Computer Networks

Exercise Session 13

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 02, 2024

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

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?

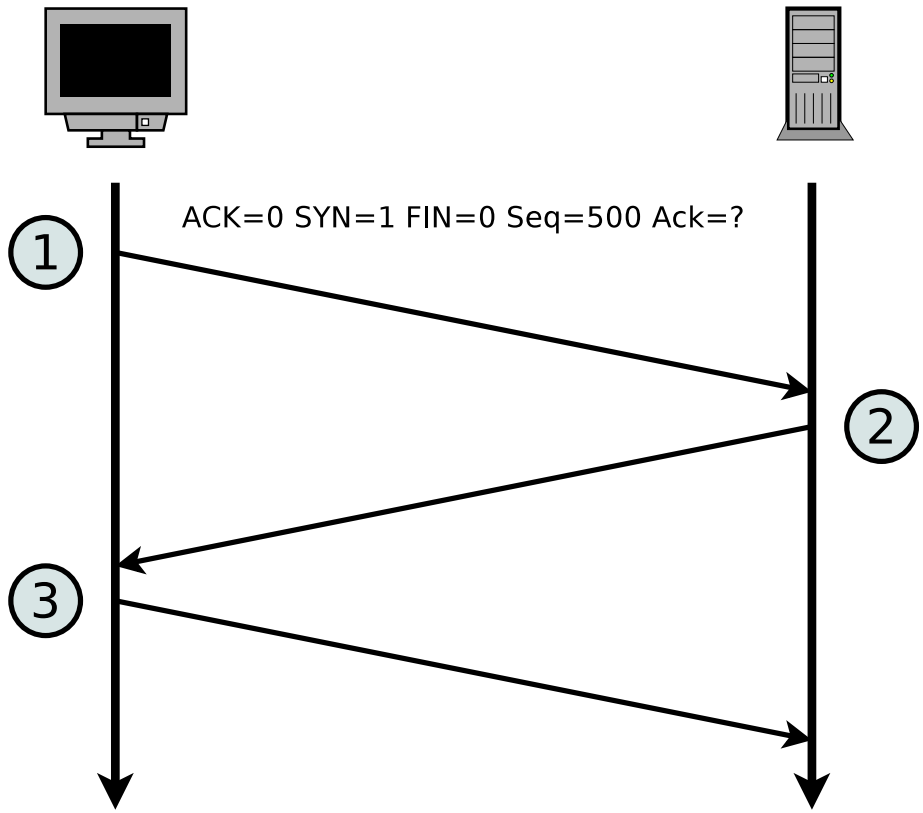
The receive window as sent by the receiver avoids congestion of the receiver, the congestion window maintained by the sender avoids congestion of the network.

11 Describe what the slow-start phase is.

The (initial) exponential growth phase.

Exercise 3: TCP Connections

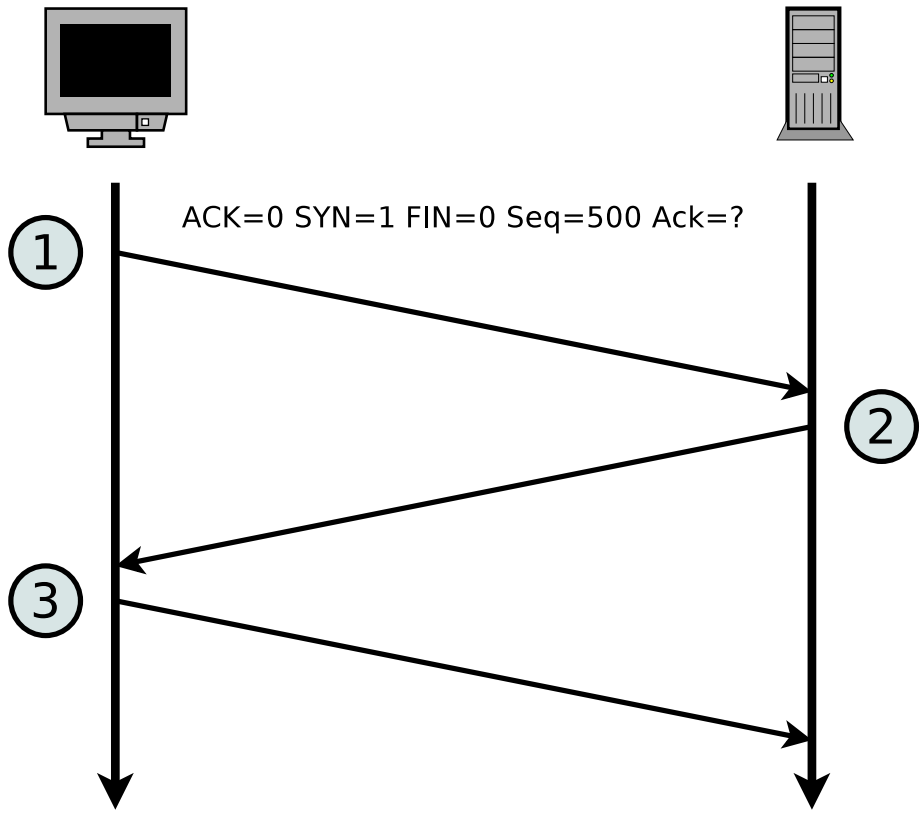
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



