# REPORT

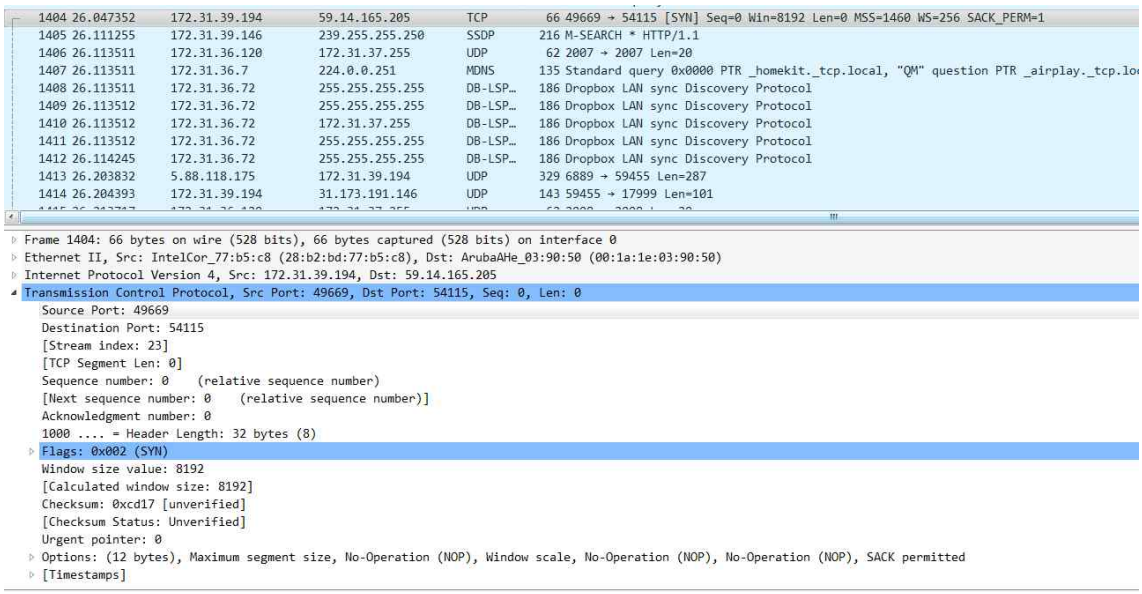

과 목 명 : 컴퓨터네트워크
담담교수 : 조경산 교수님
소　　속 : 소프트웨어학과
학　　번 : 32151671
이　　름 : 박민혁


단국대학교
Dankook University

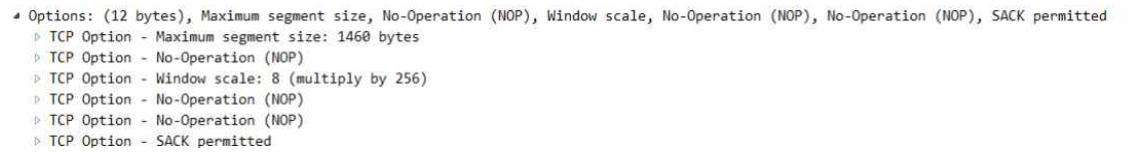
# Computer Network(fourth Homework)

## 1. Capture various Ethernet frames using Wireshark, and explain fields in the Ethernet, IP, and TCP header of SYN packet(Explain fields in the TCP header) and FIN packet.

