Computer Networks

Exercise Session 12

Prof. Dr. Oliver Hahm

Frankfurt University of Applied Sciences
Faculty 2: Computer Science and Engineering
oliver.hahm@fb2.fra-uas.de
https://teaching.dahahm.de

February 03, 2023

General Schedule

All exercises will follow this general schedule

- Identify potential understanding problems
  → Ask your questions
  → Recap of the lecture
- Address the understanding problems
  → Answer your questions
  → Repeat certain topics
- Walk through the exercises/solutions → Some hints and guidance
  → Work time or presentation of results

TCP

You have seen . . .

- the functioning and segment structure of TCP
- how flow control works in TCP
- what congestion control is
- which enhancements for TCP exist
- how a TCP connection is implemented with sockets
- what SYN Flood DOS attack is

UDP

You have seen . . .

- the functioning and segment structure of UDP
- that UDP is much simpler compared to TCP and allows for best-effort communication
- how a UDP server and client is implemented with sockets

Other Protocols

You have seen . . .

- SCTP as another connection-oriented transport layer protocol
- DCCP to be used for real-time applications
- QUIC as the newest relevant transport layer protocol to deal with shortcomings of TCP for web traffic

Domain Name System

You have seen . . .

- DNS as an essential protocol to translate between IP addresses and domain names
- the hierarchical namespace for domain names
- that every FQDN is part of a tree
- that this tree is composed below the root servers
- what a resource record is and which type it can have

Remote Shells

You have seen . . .

- Telnet and rlogin/rsh as simple examples for remote access to a host over the Internet
- that the Telnet client may serve as a debugging tool
- how SSH represents very popular and secure alternative for remote access
- that SSH can be used to tunnel traffic through

HTTP

You have seen ...

- HTTP as the basis for the WWW
- that HTTP messages are composed of a header and a body
- different types of HTTP methods and status codes
- how an HTTP request can look like
- the differences from HTTP/1.0 to HTTP/3

Email

You have seen . . .

- what a MUA and a MTA are
- that an every email consists of an envelope, a header, and a body
- SMTP, POP, and IMAP as the central protocols for email exchange
- how email suffers from various issues like Spam or Phishing
- more modern protocols and extensions to email to improve the security of the system
- how a SMTP communication looks like

More Protocols

You have seen . . .

- MQTT as a very relevant pub-sub protocol for IoT applications
- Signal as a secure protocol for instant messaging
- CoAP as a lightweight alternative to HTTP for constrained-node networks

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP $\rightarrow$ throughput

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page
      TCP → order

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page
      TCP → order
   7. Clock synchronization

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page
      TCP → order
   7. Clock synchronization
      UDP → latency

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page
      TCP → order
   7. Clock synchronization
      UDP → latency
   8. Video streaming

Exercise 1: Applications and Transport Layer Protocols

1. Select the most appropriate transport layer protocol for each of the following applications or application scenarios and explain your choice.
   1. File transfer (exchange file between two hosts over the network)
      TCP → throughput
   2. Video conferencing
      UDP → latency
   3. Instant messaging
      TCP → reliability
   4. Retrieving sensor information (e.g., temperature) from a sensor network
      UDP → low complexity
   5. Accessing a complex web page
      QUIC → avoid head of line blocking
   6. Accessing a simple web page
      TCP → order
   7. Clock synchronization
      UDP → latency
   8. Video streaming
      TCP → throughput

Exercise 1: Applications and Transport Layer Protocols

2. Many IoT applications rather use UDP as a transport layer protocol. Why?

Exercise 1: Applications and Transport Layer Protocols

2. Many IoT applications rather use UDP as a transport layer protocol. Why?

   UDP requires less resources - for the implementation and from the network. Both is often constrained in IoT scenarios.

Exercise 1: Applications and Transport Layer Protocols

2. Many IoT applications rather use UDP as a transport layer protocol. Why?

   UDP requires less resources - for the implementation and from the network. Both is often constrained in IoT scenarios.

3. CoAP is an application layer protocol designed to be used on top of UDP. However, it specifies certain features one would rather expect from a transport layer protocol. Explain the reason why no new transport layer protocol was specified instead.

Exercise 1: Applications and Transport Layer Protocols

2. Many IoT applications rather use UDP as a transport layer protocol. Why?

   UDP requires less resources - for the implementation and from the network. Both is often constrained in IoT scenarios.

3. CoAP is an application layer protocol designed to be used on top of UDP. However, it specifies certain features one would rather expect from a transport layer protocol. Explain the reason why no new transport layer protocol was specified instead.

   Introducing a new transport layer protocol on Internet scale is difficult. CoAP is designed to enable end-to-end connection between hosts in the Internet and *things*. Integrating a new transport layer implementation in all clients is difficult.

Exercise 1: Applications and Transport Layer Protocols

4. CoAP offers four different message types. Name them and describe what their meaning.

Exercise 1: Applications and Transport Layer Protocols

4. CoAP offers four different message types. Name them and describe what their meaning.
   - Requests:

     Confirmable – Expects an acknowledgement
     Non-confirmable – Does not expect an acknowledgement
   - Responses:

     Acknowledgement – Acknowledges a confirmable message
     Reset – Indicates that it had received a message but could not process it

Exercise 2: TCP and UDP

1. Explain the **differences** between TCP and UDP.

Exercise 2: TCP and UDP

1. Explain the **differences** between TCP and UDP.
   - *UDP*
     - *Connectionless Transport Layer protocol. Transmissions take place without previous connection establishment.*
     - *More simple protocol in contrast to the connection-oriented TCP. Only responsible for addressing of the segments. Does not secure the data transmission.*
     - *The receiver does not acknowledge transmissions at the sender. Segments can get lost during transmission.*
   - *TCP*
     - *Connection-oriented Transport Layer protocol.*
     - *Makes connections via IP reliable in a way that is desired or simply necessary for many applications.*
     - *Guarantees that segments reach their destination completely and the correct order. Lost or unacknowledged TCP segments are requested by the receiver at the sender.*

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

   TCP is used for SMTP transmission because reliable transport is required and the protocol implements a state machine. For file transfer the order, reliability, and throughput are important.

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

   TCP is used for SMTP transmission because reliable transport is required and the protocol implements a state machine. For file transfer the order, reliability, and throughput are important.

3. Describe **two examples**, where using the Transport Layer protocol UDP makes sense.

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

   TCP is used for SMTP transmission because reliable transport is required and the protocol implements a state machine. For file transfer the order, reliability, and throughput are important.

3. Describe **two examples**, where using the Transport Layer protocol UDP makes sense.

   UDP can be used for video conferencing or video live streaming, the only consequence of losing a segment is losing an image.

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

   TCP is used for SMTP transmission because reliable transport is required and the protocol implements a state machine. For file transfer the order, reliability, and throughput are important.

3. Describe **two examples**, where using the Transport Layer protocol UDP makes sense.

   UDP can be used for video conferencing or video live streaming, the only consequence of losing a segment is losing an image.

4. Describe what a socket is.

Exercise 2: TCP and UDP

2. Describe **two examples**, where using the Transport Layer protocol TCP makes sense.

   TCP is used for SMTP transmission because reliable transport is required and the protocol implements a state machine. For file transfer the order, reliability, and throughput are important.

3. Describe **two examples**, where using the Transport Layer protocol UDP makes sense.

   UDP can be used for video conferencing or video live streaming, the only consequence of losing a segment is losing an image.

4. Describe what a socket is.

   Sockets are the platform-independent, standardized interface between the implementation of the transport layer protocols in the OS and the applications.

Exercise 2: TCP and UDP

5 Describe what the Seq number in an TCP segment specifies.

Exercise 2: TCP and UDP

5 Describe what the Seq number in an TCP segment specifies.

The sequence number of a segment is the position of the segments first byte in the data stream.

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

   The sequence number of the next expected segment.

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

   The sequence number of the next expected segment.

7. Describe the **silly window syndrome** and its effect.

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

   The sequence number of the next expected segment.

7. Describe the **silly window syndrome** and its effect.

   The Silly window syndrome is a problem where the receiver window frequently with a very small number of bytes. This leads to a lot of overhead.

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

   The sequence number of the next expected segment.