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

Exercise 3: TCP Connections

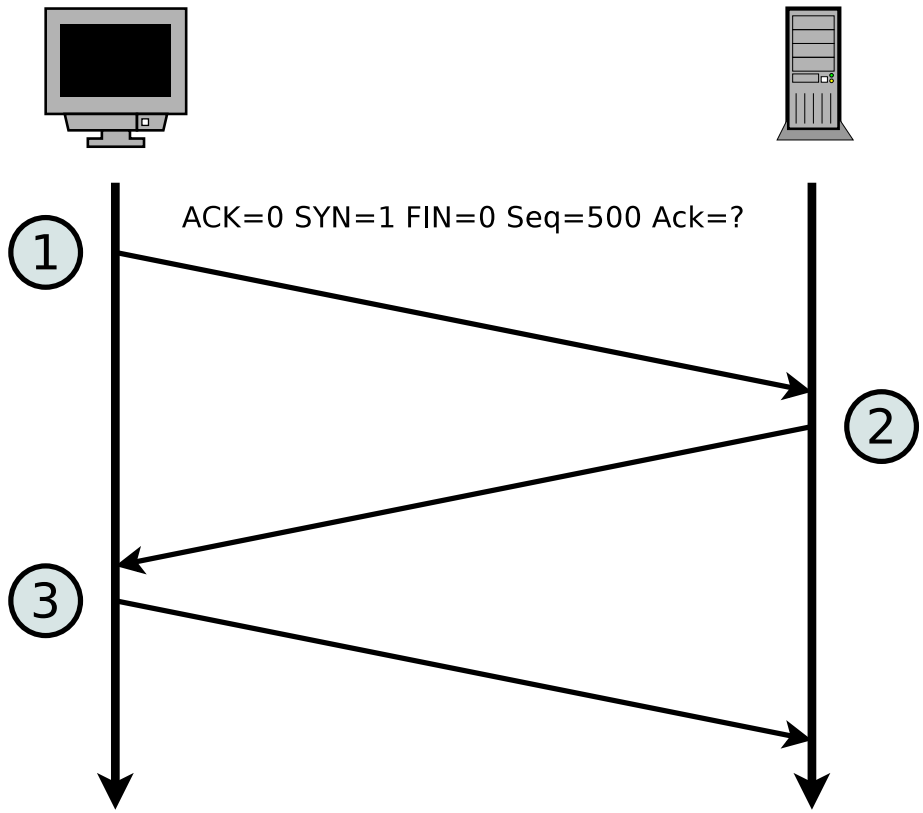
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



