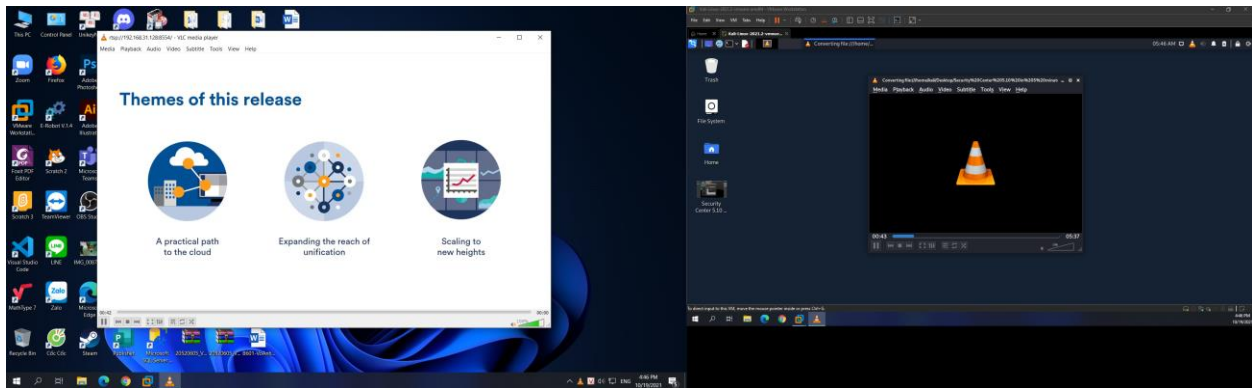


# Bài thực hành số 3

## Ảnh phần 1



Video được stream từ máy có IP là: 192.168.31.128:8554

1. Chọn một gói tin UDP, xác định các trường (field) có trong UDP header và giải thích ý nghĩa của mỗi trường đó? Gợi ý: Xem tại phần User Datagram Protocol

```
▼ User Datagram Protocol, Src Port: 53051, Dst Port: 60738
  Source Port: 53051
  Destination Port: 60738
  Length: 53
  Checksum: 0x2c8d [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
  UDP payload (45 bytes)
  > Real-Time Transport Protocol
```

Xét gói tin số 36

Source Port: Port nguồn

Destination Port: Port đích

Length: Độ dài gói tin

Checksum: Giá trị kiểm tra

2. Qua thông tin hiển thị của Wireshark, xác định độ dài (tính theo byte) của mỗi trường trong UDP header?

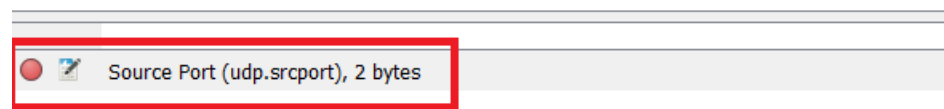
Source Port: 2 bytes

Destination Port: 2 bytes

Length: 2 bytes

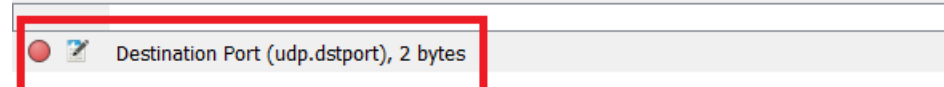
Checksum: 2 bytes

```
▼ User Datagram Protocol, Src Port: 53051, Dst Port: 60738
  Source Port: 53051
  Destination Port: 60738
  Length: 53
  Checksum: 0x2c8d [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
  UDP payload (45 bytes)
> Real-Time Transport Protocol
```



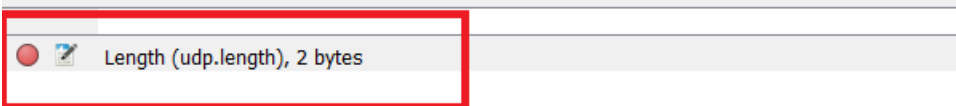
Source Port (udp.srcport), 2 bytes

```
▼ User Datagram Protocol, Src Port: 53051, Dst Port: 60738
  Source Port: 53051
  Destination Port: 60738
  Length: 53
  Checksum: 0x2c8d [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
  UDP payload (45 bytes)
> Real-Time Transport Protocol
```