7. Describe the **silly window syndrome** and its effect.

   The Silly window syndrome is a problem where the receiver window frequently with a very small number of bytes. This leads to a lot of overhead.

8. Describe the functioning of **silly window syndrome avoidance**.

Exercise 2: TCP and UDP

5. Describe what the Seq number in an TCP segment specifies.

   The sequence number of a segment is the position of the segments first byte in the data stream.

6. Describe what the ACK number in an TCP segment specifies

   The sequence number of the next expected segment.

7. Describe the **silly window syndrome** and its effect.

   The Silly window syndrome is a problem where the receiver window frequently with a very small number of bytes. This leads to a lot of overhead.

8. Describe the functioning of **silly window syndrome avoidance**.

   The receiver notifies the sender about free storage capacity in the receive window not before 25% of the reception buffer is free or a segment size of size MSS can be received.

Exercise 2: TCP and UDP

9 Which two possible **reasons** for the occurrence of congestion in computer networks exist?

Exercise 2: TCP and UDP

9 Which two possible **reasons** for the occurrence of congestion in computer networks exist?

The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

Exercise 2: TCP and UDP

9. Which two possible **reasons** for the occurrence of congestion in computer networks exist?

   The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10. Why does the sender maintain **two windows** when using TCP and not just a single one?

Exercise 2: TCP and UDP

9 Which two possible **reasons** for the occurrence of congestion in computer networks exist?

The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10 Why does the sender maintain **two windows** when using TCP and not just a single one?

The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

Exercise 2: TCP and UDP

9 Which two possible **reasons** for the occurrence of congestion in computer networks exist?

The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10 Why does the sender maintain **two windows** when using TCP and not just a single one?

The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

11 Describe what the slow-start phase is.

Exercise 2: TCP and UDP

9 Which two possible **reasons** for the occurrence of congestion in computer networks exist?

The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10 Why does the sender maintain **two windows** when using TCP and not just a single one?

The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

11 Describe what the slow-start phase is.

The (initial) exponential growth phase.

Exercise 2: TCP and UDP

9. Which two possible **reasons** for the occurrence of congestion in computer networks exist?

   The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10. Why does the sender maintain **two windows** when using TCP and not just a single one?

    The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

11. Describe what the slow-start phase is.

    The (initial) exponential growth phase.

12. Describe what the congestion avoidance phase is.

Exercise 2: TCP and UDP

9. Which two possible **reasons** for the occurrence of congestion in computer networks exist?

   The receiver can not process the received data fast enough and therefore its receive buffer becomes full. Congestion of the network occurs.

10. Why does the sender maintain **two windows** when using TCP and not just a single one?

    The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

11. Describe what the slow-start phase is.

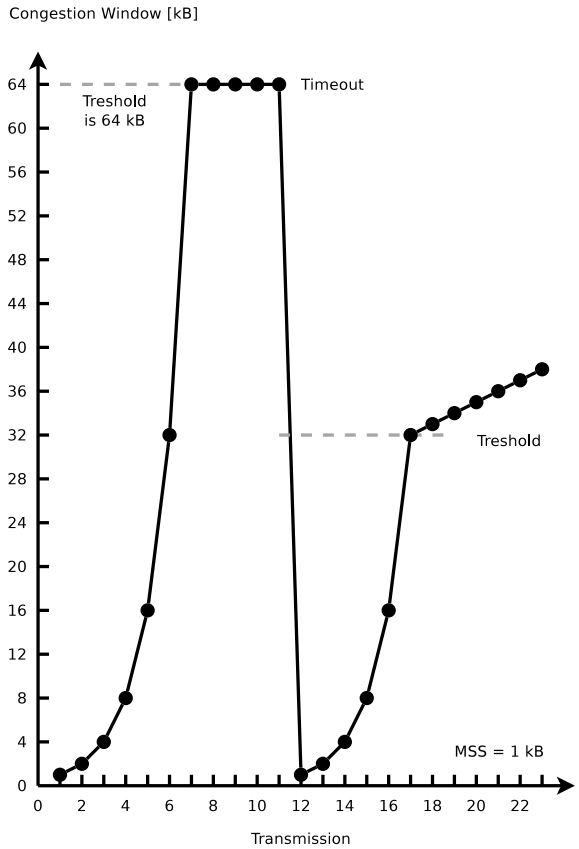
    The (initial) exponential growth phase.

12. Describe what the congestion avoidance phase is.

    The linear growth phase after a configures threshold has exceeded.

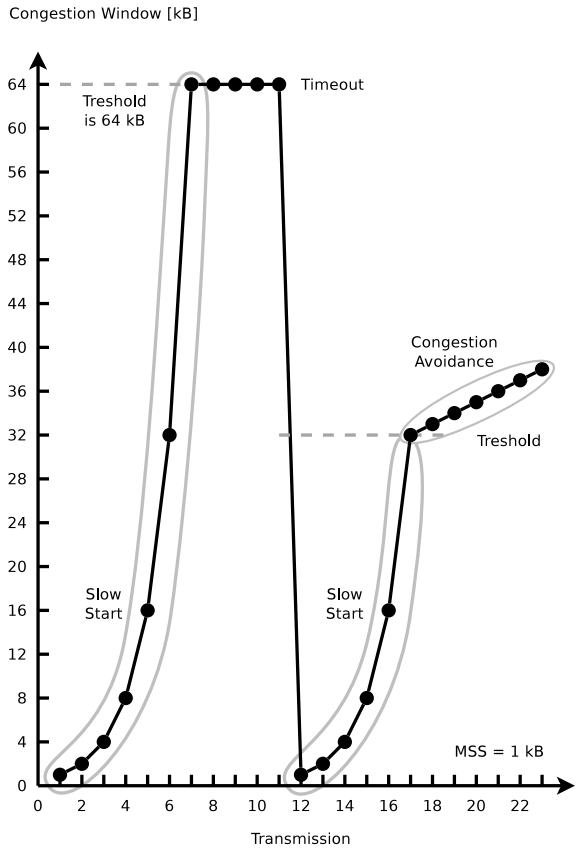
Exercise 2: TCP and UDP

13 Mark in the figure both the slow-start phase and the congestion avoidance phase.



Exercise 2: TCP and UDP

13 Mark in the figure both the slow-start phase and the congestion avoidance phase.



Exercise 2: TCP and UDP

14 Describe what fast retransmit is.

Exercise 2: TCP and UDP

14 Describe what fast retransmit is.

After three duplicate ACKs arrived, the lost segment is sent again.

Exercise 2: TCP and UDP

14 Describe what fast retransmit is.

After three duplicate ACKs arrived, the lost segment is sent again.

15 Describe what fast recovery is.

Exercise 2: TCP and UDP

14 Describe what fast retransmit is.

After three duplicate ACKs arrived, the lost segment is sent again.

15 Describe what fast recovery is.

The slow-start phase after three duplicate ACKs arrived is avoided. If three duplicate ACKs arrive, the congestion window is set directly on the threshold value.

Exercise 2: TCP and UDP

16 The concept of TCP congestion control is called **AIMD** (= Additive Increase / Multiplicative Decrease). **Describe the reason** for the aggressive reduction and conservative increase of the congestion window.

Exercise 2: TCP and UDP

16 The concept of TCP congestion control is called **AIMD** (= Additive Increase / Multiplicative Decrease). **Describe the reason** for the aggressive reduction and conservative increase of the congestion window.

The consequences of a congestion window which is too large in size are worse than for a window which is too small. If the window is too small in size, available bandwidth remains unused. If the window is too large in size, segments will get lost and must be transmitted again. This increases the congestion of the network even more! The congestion state must be left as fast as possible. Therefore, the size of the congestion window is reduced significantly.

