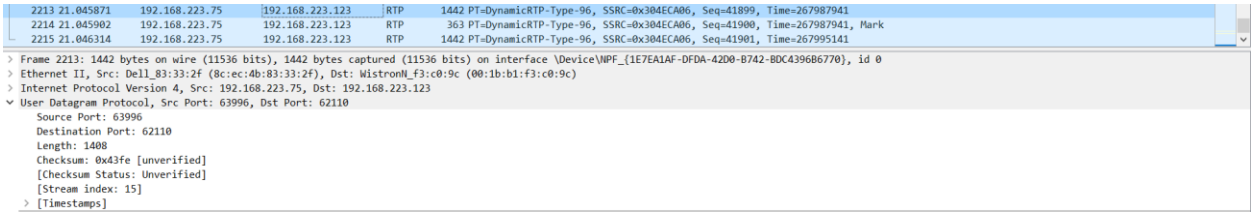
Họ Tên:Bùi Đức Anh

MSSV:19521190

BÀI THỰC HÀNH SỐ 3

Đây là bắt VLC media của bạn Ngô Đức Trí

Câu 1:



Có 4 trường UDP trong header:

-Source Port: Xác định cổng của người gửi thông tin và có ý nghĩa nếu muốn nhận thông tin từ người nhận(Cổng của Server)

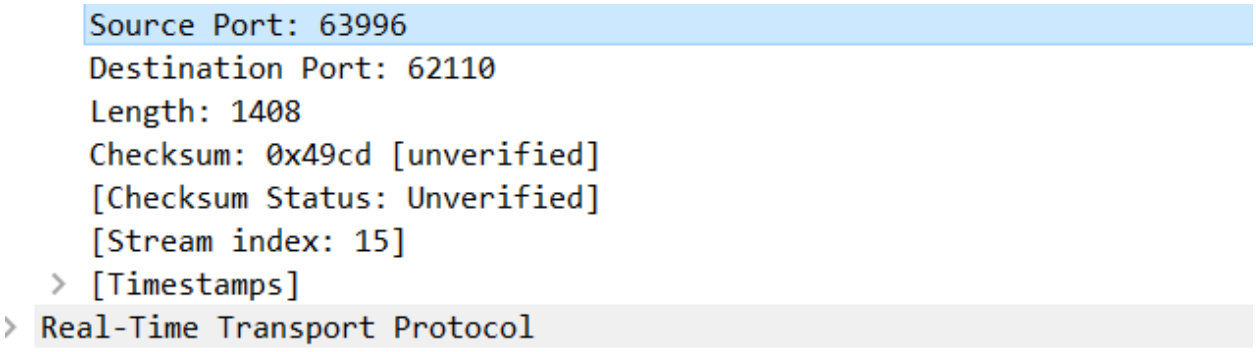
-Destination Port: Xác định cổng nhận thông tin(Cổng nhận thông tin)

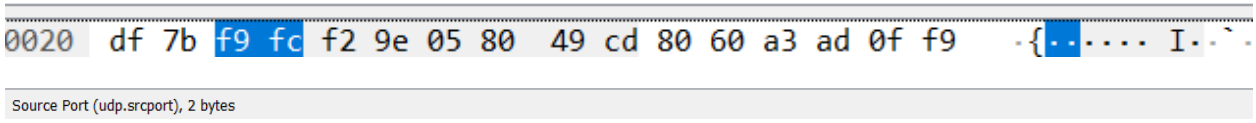
-Lenght: Xác định chiều dài của toàn bộ Datagram: phần Header và Dữ Liệu(Độ dài gói tin)

-Checksum: Dùng để kiểm tra lỗi của phần Header và Dữ liệu

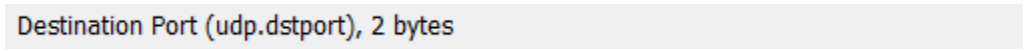
Câu 2:

Độ dài (tính theo byte) của mỗi trường trong UDP header

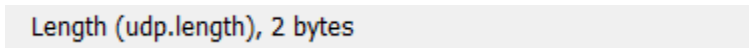




-Source Port: 2byte



- Destination Port:2 byte



- Lenght:2byte

-Checksum:2byte

Câu 3:

. Giá trị của trường Length trong UDP header là độ dài của Header+Data

```
User Datagram Protocol, Src Port: 63996, Dst Port: 62110
    Source Port: 63996
    Destination Port: 62110
    Length: 1408
    Checksum: 0x43fe [unverified]
    [Checksum Status: Unverified]
    [Stream index: 15]
  > [Timestamps]
> Real-Time Transport Protocol
```

```
0020  df 7b f9 fc f2 9e 05 80  43 fe 80 60 a3 ab 0f f9   ·{······ C··`····
0030  2b e5 30 4e ca 06 1c 81  9e c3 74 46 7f d6 aa c5   +·0N···· ··tF····
0040  35 78 3a 63 e4 92 4f 46  fd 6d 44 de b9 50 87 5d   5x:c··OF ·mD··P·]
0050  8a 77 18 74 00 0f b4 31  5c ac 2d e6 83 e7 4b f8   ·w·t···1 \·-···K·
0060  26 54 40 b7 4a 0a 6a 9d  b9 3e 72 96 e3 e8 ab 5b   &T@·J·j· ·>r····[
0070  3a 55 f9 e3 dd 78 7b 0e  a1 d0 33 97 e5 07 41 32   :U···x{· ··3···A2
0080  e0 56 01 85 d6 fc e1 8c  8a 5b 6a 0e d6 f7 5c 76   ·V····· ·[j···\v
0090  c1 64 c0 a0 9a 0d 24 d2  ff 3f 51 85 a5 26 fa 90   ·d····$· ·?Q··&··
00a0  29 b3 61 fe 40 57 95 00  d0 14 8c cb 9f 3b ca 91   )·a·@W·· ·····;··
00b0  c0 b5 fa 7e dd c6 ed 4d  05 e9 42 3a 42 65 67 87   ···~···M ··B:Beg·
00c0  57 7f 1f 62 21 76 a2 37  80 d4 1d 80 e3 60 b8 1f   W··b!v·7 ·····`··
00d0  47 21 e2 b0 cb 2d 08 8f  a8 d4 d1 6e 68 4d d9 1f   G!···-·· ···nhM··
00e0  7a 10 c4 68 2e f5 a7 80  23 89 64 05 fe 8f 64 6c   z··h.··· #·d···dl
```

User Datagram Protocol (udp), 8 bytes

```
> Frame 2213: 1442 bytes on wire (11536 bits), 1442 bytes captured (11536 bits) on interface \Device\NPF_{1E7EA1AF-DFDA-42D0-B742-BDC4396B6770}, id 0
> Ethernet II, Src: Dell_83:33:2f (8c:ec:4b:83:33:2f), Dst: WistronN_f3:c0:9c (00:1b:b1:f3:c0:9c)
> Internet Protocol Version 4, Src: 192.168.223.75, Dst: 192.168.223.123
v User Datagram Protocol, Src Port: 63996, Dst Port: 62110
    Source Port: 63996
    Destination Port: 62110
    Length: 1408
    Checksum: 0x43fe [unverified]
    [Checksum Status: Unverified]
    [Stream index: 15]
  > [Timestamps]
> Real-Time Transport Protocol
```

