**《计算机网络》Wireshark实验**





**Wireshark-TCP**

**学 号 3020205015**

**姓 名 石云天**

**学 院 未来技术学院**

**专 业 计算机科学与技术**

**年 级 2020**

**任课教师 周晓波**

**2023 年 4 月 20日**

# 一、报告摘要

本次实验通过抓包分析的方式，对TCP/IP协议进行深入学习与实践。主要包括：获取客户端和服务器的IP地址及端口号，识别TCP SYN段，分析TCP连接建立过程，找到TCP段中的HTTP POST命令，并分析该TCP连接中前六条segment的发送时间、接收时间、RTT与EstimatedRTT等。在本次实验过程中，我掌握了Wireshark工具的使用，以及分析TCP/IP协议的基本方法与技巧。通过本次实验的实践与学习，我们深刻理解了TCP/IP协议的基本原理与运作体制，对网络通信有了更深层次的认识。

# 二、任务要求

本次实验要求捕获HTTP连接的数据包，并回答以下问题：

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”

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?

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?

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?

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?

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.

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?

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

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?

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

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 250 in the text).

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

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.

14.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

# 三、实验结果与分析

首先根据实验要求捕获分组情况如下图1：

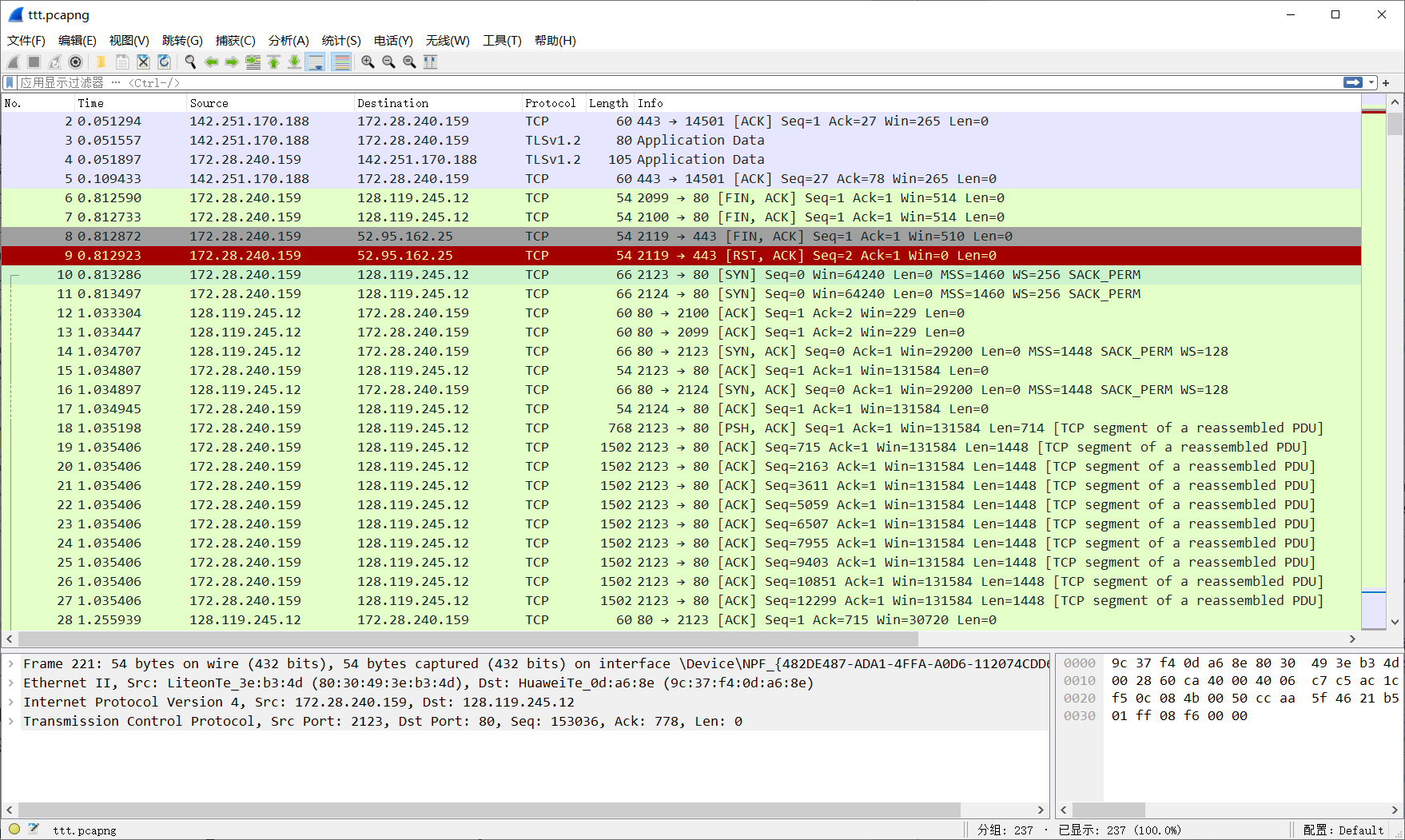


图1 分组情况

**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”**

根据HTTP头部信息可以查询客户端与服务器的IP地址及端口号，将HTTP头部信息截图如下图2（端口号已用红色标注）：

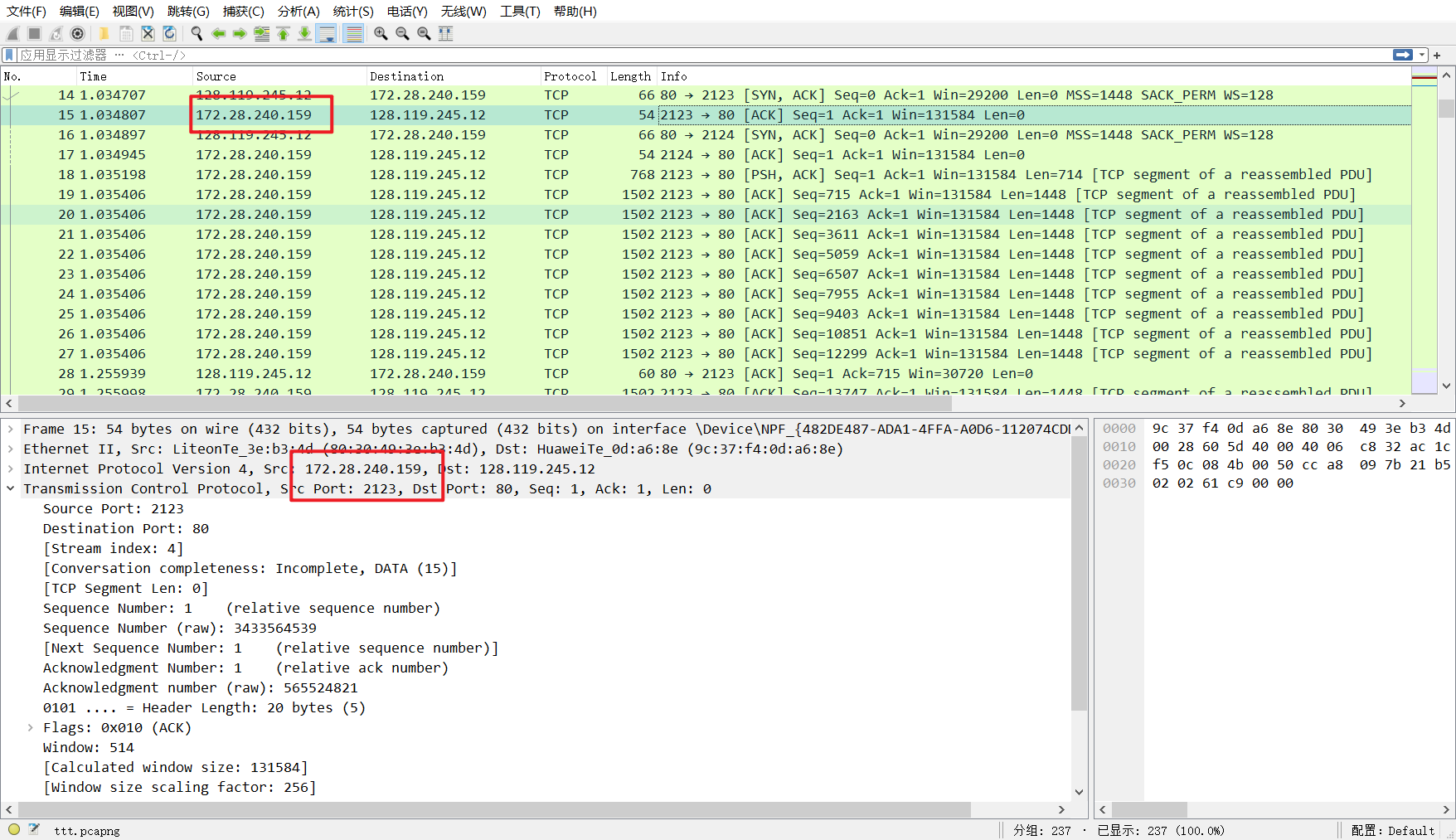


图2 头部信息

由图可知，源IP：172.28.240.159；源端口号：2123。

**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?**

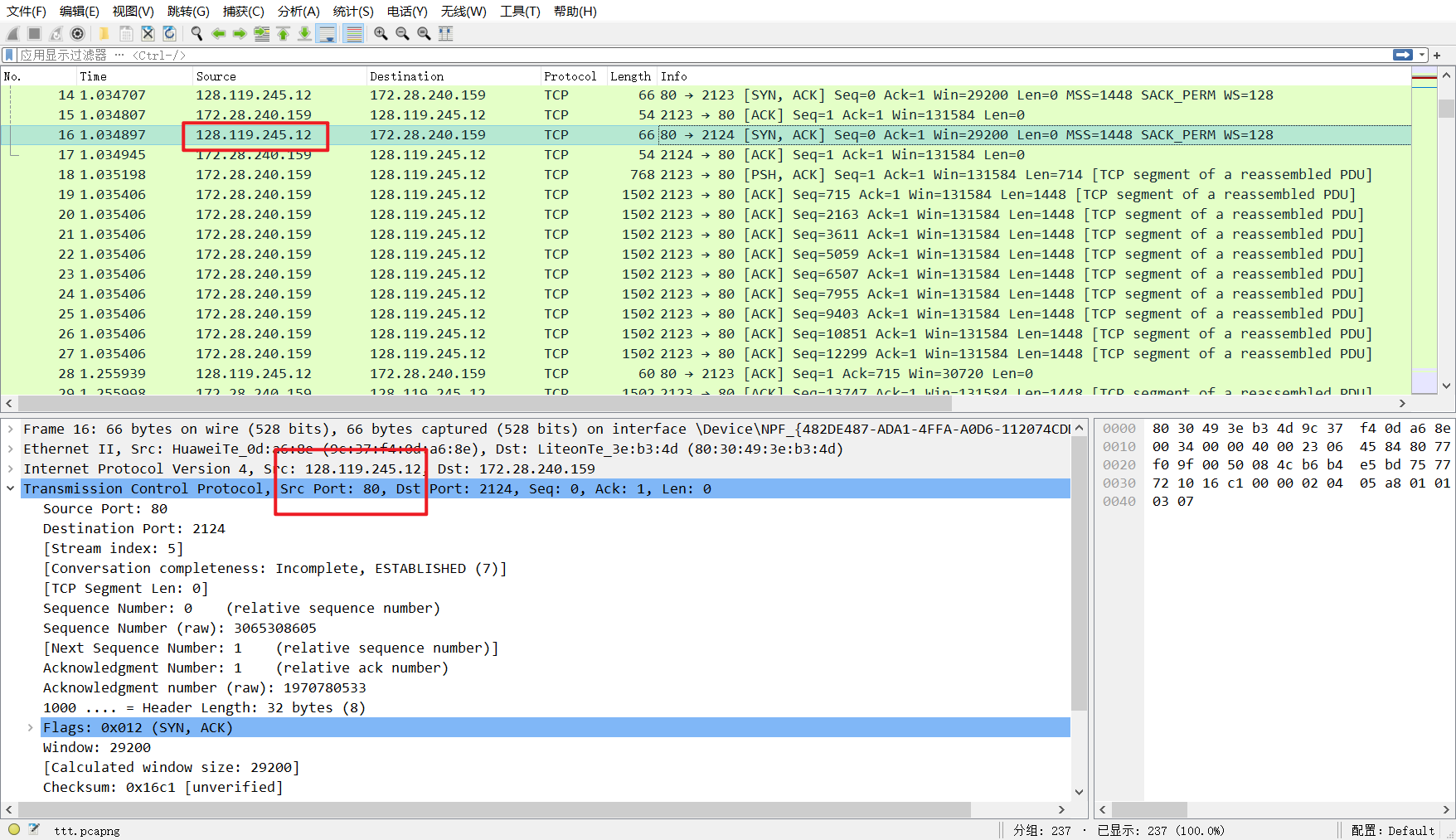


图3 头部信息

由图可知，目的IP：128.119.245.12；目的端口号：80。

**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?**

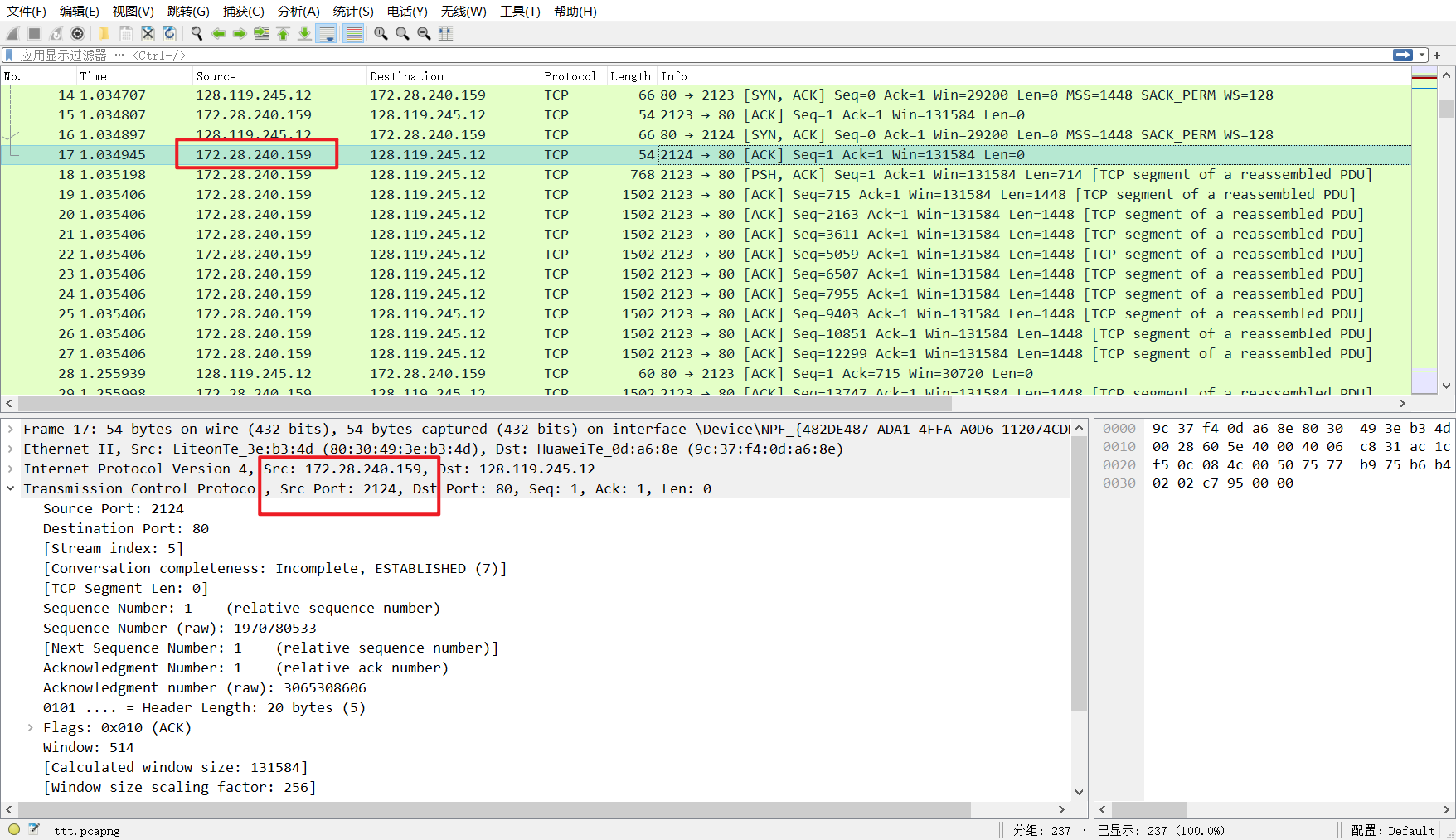
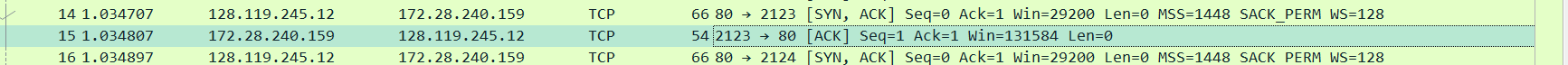


