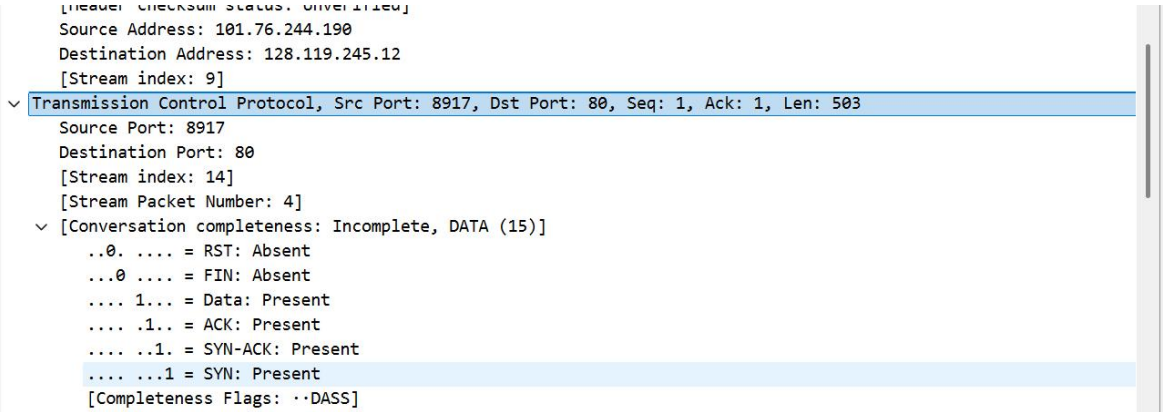


山东大学 计算机科学与技术 学院

新兴网络技术与实践 课程实验报告

学号：202300130183	姓名：宋浩宇	班级：23 级智能班
实验题目：Wireshark Lab TCP		
实验学时：2	实验日期：2025/4/2	
实验目的：学习 TCP		
<p>实验结果：</p> <p>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.</p>  <pre>[Ethernet II, Src: Intel E100, Dst: Intel E100] Source Address: 101.76.244.190 Destination Address: 128.119.245.12 [Stream index: 9] Transmission Control Protocol, Src Port: 8917, Dst Port: 80, Seq: 1, Ack: 1, Len: 503 Source Port: 8917 Destination Port: 80 [Stream index: 14] [Stream Packet Number: 4] Conversation completeness: Incomplete, DATA (15)] ...0... = RST: Absent ...0... = FIN: Absent ...1... = Data: Present ...1... = ACK: Present ...1... = SYN-ACK: Present ...1... = SYN: Present [Completeness Flags: ..DASS]</pre>		
<p>如图，ip 地址是 101.76.244.190，端口是 8917</p>		

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?

同样如上方图片，ip 地址是 128.119.245.12，端口是 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?

同样如上方图片，ip 地址是 101.76.244.190，端口是 8917

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?

```
[Completeness Flags: ..DASS]
[TCP Segment Len: 0]
Sequence Number: 0      (relative sequence number)
Sequence Number (raw): 2740126496
[Next Sequence Number: 1      (relative sequence number)]
```

如图，序列号是 2740126496

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?

```
[Completeness Flags: ..DASS]
[TCP Segment Len: 0]
Sequence Number: 0      (relative sequence number)
Sequence Number (raw): 2678331685
[Next Sequence Number: 1      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 2740126497
1000 .... = Header Length: 32 bytes (8)
```

如图，序列号是 2740126497，这是 SYN 号加 1 获得的

```

[Stream Packet Number: 2]
▼ [Conversation completeness: Incomplete, DATA (15)]
  ..0. .... = RST: Absent
  ...0 .... = FIN: Absent
  .... 1... = Data: Present
  .... .1.. = ACK: Present
  .... ..1. = SYN-ACK: Present
  .... ...1 = SYN: Present
[Completeness Flags: ..DASS]

