**Computer Networks**

**Lab 3a**

**Wireshark Lab: TCP v8.0**

**Student Name: Phạm Ngọc Sang**

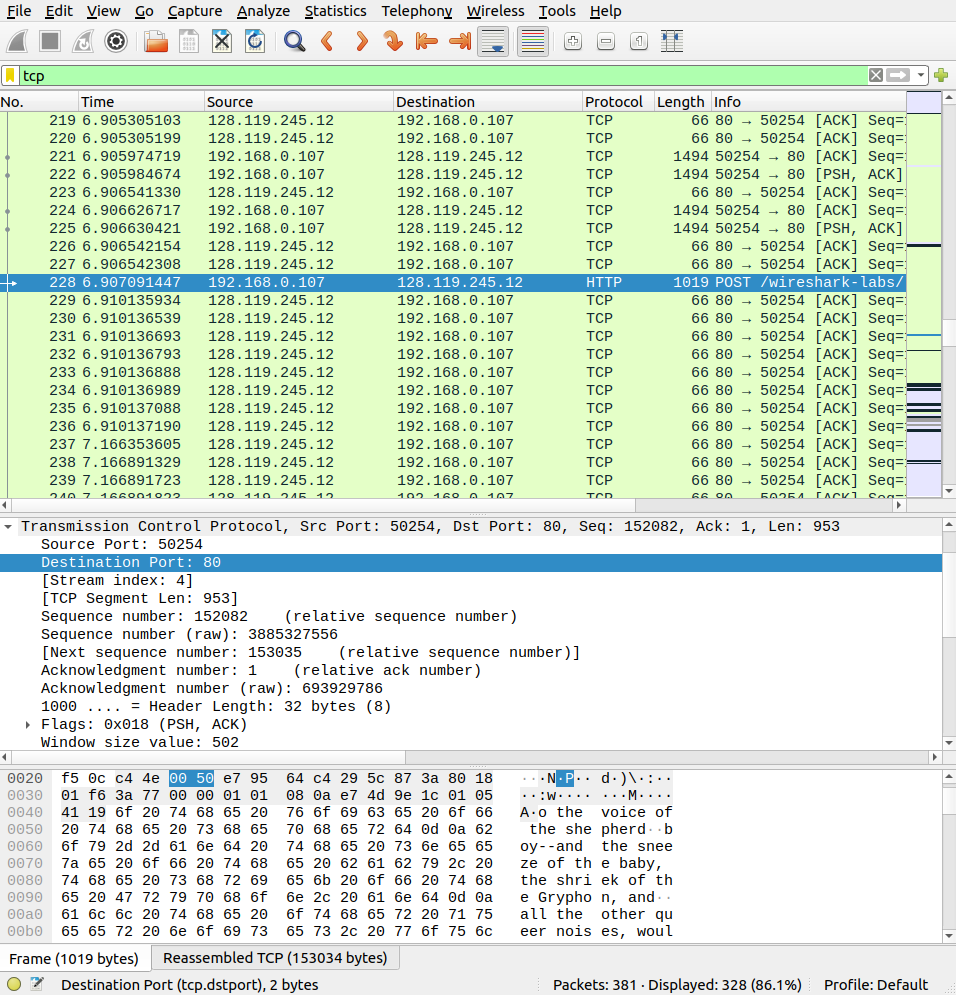
**Student No: 1813810**

1. **What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu?**

Answer:

The IP address: 192.168.0.107

Source Port: 50254



1. **What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?**

Answer:

* The IP address: 128.119.245.12
* Port: 80

1. **What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?**

Answer:

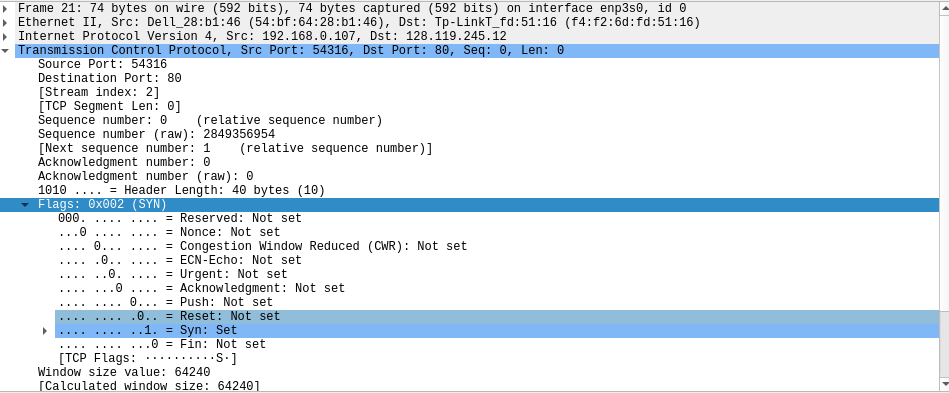
* The IP address: 192.168.0.107
* Source Port: 50254

**TCP Basics**

1. **What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?**

Answer:

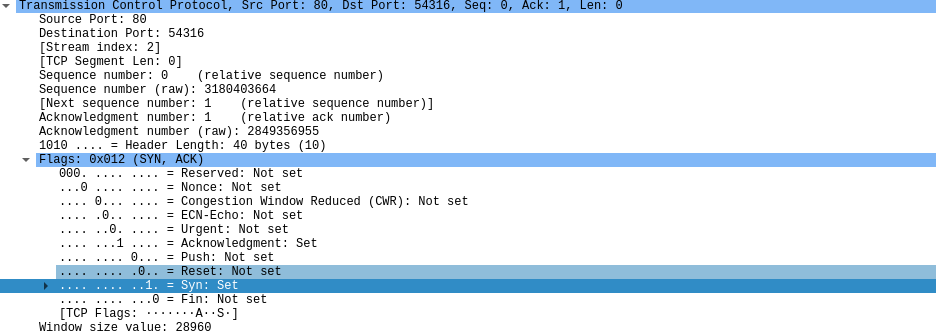
* The sequence number of the TCP SYN segment: 0
* The Flags set to 1, that identifies the segment as a SYN segment.



1. **What is the sequence number of the SYN-ACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?**

Answer:

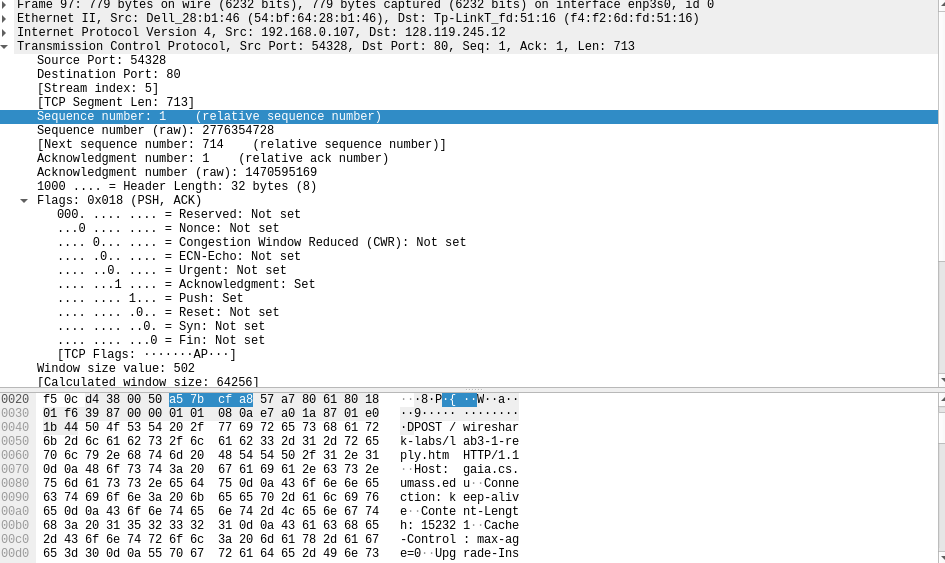
* The sequence number: 0
* The value of the Acknowledgement field: 1
* Server adds 1 to the initial sequence number of the SYN segment from the client computer.
* Syn and Acknowledgment are set to 1