图4 头部信息

由图可知，源IP：172.28.240.159；源端口号：2123。

**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?**



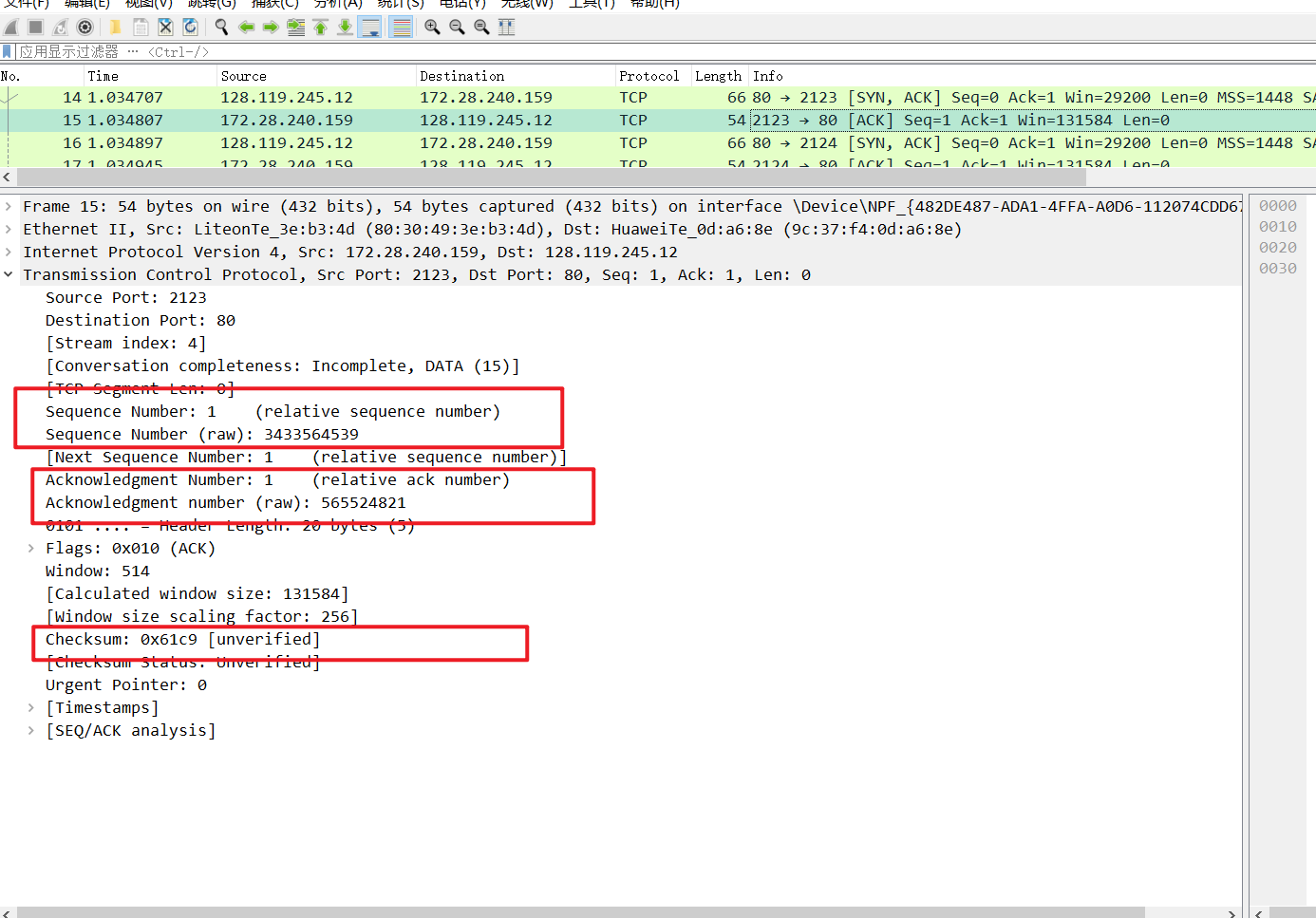


图5 报文信息

客户端向服务器发送序列号为0。序列号为0且ACK为0这一特征说明该segment用于建立TCP连接的第一条SYN。

**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?**

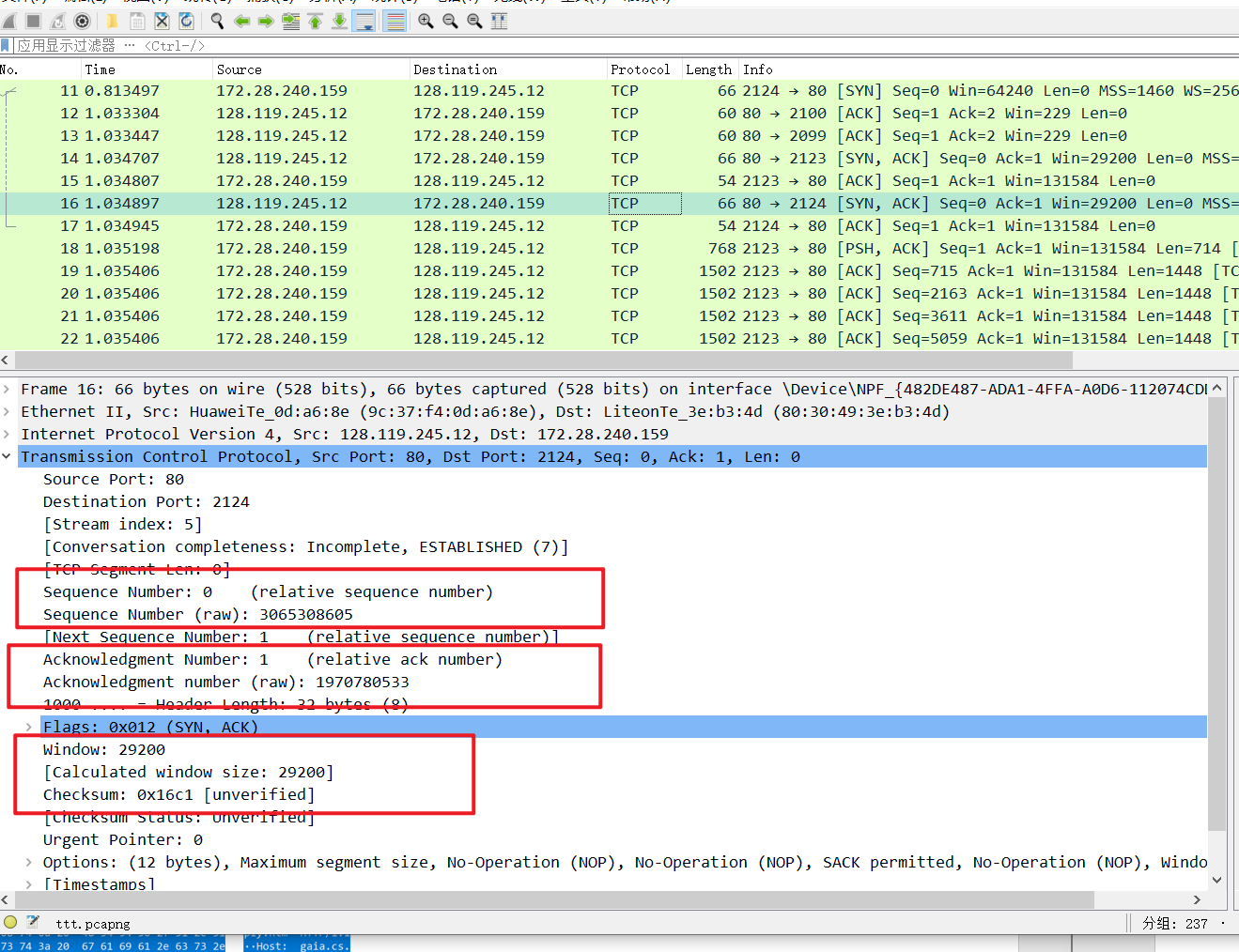


图6 报文信息

服务器向客户端返回的序列号仍为0，ACK为1。通过校验和来确定是否正确，序列号为0且ACK为0这一特征说明该segment是用于建立TCP连接的第一条SYN，序列号为0且ACK为1这一特征说明该segment是用于建立TCP连接的SYNACK。

