

# LABORATORIO 2 - CISCO CYBER OPS

## Wireshark per Osservare la Stretta di Mano TCP a 3 Vie.

### Obbiettivi:

1. Preparare gli host per catturare il traffico
2. Analizzare i pacchetti utilizzando Wireshark
3. Visualizzare i pacchetti utilizzando tcpdump

### Introduzione

In questo laboratorio, utilizzeremo Wireshark per catturare ed esaminare i pacchetti generati tra il browser del PC tramite HyperText Transfer Protocol (HTTP) e un server Web. Quando un'applicazione, come HTTP o File Transfer Protocol (FTP), viene avviata per la prima volta su un host, TCP utilizza l'handshake a tre vie per stabilire una sessione TCP affidabile tra i due host. Ad esempio, quando un PC utilizza un browser Web per navigare in Internet, viene avviata un'handshake a tre vie e viene stabilita una sessione tra l'host del PC e il server Web. Un PC può avere più sessioni TCP attive simultanee con vari siti Web.

### 1.Preparare gli host per catturare traffico

Una volta avviata la Virtual Machine andiamo ad **Mininet** con il comando “ `sudo lab.support.files/scripts/cyberops_topo.py` ” e successivamente avviamo gli host H1 e H4 con “ `xterm` ”, dove H4 verrà utilizzato come server web.

```
[analyst@secOps ~]$ sudo lab.support.files/scripts/cyberops_topo.py
[1] 586
[2] 587
bash: scripts: command not found
bash: cyberops_topo.py: command not found
[2]+  Exit 127      scripts
[analyst@secOps ~]$ sudo lab.support.files/scripts/cyberops_topo.py
[sudo] password for analyst:

CyberOPS Topology:

      -----
      | R1 |-----| H4 |
      -----
          |
          |
      -----| S1 |-----
      |         |         |
      |         |         |
      -----|         |
      | H1 |   | H2 |   | H3 |
      -----

*** Add links
*** Creating network
*** Adding hosts:
H1 H2 H3 H4 R1
*** Adding switches:
s1
*** Adding links:
(H1, s1) (H2, s1) (H3, s1) (H4, R1) (s1, R1)
*** Configuring hosts
H1 H2 H3 H4 R1
*** Starting controller

*** Starting 1 switches
s1 ...
*** Routing Table on Router:
Kernel IP routing table
Destination    Gateway      Genmask      Flags Metric Ref    Use Iface
10.0.0.0       0.0.0.0      0.0.0.0      U        0    0    0 R1-eth1
172.16.0.0     0.0.0.0      0.0.0.0      U        0    0    0 R1-eth2

*** Starting CLI:
mininet> xterm H1
mininet> xterm H4
mininet> 
```

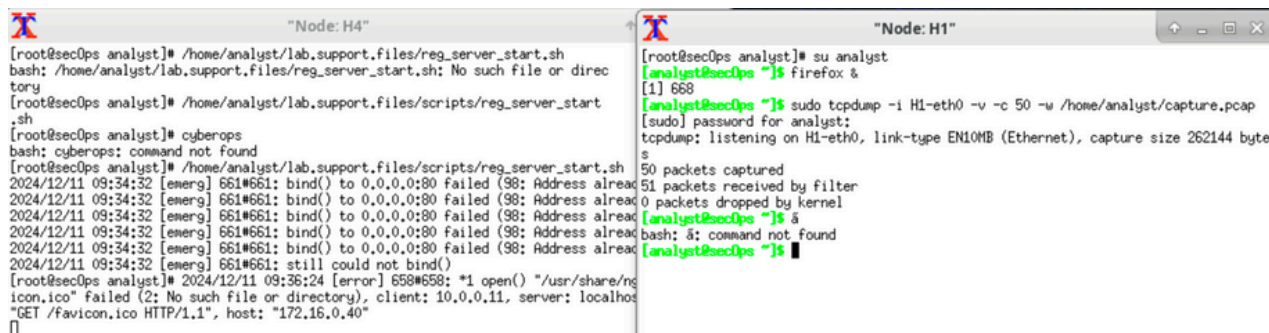
Per motivi di sicurezza, non è possibile eseguire Firefox dall'account utente root. Sull'host H1, utilizzare il comando `switch user` per passare dall'account utente root all'account utente analyst:

```
Node: H1
[analyst@secOps ~]$ sudo su analyst
[analyst@secOps ~]$ firefox &
[1] 1235
[analyst@secOps ~]$

Node: H4
[analyst@secOps ~]$ /home/analyst/lab.support.files/scripts/reg_server_start
[analyst@secOps ~]$
[analyst@secOps ~]$ switch user root
[analyst@secOps ~]$ cybersops
bash: cybersops: command not found
[analyst@secOps ~]$ /home/analyst/lab.support.files/scripts/reg_server_start.sh
2024/12/11 09:22:22 [emerg] 1230#1230: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:22:22 [emerg] 1230#1230: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:22:22 [emerg] 1230#1230: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:22:22 [emerg] 1230#1230: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:22:22 [emerg] 1230#1230: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:22:22 [emerg] 1230#1230: still could not bind()
[analyst@secOps ~]$ 
```