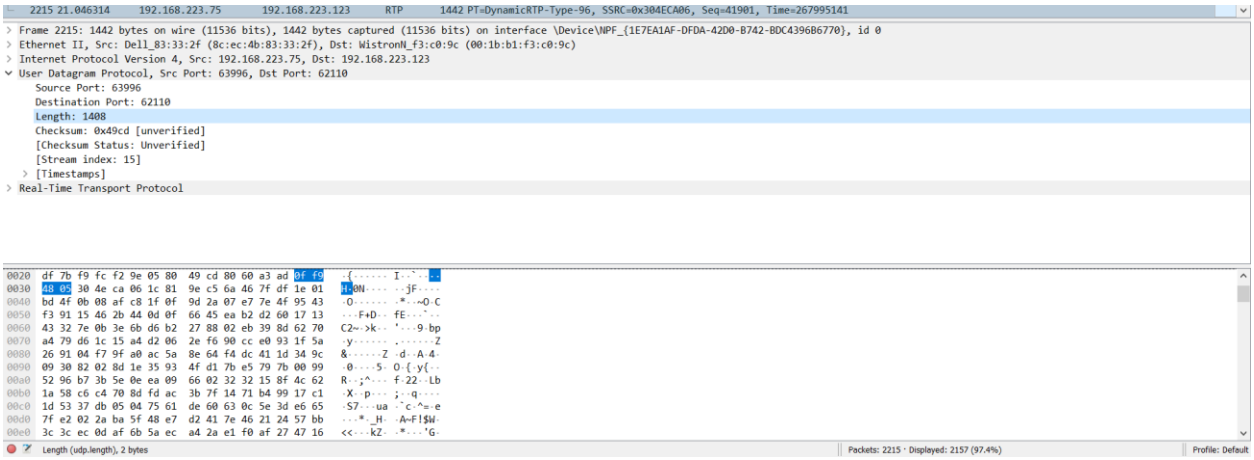
```
0020  df 7b f9 fc f2 9e 05 80  43 fe 80 60 a3 ab 0f f9   ·{······ C··`····
0030  2b e5 30 4e ca 06 1c 81  9e c3 74 46 7f d6 aa c5   +·0N···· ··tF····
0040  35 78 3a 63 e4 92 4f 46  fd 6d 44 de b9 50 87 5d   5x:c··OF ·mD··P·]
0050  8a 77 18 74 00 0f b4 31  5c ac 2d e6 83 e7 4b f8   ·w·t···1 \·-···K·
0060  26 54 40 b7 4a 0a 6a 9d  b9 3e 72 96 e3 e8 ab 5b   &T@·J·j· ·>r····[
0070  3a 55 f9 e3 dd 78 7b 0e  a1 d0 33 97 e5 07 41 32   :U···x{· ··3···A2
0080  e0 56 01 85 d6 fc e1 8c  8a 5b 6a 0e d6 f7 5c 76   ·V····· ·[j···\v
0090  c1 64 c0 a0 9a 0d 24 d2  ff 3f 51 85 a5 26 fa 90   ·d····$· ·?Q··&··
00a0  29 b3 61 fe 40 57 95 00  d0 14 8c cb 9f 3b ca 91   )·a·@W·· ·····;··
00b0  c0 b5 fa 7e dd c6 ed 4d  05 e9 42 3a 42 65 67 87   ···~···M ··B:Beg·
00c0  57 7f 1f 62 21 76 a2 37  80 d4 1d 80 e3 60 b8 1f   W··b!v·7 ·····`··
00d0  47 21 e2 b0 cb 2d 08 8f  a8 d4 d1 6e 68 4d d9 1f   G!···-·· ···nhM··
00e0  7a 10 c4 68 2e f5 a7 80  23 89 64 05 fe 8f 64 6c   z··h.··· #·d···dl
```

Real-Time Transport Protocol (rtp), 1,400 bytes | Packets: 2215 · Displayed: 2157 (97.4%)

Giá trị của trườngg Lenght trong UDP Header là 1408,là độ dài của gói tin nhận được bao gồm cả độ dài Header và payload từ hình dưới ta thấy được có Data chiếm 1400 bytes và 8 bytes phần còn lại là của Header

Câu 4:

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



Vì Lenght biểu diễn được với kích thươc là 2 bytes mà 1 byte=8bit => 2 byte = 16bit => Kích thước tối đa là 2^16 -1 =65535

Và ta không tính mỗi Lenght gồm cả UDP Header và IP Header nên số Payload lớn nhất có thể chứa sẽ là 65535 – 8 =65527(bytes)

Câu 5:

Vì Lenght biểu diễn được với kích thươc là 2 bytes mà 1 byte=8bit => 2 byte = 16bit => Kích thước tối đa là 2^16 -1 =65535

