

# Fundamentals of Telecommunications Networks

ECP 602

Eng. Rasha Samir

Electronics and Electrical Communications Dept.  
Cairo University



## Transport Layer overview

# Lecture Objectives

Upon completion of this lecture, you will be able to:

- ❖ Describe the purpose of the transport layer in managing the transportation of data in end-to-end communication.
- ❖ Describe characteristics of the TCP and UDP protocols, including port numbers and their uses.
- ❖ Explain how TCP session establishment and termination processes facilitate reliable communication.
- ❖ Explain how TCP protocol data units are transmitted and acknowledged to guarantee delivery.
- ❖ Explain the UDP client processes to establish communication with a server.
- ❖ Determine whether high-reliability TCP transmissions, or non-guaranteed UDP transmissions, are best suited for common applications.

# Lecture Overview

Introduction

Transport Layer Protocols

TCP and UDP

Summary

# Transport Layer Protocols

# Role of the Transport Layer

The transport layer is responsible for establishing a temporary communication session between two applications and delivering data between them.

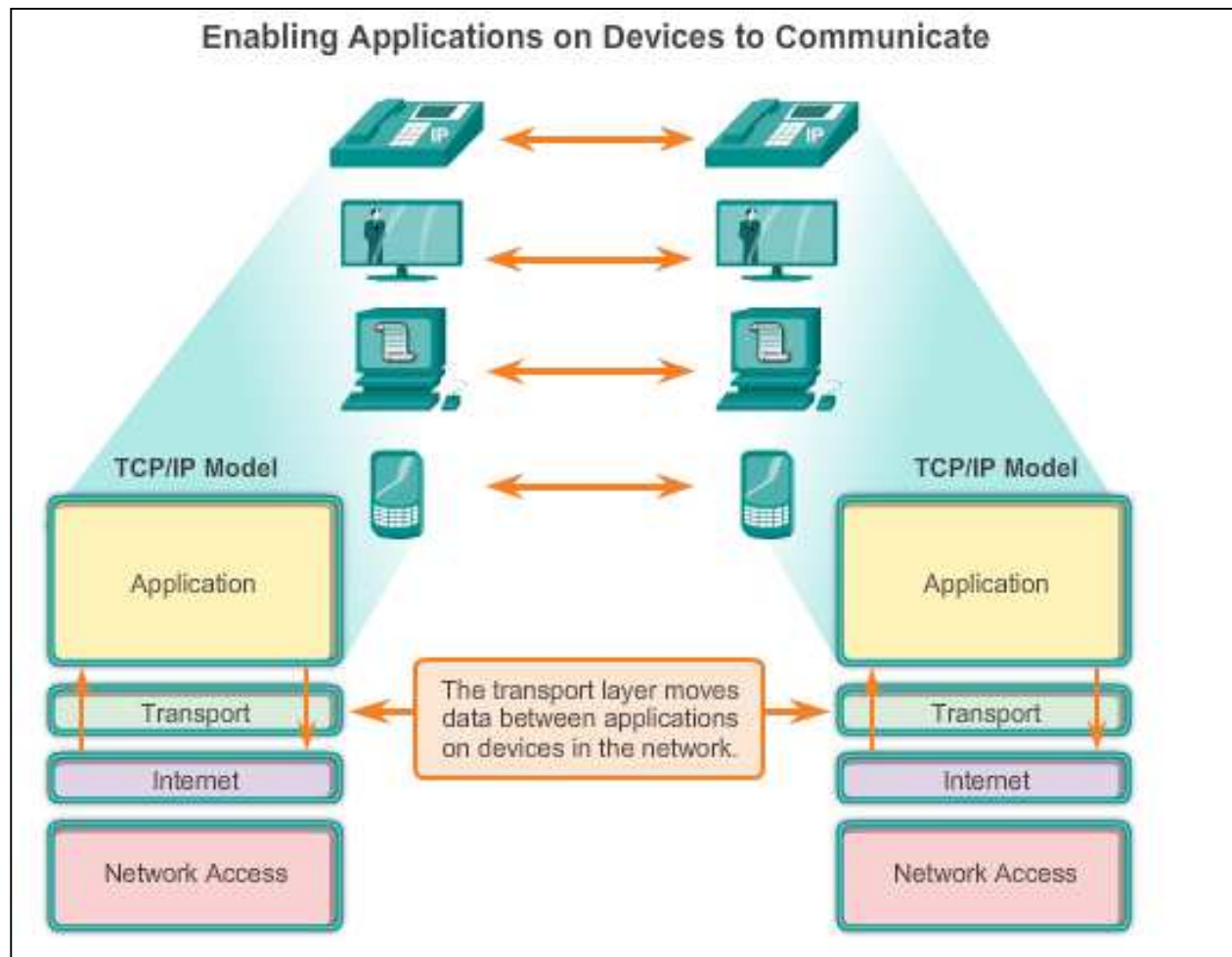
TCP/IP uses two protocols to achieve this:

- ❖ Transmission Control Protocol (TCP)
- ❖ User Datagram Protocol (UDP)

## **Primary Responsibilities of Transport Layer Protocols**

- ❖ Tracking the individual communication between applications on the source and destination hosts
- ❖ Segmenting data for manageability and reassembling segmented data into streams of application data at the destination
- ❖ Identifying the proper application for each communication stream