**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.**

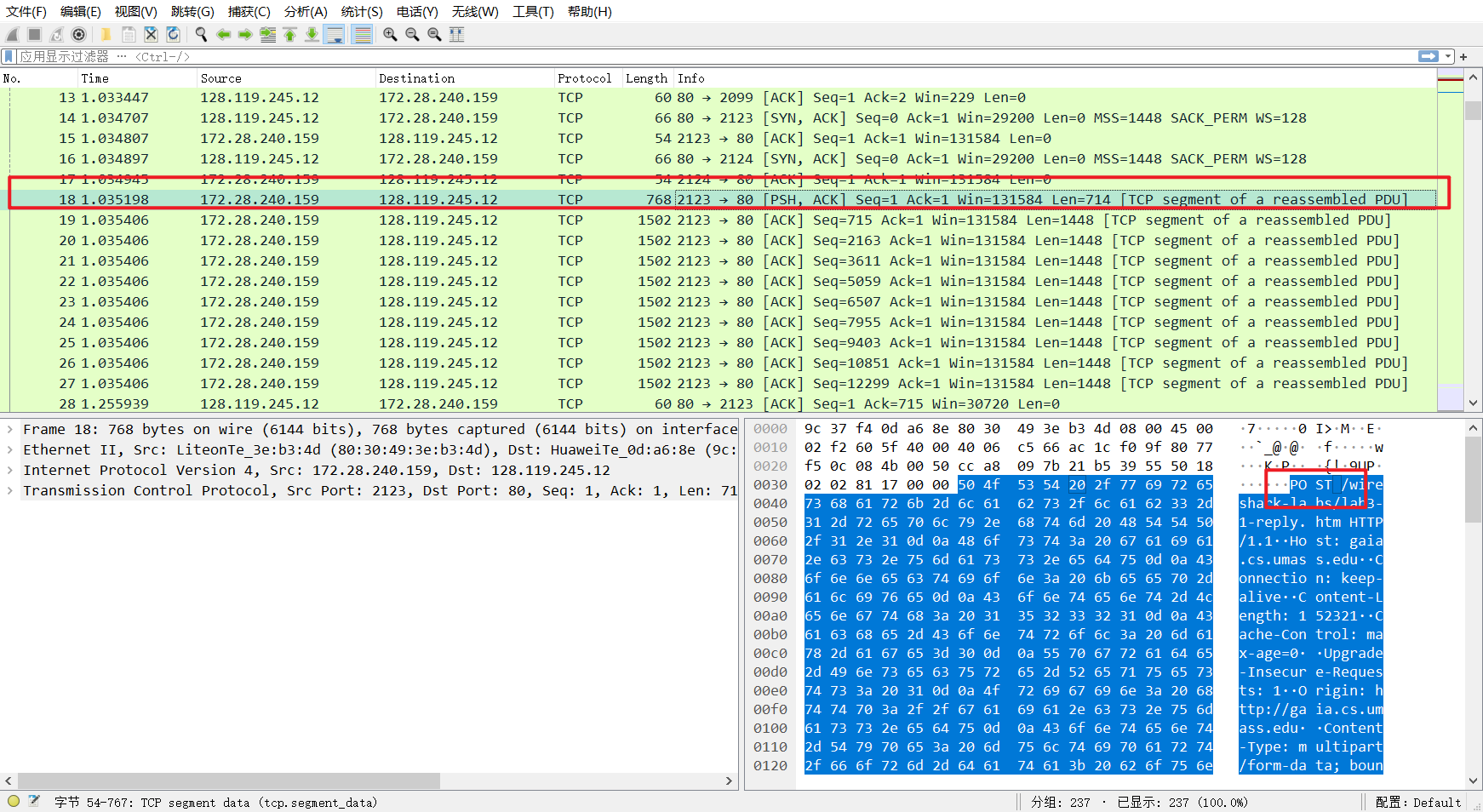


图7 报文信息

由上图知，包含HTTP POST命令的TCP段的序列号为1。

**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?**

对于POST开始的前六条segment，可列表如下：

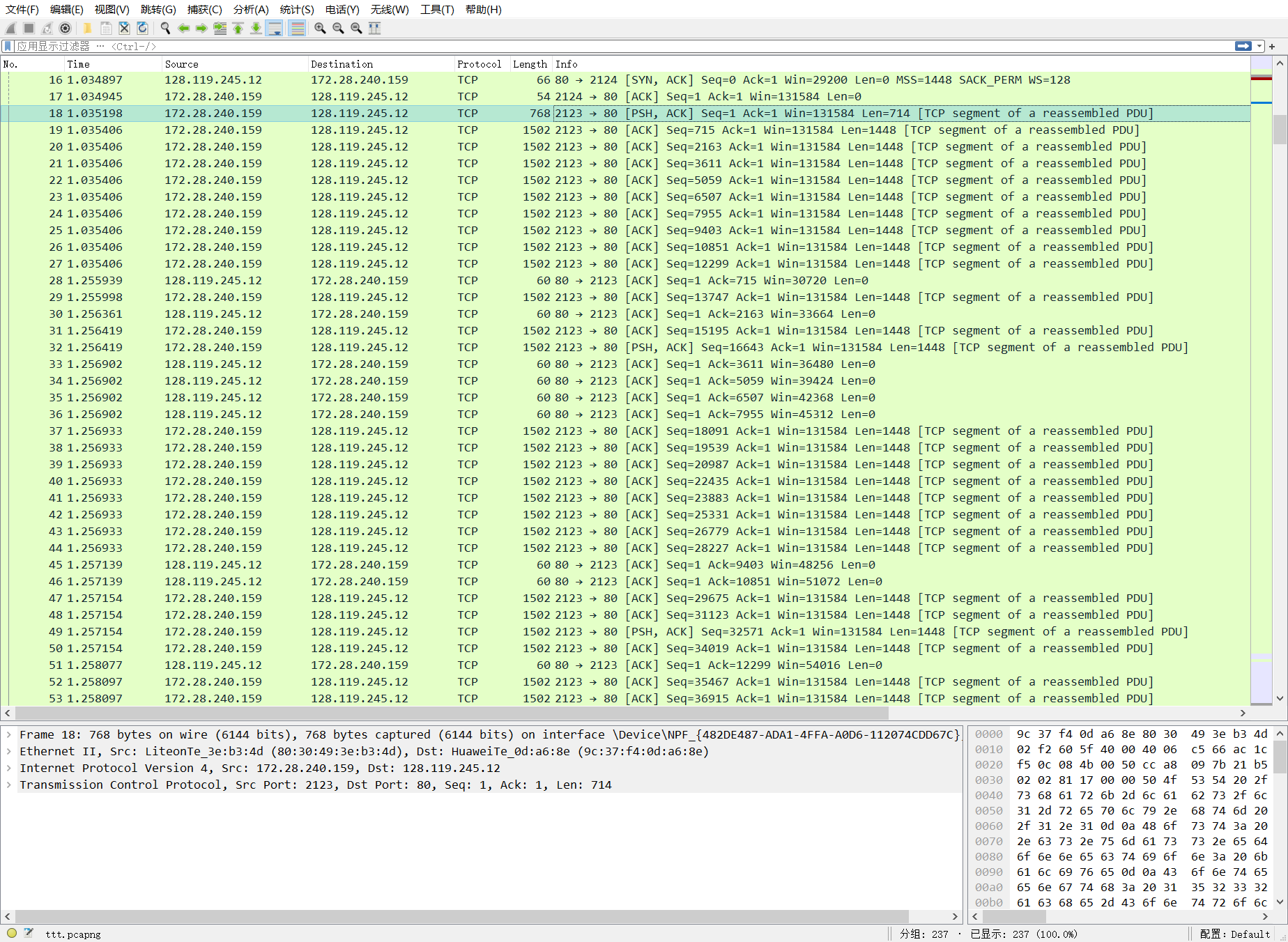


图8 前六条信息

由公式知：

取，结合测量出的RTT值，计算EstimatedRTT，结果如下表：

表1 计算结果

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数量 | 发送时间 | 接收ACK时间 | RTT | EstimatedRTT | length |
| 1 | 1.035198 | 1.255939 | 0.220741 | 0.220741 | 714 |
| 2 | 1.035406 | 1.256361 | 0.220955 | 0.220741 | 1448 |
| 3 | 1.255998 | 1.256902 | 0.000904 | 0.22076775 | 1148 |
| 4 | 1.256419 | 1.257139 | 0.00072 | 0.193284781 | 1148 |
| 5 | 1.256933 | 1.258077 | 0.001144 | 0.169214184 | 1148 |
| 6 | 1.257154 | 1.258253 | 0.001099 | 0.148205411 | 1148 |

