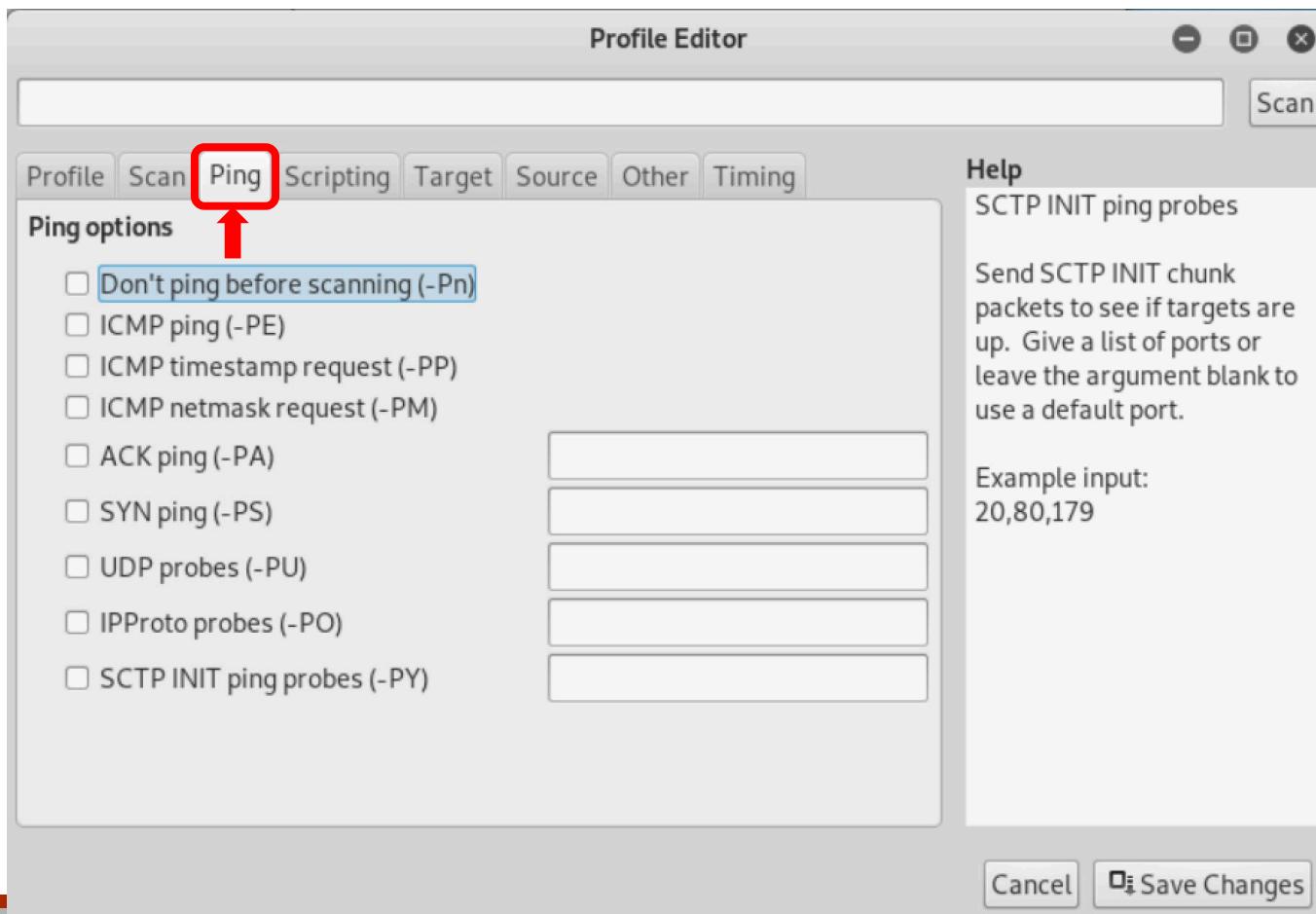
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