# Role of the Transport Layer (Cont.)



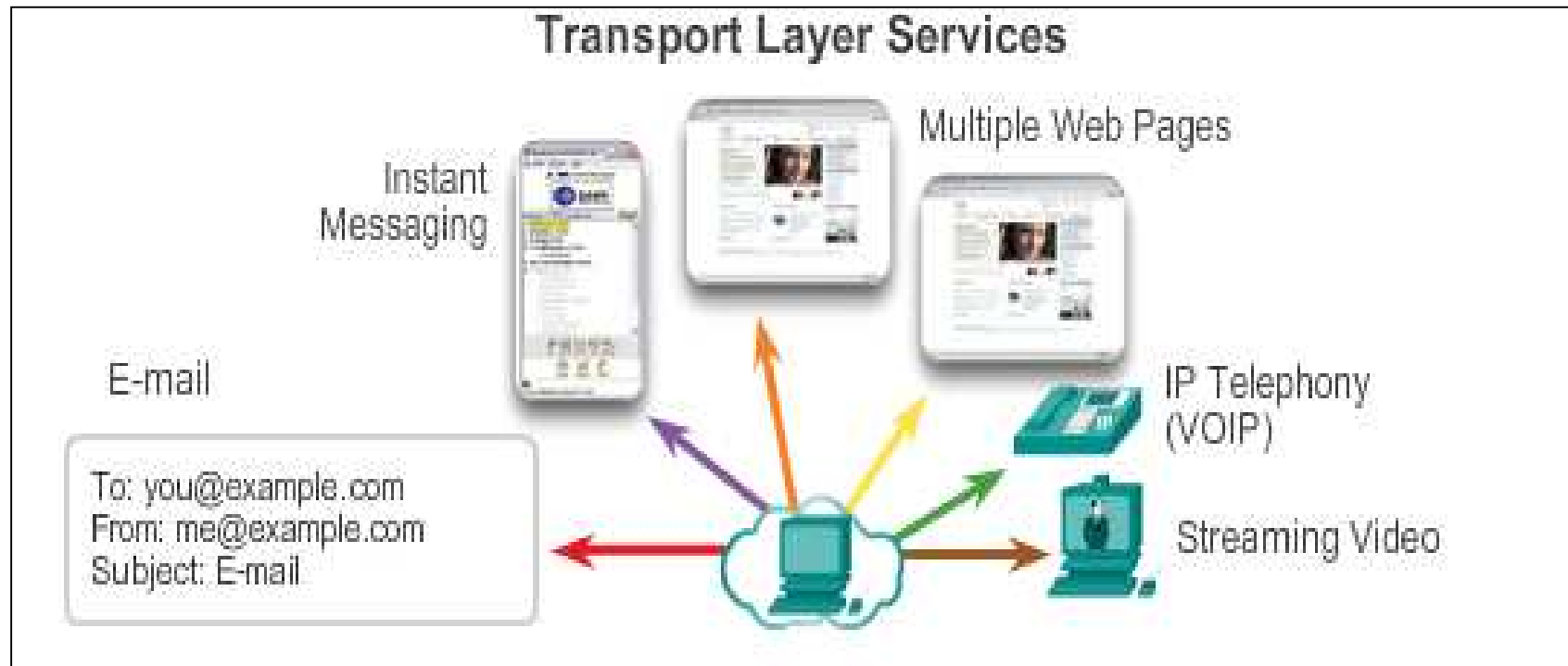
# Conversation Multiplexing

## **Segmenting the Data**

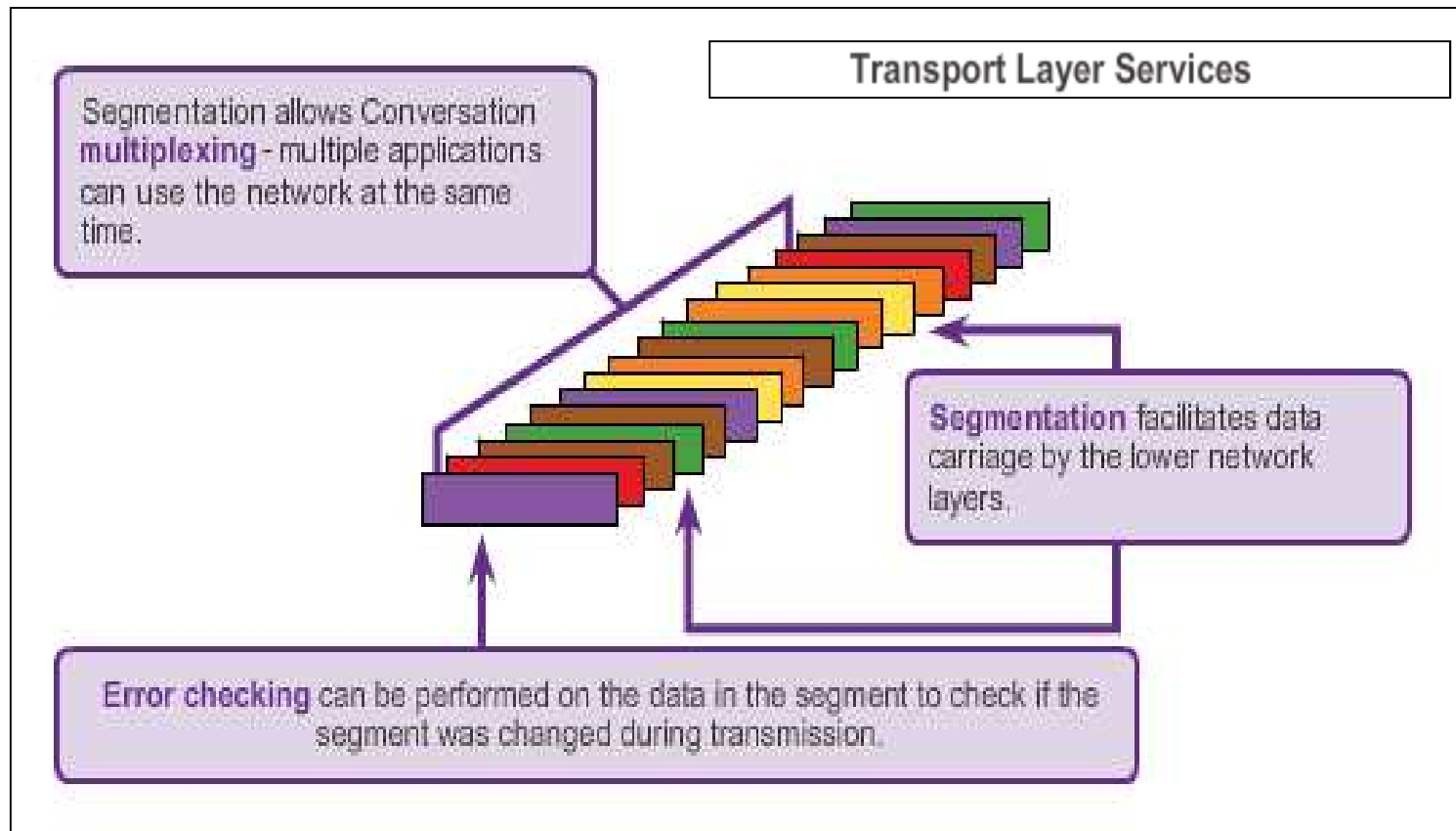
- ❖ Enables many different communications, from many different users, to be interleaved (multiplexed) on the same network, at the same time.
- ❖ Provides the means to both send and receive data when running multiple applications.
- ❖ Header added to each segment to identify it.



# Conversation Multiplexing (Cont.)



## Conversation Multiplexing (Cont.)



# Transport Layer Reliability

Different applications have different transport reliability requirements.

TCP/IP provides two transport layer protocols, **TCP** and **UDP**.

## **TCP**

- ❖ Provides reliable delivery ensuring that all of the data arrives at the destination.
- ❖ Uses acknowledged delivery and other processes to ensure delivery
- ❖ Makes larger demands on the network - more overhead.

## **UDP**

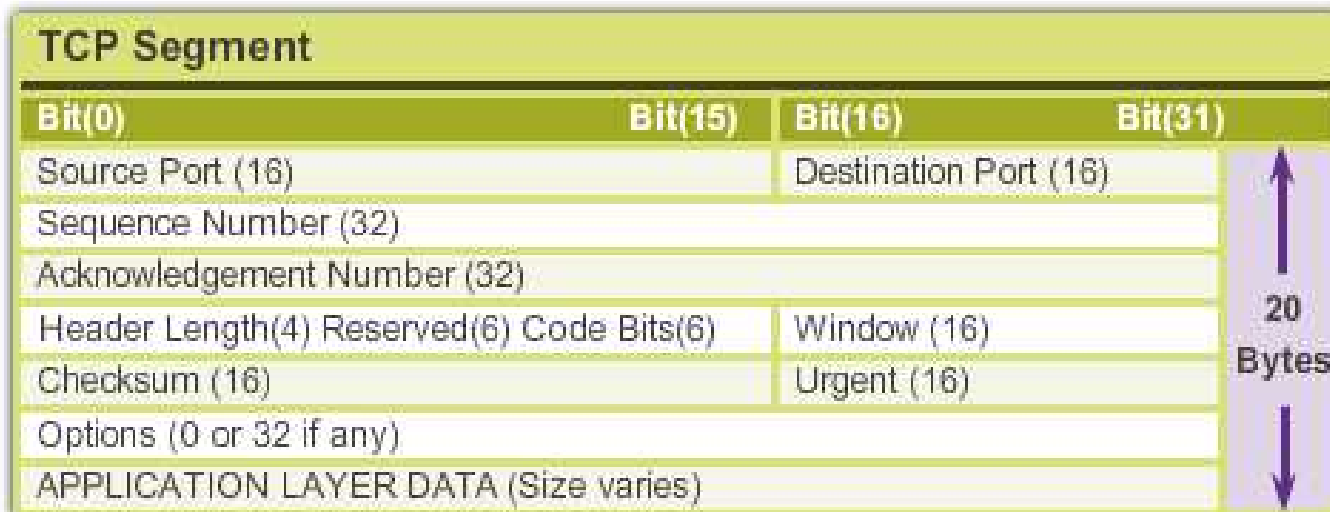
- ❖ Provides just the basic functions for delivery - no reliability.
- ❖ Less overhead.

## **TCP or UDP**

- ❖ There is a trade-off between the value of reliability and the burden it places on the network.
- ❖ Application developers choose the transport protocol based on the requirements of their applications.

# Introducing TCP

- ❖ Defined in RFC 793
- ❖ Connection-oriented - Creates a session between the source and destination
- ❖ Reliable delivery - Retransmits lost or corrupt data
- ❖ Ordered data reconstruction - Reconstructs numbering and sequencing of segments
- ❖ Flow control - Regulates the amount of data transmitted
- ❖ Stateful protocol - Tracks the session

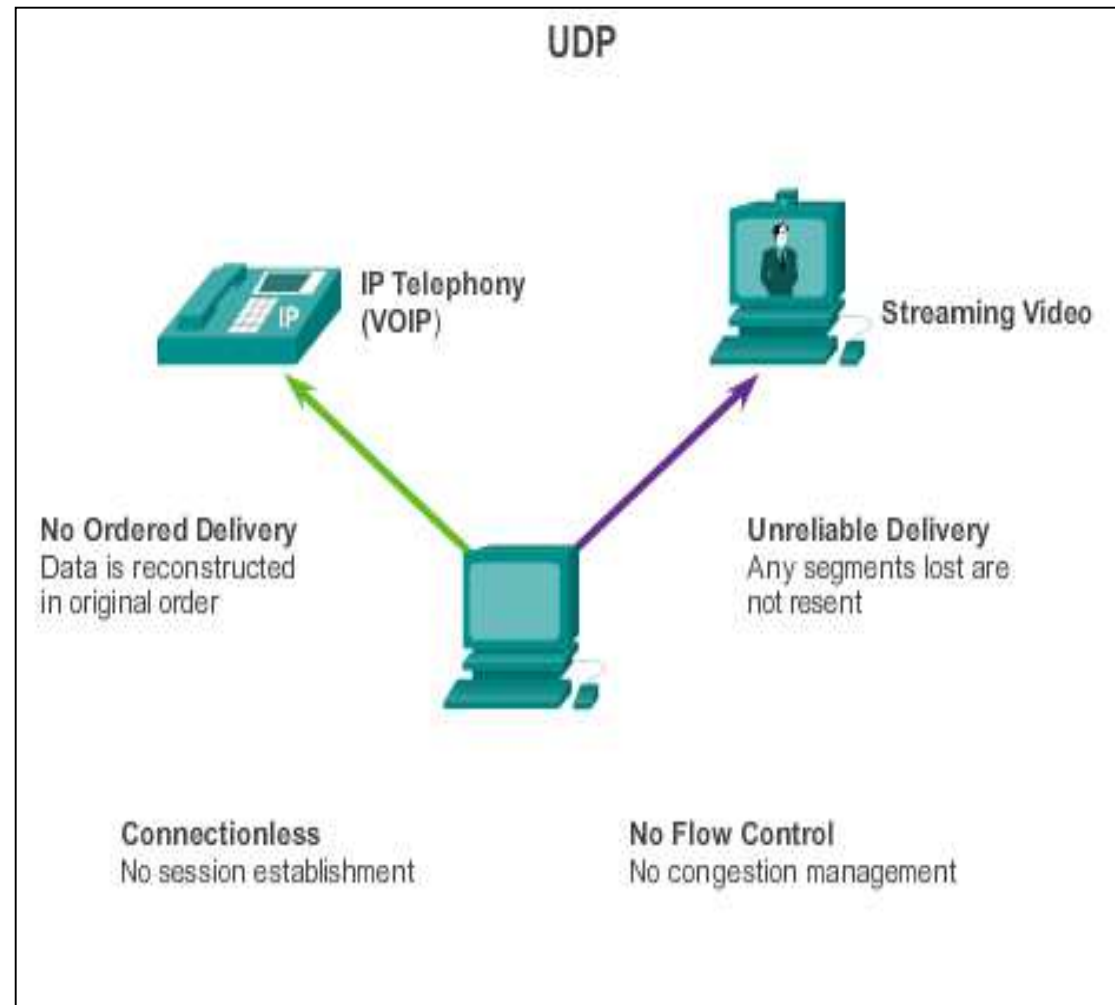


# Introducing UDP

- ❖ RFC 768
- ❖ Connectionless
- ❖ Unreliable delivery
- ❖ No ordered data reconstruction
- ❖ No flow control
- ❖ Stateless protocol

Applications that use UDP:

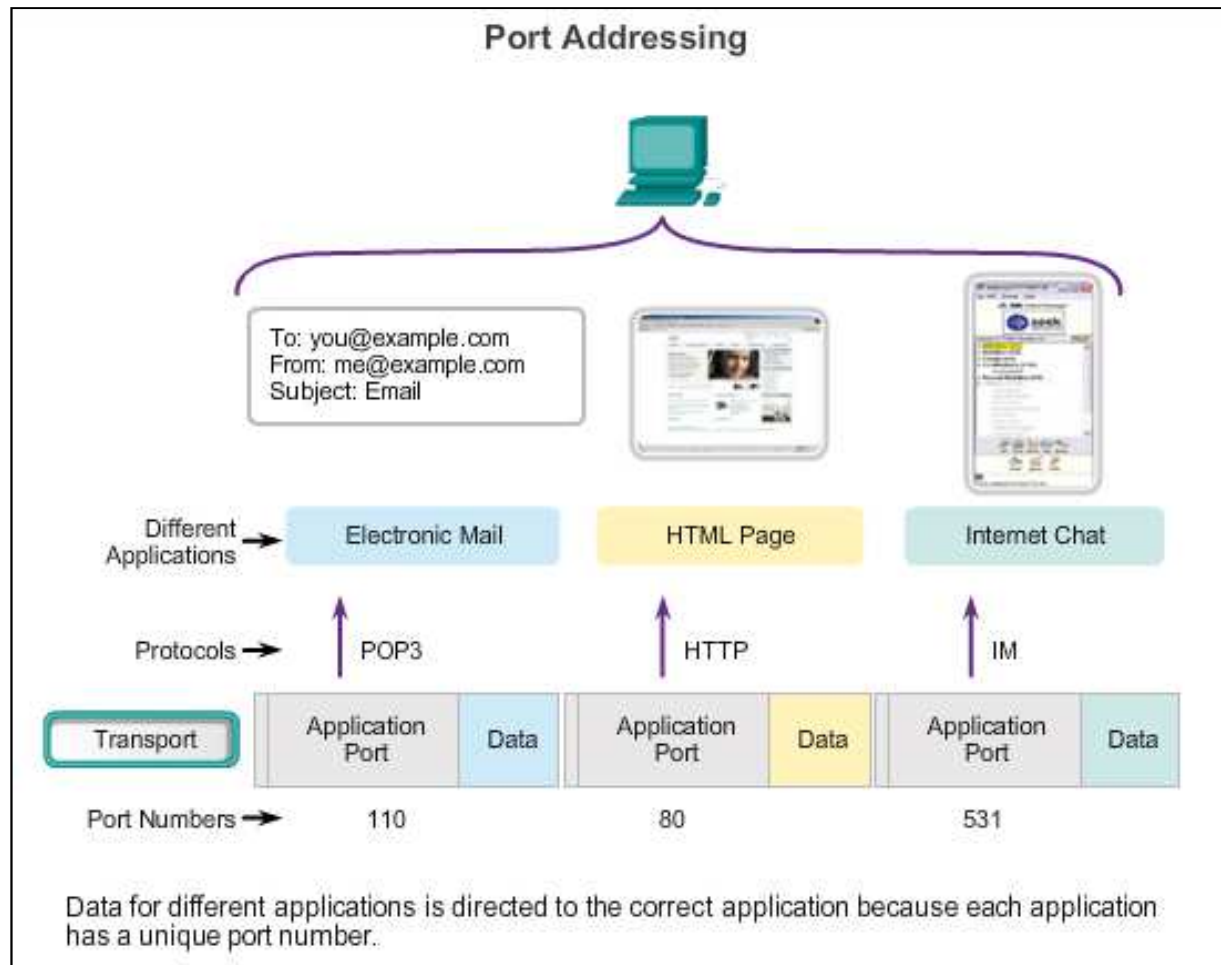
- ❖ Domain Name System (DNS)
- ❖ Video Streaming
- ❖ VoIP



## Introducing TCP and UDP

# Separating Multiple Communications

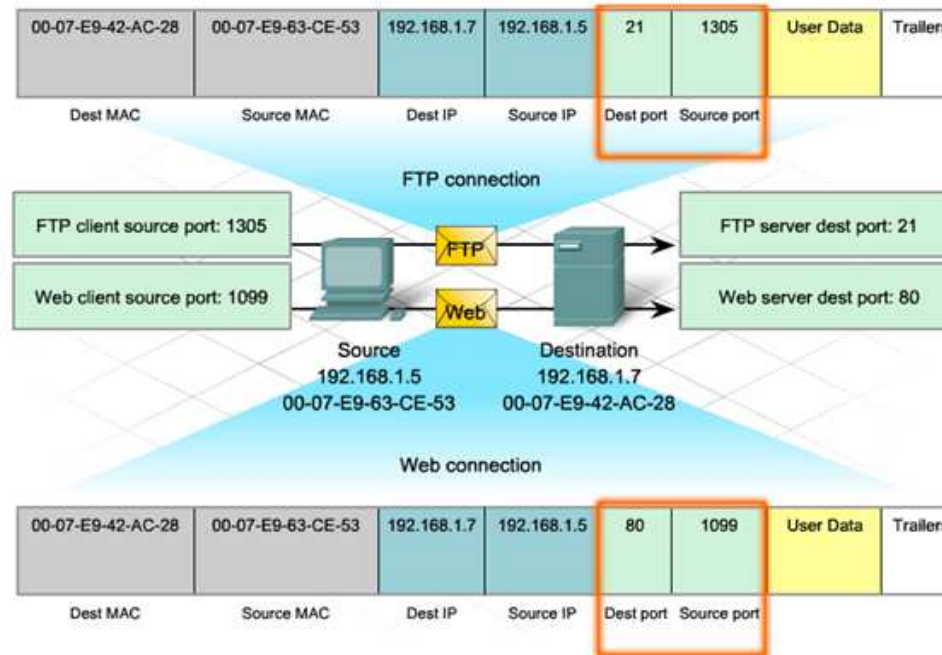
TCP and UDP use port numbers to differentiate between applications.



## Introducing TCP and UDP

# TCP and UDP Port Addressing

7.1.2.6: TCP and UDP Port Addressing



# TCP and UDP Port Addressing (Cont.)

## Port Numbers

Port Number Range	Port Group
0 to 1023	Well Known (Contact) Ports
1024 to 49151	Registered Ports
49152 to 65533	Private and/or Dynamic Ports

### Registered TCP Ports:

1863	MSN Messenger
2000	Cisco SCCP (VoIP)
8008	Alternate HTTP
8080	Alternate HTTP

### Well Known TCP Ports:

21	FTP
23	Telnet
25	SMTP
80	HTTP
110	POP3
194	Internet Relay Chat (IRC)
443	Secure HTTP (HTTPS)



## TCP and UDP Port Addressing (Cont.)

### Registered UDP Ports:

1812	RADIUS Authentication Protocol
5004	RTP (Voice and Video Transport Protocol)
5040	SIP (VoIP)

### Registered TCP/UDP Common Ports:

1433	MS SQL
2948	WAP (MMS)

### Well Known UDP Ports:

69	TFTP
520	RIP

### Well Known TCP/UDP Common Ports:

53	DNS
161	SNMP
531	AOL Instant Messenger, IRC

## TCP and UDP Port Addressing (Cont.)

Netstat is used to examine TCP connections that are open and running on a networked host.

```
C:\>netstat
```

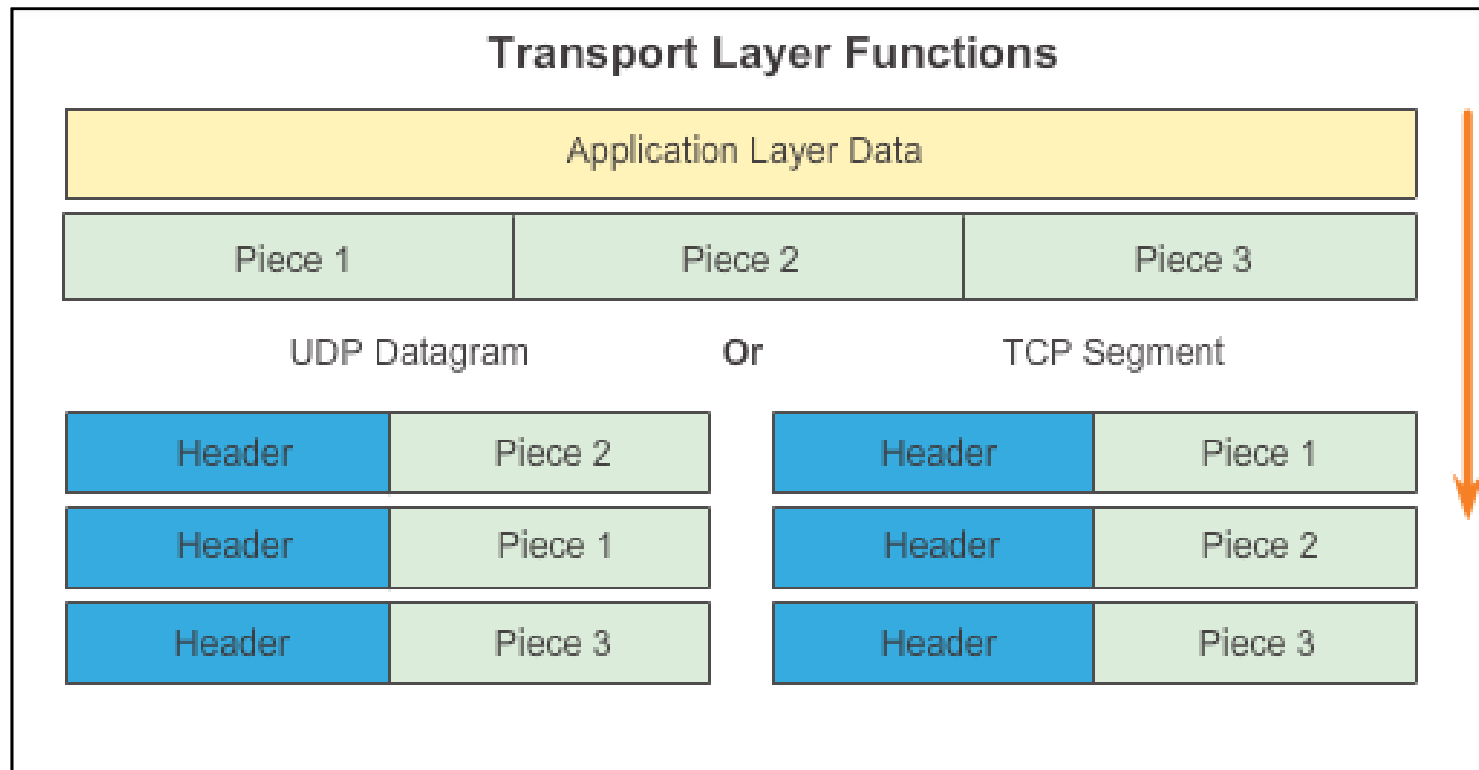
```
Active Connections
```

Proto	Local Address	Foreign Address	State
<b>TCP</b>	kenpc:3126	192.168.0.2:netbios-ssn	ESTABLISHED
TCP	kenpc:3158	207.138.126.152:http	ESTABLISHED
TCP	kenpc:3159	207.138.126.169:http	ESTABLISHED
TCP	kenpc:3160	207.138.126.169:http	ESTABLISHED
TCP	kenpc:3161	sc.msn.com:http	ESTABLISHED
TCP	kenpc:3166	www.cisco.com:http	ESTABLISHED

```
C:\>
```