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

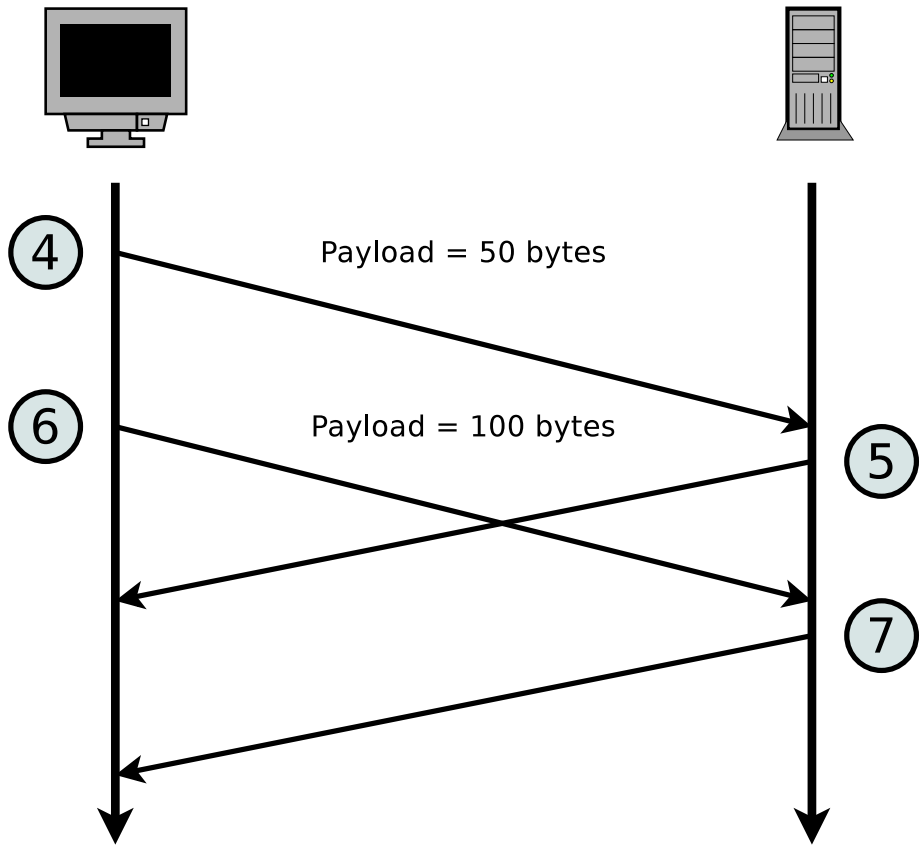
1. The diagram shows the establishment of a TCP connection. Complete the information in the table for TCP messages 2 and 3 according to TCP messages 1.



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

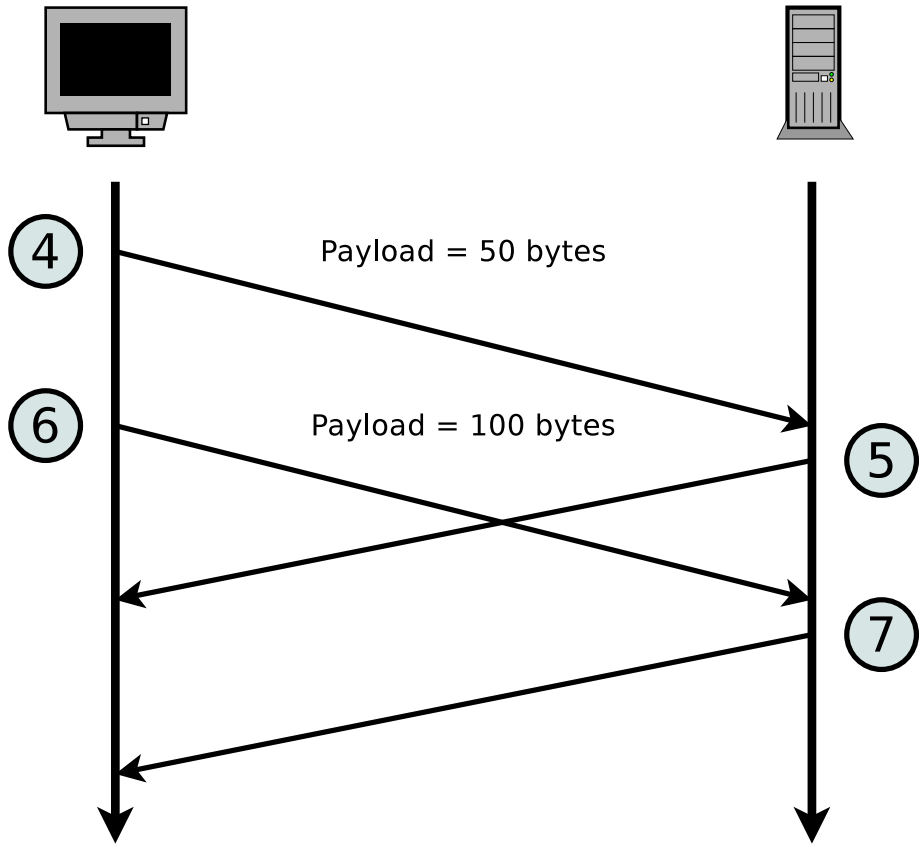
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



