CS 168 Introduction to the Internet: Architecture and Protocols
Fall 2022 Sylvia Ratnasamy **Discussion 8: Reliable Delivery, TCP**

1 TCP in Action

Consider a sender sending 1000 B of data to a receiver over **TCP**. The sender sends packets of 100B, the window size is 300B, and the ISN is 99 (so D100 is the first packet sent, then D200, and so on). Remember, TCP uses a sliding window, and retransmits the packet containing the next expected byte on a timeout or, if TCP Fast Retransmit is enabled, when three duplicate acks are received. Assume here that TCP Fast Retransmit is not enabled.

The link is **flaky**! The **initial** transmission of packets **D200 and D700** get dropped.

1. Fill in the below table with all packets sent by the sender until the receiver has received all packets and the sender knows that. For simplicity, assume that packets (data and ACKs) arrive in order. You may or may not need to fill in all lines.

#	Packet Data Offset	Sent on timeout	Dropped?	*Cumulative* ACK
1	D100			A200
2	D200		X	
3	D300			A200
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				

2. If the RTT of the link is 10ms and the timeout is initially 3 seconds, what is the total time needed for the receiver to receive all packets and for the sender to know that? Assume small packets (negligible transmission delay) and negligible processing time, and that the estimates that go into the Retransmission Timeout (RTO) remain constant during the events below.

2 TCP Calculations

In this question, quantities will be measured in bytes unless **explicitly** mentioned otherwise.

1. Suppose two hosts are about to open a TCP connection. The TCP headers used in the communication are only 20 bytes long and regular (no-options) IPv4 is being used for Layer 3. If the MTU of the link is 1260 bytes, what is the MSS?

2. When this connection starts, the sender starts with an ISN 19. The initial window for the sender is set to 10 packets. Given the previously calculated MSS, what ACK does the sender receive as part of the TCP handshake? After that, what is the first and last ACK the sender receives for this initial window? (Assume no packets were lost or reordered).

3. In this part of the question we will determine the estimated RTT using some parameters of TCP. The following equations *may* be useful. Assume that connections have been open and that there are **no** dropped packets.

$$Estimated\_Timeout(ETO) = Estimated\_RTT + 4 \cdot Estimated\_Deviation$$
$$Estimated\_Deviation = \beta \cdot Estimated\_Deviation + (1-\beta) \cdot |Estimated\_RTT - Sample\_RTT|$$
$$Estimated\_RTT = \alpha \cdot Estimated\_RTT + (1-\alpha) \cdot Sampled\_RTT$$

   (a) The sender receives the current ACK and proceeds to update its various estimations. The time from when the packet was sent to when the ACK was received was $10msec$. If the previous $Estimated\_RTT$ was $70msec$, what will the new $Estimated\_RTT$ be (using $\alpha = 0.5$)?

   (b) If the previous $Estimated\_Deviation$ was $50msec$, what is the new $Estimated\_Deviation$ (using $\beta = 0.5$)?

   (c) Based on the previous 2 questions, what will the ETO be now?

4. What is the maximum theoretical rate of data transfer for this window size if the $Estimated\_RTT$ is what was previously estimated?

5. Assume $40msec$ is the $Estimated\_RTT$. If the lowest bandwidth across this connection is $76.25MBps$, what is the smallest window that optimizes the question?

3 Reliable Transport

Bob thinks that using ACKs on every packet is very wasteful, and wants to design a transport protocol that is reliable but uses very few ACKs. He comes up with a scheme that he believes provides reliability but only sends at most $log(n)$ ACKs, where n is the number of packets. The protocol is as follows:

- Let P_i be the i^{th} packet. When the receiver sees *any* of the packets in the set

$$\{P_{2^i} \mid 1 \leq i \leq \lfloor log_2 n \rfloor\} \cup \{P_n\} \tag{1}$$

  it sends back an ACK for that packet.

- The sender will send packets $(P_{2^{i-1}}, P_{2^i}]$ in order (i starts at 0), and wait for the last packet to be ACKed. If the sender does not hear back after some time, it sends this window of packets again until the last packet is ACKed. Then, i is incremented and sends the next window of packets. One can think of this as a window that starts at packet 1 with size 1. Whenever the last packet in a window is ACKed, the window size is doubled and the sender moves to the next set of packets. This process repeats until P_n is ACKed.

(1) Is this transport protocol reliable? Why or why not?

(2) If you said the protocol was not reliable, what is a modification that you could make in order to fix that? Try to make the smallest change possible.

(3) Given that the protocol is reliable (or it wasn't and you apply your fix from the last part), does it actually save any bandwidth? Why or why not?

Alice thinks that Bob is onto something and wants to design a transport protocol that uses fewer ACKs. She decides that we can cut the number of ACKs in half by only acknowledging according to the following rules:

- Let P_{2i} be the i^{th} even packet. The receiver sends an ACK iff the receiver has received P_{2i} and P_{2i+1}. Note that we count packets indexing at 0 for the first packet.

- The sender continues to send packets until all even packets have been ACKed.

(1) Is this transport protocol reliable? Why or why not?

(2) Is there a modification to the protocol that you could make in order to fix it?

(3) (optional) Does this protocol actually reduce the number of ACKs sent? (Note: you won't be tested on this)

4 Pop Quiz: TCP vs UDP

For each statement below, mark whether describes TCP, UDP, both (mark both) or neither (mark none).

	TCP	**UDP**
Provides reliable transport		
Limits messages to a single packet		
Has source port in the header		
Requires connection establishment		
You will use this for your project 2.		
It is optional to use the L4 checksum with this protocol (under IPv4)		

5 TCP Short Questions

1. The purpose of a TCP handshake is: (circle all the apply)
   (a) Connection establishment
   (b) Exchange of initial sequence number
   (c) Transfer of client's request data
   (d) To indicate a packet loss
2. Which of the following is provided by TCP, but not by UDP? (circle all that apply)
   (a) Reliable transfer
   (b) Congestion control
   (c) Mux and demux from/to application processes
   (d) Byte-stream abstraction
3. What flag is usually set on the first packet in a TCP exchange?
   (a) RST
   (b) SYN
   (c) IETF
   (d) FIN
4. What message type does TCP send to abruptly terminate a connection?
   (a) RST
   (b) SYN
   (c) IETF
   (d) FIN