# TCP and UDP Segmentation

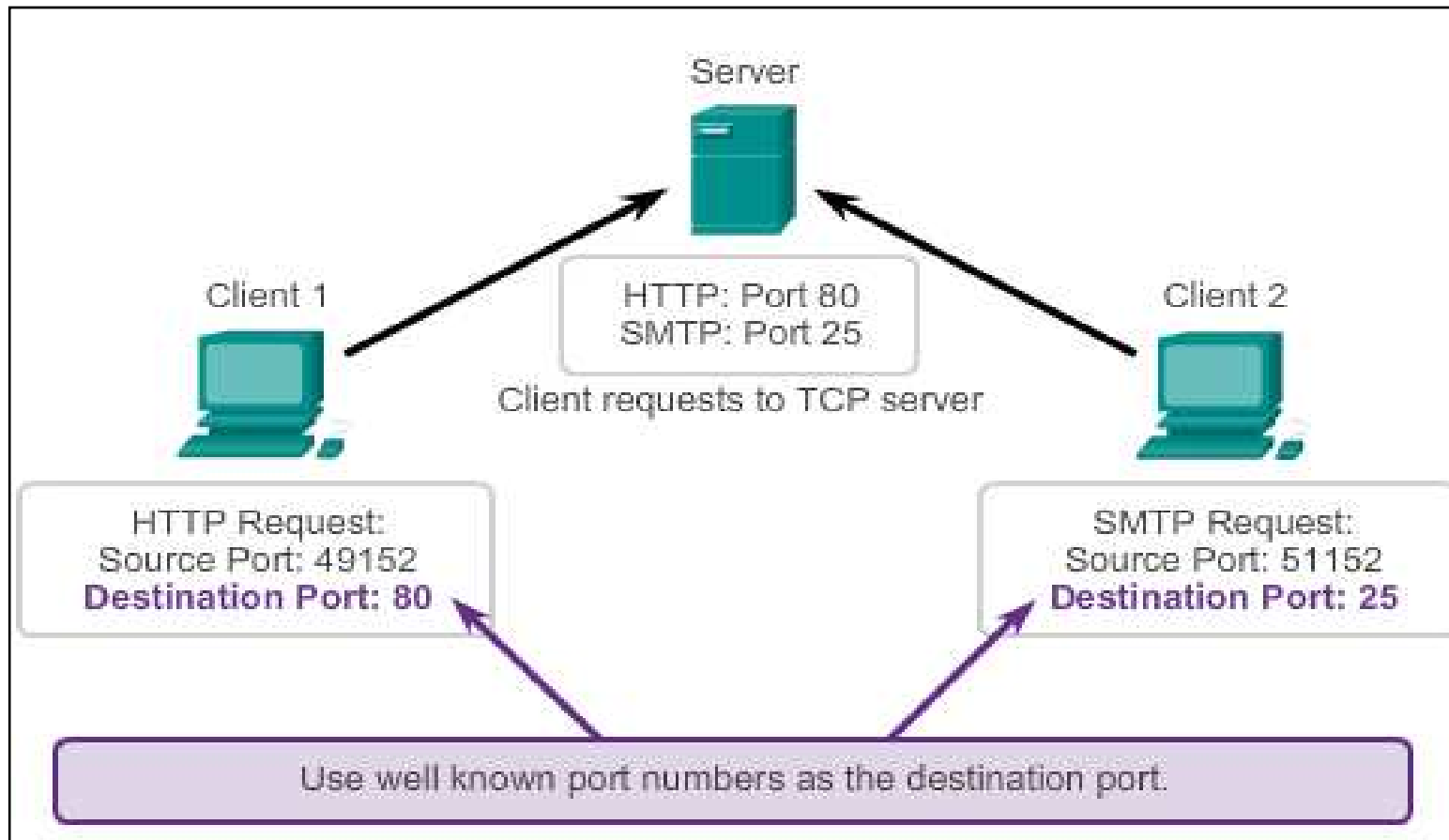
The transport layer divides the data into pieces and adds a header for delivery over the network



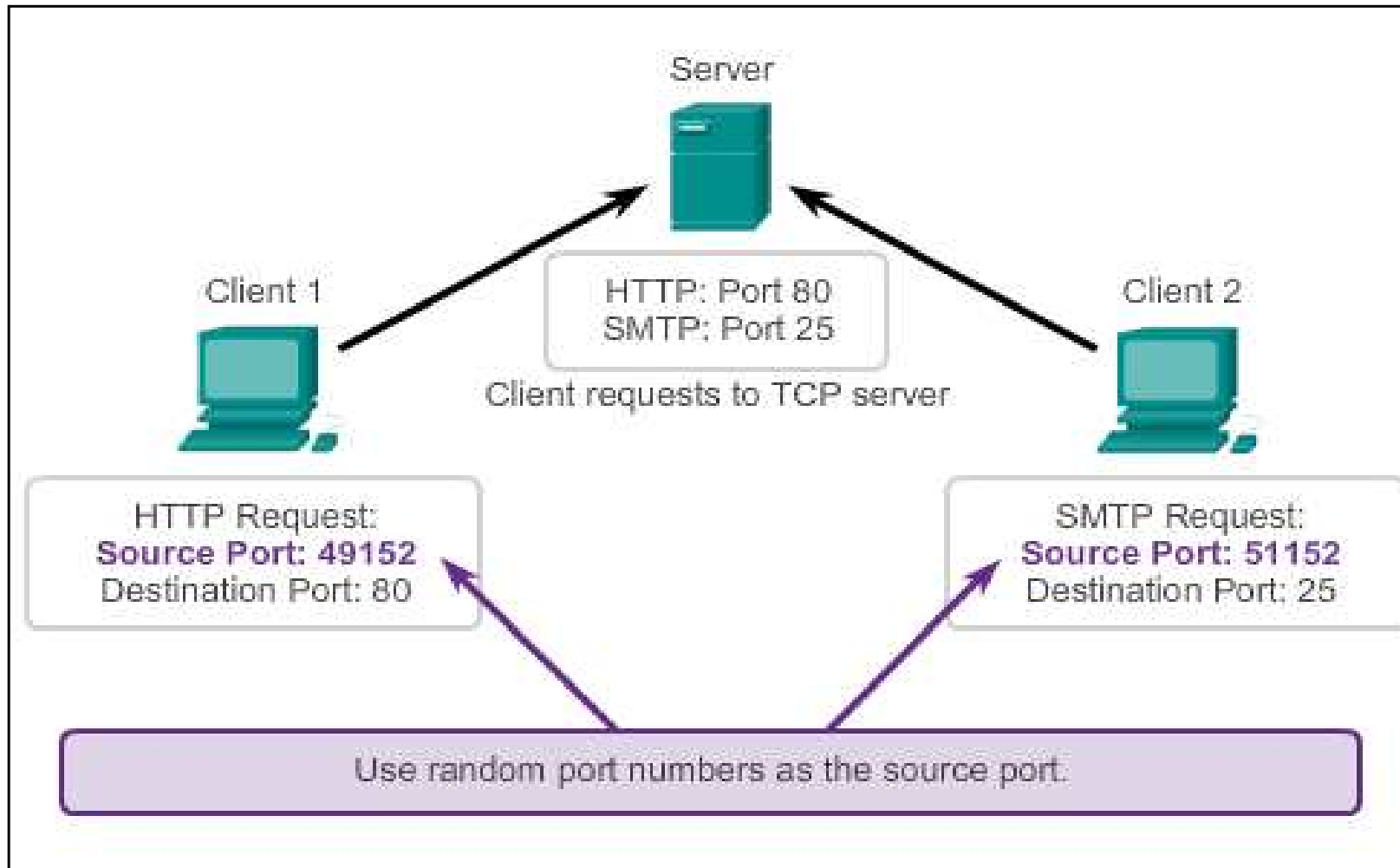
## TCP and UDP

# TCP Server Processes

## Request Destination Ports



## TCP Server Processes (Cont.)



## TCP Communication

# TCP Connection, Establishment and Termination

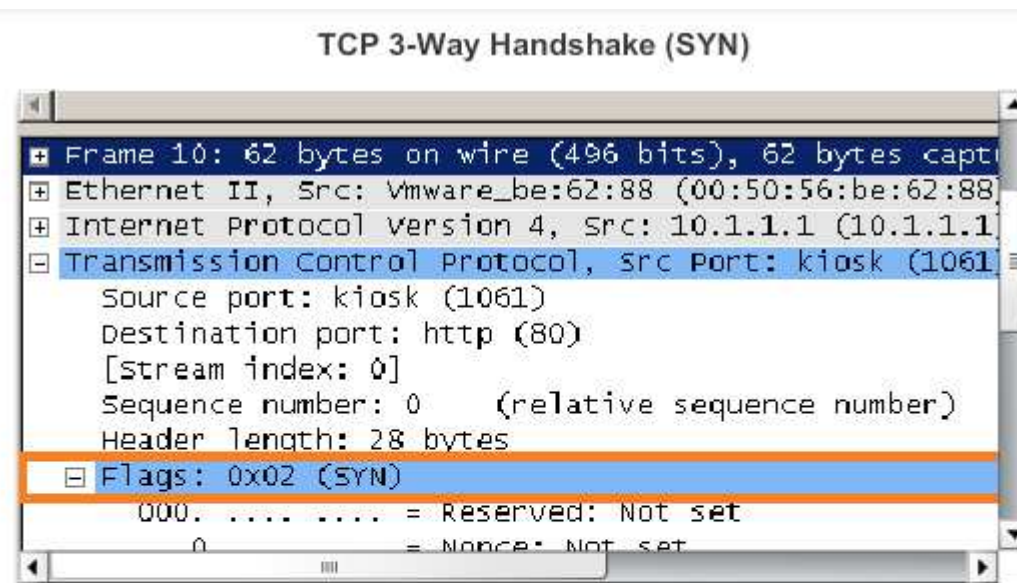
### **Three-Way Handshake**

- ❖ Establishes that the destination device is present on the network
- ❖ Verifies that the destination device has an active service and is accepting requests on the destination port number that the initiating client intends to use for the session
- ❖ Informs the destination device that the source client intends to establish a communication session on that port number

