DNS, TCP – packet 분석

2022년 05월 10일 (화)

국민대학교

최건희 (20172268)

Contents

1 Summary

2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

3 TCP 패킷 분석

2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

Contents

✓ 1 **Summary**

2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

3 TCP 패킷 분석

2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

Summary

DNS

주요 내용

- www.naver.com에 접속하여 검색 창 클릭 후 [easyboan] 검색
- WireShark를 통해 DNS query and response Message 캡쳐
- 메시지 분석 수행

Summary

TCP

- 주요 내용
  - WireShark를 통해 TCP 캡쳐
  - TCP Segment 분석 수행
  - 기존 과제 코드를 이용한 3-Way Handshake 확인

Contents

1 Summary

✓ 2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

3 TCP 패킷 분석

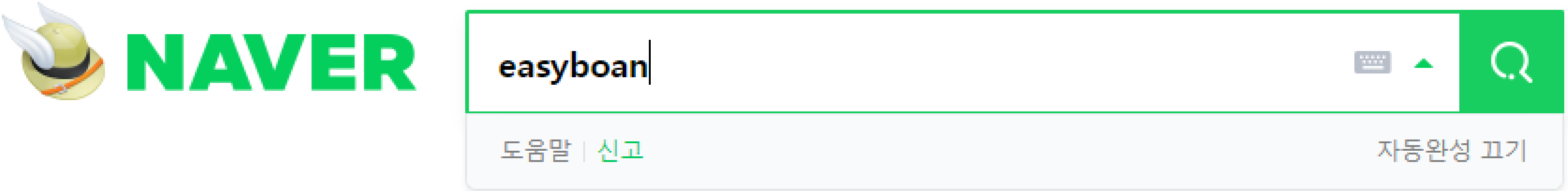
2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

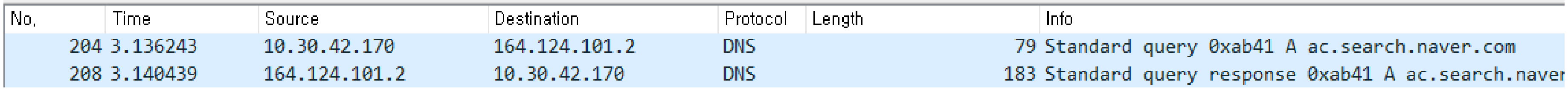
Query Message

Query & Response Message 캡쳐

- 웹사이트: www.naver.com
- NAVER 검색창에 [easyboan] 키워드 입력
- WireShark를 통해 query Message Capture



easyboan

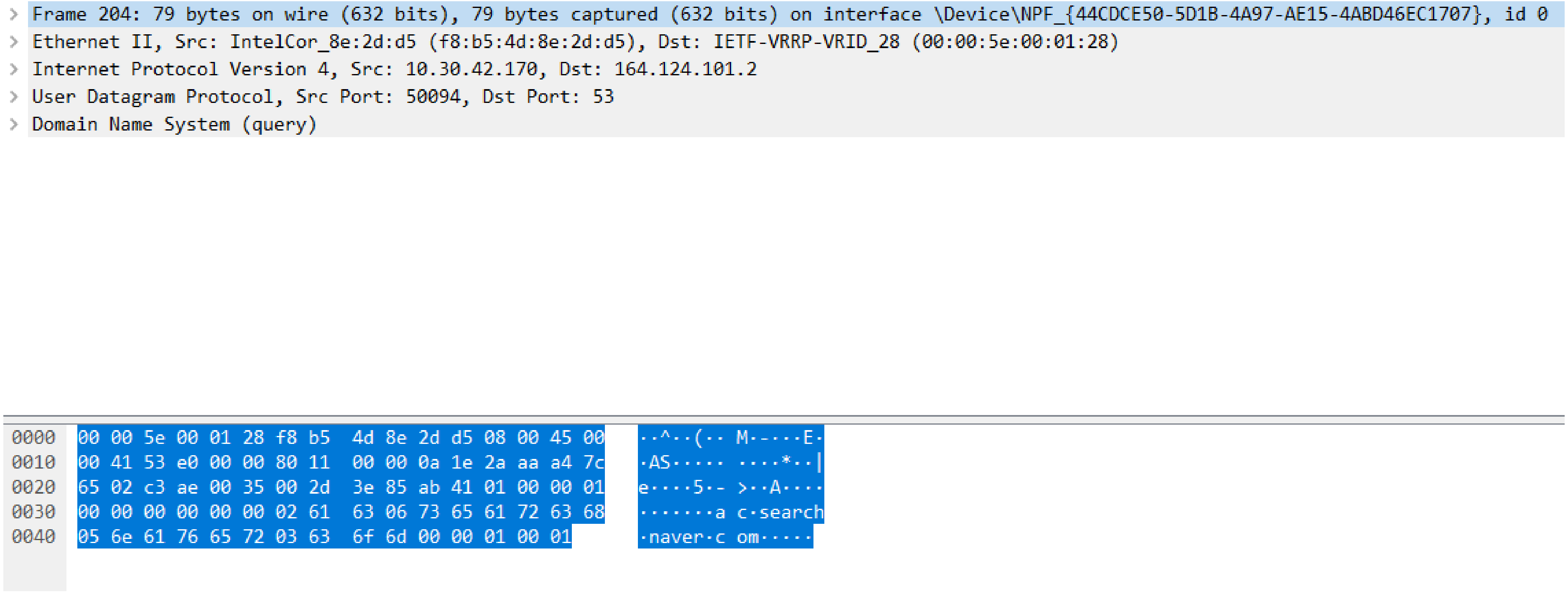


No.	Time	Source	Destination	Protocol	Length	Info
204	3.136243	10.30.42.170	164.124.101.2	DNS	79	Standard query 0xab41 A ac.search.naver.com
208	3.140439	164.124.101.2	10.30.42.170	DNS	183	Standard query response 0xab41 A ac.search.naver

Query Message

▣ Query Message 분석

❖ 5개 항목에 대한 순차적 분석



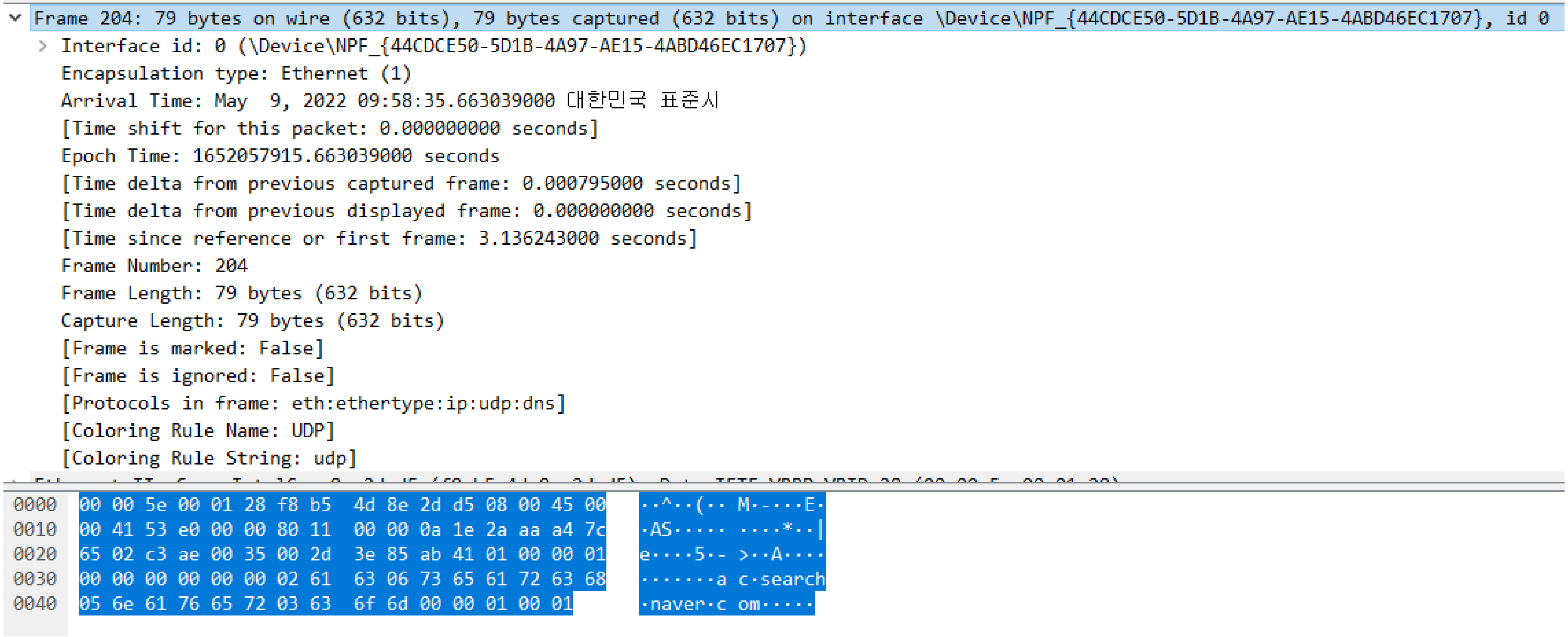
```
> Frame 204: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707}, id 0
> Ethernet II, Src: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5), Dst: IETF-VRRP-VRID_28 (00:00:5e:00:01:28)
> Internet Protocol Version 4, Src: 10.30.42.170, Dst: 164.124.101.2
> User Datagram Protocol, Src Port: 50094, Dst Port: 53
> Domain Name System (query)
```

```
0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

Query Message

Query Message 분석

- 메시지 전체에 대한 정보(meta data제공)



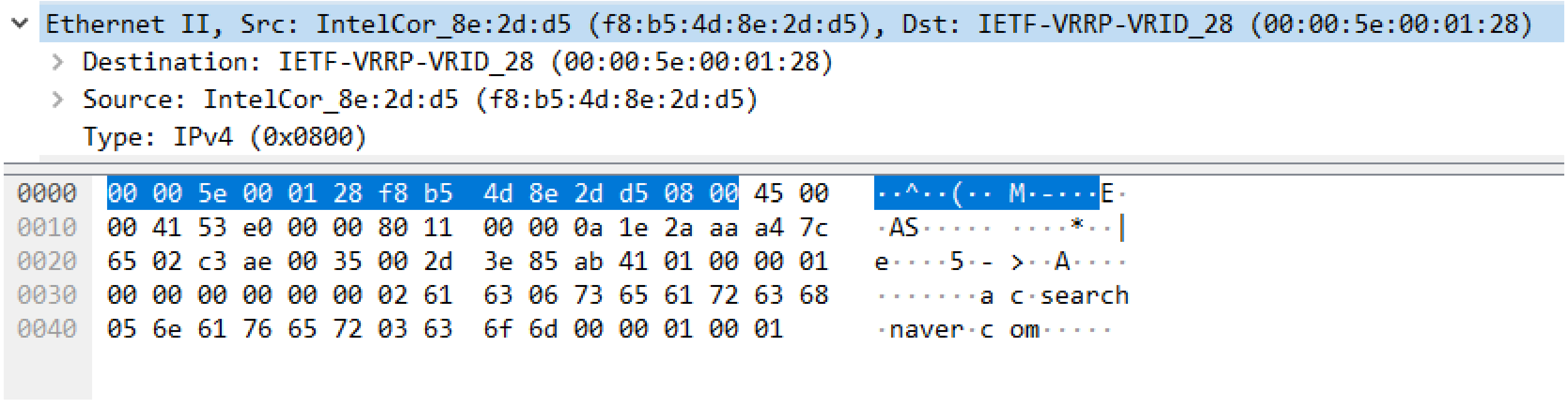
```
∨ Frame 204: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707}, id 0
  > Interface id: 0 (\Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707})
    Encapsulation type: Ethernet (1)
    Arrival Time: May  9, 2022 09:58:35.663039000 대한민국 표준시
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1652057915.663039000 seconds
    [Time delta from previous captured frame: 0.000795000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 3.136243000 seconds]
    Frame Number: 204
    Frame Length: 79 bytes (632 bits)
    Capture Length: 79 bytes (632 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:udp:dns]
    [Coloring Rule Name: UDP]
    [Coloring Rule String: udp]
> Ethernet II, Src: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5), Dst: IETF-VRRP-VRID_28 (00:00:5e:00:01:28)
```

```
0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

Query Message

Query Message 분석

- 출발지와 목적지에 관한 정보 및 네트워크에 관한 정보
  - 이더넷 II 사용

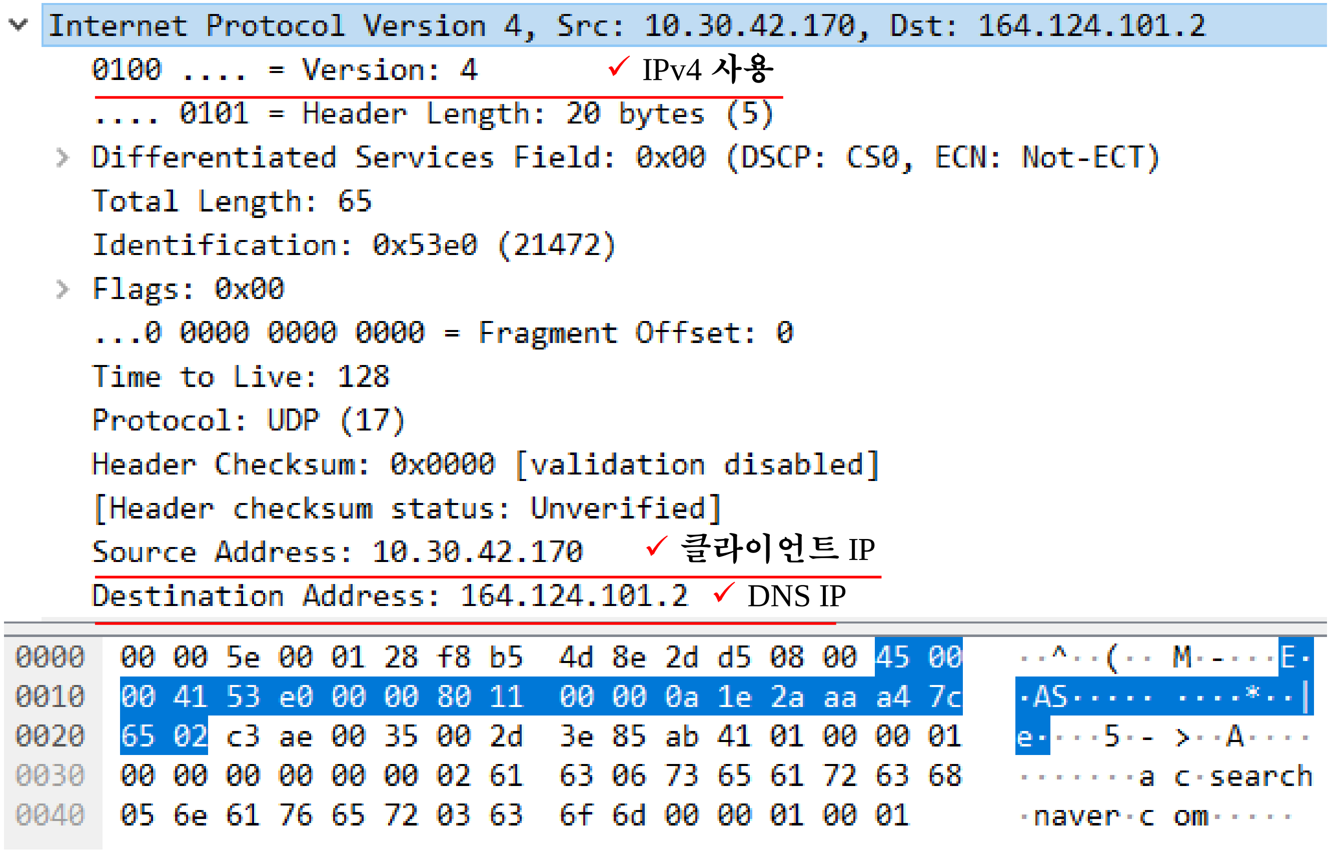


```
v Ethernet II, Src: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5), Dst: IETF-VRRP-VRID_28 (00:00:5e:00:01:28)
  > Destination: IETF-VRRP-VRID_28 (00:00:5e:00:01:28)
  > Source: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5)
    Type: IPv4 (0x0800)

0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

Query Message

Query Message 분석

- IPv4 사용
  - Source (Client): 10.30.42.170 Destination(DNS): 164.124.101.2
  - IPv4를 사용해서 클라이언트가 DNS로 Query를 전송



```
∨ Internet Protocol Version 4, Src: 10.30.42.170, Dst: 164.124.101.2
     0100 .... = Version: 4
     .... 0101 = Header Length: 20 bytes (5)
   > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
     Total Length: 65
     Identification: 0x53e0 (21472)
   > Flags: 0x00
     ...0 0000 0000 0000 = Fragment Offset: 0
     Time to Live: 128
     Protocol: UDP (17)
     Header Checksum: 0x0000 [validation disabled]
     [Header checksum status: Unverified]
     Source Address: 10.30.42.170
     Destination Address: 164.124.101.2