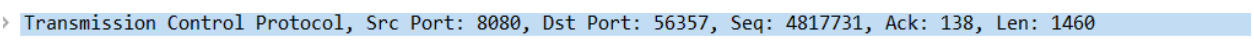
Và ta không tính mỗi Lenght gồm cả UDP Header và IP Header nên số Payload lớn nhất có thể chứa sẽ là 65535 – 8 =65527(bytes)

Giá trị lớn nhất có thể có của port nguồn (Source port) là 65535 . Port nguồn (Source port) bắt được là 63996

Câu 6:

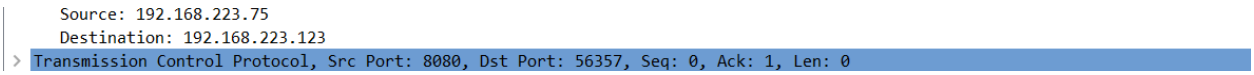
Câu 7:

IP Client(Destination) :192.168.223.123



TCP Port Client(Dst Port): 56357

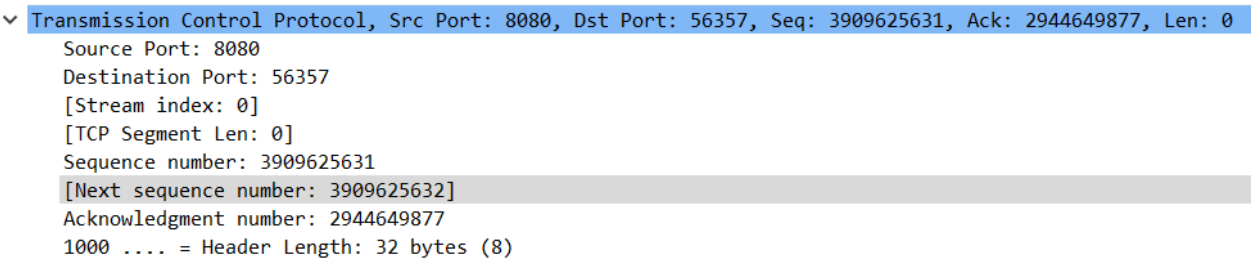
Câu 8:



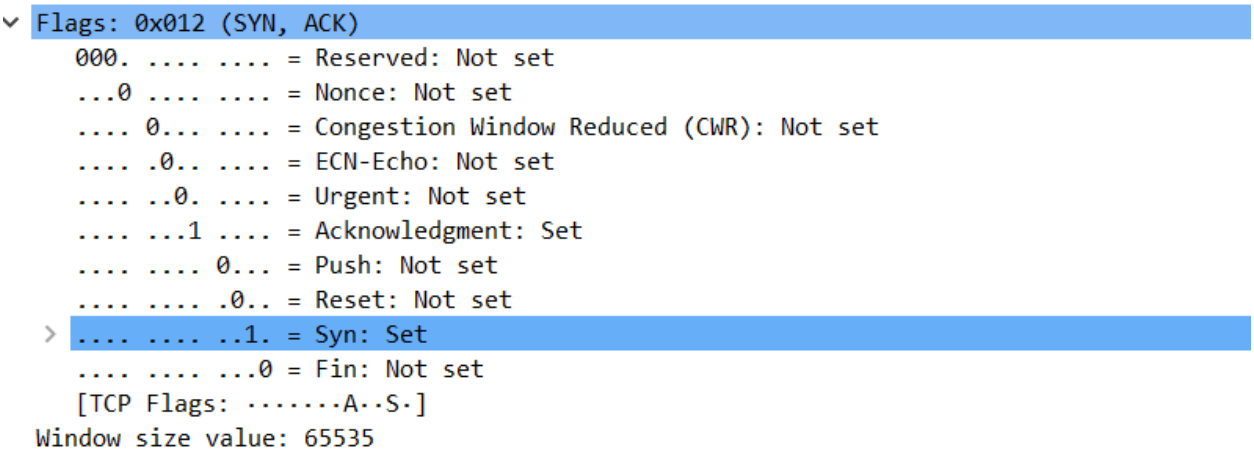
-Địa chỉ IP của Server (Source) là : 192.168.223.75

