

SCANSIONI CON NMAP

Sulla macchina virtuale (VM) Kali Linux:

Rimetto in rete interna con il comando sottostante, dopodichè nel config cambio l'IP:

```
(kali@kali)-[~]  
$ sudo nano /etc/network/interfaces
```

Sulla VM **Metasploitable**:

Ho cambiato l'IP di **Metasploitable** a 192.168.32.102

con: "sudo nano /etc/network/interfaces", modificandolo nel config

```
msfadmin@metasploitable:~$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000  
    link/ether 08:00:27:5f:e1:0c brd ff:ff:ff:ff:ff:ff  
    inet 192.168.32.102/24 brd 192.168.32.255 scope global eth0  
    inet6 fe80::a00:27ff:fe5f:e10c/64 scope link  
        valid_lft forever preferred_lft forever
```

Ho controllato se le reti sono collegate tra loro, con *ping*:

```
msfadmin@metasploitable:~$ ping 192.168.32.100  
PING 192.168.32.100 (192.168.32.100) 56(84) bytes of data.  
64 bytes from 192.168.32.100: icmp_seq=1 ttl=64 time=5.51 ms  
64 bytes from 192.168.32.100: icmp_seq=2 ttl=64 time=0.253 ms  
64 bytes from 192.168.32.100: icmp_seq=3 ttl=64 time=0.259 ms  
64 bytes from 192.168.32.100: icmp_seq=4 ttl=64 time=0.233 ms  
64 bytes from 192.168.32.100: icmp_seq=5 ttl=64 time=0.325 ms  
  
[2]+  Stopped                  ping 192.168.32.100
```

Su VM **Kali Linux**:

Eseguo l'intercettazione con **Nmap**, target: **VM Metasploitable**, IP: **192.168.32.102**:

Eseguo l'**Host Discovery** con **-Pn** (che ipotizza tutti gli *host attivi*):

```
(kali㉿kali)-[~]  
$ nmap -Pn 192.168.32.102  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-18 08:47 EDT  
Nmap scan report for 192.168.32.102  
Host is up (0.00032s latency).  
Not shown: 977 closed tcp ports (conn-refused)  
PORT      STATE SERVICE  
21/tcp    open  ftp  
22/tcp    open  ssh  
23/tcp    open  telnet  
25/tcp    open  smtp  
53/tcp    open  domain  
80/tcp    open  http  
111/tcp   open  rpcbind  
139/tcp   open  netbios-ssn  
445/tcp   open  microsoft-ds  
512/tcp   open  exec  
513/tcp   open  login  
514/tcp   open  shell  
1099/tcp  open  rmiregistry  
1524/tcp  open  ingreslock  
2049/tcp  open  nfs  
2121/tcp  open  ccproxy-ftp  
3306/tcp  open  mysql  
5432/tcp  open  postgresql  
5900/tcp  open  vnc  
6000/tcp  open  X11  
6667/tcp  open  irc  
8009/tcp  open  ajp13  
8180/tcp  open  unknown  
  
Nmap done: 1 IP address (1 host up) scanned in 14.24 seconds
```

Target: VM Metasploitable, IP: 192.168.32.102:

Eseguo la scansione del **TCP** (*Transmission Control Protocol*) con **-sV** (notiamo la **versione**):

```
(kali㉿kali)-[~]
└─$ nmap -sV -p 1-1024 192.168.32.102
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-18 08:50 EDT
Nmap scan report for 192.168.32.102
Host is up (0.00026s latency).
Not shown: 1012 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rshcd
513/tcp   open  login?       Netkit rshd
514/tcp   open  shell
Service Info: Host: metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 49.43 seconds
```

Target: VM Metasploitable, IP: 192.168.32.102:

Effettuo la scansione **SYN** con “**sudo nmap -sS -p 1-1024 192.168.32.102**”
(recuperiamo anche il **MAC Address**):

```
(kali㉿kali)-[~]
└─$ sudo nmap -sS -p 1-1024 192.168.32.102
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-18 08:56 EDT
Nmap scan report for 192.168.32.102
Host is up (0.000067s latency).
Not shown: 1012 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
MAC Address: 08:00:27:5F:E1:0C (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.21 seconds
```

