

**Experimental Report**

|  |  |
| --- | --- |
| **Course**: | Computer Networking |
| **Semester**: | 2nd semester of the academic year **2020-2021** |
| **Major**: | Software Engineering |
| **Class**: | 2019 |
| **Student Name**: | 吴嘉诚 |
| **Student ID:** | 222019321062111 |
| **Teacher:** | Zheng Yang (杨铮) |

**School of Computer and Information Science**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | | TCP\_UDP | | | |
| Date | |  | Type | | □Confirmatory  √ Design  √ Comprehensive |
| 1. **Objectives & Requirements** | | | | | |
| 1. **Experimental environment (**platform and software**)** | | | | | |
| 1. **Experimental content and design** (Main Content, Procedure, Codes and Results)      1. A first look at the captured trace 2. 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.   **It is 192.168.1.102:1161**     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?   **128.119.245.12:80**    If you have been able to create your own trace, answer the following question:   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?   **It is 192.168.43.202:52098**     1. TCP Basics 2. 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?   **It is 232129012**    **The SYN bit is 1 and ACK is 0 in the Flags that identifies it is a SYN segment**     1. 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?   **It is 883061785;**  **ACK field is 232129013;**  **This value is get from previous SYN segment’s seq-number and add 1;**  **Still flags: ACK is set 1, SYN is set 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.   **It is 232129013**     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.   *Note:* Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the “listing of captured packets” window that is being sent from the client to the gaia.cs.umass.edu server. Then select: *Statistics->TCP Stream Graph->Round Trip Time Graph.*  *(***Consider the relative seq-number would be easy to understand)**  **It is: 1---556---2026---3486---4946---6406;**  **The first 2 segment sent, then 556ACK received, then 2026 and 3486 sent, then 2026ACK received, then 4946 and 6406 sent;**  **2021-05-23 14-51-23 的屏幕截图**  **20190708105533320**   1. What is the length of each of the first six TCP segments?[[1]](#footnote-0)   **565,1460,1460,1460,1460,1460**   1. 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?   **5840;**  **The receiver’s window size grows steadily til max window size(62780) comes, and no throttle is made due to the lack of 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?   **NO;**  **I checked the seq-number of the segments sent by the source, to see if the seq-number keeps growing.**  **2021-05-23 14-52-03 的屏幕截图**   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).   **In this case, it is 1460bytes**    **Yes, it cumulatively ACK sometimes.**   1. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.   **164090byte / 5.461175s = 30046.6475B/s = 30kB/s**     1. TCP congestion control in action 2. 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.   **0~0.3s is *slow start;***  **Congestion avoidance start at 0.3s;**  **Diff: 2 segments was sent at 1.8s rather than 3segments(previous is 2)**     1. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu   **For my own trace:**    **2021-05-23 15-43-54 的屏幕截图**  **2021-05-23 15-44-17 的屏幕截图**   1. UDP lab   1. Select one UDP packet from your trace. From this packet, determine how many  fields there are in the UDP header. (You shouldn’t look in the textbook! Answer  these questions directly from what you observe in the packet trace.) Name these  fields.  2. By consulting the displayed information in Wireshark’s packet content field for  this packet, determine the length (in bytes) of each of the UDP header fields.  3. The value in the Length field is the length of what? (You can consult the text for  this answer). Verify your claim with your captured UDP packet.  4. What is the maximum number of bytes that can be included in a UDP payload?  (Hint: the answer to this question can be determined by your answer to 2. above)  5. What is the largest possible source port number? (Hint: see the hint in 4.)  6. What is the protocol number for UDP? Give your answer in both hexadecimal and  decimal notation. To answer this question, you’ll need to look into the Protocol  field of the IP datagram containing this UDP segment (see Figure 4.13 in the text,  and the discussion of IP header fields).  7. Examine a pair of UDP packets in which your host sends the first UDP packet and  the second UDP packet is a reply to this first UDP packet. (Hint: for a second  packet to be sent in response to a first packet, the sender of the first packet should  be the destination of the second packet). Describe the relationship between the  port numbers in the two packets. | | | | | |
| 1. **Analysis and discussion of experiment results**（Analysis of experimental results, and summary of gains and the existing problems） | | | | | |
| Comments & Evaluation | Content & Design (A-E) | | |  | |
| Procedure & Codes (A-E) | | |  | |
| Results (A-E) | | |  | |
| Analysis & Discussion (A-E) | | |  | |
| Score (A-E):  Feedback comments: | | | | |

1. [↑](#footnote-ref-0)