Kết nối TCP dùng để gửi và nhận các segments sử dụng port(Src Port):8080

Câu 9:



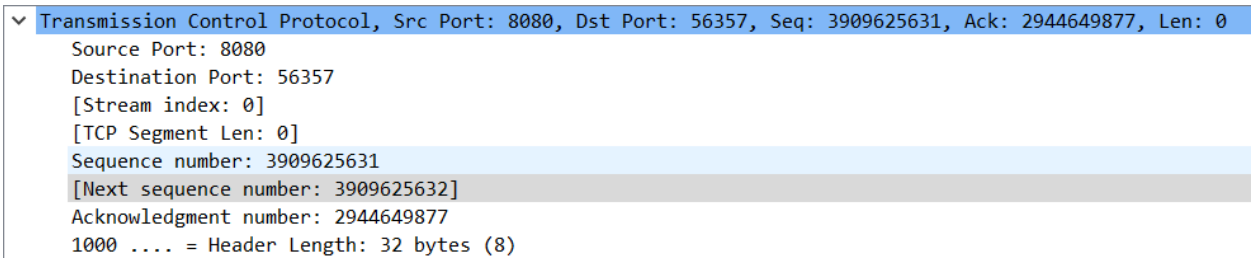
TCP SYN segment sử dụng sequence number 3909625631 để khởi tạo kết nối TCP giữa client và server



Thành phần SYN=1 trong Flags segment cho ta biết segment đó là TCP SYN segment

Câu 10:

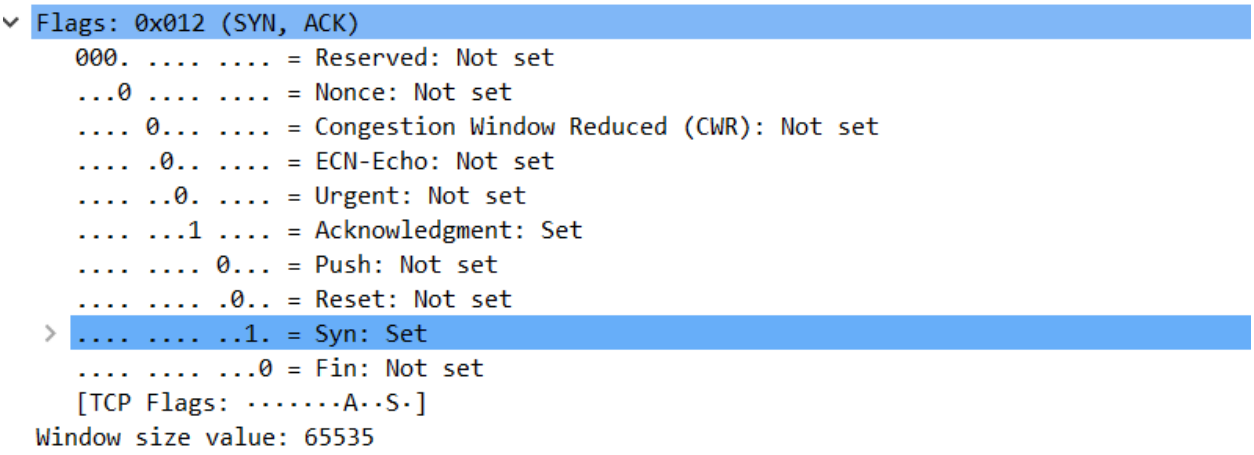
*Sequence number của gói tin SYN/ACK segment được gửi bởi server đến client để trả lời cho SYN segment



- Sequence number của gói tin SYN/ACK segment được gửi bởi server đến client là :SEQ =3909625631