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

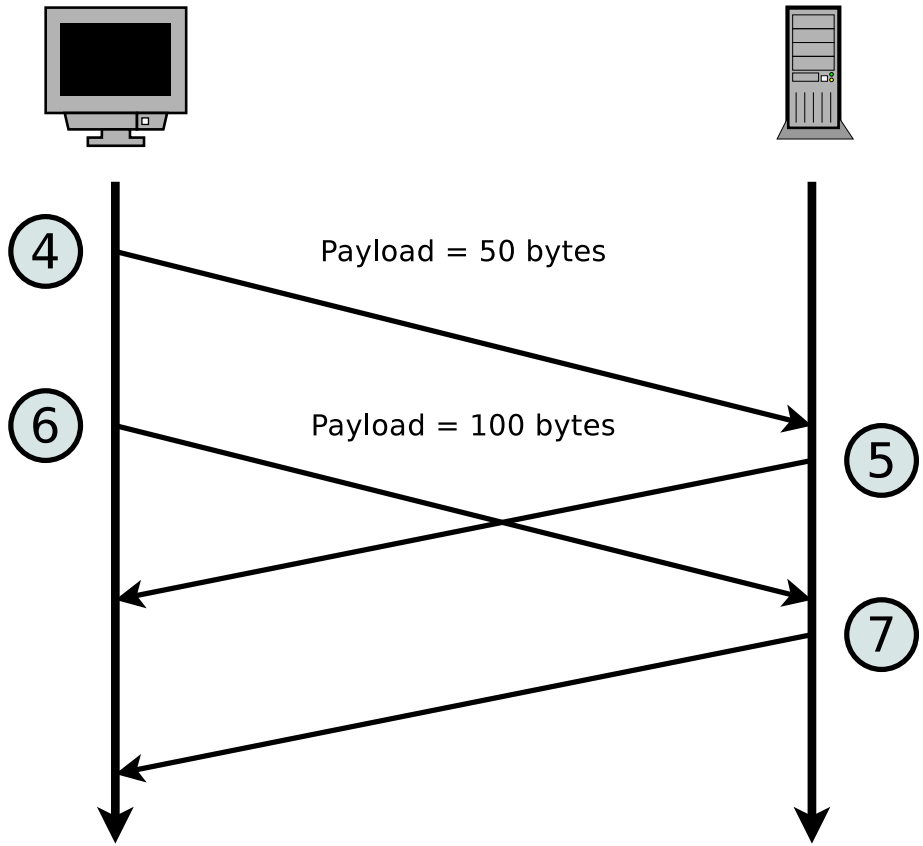
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



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

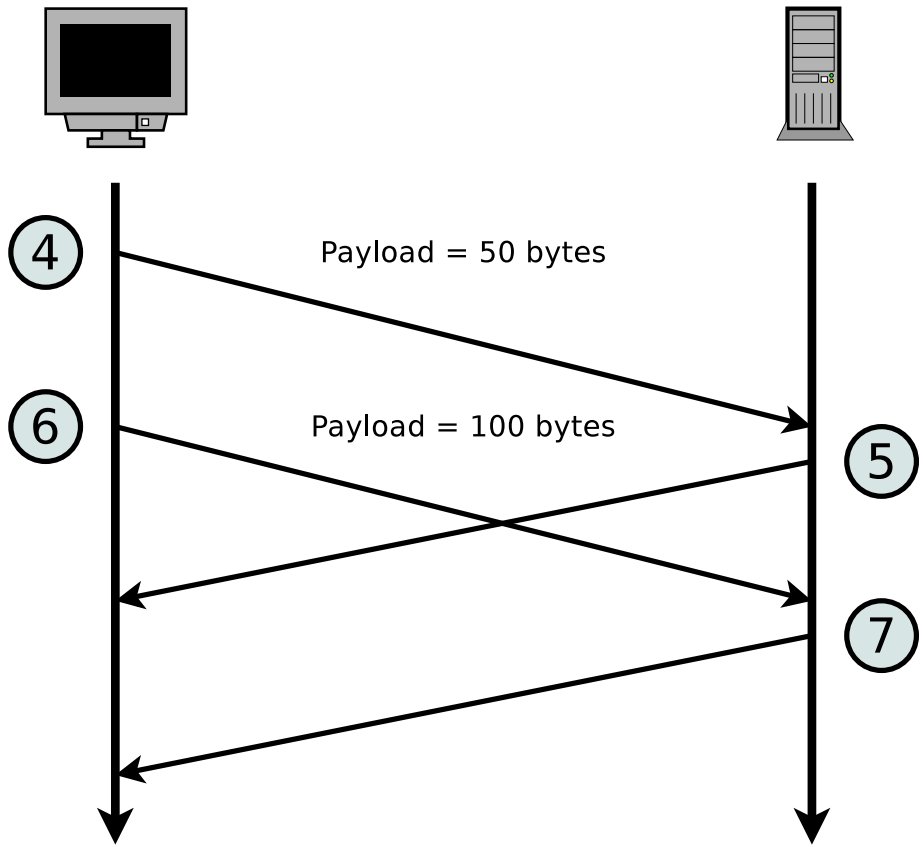
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