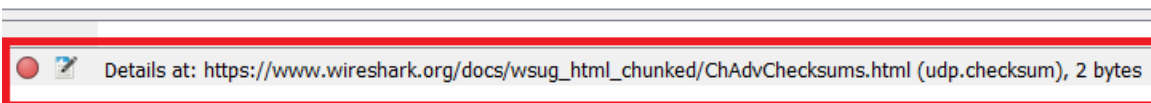


Destination Port (udp.dstport), 2 bytes

- ▼ User Datagram Protocol, Src Port: 53051, Dst Port: 60738
  - Source Port: 53051
  - Destination Port: 60738
  - Length: 53
  - Checksum: 0x2c8d [unverified]
  - [Checksum Status: Unverified]
  - [Stream index: 3]
  - > [Timestamps]
  - UDP payload (45 bytes)
- > Real-Time Transport Protocol



- ▼ User Datagram Protocol, Src Port: 53051, Dst Port: 60738
  - Source Port: 53051
  - Destination Port: 60738
  - Length: 53
  - Checksum: 0x2c8d [unverified]
  - [Checksum Status: Unverified]
  - [Stream index: 3]
  - > [Timestamps]
  - UDP payload (45 bytes)
- > Real-Time Transport Protocol



3. Giá trị của trường Length trong UDP header là độ dài của gì? Chứng minh nhận định này?

Là độ dài header + độ dài data

Chứng minh

```
> Frame 26: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface \Device\NPF_{0416F410-8671-4E38-BAFF-0D0AAA0B4F0B}, id 0
> Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: VMware_ec:a4:18 (00:0c:29:ec:a4:18)
> Internet Protocol Version 4, Src: 192.168.31.1, Dst: 192.168.31.128
v User Datagram Protocol, Src Port: 60738, Dst Port: 53051
  Source Port: 60738
  Destination Port: 53051
  Length: 12
  Checksum: 0xc68b [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
    UDP payload (4 bytes)
> Real-Time Transport Protocol
> Data (4 bytes)
```

Length (udp.length), 2 bytes | Packets: 1383 · 0

## Xét gói số 26

Length: 12 = 2 bytes (Header TCP) + 4 bytes (Data)

4. Số bytes lớn nhất mà payload (phần chứa dữ liệu gốc, không tính UDP header và IP header) của UDP có thể chứa?

Là  $(2^{16}) - 1$  (do giá trị trong 16 bit) – 8 bytes header.

=>  $65535 - 8 = 65527$  bytes

5. Giá trị lớn nhất có thể có của port nguồn (Source port)?

=>  $(2^{16}) - 1 = 65535$  bytes

6. \* Tìm và kiểm tra một cặp gói tin sử dụng giao thức UDP gồm: gói tin do máy mình gửi và gói tin phản hồi của gói tin đó. Miêu tả mối quan hệ về port number của 2 gói tin này. Gợi ý: Có thể bắt gói tin UDP ở một tình huống khác để tìm được 1 cặp gói tin như trên.

Trong quá trình gửi yêu cầu, IP nguồn gửi request packet sẽ trở thành destination và source port sẽ trở thành destination port còn IP của người gửi response sẽ trở thành IP source.

Lấy ví dụ gói tin 22 và 36

Request packet:

caulbai3.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

1 udp

No.	Time	Source	Destination	Protocol	Length	Info
10.000000	192.168.31.1	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1	
21.010657	192.168.31.1	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1	
32.015413	192.168.31.1	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1	
43.017273	192.168.31.1	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1	
22.12.320005	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
23.12.320055	192.168.31.1	192.168.31.128	RTCP	46	60737 → 53937 Len=4	
24.12.320068	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
25.12.320075	192.168.31.1	192.168.31.128	RTCP	46	60737 → 53937 Len=4	
26.12.320110	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
27.12.320147	192.168.31.1	192.168.31.128	RTCP	46	60739 → 53052 Len=4	
28.12.320158	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
29.12.320165	192.168.31.1	192.168.31.128	RTCP	46	60739 → 53052 Len=4	
31.12.320557	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
32.12.320583	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
33.12.320604	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
34.12.320622	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
36.12.326734	192.168.31.128	192.168.31.1	RTP	87	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4658, Time=724428421, Mark	
37.12.332078	192.168.31.128	192.168.31.1	RTP	86	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4659, Time=724434421, Mark	

