Chair of Distributed Systems and Security
School of Computation, Information and Technology
Technical University of Munich

Computer Networking and IT Security (INHN0012)

Tutorial 9

Problem 1 Wireshark

Given is the hexdump in Figure 1.1 in network byte order of an Ethernet frame without checksum, which is to be analyzed in the following.

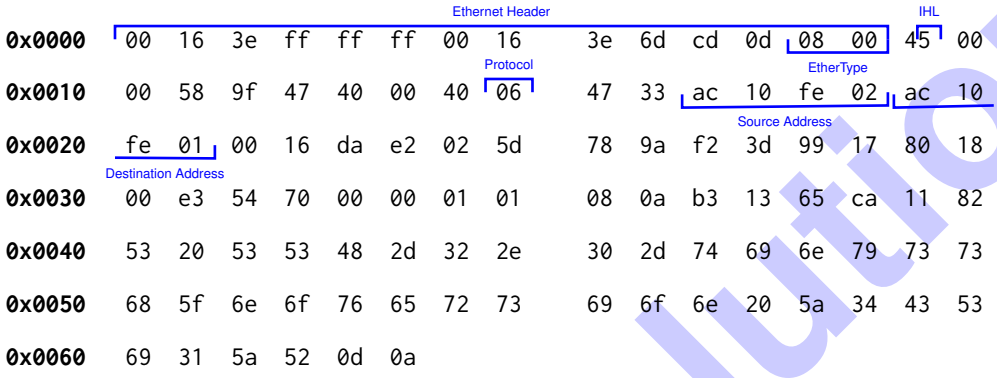

Figure 1.1: Hexdump of an Ethernet frame, without checksum, in network byte order

Note: To solve this task, information from the cheatsheet is necessary.

a)* In figure 1.1 mark the start and the end of the Ethernet header.

b) Reason, by highlighting and describing relevant header fields, which protocol is used at layer 3.

The Ethertype specifies the type of the Layer 2 payload. The value `0x0800` used here stands for IPv4.

c)* Describe how the length of the header on layer 3 is determined. Mark and name relevant sections in figure 1.1.

The header length in IPv4 is specified by the header field IHL. This is located in the lower nibble of the first byte of the IPv4 header and specifies the length of the header in multiples of 4 B. Thus, the length of the header is $5 \cdot 4\,\text{B} = 20\,\text{B}$.

d)* Mark all layer 3 adresses and name them.

e) Mark all extensions headers contained in layer 3.

The layer 3 payload is IPv4. IPv4 has no Extension Headers, only options. We know from subproblem c) that the header is 20 B long, which is the minimal length of an IPv4 header. Therefore there is nothing to mark.

f) Name and describe the 3 smallest header fields of layer 3. Indicate the size of those fields.

The 3 smallest header fields are all of size 1 bit.

RES `reserved`, reserved for potential future use.

DF `do not fragment`, informs the processors that this packet shall not be fragmented

MF `more fragments`, informs that — due to a previous fragmentation — this IP packet is split into multiple fragments

g) If there is an L3 SDU, state its type and justify the statement. Otherwise, state your thought process and discuss how this situation could occur.

The value of the IPv4 header field Protocol is `0x06`. Accordingly, the L3-SDU is TCP.

h) The bytes at offset `0x0042` and following are payload of layer 4. Specify the ASCII representation of the first 7 B of the payload.

The ASCII representation of `0x53 53 48 2d 32 2e 30` is `SSH-2.0`.

i) What application layer protocol is this probably and what is this protocol used for?

It is SSH (version 2.0), which is used for an encrypted console session on Linux/Unix and more recently also on Windows.

Problem 2 TCP Sequence Numbers

In the following, we consider a data transfer between a PC and a web server in order to understand the concept of TCP sequence and acknowledgement numbers.

a)* What is the meaning of the *Maximum Segment Size* (MSS), and what is the difference compared to the *Maximum Transmission Unit* (MTU)?

The Maximum Segment Size is the maximum amount of payload in a TCP segment, analogous to the MTU on Layer 2, which specifies the maximum payload size in an Ethernet frame. The MSS depends on the MTU and used protocols and their header lengths.

b)* Calculate the MSS for both IPv4 and IPv6 over Ethernet (MTU = 1500 B). Assume that no options or extension headers are used.

The length of the TCP header without options is $L_{h,\mathrm{TCP}} = 20\,\mathrm{B}$, as is the IPv4 header (without options) $L_{h,\mathrm{IPv4}} = 20\,\mathrm{B}$. The IPv6 header has a fixed length of $L_{h,\mathrm{IPv6}} = 40\,\mathrm{B}$.
This yields the following MSS values:

$$\mathrm{MSS}_{\mathrm{IPv4}} = \mathrm{MTU} - L_{h,\mathrm{IPv4}} - L_{h,\mathrm{TCP}} = 1500\,\mathrm{B} - 20\,\mathrm{B} - 20\,\mathrm{B} = 1460\,\mathrm{B}$$
$$\mathrm{MSS}_{\mathrm{IPv6}} = \mathrm{MTU} - L_{h,\mathrm{IPv6}} - L_{h,\mathrm{TCP}} = 1500\,\mathrm{B} - 40\,\mathrm{B} - 20\,\mathrm{B} = 1440\,\mathrm{B}$$

The PC now requests a text document hosted on the web server using HTTP 1.1 over IPv6. For this purpose, it sends a request with a length of $L_{\mathrm{Request}} = 430\,\mathrm{B}$. The web server answers the request with the HTTP response header in its own segment with $L_{\mathrm{Response}} = 160\,\mathrm{B}$. After the initial segment, the server sends the text file with a length of $L_{\mathrm{Data}} = 3080\,\mathrm{B}$, which it divides into multiple segments due to its size (*Segmentation*).

c) How many data segments will the web server send back to the client?

$$N = \left\lceil \frac{L_{\mathrm{Data}}}{\mathrm{MSS}_{\mathrm{IPv6}}} \right\rceil = \left\lceil \frac{3080\,\mathrm{B}}{1440\,\mathrm{B}} \right\rceil = 3$$

d) Determine the payload size in the data segments.

The first $N - 1$ segments will each be filled to the maximum, so $L_{S,1} = L_{S,2} = 1440\,\mathrm{B}$. The last segment contains the remaining data and therefore has a size of $L_{S,3} = 3080\,\mathrm{B} \bmod 1440\,\mathrm{B} = 200\,\mathrm{B}$.

In the following, we want to trace the progression of all transmitted segments and consider the used sequence and acknowledgement numbers, the flags, and the payload length in each of these segments. The PC and the server choose `8999` and `1839` as their initial sequence numbers. Note that the sequence numbers in segments always refer to the sender's own initial sequence number, whereas the acknowledgement numbers always refer to the peer's initial sequence number.

First, the PC initiates the connection establishment and issues the HTTP request. The server then responds with the HTTP response, which contains only the header, and subsequently sends the data segments. After it has sent all segments, the server also initiates the connection teardown. Assume that each data segment is acknowledged individually and immediately with an own acknowledgement segment.

e) Complete the connection establishment in the diagram.

f) Complete the data of the segments for transmitting the HTTP request and the HTTP response.

g) Add the segments for transmitting the data and the corresponding acknowledgements.

h) Complete the connection teardown in the diagram.