Exercise 2: TCP and UDP

17 Describe the functioning of a Denial-of-Service attack via **SYN flood**.

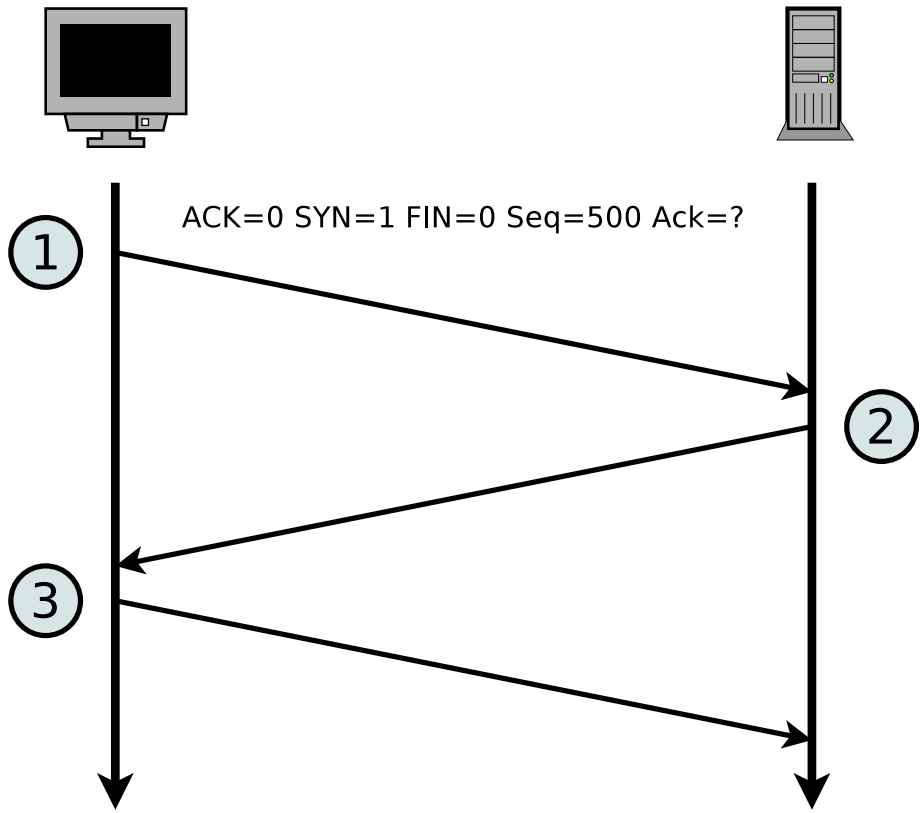
Exercise 2: TCP and UDP

17 Describe the functioning of a Denial-of-Service attack via **SYN flood**.

A client sends many connection requests (*SYN*), but does not respond to the acknowledgments (*SYN ACK*) of the server via *ACK*. The server waits some time for the acknowledgment of the clients because the delay of the confirmation could be caused by a network issue. During this period, the address of the client and the status of incomplete connection are stored in the memory of the network stack. By flooding the server with connection requests, the table which stores the TCP connections in the network stack is completely filled. This causes the server to become unable to establish new connections. The memory consumption at the server may become this large that the main memory gets completely filled and the server crashes.

Exercise 3: TCP Connections

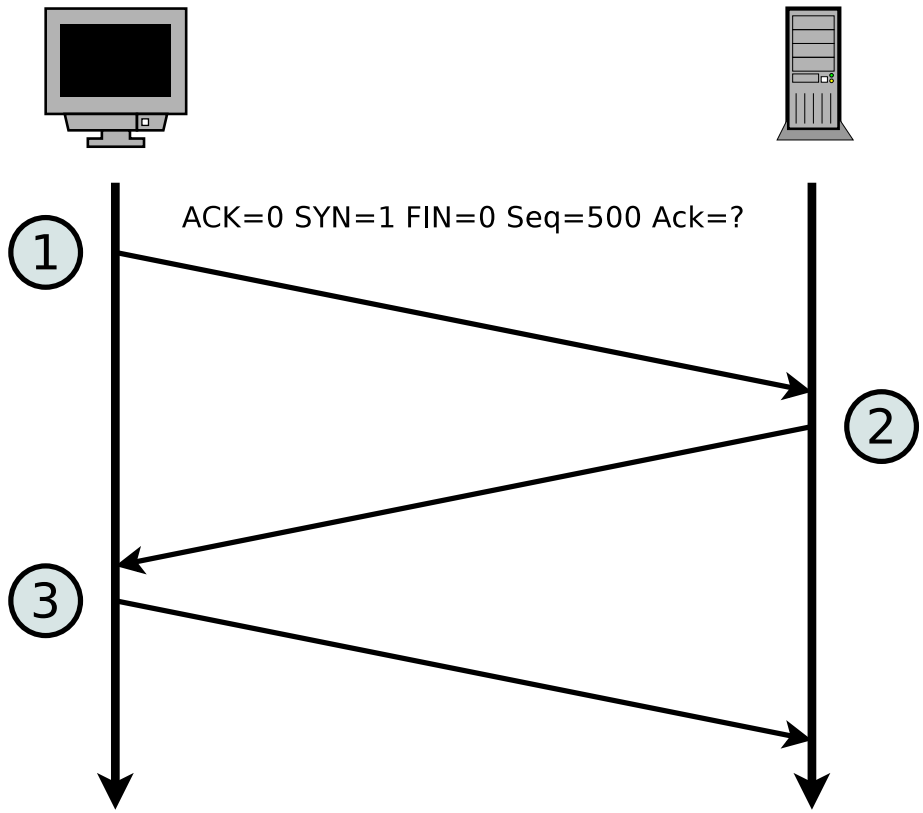
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
1	0	1	0	0	500	
2					1000	
3						

Exercise 3: TCP Connections

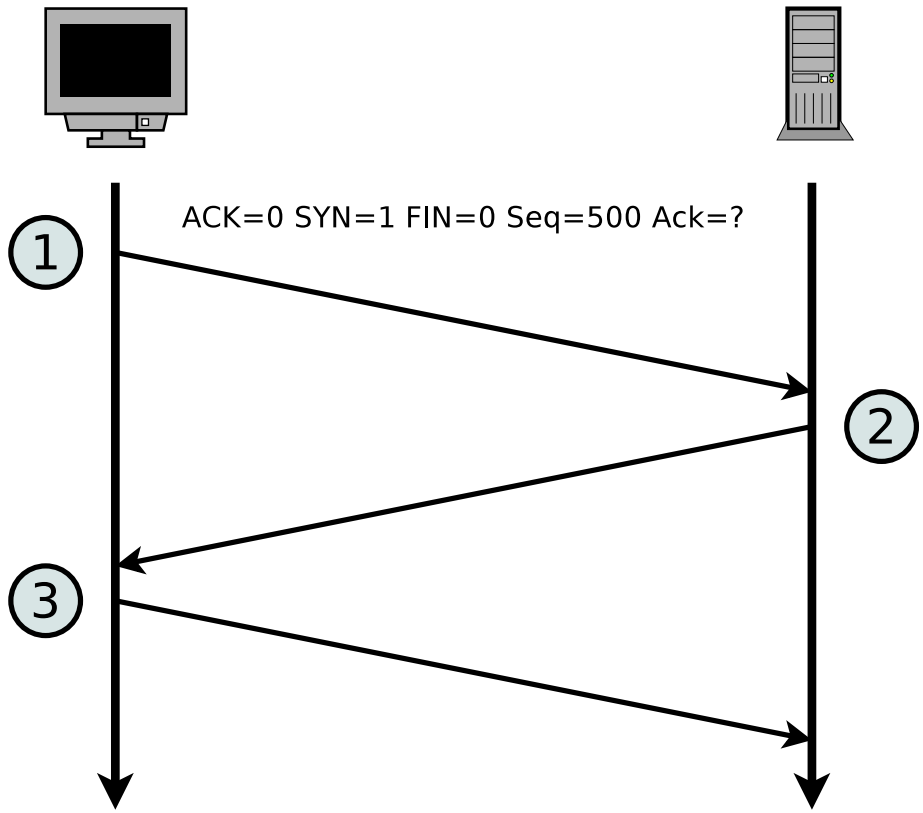
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
1	0	1	0	0	500	0
2	1	1	0	0	1000	501
3						