> Frame 22: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface \Device\NPF\_{0416F410-8671-4E38-BAFF-000AAA0B4F0B}, id 0

> Ethernet II, Src: VMware\_c0:00:08 (00:50:56:c0:00:08), Dst: VMware\_ec:a4:18 (00:0c:29:ec:a4:18)

> Internet Protocol Version 4, Src: 192.168.31.1, Dst: 192.168.31.128

> User Datagram Protocol, Src Port: 60736, Dst Port: 53936

Source Port: 60736 source port destination port

Destination Port: 53936

Length: 12

Checksum: 0xc318 [unverified]

[Checksum Status: Unverified]

[Stream index: 1]

> [Timestamps]

UDP payload (4 bytes)

> Real-Time Transport Protocol

> Data (4 bytes)

Length (udp.length), 2 bytes

Packets: 1383 · Displayed: 1348 (97.5%)

Profile: Default

## Response packet:

caulbai3.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

1 udp

No.	Time	Source	Destination	Protocol	Length	Info
23.12.320055	192.168.31.1	192.168.31.128	RTCP	46	60737 → 53937 Len=4	
24.12.320068	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
25.12.320075	192.168.31.1	192.168.31.128	RTCP	46	60737 → 53937 Len=4	
26.12.320110	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
27.12.320147	192.168.31.1	192.168.31.128	RTCP	46	60739 → 53052 Len=4	
28.12.320158	192.168.31.1	192.168.31.128	RTP	46	Unknown RTP version 3	
29.12.320165	192.168.31.1	192.168.31.128	RTCP	46	60739 → 53052 Len=4	
31.12.320557	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
32.12.320583	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
33.12.320604	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
34.12.320622	192.168.31.128	192.168.31.1	ICMP	74	Destination unreachable (Port unreachable)	
36.12.326734	192.168.31.128	192.168.31.1	RTP	87	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4658, Time=724428421, Mark	
37.12.332078	192.168.31.128	192.168.31.1	RTP	86	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4659, Time=724434421, Mark	
38.12.332479	192.168.31.128	192.168.31.1	RTP	122	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4660, Time=724449421, Mark	
39.12.333209	192.168.31.128	192.168.31.1	RTP	96	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4661, Time=724443421, Mark	
40.12.333699	192.168.31.128	192.168.31.1	RTP	86	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4662, Time=724440421, Mark	
42.12.346221	192.168.31.128	192.168.31.1	RTP	475	PT=MPEG-I/II Audio, SSRC=0x7CA67099, Seq=56629, Time=724470992, Mark	
43.12.347881	192.168.31.128	192.168.31.1	RTP	86	PT=DynamicRTP-Type-96, SSRC=0x41A2EBBF, Seq=4663, Time=724446421, Mark	

> Frame 36: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface \Device\NPF\_{0416F410-8671-4E38-BAFF-000AAA0B4F0B}, id 0

> Ethernet II, Src: VMware\_ec:a4:18 (00:0c:29:ec:a4:18), Dst: VMware\_c0:00:08 (00:50:56:c0:00:08)

> Internet Protocol Version 4, Src: 192.168.31.128, Dst: 192.168.31.1

> User Datagram Protocol, Src Port: 53051, Dst Port: 60738

Source Port: 53051 source port destination port

Destination Port: 60738

Length: 53

Checksum: 0x2c8d [unverified]

[Checksum Status: Unverified]

[Stream index: 3]

> [Timestamps]

UDP payload (45 bytes)

