

## ESERCIZIO W9D1 PARTE 2

Il prossimo esercizio ci permetterà di esplorare meglio nmap e le sue funzioni. Ci vengono proposte diverse tracce da portare a termine, cominciamo con la prima:

### Traccia 1

Vedremo da vicino nmap e i suoi comandi. Sulle base delle nozioni viste nella lezione teorica eseguiremo diversi tipi di scan sulle macchine metasploitable, come di seguito:

- Scansione TCP sulle porte well-known.
- Scansione SYN sulle porte well-known.
- Scansione con switch «-A» sulle porte well-known.
- Evidenziare la differenza tra la scansione completa TCP e la scansione SYN intercettando le richieste inviate dalle macchine sorgente con Wireshark.

### Scansione TCP sulle porte well-known

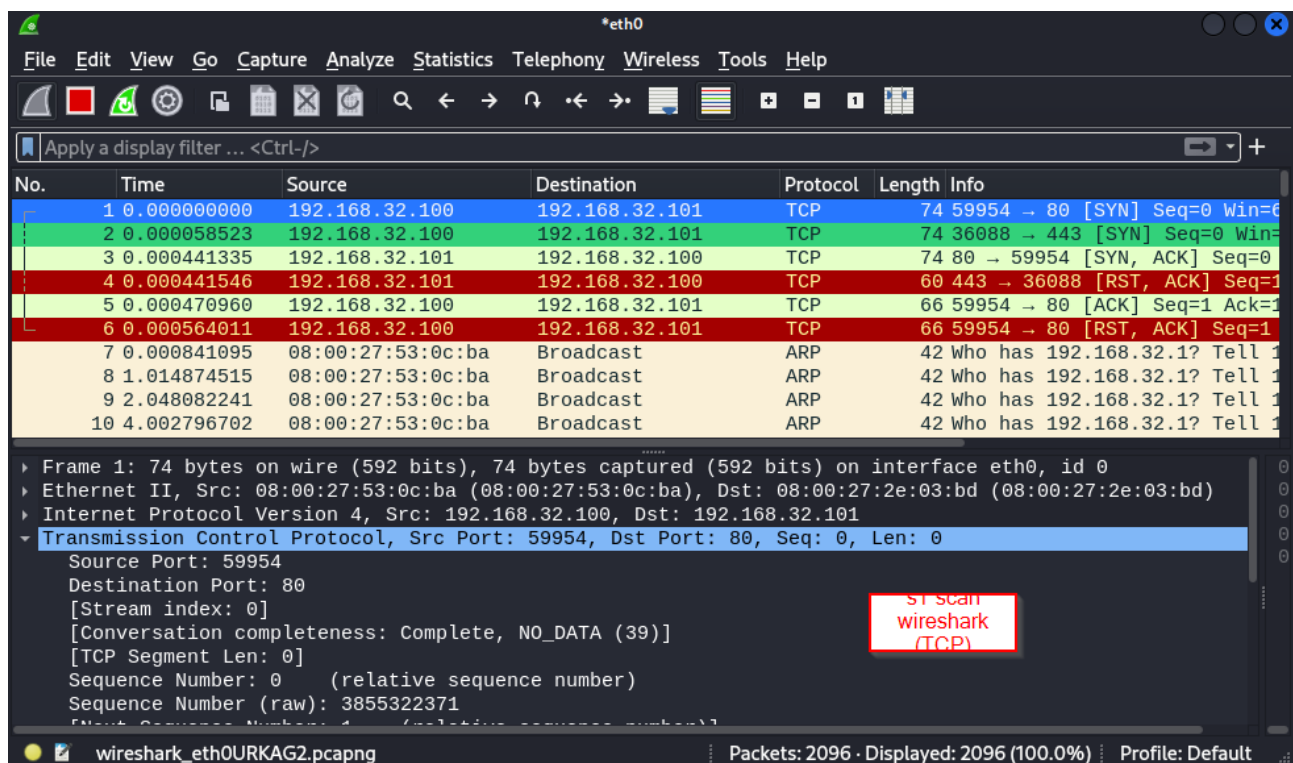
```
(kali㉿kali)-[~]
$ nmap -sT -p0-1024 192.168.32.101

Starting Nmap 7.93 ( https://nmap.org ) at 2023-12-21 06:18 EST
Nmap scan report for 192.168.32.101
Host is up (0.00049s latency).
Not shown: 1013 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

Nmap done: 1 IP address (1 host up) scanned in 13.16 seconds
```