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

TCP段长度信息通过下图length列可知：

表2 TCP长度

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| 长度 | 714 | 1448 | 1448 | 1448 | 1448 | 1448 |

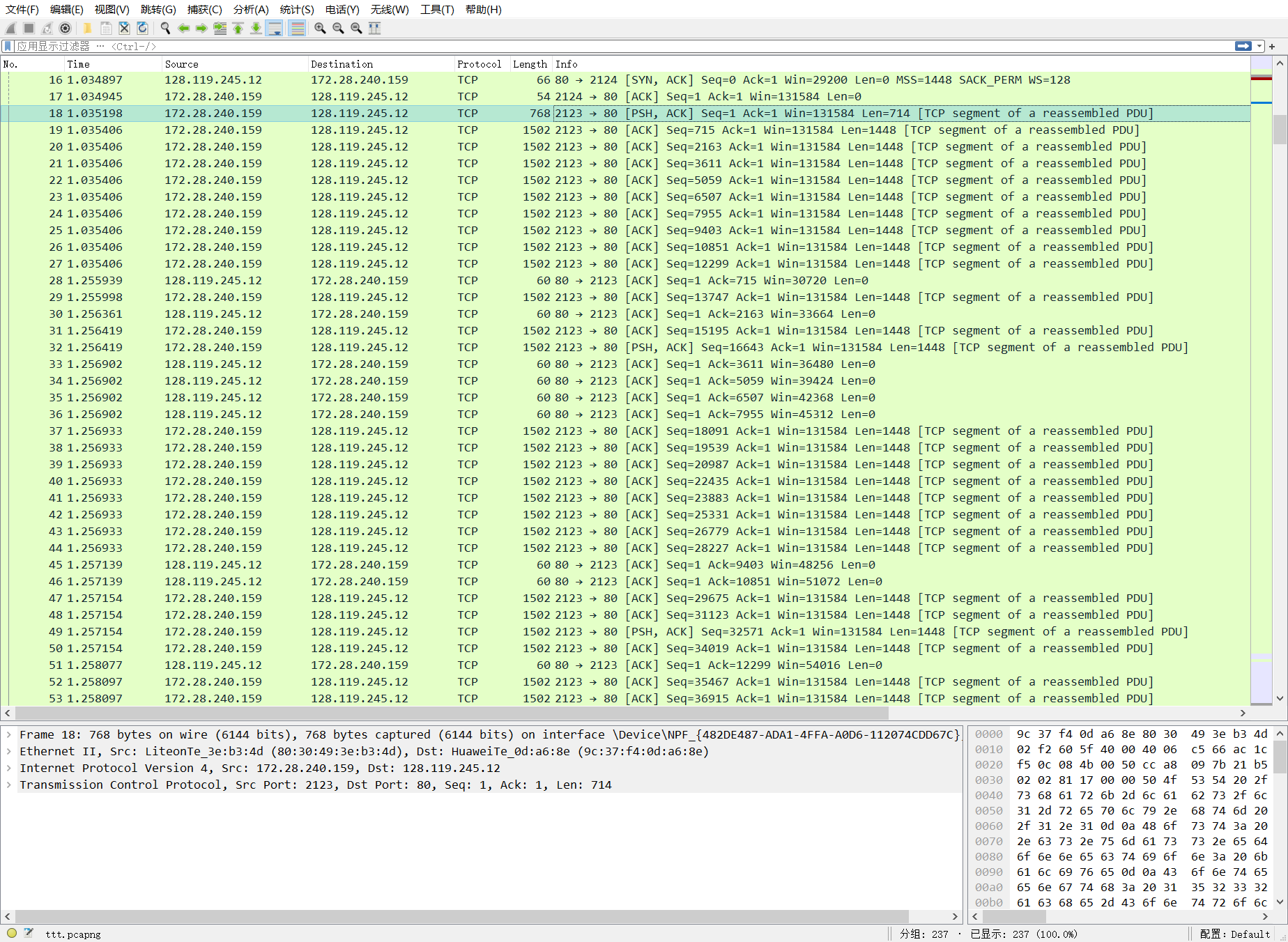


图9 TCP长度

**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?**

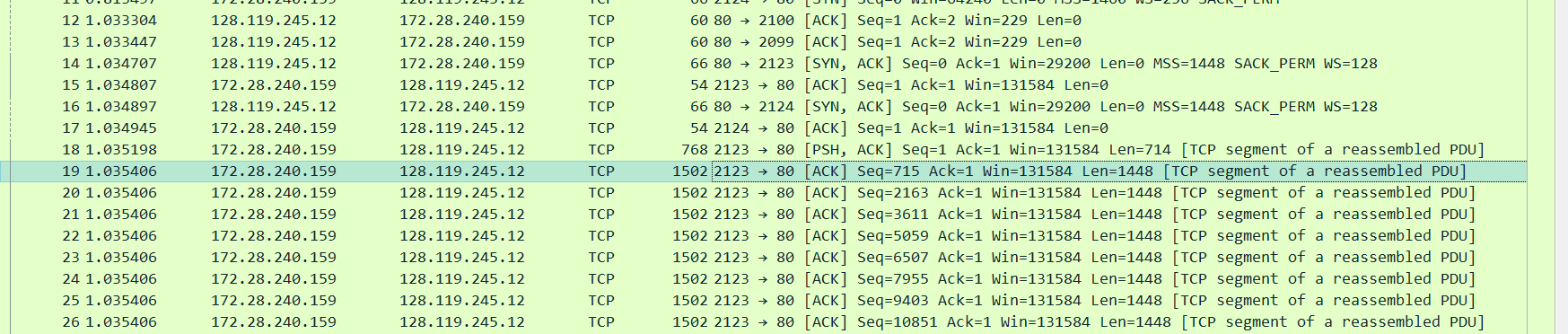


图10 缓冲区长度

接收方可用缓冲区最小空间为131584，接收区窗口大小稳步增长，直到出现最大窗口大小。由于缺乏缓冲空间，因此没有进行限制。