Target: VM Metasploitable, IP: 192.168.32.102:

Eseguo la scansione con lo switch (opzione) **-A**, con:

```
(kali㉿kali)-[~]
$ nmap -A -p 1-1024 192.168.32.102
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-18 08:58 EDT
Nmap scan report for 192.168.32.102
Host is up (0.00030s latency).
Not shown: 1012 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
| ftp-syst:
|   STAT:
|   FTP server status:
|     Connected to 192.168.32.100
|     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
|_ ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|_  1024 600fcfe1c05f6a74d69024fac4d56ccd (DSA)
|_  2048 5656240f211ddea72bae61b1243de8f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_ smtp-command: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
|_ sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
53/tcp    open  domain       ISC BIND 9.4.2
| dns-nsid:
|_  bind.version: 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_ http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_ http-title: Metasploitable2 - Linux
111/tcp   open  rpcbind      2 (RPC #100000)
| rpcinfo:
|   program version  port/proto  service
|   100000  2          111/tcp    rpcbind
|   100000  2          111/udp    rpcbind
|   100003  2,3,4      2049/tcp   nfs
|   100003  2,3,4      2049/udp   nfs
|   100005  1,2,3      33586/udp  mountd
|   100005  1,2,3      50743/tcp  mountd
|   100021  1,3,4      51917/udp  nlockmgr
|   100021  1,3,4      60536/tcp  nlockmgr
|   100024  1          46383/tcp  status
|   100024  1          47757/udp  status
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rshd
513/tcp   open  login?
514/tcp   open  shell        Netkit rshd
Service Info: Host: metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_ clock-skew: mean: 1h59m58s, deviation: 2h50m17s, median: -26s
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
|_ smb2-time: Protocol negotiation failed (SMB2)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_ System time: 2023-05-18T08:59:34-04:00
|_ nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 112.46 seconds
```

Scansioni **TCP/SYN** intercettate con Wireshark:

Per farlo mi collego su **Wireshark** su **eth0**,
metto il **filtro (per TCP)**:

tcp and ip.src == <indirizzo_IP_Kali> and ip.dst == <indirizzo_IP_Metasploitable>

Il filtro per SYN:

*tcp.flags.syn == 1 and ip.src == <indirizzo_IP_Kali> and ip.dst ==
<indirizzo_IP_Metasploitable>*

Intercettazione **TCP** con Wireshark:

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp and ip.src == <192.168.32.100> and ip.dst == <192.168.32.102>