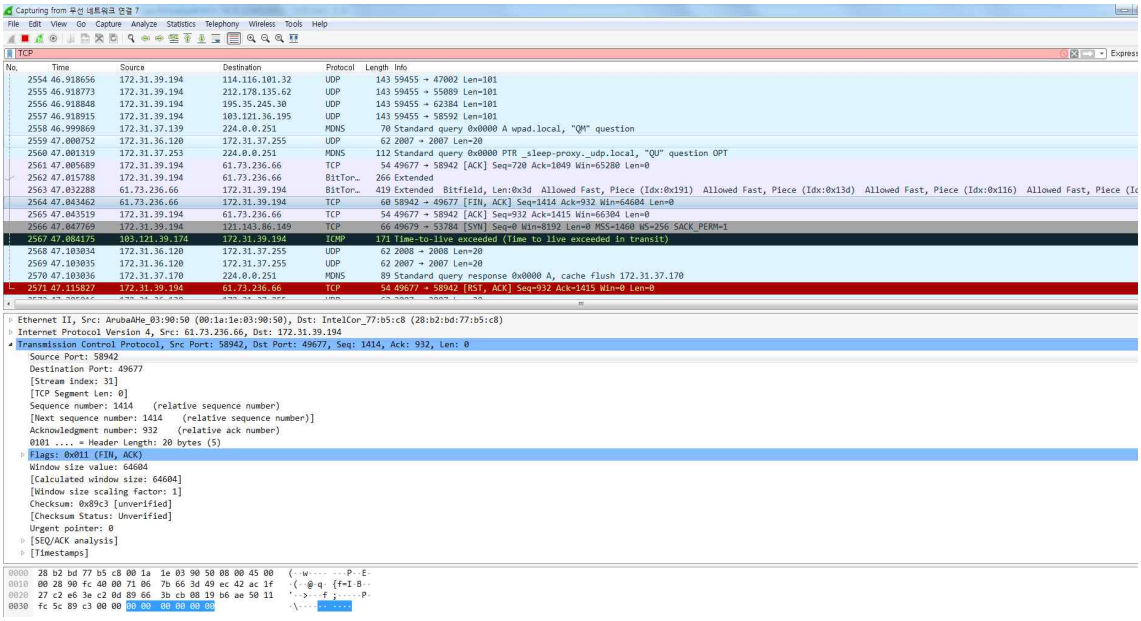
### (1). SYN




- Source Port : 49669
- Destination Port : 54115
- Sequence Number : 0
- Acknowledgment Number : 0
- Flags : SYN
- Window Size value : 8192
- Checksum : 0xcd17
- Options : 12Byte (MSS, WS)



### (2). FIN




- Source Port : 58942
- Destination Port : 49677
- Sequence Number : 1414
- Acknowledgment Number : 932
- Header Length : 20byte
- Flags : FIN, ACK
- Window Size Value : 64604
- Checksum : 0x89c3

## 2. Explain how we can get reliable transfer through unreliable protocol UDP.

- Application program에서 내부에 데이터 순서 번호기능이 구현되어야 한다.
- 데이터 재전송 기능이 구현되어야 한다.
- 또한 속도가 서로 다른 컴퓨터에서 실행되는 Application이 UDP를 이용하여 대량의 데이터를 주고 받으려면 흐름 제어 기능까지 구현해야 한다.

## 3. The following is a dump of a UDP header; 0045DF0000580000

### 1) Is the packet directed from a client to a server or vice versa?

- 송신자 Port Number가 00000000이고, 수신자 Port Number가 01000101이므로 이 packet은 처음 booting하는 내 컴퓨터가 DHCP server에게 보내는 packet이다. 즉, client가 server에게 보내는 packet이다.

### 2) What is the length of the data?

- 전체 packet의 크기가 223Byte(DF)이고 header의 크기는 8byte이므로 data 길이는 215byte이다.

**3) How the sender handled checksum for this packet?**

- sender는 UDP header 앞에 pseudo header를 붙인 후 pseudo header까지 합쳐서 보낸다.

## 4. In TCP, how many sequence numbers are consumed by each segments?

**a. SYN**

- N이 이전 packet의 header 크기이고, SYN은 이전 packet의 payload가 없으므로 SYN을 보낼 때 sequence number은 N+1이 된다.

**b. ACK**

- N이 이전 packet의 header 크기이고, A가 이전 packet의 payload 크기일 때, ACK를 보낼 때 sequence number는 N+A가 된다.

**c. FIN,ACK**

- FIN+ACK를 보낼 때 sequence number는 N+A가 된다.

**d. PSH,ACK(data-100bytes)**

- N이 이전 packet의 header 크기라 하고, A가 이전 packet의 payload 크기라고 하면, sequence number는 이전 packet의 크기를 통해 알 수 있으므로 ACK+data(100bytes)의 sequence number는 N+A가 된다.

## 5. The intruder sends a SYN segment to the server using 철수's IP address. Can the intruder create a TCP connection with the server by pretending that he is 철수? Assume that the server uses 1) a different ISN(Initial Sequence Number) for each connection or 2) the same ISN for each connection.

-ISN을 같은 숫자로 하게 된다면 공격자가 송신자를 spoofing 할 수 있다. 이 때 spoofing은 공격자가 원래 송신자의 IP Address를 사용하여 packet을 보내는 것을 의미한다. 그렇게 되면 수신자는 SYN+ACK packet을 송신자의 IP Address로 보내게 된다. 이 때 ISN이 random number라면 공격자는 SYN+ACK에 대한 대답을 보낼 수 없지만 random number가 아니라면 공격자가 SYN+ACK sequential number를 예측할 수 있게 되고 SYN+ACK의 sequential number+1을 갖는 packet을 보낼 수 있게 된다.