Dopo l'apertura della finestra di Firefox, avviata su H1, avviamo anche una sessione **tcpdump** nel terminale Node: H1 e invia l'output a un file chiamato capture.pcap . Con l'opzione -v, osserviamo l'avanzamento. Questa cattura si fermerà dopo aver catturato 50 pacchetti, poiché è configurata con l'opzione -c 50.

Dopo l'avvio accediamo al indirizzo 172.16.0.40 nel browser web Firefox.

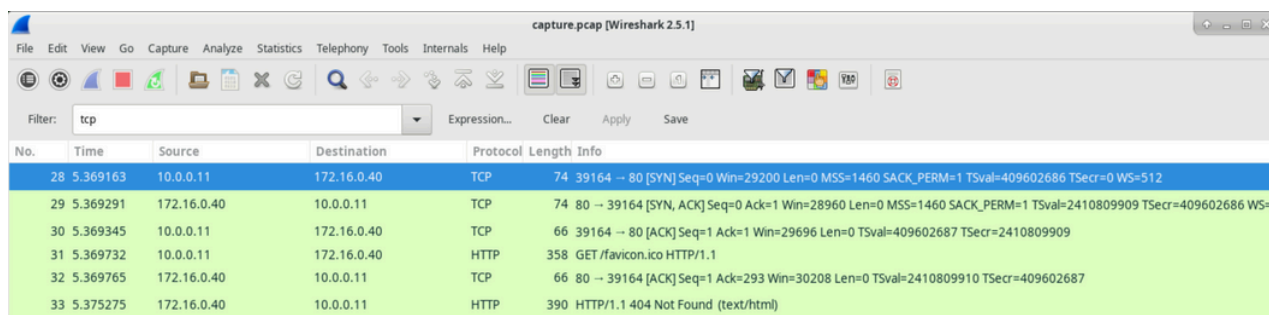


```
[root@secOps analyst]# /home/analyst/lab.support.files/reg_server_start.sh
bash: /home/analyst/lab.support.files/reg_server_start.sh: No such file or directory
[root@secOps analyst]# /home/analyst/lab.support.files/scripts/reg_server_start.sh
[root@secOps analyst]# cyberops
bash: cyberops: command not found
[root@secOps analyst]# /home/analyst/lab.support.files/scripts/reg_server_start.sh
2024/12/11 09:34:32 [emerg] 661#661: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:34:32 [emerg] 661#661: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:34:32 [emerg] 661#661: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:34:32 [emerg] 661#661: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:34:32 [emerg] 661#661: bind() to 0.0.0.0:80 failed (98: Address already in use)
2024/12/11 09:34:32 [emerg] 661#661: still could not bind()
[root@secOps analyst]# 2024/12/11 09:36:24 [error] 658#658: *1 open() "/usr/share/nginx/icon.ico" failed (2: No such file or directory), client: 10.0.0.11, server: localhost
GET /favicon.ico HTTP/1.1", host: "172.16.0.40"
```

```
[root@secOps analyst]# su analyst
[analyst@secOps ~]$ firefox &
[1] 668
[analyst@secOps ~]$ sudo tcpdump -i H1-eth0 -v -c 50 -w /home/analyst/capture.pcap
[sudo] password for analyst:
tcpdump: listening on H1-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
50 packets captured
61 packets received by filter
0 packets dropped by kernel
[analyst@secOps ~]$ &
bash: &: command not found
[analyst@secOps ~]$
```

