Transport Layer Protocols I

Course Code: CSC 3116 Course Title: Computer Networks

Dept. of Computer Science
Faculty of Science and Technology

Lecturer No:	6	**Week No:**	6	**Semester:**	Spring 201-21
Lecturer:	*Md. Sohidul Islam, Email: sohidul@aiub.edu*				

Lecture Outline

1. Introduction to transport layer
2. Types of services
3. User datagram protocol (UDP)
4. Transmission control protocol (TCP)
    - TCP Three-way handshaking
    - TCP error control scenarios

Transport Layer

Introduction

- The transport layer is located between the network layer and the application layer. The transport layer is responsible for providing services to the application layer; it receives services from the network layer.
- The transport layer, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.
- The transport layer is responsible for the delivery of a message from one process to another.

Transport Layer

Types of Services

- A transport layer protocol can either be connectionless or connection-oriented.
- What does connection mean?
  - A permanent connection (or session) between source and destination devices established prior to forwarding any traffic.
  - Session establishment prepares the devices to communicate with one another.
  - Through session establishment, the devices negotiate the amount of traffic that can be forwarded at a given time
  - The session is terminated only after all communication is completed.

Transport Layer Services

Connectionless Service

- In a connectionless service, the packets are sent from one party to another with no need for connection establishment or connection release.
- The packets are *not* numbered; they may be delayed or lost or may arrive out of sequence.
- There is no acknowledgment either.
- connectionless.

Connection Oriented *Service*

- In a connection-oriented service, a connection is first established between the sender and the receiver before data are transferred.
- At the end, the connection is released [1].

User Datagram Protocol (UDP)

- A connectionless, unreliable transport protocol.
- Does not add anything to the services of IP except to provide process-to process communication instead of host-to-host communication.
- Performs very limited error checking.
- Neither the client nor the server is obligated to keep track of the state of the communication session.
- UDP is not concerned with reliability or flow control.

User Datagram Protocol (UDP)

- Data may be lost or received out of sequence without any UDP mechanisms to recover or reorder the data.
- If reliability is required when using UDP as the transport protocol, it must be handled by the application [2].
- Due to low control overhead, limited error control and no flow control, it is faster.

User Datagram Protocol (UDP)

- Why UDP?
  - UDP is a very simple protocol using a minimum of overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP. Sending a small message by using UDP takes much less interaction between the sender and receiver than using TCP [2].

Transmission Control Protocol

- TCP, like UDP, is a process-to-process (program-to-program) protocol.
- TCP uses port numbers.
- Unlike UDP, TCP is a connection oriented protocol; it creates a virtual connection between two TCPs to send data.
- Uses extensive flow and error control mechanisms at the transport level.
- Slower compared to UDP because of flow and error control mechanism
- Delivery of data is guaranteed

Transmission Control Protocol

TCP Features

Numbering System: TCP software keeps track of the segments being transmitted or received, there are two fields called the sequence number and the acknowledgment number. These two fields refer to the byte number and not the segment number. The numbering does not necessarily start from O. Instead, TCP generates a random number between 0 and 2^{32}-1 for the number of the first byte.

- The bytes of data being transferred in each connection are numbered by TCP. The numbering starts with a randomly generated number.
- For example, if the random number happens to be 1057 and the total data to be sent are 6000 bytes, the bytes are numbered from 1057 to 7056.

Transmission Control Protocol

TCP Features

- Sequence: Number After the bytes have been numbered, TCP assigns a sequence number to each segment that is being sent. The sequence number for each segment is the number of the first byte carried in that segment.
- *Flow Control*: TCP, unlike UDP, provides *flow control.* The receiver of the data controls the amount of data that are to be sent by the sender. This is done to prevent the receiver from being submerged with data. The numbering system allows TCP to use a byte-oriented flow control.

Transmission Control Protocol

TCP Features

- *Error Control*: To provide reliable service, TCP implements an error control mechanism. Although error control considers a segment as the unit of data for error detection (loss or corrupted segments), error control is byte-oriented.
- *Congestion Control*: TCP, unlike UDP, takes into account congestion in the network. The amount of data sent by a sender is not only controlled by the receiver (flow control), but is also detennined by the level of congestion in the network [2].