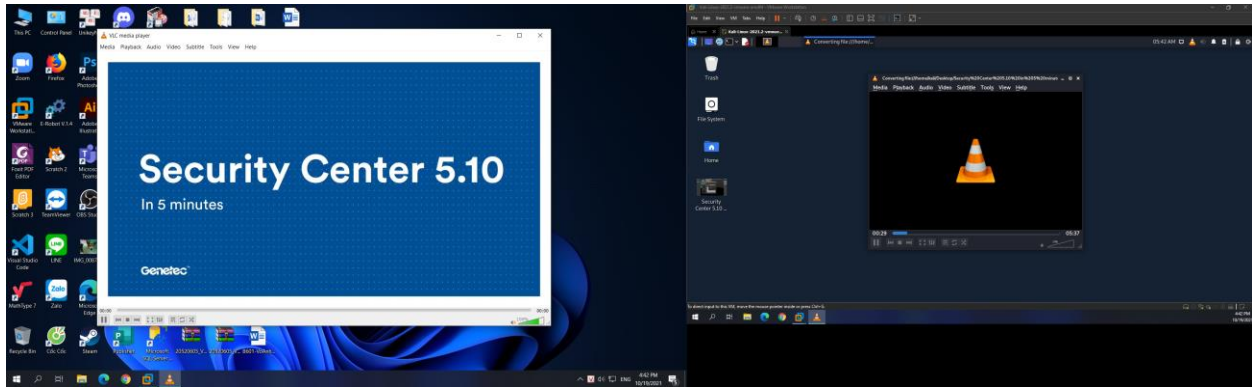
> Real-Time Transport Protocol

Length (udp.length), 2 bytes

Packets: 1383 · Displayed: 1348 (97.5%)

Profile: Default

## Ảnh phần 2



### 7. Tìm địa chỉ IP và TCP port của máy Client?

Địa chỉ IP của client: 192.168.31.1

Port của client: 55958

Wireshark packet capture analysis showing a TCP connection from 192.168.31.1 to 192.168.31.128. The packet list shows a GET request on port 8080. The packet details pane highlights the TCP segment with source port 55958 and destination port 8080. The packet bytes pane shows the raw data of the GET request.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.31.1	192.168.31.128	TCP	66	55958 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
2	0.001668	192.168.31.128	192.168.31.1	TCP	66	8080 → 55958 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
3	0.003162	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080 [ACK] Seq=1 Ack=1 Win=131328 Len=0
4	0.003223	192.168.31.1	192.168.31.128	HTTP	191	GET / HTTP/1.1
5	0.003742	192.168.31.128	192.168.31.1	TCP	60	8080 → 55958 [ACK] Seq=1 Ack=138 Win=64128 Len=0
6	0.025117	192.168.31.128	192.168.31.1	TCP	157	8080 → 55958 [PSH, ACK] Seq=1 Ack=138 Win=64128 Len=103 [TCP segment of a reassembled PDU]
7	0.067665	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080 [ACK] Seq=138 Ack=104 Win=131072 Len=0
8	0.068039	192.168.31.128	192.168.31.1	TCP	452	8080 → 55958 [PSH, ACK] Seq=104 Ack=138 Win=64128 Len=398 [TCP segment of a reassembled PDU]
9	0.110630	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080 [ACK] Seq=138 Ack=502 Win=130816 Len=0
10	0.7950646	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=502 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
11	0.7950663	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=1962 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
12	0.7950829	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=3422 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
13	0.7950844	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=4882 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
14	0.7950872	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080 [ACK] Seq=138 Ack=3422 Win=131328 Len=0
15	0.7950899	192.168.31.128	192.168.31.1	TCP	54	55958 → 8080 [ACK] Seq=138 Ack=6342 Win=131328 Len=0
16	0.7950909	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=6342 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
17	0.7950918	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=7802 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
18	0.7951004	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=9262 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
19	0.7951012	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=10722 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
20	0.7951048	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=12182 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
21	0.7951054	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=13642 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
22	0.7951111	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=15102 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
23	0.7951121	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [PSH, ACK] Seq=16562 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]
24	0.7951163	192.168.31.128	192.168.31.1	TCP	1514	8080 → 55958 [ACK] Seq=18022 Ack=138 Win=64128 Len=1460 [TCP segment of a reassembled PDU]