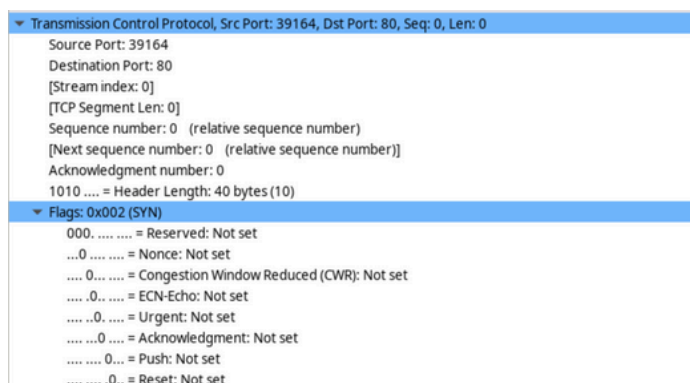
## 2. Analizzare i pacchetti utilizzando Wireshark

In questa fase andiamo ad analizzare i pacchetti avviando Wireshark e aprendo il file capture.pcap e applichiamo un filtro tcp alla cattura per analizzare più nel dettaglio ciò che ci interessa



No.	Time	Source	Destination	Protocol	Length	Info
28	5.369163	10.0.0.11	172.16.0.40	TCP	74	39164 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=409602686 TSecr=0 WS=512
29	5.369291	172.16.0.40	10.0.0.11	TCP	74	80 → 39164 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=2410809909 TSecr=409602686 WS=512
30	5.369345	10.0.0.11	172.16.0.40	TCP	66	39164 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=409602687 TSecr=2410809909
31	5.369732	10.0.0.11	172.16.0.40	HTTP	358	GET /favicon.ico HTTP/1.1
32	5.369765	172.16.0.40	10.0.0.11	TCP	66	80 → 39164 [ACK] Seq=1 Ack=293 Win=30208 Len=0 TSval=2410809910 TSecr=409602687
33	5.375275	172.16.0.40	10.0.0.11	HTTP	390	HTTP/1.1 404 Not Found (text/html)

Andiamo quindi ad esaminare le informazioni contenute nei pacchetti. il primo frame è l'inizio dell'handshake a tre vie tra il PC e il server su H4. Nel riquadro dell'elenco dei pacchetti, per esaminarlo andiamo ad esplorare la voce "Transmission Control Protocol" e "Flags"



```
▼ Transmission Control Protocol, Src Port: 39164, Dst Port: 80, Seq: 0, Len: 0
  Source Port: 39164
  Destination Port: 80
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 0 (relative sequence number)
  [Next sequence number: 0 (relative sequence number)]
  Acknowledgment number: 0
  1010 .... = Header Length: 40 bytes (10)
  ▼ Flags: 0x002 (SYN)
    000. .... = Reserved: Not set
    ...0 .... = Nonce: Not set
    ....0... = Congestion Window Reduced (CWR): Not set
    ....0... = ECN-Echo: Not set
    ....0... = Urgent: Not set
    ....0... = Acknowledgment: Not set
    ....0... = Push: Not set
    ....0... = Reset: Not set
```

In questo caso vediamo che la porta sorgente è la 39164 e quella di destinazione è la porta 80, quindi una porta nota (protocollo HTTP), inoltre possiamo vedere che è impostata la Flag SYN

