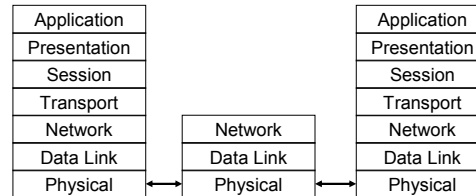


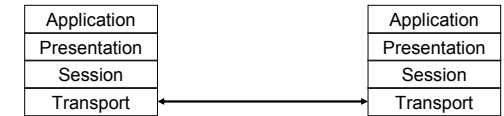
COMP476
Networked Computer Systems

Transmission Control Protocol

Network Layers



Transport Layer View



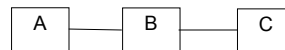
The transport layer is not concerned with routing or the existence of any intermediate nodes.

Reliable Delivery

- TCP operates on top of the Internet Protocol, a connectionless, unreliable network.
- TCP provides a connection oriented transport that corrects lost packets, corrupted packets, out-of-order packets and delayed packets.

Protocol Simulation

- Divide into groups of 3. Each group will have an “A”, “B” and “C”.
- Student A will send a sentence to C who will simultaneously send a sentence to A.
- A and C will send messages by handing them to B to *maybe* give to the other student.



Before You Start

- The whole group should determine how your protocol will work before you start.
 - How will you detect a missing packet?
 - What will you do when a packet is missing?
 - What will you do when a packet is correctly received?

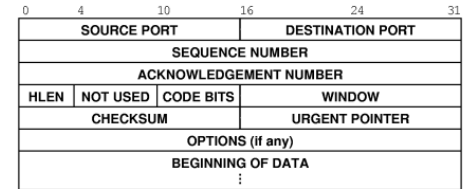
Try It

- A and C will write their sentences on small pieces of paper, one English word per piece of paper. They will “transmit” their sentence to the other student by giving the paper to B.
- If B is feeling nice, he or she will pass the message on without any changes. Otherwise B may discard the piece of paper (lost packet) or pass on a blank piece of paper (corrupted packet).

Creating Reliability

- Each message has a sequence number. The receiving side can recognize missing or out of order messages.
- Checksums detect corrupted messages.
- All correctly received messages are acknowledged by sending an ACK to the sender.

TCP Header



Initial sequence numbers are randomly chosen and sent to the other side during connection setup.

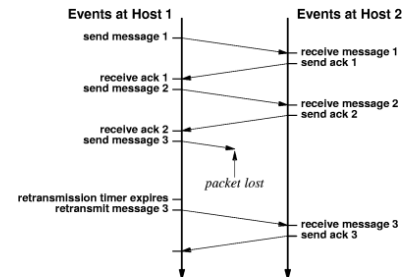
Acknowledgements

- When a packet is sent, the source sets a timer that will interrupt after a set time period. The sent packet is kept.
- When the destination gets the packet correctly, it sends an ACK to the sender.
- When the source receives the ACK, it discards the sent packet.
- If the source does not receive the ACK before the timer expires, it resends the packet.

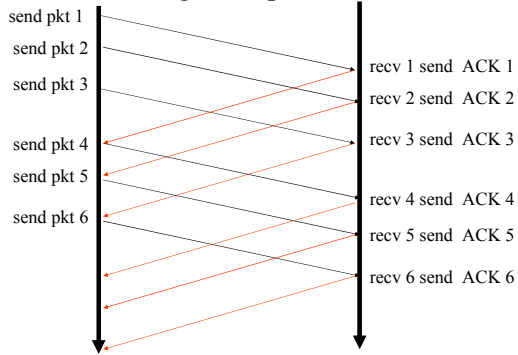
Possible Solution

- Each packet has a sequence number and an ACK number.
- When you receive a packet, send a response with the ACK number set to the sequence number.
- If you don't receive an ACK after 15 sec or so, resend all messages since the last ACK.
- You can discard sent messages when you get an ACK for them.

Simple ACK Example



Sending Multiple Packets



Flow Control

- The application sending data may operate at a different speed than the application receiving the data.
- The receiving application could run out of buffer space if it could not tell the sender to stop sending more data.

TCP Window Mechanism

- The receiver sends a window advertisement to indicate how much buffer space it has available.
- The sender cannot send more data than the receiver has space.
- As data is received, the acknowledgements contain a smaller and smaller window.

TCP Window Mechanism

- When the window advertisement is zero, the sender must stop transmitting any more data.
- When the application on the receiving side consumes some data, an acknowledgement is sent with a new window value.
- Communication is full duplex, so both sides have separate window values.

