

# NETWORK SCANNING CON NMAP

Nell'esercizio di oggi siamo andati a scansionare la macchina metasploitable con il tool nmap. Abbiamo eseguito diversi tipi di scan:

- Scansione TCP sulle porte well-known
- Scansione SYN sulle porte well known
- Scansione con switch "-A" sulle porte well-known

Abbiamo trovato le porte well-known aperte sul nostro terminale Kali, cioè tra quelle che vanno dalla porta 0 alla 1023:

PORTE	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

Tab1

Durante la scansione abbiamo aperto Wireshark intercettando le richieste inviate dalla macchina sorgente.

The screenshot displays a Kali Linux terminal window and a Wireshark network traffic capture. The terminal window shows the command `nmap 192.168.32.101 -T` and its output, which lists open ports and services. The Wireshark interface shows a packet capture of the scan traffic, with red arrows highlighting specific packets.

Terminal Output:

```
--unprivileged: Assume the user lacks raw socket privileges
-V: Print version number
-h: Print this help summary page.
EXAMPLES:
nmap -v -A scanme.nmap.org
nmap -v -sn 192.168.0.0/16 10.0.0.0/8
nmap -v -iR 10000 -f -g 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES

(kali@kali): ~
$ nmap 192.168.32.101 -T
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-10 08:18 EST
Nmap scan report for 192.168.32.101
Host is up (0.0003s latency).
Not shown: 927 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
1899/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2849/tcp  open  nfs
2121/tcp  open  cproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5908/tcp  open  vnc
6000/tcp  open  x11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

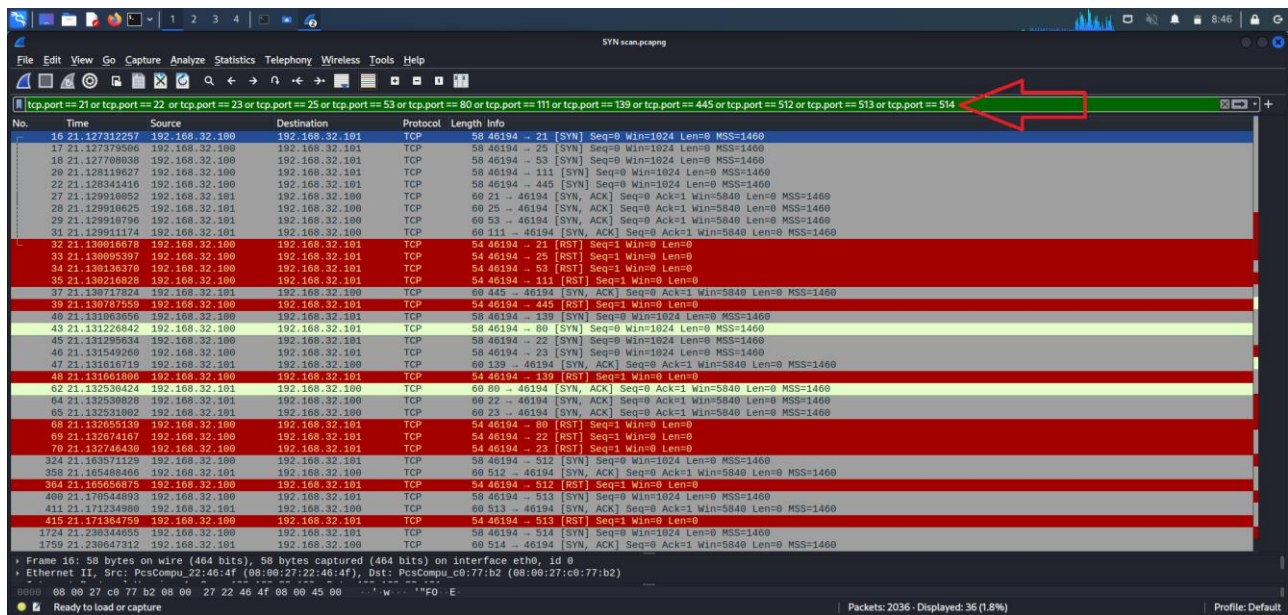
Nmap done: 1 IP address (1 host up) scanned in 13.52 seconds

(kali@kali): ~
```

