# Computer Networks

Transport Layer

The Internet Transport Protocols: TCP

- TCP (Transmission Control Protocol) was specifically designed to provide a reliable end-to-end byte stream over an unreliable internetwork.

- An internetwork differs from a single network because different parts may have wildly different topologies, bandwidths, delays, packet sizes, and other parameters.

- TCP was designed to dynamically adapt to properties of the internetwork and to be robust in the face of many kinds of failures.

TCP Segments

- The sending and receiving TCP entities, exchange data in the form of segments.

- A TCP segment consists of a fixed 20-byte header (plus an optional part) followed by zero or more data bytes.

- The TCP software decides how big segments should be. It can accumulate data from several writes into one segment or can split data from one write over multiple segments.

- Two limits restrict the segment size.
  - First, each segment, including the TCP header, must fit in the 65,515-byte IP payload.
  - Second, each network has a maximum transfer unit, or MTU, and each segment must fit in the MTU.
  - In practice, the MTU is generally 1500 bytes (the Ethernet payload size) and thus defines the upper bound on segment size.
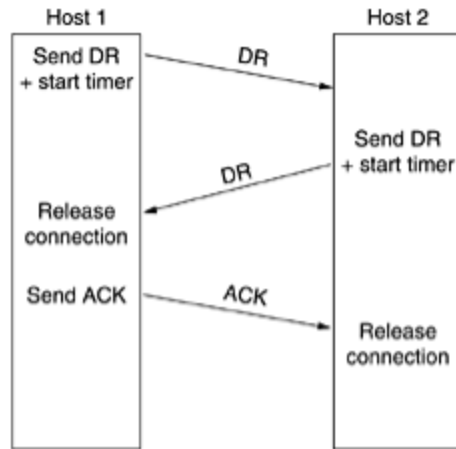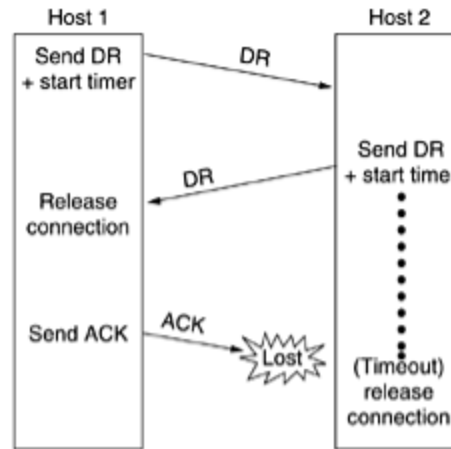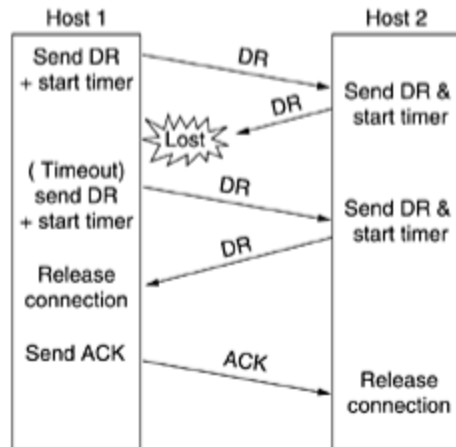
TCP Segment Header
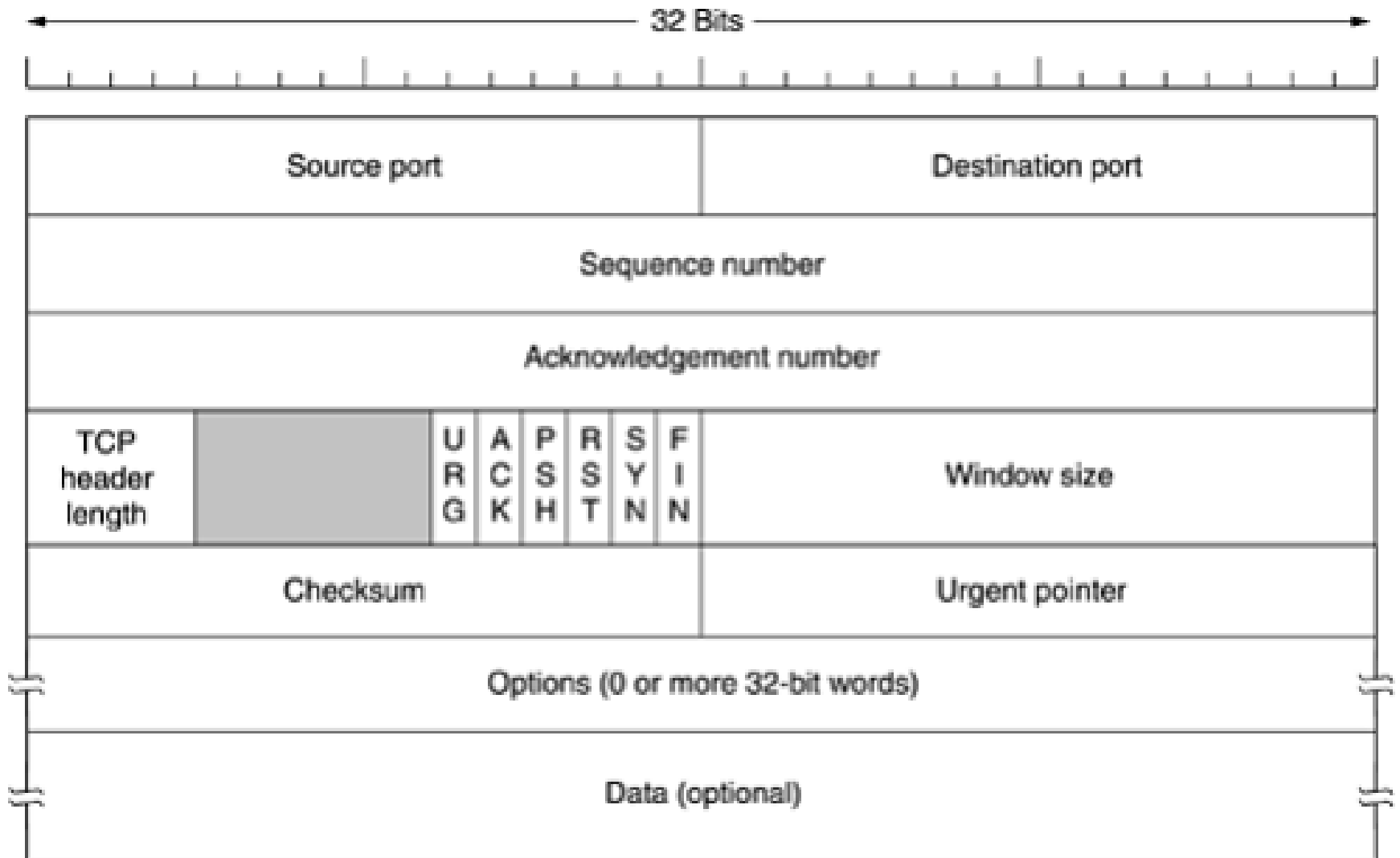
Three-Way Handshaking to Connect

