

ESERCIZIO W11L2

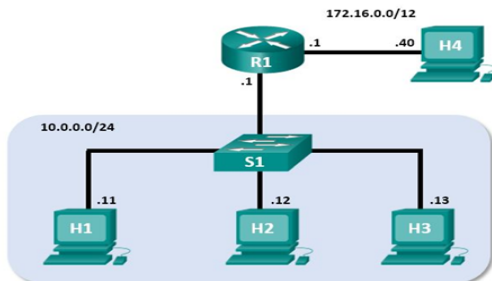
Pratica S11/L2 PDF

rk per Osservare l'Handshake a 3 Vie TCP



Usare Wireshark per Osservare l'Handshake a 3 Vie TCP

Topologia Mininet



Risorse Richieste: Macchina virtuale CyberOps Workstation

Obiettivi

- Parte 1: Preparare gli Host per Catturare il Traffico
- Parte 2: Analizzare i Pacchetti usando Wireshark
- Parte 3: Visualizzare i Pacchetti usando tcpdump

Contesto / Scenario

In questo laboratorio, userai Wireshark per catturare ed esaminare i pacchetti generati tra il browser del PC che utilizza il protocollo HTTP (HyperText Transfer Protocol) e un server web, come www.google.com. Quando un'applicazione, come HTTP o FTP (File Transfer Protocol), si avvia 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 avviato 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 simultaneamente con vari siti web.

3

Pratica S11/L2 PDF

rk per Osservare l'Handshake a 3 Vie TCP



ISTRUZIONI

Parte 1: Preparare gli Host per Catturare il Traffico

- Avviare la VM CyberOps. Accedere con nome utente analyst e password cyberops.
- Avviare Mininet.

```
[analyst@secOps ~]$ sudo lab.support.files/scripts/cyberops_topo.py
```

- Avviare gli host H1 e H4 in Mininet.

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

- Avviare il server web su H4.

```
[root@secOps analyst]# /home/analyst/lab.support.files/scripts/reg_server_start.sh
```

4

Pratica S11/L2 PDF

Usare Wireshark per Osservare l'Handshake a 3 Vie TCP



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

```
[root@secOps analyst]# su analyst
```

- Avviare il browser web su H1. Ci vorrà qualche momento.

```
[analyst@secOps ~]$ firefox &
```

- Dopo l'apertura della finestra di Firefox, avviare una sessione `tcpdump` nel terminale Node: H1 e inviare l'output a un file chiamato `capture.pcap`. Con l'opzione `-v`, è possibile osservare l'avanzamento. Questa cattura si fermerà dopo aver catturato 50 pacchetti, poiché è configurata con l'opzione `-c 50`.

```
[analyst@secOps ~]$ sudo tcpdump -i H1-eth0 -v -c 50 -w /home/analyst/capture.pcap
```

- Dopo l'avvio di `tcpdump`, navigare rapidamente a `172.16.0.40` nel browser web Firefox.

5

Parte 2: Analizzare i Pacchetti usando Wireshark

Passo 1: Applicare un filtro alla cattura salvata.

a. Premere INVIO per vedere il prompt. Avviare Wireshark su Node: H1. Fare clic su OK quando viene richiesto l'avviso riguardante l'esecuzione di Wireshark come superutente.

```
[analyst@secOps ~]$ wireshark-gtk &
```

b. In Wireshark, fare clic su File > Open. Selezionare il file pcap salvato situato in /home/analyst/capture.pcap.

c. Applicare un filtro tcp alla cattura. In questo esempio, i primi 3 frame rappresentano il traffico di interesse.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.11	172.16.0.40	TCP	74	58716 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
2	0.000081	172.16.0.40	10.0.0.11	TCP	74	80 → 58716 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460
3	0.000082	10.0.0.11	172.16.0.40	TCP	66	58716 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=38645
4	0.000194	10.0.0.11	172.16.0.40	HTTP	356	GET /favicon.ico HTTP/1.1