0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

✓ IPv4 사용

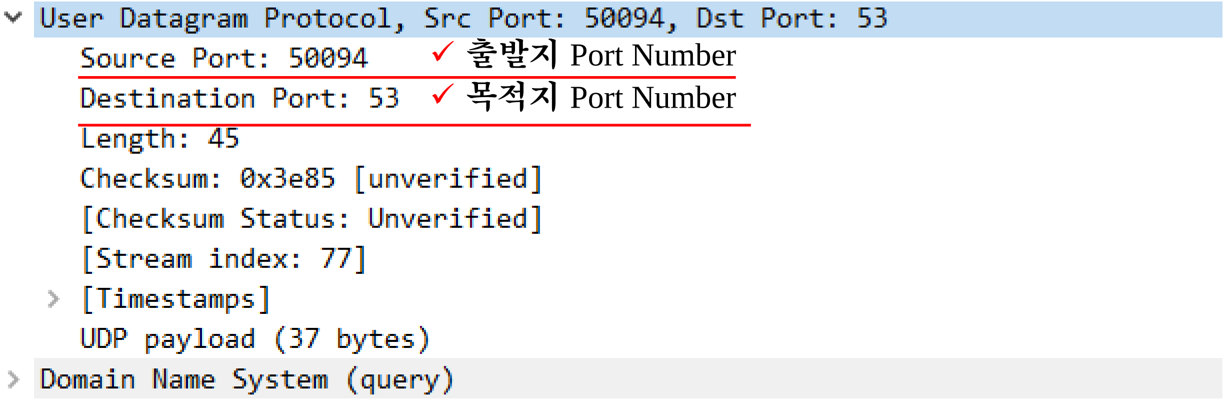
✓ 클라이언트 IP

✓ DNS IP

Query Message

Query Message 분석

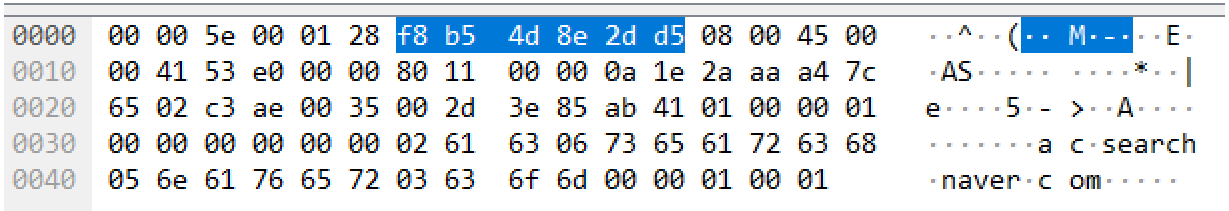
- UDP 사용
  - Source Port : 50094(임의의 number)
  - Destination Port : 53 (TCP&UDP를 사용하는 DNS 공식 포트번호)



```
⌄ User Datagram Protocol, Src Port: 50094, Dst Port: 53
    Source Port: 50094
    Destination Port: 53
    Length: 45
    Checksum: 0x3e85 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 77]
  › [Timestamps]
    UDP payload (37 bytes)
› Domain Name System (query)
```

✓ 출발지 Port Number

✓ 목적지 Port Number



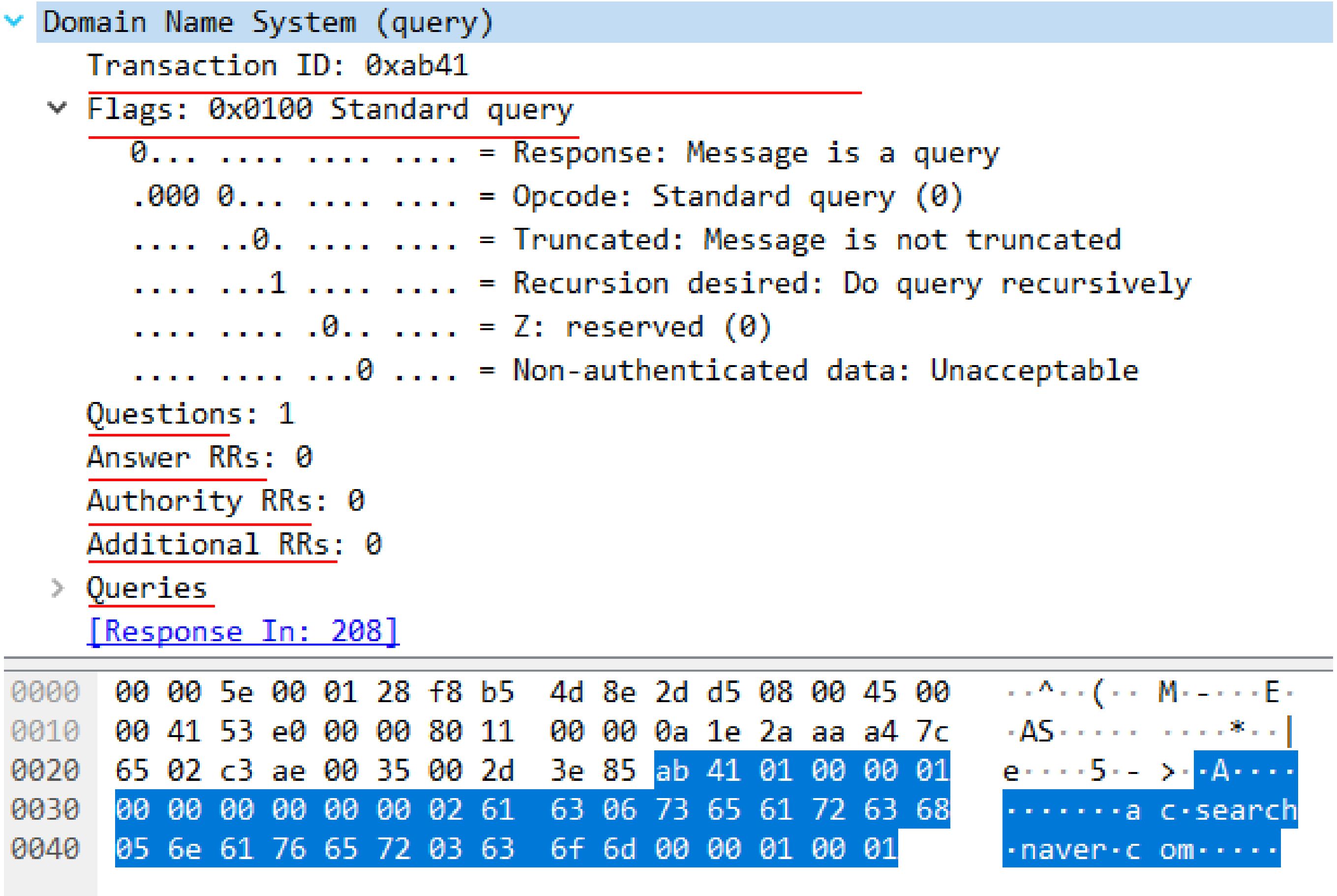
```
0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·--··E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

Query Message

■ Query Message 분석

- ❖ DNS Query 분석

- ➢ Transaction ID, Flags, QAAA, Queries의 4개 항목으로 구성



```
⌄ Domain Name System (query)
    Transaction ID: 0xab41
  ⌄ Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  > Queries
    [Response In: 208]

0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

Query Message

▣ Query Message 분석

- ❖ DNS Query 분석

➢ DNS Header Format

Tracsaction ID – Client가 보낸 Query와 수신 된 Response간 일치 여부 확인							

QR
Query(0)
Response(1)
	Opcode
	Query 유형
		AA
		공식
		DNS서버의
		응답 시 (1)
			TC
			응답 분할
			(1)
				RD
				재귀여부(1)
					RA
					DNS서버
					재귀질의가능
					여부 (1)
						Reserved
						예약공간
							rCode
							오류표시

QD Count (Questions) -Question Section의 개수를 표시 (일반적으로 1개)							
AN Count (Answer Resource Record) -Answer Section의 개수를 표시							
NS Count (Authority Resource Record) -Authority Section의 개수를 표시							
AR Count (Additional Resource Record) -Additional Section의 개수를 표시							

[1] DarkSoul.Story, 『기본적인 DNS Packet분석』, 2021.05.09, https://darksoulstory.tistory.com/62

[2] YoungQ, 『 와이어 샤크 DNS패킷』, 2021.05.09, https://youngq.tistory.com/56

Query Message

▣ Query Message 분석

❖ DNS Query 분석

➢ Opcode

0	Query
1	Inverse Query
2	Status
3	Unassigned
4	Notify
5	Update
6 ~ 15	Unassigned

➢ RCODE

0	NoError	오류 없음
1	Form Err	형식오류(쿼리가 잘못된 경우)
2	ServFai	서버 실패(운 서버자체의 문제로 실패)
3	NXDomain	네임 오류 (도메인 네임이 존재하지 않는 경우)
4	Notimp	DNS 서버가 Query를 지원하지 못함
5	Refused	거부(정책적인 이유로 Query를 거절함)

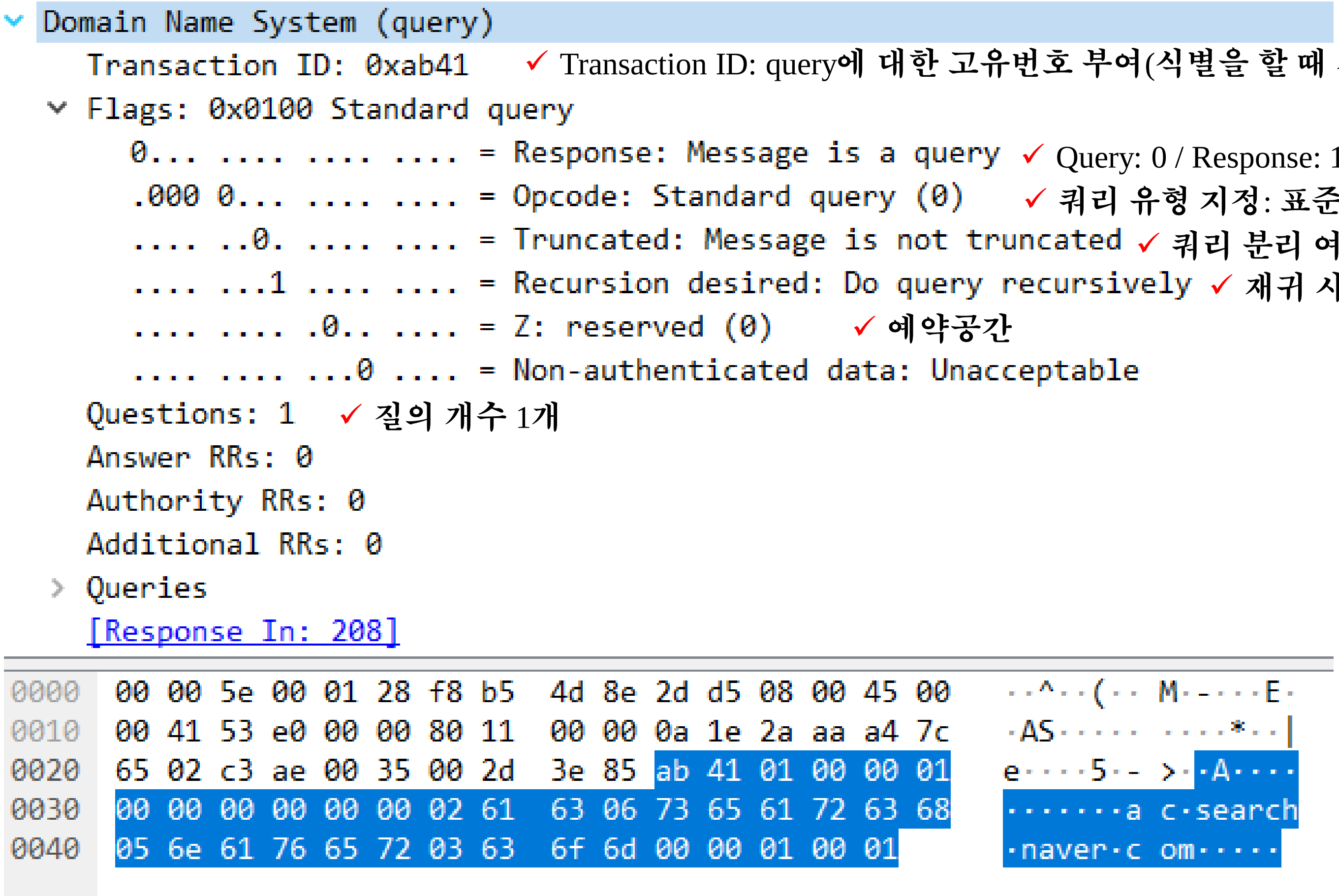
[3] iana,『Domain Name System (DNS) Parameters』, 2021.05.09, http://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml

Query Message

Query Message 분석

❖ **DNS Query 분석**

➢ **Transaction ID, Flags, QAAA에 대한 분석**



```
v Domain Name System (query)
    Transaction ID: 0xab41
  v Flags: 0x0100 Standard query
      0... .... .... .... = Response: Message is a query
      .000 0... .... .... = Opcode: Standard query (0)
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... .0.. .... = Z: reserved (0)
      .... .... ...0 .... = Non-authenticated data: Unacceptable
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  > Queries
    [Response In: 208]
```

```
0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0020  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0030  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

✓ Transaction ID: query에 대한 고유번호 부여(식별을 할 때 사용)

✓ Query: 0 / Response: 1

✓ 쿼리 유형 지정: 표준 질의(0000)

✓ 쿼리 분리 여부: 분리 시 (1)로 변환

✓ 재귀 사용 여부 표시

✓ 예약공간

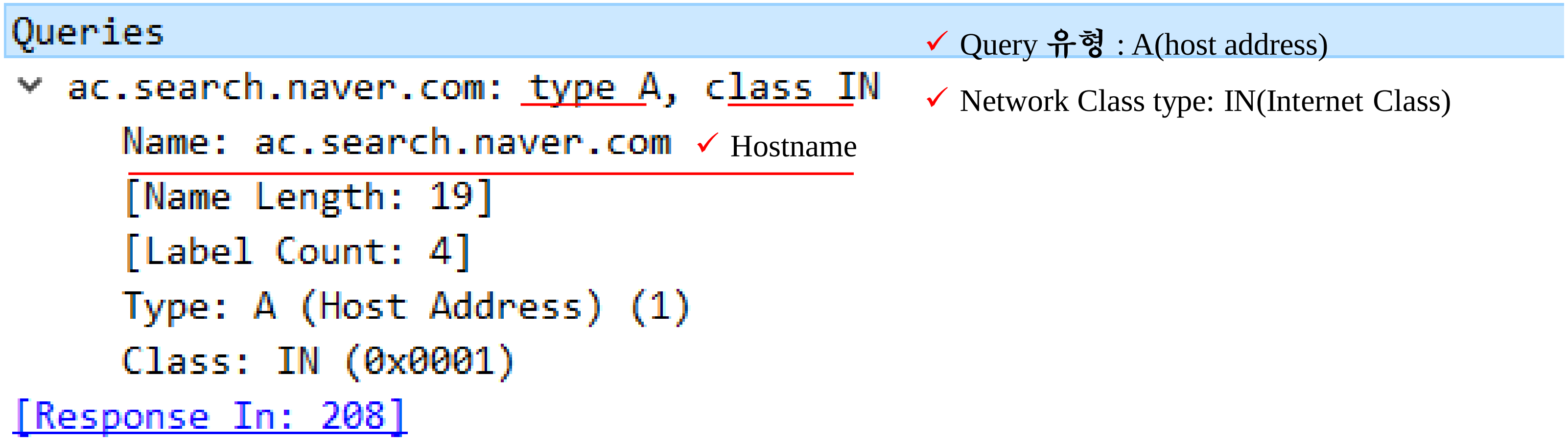
✓ 질의 개수 1개

Query Message

Query Message 분석

- DNS Query 분석

Queries에 대한 분석

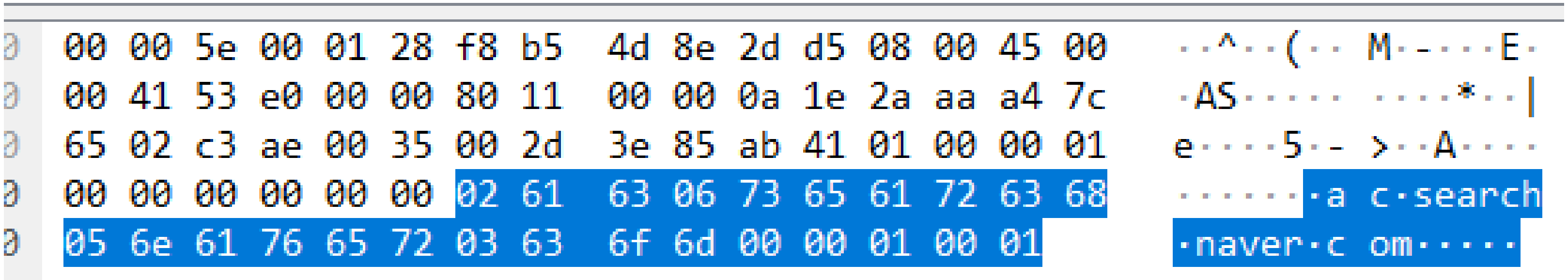