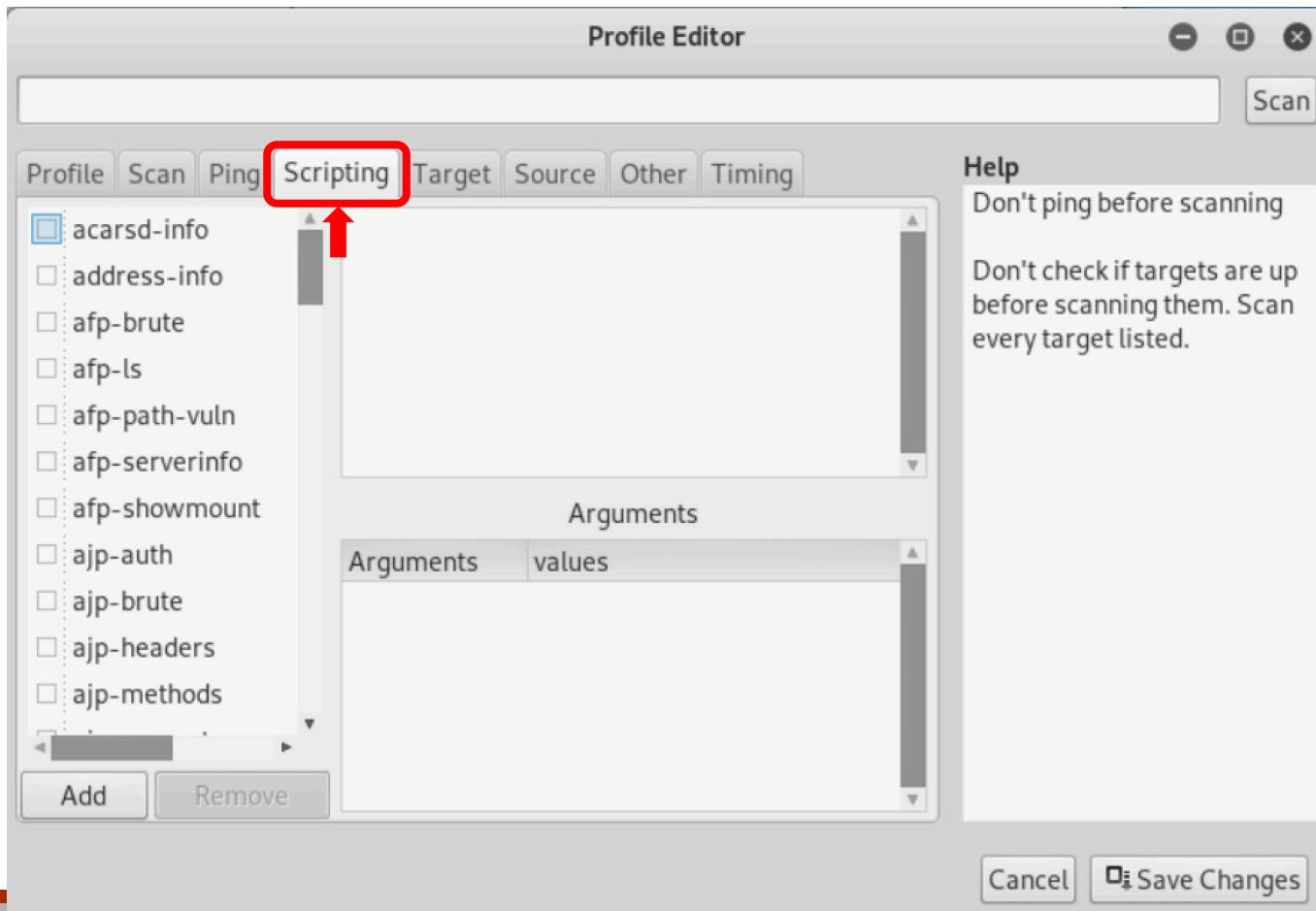
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

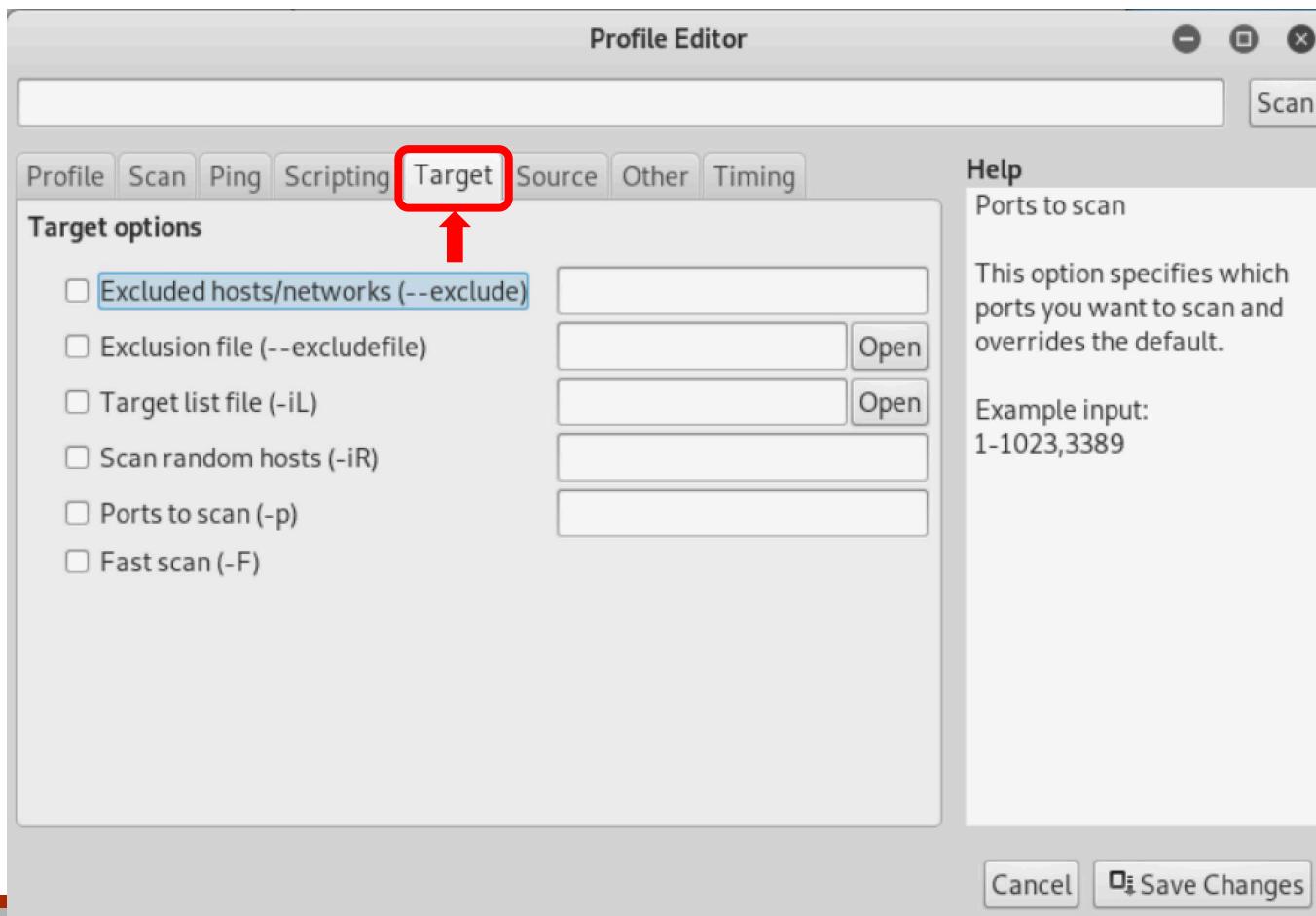
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