Successivamente andiamo a selezionare il secondo frame, che sarebbe il pacchetto successivo nell'handshake a tre vie. Questo è il server web che risponde alla richiesta iniziale di avviare una sessione.

```
Transmission Control Protocol, Src Port: 80, Dst Port: 39164, Seq: 0, Ack: 1, Len: 0
Source Port: 80
Destination Port: 39164
[Stream index: 0]
[TCP Segment Len: 0]
Sequence number: 0 (relative sequence number)
[Next sequence number: 0 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
1010 ... = Header Length: 40 bytes (10)
Flags: 0x012 (SYN, ACK)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
...0 .... = Congestion Window Reduced (CWR): Not set
...0... = ECN-Echo: Not set
...0... = Urgent: Not set
...1... = Acknowledgment: Set
...0... = Push: Not set
...0... = Reset: Not set
```

Adesso la porta di origine è la 80 e quella di destinazione è la porta 39164, a differenza del primo frame qui il numero di sequenza relativo è 0 e il numero di conferma relativo è 1.

Infine, selezioniamo il terzo frame che rappresenta l'ultimo pacchetto nell'handshake a tre vie.

```
Transmission Control Protocol, Src Port: 39164, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
Source Port: 39164
Destination Port: 80
[Stream index: 0]
[TCP Segment Len: 0]
Sequence number: 1 (relative sequence number)
[Next sequence number: 1 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
1000 ... = Header Length: 32 bytes (8)
Flags: 0x010 (ACK)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
...0 .... = Congestion Window Reduced (CWR): Not set
...0... = ECN-Echo: Not set
...0... = Urgent: Not set
...1... = Acknowledgment: Set
...0... = Push: Not set
...0... = Reset: Not set
...0... = Syn: Not set
...0... = Fin: Not set
[TCP Flags: A-]
Window size value: 58
[Calculated window size: 29696]
[Window size scaling factor: 512]
Checksum: 0xb669 (unverified)
[Checksum Status: Unverified]
Urgent pointer: 0
```

Come nel primo frame la porta di origine è la 39164 e quella di destinazione è la porta 80. Ed è impostata la Flag di conferma ACK. I numeri di sequenza e di conferma relativi vengono impostati su 1 come punto di partenza. La connessione TCP

viene stabilita e la comunicazione tra il computer sorgente e il server Web può iniziare.

### 3. Visualizzare i pacchetti utilizzando tcpdump

È anche possibile visualizzare il file pcap e filtrare le informazioni desiderate, aprendo una finestra del terminale e digitando “man tcpdump”. Utilizzando le pagine del manuale disponibili con il sistema operativo Linux, è possibile leggere o cercare al loro interno le opzioni per selezionare le informazioni desiderate dal file pcap. Per esempio cercheremo le informazioni con lo switch -r che consente di leggere il pacchetto dal file salvato, utilizzando l'opzione -w con tcpdump o altri strumenti che scrivono file pcap o pcap-ng, come Wireshark.

Quindi andiamo ad utilizzare il comando sottostante per visualizzare i primi 3 pacchetti TCP acquisiti:

```
[analyst@ecolpe ~]$ tcpdump -r /home/analyst/capture.pcap tcp -c 3
reading from file /home/analyst/capture.pcap, link-type EN10MB (Ethernet)
09:36:24.376127 IP 10.0.0.11.39164 > 172.16.0.40.http: Flags [S], seq 4207376318, win 29200, options [max 1460,sackOK,TS val 409602686 ecr 0,nop,wscale 9], length 0
09:36:24.376255 IP 172.16.0.40.http > 10.0.0.11.39164: Flags [S.], seq 1535498450, ack 4207376319, win 28940, options [max 1460,sackOK,TS val 2410809909 ecr 409602686,nop,wscale 9], length 0
09:36:24.376309 IP 10.0.0.11.39164 > 172.16.0.40.http: Flags [A.], ack 1, win 58, options [nop,nop,TS val 409602687 ecr 2410809909], length 0
[analyst@ecolpe ~]$
```