```
Queries
  ac.search.naver.com: type A, class IN
      Name: ac.search.naver.com
      [Name Length: 19]
      [Label Count: 4]
      Type: A (Host Address) (1)
      Class: IN (0x0001)
  [Response In: 208]
```

✓ Hostname

✓ Query 유형 : A(host address)

✓ Network Class type: IN(Internet Class)



```
0  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0  00 41 53 e0 00 00 80 11  00 00 0a 1e 2a aa a4 7c   ·AS····· ····*··|
0  65 02 c3 ae 00 35 00 2d  3e 85 ab 41 01 00 00 01   e····5·- >··A····
0  00 00 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01      ·naver·c om·····
```

- Query 유형 : A(Host Address), NS(Authoritatiove Name Server), MX(Mail Exchange) 등
- Class 유형 : IN(Internet Class), CS(Csnet Class), CH(Chaos Class), HS(HeSiod Class) 등

Contents

1 Summary

✓ 2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

3 TCP 패킷 분석

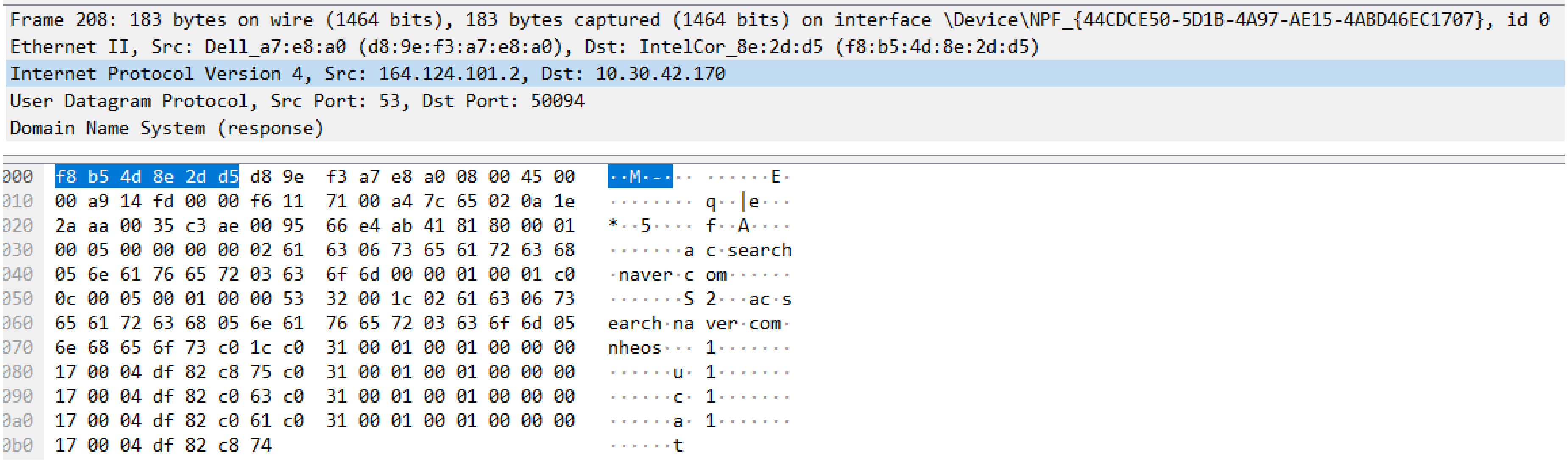
2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

Query Message

Response Message 분석

- 5개 항목에 대한 순차적 분석



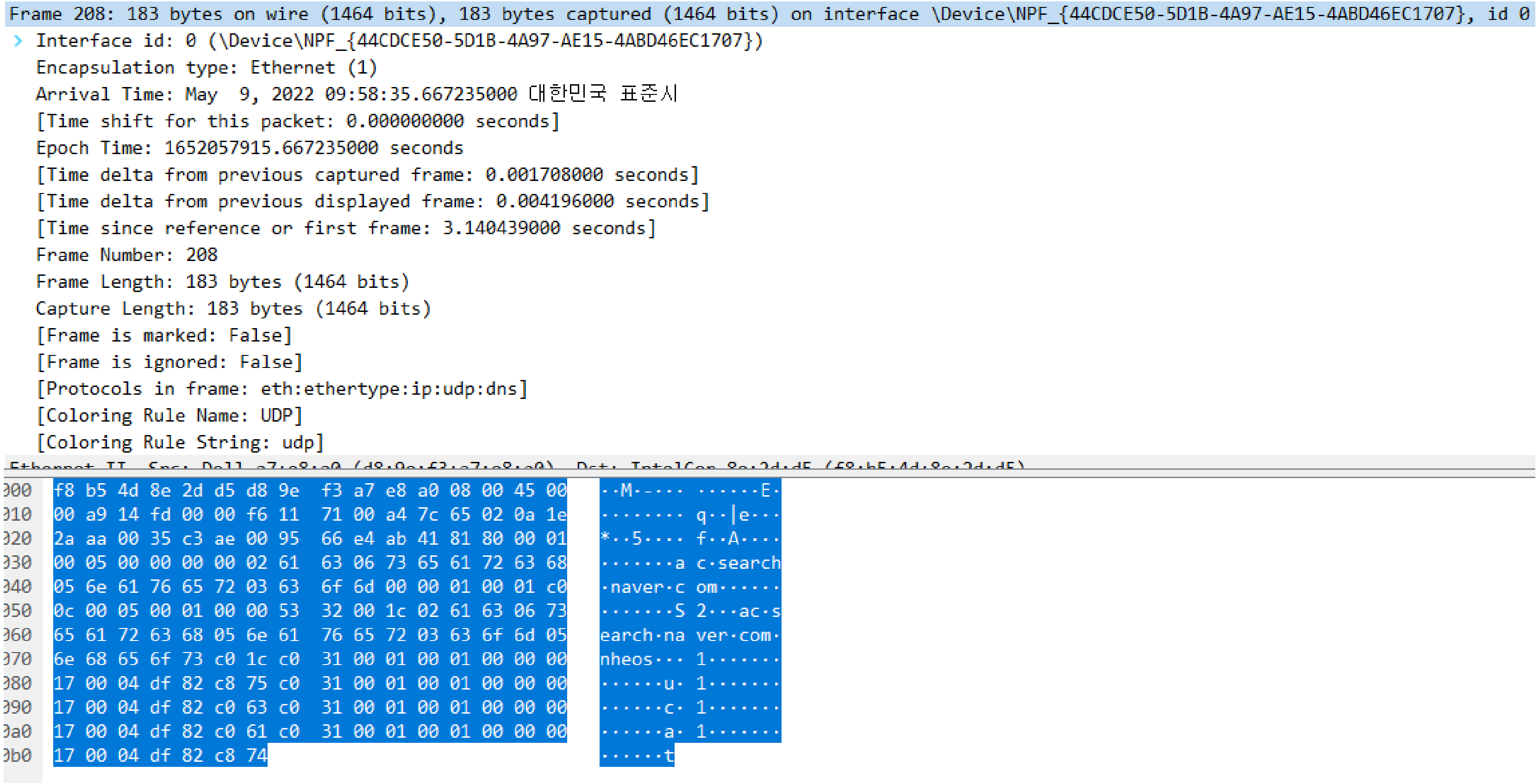
```
Frame 208: 183 bytes on wire (1464 bits), 183 bytes captured (1464 bits) on interface \Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707}, id 0
Ethernet II, Src: Dell_a7:e8:a0 (d8:9e:f3:a7:e8:a0), Dst: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5)
Internet Protocol Version 4, Src: 164.124.101.2, Dst: 10.30.42.170
User Datagram Protocol, Src Port: 53, Dst Port: 50094
Domain Name System (response)
```

```
000  f8 b5 4d 8e 2d d5 d8 9e  f3 a7 e8 a0 08 00 45 00   ··M·-··· ······E·
010  00 a9 14 fd 00 00 f6 11  71 00 a4 7c 65 02 0a 1e   ········ q··|e···
020  2a aa 00 35 c3 ae 00 95  66 e4 ab 41 81 80 00 01   *··5···· f··A····
030  00 05 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01 c0   ·naver·c om······
050  0c 00 05 00 01 00 00 53  32 00 1c 02 61 63 06 73   ·······S 2···ac·s
060  65 61 72 63 68 05 6e 61  76 65 72 03 63 6f 6d 05   earch·na ver·com·
070  6e 68 65 6f 73 c0 1c c0  31 00 01 00 01 00 00 00   nheos··· 1·······
080  17 00 04 df 82 c8 75 c0  31 00 01 00 01 00 00 00   ······u· 1·······
090  17 00 04 df 82 c0 63 c0  31 00 01 00 01 00 00 00   ······c· 1·······
0a0  17 00 04 df 82 c0 61 c0  31 00 01 00 01 00 00 00   ······a· 1·······
0b0  17 00 04 df 82 c8 74                               ······t
```

Query Message

Response Message 분석

- 메시지 전체에 대한 정보(meta data제공)



```
Frame 208: 183 bytes on wire (1464 bits), 183 bytes captured (1464 bits) on interface \Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707}, id 0
 > Interface id: 0 (\Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707})
   Encapsulation type: Ethernet (1)
   Arrival Time: May  9, 2022 09:58:35.667235000 대한민국 표준시
   [Time shift for this packet: 0.000000000 seconds]
   Epoch Time: 1652057915.667235000 seconds
   [Time delta from previous captured frame: 0.001708000 seconds]
   [Time delta from previous displayed frame: 0.004196000 seconds]
   [Time since reference or first frame: 3.140439000 seconds]
   Frame Number: 208
   Frame Length: 183 bytes (1464 bits)
   Capture Length: 183 bytes (1464 bits)
   [Frame is marked: False]
   [Frame is ignored: False]
   [Protocols in frame: eth:ethertype:ip:udp:dns]
   [Coloring Rule Name: UDP]
   [Coloring Rule String: udp]
Ethernet II, Src: Dell_a7:e8:a0 (d8:9e:f3:a7:e8:a0), Dst: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5)

0000  f8 b5 4d 8e 2d d5 d8 9e  f3 a7 e8 a0 08 00 45 00   ··M·-··· ······E·
0010  00 a9 14 fd 00 00 f6 11  71 00 a4 7c 65 02 0a 1e   ········ q··|e···
0020  2a aa 00 35 c3 ae 00 95  66 e4 ab 41 81 80 00 01   *··5···· f··A····
0030  00 05 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01 c0   ·naver·c om······
0050  0c 00 05 00 01 00 00 53  32 00 1c 02 61 63 06 73   ·······S 2···ac·s
0060  65 61 72 63 68 05 6e 61  76 65 72 03 63 6f 6d 05   earch·na ver·com·
0070  6e 68 65 6f 73 c0 1c c0  31 00 01 00 01 00 00 00   nheos··· 1·······
0080  17 00 04 df 82 c8 75 c0  31 00 01 00 01 00 00 00   ······u· 1·······
0090  17 00 04 df 82 c0 63 c0  31 00 01 00 01 00 00 00   ······c· 1·······
00a0  17 00 04 df 82 c0 61 c0  31 00 01 00 01 00 00 00   ······a· 1·······
00b0  17 00 04 df 82 c8 74                               ······t
```

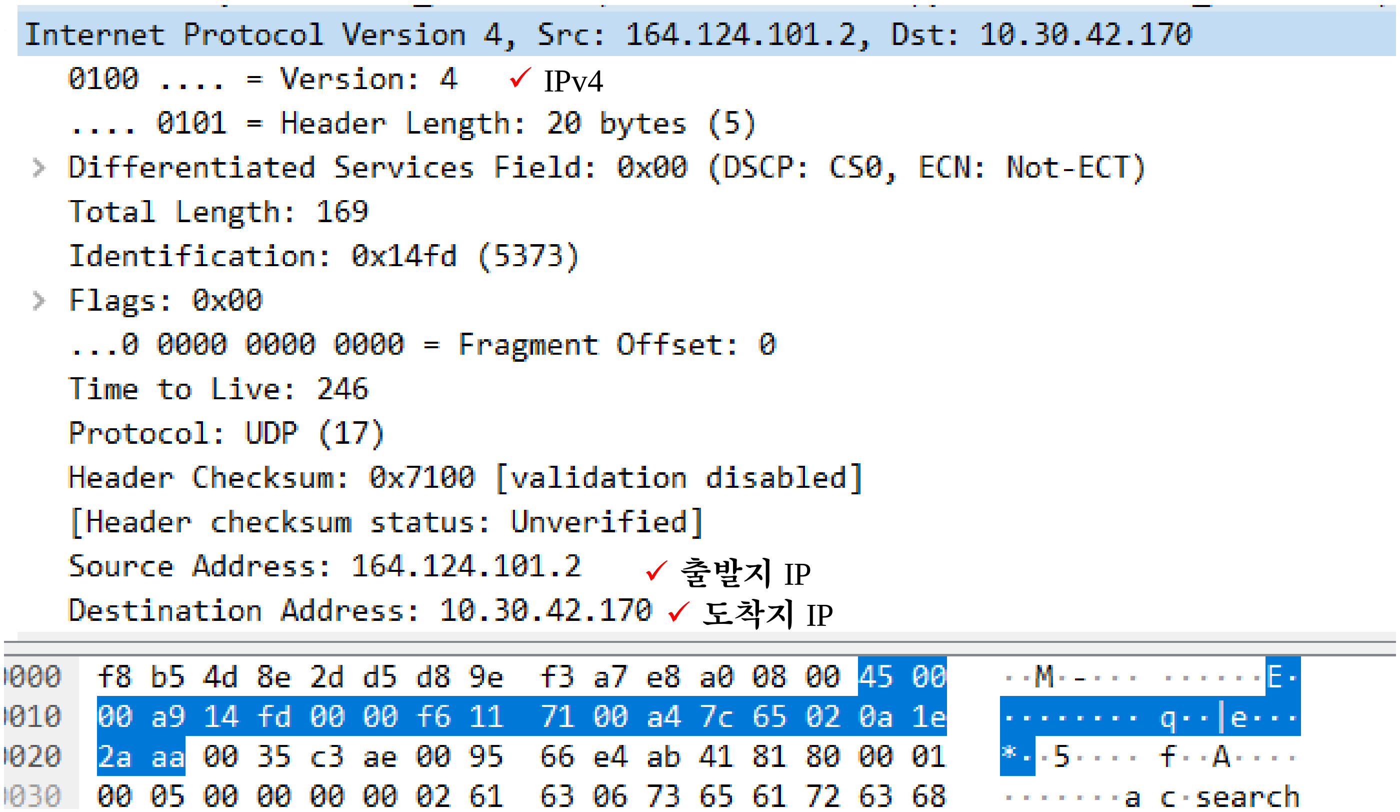
Query Message

Response Message 분석

- **IPv4 사용**

➢ **Source(DNS): 164.124.101.2**   **Destination (Client) : 10.30.42.170**

➢ **IPv4를 사용해서 Response를 전송**

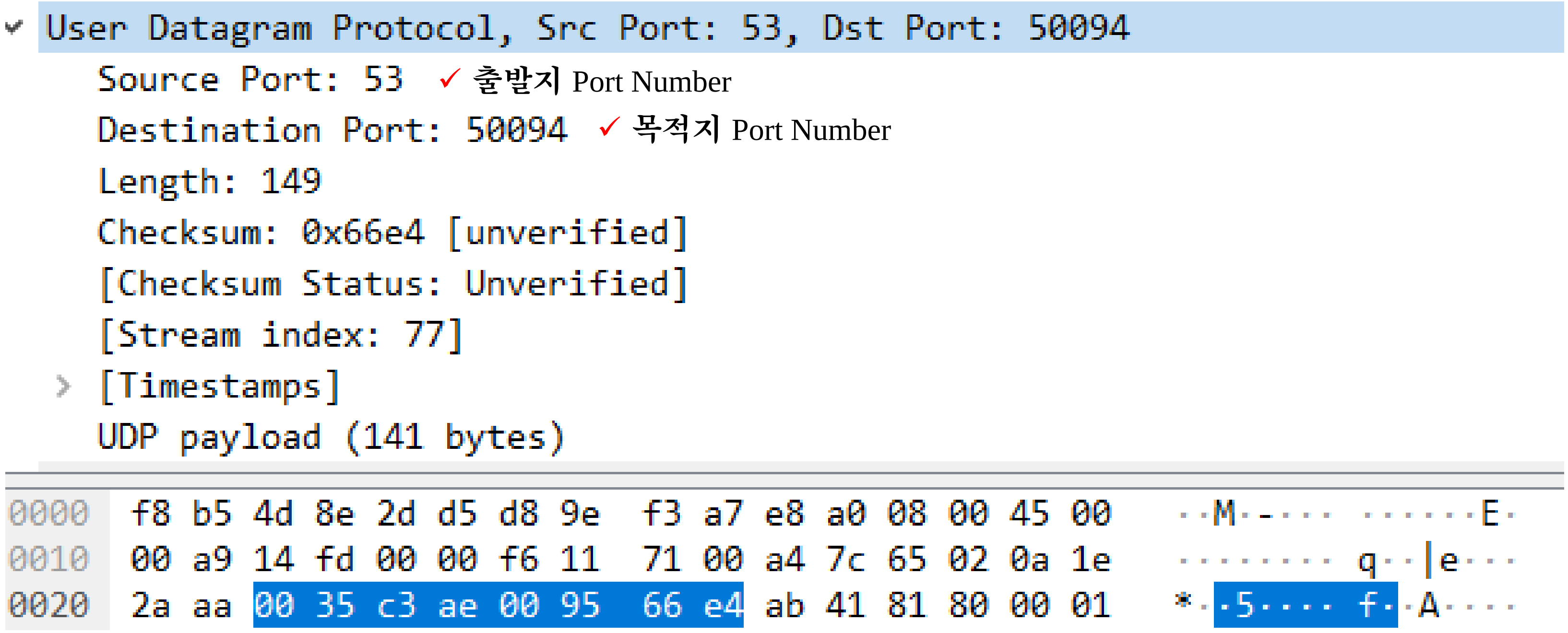