Transmission Control Protocol

TCP Features

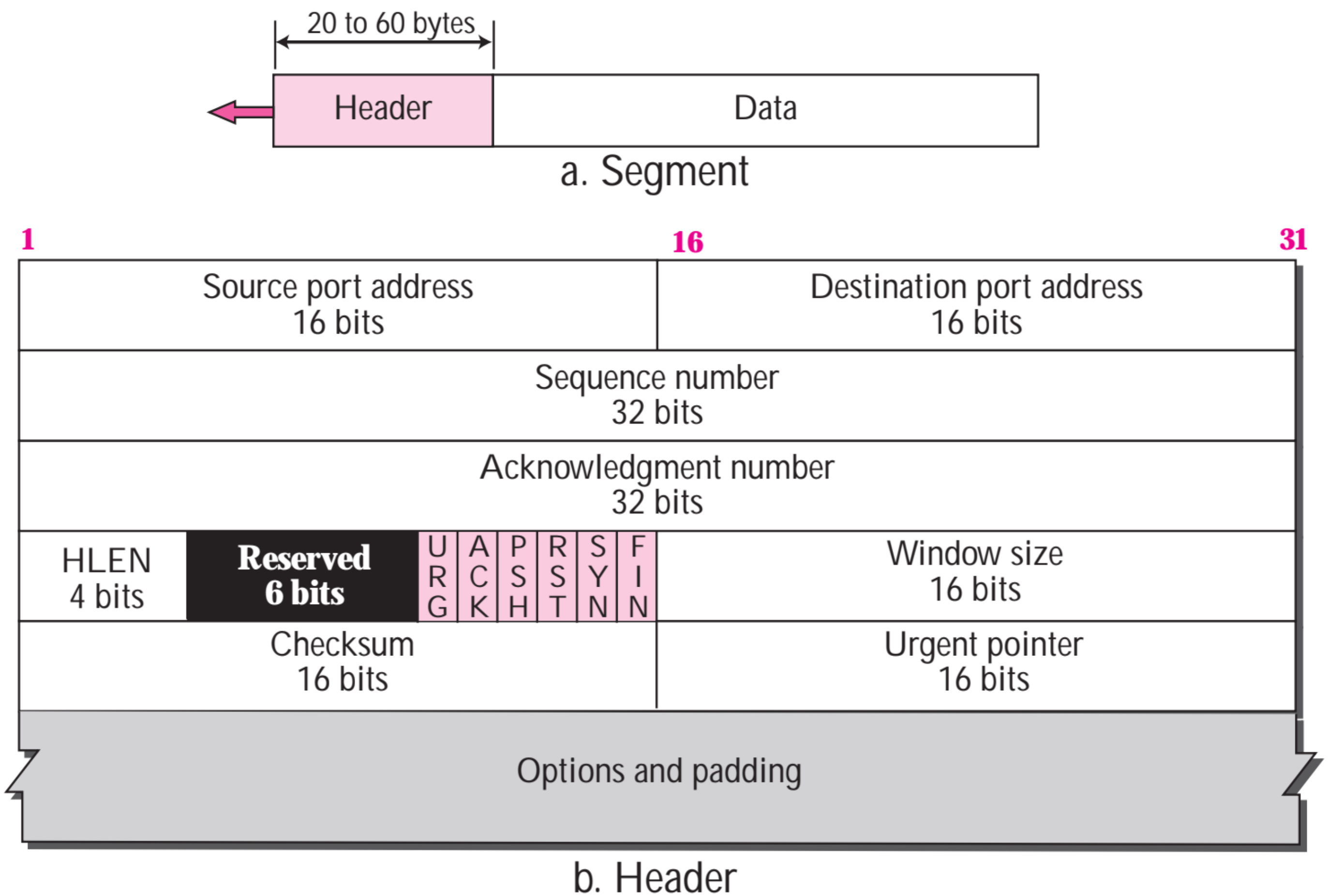

Fig. 1 TCP Segment Format [2]

Transmission Control Protocol

TCP Three-Way Handshaking

The three steps in this phase are as follows:

1. The client sends the first segment, a SYN segment, in which only the SYN flag is set. This segment is for synchronization of sequence numbers. It consumes one sequence number. When the data transfer starts, the sequence number is incremented by 1.

