Richard Aviles

10/03/17

Wireshark #3

1. The IP address of the client is 192.168.1.102. The port used for the client is 1161.
2. The IP address of the server is 128.119.245.12. The port being used by the server is 80.
3. I was having trouble running wireshark on my laptop so I was unable to use it to find my address and port number being used. I found my computer address by different ways using properties in my Wi-Fi network settings. My address is 10.88.9.224
4. The sequence number used to initiate the TCP connection is 0. It is identified as a SYN segment by the TCP flag being set to 0x002.
5. The sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN is 0. The acknowledgement field value is 1. It determined the acknowledgement value from the sequence number. It received 0 so it sent a 1 back to let the client know it received number 0 and is now waiting on segment 1. The TCP flag being set to 0x012 is what identifies the segment as a SYNACK segment.
6. The sequence number of the TCP segment containing the HTTP POST command is 1.
7. If you start at the TCP segment containing the HTTP POST and continue for six more sequence numbers, you will get the following: 1, 566, 2026, 3486, 4946, and 6406.

The sent times are: 0.026477, 0.041737, 0.054026, 0.054690, 0.077405, and 0.078157.

The ACK received times are: 0.053937, 0.077294, 0.124085, 0.169118, 0.217299, and 0.267802.

The RTT times are:

0.053937 - 0.026477 = 0.02746

0.077294 – 0.041737 = 0.035557

0.124085 – 0.054026 = 0.070059

0.169118 - 0.054690 = 0.114428

0.217299 - 0.077405 = 0.139894

0.267802 - 0.078157 = 0.189645.

The EstimatedRTT values are:

0.02746

(0.875 \* 0.02746) + (0.125 \* 0.035557) = 0.02847

(0.875 \* 0.02847) + (0.125 \* 0.070059) = 0.03367

(0.875 \* 0.03367) + (0.125 \* 0.114428) = 0.04376

(0.875 \* 0.04376) + (0.125 \* 0.139894) = 0.05578

(0.875 \* 0.05578) + (0.125 \* 0.189645) = 0.07251

1. The length of the first six TCP segments are: 565, 1460, 1460, 1460, 1460, and 1460.
2. The minimum amount of buffer space is 5840 bytes given from the window size in the very first ACK from the server. There is not anything shown in the packets trace files to show that the sender is throttled by the lack of receiver buffer space.
3. There are no retransmitted segments in the trace file. I used the Time-Sequence graph to see that nothing was sent again since all the packets continue up the graph.
4. For most of the ACKs, the amount acknowledged is 1460. I can identify a case where the receiver acknowledged two instead of one ACK. In the table, number 59 shows the receiver ACKing 35049 then in number 60 it ACKs 37969. It skipped the ACK for 36509 that was sent.
5. The throughput is the amount of data transferred over the time it took to transfer the data. The amount of data is what was ended with at 164091 minus what we started with in the beginning which was 1. Total data is 164090. The time is the ending time at 5.461175 minus the start time containing the HTTP POST at 0.026477. The total time taken is

5.434698. So the throughput is 164090/5.434698 = 30193 bytes per second.

1. From what I can see is the slow start phase is at the beginning. You can see how the dots there aren’t as vertical as the rest of the graph. After that then they seem to be pretty much the same.