Exercise 3: TCP Connections

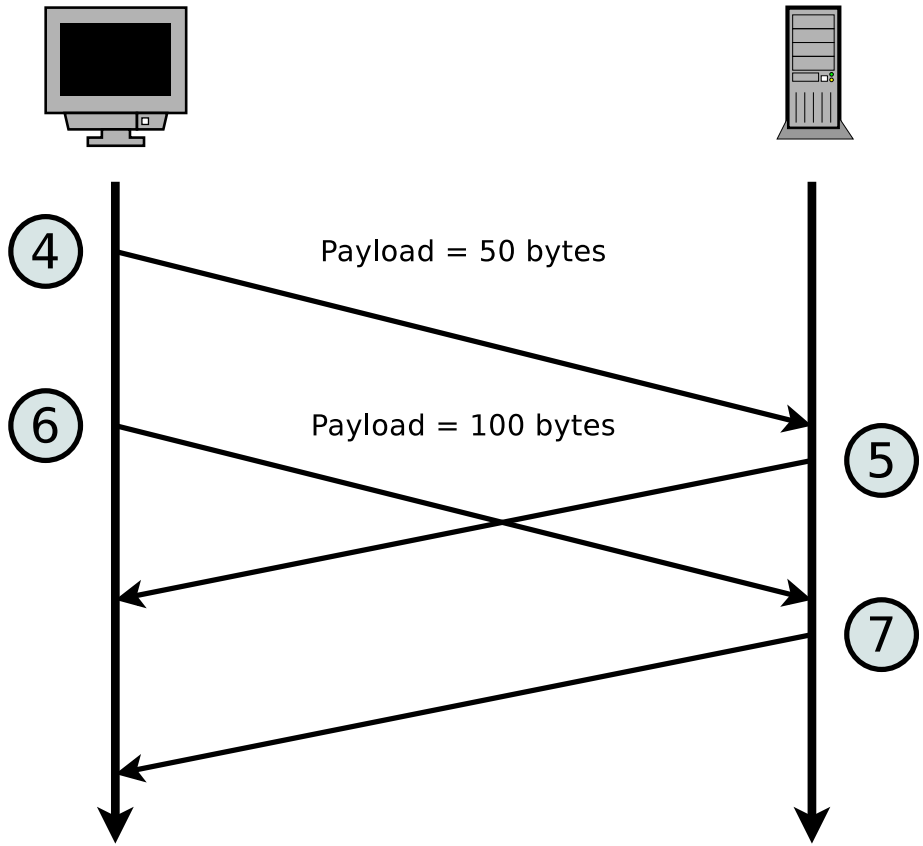
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
1	0	1	0	0	500	0
2	1	1	0	0	1000	501
3	1	0	0	0	501	1001

Exercise 3: TCP Connections

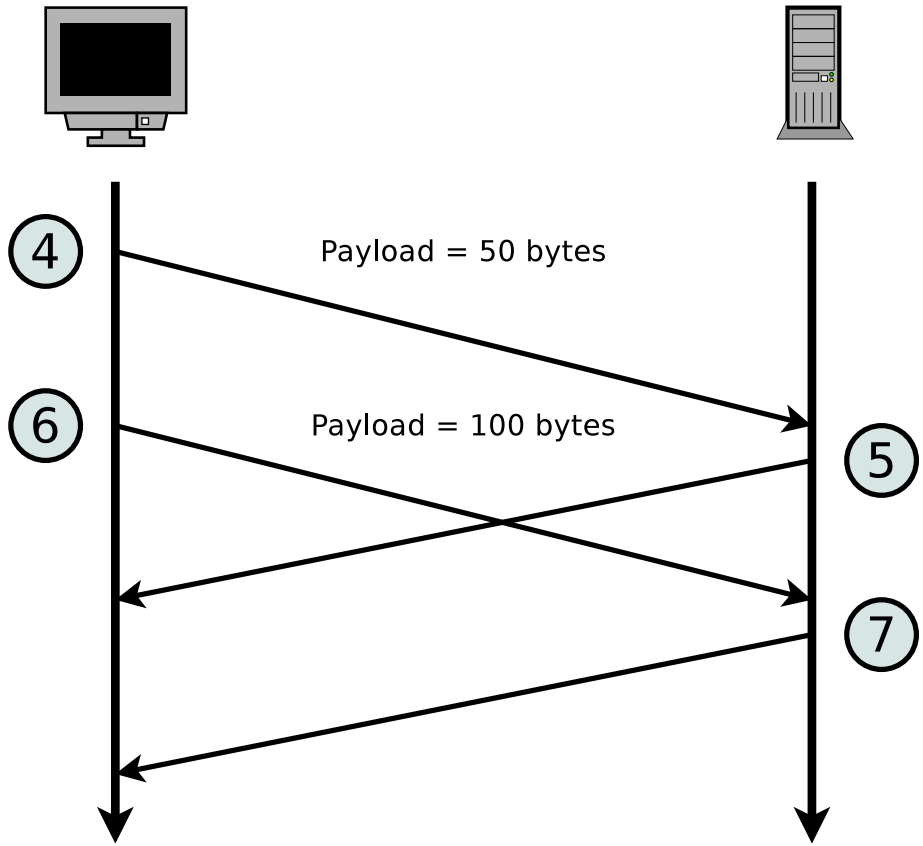
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
4	0			50	501	1001
5	1			0		
6	0			100		
7	1			0		

Exercise 3: TCP Connections

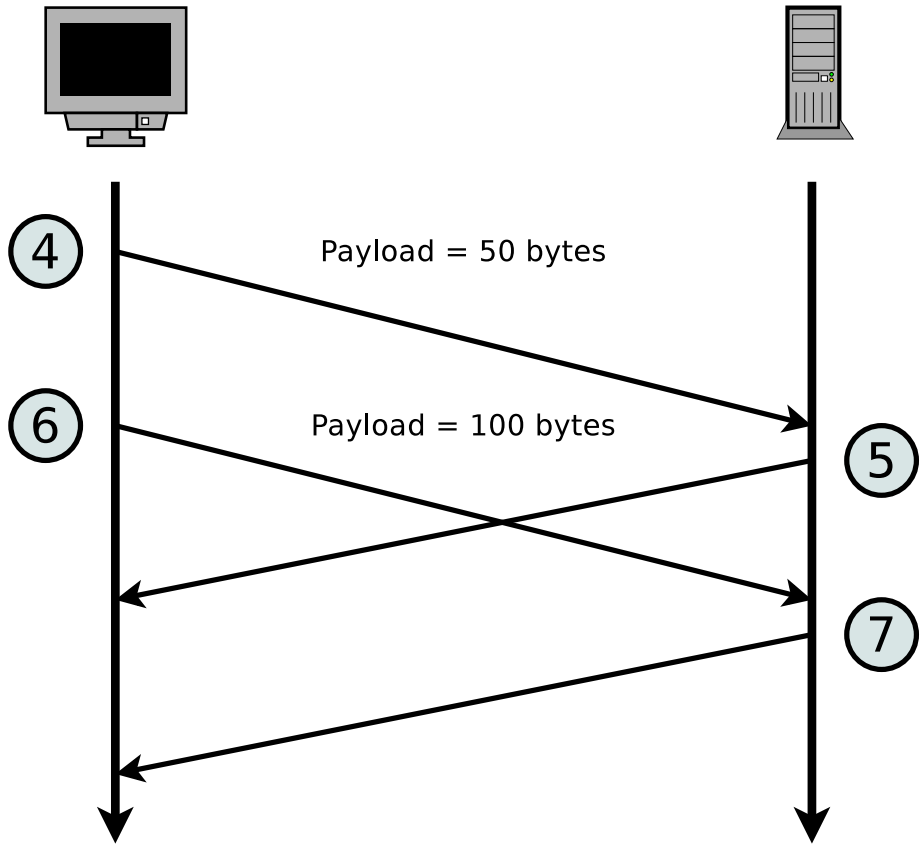
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
4	0	0	0	50	501	1001
5	1	0	0	0	1001	551
6	0			100		
7	1			0		