6

Passo 2: Esaminare le informazioni all'interno dei pacchetti, inclusi indirizzi IP, numeri di porta TCP e flag di controllo TCP.

a. In questo esempio, il frame 1 è l'inizio dell'handshake a tre vie tra il PC e il server su H4. Nel riquadro dell'elenco dei pacchetti (sezione superiore della finestra principale), selezionare il primo pacchetto, se necessario.

b. Fare clic sulla freccia a sinistra del Transmission Control Protocol nel riquadro dei dettagli del pacchetto per espanderlo ed esaminare le informazioni TCP. Localizzare le informazioni sulla porta di origine e destinazione.

c. Fare clic sulla freccia a sinistra dei Flags. Un valore di 1 significa che il flag è impostato. Localizzare il flag impostato in questo pacchetto.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.11	172.16.0.40	TCP	74	58716 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
2	0.000081	172.16.0.40	10.0.0.11	TCP	74	80 → 58716 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460
3	0.000082	10.0.0.11	172.16.0.40	TCP	66	58716 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=38645
4	0.000194	10.0.0.11	172.16.0.40	HTTP	356	GET /favicon.ico HTTP/1.1

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0 Ethernet II, Src: a6:a1:15:2c:d8:de (a6:a1:15:2c:d8:de), Dst: a2:86:17:7c:c3:65 (a2:86:17:7c:c3:65) Internet Protocol Version 4, Src: 10.0.0.11, Dst: 172.16.0.40 Transmission Control Protocol, Src Port: 58716, Dst Port: 80, Seq: 0, Len: 0	Source Port: 58716 Destination Port: 80 [Stream index: 0] [TCP Segment Len: 0] Sequence number: 0 (relative sequence number) Acknowledgment number: 0 Header Length: 40 bytes Flags: 0x002 (SYN) Window size value: 29200 [calculated window size: 29200] Checksum: 0xb671 [unverified] [Checksum Status: Unverified] Urgent pointer: 0 Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
--	--

Nota: Potrebbe essere necessario regolare le dimensioni delle finestre superiore e centrale all'interno di Wireshark per visualizzare le informazioni necessarie.

7

- Qual è il numero di porta TCP di origine?
- Come classificherei la porta di origine?
- Qual è il numero di porta TCP di destinazione?
- Come classificherei la porta di destinazione?
- Quale flag è impostato?
- A quale valore è impostato il numero di sequenza relativo?

d. Selezionare il pacchetto successivo nell'handshake a tre vie. In questo esempio, è il frame 2. Questa è la risposta del server web alla richiesta iniziale di avviare una sessione.

- Quali sono i valori delle porte di origine e destinazione?
- Quali flag sono impostati?
- A quali valori sono impostati i numeri relativi di sequenza e acknowledgment?

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.11	172.16.0.40	TCP	74	58716 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
2	0.000081	172.16.0.40	10.0.0.11	TCP	74	80 → 58716 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460
3	0.000082	10.0.0.11	172.16.0.40	TCP	66	58716 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=38645
4	0.000194	10.0.0.11	172.16.0.40	HTTP	356	GET /favicon.ico HTTP/1.1

Frame 2: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface eth0 Ethernet II, Src: a2:86:17:7c:c3:65 (a2:86:17:7c:c3:65), Dst: a6:a1:15:2c:d8:de (a6:a1:15:2c:d8:de) Internet Protocol Version 4, Src: 172.16.0.40, Dst: 10.0.0.11 Transmission Control Protocol, Src Port: 80, Dst Port: 58716, Seq: 0, Ack: 1, Len: 0	Source Port: 80 Destination Port: 58716 [Stream index: 0] [TCP Segment Len: 0] Sequence number: 0 (relative sequence number) Acknowledgment number: 1 (relative ack number) Header Length: 40 bytes Flags: 0x012 (SYN, ACK) Window size value: 28960 [calculated window size: 28960] Checksum: 0xc85a [unverified] [Checksum Status: Unverified] Urgent pointer: 0
--	--