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

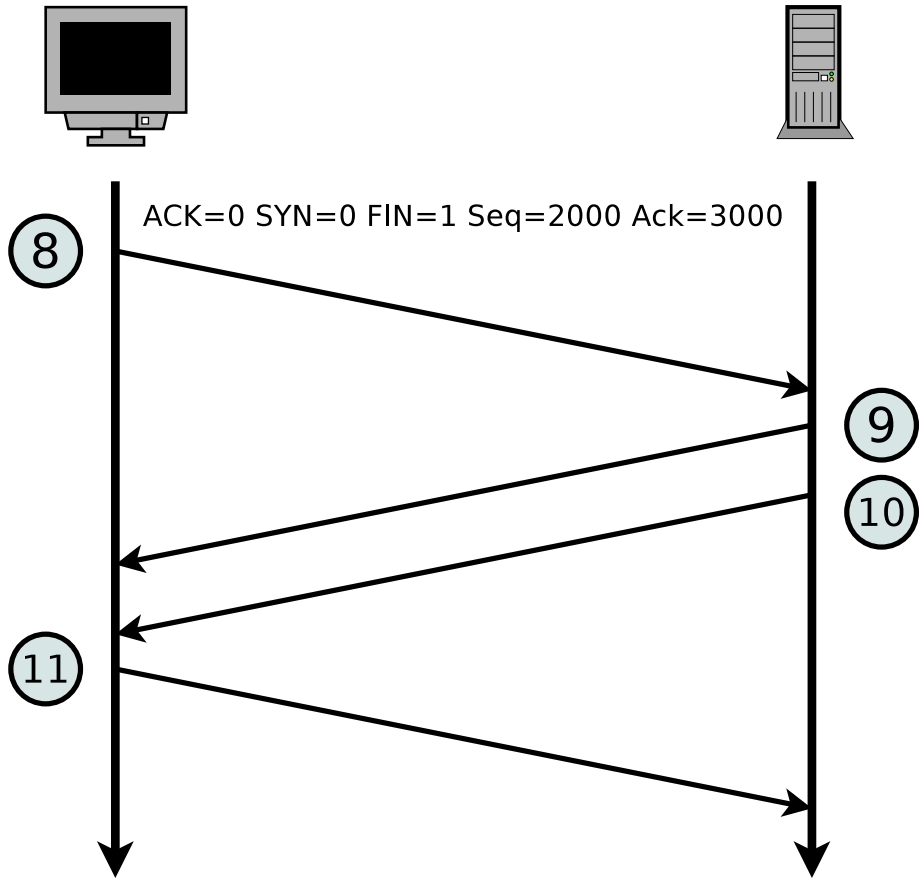
2 The diagram shows an excerpt of the transmission phase of a TCP connection. Complete the table.



