

**Lab report**

|  |  |
| --- | --- |
| **Course**: | Computer Networking |
| **Semester**: | 2nd semester of the academic year **2023-2024** |
| **Major**: | Software Engineering |
| **Class**: | 2022 |
| **Student Name**: | 吴孜远 |
| **Student ID:** | 222022321062009 |
| **Teacher:** | Chunming Wu (吴春明) |

**School of Computer and Information Science**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | |  | | | |
| Date | | 2024.5.1 | Type | | □Confirmatory  √ Design  √ Comprehensive |
| 1. **Objective & Requirements** | | | | | |
| 1. **Experimental environment (**platform and software**)** | | | | | |
| 1. **Experimental content and design** (Main Content, Procedure, Codes and Results)   **实验一：HTTP**   1. The Basic HTTP GET/response interaction      1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?     HTTP版本是1.1   1. What languages (if any) does your browser indicate that it can accept to the server?      1. What is the IP address of your computer? Of the gaia.cs.umass.edu server?     可以看到这是GET请求，由本机发送到gaia.cs.umass.edu server，所以可以看到本机ip为10.135.254.113，gaia.cs.umass.edu的ip为128.119.245.12   1. What is the status code returned from the server to your browser?     返回码为200，表示成功   1. When was the HTML file that you are retrieving last modified at the server?      1. How many bytes of content are being returned to your browser?      1. By inspecting the raw data in the packet content window, do you see any headers within the data that are not displayed in the packet-listing window? If so, name one.   比如Accept-Encoding和Accept-Language就不会在packet-listing window中显示，只能在packet content window中查看   1. The HTTP CONDITIONAL GET/response interaction 2. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?   没有   1. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?   首先可以看到返回码为200，表示成功返回文件    还可以看到     1. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE:” line in the HTTP GET? If so, what information follows the “IF-MODIFIED-SINCE:” header?     这是之前从服务器收到的最后修改日期   1. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.     返回码是304，response phrase is Not Modified。  服务器没有返回文件内容，在packet content中甚至没有content-type等header，因为这时服务器认为客户端拥有的版本仍然是最新的，就直接读取缓存。相反，如果文件已经变化，服务器将返回200 OK，并返回着更新后的文件内容   1. Retrieving Long Documents        1. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill or Rights?   如上图所示，只发送了一条HTTP GET请求。   1. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?   紧接着第二个HTTP数据包就是相应，可以看到返回200 OK。   1. What is the status code and phrase in the response?   200 OK     1. How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?   三次握手建立连接之后，分3次发送了整个text文件，从序列号的增加和length上就可以判断：每次的length为1460     1. HTML Documents with Embedded Objects      1. How many HTTP GET request messages did your browser send? To which Internet addresses were these GET requests sent?   一共发送了6个HTTP GET请求，域名如下：  其中包括请求js、图片、html     1. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.     可以看到对两个图片的请求几乎同时被发出，是连续的两个包，所以浏览器应该是并行地下载两个图片   1. HTTP Authentication      1. What is the server’s response (status code and phrase) in response to the initial HTTP GET message from your browser?   我这里的第一次GET没有直接返回401，而是紧接着请求了认证表单等js文件。  所以题中意思应该是我截图中的第二个请求，返回401 Unauthorized，表明在这次请求之后，浏览器需要添加认证信息，比如用户名和密码，才能成功获取资源。     1. When your browser’s sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?   新增了认证信息    **实验二：TCP**   1. Capturing a bulk TCP transfer from your computer to a remote server     2. A first look at the captured trace   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?   源ip是10.135.254.113，所使用的tcp端口号为23321     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?   Ip为128.119.245.12，端口使用的是默认端口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?   源ip是10.135.254.113，所使用的tcp端口号为23321    3. 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?     我们可以从flag中找到tcp段，可以看到syn位置1，     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?   “Sequence Number”字段显示服务器的相对初始序列号，便于查看。同时也显示了原始序列号。  “Acknowledgment Number”字段显示对客户端SYN段序列号的确认值，  Acknowledgment Number的值是客户端最初发送的SYN段中的Sequence Number加1  这里就是0+1=1，或者看raw原始序列号，也是65417767+1=65417768    这个SYNACK段的TCP标志位会显示SYN和ACK都是set的，表示这是一个同时设置同步和确认的响应。也就是第二次握手     1. What is the sequence number of the TCP segment containing the HTTP POST command?   这个报文段就在三次握手完成之后    因为从data中能找到post字样    序列号如下：     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.）   **EstimatedRTT = (1 - a) × EstimatedRTT + a × SampleRTT（a 使用0.125）**  考虑这6个段：    由于我发现服务器并非对每个数据包都做ACK响应，所以那些包看不到RTT，所以只能挑选服务器响应的前六个包来计算和分析了：  服务器响应包如下：  对应到frame number查看RTT    **报文段1：frame 62**    序列号：    收到第一个ACK后的响应：在frame 75    **EstimatedRTT=0.30802**  报文段2：frame 68    序列号：    收到第一个ACK后的响应：在frame 75：    **EstimatedRTT=0.875∗0.30802+0.125∗0.29755=0.30671125**  报文段3：frame 72    序列号：    收到第一个ACK后的响应：在frame 75：    **EstimatedRTT=0.875∗0.30671+0.125∗0.29755=0.30556**  报文段4：frame 80    序列号：    收到第一个ACK后的响应：在frame 98：    **EstimatedRTT=0.875∗0.30556+0.125∗0.30664=0.30569**  报文段5：frame 83    序列号：    收到第一个ACK后的响应：在frame 99：    **EstimatedRTT=0.875∗0.30569+0.125∗0.30664=0.30580**  报文段6：frame 85    序列号：    收到第一个ACK后的响应：在frame 100：    **EstimatedRTT=0.875∗0.30580+0.125∗0.30664=0.30590**     1. What is the length of each of the first six TCP segments?[[1]](#footnote-1)   第一个TCP段由于是HTTP POST，所以整个frame大小为761：    TCP data只有707：    其它的五个frame大小均为1254：    TCP data为1200：     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?   最小是29200，是第二次握手    因为 TCP 的流量控制服务会根据接收方的缓冲区空间大小来限制发送方发送TCP段的数量和速率，所以缺少接收器缓冲区空间会限制发送方传送 TCP 区段   1. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?   有重传现象，在packet list中就能看出：    这里有一些包没有收到ACK响应，所以重传，提示TCP Spurious Retransmission。  或者可以看序列号-时间图：    可以看出中间有一段下落，那就是重传的时间   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).     按理来说ACK应该一次确认1200B的数据，但在我的抓包中可以看到，它一次会确认多个TCP包的数据，数量不定，比如上图中红框的第二个ACK确认了6000B，也就是5个TCP段。   1. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.     既然是每单位时间传输的字节数，那就使用总传输量/用时  最后的序列号为153056：    那么总传输量为153056-1 = 153055  总时间由于有重传发生，所以用时比较长：10.56-4.04 = 6.14  吞吐量=153055-6.14=24927.52  4. TCP congestion control in action   1. 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.   ​​​​​慢启动是指序列号在TCP包收到ACK响应确认时都会增长，增加的大小就是已确认段的数目，就是指数增长。可以看到我们中间有一个地方突然下降，是因为前面有部分分组没有收到ACK，所以重传。由于我们的接收缓存的大小是130000左右，一直没有发生拥塞，所以没有进入拥塞控制或者快速恢复。如果接收缓存小一点，文件大一点，就能看到拥塞控制和快速恢复。 | | | | | |
| 1. **Result analysis and discussion**（Analysis of experimental results and summing up the harvest 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-1)