8



e. Infine, selezionare il terzo pacchetto nell'handshake a tre vie.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.11	172.16.0.40	TCP	74	58716 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERFECT
2	0.000081	172.16.0.40	10.0.0.11	TCP	74	80 → 58716 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460
3	0.000082	10.0.0.11	172.16.0.40	TCP	66	58716 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=3864
4	0.000194	10.0.0.11	172.16.0.40	HTTP	356	GET /favicon.ico HTTP/1.1

Frame 3: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
 Ethernet II, Src: a6:a1:15:2c:d8:de (a6:a1:15:2c:d8:de), Dst: a2:86:17:7c:c3:65 (a2:86:17:7c:c3:65)
 Internet Protocol Version 4, Src: 10.0.0.11, Dst: 172.16.0.40
 Transmission Control Protocol, Src Port: 58716, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
 Source Port: 58716
 Destination Port: 80
 [Stream index: 0]
 [TCP Segment Len: 0]
 Sequence number: 1 (relative sequence number)
 Acknowledgment number: 1 (relative ack number)
 Header Length: 32 bytes
 Flags: 0x010 (ACK)
 Window size value: 58
 [Calculated window size: 29696]
 [Window size scaling factor: 512]
 Checksum: 0xb669 [unverified]
 [Checksum Status: Unverified]
 Urgent pointer: 0

Esaminare il terzo e ultimo pacchetto dell'handshake.

Quale flag è impostato?

I numeri relativi di sequenza e acknowledgment sono impostati a 1 come punto di partenza. La connessione TCP è stabilita e la comunicazione tra il computer di origine e il server web può iniziare.

9



Parte 3: Visualizzare i pacchetti usando tcpdump

È anche possibile visualizzare il file pcap e filtrare per le informazioni desiderate.

a. Aprire una nuova finestra di terminale, inserire `man tcpdump`. Nota: Potrebbe essere necessario premere INVIO per vedere il prompt.

Utilizzando le pagine manuale (`man` pages) disponibili con il sistema operativo Linux, è possibile leggere o cercare tra le pagine manuale le opzioni per selezionare le informazioni desiderate dal file pcap.

```
[analyst@secOps ~]$ man tcpdump
TCPDUMP(1)          General Commands Manual          TCPDUMP(1)

NAME
  tcpdump - dump traffic on a network

SYNOPSIS
  tcpdump [ -AbdDefhHlJKlLnOpqStuUvXx# ] [ -B buffer_size ]
           [ -c count ]
           [ -C file_size ] [ -G rotate_seconds ] [ -F file ]
           [ -i interface ] [ -j timestamp_type ] [ -m module ] [ -M secret ]
           [ --number ] [ -Q in/out/inout ]
           [ -r file ] [ -V file ] [ -s snaplen ] [ -T type ] [ -w file ]
           [ -W filecount ]
           [ -E spi@ipaddr algo:secret,... ]
           [ -y datalinktype ] [ -z postrotate-command ] [ -Z user ]
           [ --time-stamp-precision=timestamp_precision ]
           [ --immediate-mode ] [ --version ]
           [ expression ]

<output omissio>
```

Per cercare nelle pagine man, è possibile usare / (ricerca in avanti) o ? (ricerca indietro) per trovare termini specifici, n per passare alla corrispondenza successiva e q per uscire. Ad esempio, per cercare informazioni sull'opzione `-r`, digitare `/-r`. Digitare n per passare alla corrispondenza successiva.