(a)

(b)

(c)
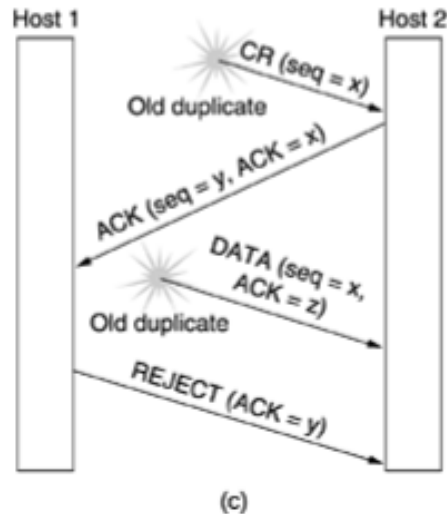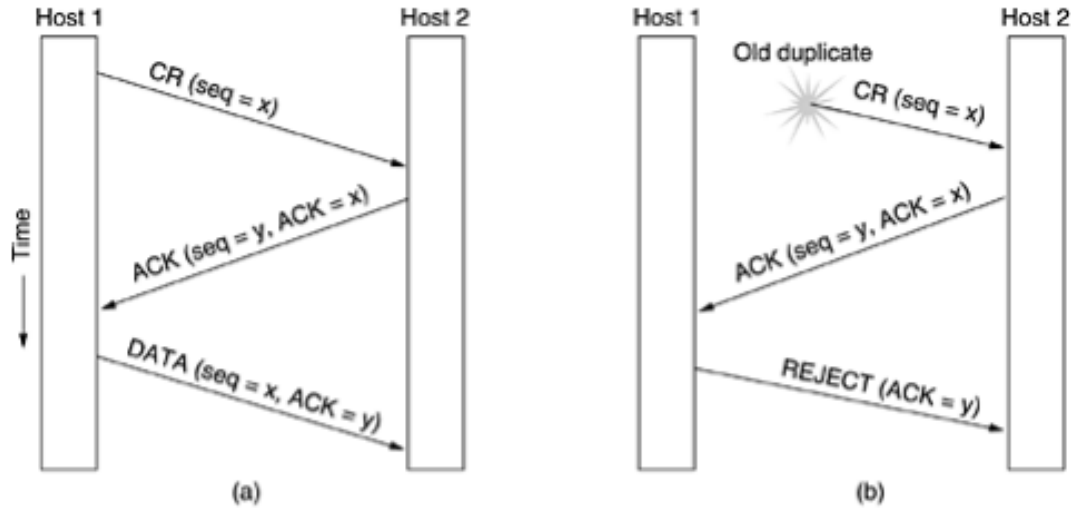
Three-Way Handshaking to Disconnect
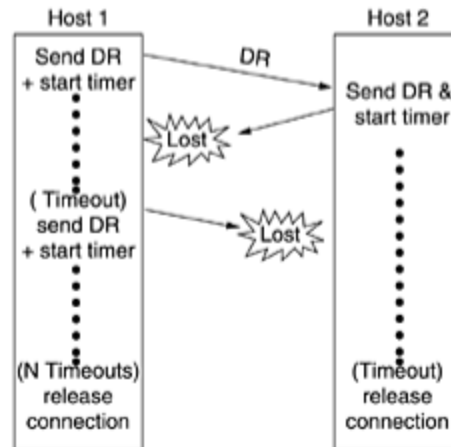
Checksum Computation