```

图中这一位标明了这个字段是 SYN-ACK 序列号

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.

```

  .... ...1 = SYN: Present
[Completeness Flags: ..DASS]
[TCP Segment Len: 503]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 2740126497
[Next Sequence Number: 504 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 2678331686
0101 .... = Header Length: 20 bytes (5)

```

如图，序列号是 2740126497

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 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.

```

.... ...1 = SYN: Present
[Completeness Flags: ..DASS]
[TCP Segment Len: 503]
Sequence Number: 1      (relative sequence number)
Sequence Number (raw): 2740126497
[Next Sequence Number: 504      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 2678331686
0101 .... = Header Length: 20 bytes (5)

.... ...1. = SYN-ACK: Present
.... ...1 = SYN: Present
[Completeness Flags: ...ASS]
[TCP Segment Len: 0]
Sequence Number: 0      (relative sequence number)
Sequence Number (raw): 3841613905
[Next Sequence Number: 1      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 3825437684
1000 .... = Header Length: 32 bytes (8)
.... ...1 = SYN: Present
[Completeness Flags: ...ASS]
[TCP Segment Len: 0]
Sequence Number: 1      (relative sequence number)
Sequence Number (raw): 3825437684
[Next Sequence Number: 1      (relative sequence number)]
Acknowledgment Number: 1      (relative ack number)
Acknowledgment number (raw): 3841613906
0101 .... = Header Length: 20 bytes (5)
v Flags: 0x010 (ACK)
[Completeness Flags: ..DASS]
[TCP Segment Len: 0]
Sequence Number: 1      (relative sequence number)
Sequence Number (raw): 2678331686
[Next Sequence Number: 1      (relative sequence number)]
Acknowledgment Number: 504      (relative ack number)
Acknowledgment number (raw): 2740127000
0101 .... = Header Length: 20 bytes (5)

```

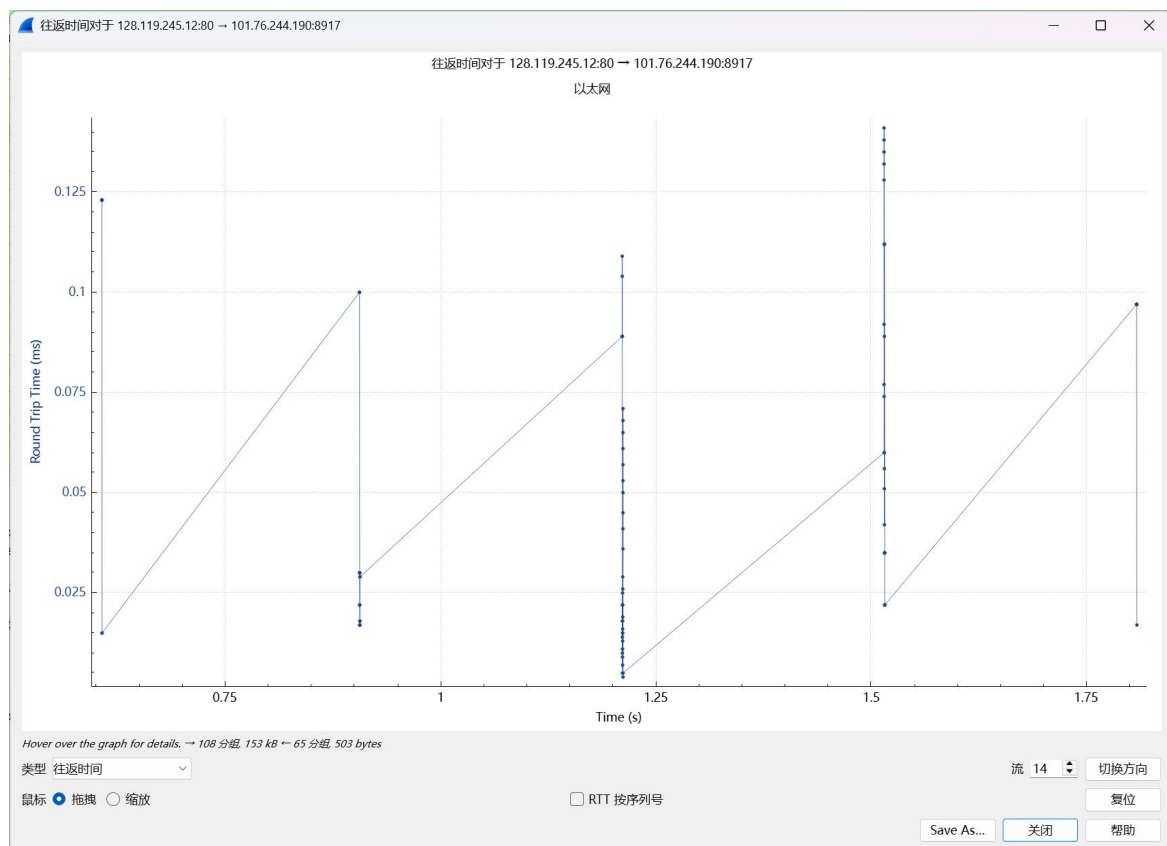
```

[Completeness Flags: ..DASS]
[TCP Segment Len: 1460]
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 2678331686
[Next Sequence Number: 1461    (relative sequence number)]
Acknowledgment Number: 504    (relative ack number)
Acknowledgment number (raw): 2740127000
0101 .... = Header Length: 20 bytes (5)

.... ...1 = SYN: Present
[Completeness Flags: ..DASS]
[TCP Segment Len: 1460]
Sequence Number: 1461    (relative sequence number)
Sequence Number (raw): 2678333146
[Next Sequence Number: 2921    (relative sequence number)
Acknowledgment Number: 504    (relative ack number)
Acknowledgment number (raw): 2740127000
0101 .... = Header Length: 20 bytes (5)