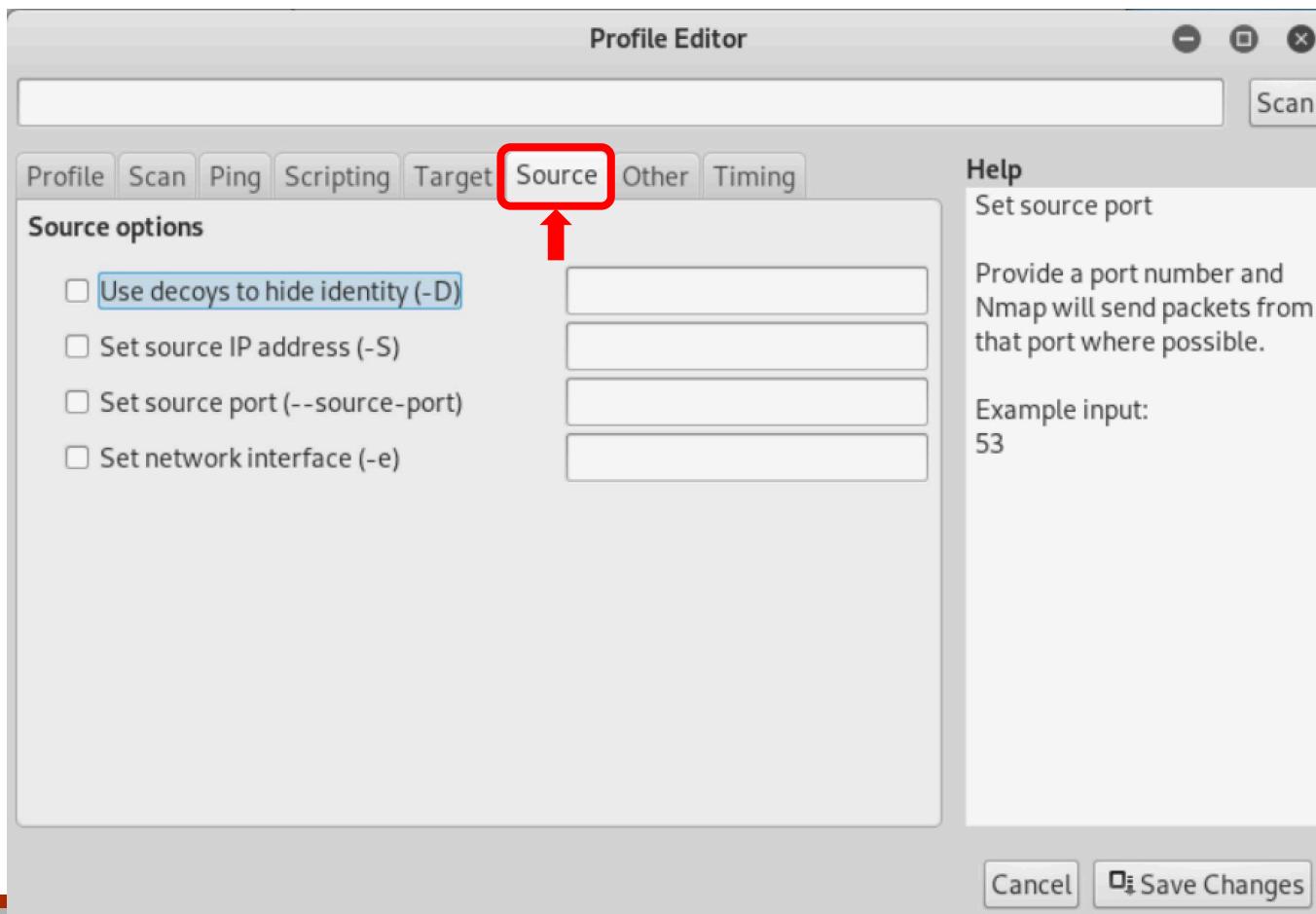
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