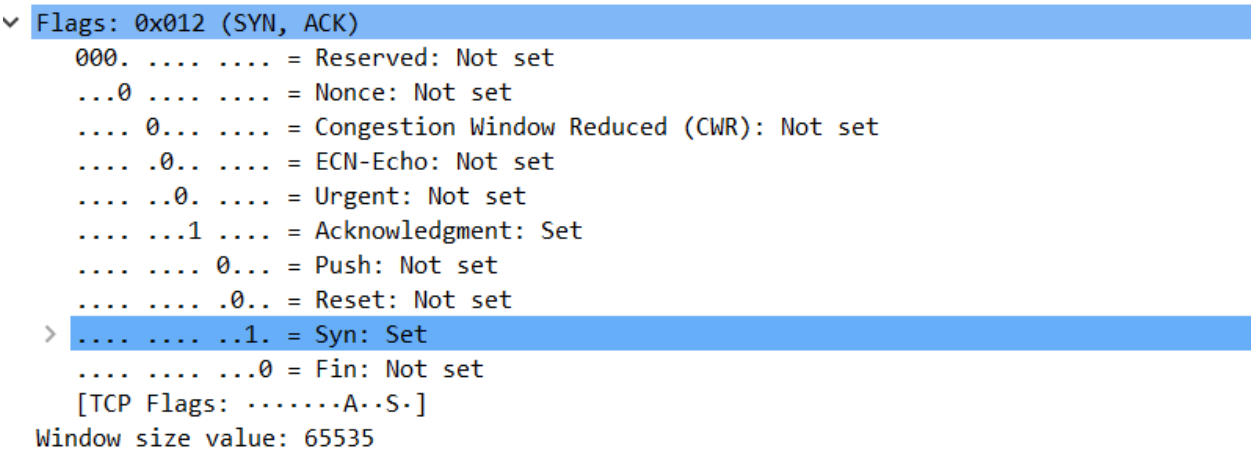
-Acknowledgment number của gói tin SYN/ACK segment được gửi bởi server đến client là:ACK=2944649877

*Tìm giá trị của Acknowledgement trong SYN/ACK segment?



- Giá trị Acknowledgement trong SYN/ACK segment=1

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



-Dựa trên giá trị Syn và Acknowledgement=1 trong phần Flags(SYN/ACK)

Câu 11:

1.

16	0.623149	192.168.223.75	192.168.223.123	TCP	157 8080 → 56357 [PSH, ACK] Seq=3909625632 Ack=2944650014 Win=262656 Len=103 [TCP segment of a reassembled PDU]
17	0.665106	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909625735 Win=131072 Len=0

2.

19	0.672159	192.168.223.75	192.168.223.123	TCP	447 8080 → 56357 [PSH, ACK] Seq=3909625735 Ack=2944650014 Win=262656 Len=393 [TCP segment of a reassembled PDU]
20	0.726925	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909626128 Win=130816 Len=0

3.

44	3.916778	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909626128 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
45	3.917383	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909627588 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
46	3.917413	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909629048 Win=131328 Len=0

4.

47	3.918816	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909629048 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
48	3.921078	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909630508 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
49	3.921107	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909631968 Win=131328 Len=0

5.

50	3.924497	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909631968 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
51	3.934513	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909633428 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
52	3.934566	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909634888 Win=131328 Len=0