- Checksum is computed by adding 16-bit fields of header and pseudo-header.
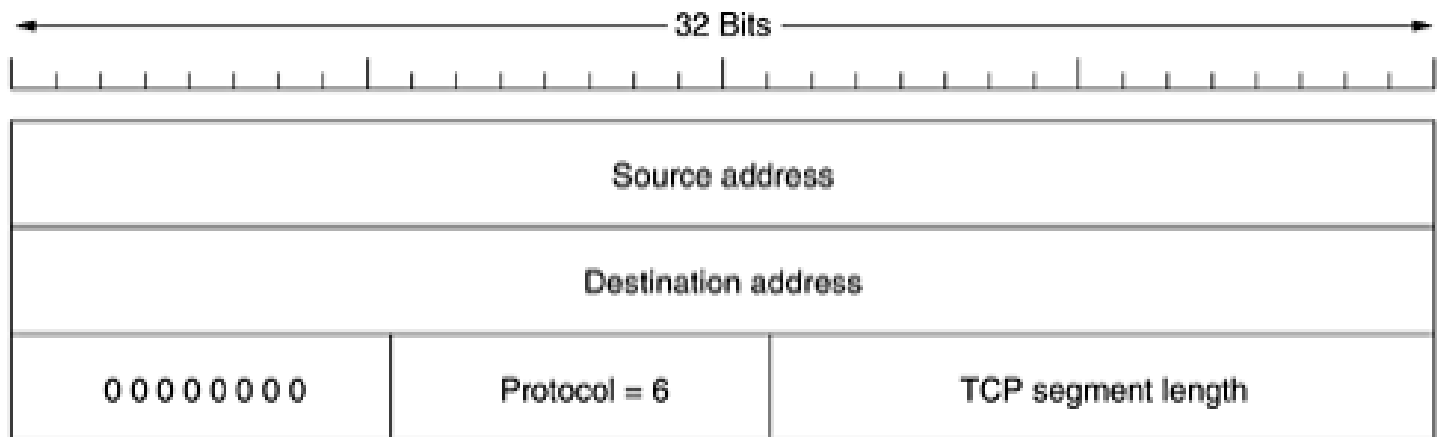
Flags

- URG is set to 1 if the Urgent pointer is in use.
- The ACK bit is set to 1 to indicate that the Acknowledgement number is valid.
- The PSH bit indicates PUSHed data.
- The RST bit is used to reset a connection that has become confused due to a host crash or some other reason.
- The SYN bit is used to establish connections.
- The FIN bit is used to release a connection.

Flow Control

- Flow control in TCP is handled using a variable-sized sliding window.
- The Window size field tells how many bytes may be sent starting at the byte acknowledged.
- A Window size field of 0 is legal and says that the bytes up to and including Acknowledgement number - 1 have been received, but that the receiver can accept no more data for the moment.
- The receiver can later grant permission to send by transmitting a segment with the same Acknowledgement number and a nonzero Window size field.