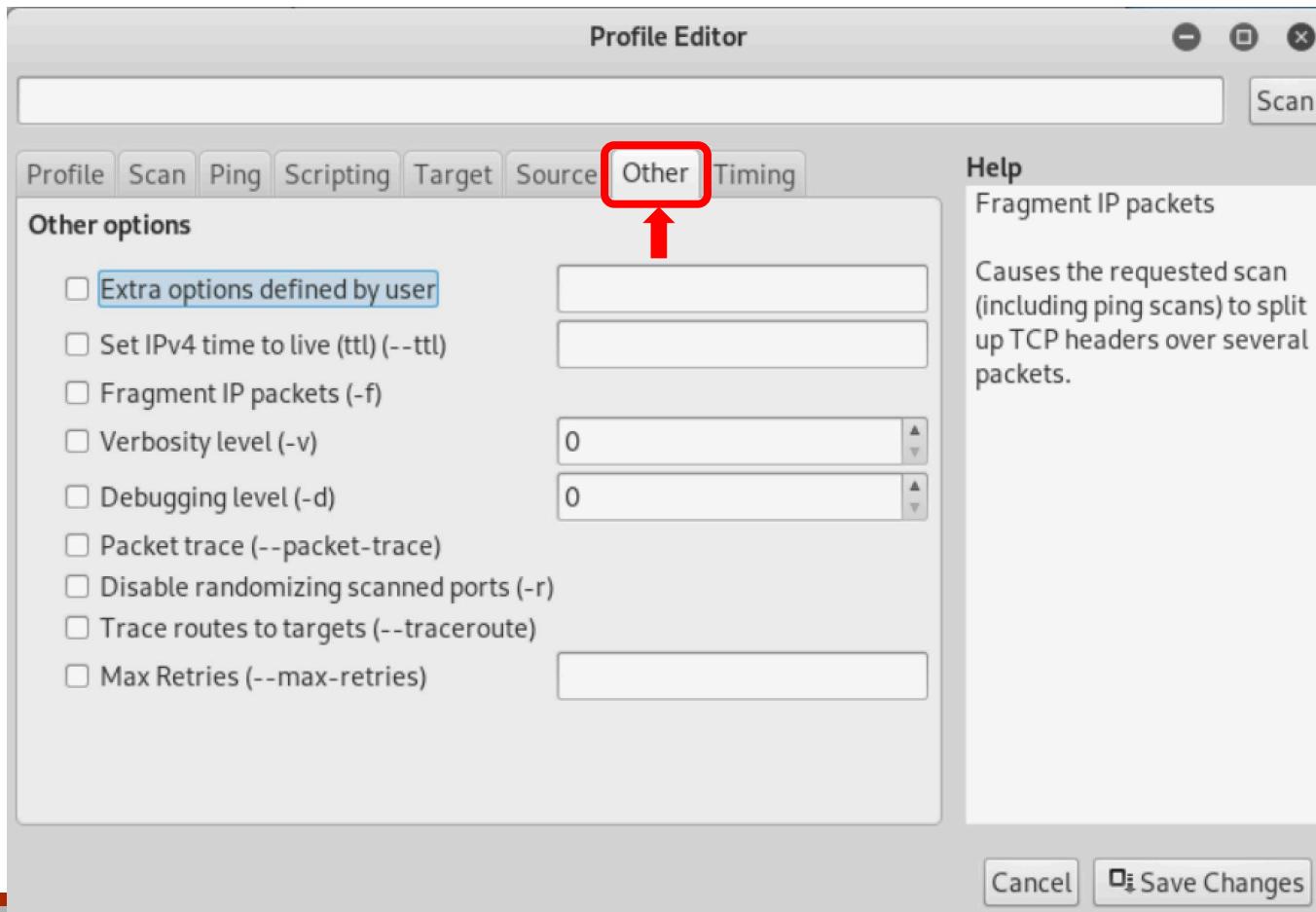
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