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

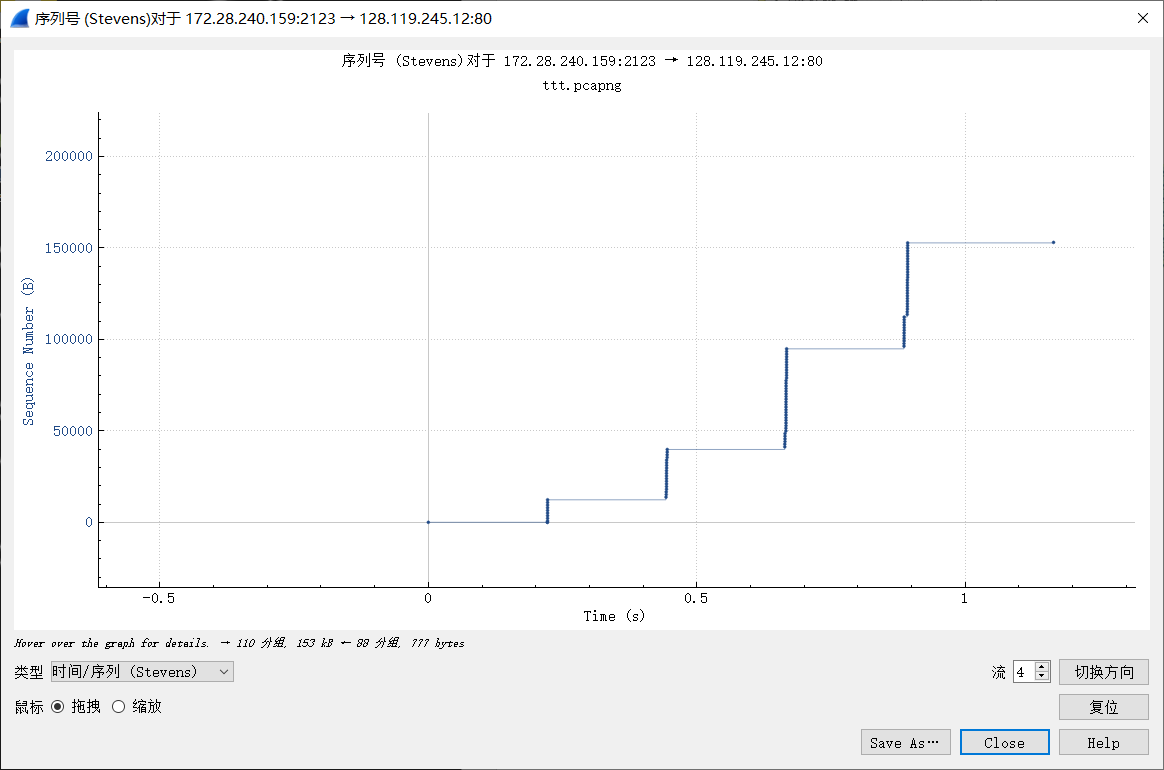


图11 传输图

本次trace file中未出现重传，通过检查发送端发送的报文段序号，发现序号持续增加，因此未出现重传。

**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 250 in the text).**

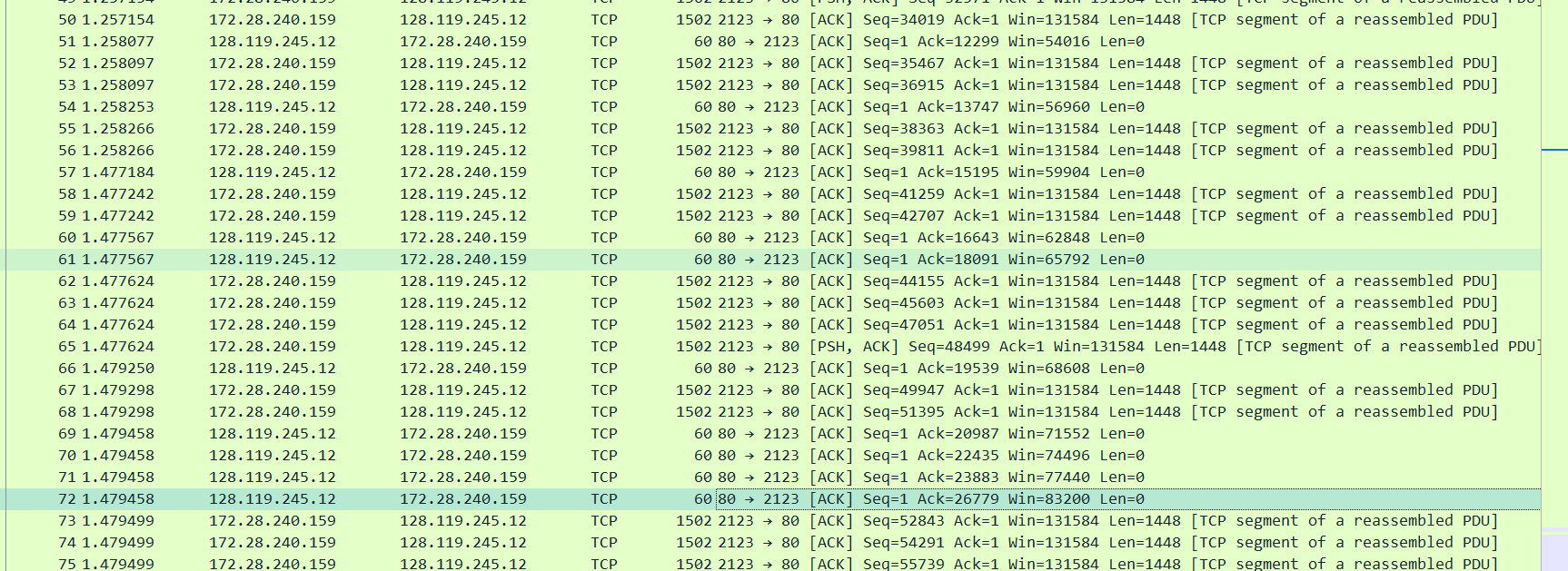


图12 数据长度

接受方要确认的数据长度主要为1448bytes，可以分辨。例如：第6个报文段的ACK是对第4及以前的报文段的累积确认，第9个报文段的ACK是对第5及以前的报文段的累积确认，共需要确认153036bytes。

