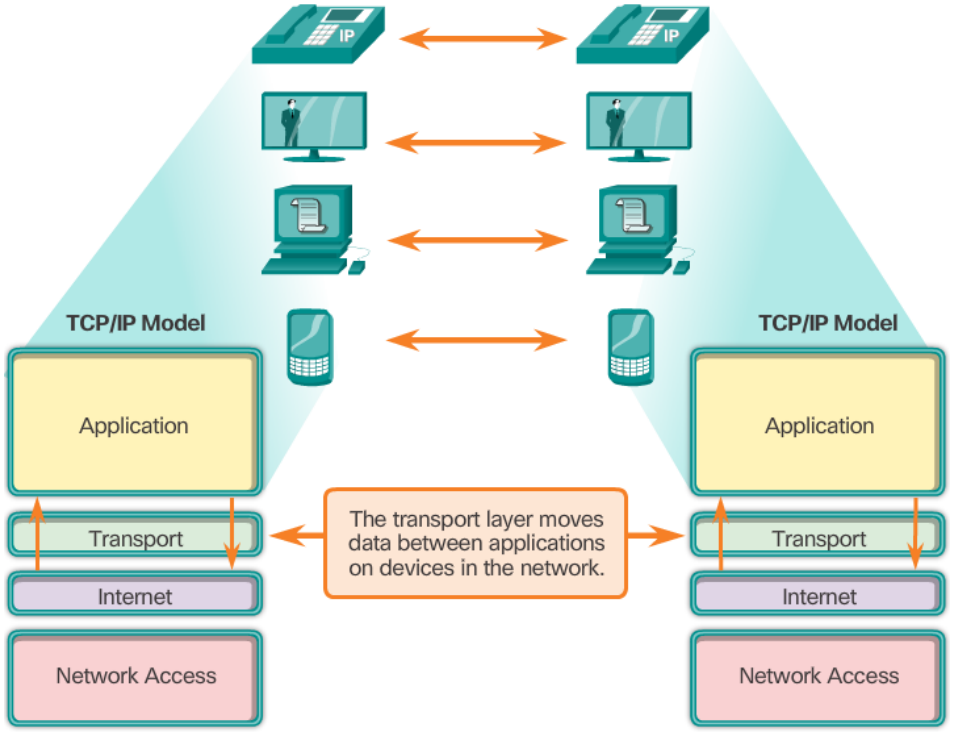
Transport Layer

Transportation of Data

Role of the Transport Layer

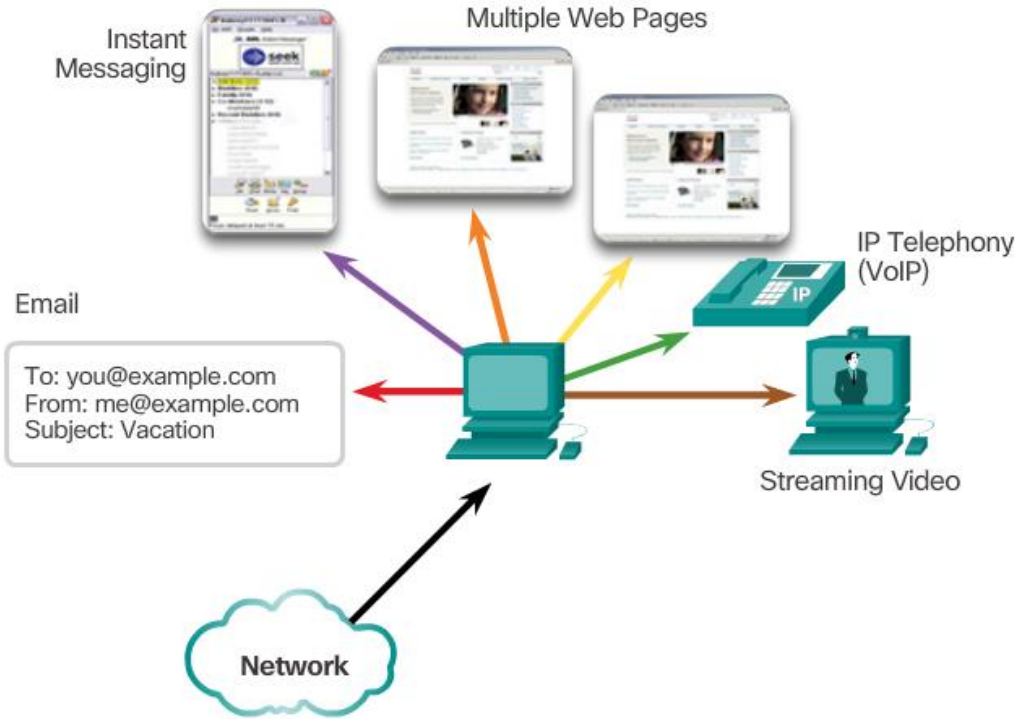
- The transport layer is responsible for establishing a temporary communication session between two applications and delivering data between them.
- The transport layer provides services, such as:
  - Connection-oriented data stream support
  - Reliability
  - Flow control
  - Multiplexing