Scan nmap  
sT(TCP)

Su Kali Linux inseriamo sul terminale il comando `nmap -sT -p0-1024 192.168.32.101`. Il termine `-sT` indica che il client completerà il three way handshake, generando più rumore a livello di rete, `p0-1024` definirà il range di porte di cui verificare la disponibilità, mentre l'indirizzo IPv4 (Metasploitable) rappresenterà il target in cui ricercare le porte. Nel frattempo teniamo pronto Wireshark, sulla scheda di rete `eth0`, per intercettare i pacchetti.



Dopo l'invio del protocollo ARP per individuare l'indirizzo MAC avente 192.168.32.101 all'interno della rete, partiranno le richieste TCP. Aiutandoci con il terminale di Linux, andiamo a scovare una delle porte individuate durante la raccolta. Inseriamo nel filtro `tcp.port == 514` per controllare meglio gli scambi tra client e server.

three way handshake completamente stabilito con la porta 514

No.	Time	Source	Destination	Protocol	Length	Info
395	13.042210139	192.168.32.100	192.168.32.101	TCP	74	54664 → 514 [SYN] Seq=0 Win=0
401	13.042347789	192.168.32.101	192.168.32.100	TCP	74	514 → 54664 [SYN, ACK] Seq=0
402	13.042367489	192.168.32.100	192.168.32.101	TCP	66	54664 → 514 [ACK] Seq=1 Ack=1
476	13.044482016	192.168.32.100	192.168.32.101	TCP	66	54664 → 514 [RST, ACK] Seq=1

Frame 395: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0, id 0  
 Ethernet II, Src: 08:00:27:53:0c:ba (08:00:27:53:0c:ba), Dst: 08:00:27:2e:03:bd (08:00:27:2e:03:bd)  
 Internet Protocol Version 4, Src: 192.168.32.100, Dst: 192.168.32.101  
 Transmission Control Protocol, Src Port: 54664, Dst Port: 514, Seq: 0, Len: 0  
 Source Port: 54664  
 Destination Port: 514  
 [Stream index: 183]  
 [Conversation completeness: Complete, NO\_DATA (39)]  
 [TCP Segment Len: 0]  
 Sequence Number: 0 (relative sequence number)  
 Sequence Number (raw): 4144670495  
 Window Sequence Number: 0 (relative sequence number)

wireshark\_eth0URKAG2.pcapng Packets: 2098 · Displayed: 4 (0.2%) Profile: Default

Il nostro client (Kali Linux) genera una porta per creare un collegamento con la porta 514. Il server risponde al protocollo SYN inviato ed il tutto viene poi sincronizzato con l'ultimo ACK del client. Questo significa che la porta è disponibile all'utilizzo.

TWH non riuscito con la porta 700

No.	Time	Source	Destination	Protocol	Length	Info
1684	13.079026893	192.168.32.100	192.168.32.101	TCP	74	42016 → 700 [SYN] Seq=0 Win=0
1689	13.079159119	192.168.32.101	192.168.32.100	TCP	60	700 → 42016 [RST, ACK] Seq=1

Questa invece è la situazione che concerne una porta non aperta. Il client invia il protocollo, ma non essendoci disponibilità, il server risponde direttamente con un pacchetto avente il flag reset (RST).