10

Cosa fa l'opzione -r?



b. Nello stesso terminale, aprire il file di cattura usando il seguente comando per visualizzare i primi 3 pacchetti TCP catturati:

```
[analyst@secOps ~]$ tcpdump -r /home/analyst/capture.pcap tcp -c 3
reading from file capture.pcap, link-type EN10MB (Ethernet)
13:58:30.647462 IP 10.0.0.11.58716 > 172.16.0.40.http: Flags [S], seq 2432755549, win 29200, options [mss 1460,sackOK,TS val 3864513189,ecn 0,nop,wscale 9], length 0
13:58:30.647543 IP 172.16.0.40.http > 10.0.0.11.58716: Flags [S.], seq 1766419191, ack 2432755550, win 28960, options [mss 1460,sackOK,TS val 58557410,ecn 3864513189,nop,wscale 9], length 0
13:58:30.647544 IP 10.0.0.11.58716 > 172.16.0.40.http: Flags [.], ack 1, win 58, options [nop,nop,TS val 3864513189,ecn 58557410], length 0
```

Per visualizzare l'handshake a 3 vie, potrebbe essere necessario aumentare il numero di righe dopo l'opzione `-c`.

c. Navigare al terminale usato per avviare Mininet. Terminare Mininet inserendo `quit` nella finestra principale del terminale della VM CyberOps.

```
mininet> quit
*** Stopping 0 controllers

*** Stopping 2 terms
*** Stopping 5 links
....
*** Stopping 1 switches
s1
*** Stopping 5 hosts
H1 H2 H3 H4 R1
*** Done
[analyst@secOps ~]$
```

d. Dopo aver chiuso Mininet, inserire `sudo mn -c` per pulire i processi avviati da Mininet. Inserire la password `cyberops` quando richiesto.

```
[analyst@secOps ~]$ sudo mn -c
[sudo] password for analyst:
```

Domande di Riflessione

1. Ci sono centinaia di filtri disponibili in Wireshark. Una rete di grandi dimensioni potrebbe avere numerosi filtri e molti tipi diversi di traffico. Elenca tre filtri che potrebbero essere utili a un amministratore di rete.

2. In quali altri modi Wireshark potrebbe essere utilizzato in una rete di produzione?

11

Qual è il numero di porta TCP di origine? 55568

Wireshark packet capture showing a SYN packet from source port 55568 to destination port 80. The packet list shows a sequence of packets including SYN, ACK, and HTTP GET requests. The packet details pane shows the structure of the TCP segment with source port 55568 and destination port 80.

No.	Time	Source	Destination	Protocol	Length	Info
39	26.126134	10.0.0.11	172.16.0.40	TCP	74	55568 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=773428504
40	26.126173	172.16.0.40	10.0.0.11	TCP	74	80 → 55568 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=
41	26.126180	10.0.0.11	172.16.0.40	TCP	66	55568 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=773428504 TSecr=247961445
42	26.126335	10.0.0.11	172.16.0.40	HTTP	377	GET / HTTP/1.1
43	26.126344	172.16.0.40	10.0.0.11	TCP	66	80 → 55568 [ACK] Seq=1 Ack=312 Win=30208 Len=0 TSval=2479614455 TSecr=773428
44	26.133512	172.16.0.40	10.0.0.11	TCP	304	80 → 55568 [PSH, ACK] Seq=1 Ack=312 Win=30208 Len=238 TSval=2479614462 TSecr=
45	26.133519	10.0.0.11	172.16.0.40	TCP	66	55568 → 80 [ACK] Seq=312 Ack=239 Win=30720 Len=0 TSval=773428512 TSecr=24796
46	26.134348	172.16.0.40	10.0.0.11	HTTP	678	HTTP/1.1 200 OK (text/html)
47	26.134351	10.0.0.11	172.16.0.40	TCP	66	55568 → 80 [ACK] Seq=312 Ack=851 Win=31744 Len=0 TSval=773428513 TSecr=24796
50	26.349311	10.0.0.11	172.16.0.40	HTTP	358	GET /favicon.ico HTTP/1.1

Frame 39: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)

