Maxwell Sullivan: Computer Science
This blog was created to submit assignments for CSC251

Wireshark Lab 4: Exploring TCP

March 11, 2013
PART 1: Capturing a bulk TCP transfer from your computer to a remote server

Lab Video:

STEPS:

1. Start up your web browser. Go the http://gaia.cs.umass.edu/wiresharklabs/alice.txt (http://gaia.cs.umass.edu/wiresharklabs/alice.txt) and retrieve an ASCII copy of Alice in Wonderland. Store this file somewhere on your computer.

2. Next go to http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html (http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html).

3. Use the Browse button in this form to enter the name of the file (full path name) on your computer containing Alice in Wonderland (or do so manually). Don't yet press the "Upload alice.txt file" button.

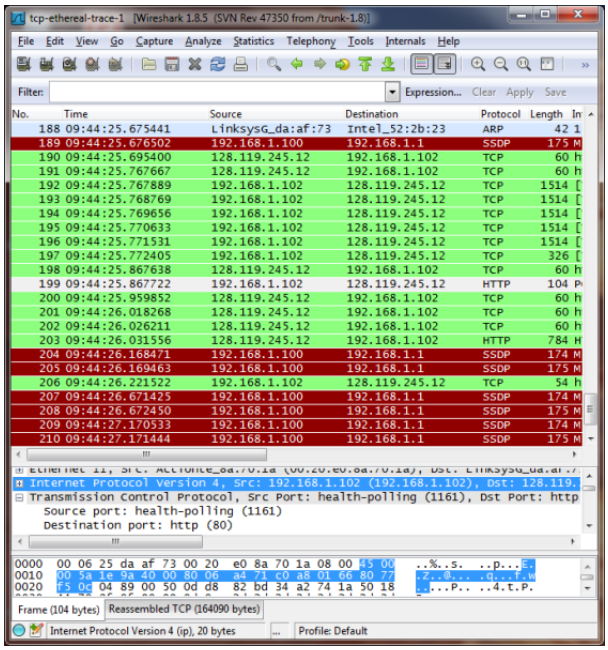
4. Now start up Wireshark and begin packet capture (Capture->Start) and then press OK on the Wireshark Packet Capture Options screen (we'll not need to select any options here).

5. Returning to your browser, press the "Upload alice.txt file" button to upload the file to the gaia.cs.umass.edu server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.

6. Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.

PART 2: A first Look At the Captured Trace

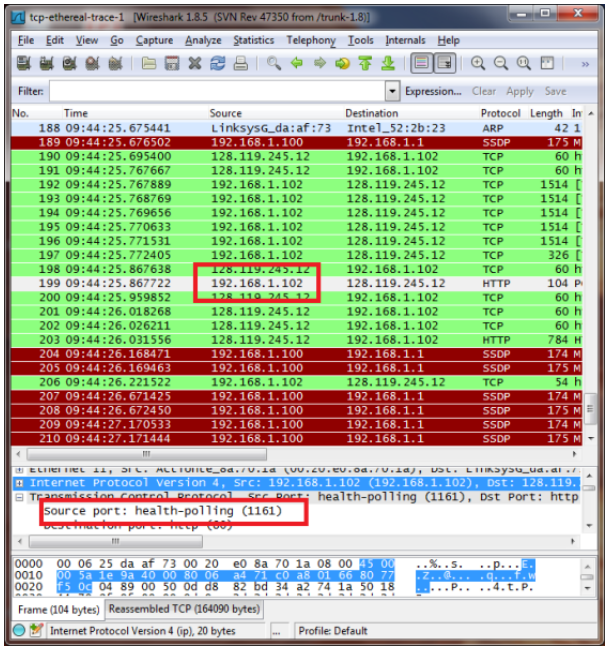
Use the online capture (shown below) to answer the following:



(https://maxwellsullivan.files.wordpress.com/2013/03/4-0.png)

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? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows.

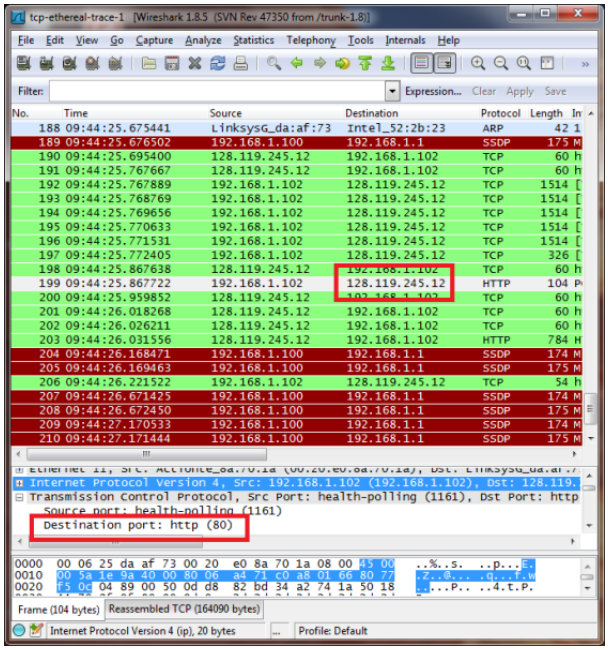
The source IP address was 192.168.102 using source port 1161.