## TCP Communication

# TCP Three-Way Handshake - Step 1

**Step 1:** The initiating client requests a client-to-server communication session with the server



A protocol analyzer shows initial client request for session in frame 10

TCP segment in this frame shows:

- SYN flag set to validate an Initial Sequence Number
- Randomized sequence number valid (relative value is 0)
- Random source port 1061
- Well-known destination port is 80 (HTTP port) indicates web server (httpd)



## TCP Communication

# TCP Three-Way Handshake - Step 2

**Step 2:** The server acknowledges the client-to-server communication session and requests a server-to-client communication session.

TCP 3-Way Handshake (SYN, ACK)

10	16.303490	10.1.1.1	192.168.254.254
11	16.304896	192.168.254.254	10.1.1.1
12	16.304925	10.1.1.1	192.168.254.254
13	16.305153	10.1.1.1	192.168.254.254
14	16.307875	192.168.254.254	10.1.1.1

Frame 11: 62 bytes on wire (496 bits), 62 bytes captured on interface 0
Ethernet II, Src: Cisco_63:74:a0 (00:0f:24:63:74:a0), Dst: 10.1.1.1
Internet Protocol Version 4, Src: 192.168.254.254, Dst: 10.1.1.1
Transmission Control Protocol, Src Port: http (80), Dst Port: 1061
TCP, Seq: 1061, Win: 0, Len: 0
Source port: http (80)

### A protocol analyzer shows server response in frame 11

- ACK flag set to indicate a valid Acknowledgement number
- Acknowledgement number response to initial sequence number as relative value of 1
- SYN flag set to indicate the Initial Sequence Number for the server to client session
- Destination port number of 1061 to corresponding to the clients source port
- Source port number of 80 (HTTP) indicating the web server service (httpd)

## TCP Communication

# TCP Three-Way Handshake - Step 3

**Step 3:** The initiating client acknowledges the server-to-client communication session.

**TCP 3-Way Handshake (ACK)**

No.	Time	Source	Destination
10	16.303490	10.1.1.1	192.168.254.254
11	16.304896	192.168.254.254	10.1.1.1
12	16.304925	10.1.1.1	192.168.254.254
13	16.305153	10.1.1.1	192.168.254.254
14	16.307875	192.168.254.254	10.1.1.1

+	Frame 12: 54 bytes on wire (432 bits), 54 bytes captured
+	Ethernet II, Src: Vmware_k8:62:88 (00:50:56:ba:62:88)
+	Internet Protocol Version 4, Src: 10.1.1.1 (10.1.1.1)
+	Transmission Control Protocol, Src Port: kiosk (1061)

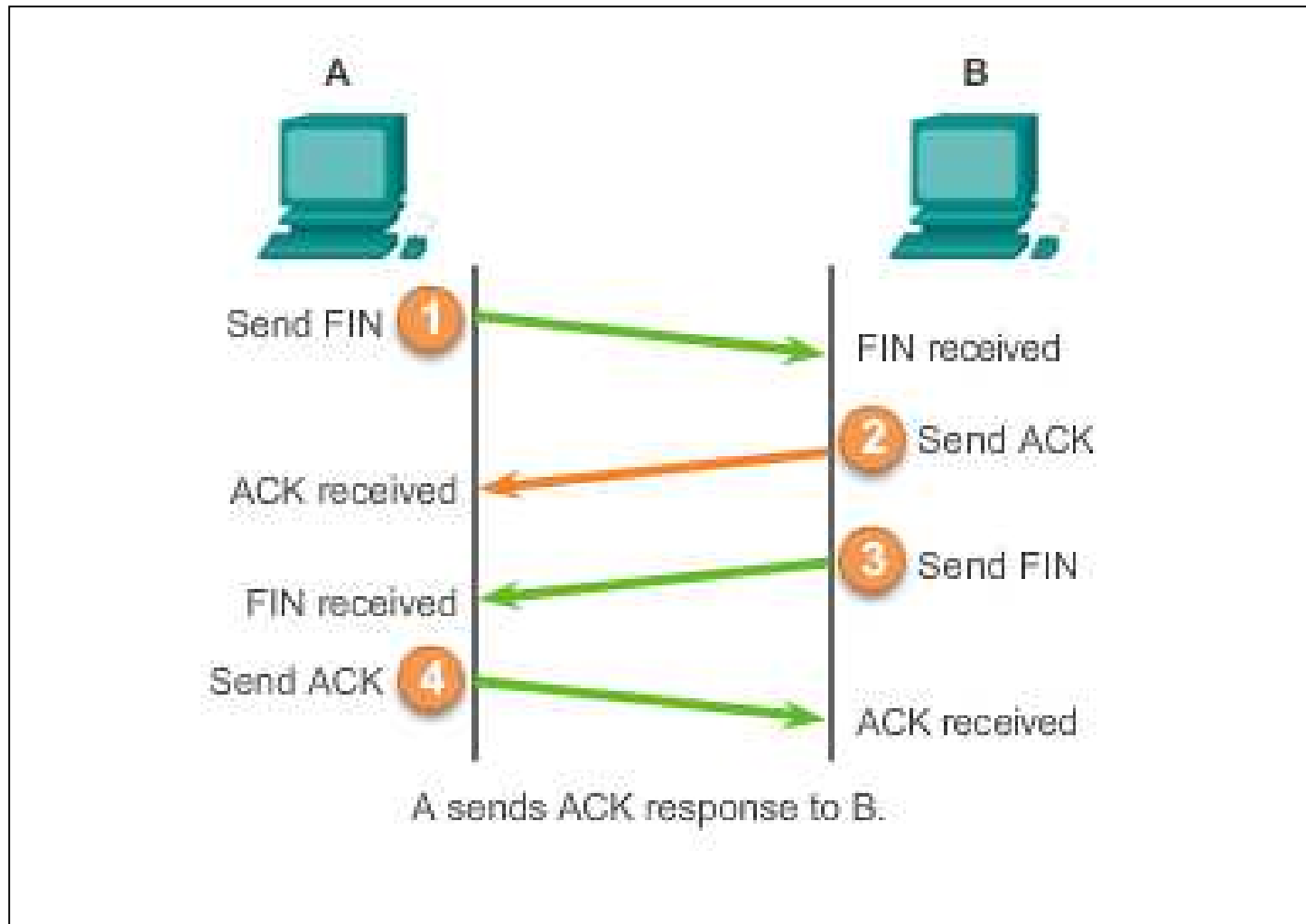
### A protocol analyzer shows client response to session in frame 12

The TCP segment in this frame shows:

- ACK flag set to indicate a valid Acknowledgement number
- Acknowledgement number response to initial sequence number as relative value of 1
- Source port number of 1061 to corresponding
- Destination port number of 80 (HTTP) indicating the web server service (httpd)