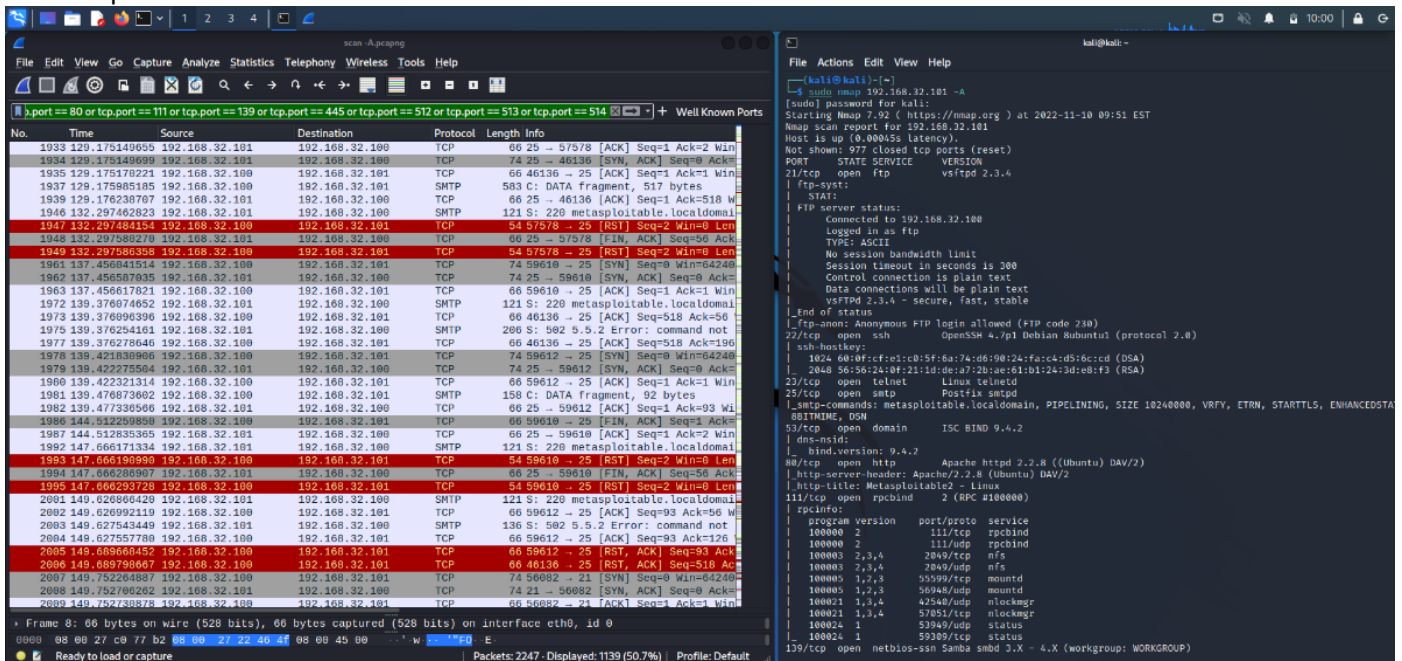
Wireshark Packet Capture:

The Wireshark interface shows a packet capture of the scan traffic. Red arrows highlight specific packets, including a SYN packet (74.34562 -> 3389 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK=0) and an ACK packet (60.3389 -> 34582 [ACK] Seq=1 Ack=1 Win=0 Len=0).