Exercise 3: TCP Connections

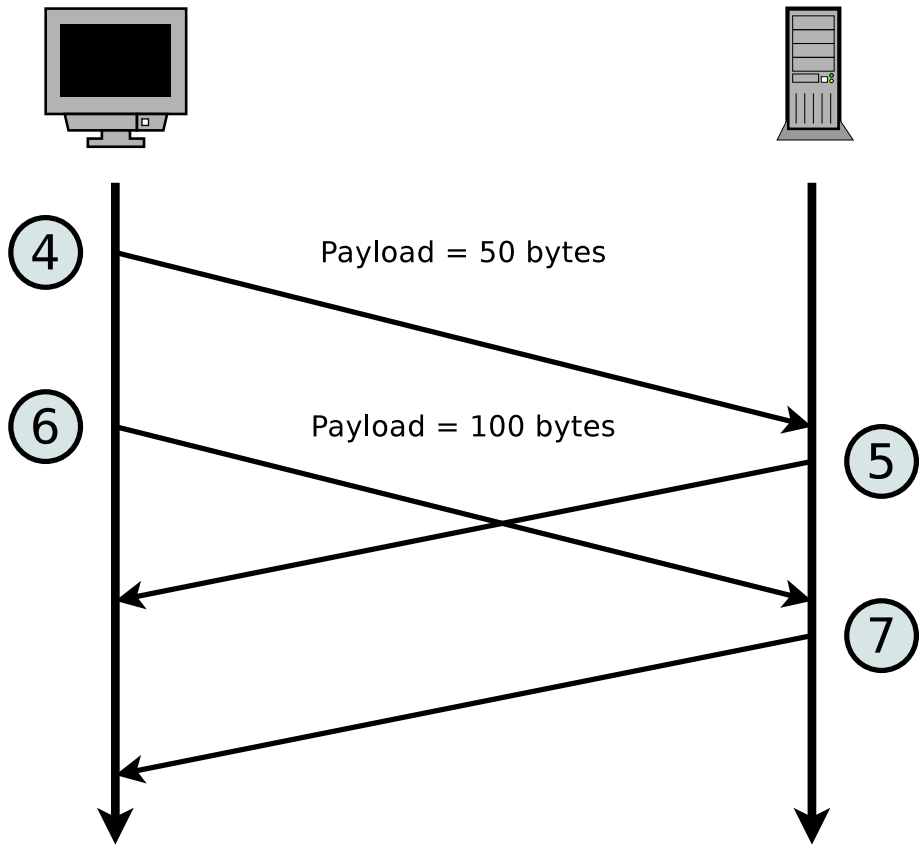
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
4	0	0	0	50	501	1001
5	1	0	0	0	1001	551
6	0	0	0	100	551	1001
7	1			0		

Exercise 3: TCP Connections

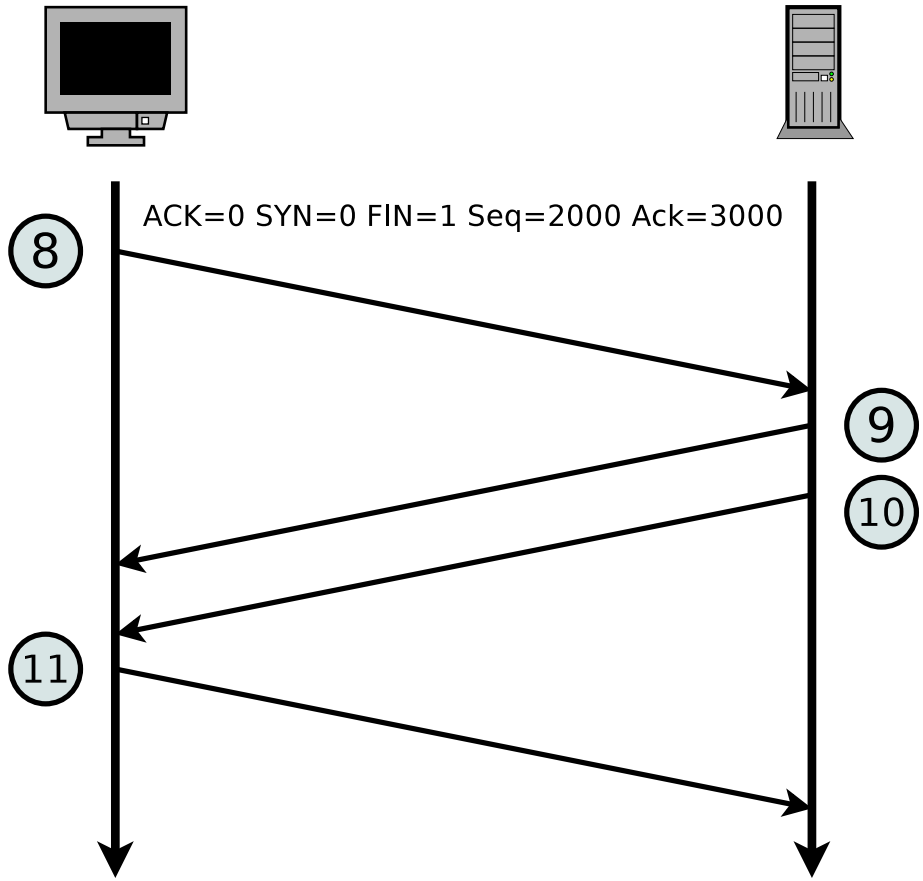
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
4	0	0	0	50	501	1001
5	1	0	0	0	1001	551
6	0	0	0	100	551	1001
7	1	0	0	0	1001	651

Exercise 3: TCP Connections

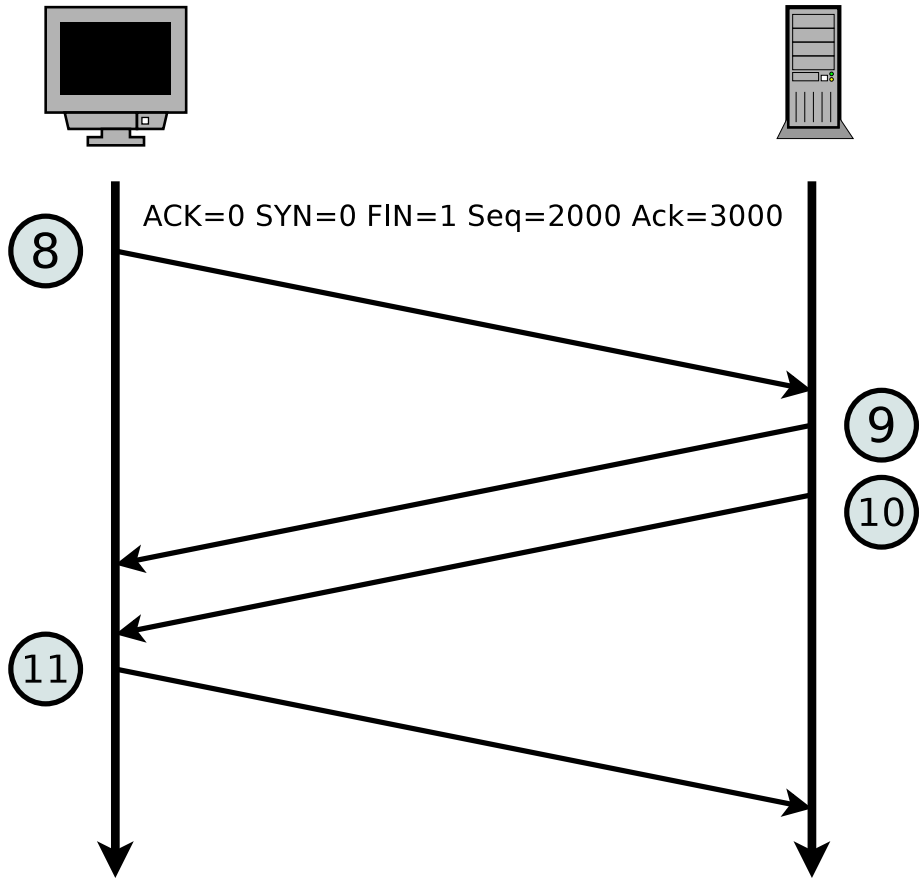
3 The diagram shows the termination of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
8	0	0	1	0	2000	3000
9				0		
10				0		
11				0		