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

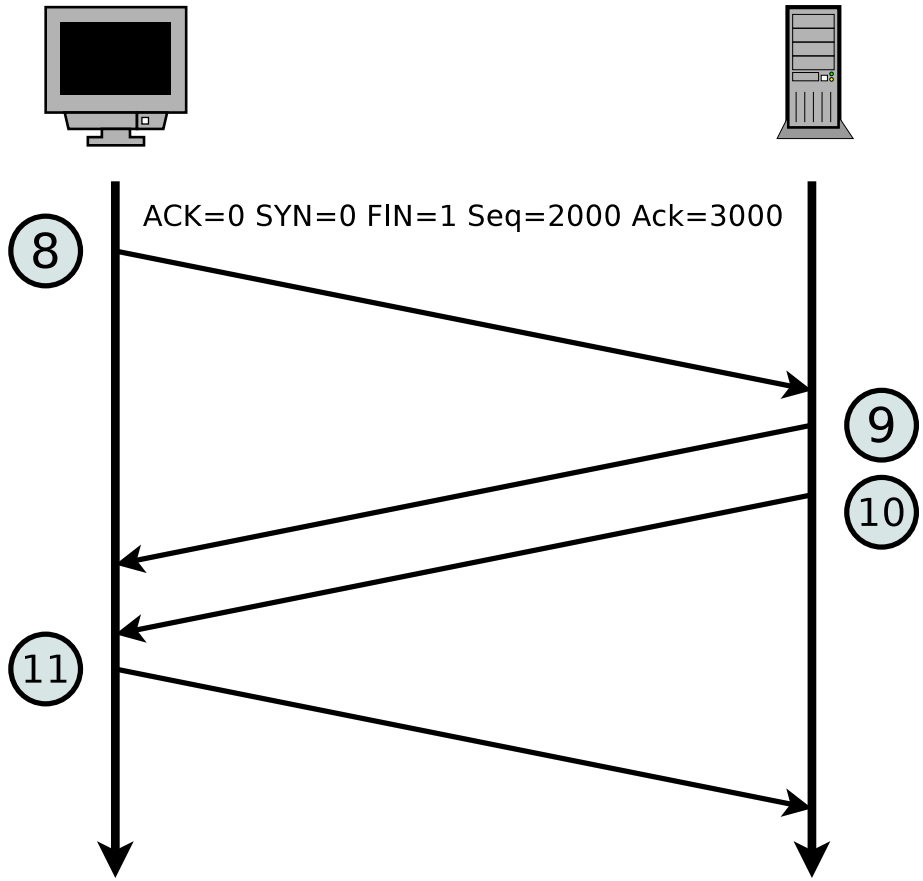
3 The diagram shows the termination of a TCP connection. Complete the table.



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

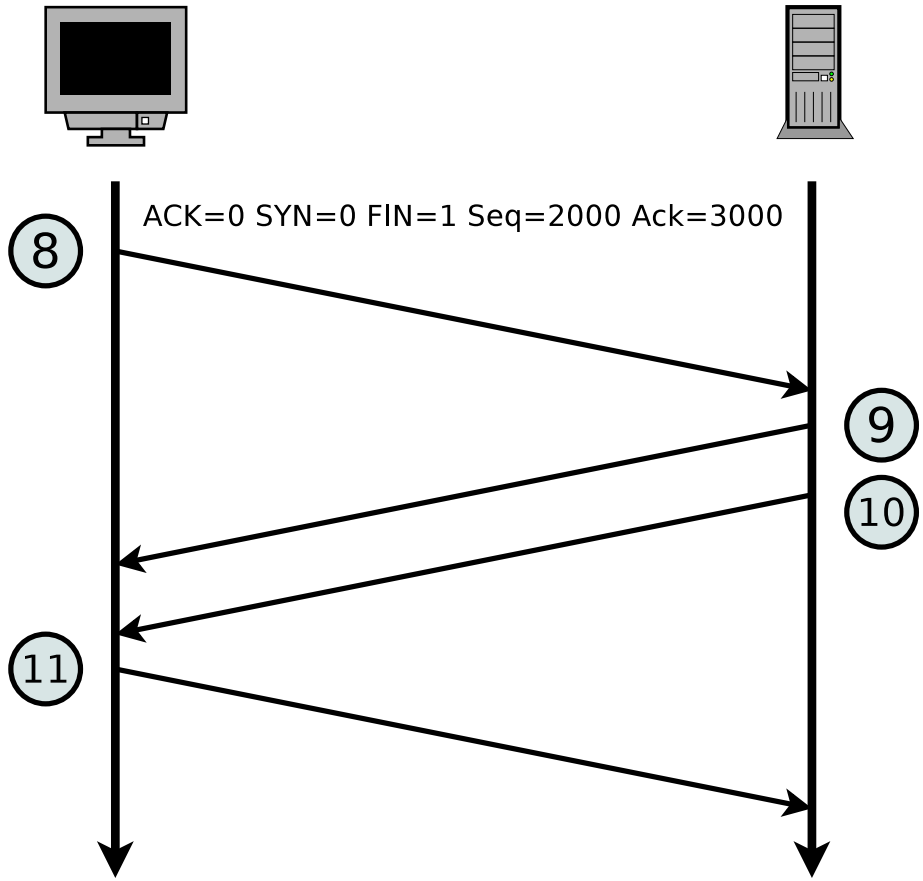
3 The diagram shows the termination of a TCP connection. Complete the table.