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

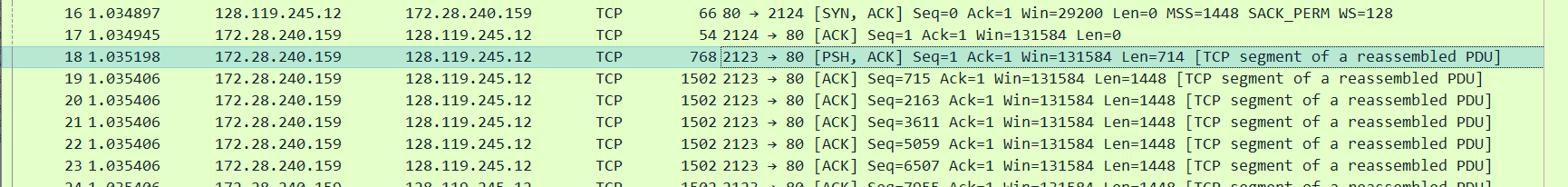


图13 第一次发送

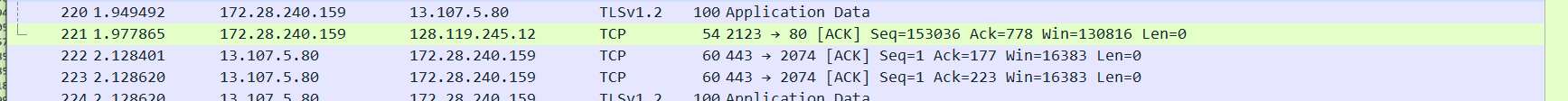


图14 最后一次发送

发送第一个报文段的时间为1.035198，发送最后一个报文段的时间为1.977865，期间发送的数据大小为153036bytes，因此：



**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.**

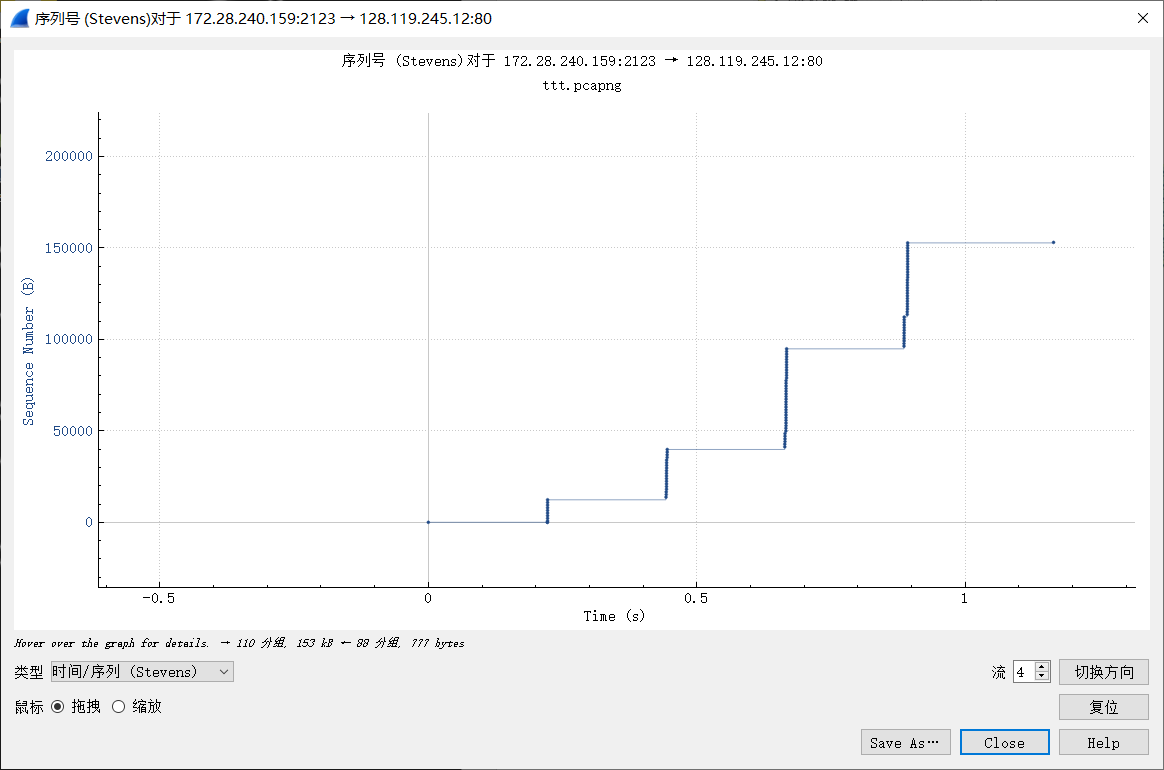


图15 时间序列

慢启动阶段：[0.225,0.229],[0.4426,0.445],[0.664,0.666],[0.886,0.892];

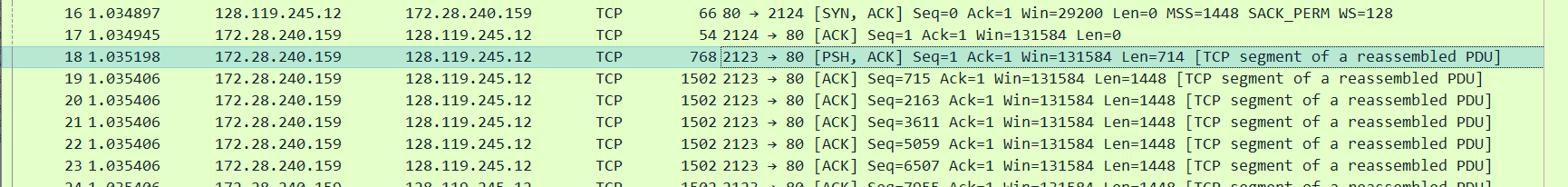
拥塞阶段：[0,0.225],[0.229,0.4426],[0.445,0.664],[0.666,0.886],[0.892,1.165];

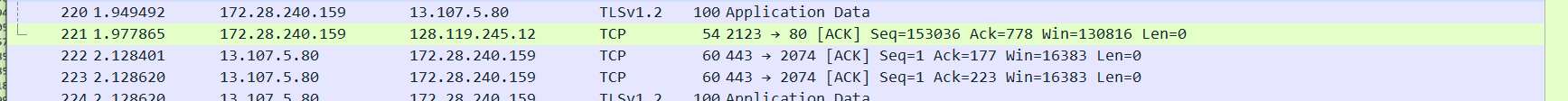
将该结果与书中的TCP理论相比较，理论上包含慢启动-快速恢复-拥塞避免三个状态，但快速恢复的状态是一个暂态，在实际过程中难以观测得到。

**14.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**

在本次实验中，我使用自己的trace，因此本题答案同上。发送第一个报文段的时间为1.035198，发送最后一个报文段的时间为1.977865，期间发送的数据大小为153036bytes，因此：







# 六、总结

在本次实验中，我成功地运用了Wireshark来捕获和分析HTTP报文。然而在实验过程中，我们也遇到了一些问题和困难。由于对Wireshark中许多窗口和选项不够熟悉，我们需要参考Wireshark实验指南以进行解释和说明。此外，我们还需查阅TCP协议的相关知识和公式，以正确计算RTT和EstimatedRTT。在这个过程中，还存在数据量过大的问题，这导致我们需要筛选出无关的信息，只保留与问题相关的数据。总的来说，通过本次实验，我们对网络协议的理解和应用有了更深入的认识。

在本次实验中，我学习并掌握了使用Wireshark抓取并分析HTTP报文的基本技能。了解了如何使用Wireshark的不同窗口和选项，如何分析HTTP头部和TCP头部，以及如何计算RTT和EstimatedRTT。通过分析HTTP报文，我还了解了HTTP协议的基本结构和数据传输的过程。此外，我们还学习了TCP协议的基本知识，如TCP三次握手和RTT计算方法等。本次实验也让我对网络通信过程和技术的理解更加深刻。