Transport Layer Responsibilities

Track Individual Conversations

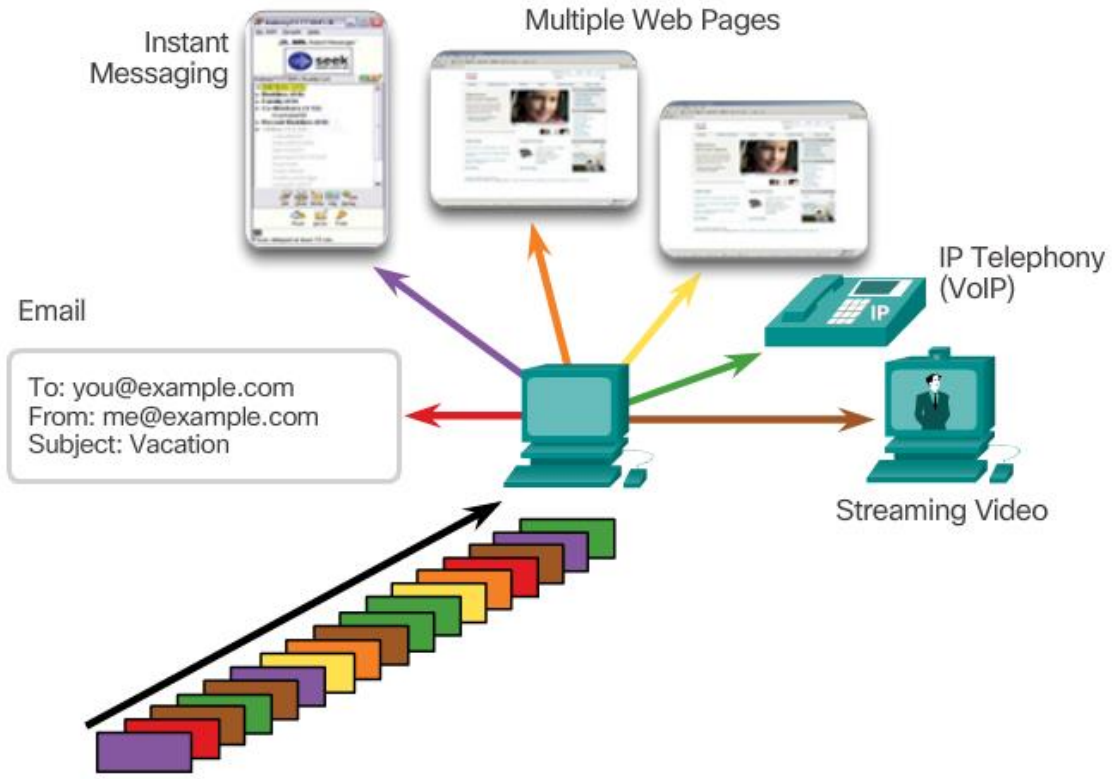
By tracking each individual conversation flowing between a source application and a destination application separately.



Transport Layer Responsibilities (cont.)