No.	Time	Source	Destination	Protocol	Length	Info
2539	42.674097217	192.168.32.100	192.168.32.102	TCP	66	37908 → sunrpc(111) [FIN, ACK] Seq=45 Ack=29 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2540	42.674019476	192.168.32.100	192.168.32.102	TCP	66	37922 → sunrpc(111) [FIN, ACK] Seq=45 Ack=29 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2541	42.674030236	192.168.32.100	192.168.32.102	HTTP	106	GET / HTTP/1.1
2542	42.674085646	192.168.32.102	192.168.32.100	TCP	66	http(80) → 43326 [ACK] Seq=1 Ack=41 Win=5888 Len=0 TSval=410635 TSecr=2084737318
2543	42.674117783	192.168.32.102	192.168.32.100	TCP	66	sunrpc(111) → 37878 [FIN, ACK] Seq=37 Ack=46 Win=5888 Len=0 TSval=410635 TSecr=2084737318
2544	42.674117846	192.168.32.102	192.168.32.100	TCP	66	sunrpc(111) → 37894 [FIN, ACK] Seq=29 Ack=46 Win=5888 Len=0 TSval=410635 TSecr=2084737318
2545	42.674123647	192.168.32.100	192.168.32.102	TCP	66	37878 → sunrpc(111) [ACK] Seq=46 Ack=38 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2546	42.674132726	192.168.32.100	192.168.32.102	TCP	66	37894 → sunrpc(111) [ACK] Seq=46 Ack=39 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2547	42.674161989	192.168.32.102	192.168.32.100	TCP	66	sunrpc(111) → 37908 [FIN, ACK] Seq=29 Ack=46 Win=5888 Len=0 TSval=410635 TSecr=2084737318
2548	42.674162041	192.168.32.102	192.168.32.100	TCP	66	sunrpc(111) → 37922 [FIN, ACK] Seq=29 Ack=46 Win=5888 Len=0 TSval=410635 TSecr=2084737318
2549	42.674166940	192.168.32.100	192.168.32.102	TCP	66	37908 → sunrpc(111) [ACK] Seq=46 Ack=39 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2550	42.674174488	192.168.32.100	192.168.32.102	TCP	66	37922 → sunrpc(111) [ACK] Seq=46 Ack=39 Win=64256 Len=0 TSval=2084737318 TSecr=410635
2551	42.679562001	192.168.32.102	192.168.32.100	TCP	1147	http(80) → 43326 [PSH, ACK] Seq=1 Ack=41 Win=5888 Len=1081 TSval=410636 TSecr=2084737318 [TCP segment of a
2552	42.679577620	192.168.32.100	192.168.32.102	TCP	66	43326 → http(80) [ACK] Seq=41 Ack=1082 Win=64128 Len=0 TSval=2084737323 TSecr=410636
2553	42.679651895	192.168.32.100	192.168.32.102	TCP	66	43326 → http(80) [FIN, ACK] Seq=41 Ack=1082 Win=64128 Len=0 TSval=2084737324 TSecr=410636
2554	42.680229360	192.168.32.102	192.168.32.100	HTTP	71	HTTP/1.1 200 OK (text/html)
2555	42.680242548	192.168.32.100	192.168.32.102	TCP	54	43326 → http(80) [RST] Seq=42 Win=0 Len=0
2556	42.680296147	192.168.32.102	192.168.32.100	TCP	66	http(80) → 43326 [FIN, ACK] Seq=1087 Ack=42 Win=5888 Len=0 TSval=410636 TSecr=2084737324
2557	42.680301094	192.168.32.100	192.168.32.102	TCP	54	43326 → http(80) [RST] Seq=42 Win=0 Len=0
2558	42.680617659	192.168.32.102	192.168.32.100	TCP	66	login(513) → 37506 [ACK] Seq=1 Ack=29 Win=5888 Len=0 TSval=410637 TSecr=2084737287
2559	43.645309993	PcsCompu. 5f:e1:0c	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.102
2560	44.645660616	PcsCompu. 5f:e1:0c	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.102
2561	47.634634711	192.168.32.102	192.168.32.100	Rlogin	67	Data: \001
2562	47.634653704	192.168.32.100	192.168.32.102	TCP	54	37506 → login(513) [RST] Seq=20 Win=0 Len=0
2563	47.634700445	192.168.32.102	192.168.32.100	TCP	66	login(513) → 37506 [RST, ACK] Seq=2 Ack=29 Win=5888 Len=0 TSval=411132 TSecr=2084737287

Flags: 0x011 (FIN, ACK)

0000 = Reserved: Not set
..... = Accurate ECN: Not set
..... = Congestion Window Reduced: Not set
..... = ECN-Echo: Not set
..... = Urgent: Not set
..... = Acknowledgment: Set
..... = Push: Not set
..... = Reset: Not set

Intercettazione **SYN** con Wireshark:

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.flags.syn == 1 and ip.src == <192.168.32.100> and ip.dst == <192.168.32.102>