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

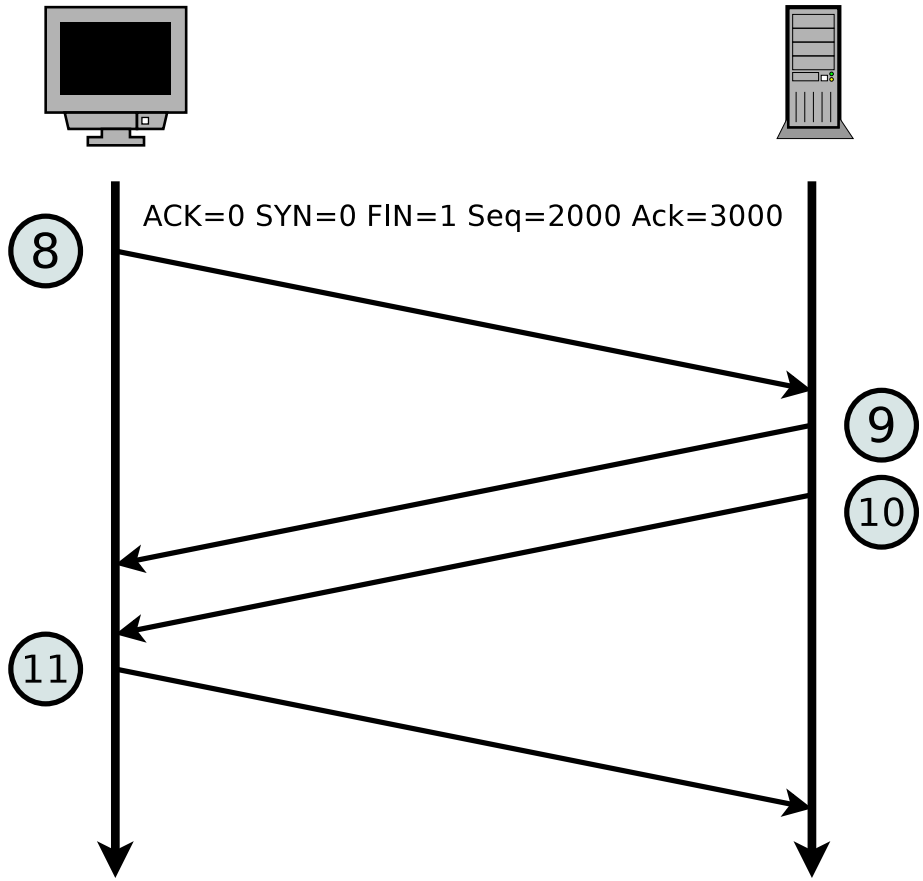
3 The diagram shows the termination of a TCP connection. Complete the table.