Exercise 3: TCP Connections

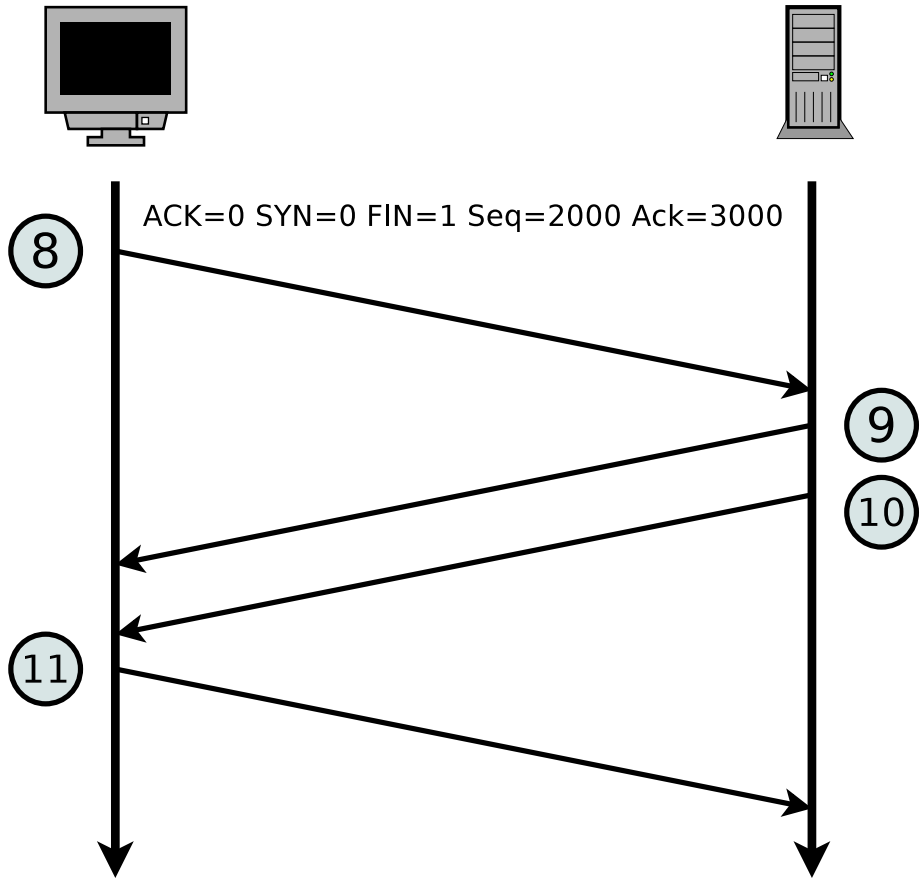
3 The diagram shows the termination of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
8	0	0	1	0	2000	3000
9	1	0	0	0	3000	2001
10				0		
11				0		

Exercise 3: TCP Connections

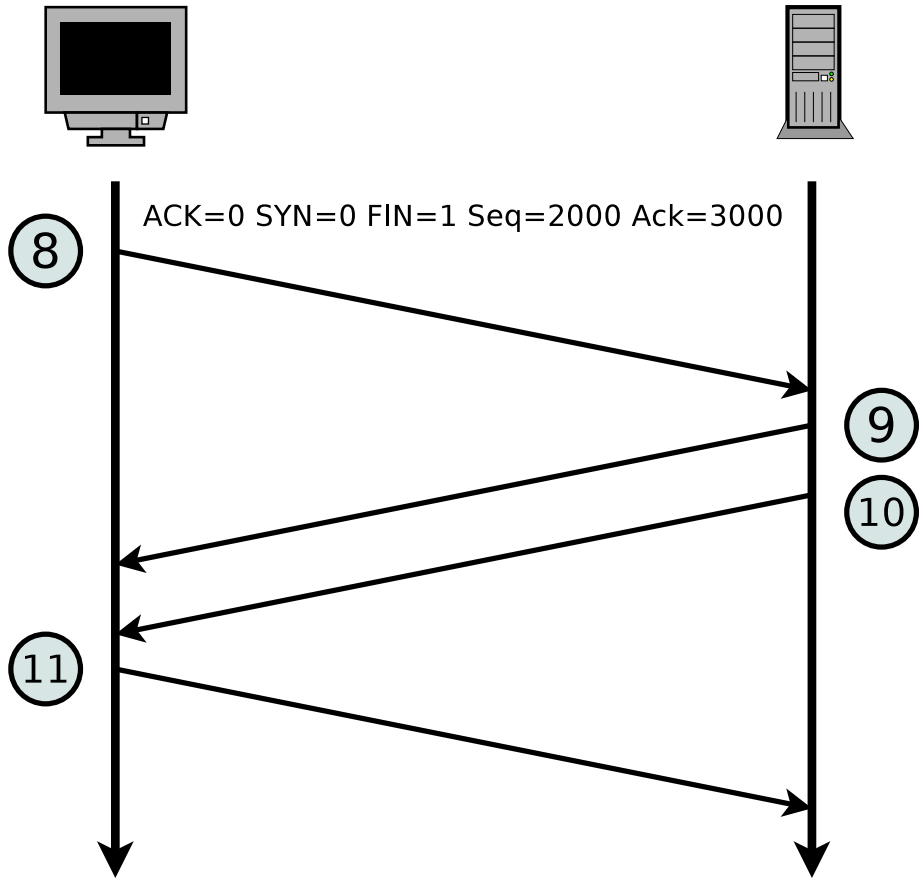
3 The diagram shows the termination of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
8	0	0	1	0	2000	3000
9	1	0	0	0	3000	2001
10	0	0	1	0	3000	2001
11				0		

Exercise 3: TCP Connections

3 The diagram shows the termination of a TCP connection. Complete the table.



Message	ACK	SYN	FIN	Payload length	Seq number	ACK number
8	0	0	1	0	2000	3000
9	1	0	0	0	3000	2001
10	0	0	1	0	3000	2001
11	1	0	0	0	2001	3001

Exercise 4: Transmission Control Protocol

1. Consider the effect of using slow start on a line with a RTT of 10 ms. The maximum segment size is 2 kB and the receive window has a size of 24 kB. How long does it take before the first full window can be sent if no congestion occurs?

Exercise 4: Transmission Control Protocol

1. Consider the effect of using slow start on a line with a RTT of 10 ms. The maximum segment size is 2 kB and the receive window has a size of 24 kB. How long does it take before the first full window can be sent if no congestion occurs?

   The first bursts contain 2 kB, 4 kB, 8 kB, and 16 kB bytes, respectively. The next one is 24 kB and occurs after 40 ms.

Exercise 4: Transmission Control Protocol

1. Consider the effect of using slow start on a line with a RTT of 10 ms. The maximum segment size is 2 kB and the receive window has a size of 24 kB. How long does it take before the first full window can be sent if no congestion occurs?

   The first bursts contain 2 kB, 4 kB, 8 kB, and 16 kB bytes, respectively. The next one is 24 kB and occurs after 40 ms.

2. Given a maximum segment size of 1 kB: Assume that the congestion window is set to 18 kB just before a timeout occurs. How big will the window be after four consecutive successful transmissions if fast recovery is **not** used?

Exercise 4: Transmission Control Protocol

1. Consider the effect of using slow start on a line with a RTT of 10 ms. The maximum segment size is 2 kB and the receive window has a size of 24 kB. How long does it take before the first full window can be sent if no congestion occurs?

   The first bursts contain 2 kB, 4 kB, 8 kB, and 16 kB bytes, respectively. The next one is 24 kB and occurs after 40 ms.

2. Given a maximum segment size of 1 kB: Assume that the congestion window is set to 18 kB just before a timeout occurs. How big will the window be after four consecutive successful transmissions if fast recovery is **not** used?

   The next transmission will be 1 MSS. Then 2, 4, and 8. So after four successes, it will be 8 kB.

Exercise 4: Transmission Control Protocol

3 A TCP machine is sending full windows of 65,535 bytes over a 1 Gb/s channel. The channel provides a one-way delay of 10 ms. What is the maximum throughput that can be achieved? What does this mean for the efficiency of the channel usage?

Exercise 4: Transmission Control Protocol

3. A TCP machine is sending full windows of 65,535 bytes over a 1 Gb/s channel. The channel provides a one-way delay of 10 ms. What is the maximum throughput that can be achieved? What does this mean for the efficiency of the channel usage?

   One window can be sent every 20 ms. This gives 50 windows/s, for a maximum data rate of about 3.3 million B/s. The line efficiency is then $\frac{26.4}{1000}$ Mb/s or 2.6 percent.