Segment Data and Reassemble Segments

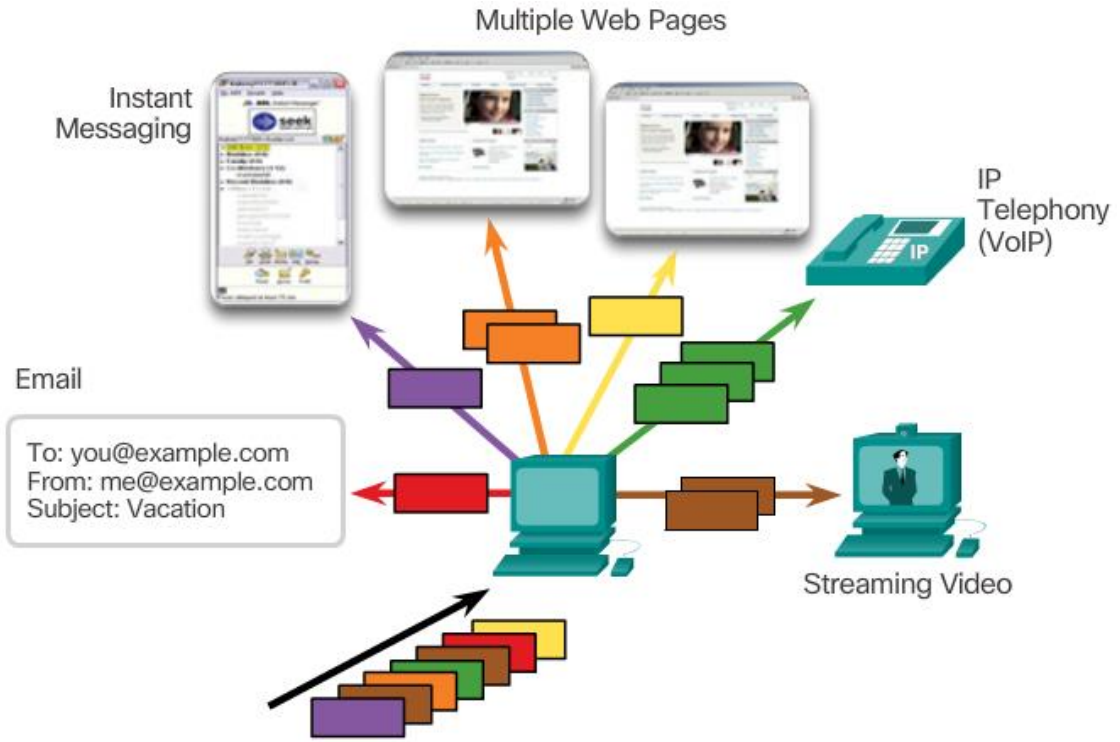
By dividing the data into segments that are easier to manage and transport.



Transport Layer Responsibilities (cont.)

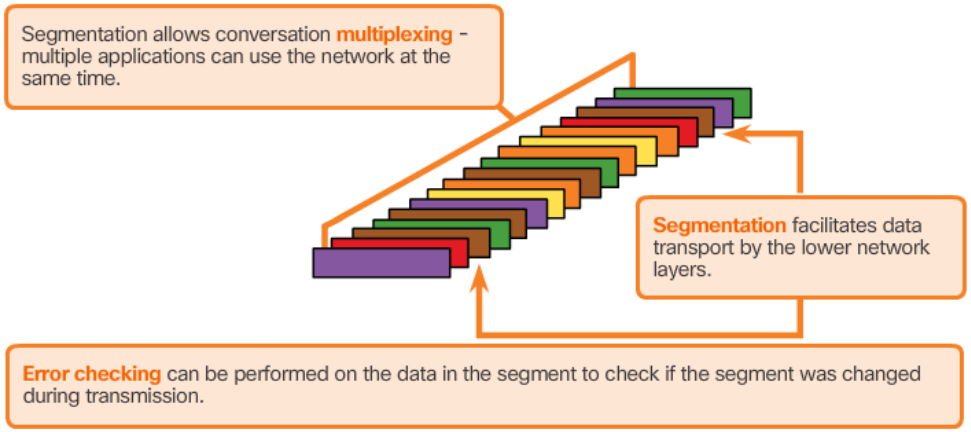
Identify the Applications

By ensuring even when multiple applications are running on a device, all applications receive the correct data.



Conversation Multiplexing

- Segmenting the data into smaller chunks enables many different communications, from many different users, to be interleaved (multiplexed) on the same network.
- The transport layer adds a header that contains binary data to identify each segment of data and to enable various transport layer protocols to perform different functions in the management of data communication.



TCP and UDP Overview

TCP Features

- In addition to supporting the basic functions of data segmentation and reassembly, TCP provides the following services:
  - Establishing a Session
  - Reliable Delivery
  - Same-Order Delivery
  - Flow Control

Establishing a session ensures the application is ready to receive the data.

