

CS485/ECE440: Homework 6

November 20, 2008

General

- This homework is (mostly) about material covered in Chapter 5, 6 and 7 of the text book.
- The answers to this homework are **due December 2, 2008**.
- Submit via e-mail to riesen@cs.unm.edu (Mail it before class on the 30th.) Your subject line must say: “Homework 6 Submission” (and **nothing else**).
- Your answers to the questions below must be in a PDF or plain text file attached as file named Homework6.pdf or Homework6.txt according to its format. (Not Hwk6.pdf, and no Microsoft Word or image files, such as jpeg, please.)
- Keep your answers (and programs) succinct.
- Obey the University rules on plagiarism. In particular, do use libraries and the web to find information you need to answer the questions, but do not copy whole answers or programs. Reference your sources. The work you turn in must be *your* work.

Exercise 1: Silly window syndrome [10 points]

Briefly explain what the silly syndrome is. What algorithm is used to prevent the syndrome from happening? How does that algorithm work, and what are some of the problems associated with it?

Exercise 2: TFTP [30 points]

TFTP is a simple file protocol. It is sometimes used to boot systems remotely. The booting machine sends out a TFTP request and a server responds with a kernel image that the client uses in the second stage of the boot.

Request For Comments (RFC) are documents that describe protocols and procedures used on the Internet. RFC 1350 describes TFTP. Look at RFC 1350 and answer the following questions:

1. Why would one use such a simple protocol running on top of an unreliable data link layer protocol such as UDP?
2. TFTP does not use a sequence number in its header. How does TFTP avoid duplicate packets?
3. How does TFTP know it has received the entire file?

4. The initial file request packet of TFTP is variable size. It contains the variable length file name. How does TFTP know how long that file name is?
5. How does TFTP know how long a data packet is?
6. **Bonus:** [10 points] show the datagrams needed to transfer a 1536 byte file.

Exercise 3: TCP state diagram [10 points]

Consider the TCP state diagram shown in Figure 5.7 on page 392 of our textbook. When a connection is closed, we have to wait in the `TIME_WAIT` state for twice the maximum segment lifetime before we can go to the `CLOSED` state. However, if we take the right-most path through the diagram we can reach `CLOSED` via `CLOSE_WAIT` and `LAST_ACK` without having to wait. Why can we do that without waiting as in the other path? Is it possible for both sides of the connection (the client and the server) to take this shortcut?

Exercise 4: TCP data transmission [10 points]

Assume host A sends two TCP segments to host B; one right after the other. The sequence numbers of the two segments are 100 and 120 respectively. Answer these questions:

1. How much data does the first segment contain?
2. If only the second segment arrives at host B (because the first one has been lost), then what will the acknowledgment number be that is sent from host B to A?

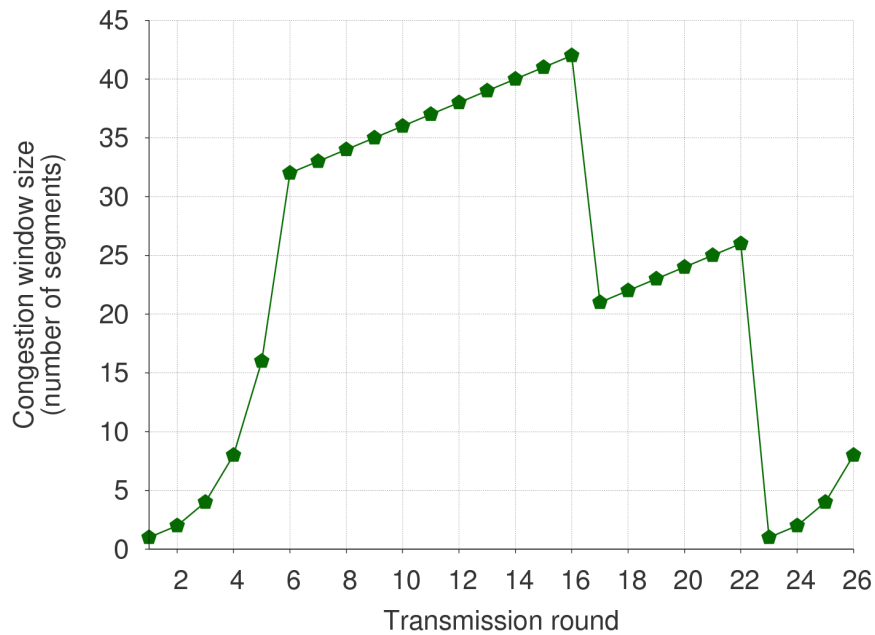
Exercise 5: Router queue management [10 points]

Do exercise 6.11 on page 529 in the textbook. However, instead of using Table 6.2 in the book, use the one given here.

Packet	Size	Flow
1	210	1
2	210	1
3	210	1
4	140	2
5	230	2
6	90	3
7	110	3
8	100	3

Exercise 6: TCP congestion window [20 points]

Have a look at the following figure and answer the questions below. The figure represents the congestion window size during the first part of a TCP connection. Each step on the x -axis represents one round trip delay. The numbers on the y -axis represent the size of the congestion window in number of segments.



For each answer you should briefly describe how you arrived at your answer.

1. Can you identify the time intervals when TCP is performing a slow start?
2. At time 16, a segment was lost. Did TCP detect that through a timeout or a duplicate ACK?
3. Answer the above question for time 22.
4. At what time is segment number 70 sent?
5. Assume that another segment is lost and TCP detects it at time 26 due to the arrival of a third duplicate ACK. What will the congestion window size be at time 27?

Exercise 7: Presentation [10 points]

In the little program you wrote as part of homework 2, you used the function `htonl()` to convert the local representation of an integer to its network representation.

1. Why did you have to do that?
2. What would have happened, if you did not do that?
3. Using the `hton` functions seems tedious and error prone. What could you do as a network designer to avoid the use of these functions? Name at least two alternatives.