TCP - Transmission Control Protocol
Overview by Michael Johnson

iReady Confidential

What is TCP?

A reliable delivery service that runs on top of IP (Internet Protocol). It has the following properties:

- Full Duplex, Point-to-Point Connection (no broadcasts or multicasts)
- Byte stream interface : sequence of octets.
- Reliable transfer: Data is delivered in order and acknowledged.
- Flow control
- Reliable startup: old connections are not confused with new.
- Graceful shutdown: data sent before closing a connection is not lost.

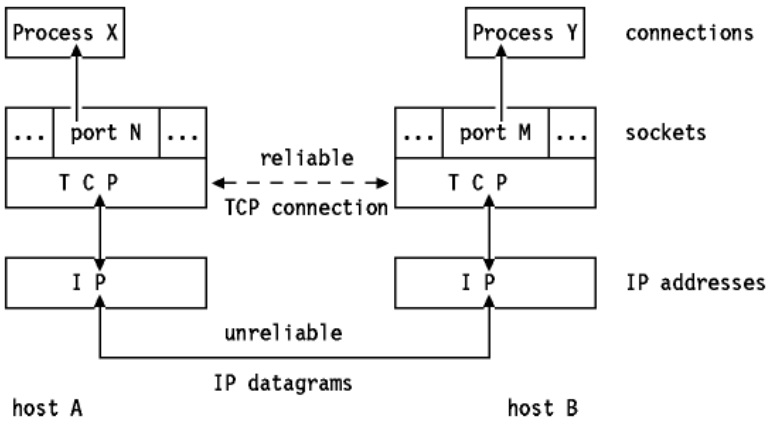
TCP Specifications

- How two hosts initiate a TCP connection and how they agree when it is complete.
- Provide the format of the data and the acknowledgements.
- How to recover lost, duplicated or out of order packets.
- Defined in a subset of a series of documents called RFC's, most notably RFC793.

Ports, Connections and Endpoints

- TCP uses connection and not the protocol port as its fundamental abstraction.
- An endpoint is identified by an IP address and a TCP port number.
- Connections are identified by a pair of endpoints.
- A TCP port can support multiple connections simultaneously.