> Frame 7: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF{0416F410-8671-4E38-BAFF-000AA0B4F0B}, id 0  
> Ethernet II, Src: VMware\_c0:00:08 (00:50:56:c0:00:08), Dst: VMware\_ec:a4:18 (00:0c:29:ec:a4:18)  
> Internet Protocol Version 4, Src: 192.168.31.1, Dst: 192.168.31.128  
> Transmission Control Protocol, Src Port: 55958, Dst Port: 8080, Seq: 138, Ack: 104, Len: 0

0000 00 0c 29 ec a4 18 00 50 56 c0 00 08 00 00 45 00 ...P V...E...  
0010 00 28 1b a8 40 00 00 06 1f 56 c0 a8 1f 01 c0 a8 ...@...V...  
0020 1f 80 da 96 1f 90 ff e8 e6 de ad 64 99 4f 50 10 ...d OP...  
0030 02 00 c6 5f 00 00 ...

### 8. Tìm địa chỉ IP của Server? Kết nối TCP dùng để gửi và nhận các segments sử dụng port nào?

Địa chỉ IP: 192.168.31.128

Port: 8080

No.	Time	Source	Destination	Protocol	Length	Info
10.000000	192.168.31.1	192.168.31.128	TCP	66	55958 → 8080	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
20.001668	192.168.31.128	192.168.31.1	TCP	66	8080 → 55958	[SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
30.003162	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080	[ACK] Seq=1 Ack=1 Win=131328 Len=0
40.003223	192.168.31.1	192.168.31.128	HTTP	191	GET / HTTP/1.1	
50.003742	192.168.31.128	192.168.31.1	TCP	60	8080 → 55958	[ACK] Seq=1 Ack=138 Win=64128 Len=0
60.025117	192.168.31.128	192.168.31.1	TCP	157	8080 → 55958	[PSH, ACK] Seq=1 Ack=138 Win=64128 Len=103 [TCP segment of a reassembled PDU]
70.067665	192.168.31.1	192.168.31.128	TCP	54	55958 → 8080	[ACK] Seq=138 Ack=104 Win=131072 Len=0
80.068039	192.168.31.128	192.168.31.1	TCP	452	8080 → 55958	[PSH, ACK] Seq=104 Ack=138 Win=64128 Len=398 [TCP segment of a reassembled PDU]

> Frame 8: 452 bytes on wire (3616 bits), 452 bytes captured (3616 bits) on interface \Device\NPF\_{0416F410-8671-4E38-BAFF-000AA0B4F0B}, id 0  
 > Ethernet II, Src: VMware\_eca:4:18 (00:0c:29:ec:a4:18), Dst: VMware\_c0:00:08 (00:50:56:c0:00:08)  
 > Internet Protocol Version 4, Src: 192.168.31.128, Dst: 192.168.31.1  
 > Transmission Control Protocol, Src Port: 8080, Dst Port: 55958, Seq: 104, Ack: 138, Len: 398

9. TCP SYN segment (gói tin TCP có cờ SYN) sử dụng sequence number nào để khởi tạo kết nối TCP giữa client và server? Thành phần nào trong segment cho ta biết segment đó là TCP SYN segment?

TCP có cờ SYN được sử dụng sequence number 0 để khởi tạo kết nối.

Như hình thấy được Flags cờ SYN được set bằng 1=> là TCP segment

No.	Time	Source	Destination	Protocol	Length	Info
10.000000	192.168.31.1	192.168.31.128	TCP	66	55958 → 8080	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
20.001668	192.168.31.128	192.168.31.1	TCP	66	8080 → 55958	[SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128

Header checksum: 0x1f4d (validation disabled)  
 [Header checksum status: Unverified]  
 Source Address: 192.168.31.1  
 Destination Address: 192.168.31.128  
 Transmission Control Protocol, Src Port: 55958, Dst Port: 8080, Seq: 0, Len: 0  
 Source Port: 55958  
 Destination Port: 8080  
 [Stream index: 0]  
 [TCP Segment Len: 0]  
 Sequence Number: 0 (relative sequence number)  
 Sequence Number (raw): 4293453396  
 [Next Sequence Number: 1 (relative sequence number)]  
 Acknowledgment Number: 0  
 Acknowledgment number (raw): 0  
 1000 .... = Header Length: 32 bytes (8)  
 Flags: 0x002 (SYN)  
 0000 .... = Reserved: Not set  
 .... 0000 = Nonce: Not set  
 .... 0... = Congestion Window Reduced (CWR): Not set  
 .... 0... = ECH: Echo: Not set  
 .... 0... = Urgent: Not set  
 .... 0... = Acknowledgment: Not set  
 .... 0... = Push: Not set  
 .... 0... = Reset: Not set  
 .... 0... = SYN: Set  
 .... 0... = FIN: Not set  
 [TCP Flags: .....S.]  
 Window: 64240  
 [calculated window size: 64240]