Ethernet II, Src: c6:e6:4b:1c:eb:e6 (c6:e6:4b:1c:eb:e6), Dst: 7a:15:55:8a:ca:c6 (7a:15:55:8a:ca:c6)

Internet Protocol Version 4, Src: 10.0.0.11, Dst: 172.16.0.40

Transmission Control Protocol, Src Port: 55568, Dst Port: 80, Seq: 0, Len: 0

Source Port: 55568
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

0000 7a 15 55 8a ca c6 e6 4b 1c eb e6 08 00 45 00 72 U.....K.....E
0010 00 3c b5 85 40 00 40 06 ce f3 0a 00 00 0b ac 10 <...@.....
0020 00 28 d9 10 00 50 b6 20 0e 92 00 00 00 0a 02 1f (...P.....
0030 72 10 b6 71 00 00 02 04 05 b4 04 02 08 0a 2e 19 f.q.....

Come classifichereesti la porta di origine?

- Le porte **effimere(49152-65535)** vengono usate dai client quando avviano una connessione verso un server (es. navigazione web, SSH, ecc.).
- **Non sono legate a servizi specifici**, quindi per sapere con certezza cosa sta usando la porta 55568, bisogna **analizzare il traffico di rete** o **verificare i processi in esecuzione**.

Qual è il numero di porta TCP di destinazione? 80

Come classifichereesti la porta di destinazione?

- Porta well known
- La porta **80/TCP** è usata dai server web per gestire richieste HTTP.
- È la porta predefinita per la **navigazione web senza cifratura** (HTTP), a differenza della porta **443**, che è usata per HTTPS (HTTP su TLS/SSL).
- Quando digiti un indirizzo come <http://example.com>, il browser comunica con il server sulla porta 80.

Quale flag è impostato? 0x002 (SYN)

▶ Ethernet II, Src: c6:e6:4b:1c:eb:e6 (c6:e6:4b:1c:eb:e6), Dst: 7a:15:55:8a:ca:c6 (7a:15:55:8a:ca:c6)
▶ Internet Protocol Version 4, Src: 10.0.0.11, Dst: 172.16.0.40
▼ Transmission Control Protocol, Src Port: 55568, Dst Port: 80, Seq: 0, Len: 0
Source Port: 55568
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)
Window size value: 29200
[Calculated window size: 29200]
Checksum: 0xb671 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
▶ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
▶ [Timestamps]

A quale valore è impostato il numero di sequenza relativo? 0

Quali sono i valori delle porte di origine e destinazione?

80 origine, 55568 destinazione

▶ Frame 40: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▶ Ethernet II, Src: 7a:15:55:8a:ca:c6 (7a:15:55:8a:ca:c6), Dst: c6:e6:4b:1c:eb:e6 (c6:e6:4b:1c:eb:e6)
▶ Internet Protocol Version 4, Src: 172.16.0.40, Dst: 10.0.0.11
▼ Transmission Control Protocol, Src Port: 80, Dst Port: 55568, Seq: 0, Ack: 1, Len: 0
Source Port: 80
Destination Port: 55568
[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)
Window size value: 28960
[Calculated window size: 28960]
Checksum: 0xb671 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
▶ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
▶ [RFC/ACK analysis]

Quali flag sono impostati? 0x012 (SYN ACK)

A quali valori sono impostati i numeri relativi di sequenza e acknowledgment?