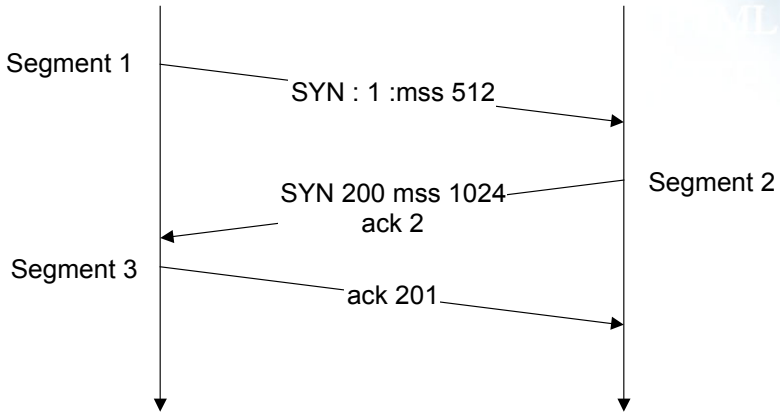
TCP Connection



Establishing and Terminating TCP Connections

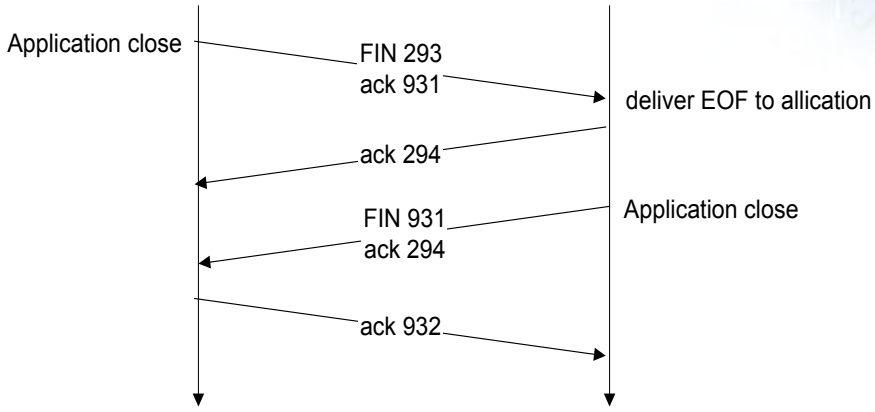
- Three-Way Handshake
  - Options/MTU
- Initial Sequence Number (ISN)
- FIN-ACK-FIN-ACK
- Half Close
- TIME-WAIT
- TCP Reset (RST)

Connection Establishment

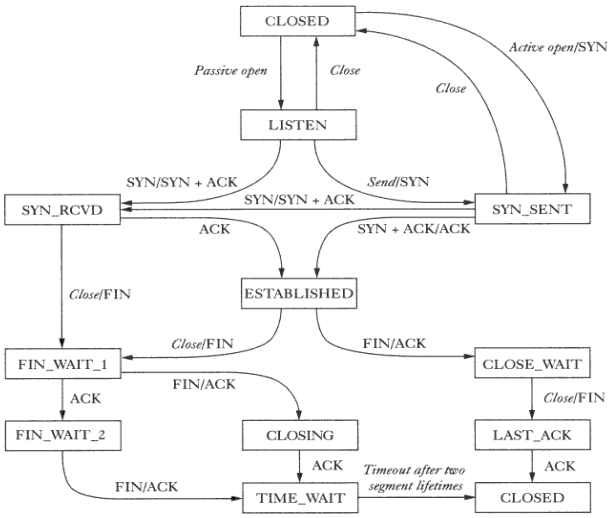


"3 Way Handshake"

Normal Connection Close



TCP State Diagram



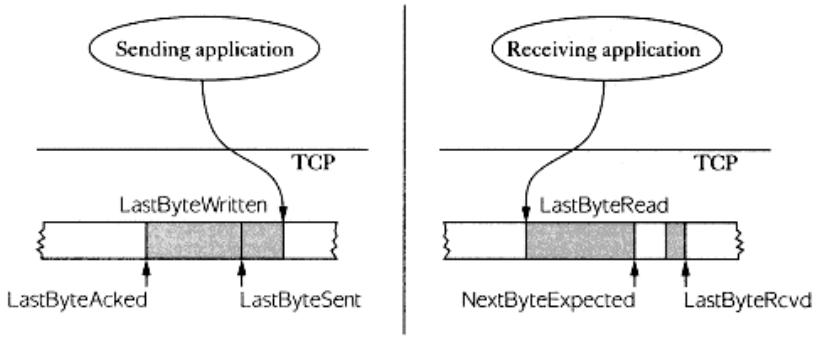
Moving Data

- TCP divides data streams into packages called segments.
- Each TCP segment is carried over a single IP packet.
- Segment reception is acknowledged.
- A sliding window mechanism is used for efficiency and flow control.
- Each end of a connection advertises it's window size.
- Flow control is achieved by restricting transmissions until buffer space is available.

Sequencing the stream

- Octets of the data stream are numbered sequentially based on the ISN.
- Three stream pointers into the stream are maintained:
  - Beginning of sliding window.
  - End of sliding window.
  - Boundary inside the sliding window that separates the octets that have been sent from those octets that have not been sent.
- The end of the sliding window expands when the host adds more data to send on the connection.
- The beginning of the sliding window contracts when data sent to the peer has been acknowledged.
- The internal boundary moves toward the end of the window as data is sent is transmitted.

Sliding Window



Flow Control

- Window size can vary over time to facilitate flow control.
- Each acknowledgement contains a window advertisement that specifies how much data the receiver is prepared to accept.
- In the case of an increased window advertisement, the sender increases the amount of data it will send unacknowledged.
- Advertised windows should not shrink, but can close.
- When an advertisement contains a zero window size the sender stops transmitting.
- The receiver later advertises a non-zero window size to trigger the flow of data again.