Transmission Control Protocol

TCP Three-Way Handshaking

1. The server sends the second segment, a SYN + ACK segment, with two flag bits set: SYN and ACK. This segment has a dual purpose. It is a SYN segment for communication in the other direction and serves as the acknowledgment for the SYN segment. It consumes one sequence number.
2. The client sends the third segment. This is just an ACK segment. It acknowledges the receipt of the second segment with the ACK flag and acknowledgment number field. Note that the sequence number in this segment is the same as the one in the SYN segment; the ACK segment does not consume any sequence numbers [2].

Scenarios of Error Control Mechanisms

TCP Three-Way Handshaking

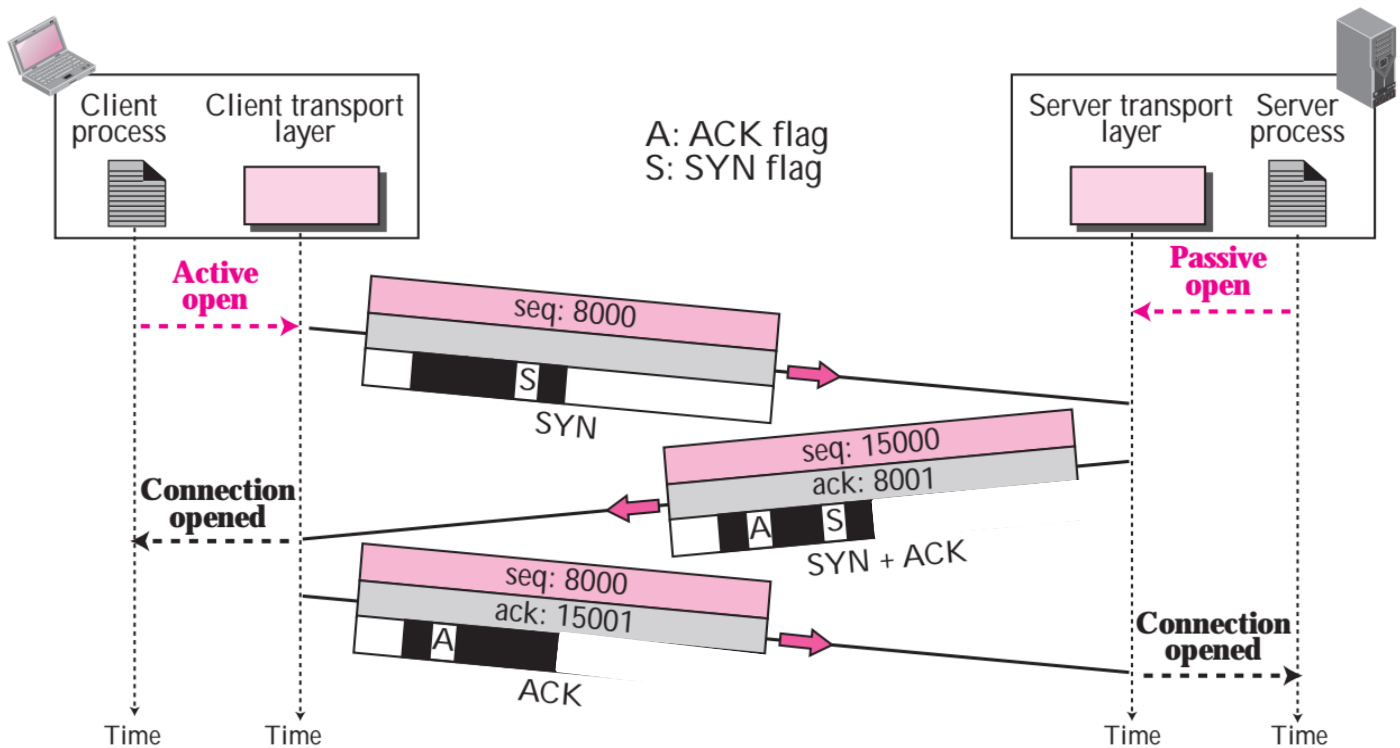

Fig. 2 Illustration of TCP three-way handshaking

Scenarios of Error Control Mechanisms

Lost acknowledgment

- Host A sends one segment to host B.
- Suppose that this segment has sequence number 92 and contains 8 bytes of data.
- After sending this segment, host A waits for a segment from B with acknowledgment number 100. Although the segment from A is received at B, the acknowledgment from B to A gets lost.

