Assignment #02

1. Consider the following behavior of a TCP connection (using the congestion control algorithm we learned in class).

At time 0, a TCP sender initiates a connection. As soon as the connection is established, the TCP sender will begin sending data. The MSS is 1KB and RTT is 100 ms.

1) Assuming the connection does not lose any data or experience any timeouts, at what time will the sender's congestion window be 16KB? (Assuming *threshold* is 32MSS)

Answer:

First,build a TCP connection ,it costs 1 RTT.
The threshold is 32MSS=32KB>16KB,based on the congestion control algorithm,when the congestion window reaches 64KB and become lose data.
So,between start and the congestion window becomes 16KB,it is a Slow Start process,the congestion window grows exponentially.It needs $\log_2^{16}$=4 RTT
The total is 5 RTT=500ms.

So, **500ms** later,at time 5,the sender's congestion window is 16KB.

Right after the sender's congestion window has reached a size of 16KB, a timeout occurs. After the timeout is detected, the sender continues sending more data over the established connection.

2) Assuming no additional packets loss or timeouts, how long (since the observed timeout) will it take for the congestion window to build to size 14KB?

Answer:

Right before the timeout occurs,the congestion window = 16KB,threshold= 32MSS,
Then a timeout occurs,based on the congestion control algorithm,
First,the threshold becomes half of right congestion window,threshold=8KB,and the cwnd=1,and then enter Slow Start.
Second,the congestion window grows exponentially, until it reaches threshold(8KB),it need $\log_2^8$=3 RTT.
Then,TCP enters the congestion avoidance, and increases linearly until 14KB,it costs 14-8=6 RTT.
The total is 9 RTT=900ms.

So,since the observed timeout,it takes **900ms** to build to size 14KB.

3) While its congestion window is at 14KB, the sender receives triple duplicate acknowledgements for the same sequence number. How long after receiving the third duplicate acknowledgement will it take for the sender's congestion window to be at least 9KB again?

Answer:

Right before receive 3 duplicate ACK,the congestion window =14KB,the threshold=8KB.
Then receive 3duplicate ACK,based on the congestion control algorithm,
First,the threshold and congestion becomes half of right congestion window,threshold=congestion window =7KB,
Then,TCP enters the congestion avoidance and increases linearly until to 9KB,it need 2 RTT.
The total time is 2 RTT=200ms.

So,after receiving the 3 duplicate ACK,it takes **200ms** for the congestion window to reach 9KB.

2. Consider a scenario with two hosts, Alice and Bob. A web server running on Alice is trying to send data to a browser on Bob. For each TCP connection, Alice's TCP stack maintains a send buffer of 512 bytes and Bob's TCP stack maintains a receive buffer of 1024 bytes. For simplicity, assume TCP sequence numbers began at 0 in this problem.

1) Bob's stack received up to byte 560 in order from Alice, although its browser has only read up to the first 60 bytes. What will be the ***ACK#*** and ***rcvr window size*** in the TCP headers that Bob next sends to Alice?

Answer:

The receiver Bob's ACK#=sender's seq +1 meaning that the receiver has received the sender's data and has been expecting the seq+1 data.
The receiver Bob has received byte 560 ,it is the sequence of the data that the TCP transfer.The browser has read 60 bytes.rcvr window indicates the

size of available buffer space on the receiving side.

So ACK#=560+1=561;

Rcvr window size=1024-(561-60)=523 bytes (from 0 begin)

2) Later in the same connection, Alice's congestion window is set to 1 MSS = 536 bytes and the advertised flow-control window from Bob is 560 bytes. The last ***ACK#*** that Alice received from Bob is byte 700, and the last byte that Alice sends to Bob is byte 900.

A) What is the smallest byte number that Bob will not accept?

Answer:

The flow-control window from Bob is 560 bytes,and after Alice received ACK#700 ,she has sent 699 byte,she continue to send from 700 to 900 meaning continue to send 201 bytes.

So the flow-control window of Bob =560-201=359 bytes.

Alice can continue to send 359 bytes,until 900+359= byte 1259.

Bob can't receive any data which byte is larger than 1259 and it is 1260.

So the smallest byte number that Bob will not accept is **1260.**

B) Assuming that Alice doesn't receive any more ACKs and her window does not change, what is the greatest byte number that Alice can send?

Answer:

Window size=min{flow-control-window,congestion-window}
=min{560 ,536}=536 bytes.

Alice doesn't receive any more ACKs ,the window can't slid front.

But the send buffer is only 512 bytes,and no ACKs were received any more so no byte was cleared from send buffer.

512<536

Packet 700-900 is still in the window occupies 201 bytes,

512-201=311 bytes

Before has sent 900 bytes

900+311=1211 bytes.

So the greatest number of Alice can send is 1211 bytes.

C) Again assuming that Alice doesn't receive any additional ACKs, how many more bytes can the web server running on Alice write to its network socket before blocking?

Answer:

Alice doesn't receive any ACKSs,it mean Alice still need to retransfer packet 700-900 which size is 201.The window can't slid front,and this 201 packets is still in the buffer.The total buffer of Alice is 512bytes,512-201=311bytes.When use up the buffer,Alice will block.

So Alice can write 311 bytes.