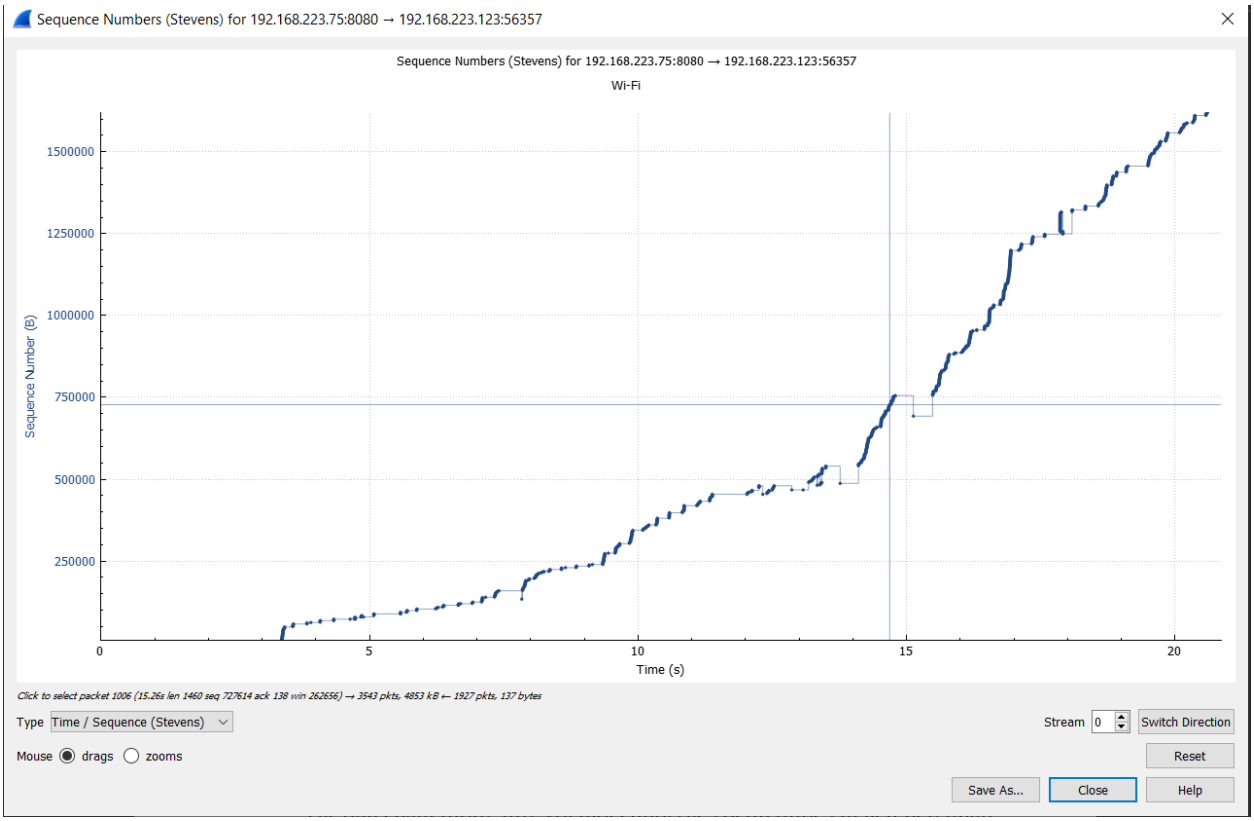
6.

53	3.937692	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909634888 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
54	3.937718	192.168.223.75	192.168.223.123	TCP	1514 8080 → 56357 [ACK] Seq=3909636348 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembled PDU]
55	3.937731	192.168.223.123	192.168.223.75	TCP	54 56357 → 8080 [ACK] Seq=2944650014 Ack=3909637808 Win=131328 Len=0

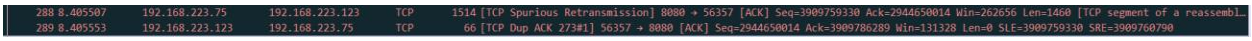
STT	Segment được gửi từ Server		Gói ACK gửi từ Client		RTT (Round Trip Time)
	SEQ Number	Time	ACK	Time	
1	25632	0.623149	25735	0.665106	0.041957
2	25735	0.672159	26128	0.726925	0.054766
3	26128	3.916778	29048	3.917413	0.000635
4	29048	3.918816	31968	3.921107	0.002291
5	31968	3.924497	34888	3.934566	0.010069
6	34888	3.937692	37808	3.937731	0.000039