```
Internet Protocol Version 4, Src: 164.124.101.2, Dst: 10.30.42.170
    0100 .... = Version: 4     ✓ IPv4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 169
    Identification: 0x14fd (5373)
  > Flags: 0x00
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 246
    Protocol: UDP (17)
    Header Checksum: 0x7100 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 164.124.101.2      ✓ 출발지 IP
    Destination Address: 10.30.42.170 ✓ 도착지 IP

000  f8 b5 4d 8e 2d d5 d8 9e  f3 a7 e8 a0 08 00 45 00   ··M·-··· ······E·
010  00 a9 14 fd 00 00 f6 11  71 00 a4 7c 65 02 0a 1e   ········ q··|e···
020  2a aa 00 35 c3 ae 00 95  66 e4 ab 41 81 80 00 01   *··5···· f··A····
030  00 05 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a c·search
```

Query Message

Response Message 분석

- **UDP 사용**
  - **Source Port :53**
  - **Destination Port : 50094**



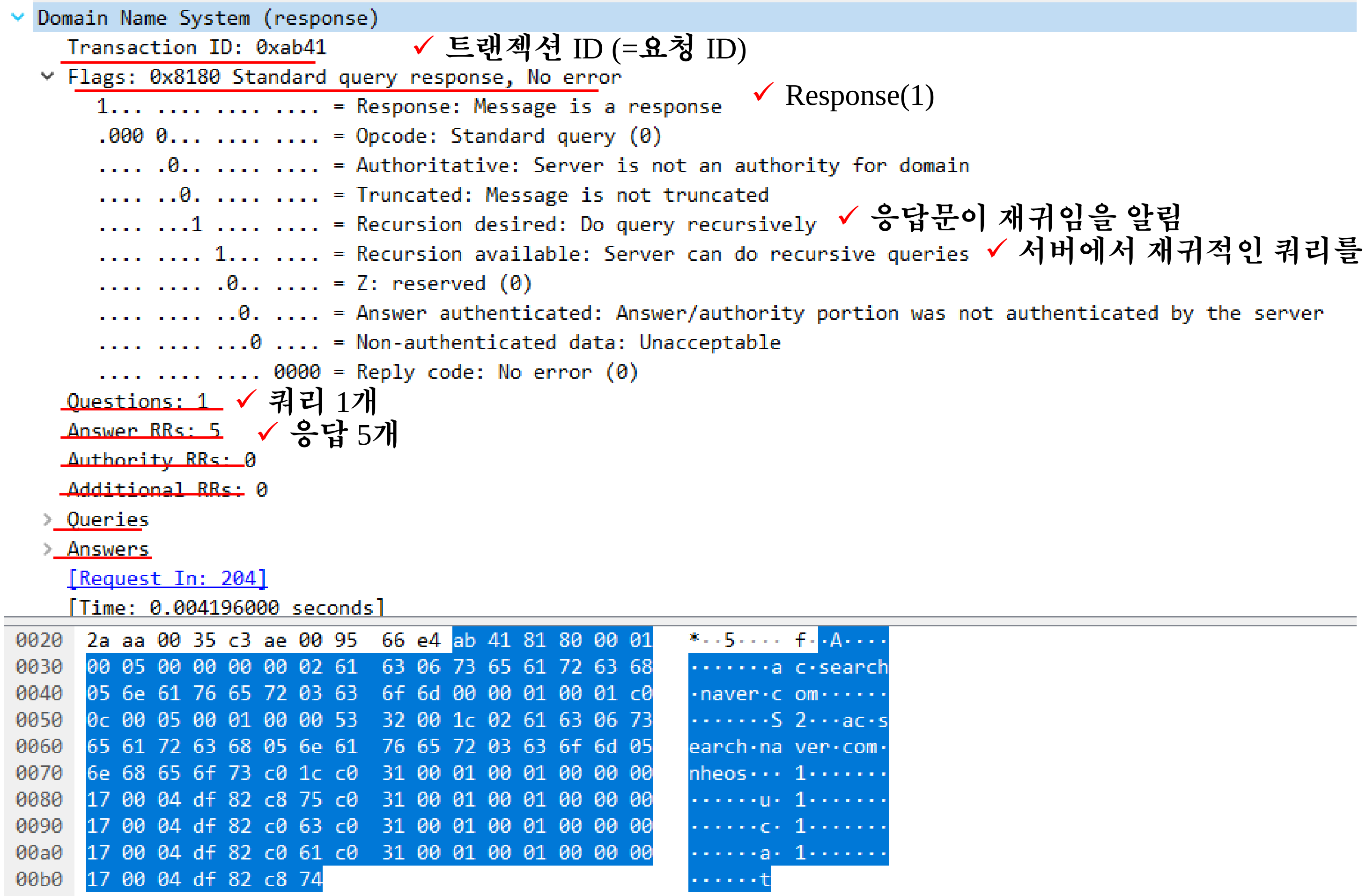
```
⌄ User Datagram Protocol, Src Port: 53, Dst Port: 50094
    Source Port: 53  ✓ 출발지 Port Number
    Destination Port: 50094  ✓ 목적지 Port Number
    Length: 149
    Checksum: 0x66e4 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 77]
  › [Timestamps]
    UDP payload (141 bytes)

0000  f8 b5 4d 8e 2d d5 d8 9e  f3 a7 e8 a0 08 00 45 00   ··M·-··· ······E·
0010  00 a9 14 fd 00 00 f6 11  71 00 a4 7c 65 02 0a 1e   ········ q··|e···
0020  2a aa 00 35 c3 ae 00 95  66 e4 ab 41 81 80 00 01   *··5···· f··A····
```

Query Message

■ Query Message 분석

❖ DNS Query 분석

➢ **Transaction ID, Flags, QAAA, Queries, Answers의 5개 항목으로 구성**



```
v Domain Name System (response)
    Transaction ID: 0xab41
  v Flags: 0x8180 Standard query response, No error
      1... .... .... .... = Response: Message is a response
      .000 0... .... .... = Opcode: Standard query (0)
      .... .0.. .... .... = Authoritative: Server is not an authority for domain
      .... ..0. .... .... = Truncated: Message is not truncated
      .... ...1 .... .... = Recursion desired: Do query recursively
      .... .... 1... .... = Recursion available: Server can do recursive queries
      .... .... .0.. .... = Z: reserved (0)
      .... .... ..0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
      .... .... ...0 .... = Non-authenticated data: Unacceptable
      .... .... .... 0000 = Reply code: No error (0)
    Questions: 1
    Answer RRs: 5
    Authority RRs: 0
    Additional RRs: 0
  > Queries
  > Answers
    [Request In: 204]
    [Time: 0.004196000 seconds]
```

```
0020  2a aa 00 35 c3 ae 00 95  66 e4 ab 41 81 80 00 01   *··5····  f··A····
0030  00 05 00 00 00 00 02 61  63 06 73 65 61 72 63 68   ·······a  c·search
0040  05 6e 61 76 65 72 03 63  6f 6d 00 00 01 00 01 c0   ·naver·c  om······
0050  0c 00 05 00 01 00 00 53  32 00 1c 02 61 63 06 73   ·······S  2···ac·s
0060  65 61 72 63 68 05 6e 61  76 65 72 03 63 6f 6d 05   earch·na  ver·com·
0070  6e 68 65 6f 73 c0 1c c0  31 00 01 00 01 00 00 00   nheos···  1·······
0080  17 00 04 df 82 c8 75 c0  31 00 01 00 01 00 00 00   ······u·  1·······
0090  17 00 04 df 82 c0 63 c0  31 00 01 00 01 00 00 00   ······c·  1·······
00a0  17 00 04 df 82 c0 61 c0  31 00 01 00 01 00 00 00   ······a·  1·······
00b0  17 00 04 df 82 c8 74                               ······t
```

✓ 트랜젝션 ID (=요청 ID)

✓ Response(1)

✓ 응답문이 재귀임을 알림

✓ 서버에서 재귀적인 쿼리를 처리할 수 있음을 알림

✓ 쿼리 1개

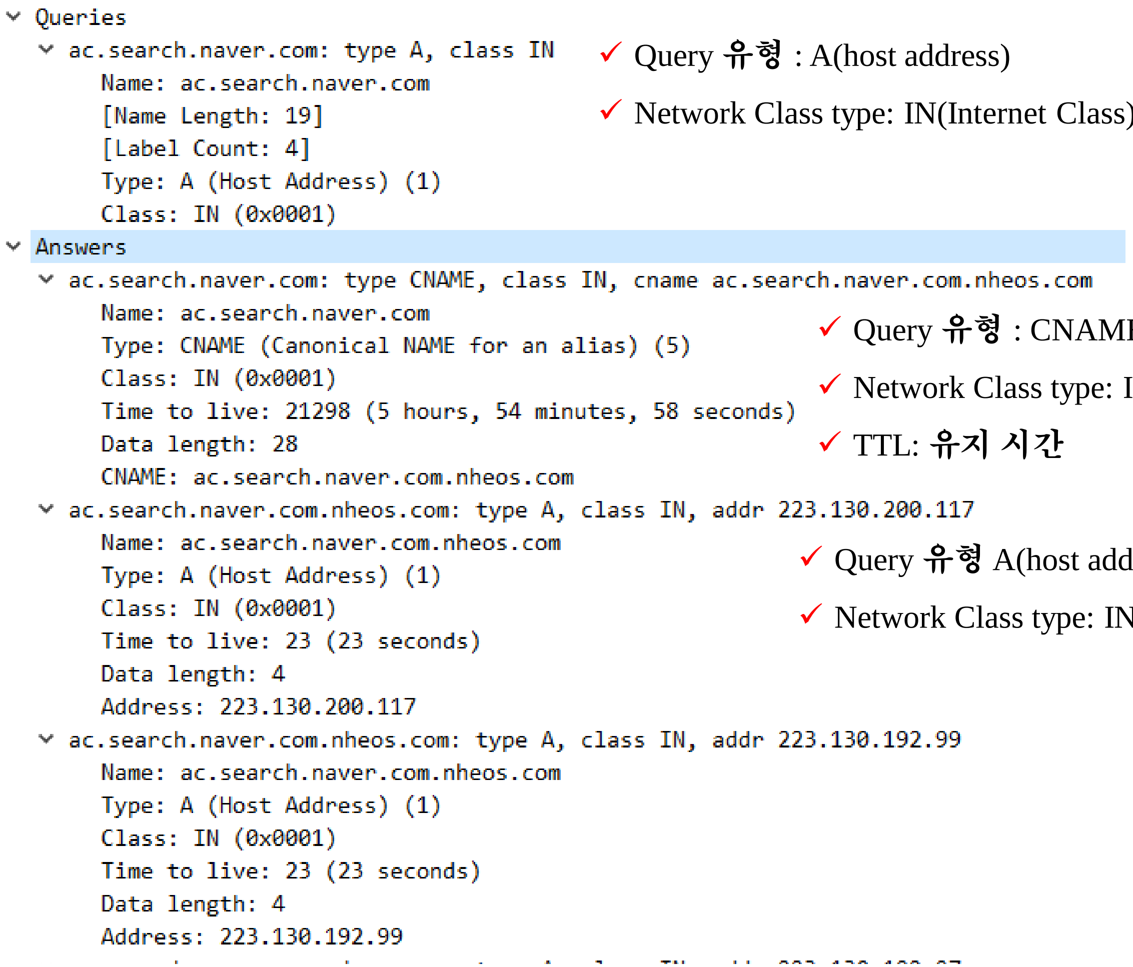
✓ 응답 5개

Query Message

Query Message 분석

DNS Query 분석

Queries & Answers



```
⌄ Queries
  ⌄ ac.search.naver.com: type A, class IN
      Name: ac.search.naver.com
      [Name Length: 19]
      [Label Count: 4]
      Type: A (Host Address) (1)
      Class: IN (0x0001)
⌄ Answers
  ⌄ ac.search.naver.com: type CNAME, class IN, cname ac.search.naver.com.nheos.com
      Name: ac.search.naver.com
      Type: CNAME (Canonical NAME for an alias) (5)
      Class: IN (0x0001)
      Time to live: 21298 (5 hours, 54 minutes, 58 seconds)
      Data length: 28
      CNAME: ac.search.naver.com.nheos.com
  ⌄ ac.search.naver.com.nheos.com: type A, class IN, addr 223.130.200.117
      Name: ac.search.naver.com.nheos.com
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 23 (23 seconds)
      Data length: 4
      Address: 223.130.200.117
  ⌄ ac.search.naver.com.nheos.com: type A, class IN, addr 223.130.192.99
      Name: ac.search.naver.com.nheos.com
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 23 (23 seconds)
      Data length: 4
      Address: 223.130.192.99
```

- ✓ Query 유형 : A(host address)
- ✓ Network Class type: IN(Internet Class)

- ✓ Query 유형 : CNAME(별칭의 정식 이름)
- ✓ Network Class type: IN(Internet Class)
- ✓ TTL: 유지 시간

- ✓ Query 유형 A(host address)
- ✓ Network Class type: IN(Internet Class)

Contents

1 Summary

2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

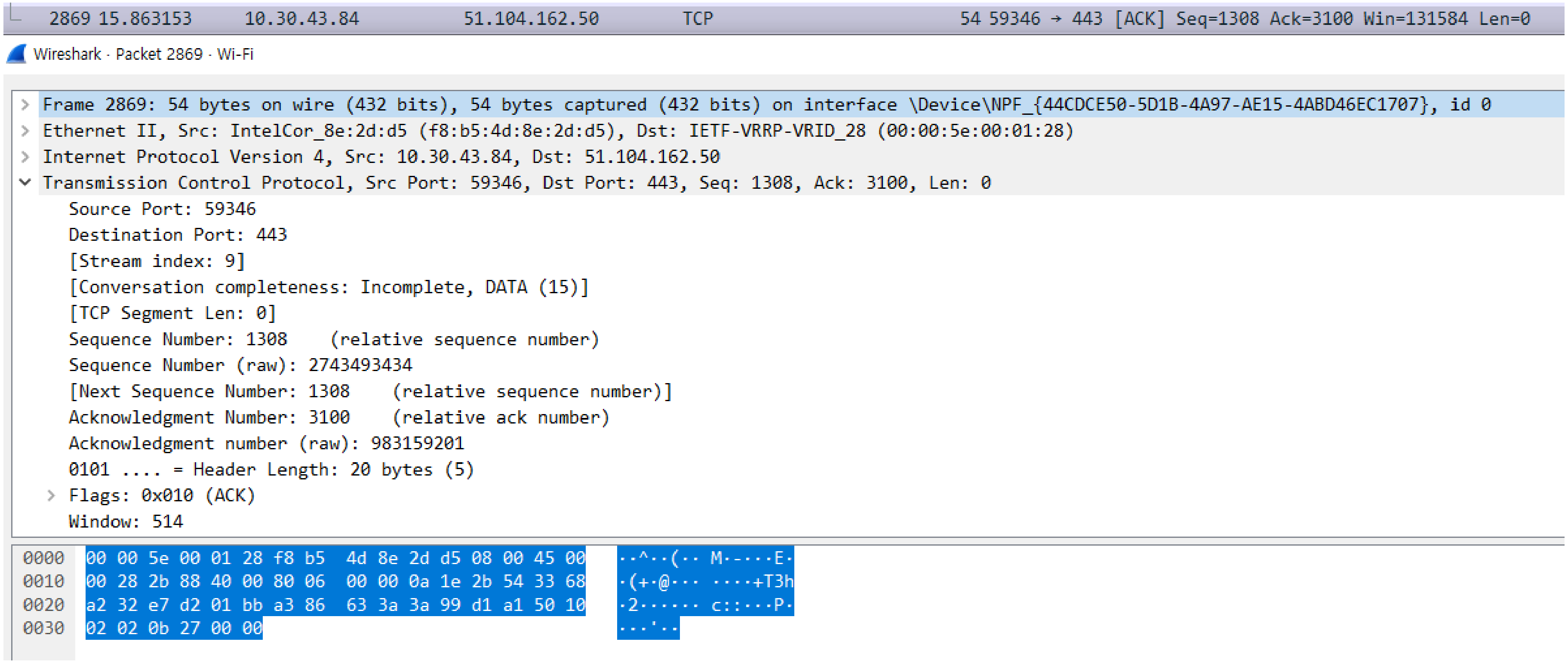
✓ 3 TCP 패킷 분석

2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

WireShark Capture

TCP Capture