Exercise 4: Transmission Control Protocol

3. A TCP machine is sending full windows of 65,535 bytes over a 1 Gb/s channel. The channel provides a one-way delay of 10 ms. What is the maximum throughput that can be achieved? What does this mean for the efficiency of the channel usage?

   One window can be sent every 20 ms. This gives 50 windows/s, for a maximum data rate of about 3.3 million B/s. The line efficiency is then $\frac{26.4}{1000}$ Mb/s or 2.6 percent.

4. What is the impact of the bandwidth-delay product on flow control?

Exercise 4: Transmission Control Protocol

3. A TCP machine is sending full windows of 65,535 bytes over a 1 Gb/s channel. The channel provides a one-way delay of 10 ms. What is the maximum throughput that can be achieved? What does this mean for the efficiency of the channel usage?

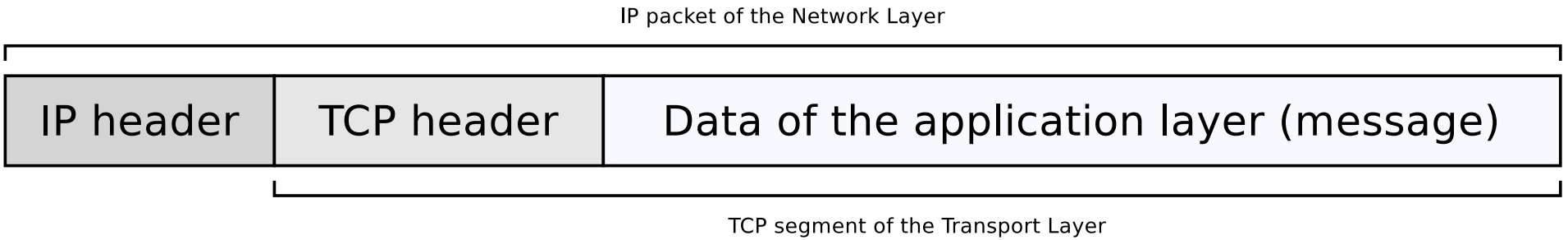
   One window can be sent every 20 ms. This gives 50 windows/s, for a maximum data rate of about 3.3 million B/s. The line efficiency is then $\frac{26.4}{1000}$ Mb/s or 2.6 percent.

4. What is the impact of the bandwidth-delay product on flow control?

   The higher the bandwidth-delay product, the more data is *stored* in the line. The more data is stored in the line, the longer a sender has to wait for an acknowledgement. The longer a sender has to wait for an ACK, the lower the efficiency.

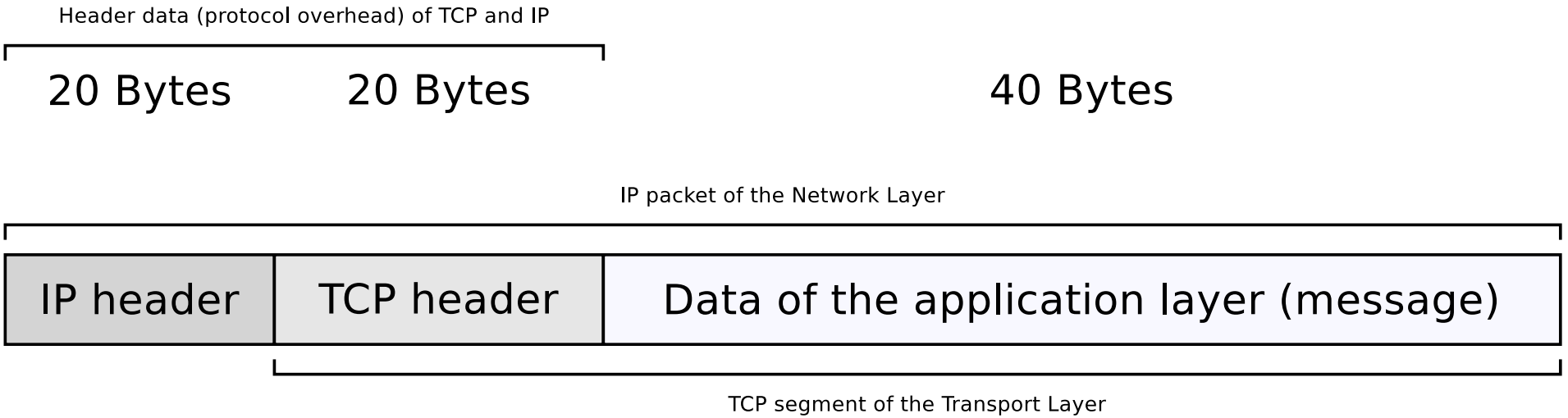
Exercise 5: Header and Payload

An application generates 40 bytes payload which is first packed into a single TCP segment, and then packed into a single IP packet. What is the percentage of header data in the IP packet and what is the percentage of application generated payload?



Exercise 5: Header and Payload

An application generates 40 bytes payload which is first packed into a single TCP segment, and then packed into a single IP packet. What is the percentage of header data in the IP packet and what is the percentage of application generated payload?



TCP header = usually 20 bytes
IP header = usually 20 bytes
$\Longrightarrow$ the IP packet contains usually 40 bytes (= 50%) header data.

Exercise 6: Domain Name System

1. DNS uses UDP instead of TCP. In case of packet loss on the network layer, there is no automatic recovery. Does this cause a problem, and if so, how is it solved?

Exercise 6: Domain Name System

1. DNS uses UDP instead of TCP. In case of packet loss on the network layer, there is no automatic recovery. Does this cause a problem, and if so, how is it solved?

   DNS is idempotent. Operations can be repeated without harm. When a process makes a DNS request, it starts a timer. If the timer expires, it just makes the request again. No harm is done.

Exercise 6: Domain Name System

1. DNS uses UDP instead of TCP. In case of packet loss on the network layer, there is no automatic recovery. Does this cause a problem, and if so, how is it solved?

   DNS is idempotent. Operations can be repeated without harm. When a process makes a DNS request, it starts a timer. If the timer expires, it just makes the request again. No harm is done.

2. In addition to being subject to loss, UDP packets have a maximum length, potentially as low as 576 bytes. What happens when a DNS name to be looked up exceeds this length? Can it be sent in two packets?

Exercise 6: Domain Name System

1. DNS uses UDP instead of TCP. In case of packet loss on the network layer, there is no automatic recovery. Does this cause a problem, and if so, how is it solved?

   DNS is idempotent. Operations can be repeated without harm. When a process makes a DNS request, it starts a timer. If the timer expires, it just makes the request again. No harm is done.

2. In addition to being subject to loss, UDP packets have a maximum length, potentially as low as 576 bytes. What happens when a DNS name to be looked up exceeds this length? Can it be sent in two packets?

   The problem usually does not occur. DNS names must be shorter than 256 bytes. The standard requires this. Thus, all DNS names fit in a single minimum-length packet. In other cases (for other RR types) TCP can be used.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin. → valid
   - www1.frankfurt-university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin. → valid
   - www1.frankfurt-university.de. → valid
   - 1www.frankfurt-university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin. → valid
   - www1.frankfurt-university.de. → valid
   - 1www.frankfurt-university.de. → valid
   - www.frankfurt.-university.de.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin. → valid
   - www1.frankfurt-university.de. → valid
   - 1www.frankfurt-university.de. → valid
   - www.frankfurt.-university.de. → invalid, a label must not start with a hypen
   - myhost.local.domain.

Exercise 6: Domain Name System

3. The TTL of resource record may cause a delay of various hours or even days until the change of an IP address for a given name is updated for every host. Hence, would it be a good idea to use only very small values for the TTL? Explain why or why not.

   The shorter the TTL, the shorter an entry might be cached. Hence, more requests are required, putting more load on the network.

4. Which of the following specifies a valid domain name:
   - mail.frankfurt-university.de. → valid
   - www.frankfurt/university.de. → invalid, only the hyphen as special character is allowed
   - sea-01.cit.frankfurt-university.de. → valid
   - university.berlin. → valid
   - www1.frankfurt-university.de. → valid
   - 1www.frankfurt-university.de. → valid
   - www.frankfurt.-university.de. → invalid, a label must not start with a hypen
   - myhost.local.domain. → valid, but won't resolve in the Internet

Exercise 7: Networking Applications

1. Describe which protocols are involved when you boot up your computer, open a web browser, go to the `https://webmail.frankfurt-university.de`, login, and send an email to `oliver.hahm@fb2.fra-uas.de`.

