# COMP 790-088
## Networked and Distributed Systems

# Transport Layer
## *Multiplexing and Connection Setup*

*Jasleen Kaur*

September 28, 2009



## Transport Protocols
### What Issues Will We Focus On?

- Basic functionality:
  - » Provide a "process-to-process" communication channel
    - ❖ Versus the host-to-host abstraction provided by the lower layers
- Wish-list from such a channel:
  - » Guarantees message **delivery**
  - » Guarantees **in-order** delivery
  - » Guarantees **no duplicate** messages
  - » Supports arbitrarily **large** messages
  - » Guarantees **bounded delay**
    - ❖ Helps support synchronization between the sender and the receiver
  - » Allows receiver to **control data flow** from sender
  - » Supports **multiple application** processes on each host
  - » **Security**

***How to provide these services on top of IP's best-effort service?***

# User Datagram Protocol (UDP)
## Multiplexing/Demultiplexing

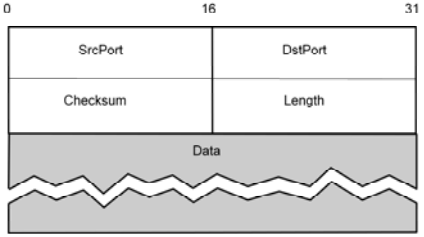
- Simplest possible service
  - » Just extend from host-to-host to process-to-process communication
  - » Simply add a level of de-multiplexing
    - ❖ Since there may be several processes running on a host
- How to identify processes?
  - » Process IDs
    - ❖ ☹: OS-dependent (will work only in networks where everyone runs same OS)
  - » Ports/mailboxes:
    - ❖ Abstract locator for sending messages to and receiving messages from
    - ❖ Allows multiple channels to be established in same process
    - ❖ Host-local scope: host ID + port number uniquely identify a channel
- How to learn of destination port number?
  - » Servers use well-known port numbers (or port-mapper service)

*COMP 790-088* 3



# User Datagram Protocol
## Segment Format

- 16-bit port identifiers used
  - » Considered enough for host-local scope
- Checksum is optional
  - » Same algo as IP checksum (sum of 16-bit words)
  - » Computed on the "pseudo-header" + UPDheader + Data
    - ❖ Pseudo-header: (protocol, srcIP, destIP, UDPlength) fields
    - ❖ Includes fields already included in the IP checksum

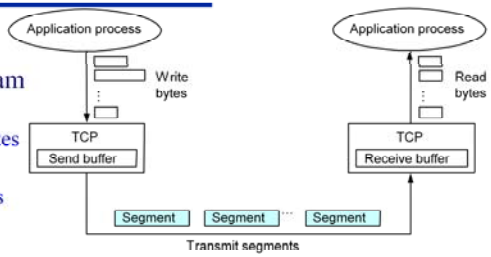



*COMP 790-088* 4

# Transmission Control Protocol (TCP)

## Services Offered

- Connection-oriented, full-duplex, byte-stream service
  - » Application writes bytes
  - » TCP sends segments
  - » Application read bytes



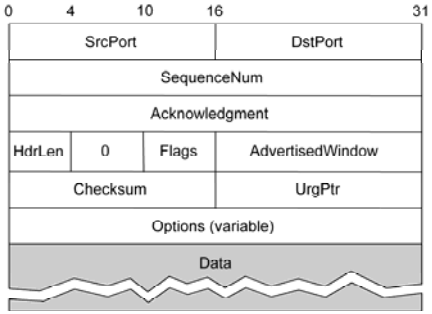

- Service
  - » Reliable delivery
  - » In-order delivery
  - » Receiver-limited flow control
    - ❖ An end-to-end issue
  - » Congestion control
    - ❖ Host-to-network interaction

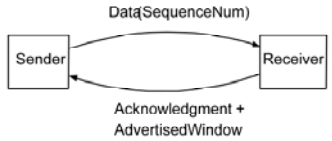


# TCP

## Segment Format

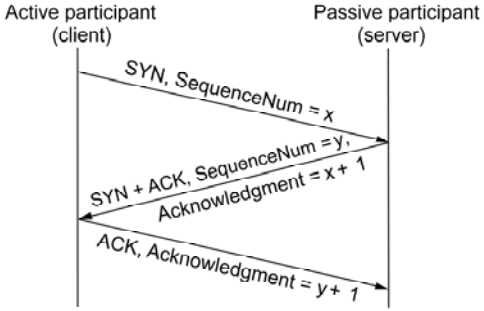
- Ports + IP addresses uniquely identify a TCP connection (what about UDP?)
- Each byte has a sequence number (why?)
- ACK, AdvWin carry info about data sent in opposite direction
- Header length in units of 32 bit words
- Flags used for control information (SYN, FIN, ACK, RESET, URG, PUSH)



# TCP
## Connection Establishment

- Three-way Handshake:
  - » Two sides agree on starting sequence numbers to use
    - ❖ Why not start from 0?
  - » SequenceNum = Next sequence expected

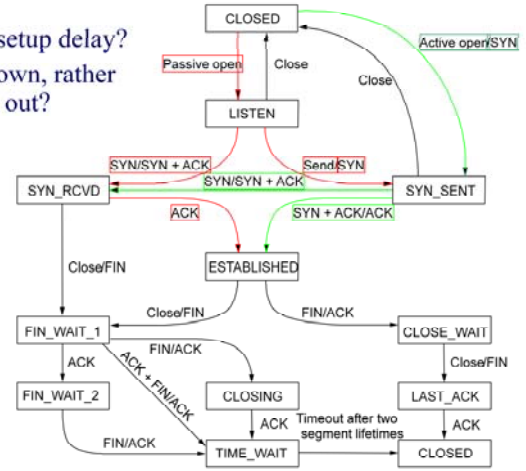



COMP 790-088 7



# TCP
## State Transition Diagram

- Why incur 1-RTT setup delay?
- Why explicit teardown, rather than simply timing out?




8