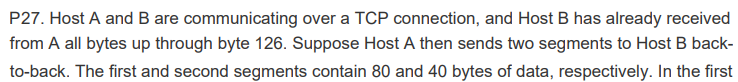
Computer Network

Homework #1

19102100 Choi Jiwoo

**#27, 28, 32, 34, 36, 40**



텍스트, 스크린샷, 폰트, 문서이(가) 표시된 사진

자동 생성된 설명

Answer

1. Since previous sequence number was 127, the sequence number of second segment is 207 (127+80). The port number is 302 and destination number is 80.
2. The acknowledgement number is 207. The port and destination number is reversed so that it is 80, 302, respectively.
3. If the second segment arrives before the first segment, however, the receiver will wait for 127 bytes, so the acknowledgement number is 127.
4. 도표, 텍스트, 라인, 그림이(가) 표시된 사진

   자동 생성된 설명

**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.

Answer

Since the capacity of the link is only 100Mbps, the highest rate will be 100Mbps. However, still the host A sends data into the receiver’s buffer faster than Host B’s capacity of handling received data. As a result, the buffer of the receiver will be full someday, and host B will let host A not send data anymore by setting RcvWindow = 0. Host A will be stopped and continue the next transmission after RcvWindow is changed to more than 0. On average, the long-term rate at which Host A sends data to Host B is less than 60Mbps.

**P32**. Consider the TCP procedure for estimating RTT. Suppose that α=0.1. Let SampleRTT1 be the most recent sample RTT, let SampleRTT2 be the next most recent sample RTT, and so on.

1. For a given TCP connection, suppose four acknowledgments have been returned with corresponding sample RTTs: SampleRTT4 , SampleRTT3 , SampleRTT2 , and SampleRTT1 . Express EstimatedRTT in terms of the four sample RTTs.
2. Generalize your formula for n sample RTTs.
3. For the formula in part (b) let n approach infinity. Comment on why this averaging procedure is called an exponential moving average.

Answer

1. *For x = 0.1*

*EstimatedRTT*

*= xSampleRTT1 + (1-x)[SampleRTT2 + (1-x)[xSampleRTT3 + (1-x)SampleRTT4]]*

*= xSampleRTT1 + (1-x)xSampleRTT2 + (1-x)2xSampleRTT3 + (1-x)3SampleRTT4*

1. *EstimatedRTT(n) = x*
2. *EstimatedRTT(*∞*) = x*

**P34**. What is the relationship between the variable SendBase in Section 3.5.4 and the variable LastByteRcvd in Section 3.5.5?

Answer

At a given time t, SendBase – 1 is the sequence number of the last byte that the sender knows to have been correctly received in order by the receiver. If there is an acknowledgment in the pipe, the last byte actually received (correctly and in order) from the receiver at time t may be larger. Thus, SendBase–1 is equal or smaller than LastByteRcvd.

**P36**. In Section 3.5.4 , we saw that TCP waits until it has received three duplicate ACKs before performing a fast retransmit. Why do you think the TCP designers chose not to perform a fast retransmit after the first duplicate ACK for a segment is received?

Answer

Reliability must be guaranteed in TCP, but that doesn't mean speed should be sacrificed. The important thing is to provide communications that are both reliable and fast. If you perform a fast retransmit immediately when the first duplicate occurs, the reliability of the protocol will be guaranteed, but you will inevitably experience a slowdown in speed. Waiting until the third duplicate provides additional assurance that packet loss is not a temporary network issue. Therefore, I believe that TCP designers designed the network to maintain fast retransmissions while ensuring reliability by making a tradeoff of increasing the number of duplicates tolerated.

텍스트, 라인, 스크린샷, 도표이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Answer

1. Since the exponential graph which begins at around 1 second ends at around 6 seconds, TCP slow start phase interval is about 5 seconds. Also there is another exponential graph between 23 and 26 seconds. As a result, intervals of time when TCP slow start is operating are [1,6], [23,26]
2. When it comes to linear graph after exponential one, we consider it congestion avoidance phase. Here, [6, 16] and [17, 22]
3. After the 16th transmission round, segment loss was detected by a triple duplicate ACK because if timeout had happened, then the ConWin(Congestion window) size would have been set to 1.
4. Since the ConWin size was set to 1, the reason for segment loss was timeout.
5. It was about 32 because the congestion avoidance began at that point.
6. When the packet loss is detected, *ssthresh* is set to half of current ConWin value. Hence, ssthresh at the 18th transmission round is 42/2=21
7. Same case of f, therefore the value will be 29/2 ~= 14 (taking the lower floor of 14.5)
8. During first slow start phase : 1+2+4+8+1zx 6+32 = 63. During 7th transmission round, 32 bytes will be sent including 70th segment.
9. ConWin size will be 7 which is half of the current value added by 3 (considering previous segment loss caused by 3-duplicates) and ssthresh will be set to 4.
10. Since TCP Tahoe does not distinguish three duplicate and timeout case, congestion window after segment loss will be 1. The value of threshold will be 21, the half of current congestion window value. Thus, Threshold is 21, and congestion window size is 1.
11. At 17th round, congestion window size begins with 1 and it will be increased exponentially. 17th~22nd = 1+2+4+8+16+21(threshold) = 52