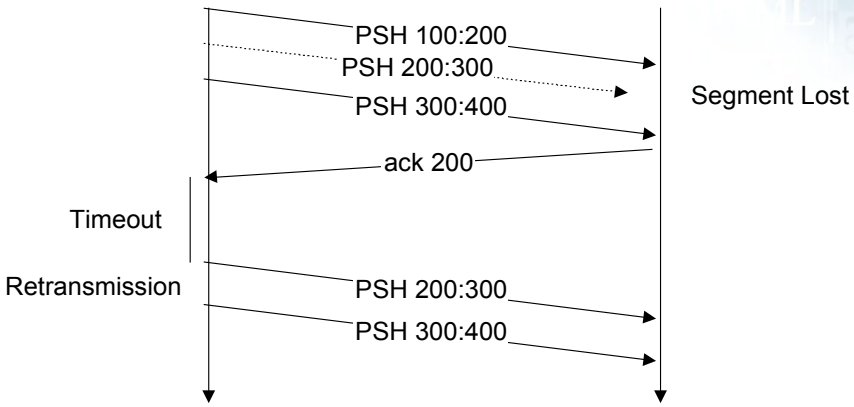
Acknowledgements and Retransmissions

- Each acknowledgement specifies a sequence value one greater than the highest octet position it received.
- Acknowledgements are cumulative relative to the ISN.
- On timeout, data is retransmitted from the sending steam starting at highest acknowledgement received.

Timeouts

- Delay incurred by consecutive IP packets belonging to the same TCP connection can incur different amounts of delay.
- TCP notes the time at which a segment was sent and it's acknowledgement arrives. Round trip time is computed from this data.
- Round trip time samples are constantly applied to the retransmission algorithm to calculate an acceptable timeout for the connection.

Timeout and Retransmission



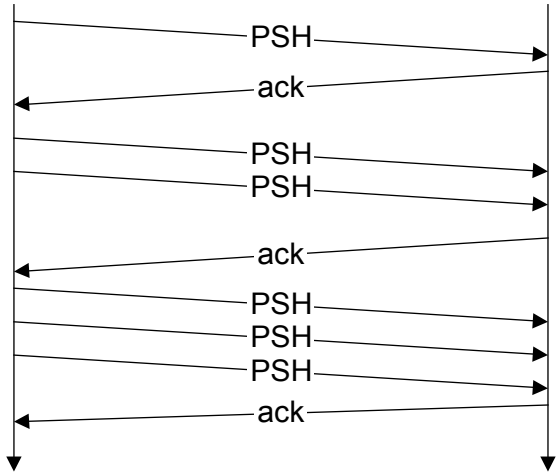
Congestion Window

- Congestion window is a value maintained by the TCP protocol.
- In a non congested connection the connection window is the same size as the receivers advertised window.

Congestion Response and Avoidance

- Multiplicative Decreased Congestion Avoidance- Upon loss of a segment, reduce the congestion window by half and backoff the retransmission timer exponentially.
- Slow-Start/Recovery – Whenever starting traffic on a new connection or increasing traffic after a period of congestion the congestion window starts at the size of a single segment (MTU) and increases by one segment each time an acknowledgement arrives.

Slow Start



Silly Window Syndrome

- When opening a window wait for space to become available that is at least MSS or 50% of the total RX buffer size.
- Only allow one < MSS sized segment in flight at one time (Nagle Algorithm).

Advanced TCP Topics

- Window Scaling
  - 2^16/RTT to 2^32/RTT
- TCP Selective Acknowledgement (SACK)
- Fast Retransmit/Fast Recovery
- Protect Against Wrapped Sequence Numbers (PAWS)
- Explicit Congestion Notification (ECN)
- URG data
- Keep Alive

Questions?