Last Time

- We finished up the Network layer
  - Internetworks (IP)
  - Routing (DV/RIP, LS/OSPF)
  - Scalable addressing/routing (BGP, CIDR)
  - Routers
This Time

• We begin on the Transport layer

• Focus
  – How do we send information reliably?

• Topics
  – The Transport layer
  – Acknowledgements and retransmissions (ARQ)
  – End-to-End argument (E2E)
The Transport Layer

• Builds on the services of the Network layer

• Communication between processes running on hosts
  – Naming/Addressing

• Stronger guarantees of message delivery
  – Reliability
Example – Common Properties

TCP
• Guaranteed delivery
• In-order delivery
• Single delivery
• Arbitrarily long messages
• Synchronization
• Flow control
• Multiple processes

IP
• Lost packets
• Reordered packets
• Duplicate packets
• Limited size packets
Internet Transport Protocols

- **UDP**
  - Datagram abstraction between processes
  - With error detection

- **TCP**
  - Bytestream abstraction between processes
  - With reliability
  - Plus congestion control (later!)
Automatic Repeat Request (ARQ)

- Packets can be corrupted or lost. How do we add reliability?
- Acknowledgments (ACKs) and retransmissions after a timeout
- ARQ is a generic name for protocols based on this strategy
The Need for Sequence Numbers

- In the case of ACK loss (or poor choice of timeout) the receiver can’t distinguish this message from the next – Number packets; here, a single bit will do
Stop-and-Wait

- Only one outstanding packet at a time
- Also called alternating bit protocol
Limitation of Stop-and-Wait

- Lousy performance if wire time $\ll$ prop. delay
  - How bad? You do the math
- Want to utilize all available bandwidth
  - Need to keep more data “in flight”
  - How much? Bandwidth-delay product
- Leads to Sliding Window Protocol
Sliding Window – Sender

- Window bounds outstanding data
  - Implies need for buffering at sender
- “Last” ACK applies to in-order data
- Sender maintains timers too
  - Go-Back-N: one timer, send all unacknowledged on timeout
  - Selective Repeat: timer per packet, resend as needed
Sliding Window – Timeline

Sender

Receiver

Time

...
Sliding Window – Receiver

Receiver: ...

Receive Window

“Last” Received  Largest Accepted

- Receiver buffers too:
  - data may arrive out-of-order
  - or faster than can be consumed (flow control)

- Receiver ACK choices:
  - Individual, Cumulative (TCP), Selective (newer TCP), Negative
Sliding Window Functions

- Sliding window is a mechanism
- It supports multiple functions:
  - Reliable delivery
  - In-order delivery
  - Flow control
Which layer provides Reliability?

- We’ve been talking about the Transport layer but ...

- ARQ is used by some link layers
  - Acknowledgements in 802.11

- Error detection/correction codes boost reliability
  - Ethernet CRC, IP header checksum, etc.

- Where is the “right” place in the protocol stack?
End-to-End Argument

- Key design principle applied in the Internet
- Reliability is needed end-to-end and can’t be replaced by lower layer mechanisms. So put it end-to-end; use lower mechanisms to improve performance as needed.

- TCP provides reliable delivery
  - Checksums packet data as well
- Lower layers keep their residual error rate is low
  - CRC enough for Ethernet; wireless links more problematic
Key Concepts

- Transport layer allows processes to communicate with stronger guarantees, e.g., reliability
- Basic reliability is provided by ARQ mechanisms
  - Stop-and-Wait through Sliding Window
- End-to-End principle guides placement of functions

coming next: Connections and Congestion Control
Read Keshav 12.4 and Ch 13, esp. 13.4