```
2869 15.863153    10.30.43.84    51.104.162.50    TCP    54 59346 → 443 [ACK] Seq=1308 Ack=3100 Win=131584 Len=0
```

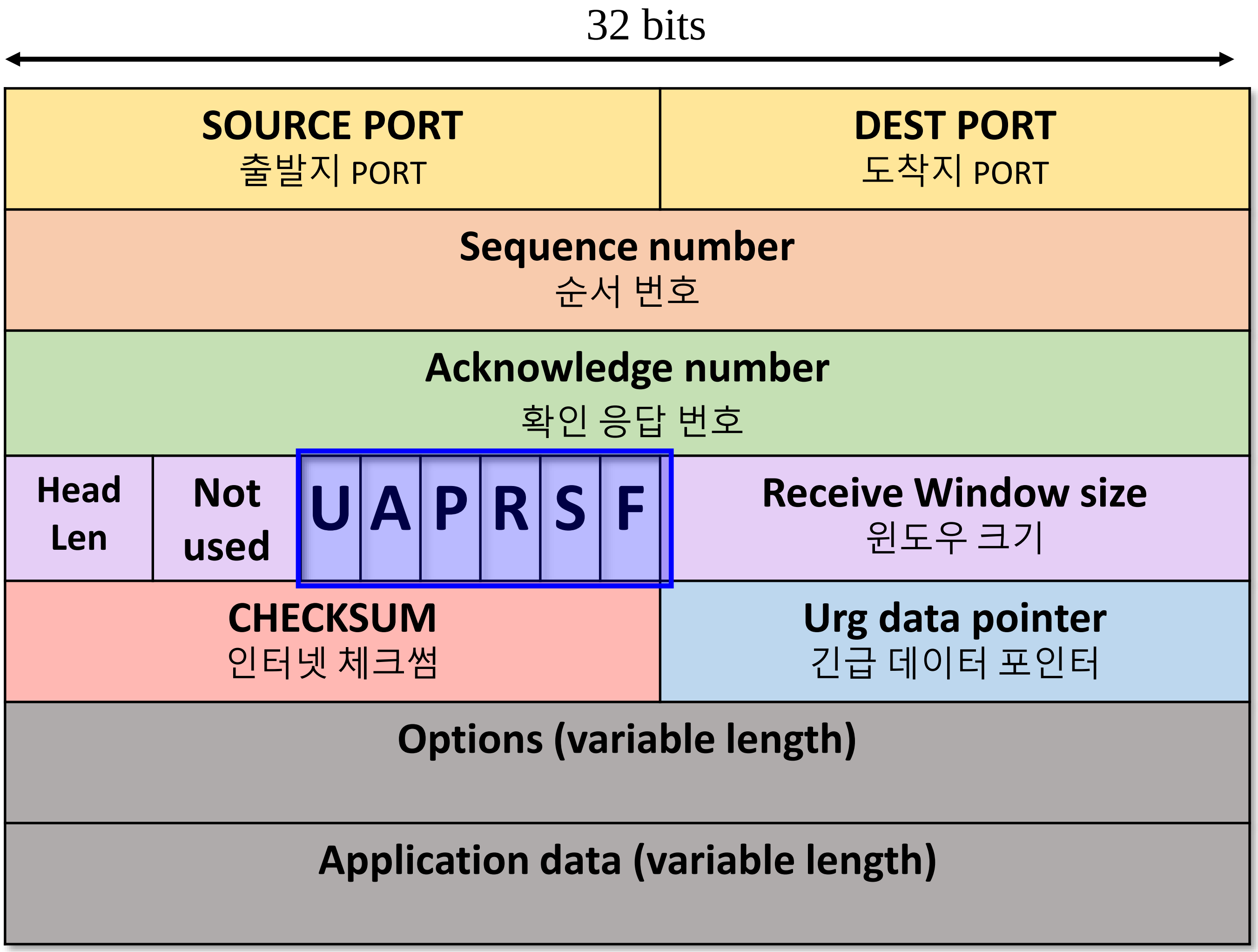
Wireshark · Packet 2869 · Wi-Fi

```
> Frame 2869: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{44CDCE50-5D1B-4A97-AE15-4ABD46EC1707}, id 0
> Ethernet II, Src: IntelCor_8e:2d:d5 (f8:b5:4d:8e:2d:d5), Dst: IETF-VRRP-VRID_28 (00:00:5e:00:01:28)
> Internet Protocol Version 4, Src: 10.30.43.84, Dst: 51.104.162.50
v Transmission Control Protocol, Src Port: 59346, Dst Port: 443, Seq: 1308, Ack: 3100, Len: 0
    Source Port: 59346
    Destination Port: 443
    [Stream index: 9]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 1308    (relative sequence number)
    Sequence Number (raw): 2743493434
    [Next Sequence Number: 1308    (relative sequence number)]
    Acknowledgment Number: 3100    (relative ack number)
    Acknowledgment number (raw): 983159201
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 514
```

```
0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
0020  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
0030  02 02 0b 27 00 00                                  ···'··
```

WireShark Capture

TCP segment structure



32 bits

SOURCE PORT
출발지 PORT

DEST PORT
도착지 PORT

Sequence number
순서 번호

Acknowledge number
확인 응답 번호

Head Len

Not used

U A P R S F

Receive Window size
윈도우 크기

CHECKSUM
인터넷 체크썸

Urg data pointer
긴급 데이터 포인터

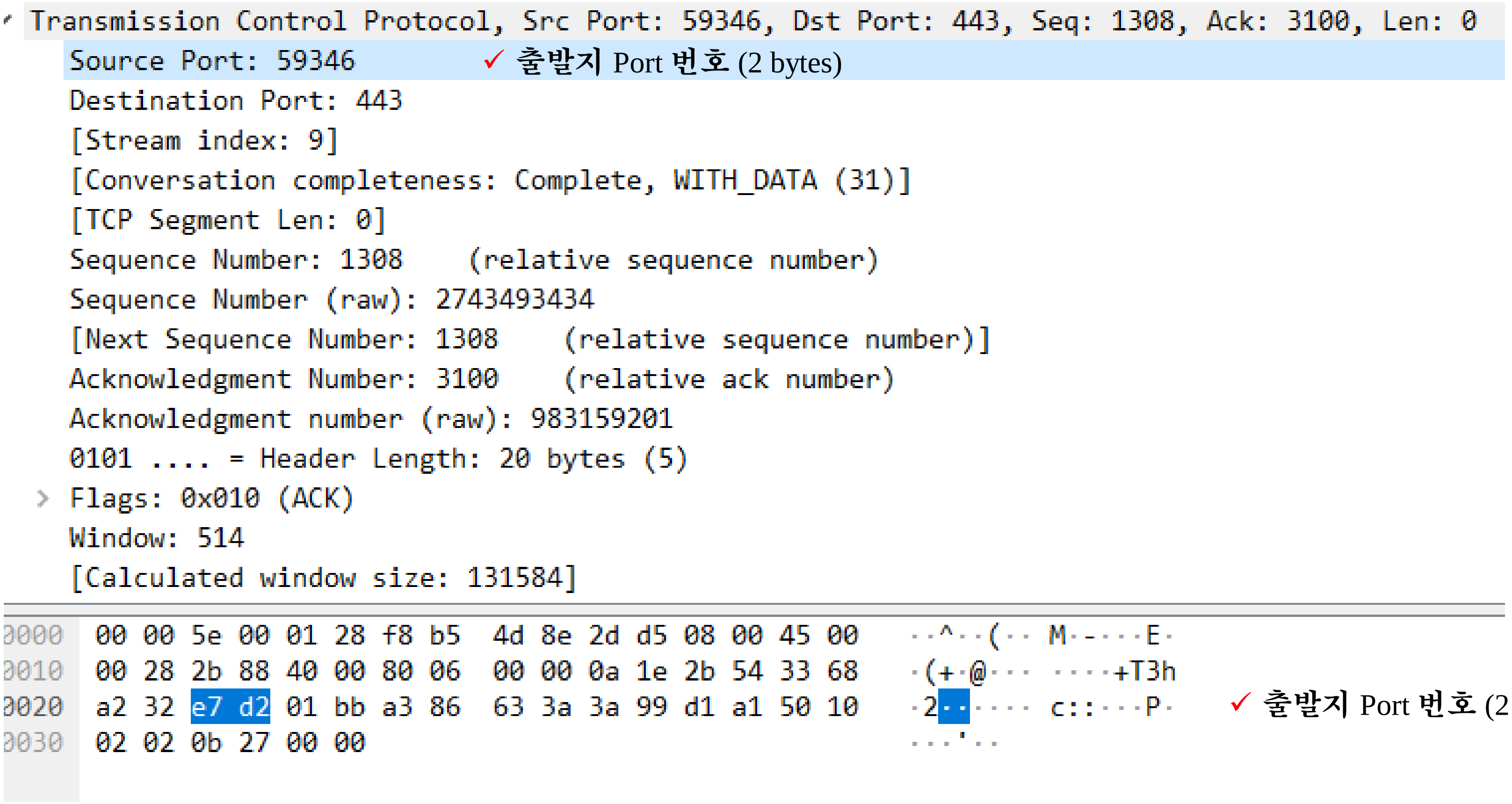
Options (variable length)

Application data (variable length)

🟦 URG: urgent data / ACK: ACK is valid / PSH: push data / RST, SYN, FIN: connection management

TCP Segment 분석

Source Port



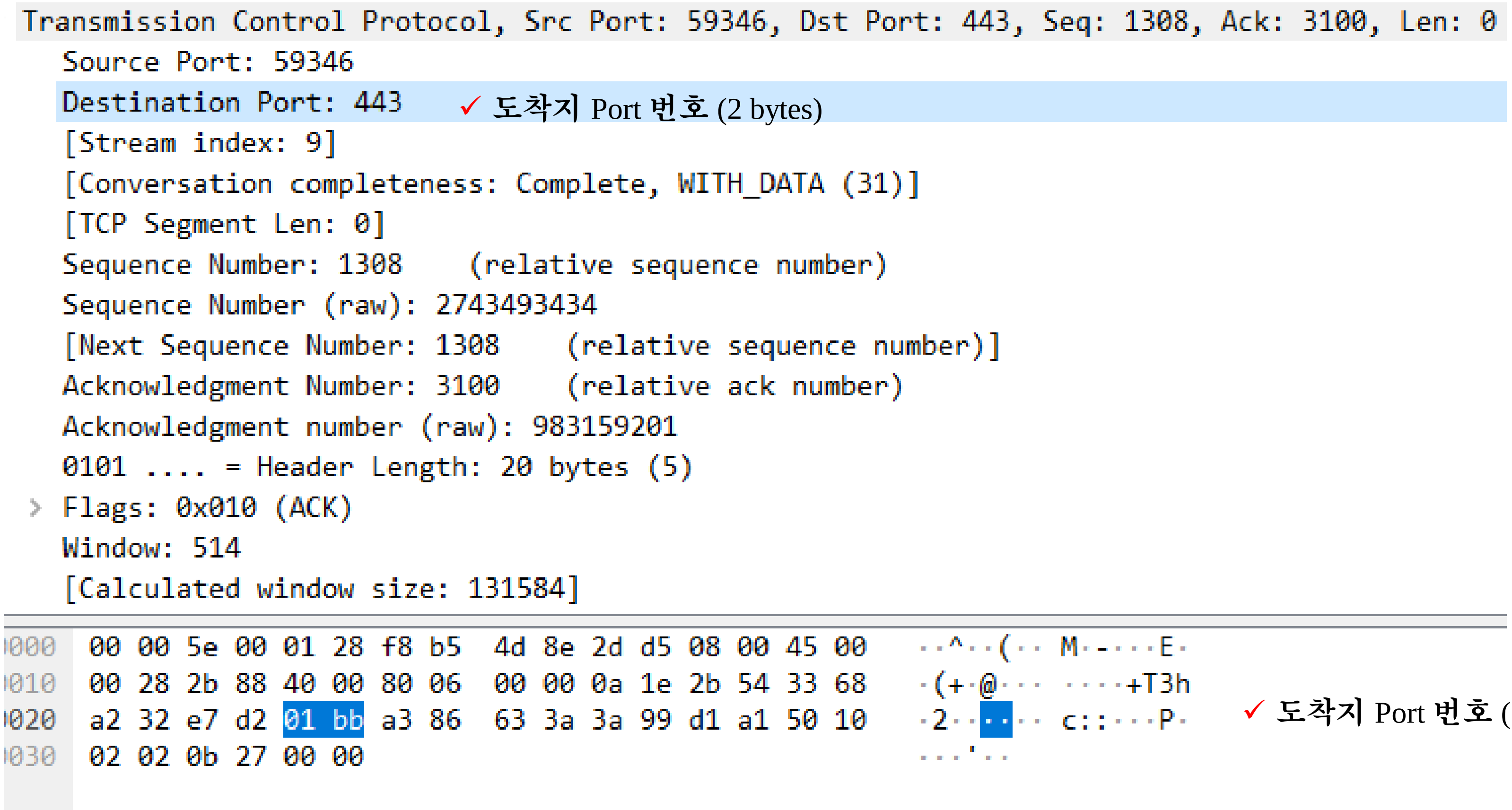
```
Transmission Control Protocol, Src Port: 59346, Dst Port: 443, Seq: 1308, Ack: 3100, Len: 0
    Source Port: 59346
    Destination Port: 443
    [Stream index: 9]
    [Conversation completeness: Complete, WITH_DATA (31)]
    [TCP Segment Len: 0]
    Sequence Number: 1308    (relative sequence number)
    Sequence Number (raw): 2743493434
    [Next Sequence Number: 1308    (relative sequence number)]
    Acknowledgment Number: 3100    (relative ack number)
    Acknowledgment number (raw): 983159201
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 514
    [Calculated window size: 131584]

0000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0010  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
0020  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
0030  02 02 0b 27 00 00                                  ···'··
```

✓ 출발지 Port 번호 (2 bytes)

✓ 출발지 Port 번호 (2 bytes)

TCP Segment 분석

Destination Port



```
Transmission Control Protocol, Src Port: 59346, Dst Port: 443, Seq: 1308, Ack: 3100, Len: 0
    Source Port: 59346
    Destination Port: 443
    [Stream index: 9]
    [Conversation completeness: Complete, WITH_DATA (31)]
    [TCP Segment Len: 0]
    Sequence Number: 1308    (relative sequence number)
    Sequence Number (raw): 2743493434
    [Next Sequence Number: 1308    (relative sequence number)]
    Acknowledgment Number: 3100    (relative ack number)
    Acknowledgment number (raw): 983159201
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 514
    [Calculated window size: 131584]
```