## Scansione SYN sulle porte well-known

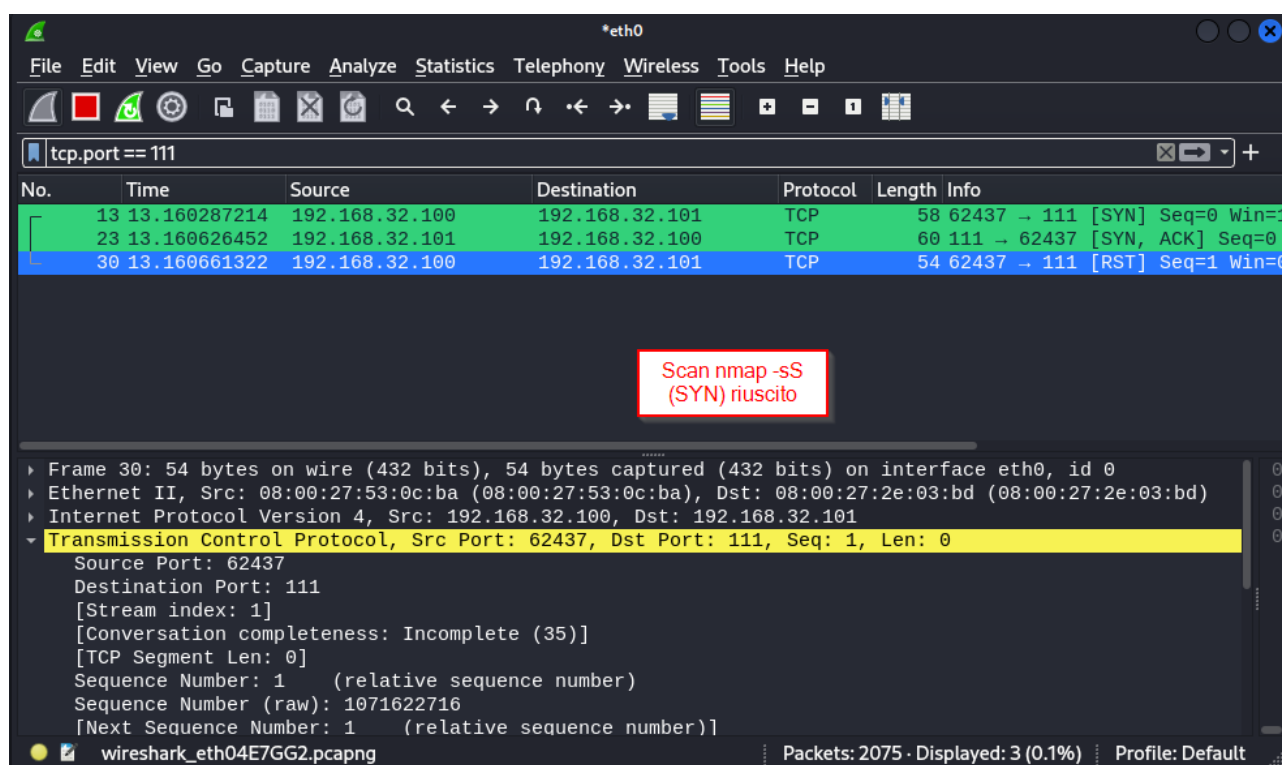
```
(kali@kali)-[~]
$ sudo nmap -sS -p0-1024 192.168.32.101

Starting Nmap 7.93 ( https://nmap.org ) at 2023-12-21 09:00 EST
Nmap scan report for 192.168.32.101
Host is up (0.00026s latency).
Not shown: 1013 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: 08:00:27:2E:03:BD (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.38 seconds
```

Scan -sS  
nmap

Per cambiare il tipo di scan ci basterà semplicemente inserire -sS invece che -sT. Lo scan di nmap con -sS è un SYN scan, una tipologia che interrompe la comunicazione del client con il server durante la fase di sincronizzazione, andando a inviare un pacchetto con il protocollo RTS. Osserviamo subito la differenza con Wireshark.