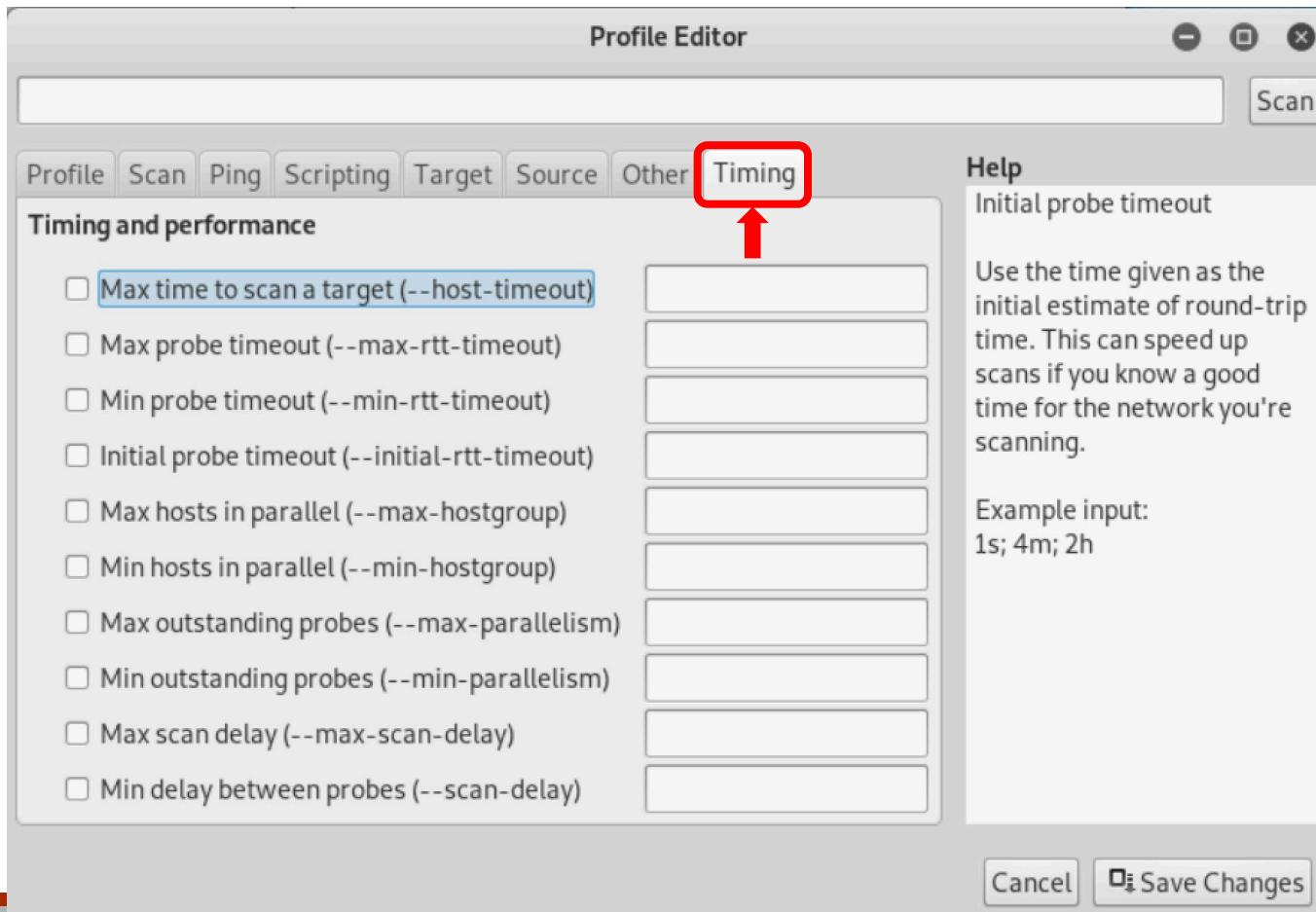
Creazione Profili di Scansione Personalizzati