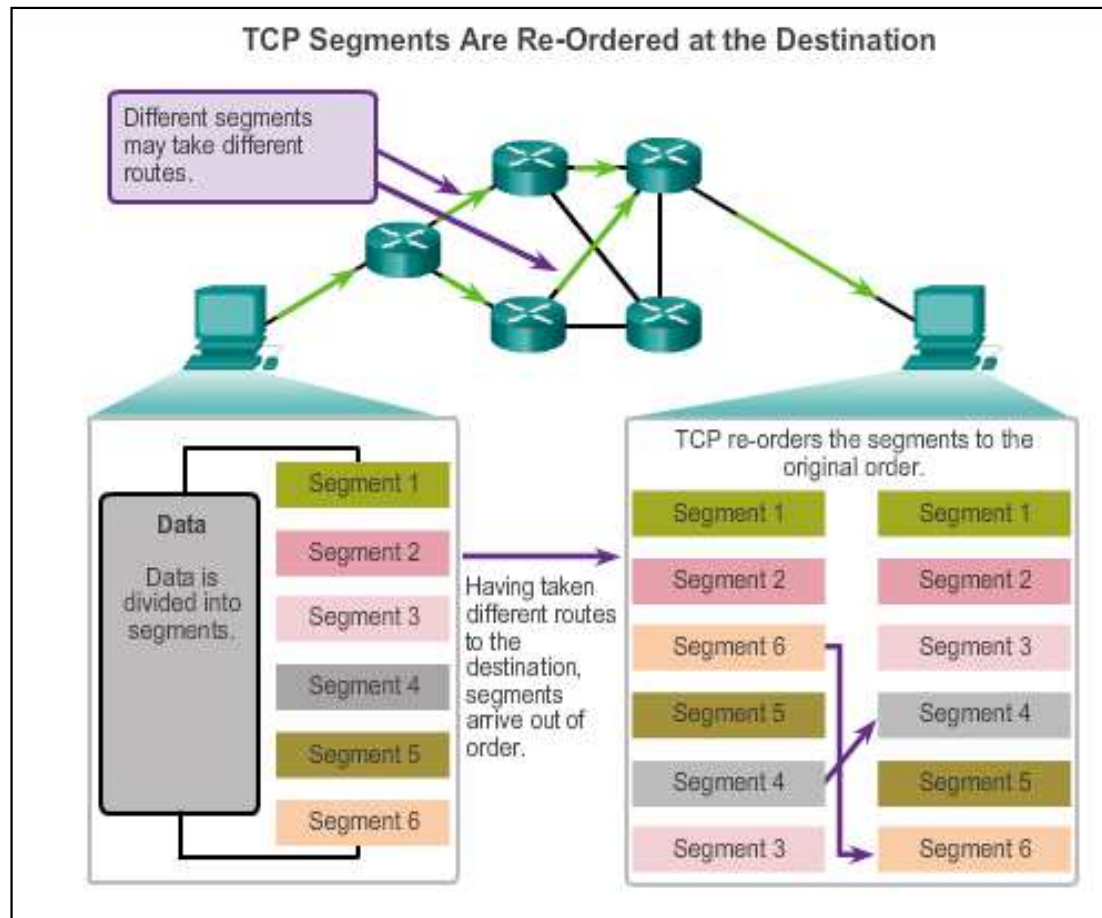
## TCP Communication

# TCP Session Termination



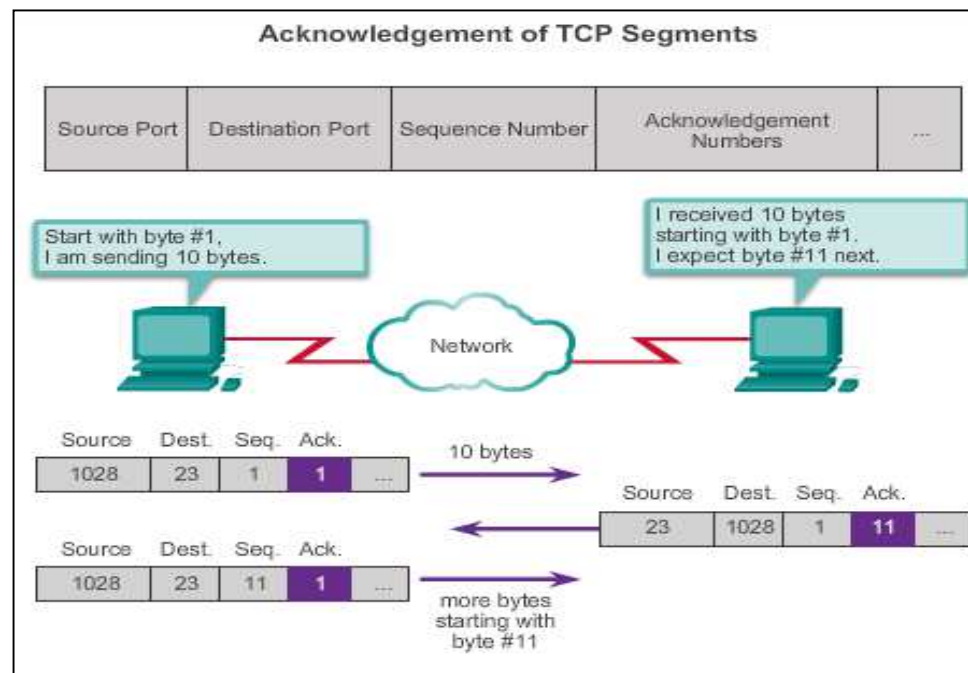
# TCP Reliability - Ordered Delivery

Sequence numbers are used to reassemble segments into their original order.



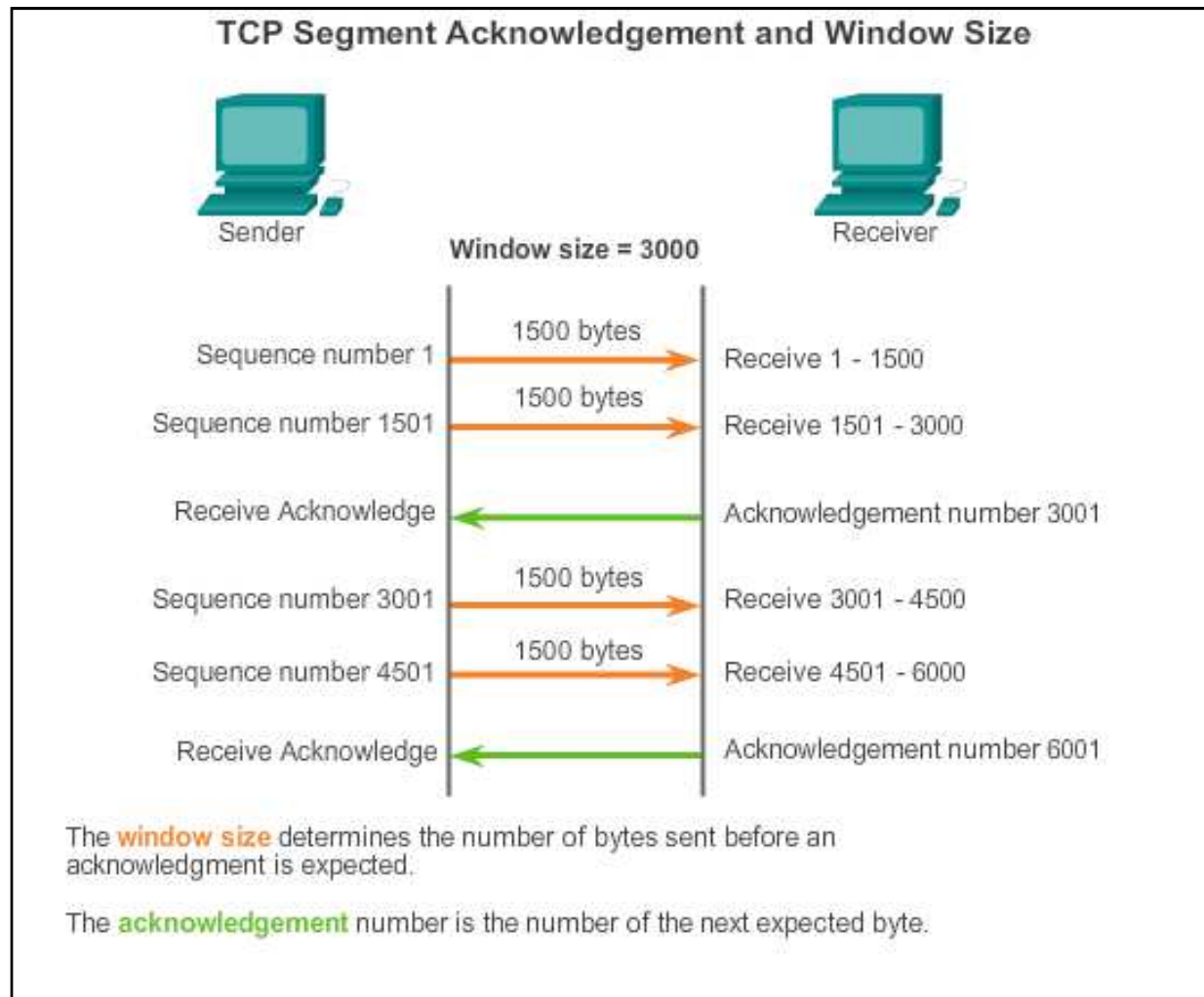
# Acknowledgement and Window Size

The sequence number and acknowledgement number are used together to confirm receipt.

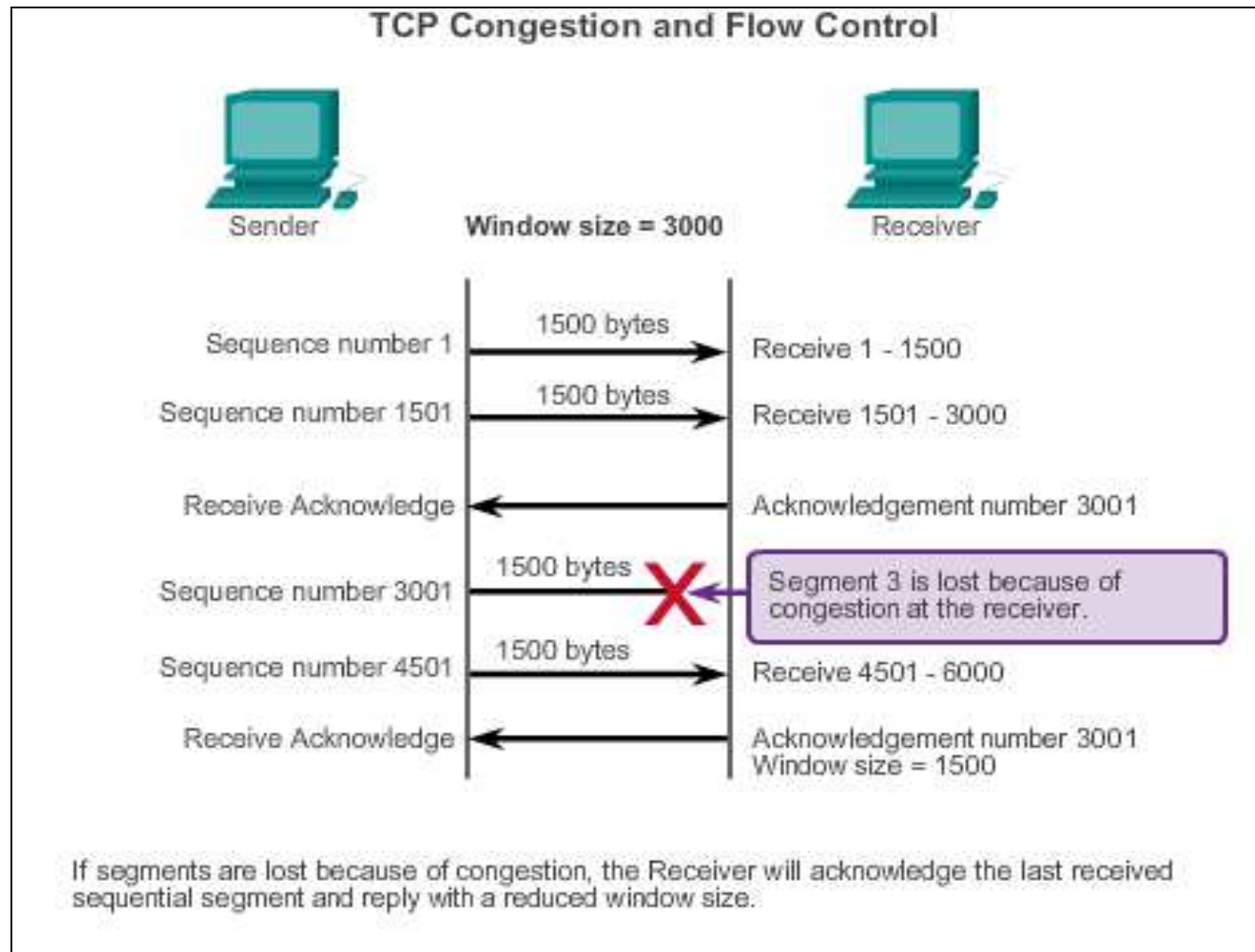


The window size is the amount of data that a source can transmit before an acknowledgement must be received.