```

如图这是这六个段的序列号。



如图，这是每个段的 RTT 值，根据公式计算出来的 RTT 值是 0.304467

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


```

HTTP      557 GET /wireshark-labs/alice.txt HTTP/1.1
TCP       66 80 → 8918 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=128
TCP       54 8918 → 80 [ACK] Seq=1 Ack=1 Win=65280 Len=0
TCP       60 80 → 8917 [ACK] Seq=1 Ack=504 Win=30336 Len=0
TCP      1514 80 → 8917 [ACK] Seq=1 Ack=504 Win=30336 Len=1460 [TCP PDU reassembled in 704]
TCP      1514 80 → 8917 [ACK] Seq=1461 Ack=504 Win=30336 Len=1460 [TCP PDU reassembled in 704]
TCP      1514 80 → 8917 [ACK] Seq=2921 Ack=504 Win=30336 Len=1460 [TCP PDU reassembled in 704]
TCP      1514 80 → 8917 [ACK] Seq=4381 Ack=504 Win=30336 Len=1460 [TCP PDU reassembled in 704]

```

分别是 503、0、0、0、1460、1460

9. 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?

```

.....0 = Fin: Not set
[TCP Flags: .....A..S.]
Window: 29200
[Calculated window size: 29200]
Checksum: 0x54f7 [unverified]

```

最小的一次是 29200，接收方的缓冲区并没有限制发送方。

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

我的记录中并没有重传的段，我是通过 seq 字段判断的。

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).

```

Window: 237
[Calculated window size: 3033]
[Window size scaling factor:
Checksum: 0xb346 [unverified]
[Checksum Status: Unverified]
Window: 237
[Calculated window size: 30336]
[Window size scaling factor: 128]
Checksum: 0x5afd [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0

```

比如图中这一次确认了 $45894 - 23293 = 22,601$ 位

```

[Calculated window size: 3033
[Window size scaling factor:
Checksum: 0x002a [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
Window: 237
[Calculated window size: 30336]
[Window size scaling factor: 128]
Checksum: 0xd26a [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
Window: 237
[Calculated window size: 30336]
[Window size scaling factor: 128]
Checksum: 0x06f8 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0

