

Họ Tên:Bùi Đức Anh

MSSV:19521190

## BÀI THỰC HÀNH SỐ 3



Đây là bắt VLC media của bạn Ngô Đức Trí

Câu 1:

```
2213 21.045871 192.168.223.75 [192.168.223.123] RTP 1442 PT=DynamicRTP-Type=96, SSRC=0x304ECA06, Seq=41899, Time=267987941
2214 21.045902 192.168.223.75 [192.168.223.123] RTP 363 PT=DynamicRTP-Type=96, SSRC=0x304ECA06, Seq=41900, Time=267987941, Mark
2215 21.046314 192.168.223.75 [192.168.223.123] RTP 1442 PT=DynamicRTP-Type=96, SSRC=0x304ECA06, Seq=41901, Time=267995141

> 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 (08:ec:4b:83:33:2f), Dst: WiutronI_F3:c0:9c (00:1b:b1:f3:c0:9c)
> Internet Protocol Version 4, Src: 192.168.223.75, Dst: 192.168.223.123
> 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]
```

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 )

-Length: 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: 63996
Destination Port: 62110
Length: 1408
Checksum: 0x49cd [unverified]
[Checksum Status: Unverified]
[Stream index: 15]
> [Timestamps]
```

> Real-Time Transport Protocol

```
0020 df 7b f9 fc f2 9e 05 80 49 cd 80 60 a3 ad 0f f9 .{.....I..`.
```

Source Port (udp.srcport), 2 bytes

-Source Port: 2byte

Destination Port (udp.dstport), 2 bytes

- Destination Port: 2 byte

Length (udp.length), 2 bytes

- Length: 2byte

Details at: [https://www.wireshark.org/docs/wsug\\_html\\_chunked/ChAdvChecksums.html](https://www.wireshark.org/docs/wsug_html_chunked/ChAdvChecksums.html) (udp.checksum), 2 bytes

-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	+·ON.... ..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	bb	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...d1

>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
> 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	+·ON.... ..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	bb	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...d1

Real-Time Transport Protocol (rtp), 1,400 bytes

Packets: 2215 - Displayed: 2157 (97.4%)

Giá trị của trường Length 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?

```

2215 21.046314 192.168.223.75 192.168.223.123 RTP 1442 PT=DynamicRTP-Type-96, SSRC=0x304ECA06, Seq=41901, Time=267995141
> Frame 2215: 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 (0c: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
> User Datagram Protocol, Src Port: 63996, Dst Port: 62110
    Source Port: 63996
    Destination Port: 62110
    Length: 1408
    Checksum: 0x49cd [unverified]
    [Checksum Status: Unverified]
    [Stream index: 15]
    > [Timestamps]
    > Real-Time Transport Protocol

```

Hex View:

```

0020 df 7b f9 fc f2 9e 05 80 49 cd 80 60 a3 ad 0f ff { ..... I`... .
0030 4b 05 30 4e ca 08 1c 81 9e c5 6a 46 7f df 1e 01 10N.....JF...
0040 bd 4f 08 af c8 1f 0f 9d 2a 07 e7 7e 4f 95 43 0.....*..0.C
0050 f3 91 15 46 2b 44 0d 0f 66 45 ea b2 d2 60 17 13 .-F+D- fe`...
0060 43 32 7e 0b 3e 6b d6 b2 27 88 02 eb 39 8d 62 70 C2->k- '...9 bp
0070 a4 79 d6 1c 15 a4 d2 06 2e f6 90 cc e0 93 1f 5a y.....Z
0080 26 91 04 f7 9f a0 ac 5a 9e 64 f4 dc 41 1d 34 9a &.....Z -d- A 4-
0090 09 30 82 02 8d 1e 35 93 4f d1 7b e5 79 7b 00 99 0.....5 O-{y{...
00a0 52 96 b7 3b 5e 0e ea 09 66 02 32 32 15 8f 4c 62 R..;^....f.22..Lb
00b0 1a 58 c6 c4 70 8d fd ac 3b 7f 14 71 b4 99 17 c1 -X- p...; q...
00c0 1d 53 37 db 05 01 75 61 de 60 63 0c 5e 3d e6 65 .S7- ua 'c.^=e
00d0 7f e2 02 2a 5f 48 e7 d2 41 7e 46 21 24 57 bb ...v..H- A-F!$W
00e0 3c 3c ec 0d af 6b 5a ec a4 2a 1e f0 af 27 47 16 <<- kZ- *...`G

```

Length (udp.length), 2 bytes | Packets: 2215 - Displayed: 2157 (97.4%) | Profile: Default

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

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

## Câu 5:

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

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

Source Port: 63996  
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:

Source: 192.168.223.75  
Destination: 192.168.223.123  
> Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 4817731, Ack: 138, Len: 1460

IP Client(Destination) :192.168.223.123

> Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 4817731, Ack: 138, Len: 1460

TCP Port Client(Dst Port): 56357

## Câu 8:

```
| Source: 192.168.223.75  
| Destination: 192.168.223.123  
> Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 0, Ack: 1, Len: 0
```

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

```
> Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 0, Ack: 1, Len: 0
```

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

## Câu 9:

```
✓ Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 3909625631, Ack: 2944649877, Len: 0  
  Source Port: 8080  
  Destination Port: 56357  
  [Stream index: 0]  
  [TCP Segment Len: 0]  
  Sequence number: 3909625631  
  [Next sequence number: 3909625632]  
  Acknowledgment number: 2944649877  
  1000 .... = Header Length: 32 bytes (8)
```

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

```
✓ 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  
> .... .... .1. = Syn: Set  
  .... .... ....0 = Fin: Not set  
  [TCP Flags: .....A..S.]  
  Window size value: 65535
```

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

```
✓ Transmission Control Protocol, Src Port: 8080, Dst Port: 56357, Seq: 3909625631, Ack: 2944649877, Len: 0
  Source Port: 8080
  Destination Port: 56357
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 3909625631
  [Next sequence number: 3909625632]
  Acknowledgment number: 2944649877
  1000 .... = Header Length: 32 bytes (8)
```

- 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?

```
✓ 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
  > .... .... .1. = Syn: Set
  .... .... ...0 = Fin: Not set
  [TCP Flags: .....A..S..]
  Window size value: 65535
```

- 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?

```
✓ 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
  > .... .... .1. = Syn: Set
  .... .... ...0 = Fin: Not set
  [TCP Flags: .....A..S..]
  Window size value: 65535
```

- 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

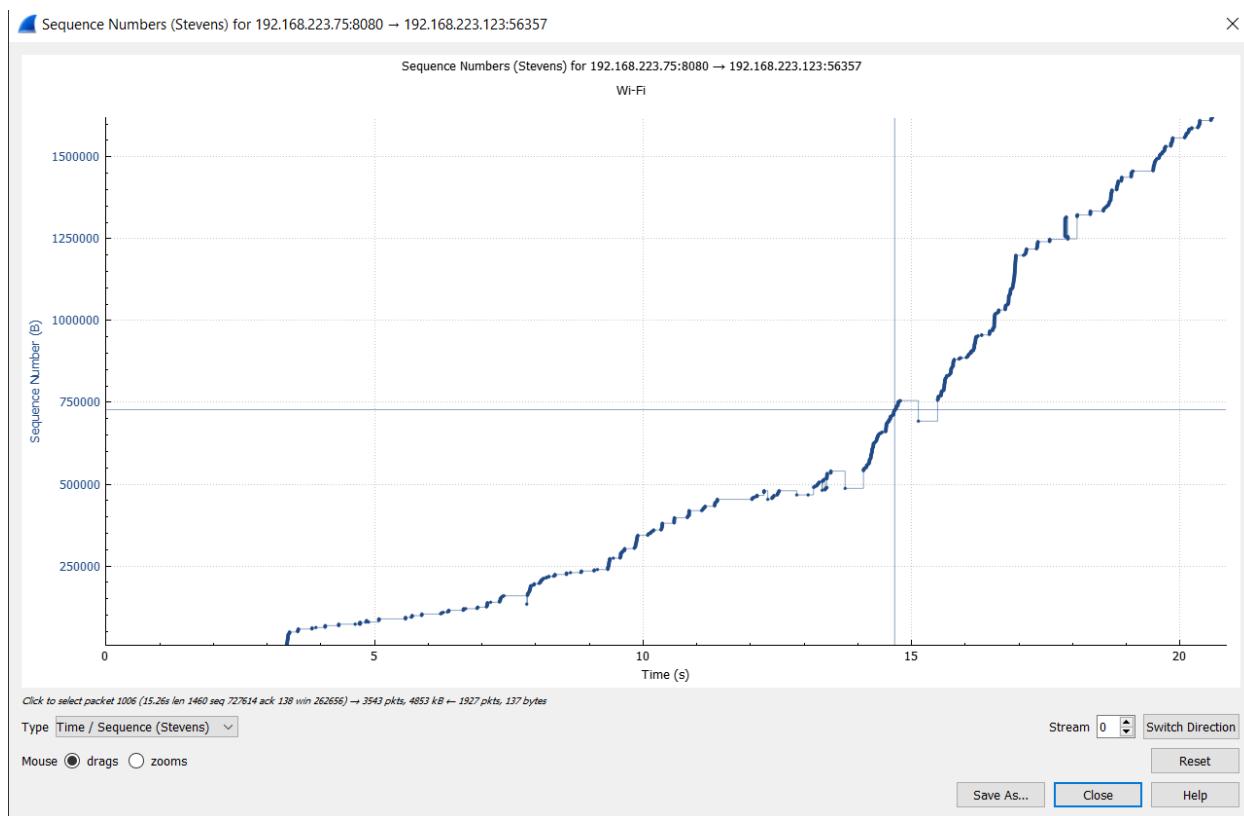
6.

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 chính là những segment được gửi lại

```
288 8.405507 192.168.223.75 192.168.223.123 TCP 1514 [TCP Spurious Retransmission] 8080 + 56357 [ACK] Seq=3909759330 Ack=2944650014 Win=262656 Len=1460 [TCP segment of a reassembl...
289 8.405553 192.168.223.123 192.168.223.75 TCP 66 [TCP Dup ACK 27#1] 56357 + 8080 [ACK] Seq=2944650014 Ack=3909786289 Win=0 SLE=3909759330 SRE=3909760798
```