# Window Size and Acknowledgements



# TCP Flow Control - Congestion Avoidance





## Reliability and Flow Control

# TCP Reliability - Acknowledgements

Introduction to Networks x  
review.netacad.net/review2.0/courses/IntroNet/en/trunk/course/module7/?id=7.2.2.6

Play Free Games Suggested Sites WhitePages eBay Imported From IE CheckoutReview Options - Under the ... History Electronics, Cars, Fa... BestBuy.com - Cart Other bookmarks

**Cisco Networking Academy**

Chapter 7: Transport Layer

Delivery

- 7.2.2.2 TCP Reliability - Acknowledgement and Window Size
- 7.2.2.3 TCP Reliability - Data Loss and Retransmission
- 7.2.2.4 TCP Flow Control - Window Size and Acknowledgements
- 7.2.2.5 TCP Flow Control - Congestion Avoidance
- 7.2.2.6 TCP Reliability - Acknowledgements**
- 7.2.3 UDP Communication
- 7.2.4 TCP or UDP, that is the Question
- 7.3 Summary

**TCP and UDP**

**Reliability and Flow Control**

one for each successive segment it received.

**Step 4.** The server receives the multiple Acknowledgement 2s from the client. The server must resend Segment 2. This is where the inefficiency happens. The server has no indication whether Segments 3 through 5 arrived at the client. The server must not only resend the lost segment, Segment 2, but also Segments 3 to 5.

**Step 5.** The client receives the missing segment, Segment 2 along with duplicate segments, Segments 3 to 5. The client sends an acknowledgement for each of the segments.

**Step 6.** The server receives the acknowledgments from the client and sends the next segment in its transmission queue, Segment 6.

Client Server

1 Segment 1  
2 Segment 2  
3 Segment 3  
4 Segment 4  
5 Segment 5  
6 Segment 6

ACK=2  
ACK=2 (Duplicate)  
ACK=2 (Duplicate)  
ACK=2 (Duplicate)

ACK=3  
ACK=4  
ACK=5  
ACK=6

Previous Next

5:33 PM  
3/29/2013



# UDP Low Overhead vs. Reliability

## UDP

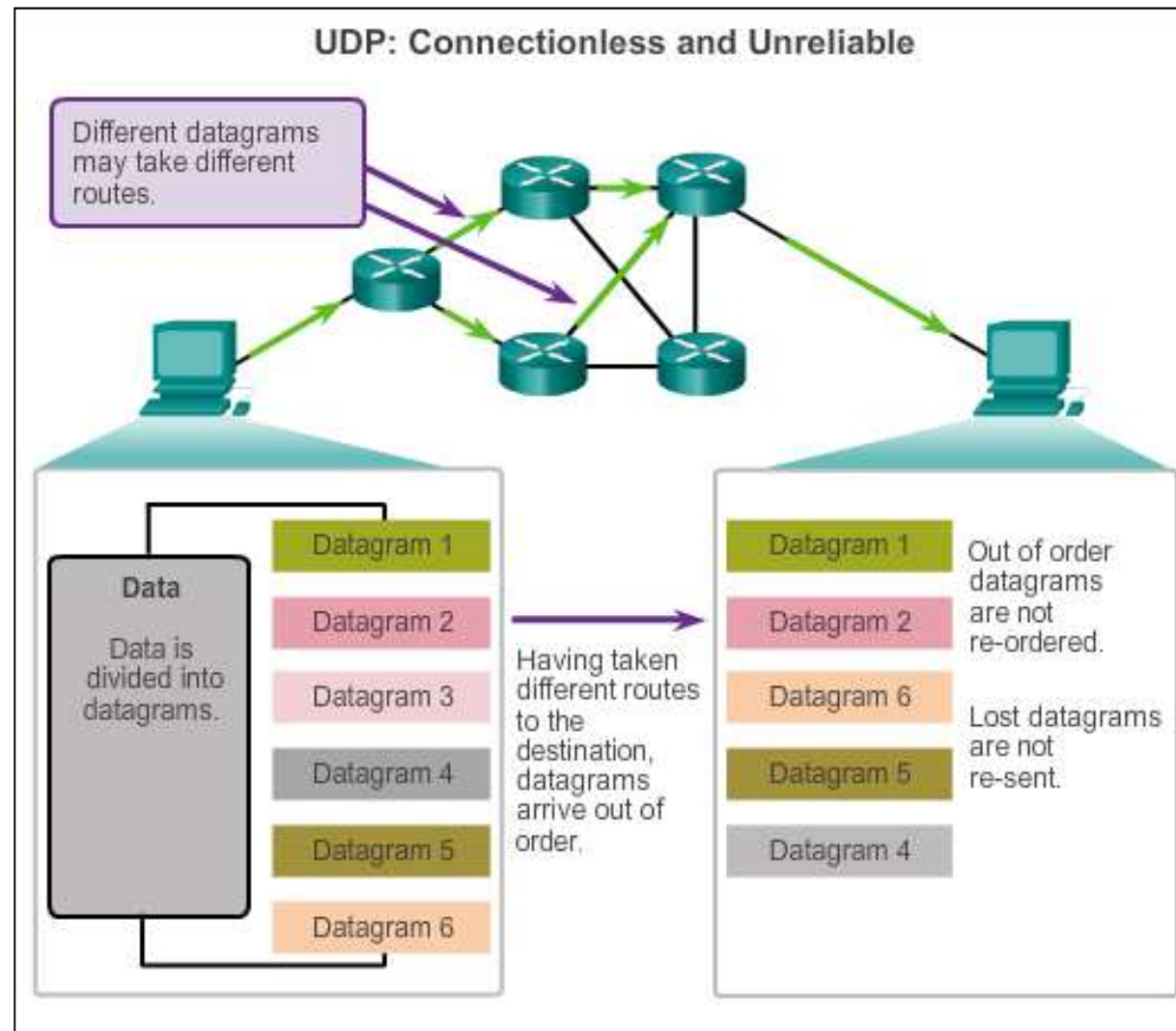
- ❖ Simple protocol that provides the basic transport layer function
- ❖ Used by applications that can tolerate small loss of data
- ❖ Used by applications that cannot tolerate delay

## Used by

- ❖ DNS
- ❖ Simple Network Management Protocol (SNMP)
- ❖ Dynamic Host Configuration Protocol (DHCP)
- ❖ Trivial File Transfer Protocol (TFTP)
- ❖ IP telephony or VoIP
- ❖ Online games