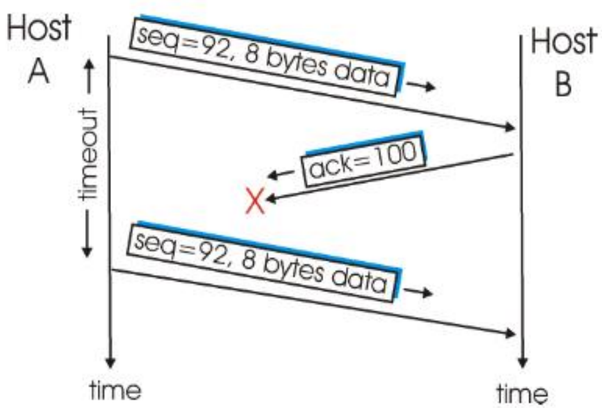

Fig 3: Retransmission due to a lost acknowledgment

Scenarios of Error Control Mechanisms

Lost acknowledgment

- In this case, the timer expires, and host A retransmits the same segment. Of course, when host B receives the retransmission, it will observe that the bytes in the segment duplicate bytes it has already deposited in its receive buffer.
- Thus TCP in host B will discard the bytes in the retransmitted segment [3].

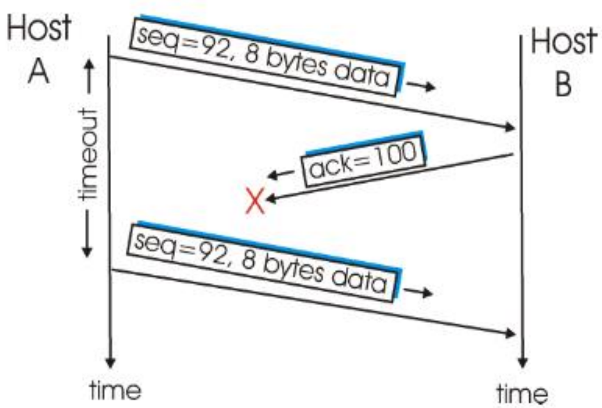

Fig 3: Retransmission due to a lost acknowledgment

Scenarios of Error Control Mechanisms

Acknowledgment arrives before the timeout

- Host A sends two segments back to back. The first segment has sequence number 55 and 5 bytes of data, and the second segment has sequence number 60 and 20 bytes of data.
- Suppose that both segments arrive intact at B, and B sends two separate acknowledgements (60, 80) for each of these segments.
- Suppose now that neither of the acknowledgements arrive at host A before the timeout of the first segment.

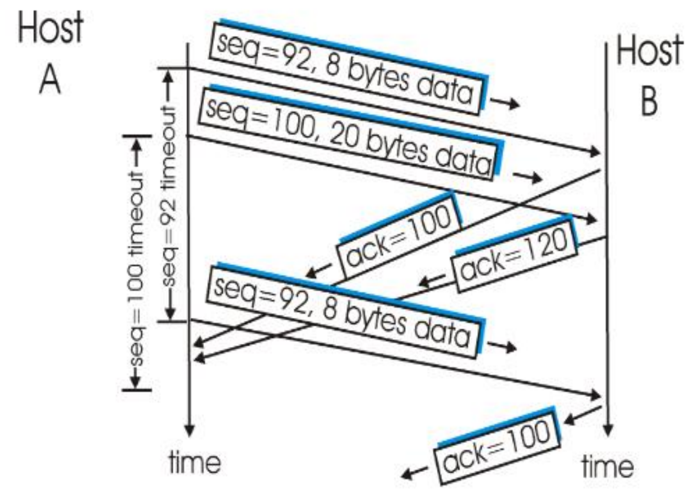

Fig 4 Segment is not retransmitted Because its acknowledgment arrives before the timeout.

Scenarios of Error Control Mechanisms

Acknowledgment arrives before the timeout

- When the timer expires, host A resends the first segment with sequence number 55.
- Now host A resends the segment only if the timer expires before the arrival of an acknowledgment with an acknowledgment number of 80 or greater.
- Thus, as shown in figure, if the second acknowledgment does not get lost and arrives before the timeout of the second segment, A does not resend the second segment [3].