Enumerating Target e Port Scanning

Zenmap

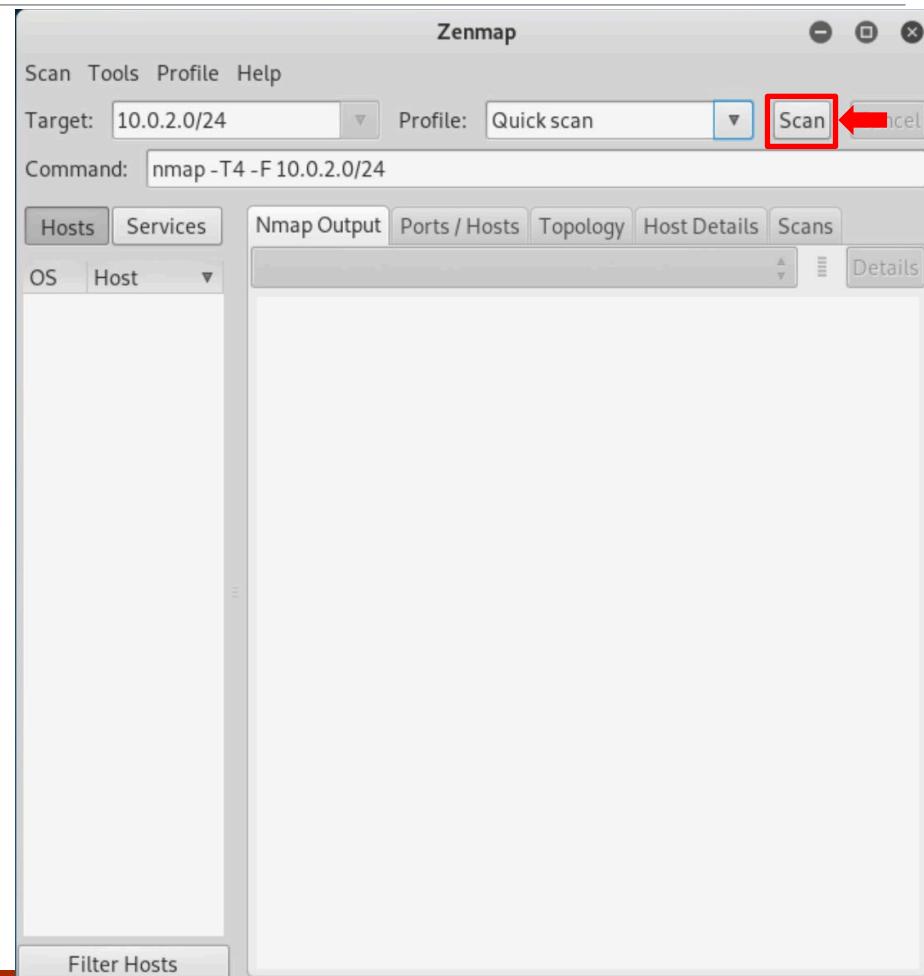
Creazione Profili di Scansione Personalizzati



Zenmap

Esempio

- Scansione sullo spazio di indirizzamento 10.0.2.0/24
- Scansione di tipo «Quick scan»
- Quick scan corrisponde al comando Nmap
 - `nmap -T4 -F 10.0.2.0/24`



Zenmap

Esempio

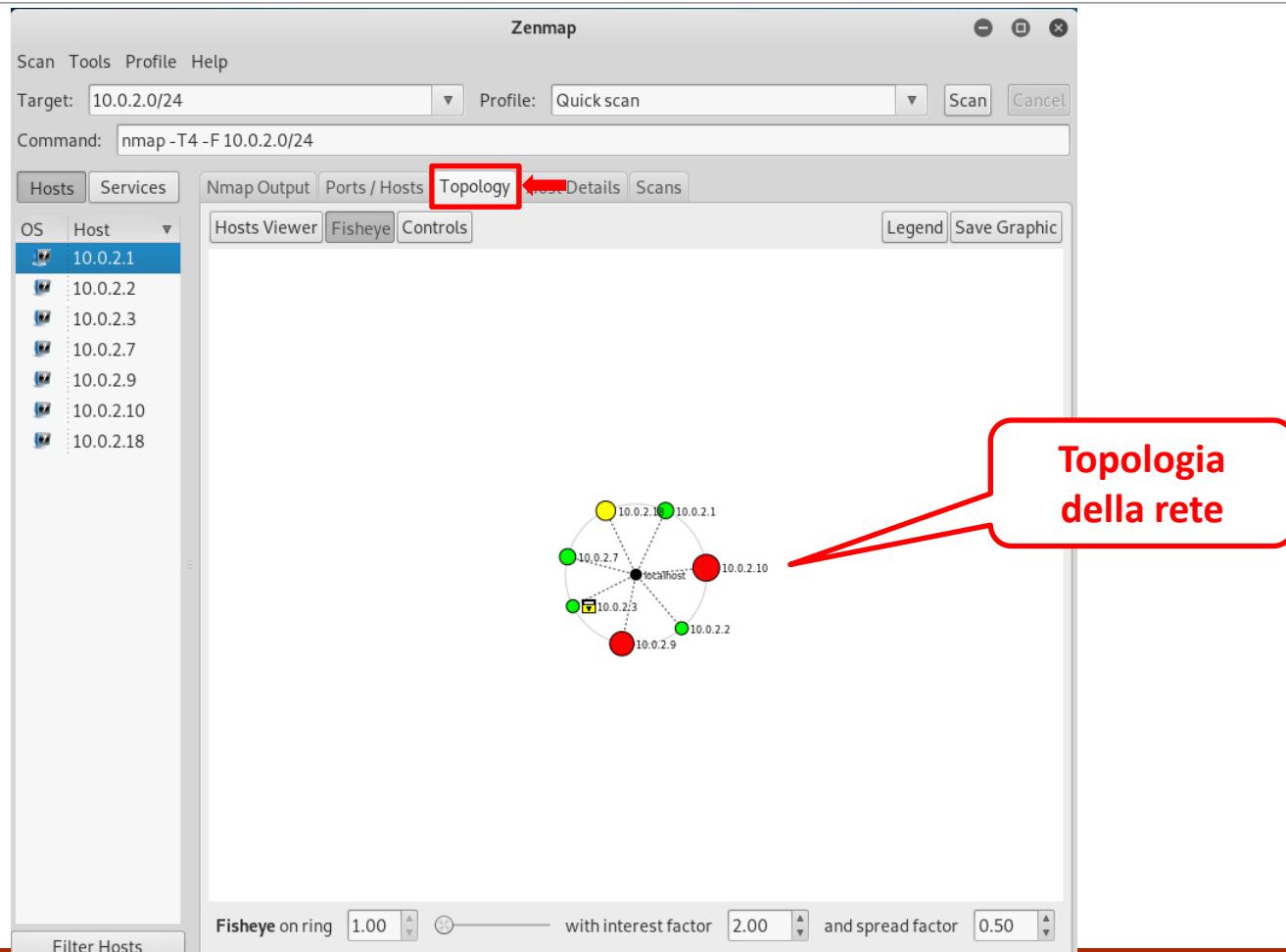
The screenshot shows the Zenmap interface with the following details:

- Scan Tools Profile Help**: Top menu bar.
- Target: 10.0.2.0/24**, **Profile: Quick scan**, **Scan** button: Scan configuration.
- Command: nmap -T4 -F 10.0.2.0/24**: Scan command entered in the command line.
- Hosts Services**: Tab bar.
- Nmap Output**: Active tab, displaying the scan results for host 10.0.2.1.
 - Starting Nmap 7.80 (https://nmap.org) at 2019-11-04 03:05 EST
 - Nmap scan report for **10.0.2.1**
 - Host is up (0.0088s latency).
 - Not shown**: 99 closed ports
 - PORT STATE SERVICE**
 - 53/tcp open domain**
 - MAC Address:** 52:54:00:12:35:00 (QEMU virtual NIC)
- Ports / Hosts Topology Host Details Scans**: Other tabs in the tab bar.
- OS Host**: Column headers for the host list.
- Host List**: List of hosts scanned:
 - 10.0.2.1
 - 10.0.2.2
 - 10.0.2.3
 - 10.0.2.7
 - 10.0.2.9
 - 10.0.2.10
 - 10.0.2.18
- Details**: A small panel on the right showing detailed information for the selected host (10.0.2.1).
- Filter Hosts**: Bottom left button.

Risultato della
scansione

Zenmap

Esempio



Enumerating Target e Port Scanning

Outline

- Concetti Introduttivi
- Suite Protocollare TCP/IP
- Formato dei Messaggi TCP e UDP
- Active Enumeration
 - Network Scanner Nmap
 - Zenmap
 - Unicornscan
 - Masscan
- Passive Enumeration
 - Shodan
 - ZoomEye
 - FOFA
 - Censys

Unicornscan

- Strumento molto potente, efficiente e versatile che consente di effettuare
 - Port Scanning: Rilevare le porte aperte sulle macchine target
 - Network Mapping: Identificare le macchine attive ed i relativi servizi
 - Banner Grabbing: Ottenere informazioni sulle versioni dei servizi (ad es., HTTP, SSH, etc)
- In dettaglio, fornisce le seguenti funzionalità
 - Scansioni TCP/UDP asincrone (non bloccanti)
 - Supporto a vari protocolli di rete (TCP, UDP, ICMP, etc)
 - Possibilità di modificare *packet rate* e *timing* (per realizzare Firewall Evasion)
 - Identifica i servizi di rete in esecuzione ed effettua OS Fingerprinting
 - Supporta vari formati di output (XML, testuale e binary log)



Unicornscan

- Per ottenere informazioni su Unicornscan
- **man unicornscan**

```
UNICORNSCAN(1)          Network Tools          UNICORNSCAN(1)

NAME
unicornscan Version 0.4.6b is a asynchronous network stimulus delivery/re-
sponse recording tool.

SYNOPSIS
unicornscan [-b, --broken-crc layer] [-B, --source-port port] [-d, --de-
lay-type type] [-D, --no-defpayload] [-e, --enable-module modules] [-E,
--proc-errors] [-F, --try-frags] [-G, --payload-group group] [-h, --help]
[-H, --do-dns] [-i, --interface interface] [-I, --immediate] [-j,
--ignore-seq ignore] [-l, --logfile file] [-L, --packet-timeoutdelay] [-m,
--mode mode] [-M, --module-dir directory] [-p, --ports string] [-P,
--pcap-filter filter] [-q, --covertness covertness] [-Q, --quiet] [-r,
--pps rate] [-R, --repeats repeats] [-s, --source-addr address] [-S,
--no-shuffle] [-t, --ip-ttl TTL] [-T, --ip-tos TOS] [-w, --safefile file]
[-W, --fingerprint fingerprint] [-v, --verbose] [-V, --version] [-z,
--sniff] [-Z, --drone-type type] target list

DESCRIPTION
unicornscan: ...

OPTIONS
```