```

从这连续的几个确认看，并不是一段一确认的。

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

1,921,319.796954315 bit/s

其中长度数据和时间数据分别来自：

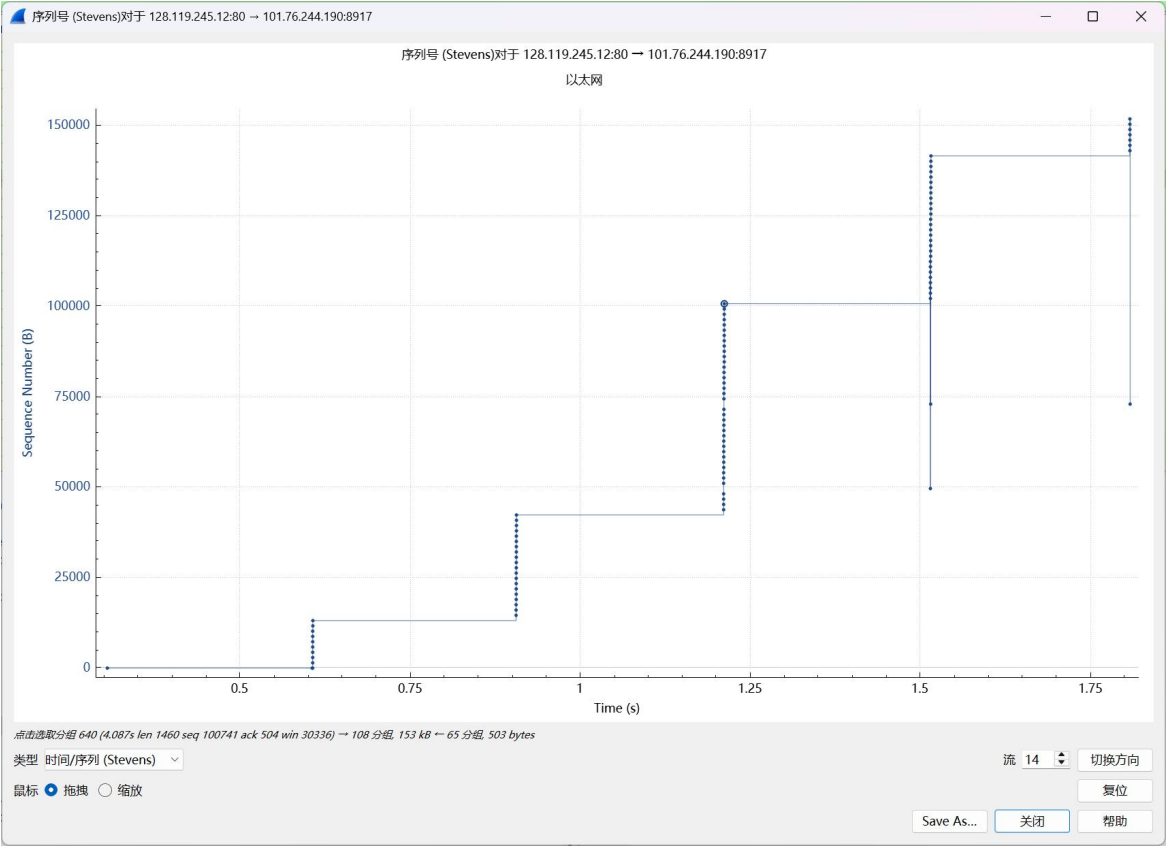
```

TCP 60 80 →
TCP 1514 80 →
TCP 1514 80 →
[Checksum Status: Unverified]
Urgent Pointer: 0
✓ [Timestamps]
[Time since first frame in this TCP stream: 0.606313000 seconds]
[Time since previous frame in this TCP stream: 0.000788000 seconds]
✓ [SEQ/ACK analysis]