```
000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
010  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
020  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
030  02 02 0b 27 00 00                                  ···'··
```

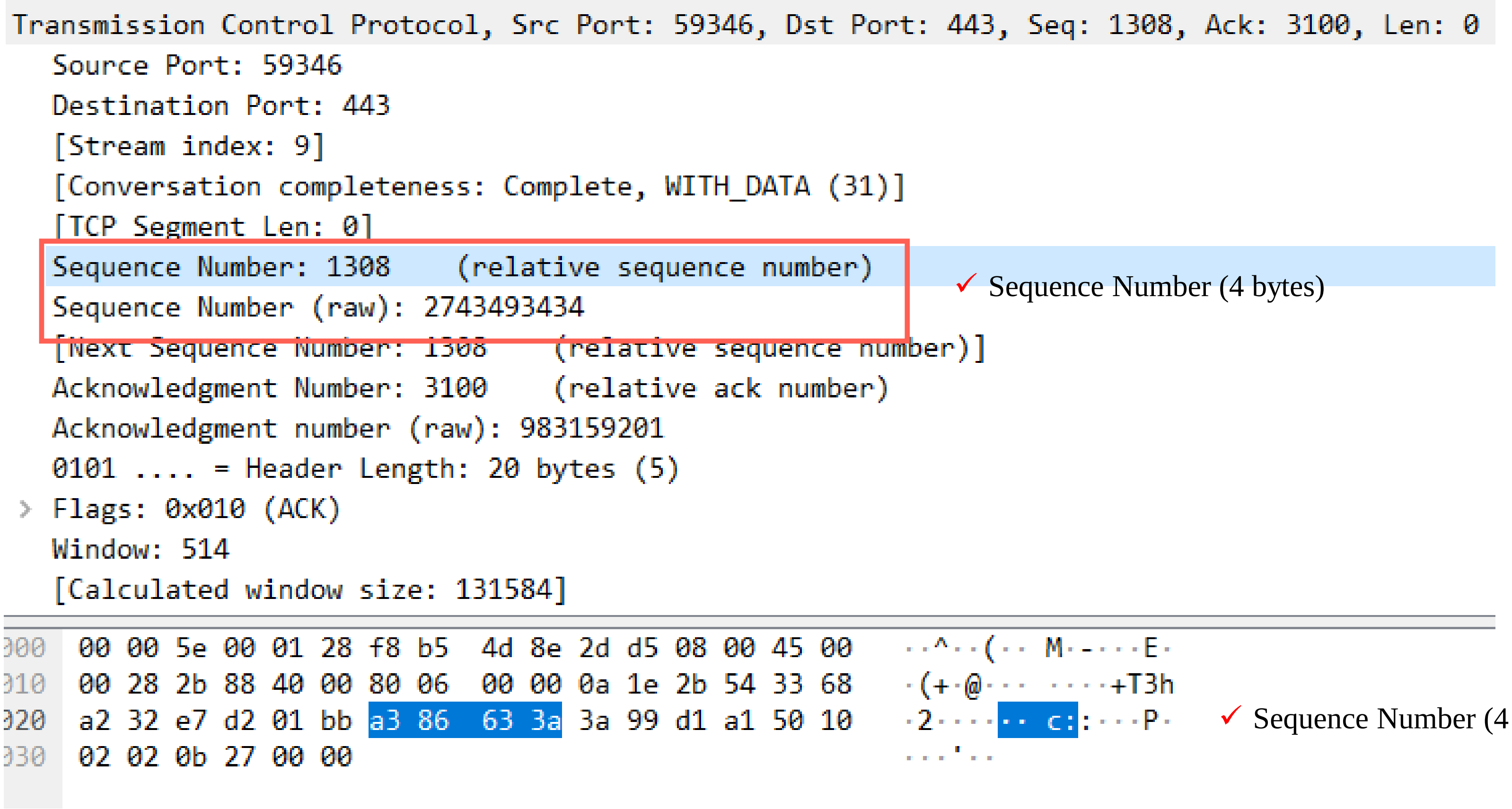
✓ 도착지 Port 번호 (2 bytes)

✓ 도착지 Port 번호 (2 bytes)

TCP Segment 분석

Sequence Number

- 순서 번호 필드 (Sequence number)
- 신뢰적인 데이터 전송 서비스 구현을 위해 사용된다



```
Transmission Control Protocol, Src Port: 59346, Dst Port: 443, Seq: 1308, Ack: 3100, Len: 0
    Source Port: 59346
    Destination Port: 443
    [Stream index: 9]
    [Conversation completeness: Complete, WITH_DATA (31)]
    [TCP Segment Len: 0]
    Sequence Number: 1308    (relative sequence number)
    Sequence Number (raw): 2743493434
    [Next Sequence Number: 1308    (relative sequence number)]
    Acknowledgment Number: 3100    (relative ack number)
    Acknowledgment number (raw): 983159201
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 514
    [Calculated window size: 131584]

000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
010  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
020  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
030  02 02 0b 27 00 00                                   ···'··
```

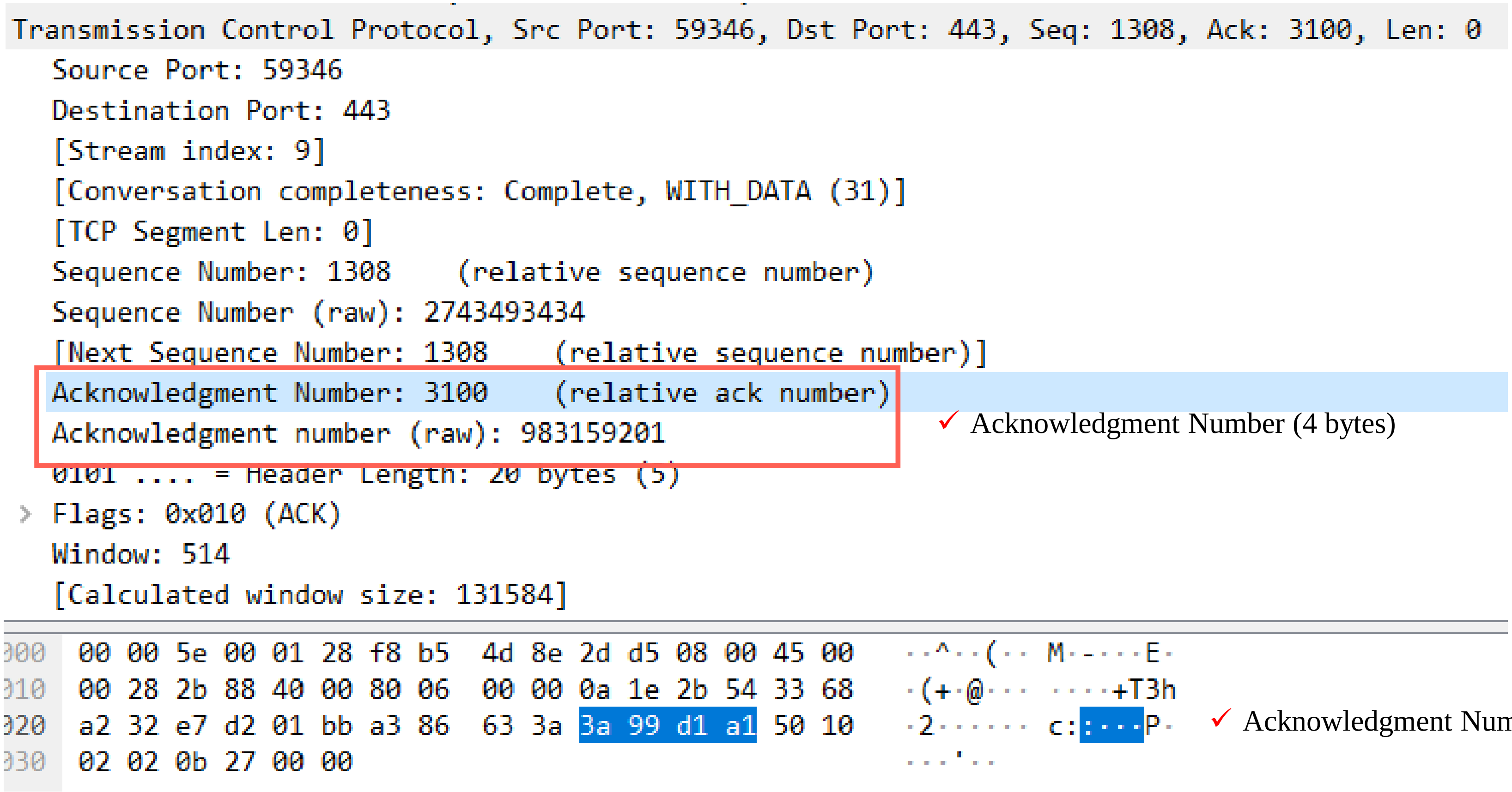
✓ Sequence Number (4 bytes)

✓ Sequence Number (4 bytes)

TCP Segment 분석

Acknowledge Number

- 확인 응답 번호 필드 (Acknowledge number)
- 신뢰적인 데이터 전송 서비스 구현을 위해 사용된다



```
Transmission Control Protocol, Src Port: 59346, Dst Port: 443, Seq: 1308, Ack: 3100, Len: 0
    Source Port: 59346
    Destination Port: 443
    [Stream index: 9]
    [Conversation completeness: Complete, WITH_DATA (31)]
    [TCP Segment Len: 0]
    Sequence Number: 1308    (relative sequence number)
    Sequence Number (raw): 2743493434
    [Next Sequence Number: 1308    (relative sequence number)]
    Acknowledgment Number: 3100    (relative ack number)
    Acknowledgment number (raw): 983159201
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window: 514
    [Calculated window size: 131584]

000  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
010  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
020  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
030  02 02 0b 27 00 00                                  ···'··
```

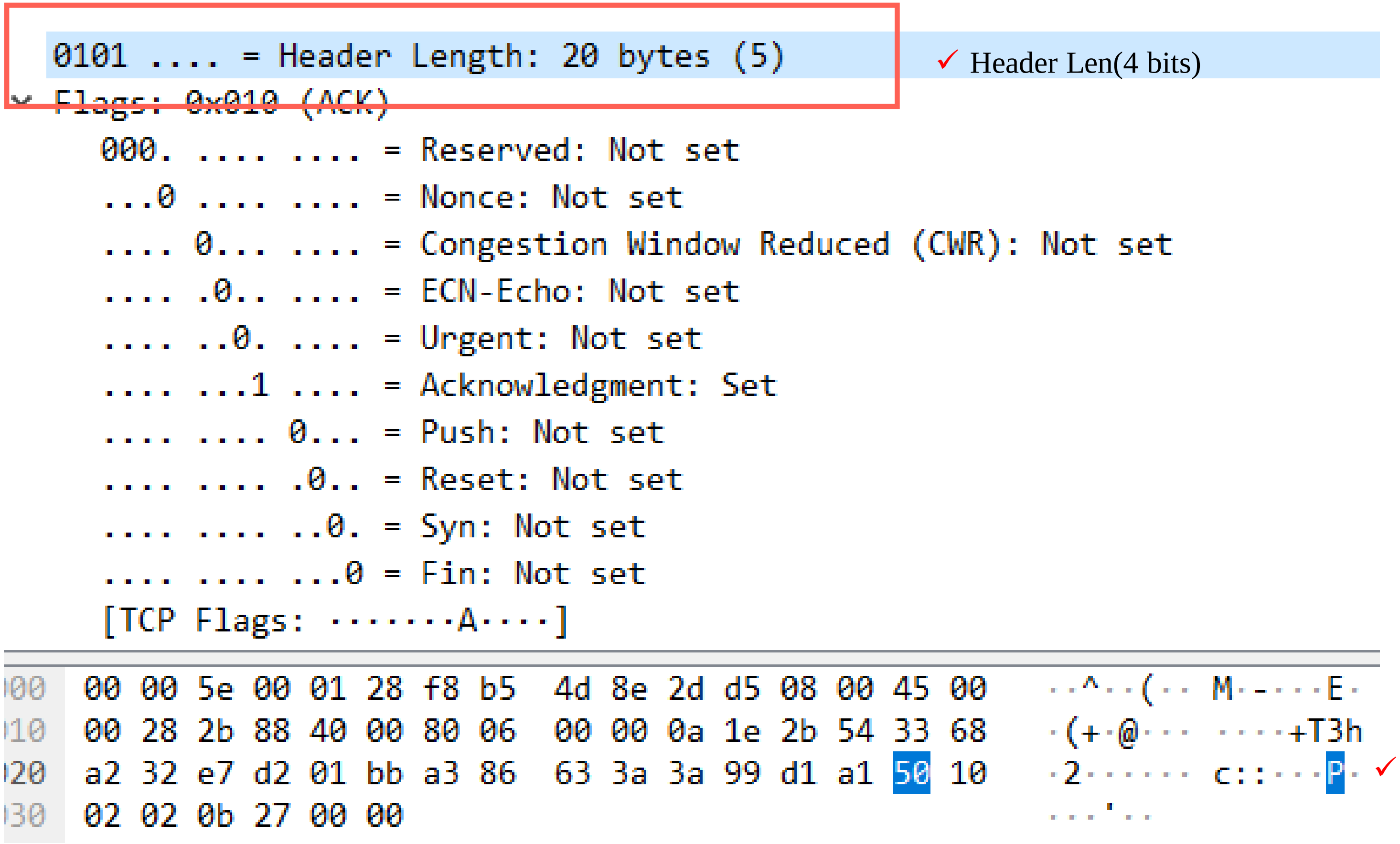
✓ Acknowledgment Number (4 bytes)

✓ Acknowledgment Number (4 bytes)

TCP Segment 분석

Header Len (=Hlen)

- 헤더 길이 (header line)
- 32비트 워드 단위로 TCP헤더의 길이를 나타낸다 (일반적으로 20바이트)



```
  0101 .... = Header Length: 20 bytes (5)
v 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····]
```

```
00  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
10  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
20  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
30  02 02 0b 27 00 00                                  ···'··
```

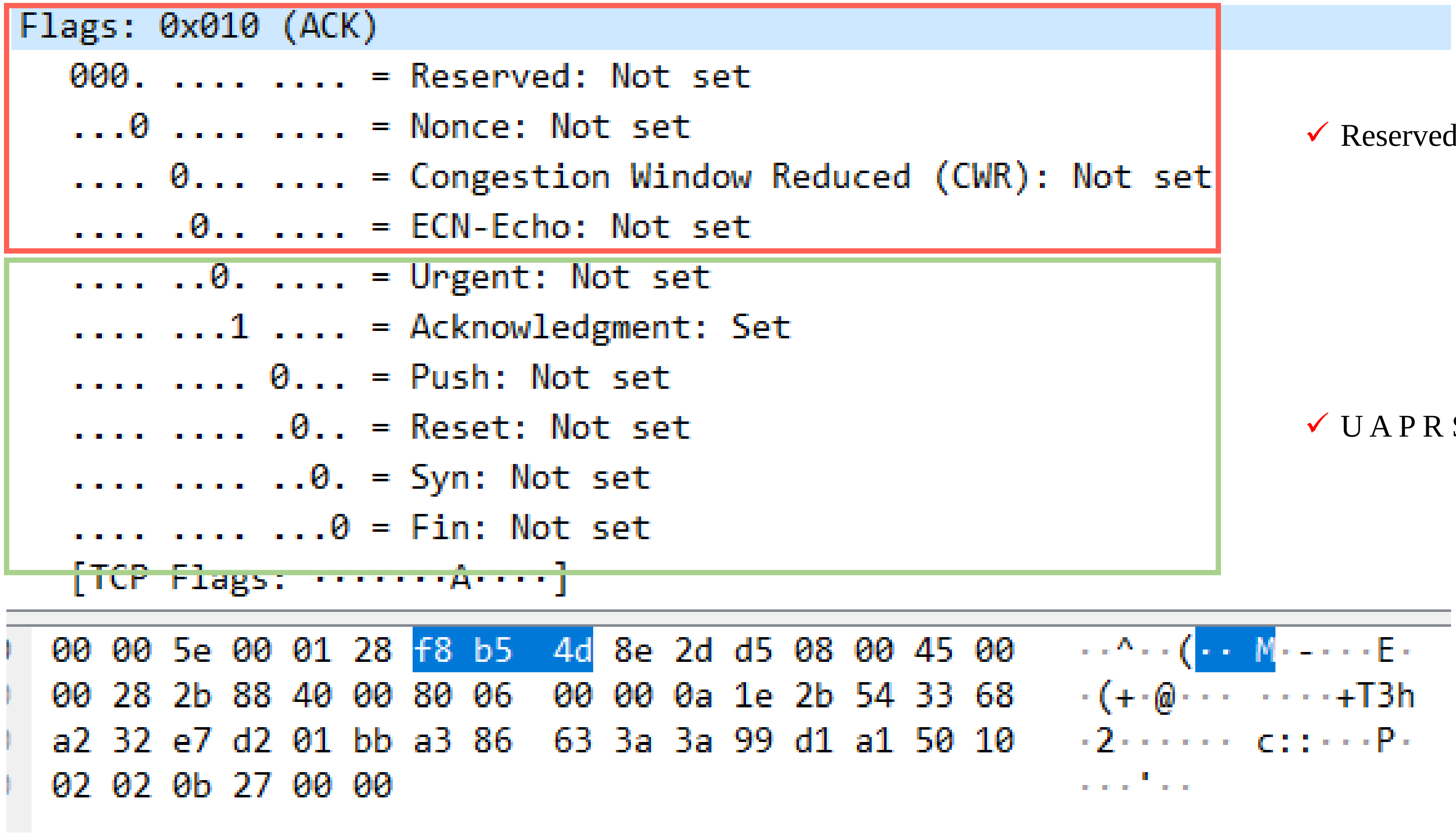
✓ Header Len(4 bits)

✓ Header Len(4 bits)

TCP Segment 분석

Reserved space & Flags

- Reserved space: 일반적으로는 사용하지 않으나, 혼잡 제어 시 일부 비트를 이용
- UAPRSF (FLAGS): 6비트를 통해 상태를 나타냄 (세부내용은 앞서 설명했으므로 생략)