(https://maxwellsullivan.files.wordpress.com/2013/03/4-1.png)

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

The destination IP address is 128.119.245.12 receiving on port 80

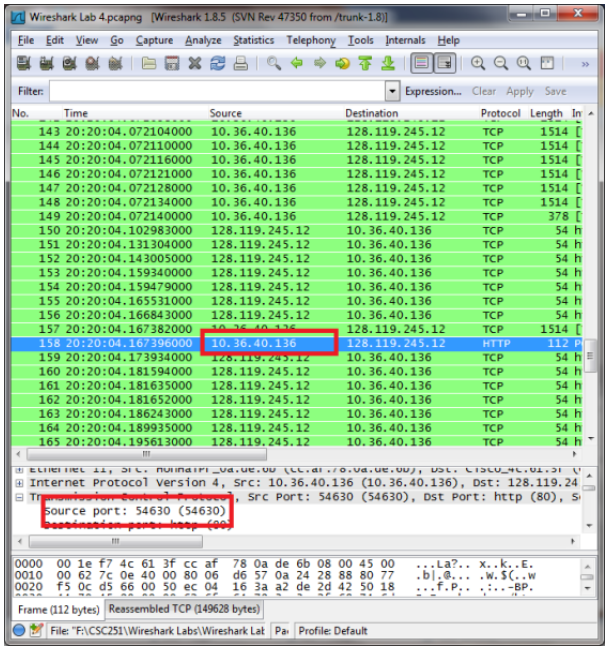


(https://maxwellsullivan.files.wordpress.com/2013/03/4-2.png)

Use your own Capture to answer the following:

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

My IP address source is 10.36.40.136 sending on port 54360.

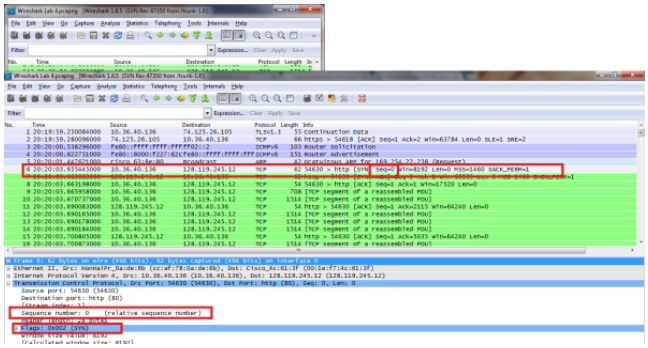


(https://maxwellsullivan.files.wordpress.com/2013/03/4-3.png)

PART 3: TCP Basics

4. 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?

> The sequence number of the segment used to initiate the TCP connection is 0. We can see that the message contains a SYN flag indicating that it is a SYN segment.

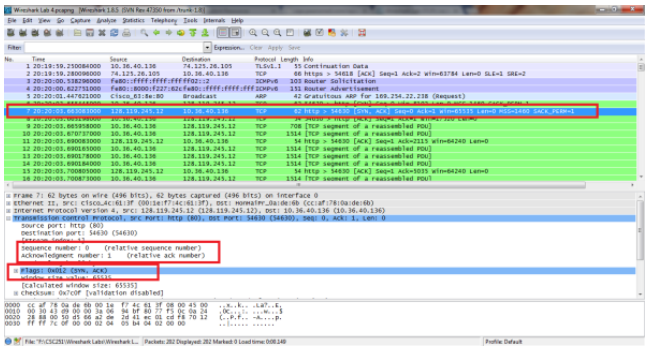


(https://maxwellsullivan.files.wordpress.com/2013/03/4-4.png)

5. What is the sequence number of the SYNACK 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?

> The sequence number of the SYNACK segment is 0.
> The value of the acknowledgement field is 1. This value is determined by the initial sequence number +1.