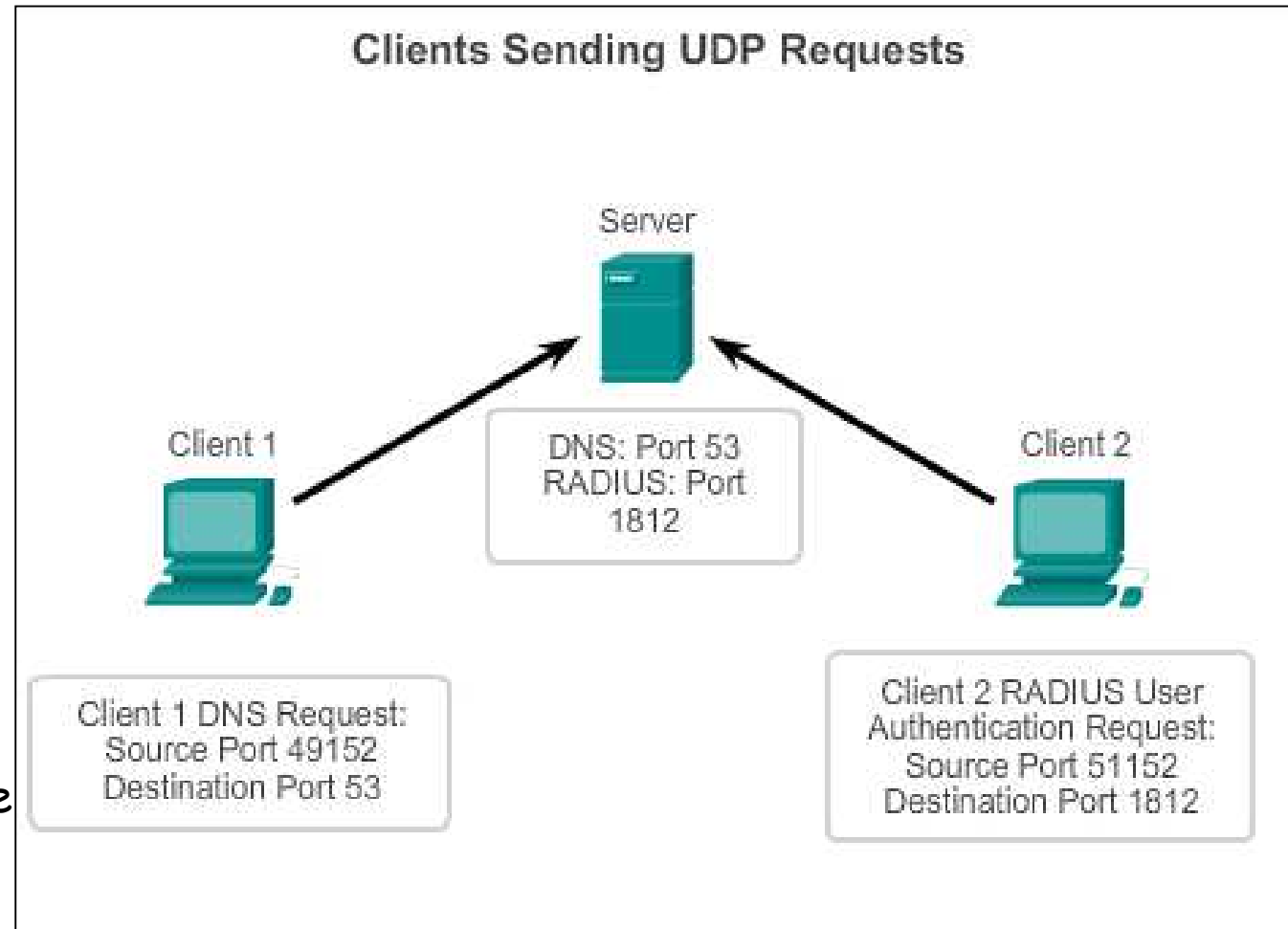
## UDP Communication

# Datagram Reassembly



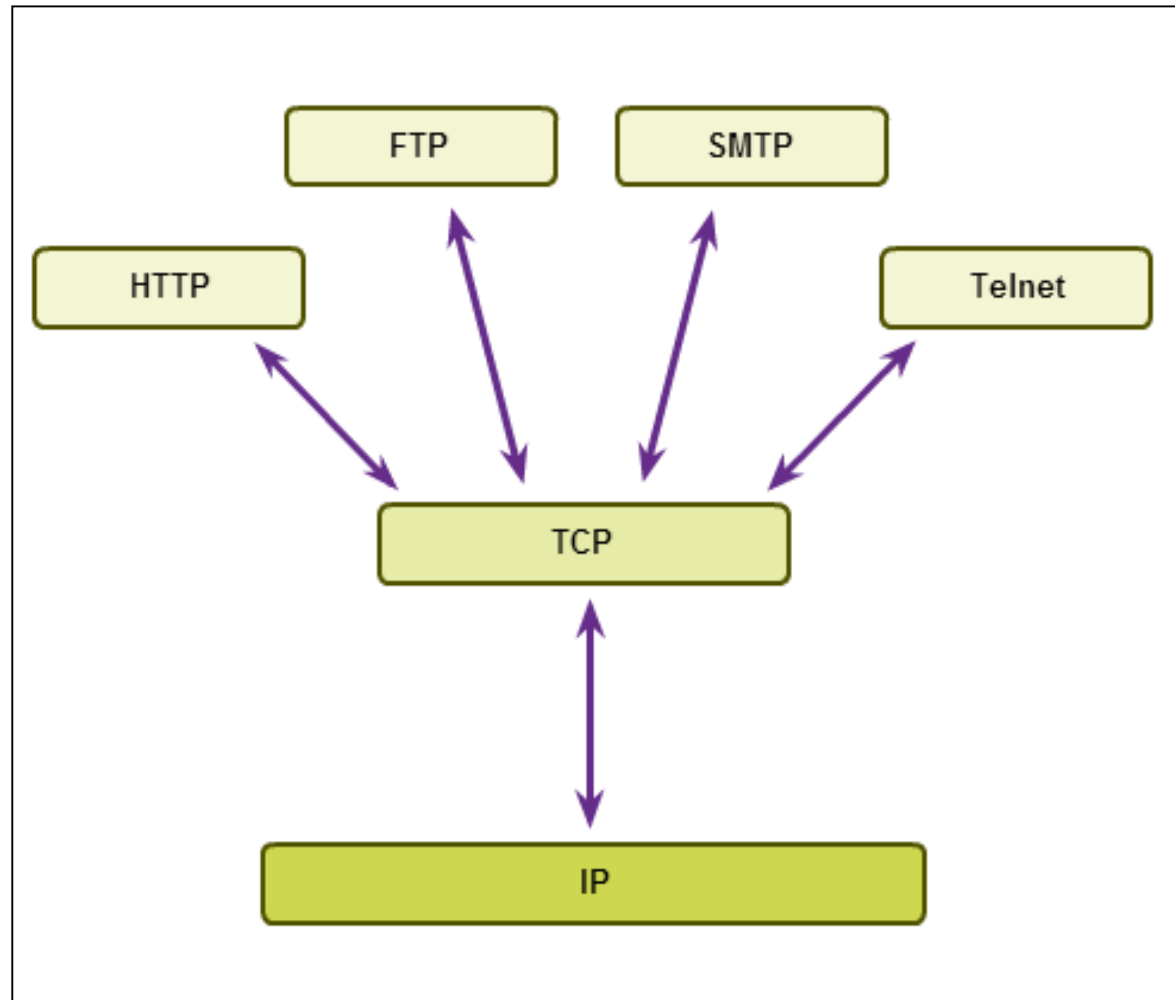
# UDP Server and Client Processes

- ❖ UDP-based server applications are assigned well-known or registered port numbers.
- ❖ UDP client process randomly selects port number from range of dynamic port numbers as the source port.



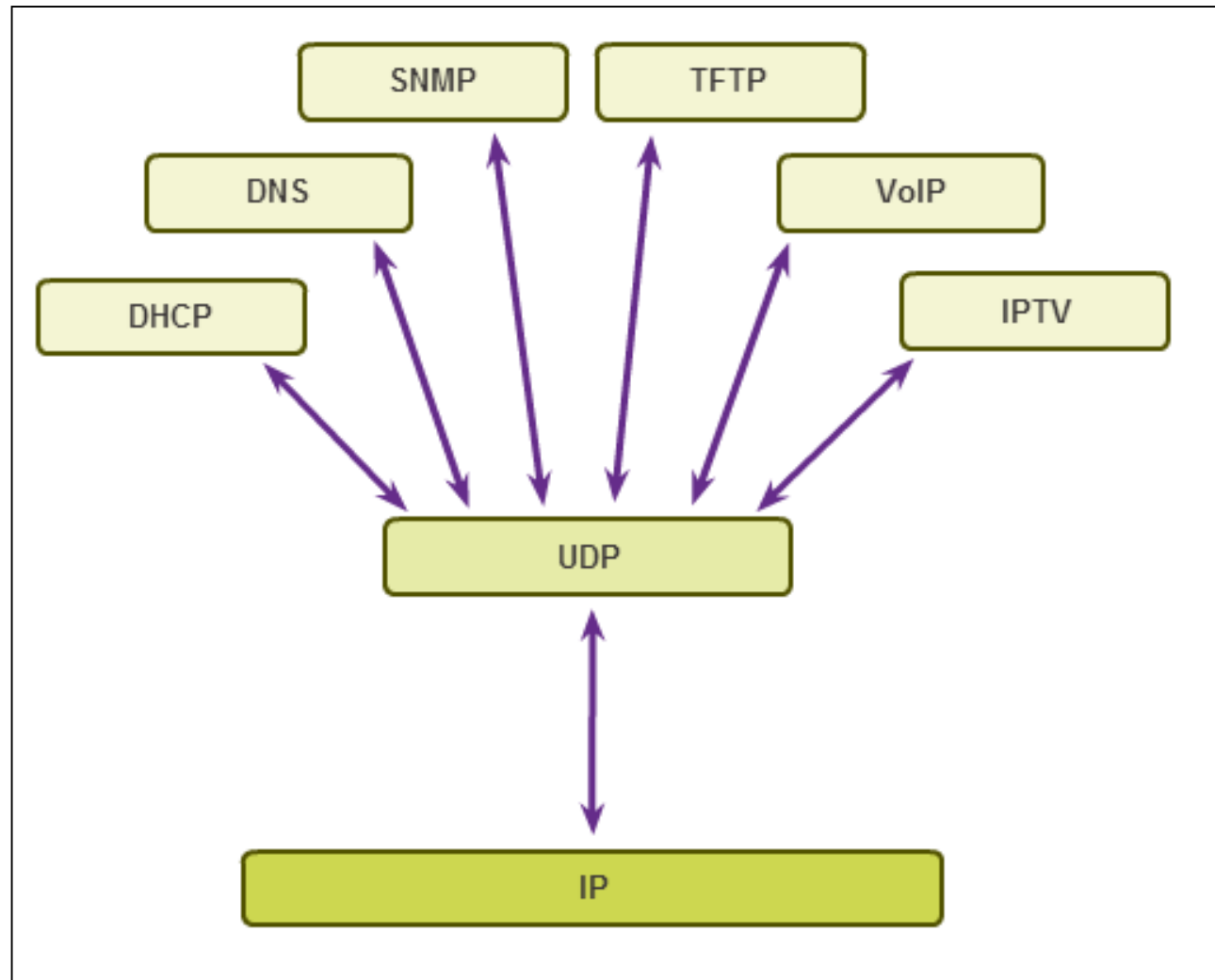
TCP or UDP

# Applications that use TCP



TCP or UDP

# Applications That Use UDP



## Summary

# Summary

In this Lecture, you learned:

- ❖ The role of the transport layer is to provide three main services: multiplexing, segmentation and reassembly, and error checking. It does this by:
  - Dividing data received from an application into segments.
  - Adding a header to identify and manage each segment.
  - Using the header information to reassemble the segments back into application data.
  - Passing the assembled data to the correct application.
- ❖ How TCP and UDP operate and which popular applications use each protocol.
- ❖ Transport Layer functions are necessary to address issues in QoS and security in networks.
- ❖ Ports provide a “tunnel” for data to get from the transport layer to the appropriate application at the destination.