10. Tìm sequence number của gói tin SYN/ACK segment được gửi bởi server đến client để trả lời cho SYN segment?

Tìm giá trị của Acknowledgement trong SYN/ACK segment?

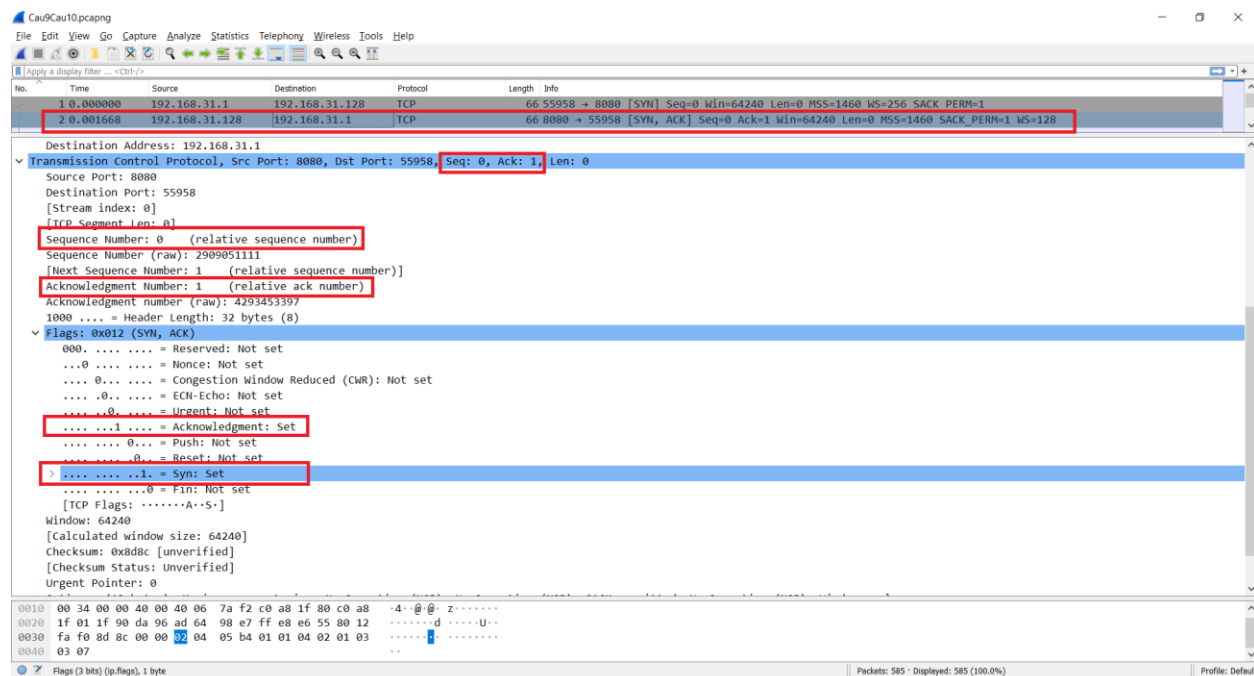
Làm sao server có thể xác định giá trị đó? Thành phần nào trong segment cho ta biết segment đó là SYN/ACK segment?

Thông tin trong ảnh cung cấp:

Sequence number: 0

Acknowledgement: 1

Giá trị của Acknowledgement trong gói SYN/ACK được xác định bởi server. Server khởi tạo sequence number đầu tiên (ISN) SYN segment từ client, là 0. => giá trị của Acknowledgement trong gói SYN/ACK là 1. Một segment sẽ là một SYN/ACK segment nếu có cả cờ SYN và cờ ACK đều set là 1.