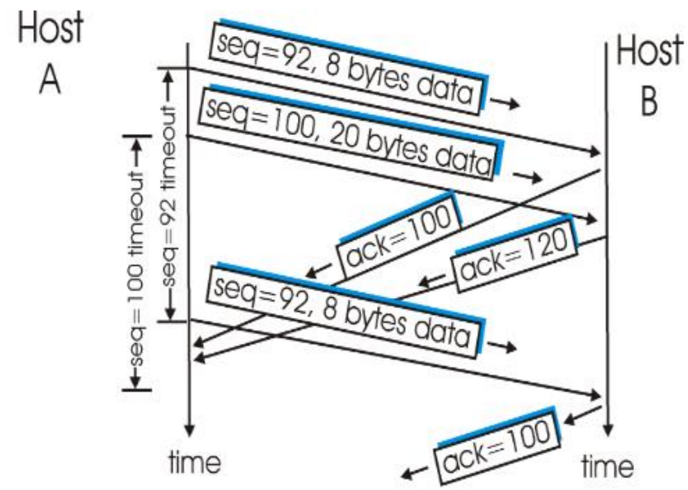

Fig 4 Segment is not retransmitted Because its acknowledgment arrives before the timeout.

Scenarios of Error Control Mechanisms

Cumulative acknowledgment

- Host A sends the two segments, exactly as in the second example.
- The acknowledgment of the first segment is lost in the network, but just before the timeout of the first segment, host A receives an acknowledgment with acknowledgment number 80.
- Host A therefore knows that host B has received everything up through byte 79; so host A does not resend either of the two segments.

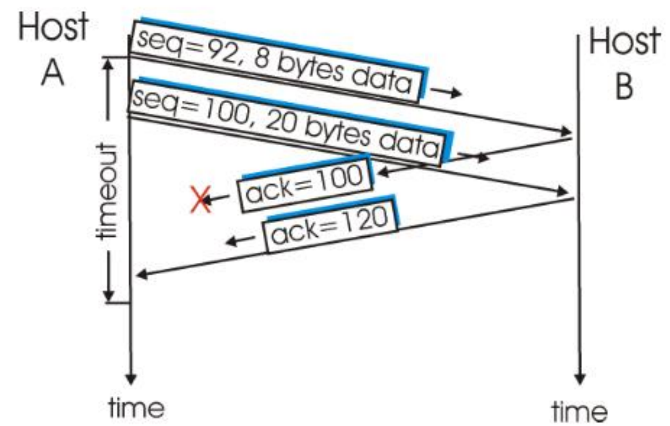

Fig: A cumulative acknowledgment avoids retransmission of first segment

Applications

- Where to use UDP?
  - Applications that can tolerate some data loss, but require little or no delay
  - Applications with simple request and reply transactions
  - Unidirectional communications where reliability is not required or can be handled by the application
  - Examples
    - All audio and video transmission (VoIP, IPTV)
    - DHCP
    - DNS (may also use TCP)
    - SNMP
    - TFTP (has own control mechanism, that TCP is not required)

Applications

- Where to use TCP
  - Applications that requires reliable data transfer and can tolerate delay
  - Example
    - HTTP
    - FTP
    - Telnet
    - SMTP

References

[1] My Reading Room, "Process-to-Process Delivery Concepts ," http://www.myreadingroom.co.in/notes-and-studymaterial/68-dcn/847-process-to-process-delivery-concepts.html", [Accessed: April. 22, 2020].

[2] B. A. Forouzan, *TCP/IP Protocol Suite*, *4*th ed., McGraw Hill Companies, Inc., USA, 2010, pp. 415-441.

[3] J. F., Kurose, K. W. Ross, *Computer Networking: A Top-Down Approach*, 7th ed., Pearson Education, Inc., USA, 2017, pp. 274-277.

Recommended Books

1. **Data Communications and Networking**, *B. A. Forouzan*, McGraw-Hill, Inc., Fourth Edition, 2007, USA.
2. **Computer Networking: A Top-Down Approach**, *J. F., Kurose, K. W. Ross*, Pearson Education, Inc., Sixth Edition, USA.
3. **Official Cert Guide CCNA 200-301 , vol. 1**, *W. Odom*, Cisco Press, First Edition, 2019, USA.
4. **CCNA Routing and Switching**, *T. Lammle*, John Wily & Sons, Second Edition, 2016, USA.
5. **TCP/IP Protocol Suite**, *B. A. Forouzan*, McGraw-Hill, Inc., Fourth Edition, 2009, USA.
6. **Data and Computer Communication**, *W. Stallings*, Pearson Education, Inc., Tenth Education, 2013, USA.