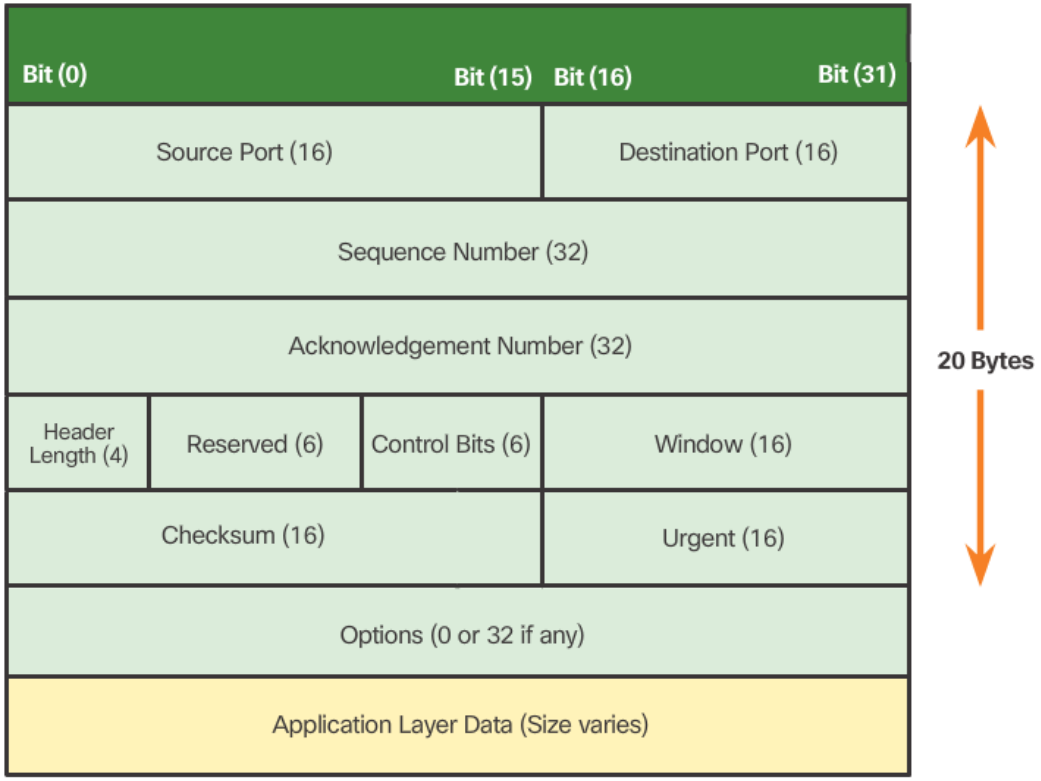
Same order delivery ensures that the segments are reassembled into the proper order.

Reliable delivery means lost segments are resent so the data is received complete.

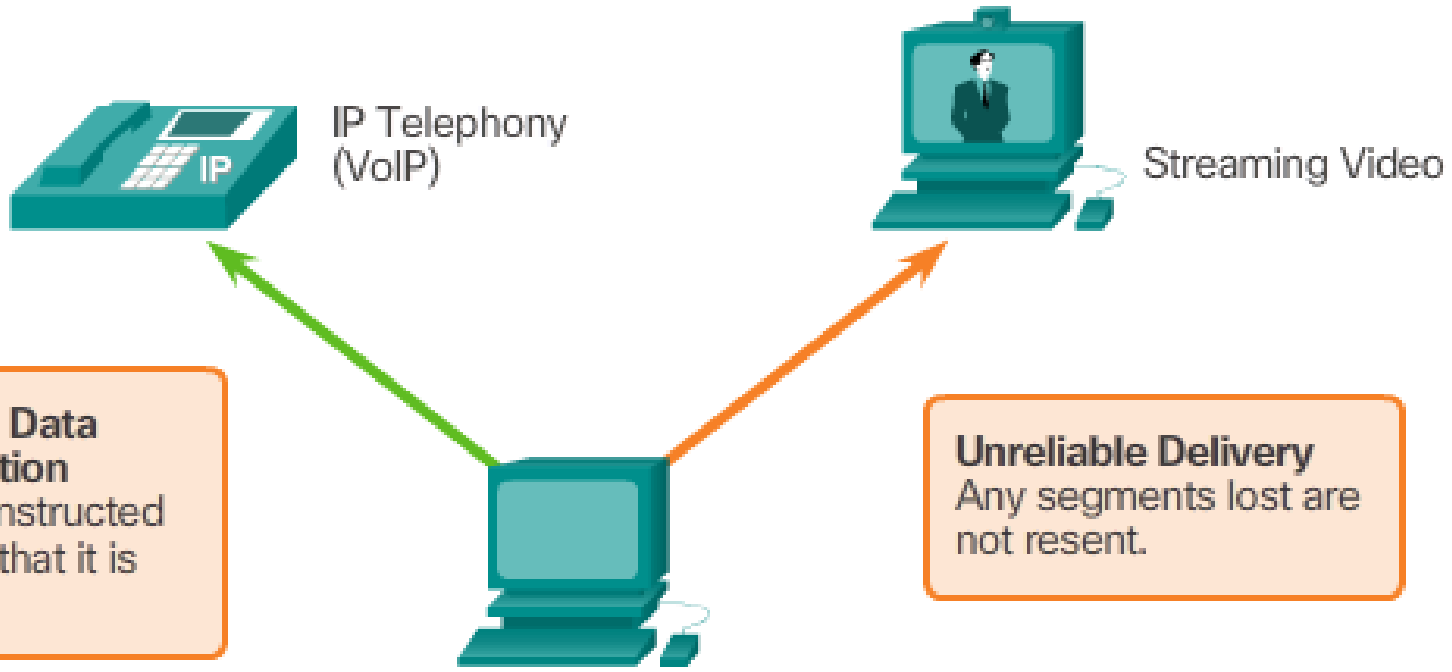
Flow control ensures that the receiver is able to process the data received.