```
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····]
```

```
00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@···  ····+T3h
a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······  c::···P·
02 02 0b 27 00 00                                  ···'··
```

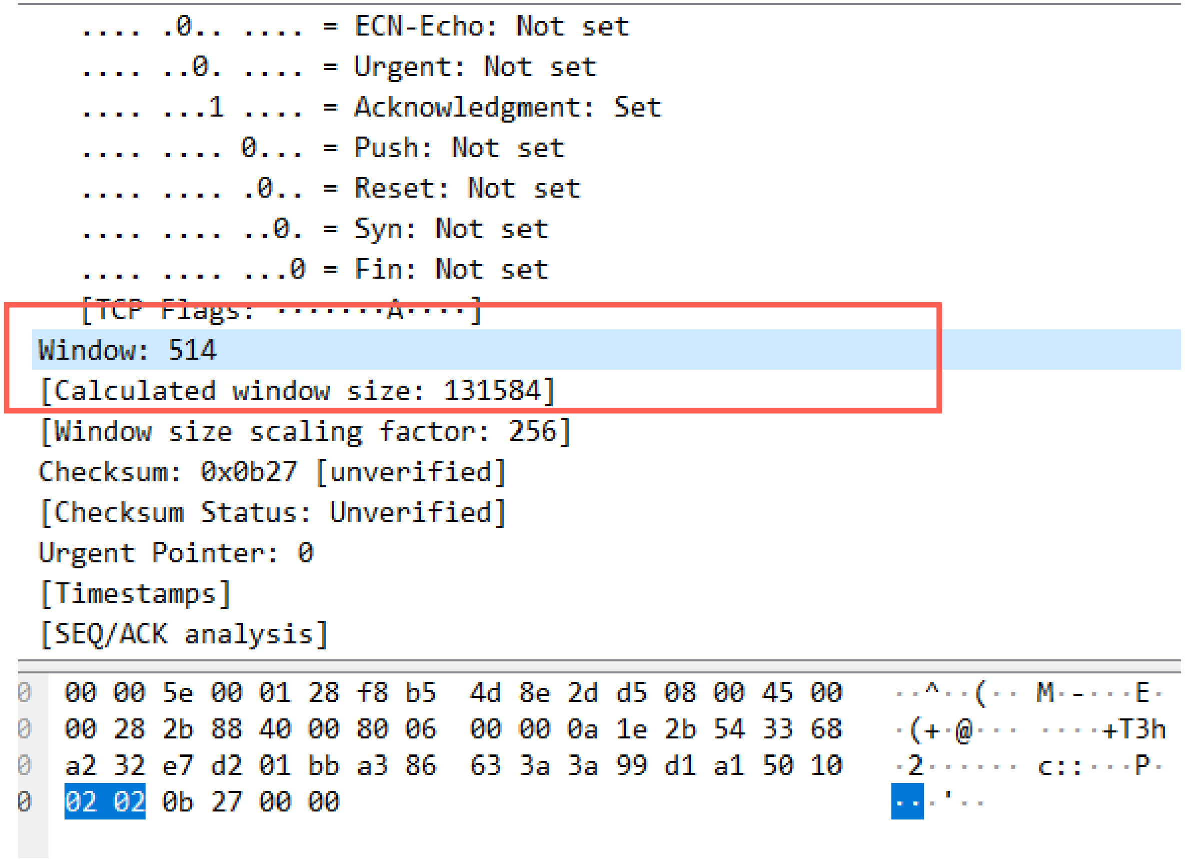
✓ Reserved space(6 bits)

✓ U A P R S F (6 bits)

TCP Segment 분석

Received Window size

- 수신 윈도우: 흐름제어에 사용된다. (Go-back-N, Selective repeat 등에 필요 정보)



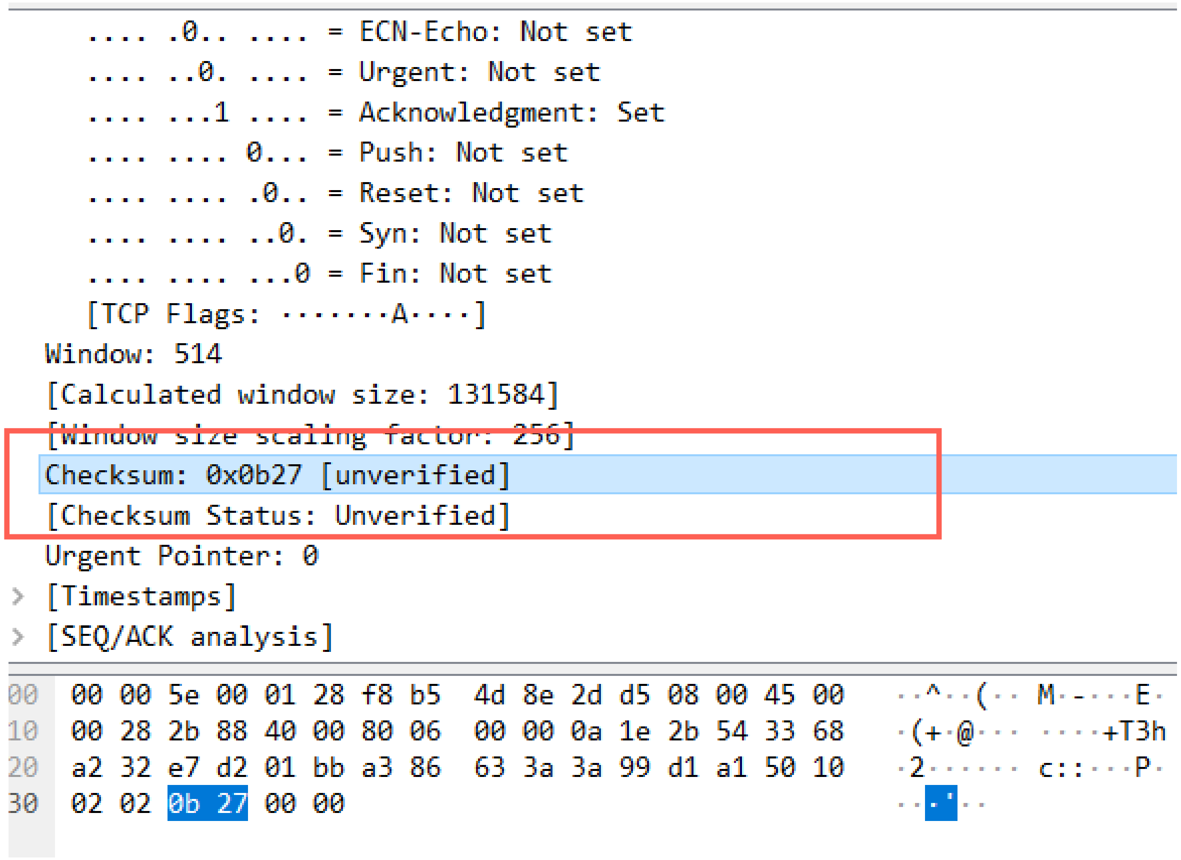
```
    .... .0.. .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...1 .... = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
    .... .... ...0 = Fin: Not set
    [TCP Flags: ·······A····]
  Window: 514
  [Calculated window size: 131584]
  [Window size scaling factor: 256]
  Checksum: 0x0b27 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
  [Timestamps]
  [SEQ/ACK analysis]

0  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
0  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
0  02 02 0b 27 00 00                                  ··'··
```

✓ Windows size(2 bytes)

✓ Windows size(2 bytes)

TCP Segment 분석

CheckSum



```
    .... .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: 514
  [Calculated window size: 131584]
  [Window size scaling factor: 256]
  Checksum: 0x0b27 [unverified]
  [Checksum Status: Unverified]
  Urgent Pointer: 0
> [Timestamps]
> [SEQ/ACK analysis]
```

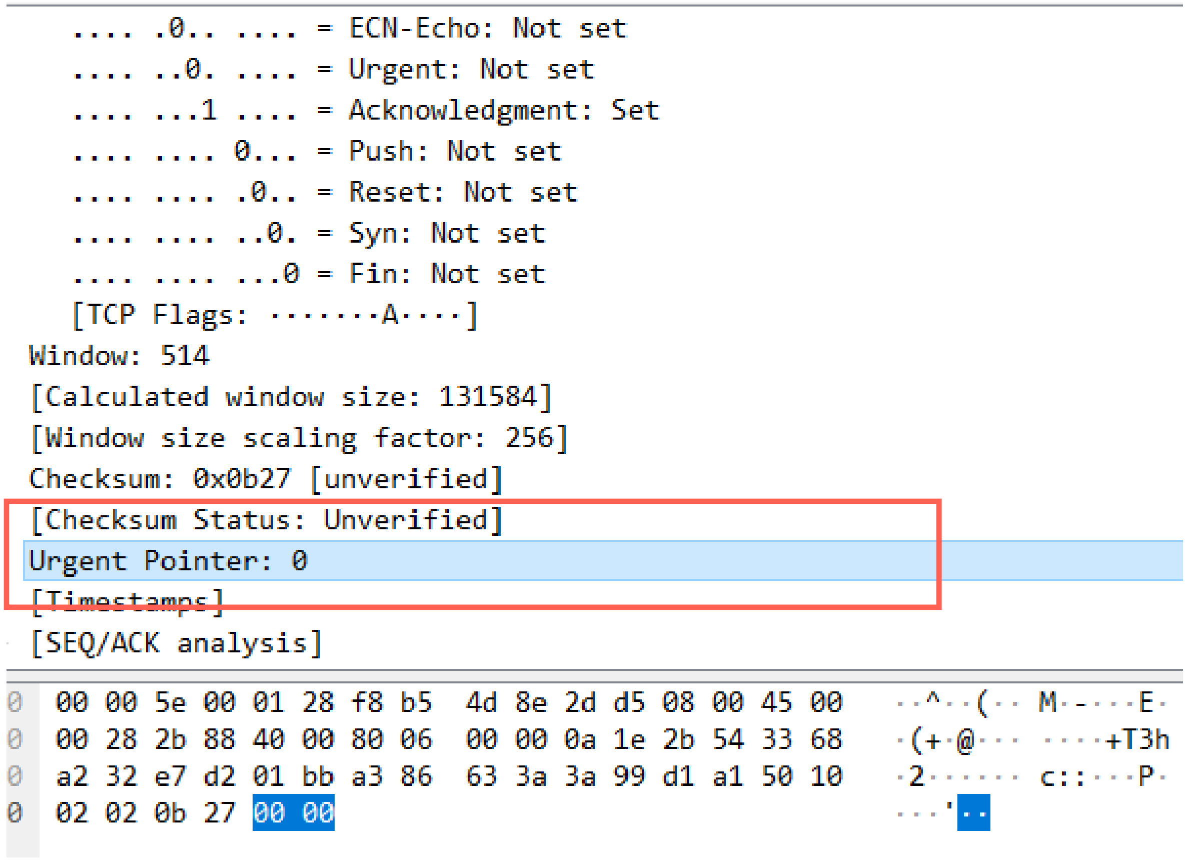
```
00  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
10  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
20  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
30  02 02 0b 27 00 00                                  ···'··
```

✓ Check Sum(2 bytes)

✓ Check Sum(2 bytes)

TCP Segment 분석

Urgent Pointer



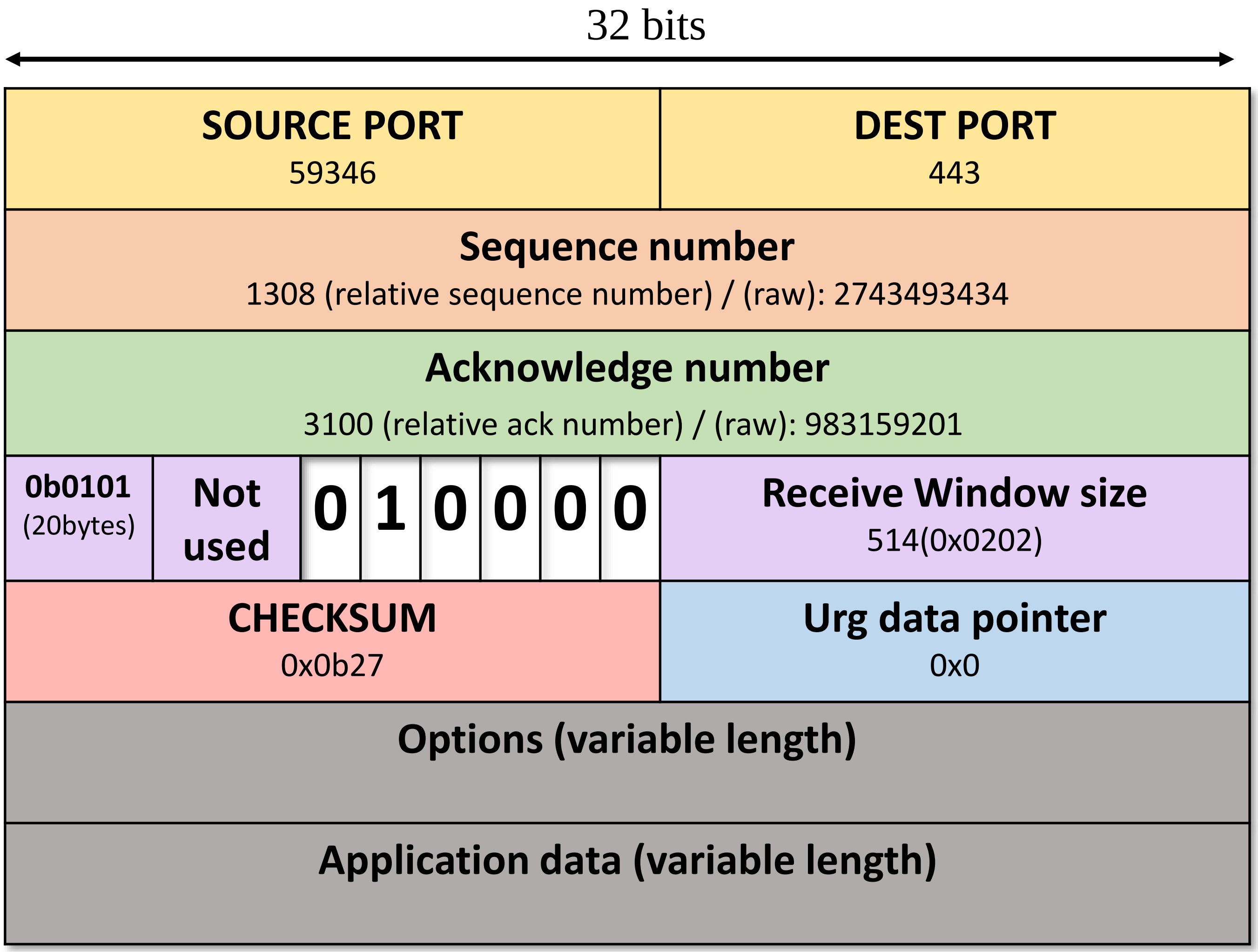
```
   .... .0.. .... = ECN-Echo: Not set
   .... ..0. .... = Urgent: Not set
   .... ...1 .... = Acknowledgment: Set
   .... .... 0... = Push: Not set
   .... .... .0.. = Reset: Not set
   .... .... ..0. = Syn: Not set
   .... .... ...0 = Fin: Not set
   [TCP Flags: ·······A····]
 Window: 514
 [Calculated window size: 131584]
 [Window size scaling factor: 256]
 Checksum: 0x0b27 [unverified]
 [Checksum Status: Unverified]
 Urgent Pointer: 0
 [Timestamps]
 [SEQ/ACK analysis]