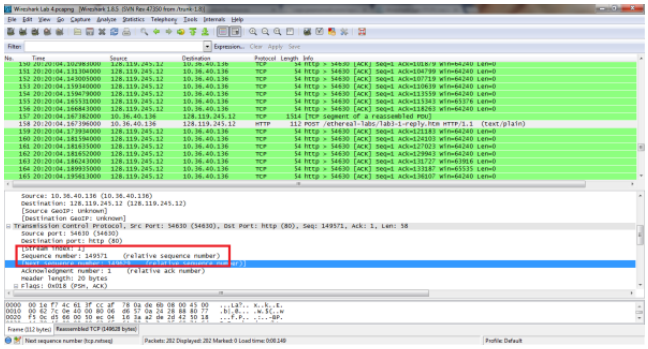
The message carries flags that show it to be a SYN ACK message.



(https://maxwellsullivan.files.wordpress.com/2013/03/4-5.png)

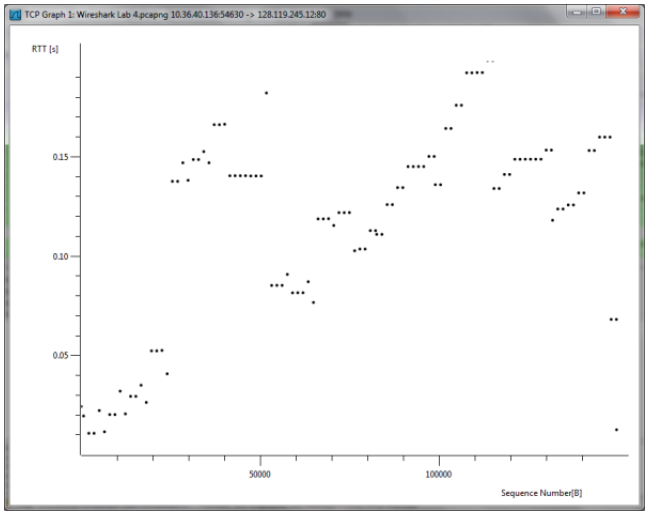
6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

The sequence number of the TCP segment containing the HTTP Post Command is 149571.



(https://maxwellsullivan.files.wordpress.com/2013/03/4-6.png)

7. 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 239 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 239 for all subsequent segments.



(https://maxwellsullivan.files.wordpress.com/2013/03/4-7.png)

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

The length of each of the first TCP segment is 708. The following segments are all 1514.

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

The minimum amount of available buffer space is listed as 65535. The sender is never throttled because we never reach full capacity of the window.

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

No, no segments were ever retransmitted. This is shown by the fact that an old Acknowledgement number was never resent in order to re-request former packets.

11. 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 247 in the text).

The receiver is typically acking 432 bits. There are cases where the receiver acks every other segment. This is shown when more than one ack occurs in a row.

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

The throughput can be calculated by using the value of the last ack(149,629)- the first sequence number(1) divided by the time since first frame (1.6) = 93517.6 bps.

PART 4: TCP Congestion Control In Action

STEPS:

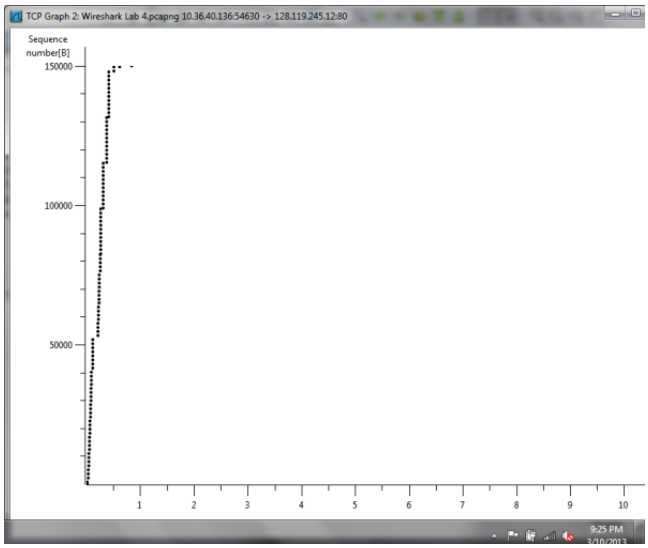
1. Select a TCP segment in the Wireshark's "listing of captured-packets" window. Then select the menu : Statistics->TCP Stream Graph-> Time-SequenceGraph(Stevens).

QUESTIONS:

Answer Question 13 Using the provided Capture

13. Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.

The TCP slowstart phase begins at just above seq number 5000, and ends just before sequence number 10000. Congestion avoidance takes over at 10000.



(https://maxwellsullivan.files.wordpress.com/2013/03/4-13.png)

From → Wireshark Labs

Leave a Comment