Output Parziale



Unicornscan

- Unicornscan permette di
 - Otttenere scansioni più veloci rispetto ad Nmap
 - Soprattutto per quanto riguarda le scansioni *UDP*
 - Definire quanti pacchetti inviare al secondo - *Packets Per Second (PPS)*
 - Più alto è il valore di *Packets Per Second (PPS)*
 - Più veloce sarà la scansione
 - Maggiore sarà il carico di rete
 - Il valore di default relativo ai *PPS* è 300



Unicornscan

Esempio

- Effettuiamo una scansione *UDP* (parametro **-m U**) per le porte da **1** a **65535** e mostriamo il risultato in maniera «verbose» (parametro **-Iv**), inviando **10000** pacchetti al secondo (parametro **-r**)
 - **unicornscan -m U -Iv 10.0.2.6:1-65535 -r 10000**

```
root@kali:~# unicornscan -m U -Iv 10.0.2.6:1-65535 -r 10000
adding 10.0.2.6/32 mode `UDPscan' ports `1-65535' pps 10000
using interface(s) eth0
scanning 1.00e+00 total hosts with 6.55e+04 total packets, should take a little
e longer than 13 Seconds
UDP open 10.0.2.6:53544 ttl 64
UDP open 10.0.2.6:137 ttl 64
UDP open 10.0.2.6:53 ttl 64
UDP open 10.0.2.6:111 ttl 64
UDP open 10.0.2.6:2049 ttl 64
sender statistics 9790.8 pps with 65544 packets sent total
listener statistics 14 packets received 0 packets dropped and 0 interface drops
UDP open domain[ 53] from 10.0.2.6 ttl 64
UDP open sunrpc[ 111] from 10.0.2.6 ttl 64
UDP open netbios-ns[ 137] from 10.0.2.6 ttl 64
UDP open shilp[ 2049] from 10.0.2.6 ttl 64
UDP open unknown[53544] from 10.0.2.6 ttl 64
```

Outline

- Concetti Introduttivi
- Suite Protocollare TCP/IP
- Formato dei Messaggi TCP e UDP
- Active Enumeration
 - Network Scanner Nmap
 - Zenmap
 - Unicornscan
 - Masscan
- Passive Enumeration
 - Shodan
 - ZoomEye
 - FOFA
 - Censys

Masscan

Cos'è Masscan?

- Port scanner open source progettato per eseguire scansioni su larga scala in tempi rapidissimi
- Può inviare fino a 25 milioni di pacchetti al secondo, permettendo di scansionare l'intera rete Internet in pochi minuti
- Utilizza una trasmissione asincrona ed uno stack TCP/IP personalizzato, separato da quello del sistema operativo
- Produce output simili a quelli di Nmap, ma con prestazioni nettamente superiori in termini di velocità

Masscan

Caratteristiche Principali

- Fornisce
 - Supporto per scansioni di tipo TCP e UDP
 - Output in vari formati: XML, JSON, Grepable
 - Funzionalità avanzate: banner grabbing, scansione randomizzata, esclusione di IP, configurazione tramite file, etc
- Garantisce buona compatibilità con la sintassi di Nmap (ad es., **-p**, **-oX**, **--top-ports**)

Masscan

Esempi di Utilizzo

- Scansione di una rete locale:
 - `masscan 192.168.1.0/24 -p22,80,443`
- Scansione dell'intera Internet per la porta 443:
 - `masscan 0.0.0.0/0 -p443 --rate=1000000`
- Salvataggio dei risultati in formato XML:
 - `masscan 10.0.0.0/8 -p80 -oX risultati.xml`
- Scansione dei primi 1000 porti più comuni:
 - `masscan 192.168.0.0/16 --top-ports 1000`

Masscan

Esempi di Utilizzo

- Scansione di una rete locale:

- `masscan 192.168.1.0/24 -p22,80,443`

- Scansione dell'intera Internet per la porta 443:

- `masscan 0.0.0.0/0 -p443 --rate=1000000`

- Salvataggio dei risultati in formato XML:

- `masscan 10.0.0.0/8 -p80 -oX risultati.xml`

- Scansione dei primi 1000 porti più comuni:

- `masscan 192.168.0.0/16 --top-ports 1000`

L'indirizzo 0.0.0.0/0 è una notazione CIDR (Classless Inter-Domain Routing) che significa: «Tutti gli indirizzi IPv4 esistenti»

0.0.0.0 è l'indirizzo iniziale.

/0 indica che nessun bit è fissato, quindi la rete comprende tutti i 4,3 miliardi di indirizzi IPv4 (da 0.0.0.0 a 255.255.255.255)