0  00 00 5e 00 01 28 f8 b5  4d 8e 2d d5 08 00 45 00   ··^··(·· M·-···E·
0  00 28 2b 88 40 00 80 06  00 00 0a 1e 2b 54 33 68   ·(+·@··· ····+T3h
0  a2 32 e7 d2 01 bb a3 86  63 3a 3a 99 d1 a1 50 10   ·2······ c::···P·
0  02 02 0b 27 00 00                                  ···'··
```

✓ Urgent Pointer(2 bytes)

✓ Urgent Pointer(2 bytes)

WireShark Capture

TCP segment structure Analysis



32 bits

SOURCE PORT
59346

DEST PORT
443

Sequence number
1308 (relative sequence number) / (raw): 2743493434

Acknowledge number
3100 (relative ack number) / (raw): 983159201

0b0101
(20bytes)

Not used

0 1 0 0 0 0

Receive Window size
514(0x0202)

CHECKSUM
0x0b27

Urg data pointer
0x0

Options (variable length)

Application data (variable length)

Contents

1 Summary

2 DNS 패킷 분석

2.1 DNS query Message

2.2 DNS response Message

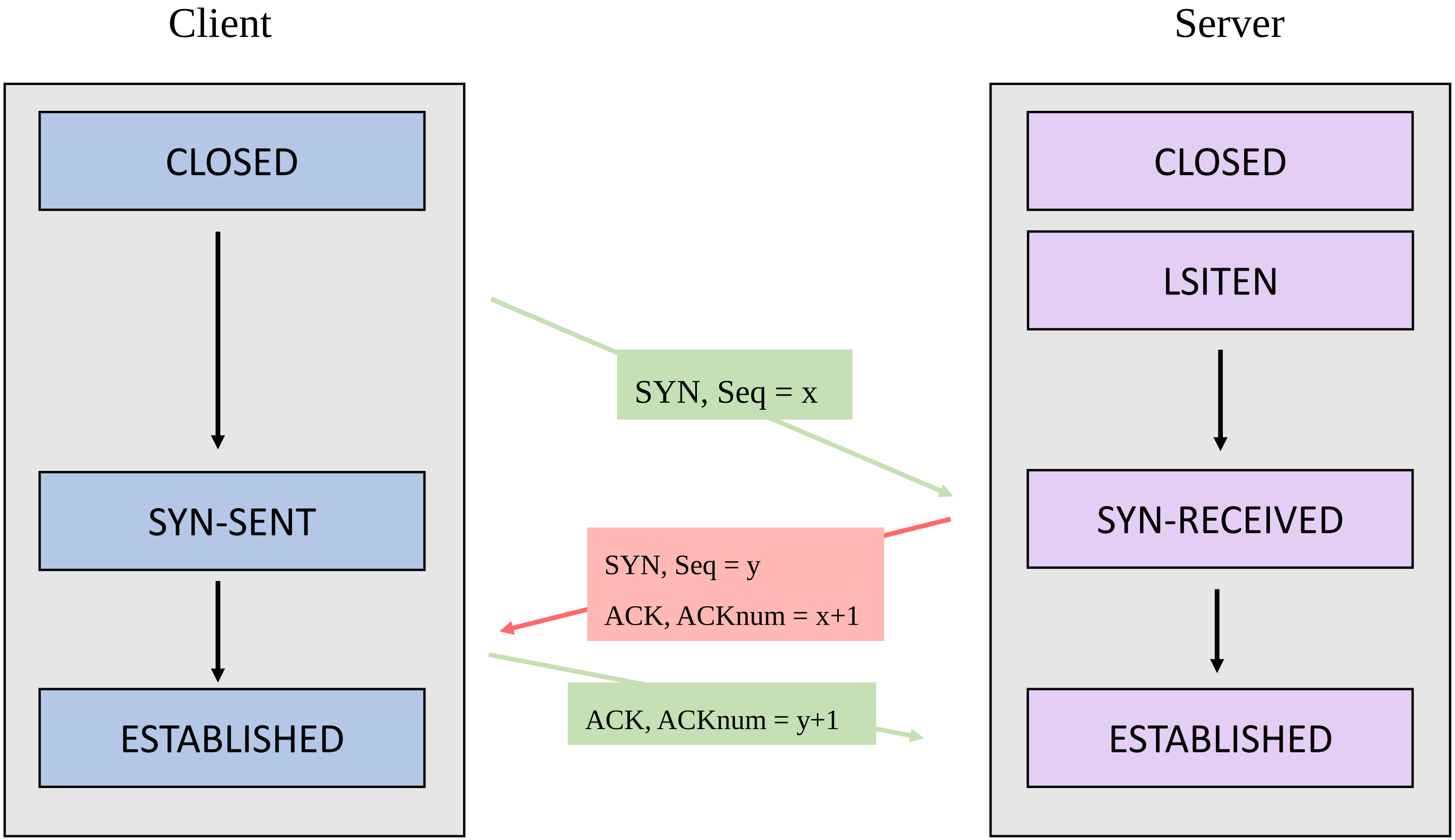
✓ 3 TCP 패킷 분석

2.2 TCP 패킷 개념 및 실습

2.2 중간 과제 code를 이용한 3-way handshake 연결

3-Way Handshake 연결 과정

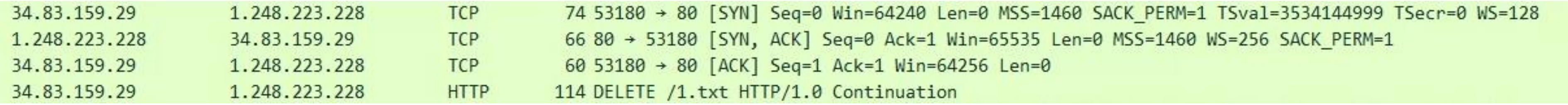
- Client-Server 간 준비가 되어있음을 보장하고, 실제로 data 전달 시작 전 상대가 준비되어 있음을 알 수 있다.



중간 과제 (HTTP) 코드를 통한 Wireshark Capture

HTTP 통신 이전에 TCP에서는 3-Way Handshake를 통해 연결

- Client IP 주소: 34.83.159.29
- Server IP 주소: 1.248.223.228



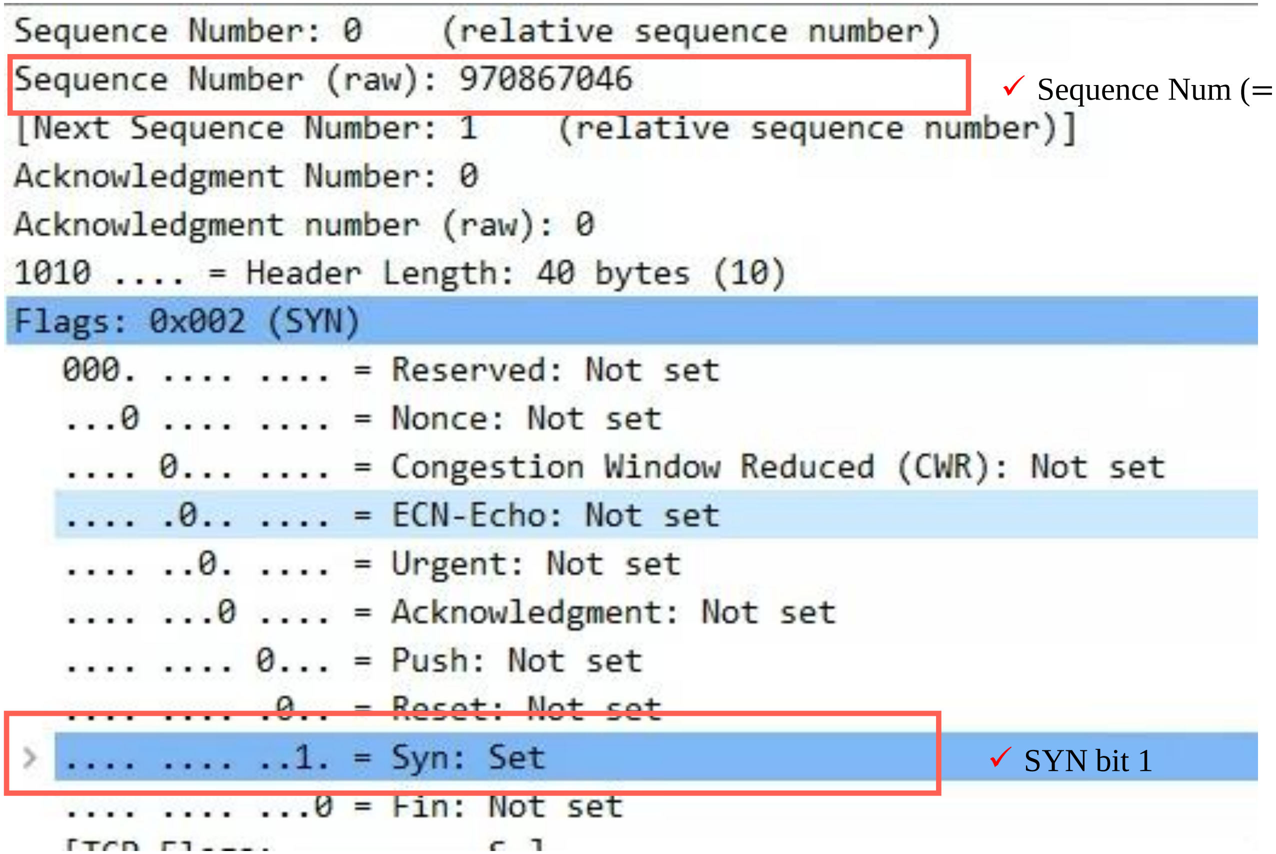
```
34.83.159.29     1.248.223.228    TCP      74 53180 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3534144999 TSecr=0 WS=128
1.248.223.228    34.83.159.29     TCP      66 80 → 53180 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
34.83.159.29     1.248.223.228    TCP      60 53180 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0
34.83.159.29     1.248.223.228    HTTP    114 DELETE /1.txt HTTP/1.0 Continuation
```

- Client >> Server: SYN 전송
- Server >> Client: SYN, ACK 전송
- Client >> Server: ACK 전송

중간 과제 (HTTP) 코드를 통한 Wireshark Capture

Client >> Server: SYN 전송

- Sequence Number: 0
- Flag- SYN bit: 1



```
Sequence Number: 0    (relative sequence number)
Sequence Number (raw): 970867046
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 0
Acknowledgment number (raw): 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
 > .... .... ..1. = Syn: Set
   .... .... ...0 = Fin: Not set
   [TCP Flags:
```

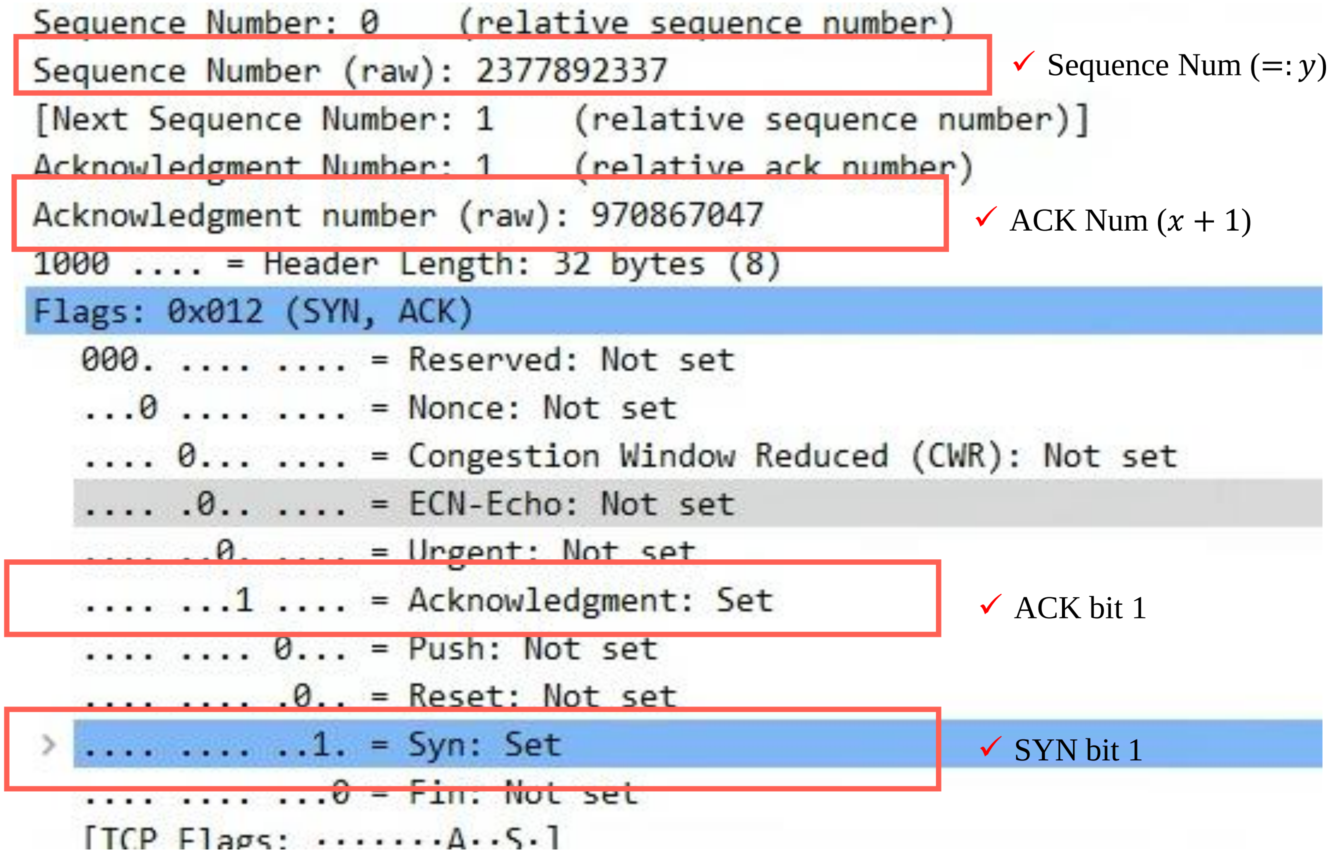
✓ Sequence Num (=: x)

✓ SYN bit 1

중간 과제 (HTTP) 코드를 통한 Wireshark Capture

■ Server >> Client: SYN, ACK 전송

- Sequence Number: 0
- Flag- SYN,ACK bit: 1



```
Sequence Number: 0    (relative sequence number)
Sequence Number (raw): 2377892337
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 1    (relative ack number)
Acknowledgment number (raw): 970867047
1000 .... = Header Length: 32 bytes (8)
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·]
```

✓ Sequence Num (=: y)

✓ ACK Num ($x + 1$)

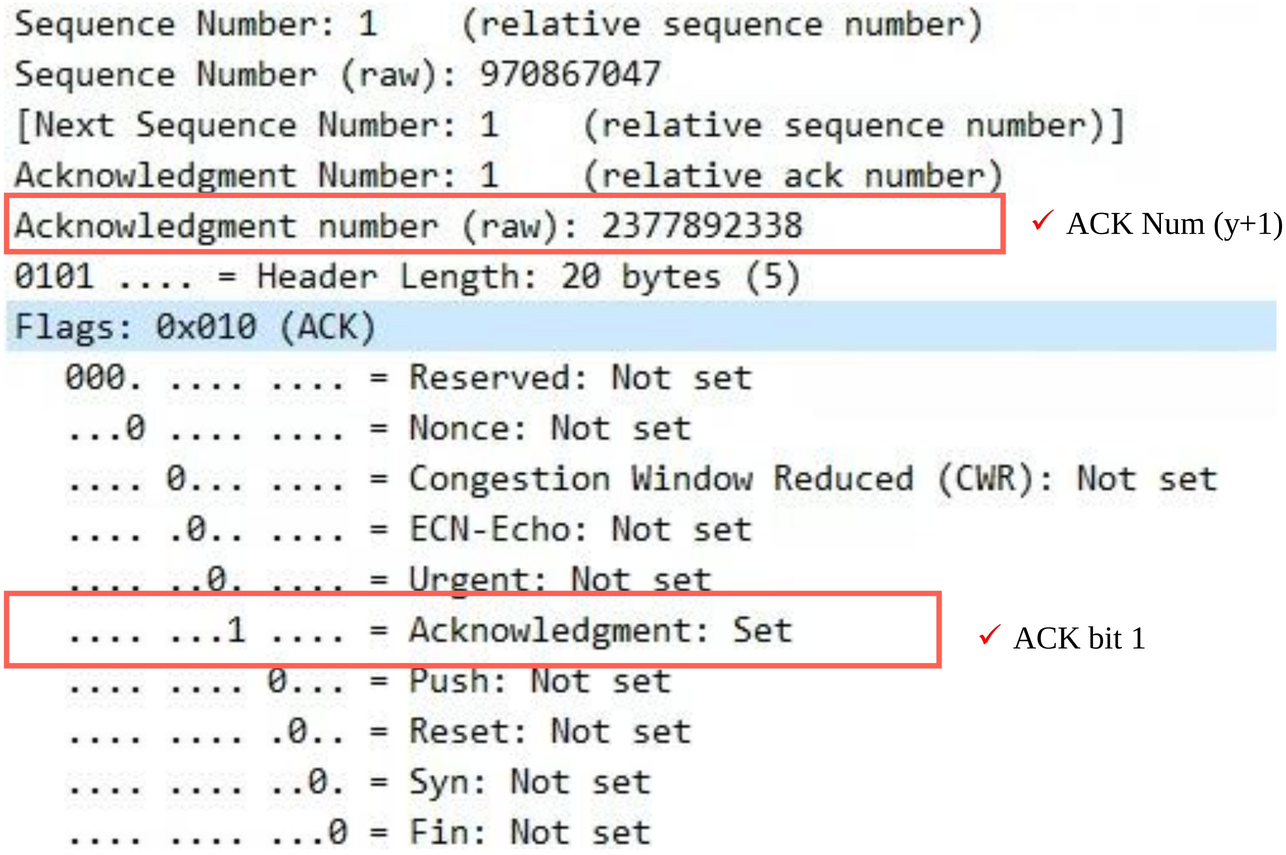
✓ ACK bit 1

✓ SYN bit 1

중간 과제 (HTTP) 코드를 통한 Wireshark Capture

Client >> Server: ACK 전송

- Sequence Number: 1
- Flag- ACK bit: 1



```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 970867047
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 1    (relative ack number)
Acknowledgment number (raw): 2377892338
0101 .... = Header Length: 20 bytes (5)
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
```

✓ ACK Num (y+1)

✓ ACK bit 1

Q&A

Email: easyboan@gmail.com