0 1

Quale flag è impostato (terzo pacchetto)? 0x010 (ACK)

▶ Frame 41: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)

▶ Ethernet II, Src: c6:e6:4b:1c:eb:e6 (c6:e6:4b:1c:eb:e6), Dst: 7a:15:55:8a:ca:c6 (7a:15:55:8a:ca:c6)

▶ Internet Protocol Version 4, Src: 10.0.0.11, Dst: 172.16.0.40

▼ Transmission Control Protocol, Src Port: 55568, Dst Port: 80, Seq: 1, Ack: 1, Len: 0

Source Port: 55568

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)

Window size value: 58

[Calculated window size: 29696]

[Window size scaling factor: 512]

Checksum: 0xb669 [unverified]

[Checksum Status: Unverified]

Urgent pointer: 0

▶ Options: (12 bytes) No-Operation (NOP), No-Operation (NOP), Timestamps

0000 7a 15 55 8a ca c6 c6 e6 4b 1c eb e6 08 00 45 00 z.U.....K.....E.
0010 00 34 b5 86 40 00 40 06 ce fa 0a 00 00 0b ac 10 .4..@.@.

Cosa fa l'opzione -r?

```

[root@secOps analyst]# su analyst
[analyst@secOps ~]$ ^C
[analyst@secOps ~]$ ^C
[analyst@secOps ~]$ tcpdump -r /home/analyst/capture.pcap
reading from file /home/analyst/capture.pcap, link-type EN10MB (Ethernet)
13:16:41.415563 IP 10.0.0.11.45954 > 8.8.4.4.domain: 15478+ A? detectportal.firefox.com. (42)
13:16:41.415609 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 78
13:16:41.415622 IP 10.0.0.11.45954 > 8.8.4.4.domain: 6276+ AAAA? detectportal.firefox.com. (42)
13:16:41.415629 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 78
13:16:46.416505 IP 10.0.0.11.54768 > 209.165.200.235.domain: 15478+ A? detectportal.firefox.com. (42)
13:16:46.416559 IP _gateway > 10.0.0.11: ICMP net 209.165.200.235 unreachable, length 78
13:16:46.416577 IP 10.0.0.11.54768 > 209.165.200.235.domain: 6276+ AAAA? detectportal.firefox.com. (42)
13:16:46.416585 IP _gateway > 10.0.0.11: ICMP net 209.165.200.235 unreachable, length 78
13:16:46.774683 ARP, Request who-has _gateway tell 10.0.0.11, length 28
13:16:46.775302 ARP, Reply _gateway is-at 7a:15:55:8a:ca:c6 (oui Unknown), length 28
13:16:51.422183 IP 10.0.0.11.32963 > 8.8.4.4.domain: 36663+ A? detectportal.firefox.com. (42)
13:16:51.422217 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 78
13:16:51.422227 IP 10.0.0.11.32963 > 8.8.4.4.domain: 15170+ AAAA? detectportal.firefox.com. (42)
13:16:51.422232 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 78
13:16:52.406777 IP6 secOps > ff02::2: ICMP6, router solicitation, length 16
13:16:56.427535 IP 10.0.0.11.56040 > 209.165.200.235.domain: 36663+ A? detectportal.firefox.com. (42)
13:16:56.427592 IP _gateway > 10.0.0.11: ICMP net 209.165.200.235 unreachable, length 78
13:16:56.427609 IP 10.0.0.11.56040 > 209.165.200.235.domain: 15170+ AAAA? detectportal.firefox.com. (42)
13:16:56.427618 IP _gateway > 10.0.0.11: ICMP net 209.165.200.235 unreachable, length 78
13:17:00.192061 IP 10.0.0.11.52511 > 8.8.4.4.domain: 28091+ A? blocklists.settings.services.mozilla.com. (58)
13:17:00.192089 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 94
13:17:00.192097 IP 10.0.0.11.52511 > 8.8.4.4.domain: 55240+ AAAA? blocklists.settings.services.mozilla.com. (58)
13:17:00.192102 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 94
13:17:00.230540 IP 10.0.0.11.56272 > 8.8.4.4.domain: 19050+ A? firefox.settings.services.mozilla.com. (55)
13:17:00.230554 IP 10.0.0.11.56272 > 8.8.4.4.domain: 7279+ AAAA? firefox.settings.services.mozilla.com. (55)
13:17:01.432578 IP 10.0.0.11.32963 > 8.8.4.4.domain: 36663+ A? detectportal.firefox.com. (42)
13:17:01.432664 IP _gateway > 10.0.0.11: ICMP net 8.8.4.4 unreachable, length 78
13:17:01.432688 IP 10.0.0.11.32963 > 8.8.4.4.domain: 15170+ AAAA? detectportal.firefox.com. (42)
13:17:05.197417 IP 10.0.0.11.45300 > 209.165.200.235.domain: 28091+ A? blocklists.settings.services.mozilla.com. (58)
13:17:05.197459 IP _gateway > 10.0.0.11: ICMP net 209.165.200.235 unreachable, length 94
13:17:05.197470 IP 10.0.0.11.45300 > 209.165.200.235.domain: 55240+ AAAA? blocklists.settings.services.mozilla.com. (58)

