CS 444
Computer Networks
Spring, 2017

Homework #3

Due: Feb 26, at 10pm, via Blackboard

This assignment will take quite some time, so please start early. I suggest reading Chapter 3 and starting the assignment, even before lecture, doing what you can and holding off on problems where you need the lecture.

Show all your work. Partial credit will be given.

Problem 1 (R4). Describe why an application developer might choose to run an application over UDP rather than TCP.

Problem 2 (R6). Is it possible for an application to enjoy reliable data transfer even when the application runs over UDP? If so, how?

Problem 3 (R10). In our rdt [discussed in class and in chapter] protocols, why did we need to introduce timers?

Problem 4 (R14). True or false?

a. Host A is sending Host B a large file over a TCP connection. Assume Host B has no data to send Host A. Host B will not send acknowledgments to Host A because Host B cannot piggyback the acknowledgments on data.
b. The size of the TCP rwnd never changes throughout the duration of the connection.
c. Suppose Host A is sending Host B a large file over a TCP connection. The number of unacknowledged bytes that A sends cannot exceed the size of the receive buffer.
d. Suppose Host A is sending a large file to Host B over a TCP connection. If the sequence number for a segment of this connection is m, then the sequence number for the subsequent segment will necessarily be $m + 1$.
e. The TCP segment has a field in its header for rwnd.
f. Suppose that the last SampleRTT in a TCP connection is equal to 1 sec. The current value of TimeoutInterval for the connection will neces- sarily be ≥ 1 sec.
g. Suppose Host A sends one segment with sequence number 38 and 4 bytes of data over a TCP connection to Host B. In this same segment the acknowledgment number is necessarily 42.

Problem 5 (R15). Suppose Host A sends two TCP segments back to back to Host B over a TCP connection. The first segment has sequence number 90; the second has sequence number 110.

a. How much data is in the first segment?
b. Suppose that the first segment is lost but the second segment arrives at B. In the acknowledgment that Host B sends to Host A, what will be the acknowledgment number?

Problem 6 (P3). UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01010011, 01100110, 01110100. What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sums.) Show all work. Why is it that UDP takes the 1s complement of the sum; that is, why not just use the sum? With the 1s complement scheme, how does the receiver detect errors? Is it possible that a 1-bit error will go undetected? How about a 2-bit error?

Problem 7 (P5). Suppose that the UDP receiver computes the Internet checksum for the received UDP segment and finds that it matches the value carried in the checksum field. Can the receiver be absolutely certain that no bit errors have occurred? Explain.

Problem 8 (P14). Consider a reliable data transfer protocol that uses only negative acknowledgments (NAKs). Suppose the sender sends data only infrequently. Would a NAK-only protocol be preferable to a protocol that uses ACKs? Why? Now suppose the sender has a lot of data to send and the end-to-end connection experiences few losses. In this second case, would a NAK-only protocol be preferable to a protocol that uses ACKs? Why?

Problem 9 (P15). Consider two hosts on opposite sides of the US, connected by a 1Gbps (10^9 bits per second) connection. The two hosts have a 30ms RTT. How big would the window size have to be for the channel utilization to be greater than 98 percent? Suppose that the size of a packet is 1,500 bytes, including both header fields and data.

Problem 10. Consider the problem of *multicasting* packets from one sender to multiple receivers. In multicast, the sender transmits a single copy of a packet, and the network and data link layer(s) handle the problem of routing a copy of that packet to multiple receivers. Could you use TCP with multicast? If not, describe how you would build an alternative *reliable* protocol that achieves reliable in-order delivery to all receivers.

Problem 11 (P28). Host A and B are directly connected with a 100 Mbps link. There is one TCP connection between the two hosts, and Host A is sending to Host B an enormous file over this connection. Host A can send its application data into its

TCP socket at a rate as high as 120 Mbps but Host B can read out of its TCP receive buffer at a maximum rate of 50 Mbps. Describe the effect of TCP flow control.

Problem 12. Imagine a bad guy forges many initial SYN packets (as though for a TCP handshake) and labels them each with different, forged source IP addresses. What is the effect of this on the server? At some point will the server stop being able to process incoming connections?

Problem 13 (P33). In class and in the book we discussed TCP's estimation of RTT. Why do you think TCP avoids measuring the SampleRTT for retransmitted segments?

Problem 14 (P55). In this problem we investigate whether either UDP or TCP provides a degree of end-point authentication.

h. Consider a server that receives a request within a UDP packet and responds to that request within a UDP packet (for example, as done by a DNS server). If a client with IP address X spoofs its address with address Y, where will the server send its response?

i. Suppose a server receives a SYN with IP source address Y, and after responding with a SYNACK, receives an ACK with IP source address Y with the correct acknowledgment number. Assuming the server chooses a random initial sequence number and there is no "man-in-the-middle," can the server be certain that the client is indeed at Y (and not at some other address X that is spoofing Y)?