Exercise 7: Networking Applications

1. Describe which protocols are involved when you boot up your computer, open a web browser, go to the `https://webmail.frankfurt-university.de`, login, and send an email to `oliver.hahm@fb2.fra-uas.de`.

   At boot time DHCP (over UDP, IPv4, and Ethernet/WLAN) or SLAAC (over IPv6) may be used to obtain an IP address

   Accessing the webmail page Uses HTTP over TCP, IP and layer 2

   Login to the mailserver Uses HTTP POST and IMAP (again over TCP/IP and layer 2)

   Send a mail Uses HTTP POST (and maybe PUT) plus SMTP (again over TCP/IP and layer 2)

Exercise 7: Networking Applications

2. Explain the purpose for each of the protocols from the previous question.

Exercise 7: Networking Applications

2. Explain the purpose for each of the protocols from the previous question.

DHCP	For address autoconfiguration
HTTP	To communicate with the webserver hosting the webmail frontend
IMAP	To retrieve (read) the emails
SMTP	To send an email
UDP	As transport layer for DHCP
TCP	As reliable transport layer for all other application layer protocols
IP	To address the web server or mailserver and find the best path
Ethernet/WLAN	To connect your computer to your local router/gateway

Exercise 7: Networking Applications

2. Explain the purpose for each of the protocols from the previous question.

DHCP	For address autoconfiguration
HTTP	To communicate with the webserver hosting the webmail frontend
IMAP	To retrieve (read) the emails
SMTP	To send an email
UDP	As transport layer for DHCP
TCP	As reliable transport layer for all other application layer protocols
IP	To address the web server or mailserver and find the best path
Ethernet/WLAN	To connect your computer to your local router/gateway

3. Which of these protocols act on the application layer?

Exercise 7: Networking Applications

2. Explain the purpose for each of the protocols from the previous question.

   DHCP For address autoconfiguration

   HTTP To communicate with the webserver hosting the webmail frontend

   IMAP To retrieve (read) the emails

   SMTP To send an email

   UDP As transport layer for DHCP

   TCP As reliable transport layer for all other application layer protocols

   IP To address the web server or mailserver and find the best path

   Ethernet/WLAN To connect your computer to your local router/gateway

3. Which of these protocols act on the application layer?

   DHCP, HTTP, SMTP, IMAP

Exercise 7: Networking Applications

4 The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

Exercise 7: Networking Applications

4 The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

A label in a DNS name may not be alnumeric.

Exercise 7: Networking Applications

4. The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

   A label in a DNS name may not be alnumeric.

5. When you try to access my personal web page via `https://176.9.70.110/index.html` you will get an HTTP status code 404. When you access it via `https://teaching.dahahm.de/index.html` you will get HTTP status HTTP status code 200. Explain the meaning of both status codes. Can you imagine why the result is different?

Exercise 7: Networking Applications

4. The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

   A label in a DNS name may not be alnumeric.

5. When you try to access my personal web page via `https://176.9.70.110/index.html` you will get an HTTP status code 404. When you access it via `https://teaching.dahahm.de/index.html` you will get HTTP status HTTP status code 200. Explain the meaning of both status codes. Can you imagine why the result is different?

   The webserver manages multiple *vhosts*. If no hostname is given, a default configuration may be used (or not).

Exercise 7: Networking Applications

4. The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

   A label in a DNS name may not be alnumeric.

5. When you try to access my personal web page via `https://176.9.70.110/index.html` you will get an HTTP status code 404. When you access it via `https://teaching.dahahm.de/index.html` you will get HTTP status HTTP status code 200. Explain the meaning of both status codes. Can you imagine why the result is different?

   The webserver manages multiple *vhosts*. If no hostname is given, a default configuration may be used (or not).

6. For the exchange of emails more than one protocol is used. Name at least two of them and explain the provided service for each of them.

Exercise 7: Networking Applications

4. The DNS *A record* for `teaching.dahahm.de` resolves to `176.9.70.110`. An alternative way to enter the URL into the browser's address field is: `https://176.9.70.110/index.html` How does the browser know whether the given name is a DNS name or an IP address?

   A label in a DNS name may not be alnumeric.

5. When you try to access my personal web page via `https://176.9.70.110/index.html` you will get an HTTP status code 404. When you access it via `https://teaching.dahahm.de/index.html` you will get HTTP status HTTP status code 200. Explain the meaning of both status codes. Can you imagine why the result is different?

   The webserver manages multiple *vhosts*. If no hostname is given, a default configuration may be used (or not).

6. For the exchange of emails more than one protocol is used. Name at least two of them and explain the provided service for each of them.

   SMTP is used to between MTAs and for sending mails from the MUA. IMAP and POP can be used to retrieve mails from the server.

Exercise 8: Do some research

1. The checksum in UDP is optional, i.e., it can be used to protect the integrity of the entire datagram or not. Is there also a way to *partially* protect the payload against transmission errors?

Exercise 8: Do some research

1. The checksum in UDP is optional, i.e., it can be used to protect the integrity of the entire datagram or not. Is there also a way to *partially* protect the payload against transmission errors?

   RFC 3828 describes UDP-Lite *"which is similar to the User Datagram Protocol (UDP) (RFC 768), but can also serve applications in error-prone network environments that prefer to have partially damaged payloads delivered rather than discarded. If this feature is not used, UDP-Lite is semantically identical to UDP."*

Exercise 8: Do some research

1. The checksum in UDP is optional, i.e., it can be used to protect the integrity of the entire datagram or not. Is there also a way to *partially* protect the payload against transmission errors?

   RFC 3828 describes UDP-Lite *"which is similar to the User Datagram Protocol (UDP) (RFC 768), but can also serve applications in error-prone network environments that prefer to have partially damaged payloads delivered rather than discarded. If this feature is not used, UDP-Lite is semantically identical to UDP."*

2. The original congestion control algorithm in TCP was called Tahoe. Many other algorithms were introduced over the last decades. Name two of them that can be used without any knowledge about the TCP implementation on the receiver side and two that requires information about the receiver's TCP implementation.

Exercise 8: Do some research

1. The checksum in UDP is optional, i.e., it can be used to protect the integrity of the entire datagram or not. Is there also a way to *partially* protect the payload against transmission errors?

   RFC 3828 describes UDP-Lite *"which is similar to the User Datagram Protocol (UDP) (RFC 768), but can also serve applications in error-prone network environments that prefer to have partially damaged payloads delivered rather than discarded. If this feature is not used, UDP-Lite is semantically identical to UDP."*

2. The original congestion control algorithm in TCP was called Tahoe. Many other algorithms were introduced over the last decades. Name two of them that can be used without any knowledge about the TCP implementation on the receiver side and two that requires information about the receiver's TCP implementation.

   Vegas and CUBIC require no changes at the receiver side. TFRC and MaxNet require modifications on the receiver side as well.

Exercise 8: Do some research

3. The FTP protocol specification requires two ports. Why?

Exercise 8: Do some research

3. The FTP protocol specification requires two ports. Why?

   FTP needs two ports (one for sending and one for receiving) because it was originally designed to operate on Network Control Program (NCP), which was a simplex protocol that utilized two port addresses, establishing two connections, for two-way communications.

Exercise 8: Do some research

3. The FTP protocol specification requires two ports. Why?

   FTP needs two ports (one for sending and one for receiving) because it was originally designed to operate on Network Control Program (NCP), which was a simplex protocol that utilized two port addresses, establishing two connections, for two-way communications.

4. Explain the term *Open Relay*.

Exercise 8: Do some research

3. The FTP protocol specification requires two ports. Why?

   FTP needs two ports (one for sending and one for receiving) because it was originally designed to operate on Network Control Program (NCP), which was a simplex protocol that utilized two port addresses, establishing two connections, for two-way communications.

4. Explain the term *Open Relay*.

   A SMTP should only accept mails that it can deliver locally or forward (*relay*) mails from an authenticated and authorized user. If a mail server relays all mails of any user it its called an *open relay* and can be exploited to send spam mails.