No.	Time	Source	Destination	Protocol	Length	Info
2034	13.131610046	192.168.32.102	192.168.32.100	TCP	60	ptp-event(319) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2035	13.131610395	192.168.32.100	192.168.32.102	TCP	58	58876 → sun-dr(665) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2036	13.131644993	192.168.32.102	192.168.32.100	TCP	60	exp2(1022) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2037	13.131644947	192.168.32.102	192.168.32.100	TCP	60	sun-dr(193) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2038	13.131675261	192.168.32.102	192.168.32.100	TCP	60	sun-dr(665) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2039	13.131682074	192.168.32.100	192.168.32.102	TCP	58	58876 → alpes(463) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2040	13.131691787	192.168.32.100	192.168.32.102	TCP	58	58876 → 971 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2041	13.131712279	192.168.32.100	192.168.32.102	TCP	58	58876 → 959 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2042	13.131717628	192.168.32.100	192.168.32.102	TCP	58	58876 → 938 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2043	13.131744277	192.168.32.102	192.168.32.100	TCP	60	alpes(463) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2044	13.131744316	192.168.32.102	192.168.32.100	TCP	60	971 → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2045	13.131744386	192.168.32.102	192.168.32.100	TCP	60	959 → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2046	13.131744301	192.168.32.102	192.168.32.100	TCP	60	938 → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2047	13.131757806	192.168.32.100	192.168.32.102	TCP	58	58876 → 803 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2048	13.131792400	192.168.32.100	192.168.32.102	TCP	58	58876 → magenta-logic(313) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2049	13.131798874	192.168.32.100	192.168.32.102	TCP	58	58876 → 982 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2050	13.131817432	192.168.32.102	192.168.32.100	TCP	60	803 → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2051	13.131843471	192.168.32.102	192.168.32.100	TCP	60	magenta-logic(313) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2052	13.131843495	192.168.32.102	192.168.32.100	TCP	60	982 → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2053	13.131854884	192.168.32.100	192.168.32.102	TCP	58	58876 → tempo(526) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2054	13.131861730	192.168.32.100	192.168.32.102	TCP	58	58876 → clearcase(371) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2055	13.131891116	192.168.32.100	192.168.32.102	TCP	58	58876 → dec-dlm(625) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2056	13.131909322	192.168.32.100	192.168.32.102	TCP	58	58876 → daytime(13) [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2057	13.131917100	192.168.32.102	192.168.32.100	TCP	60	tempo(526) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2058	13.131917141	192.168.32.102	192.168.32.100	TCP	60	clearcase(371) → 58876 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

0110 = Header Length: 24 bytes (6)

Flags: 0x002 (SYN)

0000 = Reserved: Not set
..... = Accurate ECN: Not set
..... = Congestion Window Reduced: Not set
..... = ECN-Echo: Not set
..... = Urgent: Not set
..... = Acknowledgment: Not set
..... = Push: Not set

Syn (tcp.flags.syn), 1 byte

Packets: 2070 - Displayed: 2070 (100.0%) Profile: Default

Che differenze notiamo con le intercettazioni TCP/SYN:

- **Tipi di pacchetti:** Durante una scansione TCP, possiamo vedere pacchetti con flags TCP come SYN, ACK, e FIN. La scansione SYN, invece, si concentra principalmente su pacchetti con flag SYN.
- **Stato della connessione:** Durante una scansione TCP, osserviamo la sequenza di pacchetti che costituiscono una connessione TCP completa, ad esempio la sequenza SYN-ACK-SYN-ACK. Durante la scansione SYN, vediamo principalmente pacchetti SYN inviati ai sistemi di destinazione.
- **Risposte dei pacchetti:** Durante una scansione TCP, osserviamo pacchetti di risposta come SYN-ACK e ACK per stabilire la connessione. Durante la scansione SYN, vediamo principalmente risposte SYN-ACK dai sistemi di destinazione.
- **Porte di destinazione:** Durante una scansione TCP, vediamo i pacchetti TCP inviati a porte specifiche per determinare se sono aperte o chiuse. Durante la scansione SYN, saremo principalmente interessati a identificare le porte che rispondono con pacchetti SYN-ACK.
- **Numero di sequenza:** Durante una scansione TCP, osserviamo i numeri di sequenza che vengono scambiati tra i sistemi durante la connessione. Durante la scansione SYN, vediamo principalmente i numeri di sequenza nei pacchetti SYN e SYN-ACK.