TCP Options

- The Options field provides a way to add extra facilities not covered by the regular header such as:
  - the maximum TCP payload (default 536 bytes)
  - Window scale
  - Selective repeat instead of go-back-n protocol

Congestion Control

- When the load offered to any network is more than it can handle, congestion builds up.

- To solve the congestion problem, the idea is to refrain from injecting a new packet into the network until an old one is delivered.

- TCP attempts to achieve this goal by dynamically manipulating the window size.

Congestion Control

- Each sender maintains two windows:
  - the window the receiver has granted
  - the congestion window
- Each window reflects the number of bytes the sender may transmit.
- The number of bytes that may be sent is the minimum of the two windows.

Congestion Control

- When a connection is established, the sender initializes the congestion window to the size of the maximum segment in use on the connection.

- It then sends one maximum segment. If this segment is acknowledged before the timer goes off, it adds one segment's worth of bytes to the congestion window to make it two maximum size segments and sends two segments.

- As each of these segments is acknowledged, the congestion window is increased by one maximum segment size.

- When the congestion window is **n** segments, if all n are acknowledged on time, the congestion window is increased by the byte count corresponding to n segments.

- In effect, each burst acknowledged doubles the congestion window.

Congestion Control

- The congestion window keeps growing exponentially until either a timeout occurs or the receiver's window is reached.
- This algorithm is called slow start, but it is not slow at all. It is exponential.
- All TCP implementations are required to support it.
- It uses a third parameter, the threshold, initially 64 KB, in addition to the receiver and congestion windows.
- When a timeout occurs, the threshold is set to half of the current congestion window, and the congestion window is reset to one maximum segment.
- Slow start is then used to determine what the network can handle, except that exponential growth stops when the threshold is hit.
- From that point on, successful transmissions grow the congestion window linearly (by one maximum segment for each burst) instead of one per segment.
- In effect, this algorithm is guessing that it is probably acceptable to cut the congestion window in half, and then it gradually works its way up from there.

Optimizing TCP

- Sending small segments (1 or 2 bytes) wastes the bandwidth because each segment contains at least 40 bytes of overhead (TCP and IP headers)

Nagle's Algorithm

- When data come into the sender one byte at a time, just send the first byte and buffer all the rest until the outstanding byte is acknowledged.

- Then send all the buffered characters in one TCP segment and start buffering again until they are all acknowledged.

- The algorithm additionally allows a new packet to be sent if
  - enough data are in the sender's buffer to fill half the window
 - there exist as much data in buffer as a maximum segment.

Clark's Algorithm

- Clark's solution is to prevent the receiver from sending a window update for 1 byte.

- Instead the receiver is forced to wait until it has as much buffer space available as:

  - the maximum segment size it advertised when the connection was established

  - buffer is half empty.

  (whichever is smaller)

User Datagram Protocol (UDP)

- UDP is a connectionless transport protocol which provides a way for applications to send encapsulated IP datagrams and send them without having to establish a connection.

- UDP transmits segments consisting of an 8-byte header followed by the payload.

UDP Header
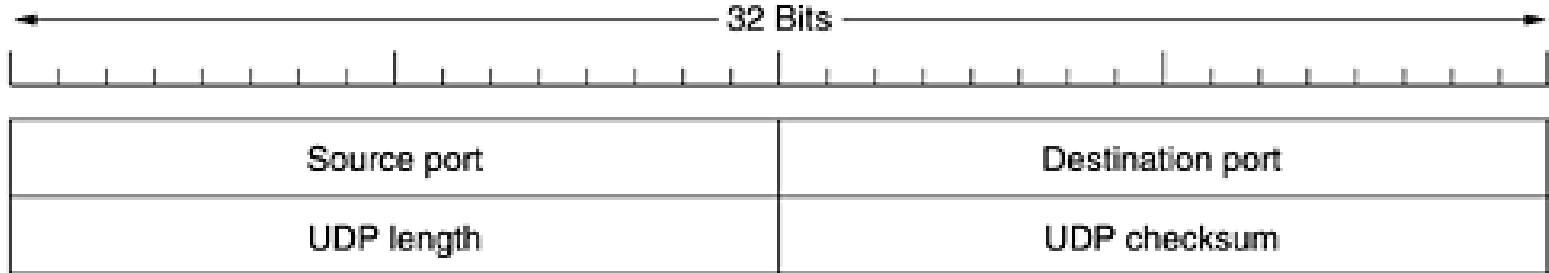
- UDP length includes 8 bytes header plus payload length.