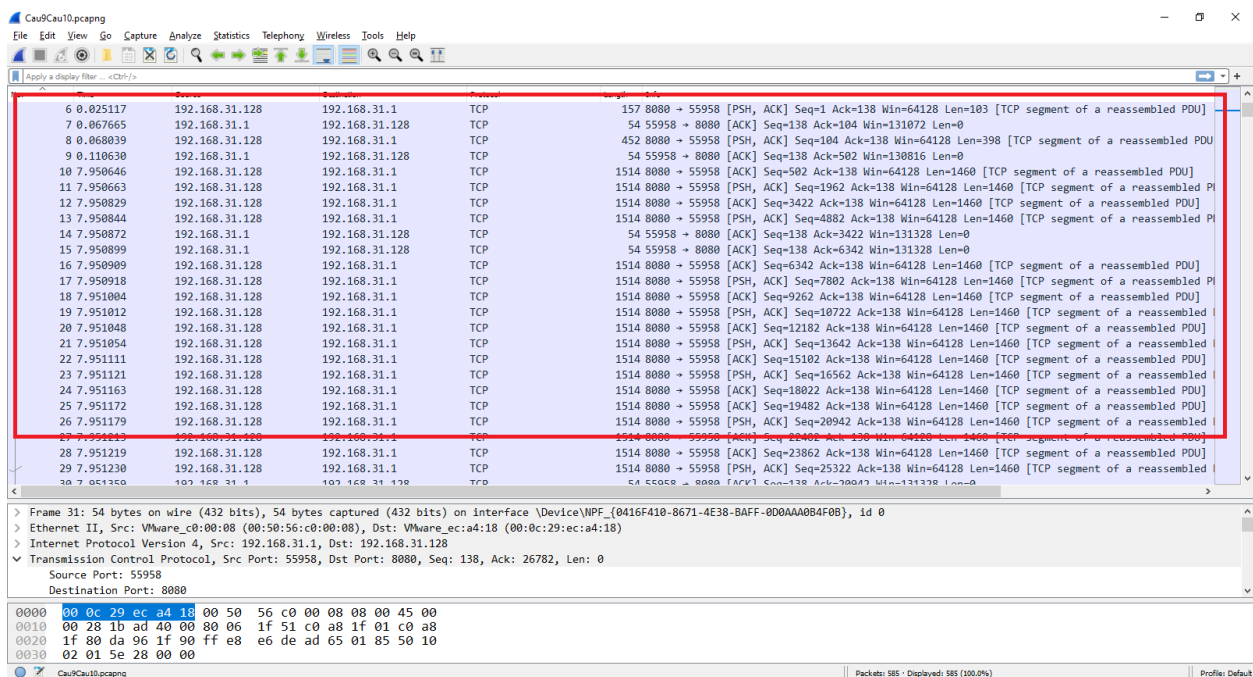
11. Chỉ ra 6 segment đầu tiên mà server gửi cho Client (dựa vào Số thứ tự gói – No)

- Tìm sequence number của 6 segments đầu tiên đó?
- Xác định thời gian mà mỗi segment được gửi, thời gian ACK cho mỗi segment được nhận?



- Đưa ra sự khác nhau giữa thời gian mà mỗi segment được gửi và thời gian ACK cho mỗi segment được nhận bằng cách tính RTT (Round Trip Time) cho 6 segments này?

STT gói tin	Thời gian gửi	Thời gian nhận ack	RTT
6	0.025117	0.067665	0.042548
8	0.068039	0.11063	0.042591
10	7.950646	7.950872	0.000226
12	7.950829	7.950899	0.00007
16	7.950909	7.951359	0.00045
26	7.951179	7.951386	0.000207



Do có một số lỗi nên có thể xảy ra sai số

12. Có segment nào được gửi lại hay không? Thông tin nào trong quá trình truyền tin cho chúng ta biết điều đó?

Không có segment nào được gửi lại vì dựa trên hình của wireshark ta thấy không có 1 packet nào bị trùng số sequence number ở trong một thời gian khác nhau.

