CMPE 150/L – Winter ‘19

Sample questions #3 solutions

1. Rdt protocols use timers to be able to react to losses in the network.

   (a) What kind(s) of losses do timers help detect? Explain how they do that.
   Timers help detect packet drops as follows: they are set to a value (the retransmission timeout) and, when a segment is transmitted, the timer starts counting down. When they reach 0, they trigger a "timeout" event to the sender, who then assumes the segment got lost, and retransmits it.

   (b) What are the trade-offs in setting up the retransmission timer in rdt protocols?
   If the retransmission timeout value is set too low, it may cause unnecessary retransmissions, which result in adding load to the network unnecessarily.

2. TCP assumes that packet loss is an indication of congestion.

   (a) Is that a reasonable assumption when applied to the Internet? Why or why not?

   One can argue that this assumption is a reasonable one when considering the Internet since packet losses due to transmission errors are relatively rare, as the Internet has become quite reliable as far as its underlying communication infrastructure. Furthermore, reliability implemented by data-link layer protocols also help recover from eventual transmission errors.

   (b) Is that a reasonable assumption for networks that use wireless links? Why or why not?

   In the case of wireless networks (especially in the case of multi-hop wireless links), transmission errors may occur more often. Therefore, in these environments, assuming that losses are always due to congestion is no longer valid and may cause significant performance deterioration.

3. In the computation of TCP's RTT, how can we make TCP more responsive to the current conditions of the network?

   TCP’s RTT computation uses an exponentially weighted moving average expression, which accounts for both current RTT measurements and past RTT estimates. The parameter α controls how much weight is given to current measurements over past history. In order to make TCP more responsive to current conditions, more weight can be assigned to current RTT measurements. To this end, a higher value of α can be used.

4. TCP and acknowledgments

(a) Why does TCP use cumulative acknowledgments?

TCP primarily uses a cumulative acknowledgment scheme, where the receiver sends an acknowledgment signifying that the receiver has received all data preceding the acknowledged sequence number. TCP's cumulative acknowledgement scheme avoids generating excessive control overhead (in the form of ACK segments).

(b) In order to use cumulative acknowledgments, does the TCP receiver have to employ a timer? Explain.

The TCP receiver needs a timer for the cumulative ACK mechanism so that, in case there is no additional data being transmitted by the sender, the timer will cause a timeout and will trigger a regular ACK to be generated.

(c) What are duplicate acknowledgments?

Duplicate ACKs are used by the receiver to signal out-of-order delivery of segments, which may be caused by segment(s) being lost or delayed. TCP's Fast Retransmit mechanism is triggered by 3 duplicate ACKs, which cause immediate retransmission of the supposedly lost segment. Fast Retransmits tries to avoid timeouts and the resulting reset of TCP's congestion window to 1 segment.

5. Use a time diagram to show an example scenario of a premature timeout in TCP.

[Look at one of the examples with did in class.]

6.

(a) Use a time diagram to show an example scenario of a packet loss event and how the TCP sender and receiver detect and react to it.

[Look at one of the examples we did in class.]

(b) Same for a lost acknowledgment.
[Same here.]

(c) Do the TCP sender and receiver behave the same way in both scenarios? Explain.

In the case of a packet (segment) loss, the sender will timeout and retransmit the lost segment. Upon reception, the receiver will take the retransmission and send an acknowledgement. In the event of a lost ACK, the sender will also timeout and retransmit; however, the receiver will discard the duplicate segment (since it had received the original transmission) and generate another ACK.