```

Legge i pacchetti da un **file pcap** (anziché dalla rete in tempo reale).

- Quando usi **-r**, **tcpdump non ascolta la rete**, ma apre un file .pcap già esistente (registrato ad esempio con tcpdump -w o con Wireshark/tshark).
- Viene spesso usata per analizzare pacchetti dopo averli catturati.

1. Ci sono centinaia di filtri disponibili in Wireshark. Una rete di grandi dimensioni potrebbe avere numerosi filtri e molti tipi diversi di traffico. Elenca tre filtri che potrebbero essere utili a un amministratore di rete.

- **Filtro per un protocollo specifico**
Per visualizzare solo il traffico HTTP: http
Utile per: monitorare il traffico web in chiaro, identificare richieste sospette, troubleshooting di applicazioni web.
- **Filtro per indirizzo IP specifico**
Per visualizzare solo il traffico da o verso un IP (esempio: 192.168.1.10):
ip.addr == 192.168.1.10
Utile per: isolare il traffico di un singolo host (es. un server, un PC compromesso, un dispositivo IoT).
- **Filtro per una porta specifica**
Per monitorare il traffico sulla porta HTTPS (443):

tcp.port == 443

Utile per: verificare se il traffico cifrato funziona correttamente, controllare il volume di traffico TLS/SSL.

2. In quali altri modi Wireshark potrebbe essere utilizzato in una rete di produzione?

Wireshark, in una **rete di produzione**, è uno strumento potentissimo — non solo per "sniffare pacchetti", ma anche per diagnosticare, ottimizzare e proteggere la rete.

Ecco alcuni **usi concreti** che un amministratore di rete o un analista può fare in produzione:

Risoluzione dei problemi (troubleshooting)

- Diagnosi di problemi di connettività (perché un client non riesce a raggiungere un server?).
- Identificazione di latenze elevate (ping elevati, handshake lenti).
- Analisi di problemi DNS (DNS lento o risposte errate).
- Verifica di problemi di handshake TLS/SSL.

Analisi delle prestazioni della rete

- Misurazione del throughput effettivo tra client e server.
- Rilevamento di collo di bottiglia in alcune tratte della rete.
- Controllo di retransmissioni TCP (segno di problemi fisici o saturazione della banda).

Monitoraggio della sicurezza

- Identificazione di traffico sospetto (esfiltrazione dati, scansioni di rete).
- Cattura e analisi di tentativi di attacco (esempio: exploit HTTP, attacchi ARP spoofing, anomalie DNS).
- Analisi forense post-incident response (cosa ha fatto un malware? quali connessioni ha aperto?).
- Controllo della compliance (è presente traffico non cifrato su servizi che dovrebbero essere cifrati?).

Documentazione e analisi storica

- Generazione di report dettagliati sul traffico di rete.
- Cattura di sessioni specifiche per audit o per verificare il rispetto delle policy aziendali.

Supporto allo sviluppo

- Analisi e debug di applicazioni in sviluppo che comunicano sulla rete.

- Verifica della correttezza dei protocolli implementati.
- Misurazione dei tempi di risposta applicativi.