TCP Header

- TCP is a stateful protocol. It keeps track of the state of the communication session by recording which information it has sent and which information has been acknowledged.
- Each TCP segment has 20 bytes of overhead in the header encapsulating the application layer data, as shown in this image.



UDP Features



No Ordered Data Reconstruction
Data is reconstructed in the order that it is received.

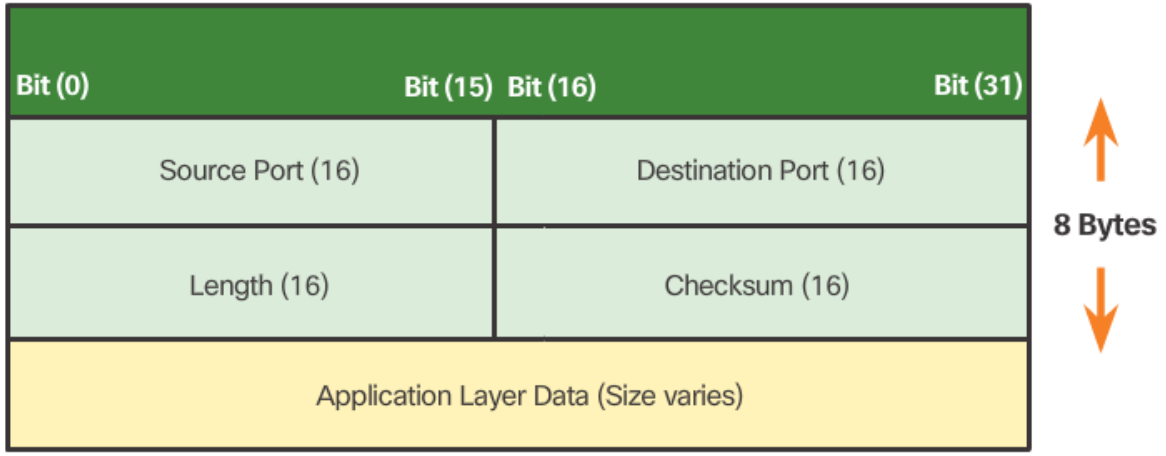
Unreliable Delivery
Any segments lost are not resent.

Connectionless
No session establishment.

No Flow Control
Does not inform the sender about resource availability.

UDP Header

- UDP is a stateless protocol. Neither the sender or the receiver is obligated to keep track of the state of the communication session.
- Reliability must be handled by the application.
- Live video and voice applications must quickly deliver data and can tolerate some data loss; they are perfectly suited to UDP.
- The pieces of communication in UDP are called datagrams.
- These datagrams are sent as best-effort by the transport layer protocol.
- UDP has a low overhead of 8 bytes.



Port Numbers

- Source Port
  - The source port number is dynamically chosen by the sending device to identify a conversation between two devices.
  - An HTTP client usually sends multiple HTTP requests to a web server at the same time. Each separate HTTP conversation is tracked based on the source ports.
- Destination Port
  - Used to identify an application or service running in the server.
  - A server can offer more than one service at the same time, offering a web service on port 80 and FTP on port 21 simultaneously.