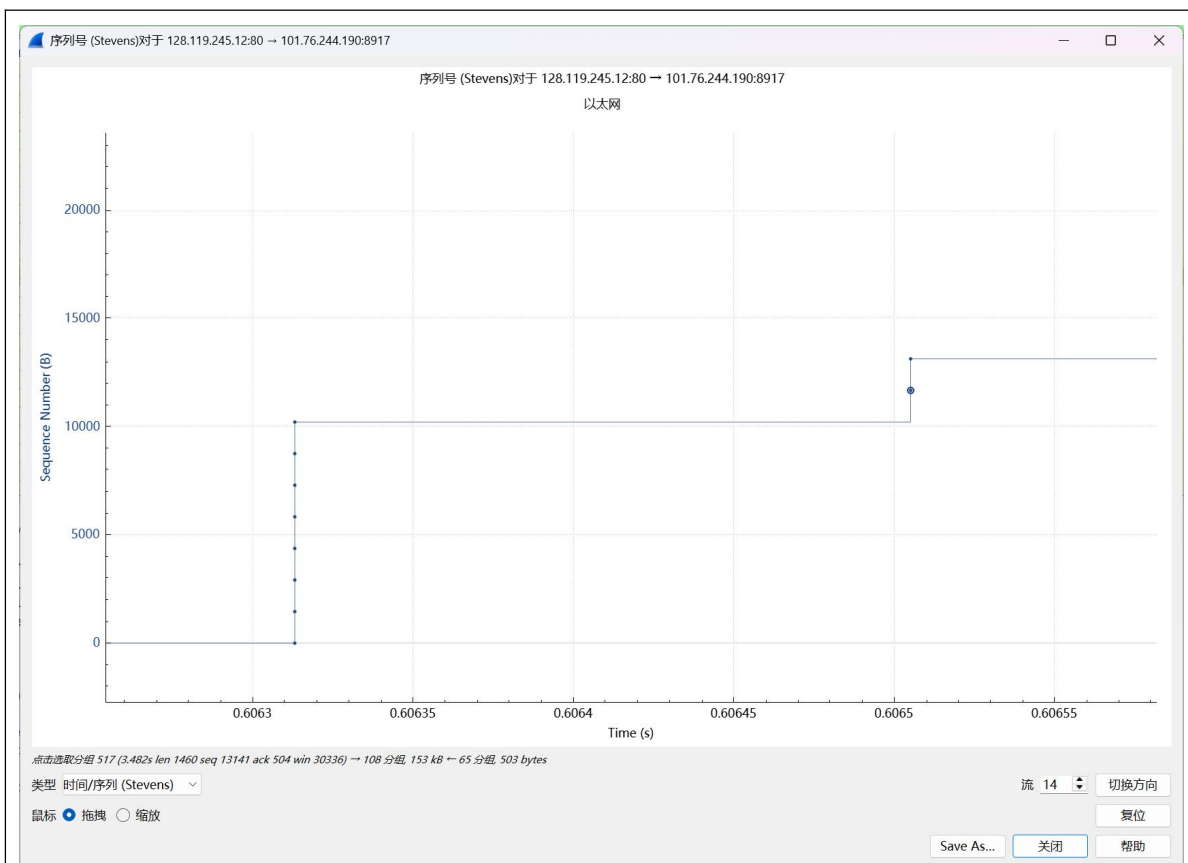
```

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.



这个是一个宏观的图，



放大后是这样的，可以看出这和教材上说的是基本一致的。

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`

请见上方问题的回答。

问题及收获：

通过本次 Wireshark 实验，我对 TCP 协议的工作机制有了更深入的理解。我观察到了 TCP 连接建立过程中的三次握手，包括 SYN、SYNACK 和 ACK 段的交互。通过分析序列号和确认号，我理解了 TCP 如何确保数据的可靠传输。时间序列图让我直观地看到了慢启动和拥塞避免阶段的转换，加深了我对 TCP 拥塞控制算法的认识。实验还让我了解了接收方通告的缓冲区空间如何影响发送方的发送速率。这些实践操作让我将理论知识与实际应用相结合，提升了对网络协议的理解和分析能力。