Abbiamo filtrato per vedere solo le porte interessate (Tab1):



Fatte le ricerche per nmap 192.168.32.101 -sT e per nmap 192.168.32.101 -sS abbiamo fatto una scansione nmap 192.168.32.101 -A:



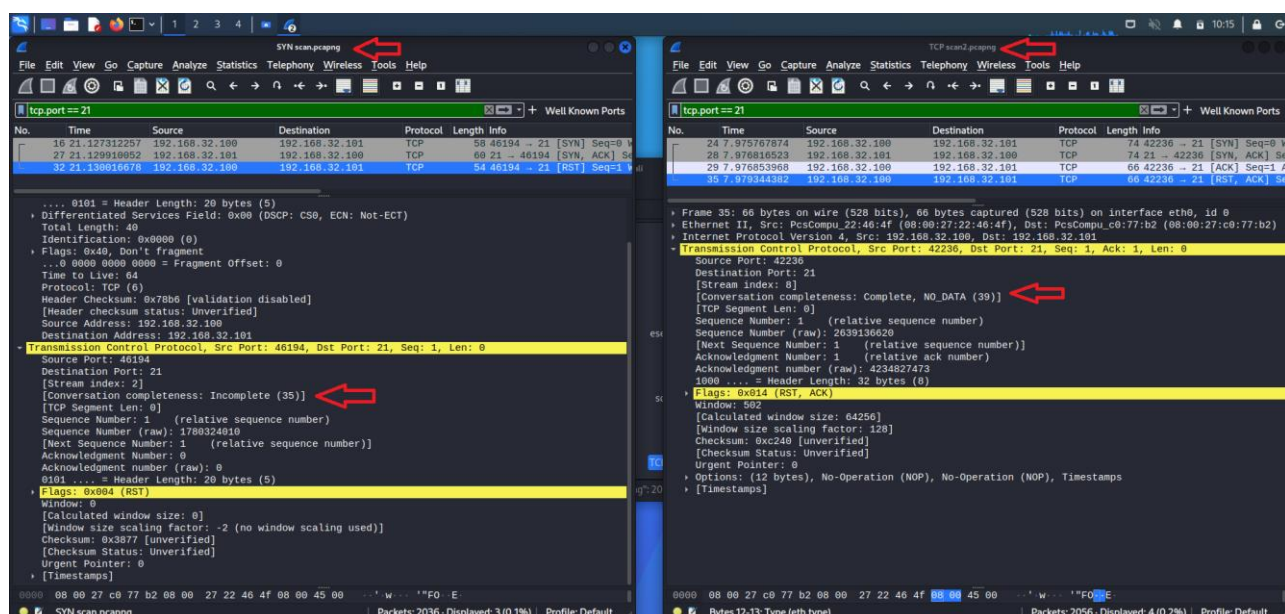
Come possiamo vedere la scansione nmap 192.168.32.101 -A fa una scansione molto più approfondita e aggressiva rispetto alle altre viste prima (img sopra e sotto).

```
(kali@kali)~$ sudo nmap 192.168.32.101 -A
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-10 08:56 EST
Nmap scan report for 192.168.32.101
Host is up (0.0013s latency).
Not shown: 977 closed tcp ports (reset)
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 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
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (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)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_smtp-command: metaspoitable.localdomain, PIPELINING, SIZE 1024000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
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 55599/tcp mountd
|_100005 1,2,3 56948/udp mountd
|_100021 1,3,4 42540/udp nlockmgr
|_100021 1,3,4 57051/tcp nlockmgr
|_100024 1 53949/udp status
|_100024 1 59309/tcp 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)
```

Abbiamo poi evidenziato le porte well-known filtrando le scansioni di Wireshark per notare le differenze:

Qui vediamo un esempio di scansione della stessa porta (porta 21) dove possiamo vedere come la scansione -sT, essendo più invasiva rispetto alla -sS, completa la connessione 3-way-handshake.

Infatti, la scansione TCP darà la connessione completa, mentre, la scansione SYN darà connessione incompleta, dato che verrà chiusa dopo il SYN/ACK.



FONTE SCAN	TARGET SCAN	TIPO DI SCAN	RISULTATO OTTENUTO
192.168.32.100	192.168.32.101	nmap 192.168.32.101 -sT	23 servizi aperte di cui 12 di porte well-known

192.168.32.100	192.168.32.101	nmap 192.168.32.101 -sS	23 servizi aperte di cui 12 di porte well-known
----------------	----------------	-------------------------	--