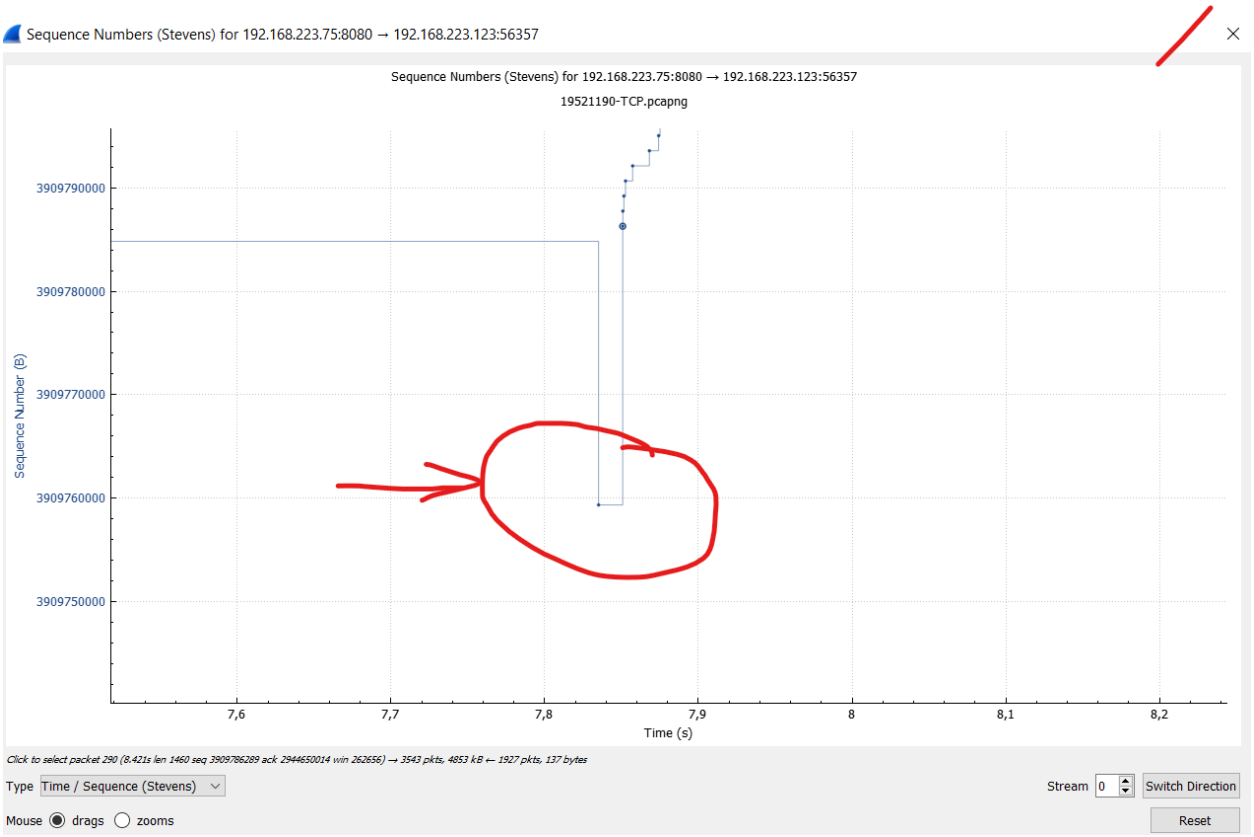
Câu 12:

Segment được gửi lại ,dựa vào biểu đồ ta thấy được :



Có Segment được gửi lại vì dựa trên đồ thị của Sequence number,những điểm bị trùng xuống xong rồi lại tăng lên lại đó chinh là những segment được gửi lại




Sequence Numbers (Stevens) for 192.168.223.75:8080 → 192.168.223.123:56357
Sequence Numbers (Stevens) for 192.168.223.75:8080 → 192.168.223.123:56357
19521190-TCP.pcapng
3909790000
3909780000
3909770000
3909760000
3909750000
Sequence Number (B)
7,6
7,7
7,8
7,9
8
8,1
8,2
Time (s)
Click to select packet 290 (8.421s len 1460 seq 3909786289 ack 2944650014 win 262656) → 3543 pkts, 4853 kB ← 1927 pkts, 137 bytes
Type
Time / Sequence (Stevens)
Stream
0
Switch Direction
Mouse
drags
zooms
Reset