Come si può vedere dall'immagine, la terza fase di sincronizzazione viene saltata con un pacchetto finale RTS da parte del client. A differenza dello scan -st, lo scan -sS non genera il tipico overload causato da un canale di comunicazione ben stabilito.

## Scansione con switch «-A» sulle porte well-known

```
(kali@kali)-[~]
$ nmap -A -p0-1024 192.168.32.101
Starting Nmap 7.93 ( https://nmap.org ) at 2023-12-21 09:36 EST
Nmap scan report for 192.168.32.101
Host is up (0.00074s latency).
Not shown: 1013 closed tcp ports (conn-refused)
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 600fcfe1c05f6a74d69024fac4d56ccd (DSA)
|_   2048 5656240f211ddea72bae61b1243de8f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCED
STATUSCODES, 8BITMIME, DSN
|_ssl2:
|_   SSLv2 supported
|_   ciphers:
|_     SSL2_RC4_128_EXPORT40_WITH_MD5
|_     SSL2_RC2_128_CBC_WITH_MD5
|_     SSL2_RC4_128_WITH_MD5
|_     SSL2_DES_64_CBC_WITH_MD5
|_     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_     SSL2_DES_192_EDE3_CBC_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
```

Scan nmap -A dettagliato



```

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        42919/tcp  mountd
|   100005   1,2,3        60055/udp  mountd
|   100021   1,3,4        49211/udp  nlockmgr
|   100021   1,3,4        57706/tcp  nlockmgr
|   100024   1            38347/tcp  status
|   100024   1            56822/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 rexecd
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:
|_nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
|_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: 2023-12-21T09:37:41-05:00
|_smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
|_clock-skew: mean: 2h30m08s, deviation: 3h32m17s, median: 1s

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 97.65 seconds
--(kali@kali)-[~]

```

Scan nmap -A  
parte 2

Lo scan nmap con lo switch -A esegue un'analisi dettagliata del sistema e di tutte le porte attive specificate nel range del comando. Ad esempio, la porta 21 appartiene al servizio FTP, il quale è dotato di uno stato, un tipo, una versione etc. Più in basso invece troviamo il target Host, il nome del computer, il nome di dominio, il livello di autenticazione etc.

## Traccia 2

La scansione dei servizi di rete è il primo passo per capire quali servizi potrebbero essere vulnerabili, ed essere sfruttati successivamente per ottenere accesso alle macchine. E' molto importante in questa fase essere organizzati e strutturati. Dunque, per ognuno degli scan effettuati, lo

studente è invitato a riprodurre un report Excel / altro (tabella su word ad esempio) che riporti in maniera chiara:

- La fonte dello scan.
- Il target dello scan.
- Il tipo di scan.
- I risultati ottenuti (e.s. trovati 50 servizi attivi sulla macchina).

Scan Source	Scan Target	Scan Type	Result
-OS: Kali GNU/Linux(2023.4) -IPV4: 192.168.32.100 -Tool: nmap	-OS: Unix (Samba 3.0.20- Debian) -IPV4: 192.168.32.101	-Scansione TCP(-sT)	-12 porte trovate attive ed i relativi servizi associati, come FTP, SSH, telnet, SMTP, DNS e HTTP -1 host attivo
-OS: Kali GNU/Linux(2023.4) -IPV4: 192.168.32.100 -Tool: nmap	-OS: Unix(Samba 3.0.20-Debian) -IPV4: 192.168.32.101	-Scansione SYN(-sS)	-12 porte trovate attive ed i relativi servizi associati, come FTP, SSH, telnet, SMTP, DNS e HTTP -1 host attivo -Target MAC address: 08:00:27:2E:03:BD
-OS: Kali GNU/Linux(2023.4) -IPV4: 192.168.32.100 -Tool: nmap	-OS: Unix(Samba 3.0.20-Debian) -IPV4: 192.168.32.101	-Scansione - A(dettagliata)	-12 porte trovate attive ed i relativi servizi associati, come FTP, SSH, telnet, SMTP, DNS e HTTP -1 host attivo