1. **What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you’ll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a “POST” within its DATA field.**

Answer:

The sequence number: 1



1. **Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value (see Section 3.5.3, page 242 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 242 for all subsequent segments.**

Answer:

Sequence number:

Seg 1: 1

Seg 2: 714

Seg 3: 2142

Seg 4: 3570

Seg 5: 4998

Seg 6: 6426

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sent time | Ack received time | RTT |
| Seg 1 | 0.265043000 | 0.651753000 | 0.38671 |
| Seg 2 | 0.265544000 | 1.078294000 | 0.81275 |
| Seg 3 | 0.265544000 | 1.078294000 | 0.81275 |
| Seg 4 | 0.265544000 | 1.078294000 | 0.81275 |
| Seg 5 | 0.265544000 | 1.078294000 | 0.81275 |
| Seg 6 | 0.265544000 | 1.078294000 | 0.81275 |

What is the EstimatedRTT value (see Section 3.5.3, page 242 in text) after the receipt of each ACK?

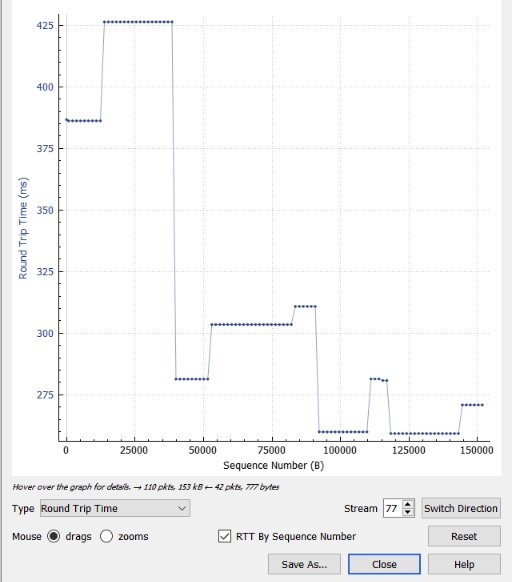
Answer:

EstimatedRTT = 0.875 \* EstimatedRTT + 0.125 \* SampleRTT

EstimatedRTT after the receipt of the ACK of segment 1:  
EstimatedRTT = RTT for Segment 1 = 0.38671

EstimatedRTT after the receipt of the ACK of segment 2:  
EstimatedRTT = 0.875 \* 0.38671 + 0.125 \* 0.81275 = 0.439965

EstimatedRTT after the receipt of the ACK of segment 3:  
EstimatedRTT = 0.875 \* 0.439965 + 0.125 \* 0.81275 = 0.486563125  
EstimatedRTT after the receipt of the ACK of segment 4:  
EstimatedRTT = 0.875 \* 0.486563125 + 0.125 \* 0.81275 = 0.52733648437  
EstimatedRTT after the receipt of the ACK of segment 5:  
EstimatedRTT = 0.875 \* 0.52733648437 + 0.125 \* 0.81275 = 0.56301317382  
EstimatedRTT after the receipt of the ACK of segment 6:  
EstimatedRTT = 0.875 \* 0.56301317382 + 0.125 \* 0.81275 = 0.59423027709



1. **What is the length of each of the first six TCP segments?**

Answer:

Length if each segment:

Seg 1: 713 bytes

Seg 2: 1248 bytes

Seg 3: 1248 bytes

Seg 4: 1248 bytes

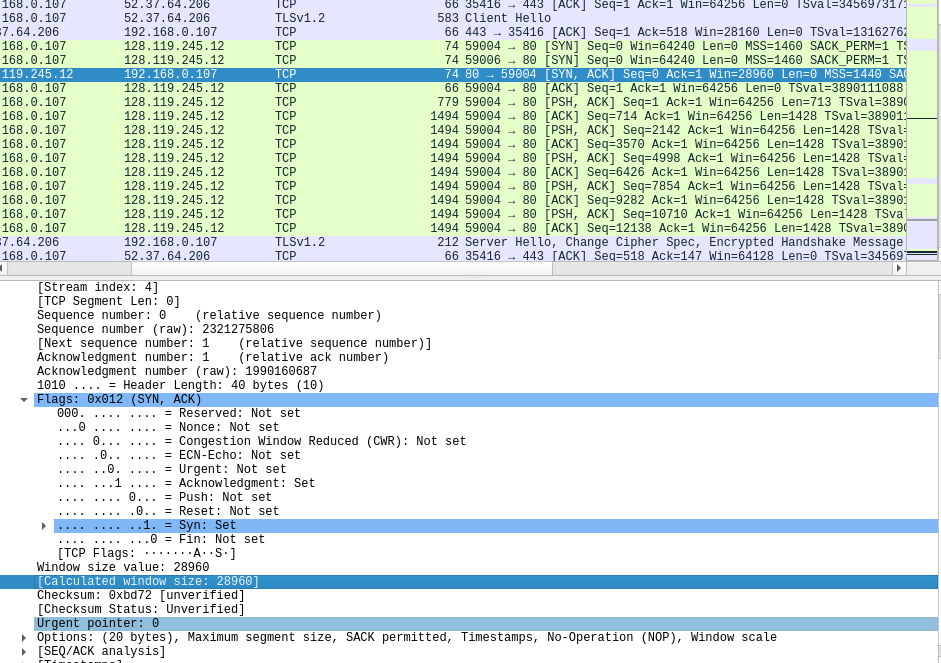
Seg 5: 1248 bytes

Seg 6: 1248 bytes

1. **What is the minimum amount of available buffer space advertised at the receiver for the entire trace? Does the lack of receiver buffer space ever throttle the sender?**

Answer:

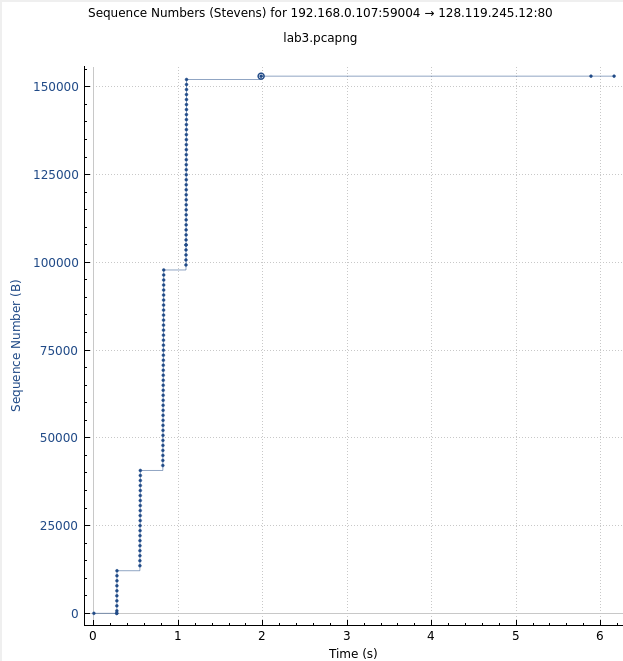
* The minimum amount of available buffer space advertised at the received for the entire trace: 28960 bytes
* Never throttled due to lacking of receiver buffer space



1. **Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?**

Answer:

There are no retransmitted segments in the trace.



1. **How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 250 in the text).**

Answer: The receiver usually acknowledges 1248 bytes in an ack. When the segment is ACKing every other if the data is doubled.

1. **What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.**

Answer:

Kích thước của file alice.txt: 152,138 bytes

Time incurred: 1.382188000 - 0.651753000 = 0.730435

Throughput = 152,138/0.730435 = 208.284 kbytes/s