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

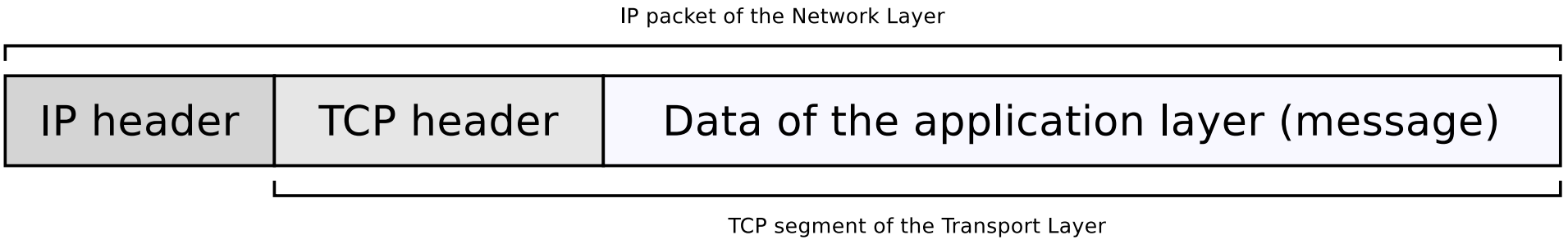
3 The diagram shows the termination of a TCP connection. Complete the table.



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

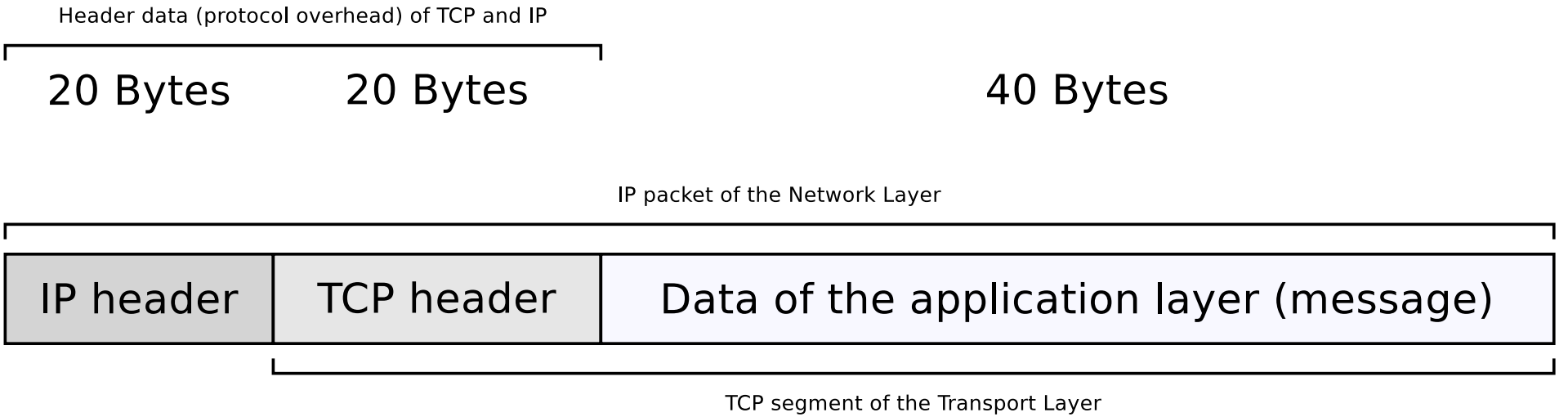
Exercise 4: Header and Payload

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 4: 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 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 You have learned that **sockets** can be either **stream sockets** or **datagram sockets**. However, during one of the exercises we have used a third type of socket. Explain where and why.

Exercise 8: Do some research

3 You have learned that **sockets** can be either **stream sockets** or **datagram sockets**. However, during one of the exercises we have used a third type of socket. Explain where and why.

Raw Sockets can be used to send Ethernet frames, IP packets, or ICMP packets. They require superuser privileges.

Exercise 8: Do some research

3. You have learned that **sockets** can be either **stream sockets** or **datagram sockets**. However, during one of the exercises we have used a third type of socket. Explain where and why.

   Raw Sockets can be used to send Ethernet frames, IP packets, or ICMP packets. They require superuser privileges.

4. Name an API that has been designed as an alternative to the **BSD socket API**.

Exercise 8: Do some research

3. You have learned that **sockets** can be either **stream sockets** or **datagram sockets**. However, during one of the exercises we have used a third type of socket. Explain where and why.

   Raw Sockets can be used to send Ethernet frames, IP packets, or ICMP packets. They require superuser privileges.

4. Name an API that has been designed as an alternative to the **BSD socket API**.

   Transport Layer Interface (TLI) has been introduced in UNIX System V Release 3. While BSD sockets were designed with TCP/IP in mind, TLI was focusing on the OSI reference model.