## 6. Following is the output from netstat command.

| Proto | Local Address | Foreign Address | State |
|---|---|---|---|
| TCP | 192.13.201.215:1059 | 0.0.0.0:0 | LISTENING |
| TCP | 192.13.201.215:61032 | 211.234.249.226:1524 | TIME_WAIT |

| | | | |
|---|---|---|---|
| **TCP** | **192.13.201.215:62029** | **211.233.16.71:80** | **ESTABLISHED** |

**1) Explain the values of state – LISTENING, ESTABLISHED, TIME_WAIT.**

- LISTENING은 요청을 받을 수 있도록 연결 요구를 기다리는 상태를 의미한다. 즉, 포트가 열려 있는 상태이다.
- ESTABLISHED는 서로 연결중인 상태를 의미한다.
- TIME_WAIT는 연결은 종료 되었으나 원격의 수신 보장을 위해 기다리고 있는 상태를 의미한다.

**2) Is 192.13.201.215:1059 server or client?**

- Server이다.

**3) Explain "219.240.16.226:80" in Foreign Address in two parts.**

- Foreign Address의 "219.240.16.226:80"은 나와 Network로 연결이 이루어지고 있는 System Address이다.

**7. An HTTP clients opens a TCP connection using an ISN of 100 and port number of 50,000. The server opens the connection with an ISN of 200. If the client defines receive buffer of 1024 and the server defines receive buffer of 4096, show the header of $2^{nd}$ segment during the connection establishment. Ignore the calculation of checksum field.**

- SOURCE PORT : 송신자 Address
- DESTINATION PORT : 수신자 Address
- SEQUENCE NUMBER : 100
- ACKNOWLEDGEMENT NUMBER : 100+100=200
- HLEN : 5 (20Byte)
- CODE BITS : ACK와 PSH만 1, 나머지는 0
- WINDOW : 1024
-

**8. Explain flow control, congestion control, and SYN flooding.**

- Flow Control : 흐름제어 ( 송신자가 보내는 데이터 양 제어, 수신자가 받을 수 있는 만큼)
- Congestion Control : SEQUENCE NUMBER -> 똑같은 packet이 2번 오거나 순서가 뒤죽박죽인 것을 해결한다.
- SYN Flooding : 존재하지 않는 Client가 Server별로 한정되어 있는 접속 가능한 공간에 접속한 것처럼 속여, 다른 사용자가 Server의 Service를 제공받지 못하게 하는 것이다.

**9. Compare the TCP header and the UDP header. List the fields in the TCP header that are not part of the UDP header, and list the fields in the UDP header that are not part of the TCP header. Give the reason for each missing field.**

- TCP header에는 있지만 UDP header에는 없는 부분
-> SEQUENCE NUMBER, ACKNOWLEDGEMENT NUMBER : UDP는 Application에서 보낸 data를 byte 단위로 끊지 않고 그대로 보내기 때문에 packet 간의 순서를 함께 보낼 필요가 없다. 그리고 TCP는 신뢰성을 제공하기 위해 SEQUENCE NUMBER를 사용하는데 UDP는 신뢰성을 제공하지 않는다.
-> Header Length : TCP는 Header Length가 option에 따라 변할 수 있지만 (20Byte에서 4Byte 단위로 증가) UDP는 8Byte로 고정되어 있기 때문에 header 길이를 쓰지 않는다.
-> CODE BITS : CODE BITS는 TCP가 packet을 보낼 때 WINDOW, SEQUENCE NUMBER, SYN인지 FIN인지 등 추가적인 정보를 제공하기 위해서 필요한 bit들인데 UDP는 위 내용들을 포함하고 있지 않으므로 CODE BITS가 필요 없다.
-> WINDOW : TCP는 신뢰성을 제공하기 위해 Congestion Control과 Flow Control을 하기 때문에 WINDOW를 표시해준다.
-> URGENT POINTER : Urgent한 data가 있으면 먼저 해결해 주기 위해 필요하다.

- UDP header에는 있지만 TCP header에는 없는 부분
-> UDP MESSAGE LENGTH : TCP는 SEQUENCE NUMBER에 이전 data의 크기를 알 수 있으므로 따로 적지 않고 header 크기만 표시해준다.
-> Pseudo Header : UDP의 header에는 꼭 필요한 부분만 있어서 추가적인 정보를 제공하기 위해서 Pseudo Header를 사용한다. 이 header는 수신자에게 보내지는 않고 정보만 제공해준다.

## 10. Which of UDP and TCP is better for the communication between DNS server and client. (consists of two packets – DNS request, DNS reply)

- DNS 사용 PORT는 일반적으로 UDP, Message 정보가 512Byte보다 클 경우에는 TCP를 사용한다.