# 计算机学院 计算机网络 课程实验报告

| 实验题目： TCP | | 学号：202200130048 |
|---|---|---|
| 日期：4.2 | 班级： 6 | 姓名： 陈静雯 |
| Email：1205037094@qq.com | | |

实验方法介绍：

通过 wireshark 捕获 tcp 传输，具体根据信息和图表了解 TCP 相关知识

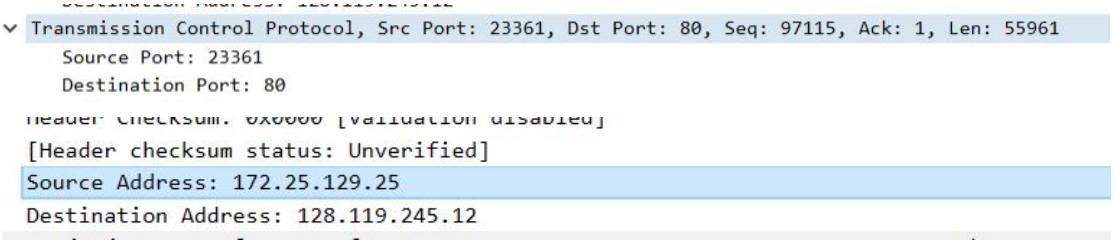
实验过程描述：

1. 捕获从计算机到远程服务器的批量 TCP 传输
2. 首先看一下捕获的跟踪
3. TCP 基础知识
4. TCP 拥塞控制在行动

结论分析：

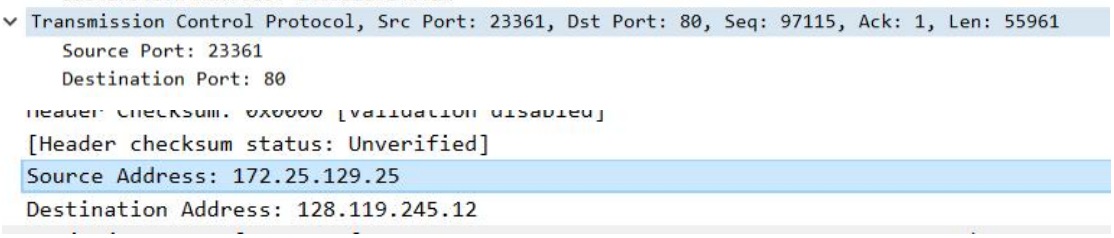
1. What is the IP address and TCP port number used by the client computer (source) that is transferring the alice.txt file to gaia.cs.umass.edu?

源 IP:172.25.129.25 端口号：23361



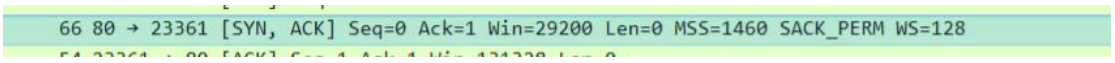

2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

IP：128.119.245.12 端口号：80




3. 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?

Seq=27153442391（raw）；seq=0（relative）



```
Sequence Number: 0    (relative sequence number)
Sequence Number (raw): 2715342391
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 0
Acknowledgment number (raw): 0
1000 .... = Header Length: 32 bytes (8)
```

What is it in this TCP segment that identifies the segment as a SYN segment? Will the TCP receiver in this session be able to use Selective Acknowledgments?

Flags； not selective

```
Flags: 0x002 (SYN)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Accurate ECN: Not set
    .... 0... .... = Congestion Window Reduced: Not set
    .... .0.. .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...0 .... = Acknowledgment: Not set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
  .... .... ..1. = Syn: Set
    [Expert Info (Chat/Sequence): Connection establish request (SYN): server port 80]
        [Connection establish request (SYN): server port 80]
        [Severity level: Chat]
        [Group: Sequence]
```

4. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN?

Seq=428255010；

```
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 0    (relative sequence number)
Sequence Number (raw): 428255010
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 1    (relative ack number)
Acknowledgment number (raw): 2715342392
```

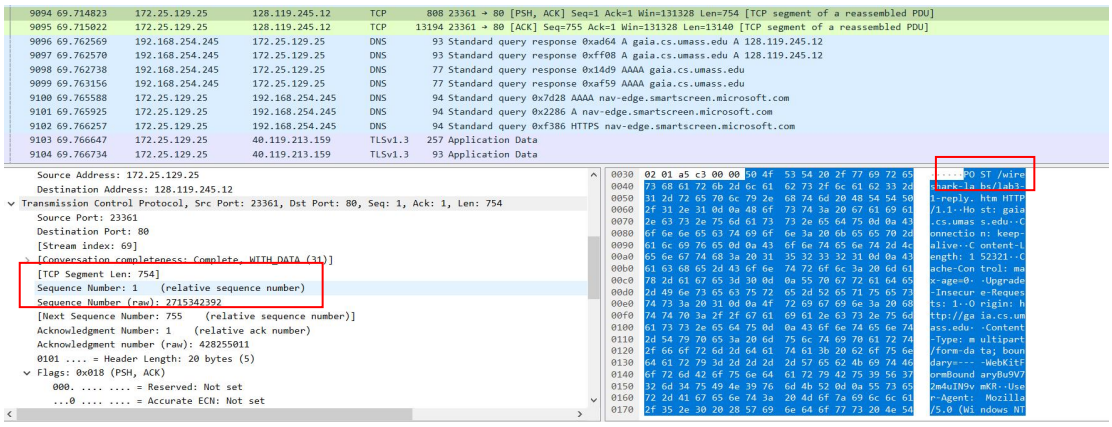
What is it in the segment that identifies the segment as a SYNACK segment? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value?

Flags； set ；确认连接请求，返回一个 SYN ACK 段，ack 为 1

```
Flags: 0x012 (SYN, ACK)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Accurate ECN: Not set
    .... 0... .... = Congestion Window Reduced: Not set
    .... .0.. .... = ECN-Echo: Not set
    .... ..0. .... = Urgent: Not set
    .... ...1 .... = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
  .... .... ..1. = Syn: Set
    .... .... ...0 = Fin: Not set
    [TCP Flags: ·····A··S·]
```

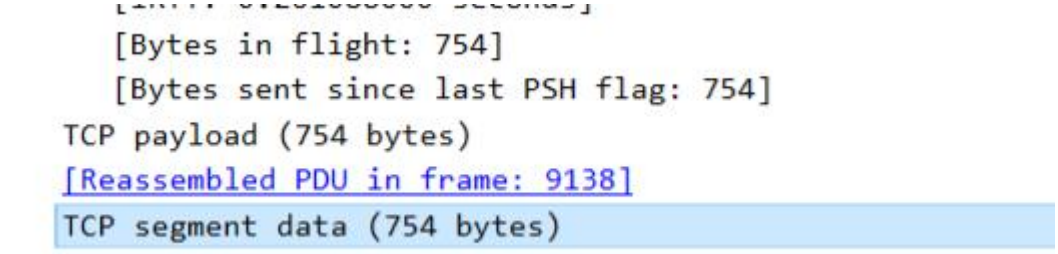
5. What is the sequence number of the TCP segment containing the header of the HTTP POST command?

Seq=2715342392（raw）；seq（relative）=1




How many bytes of data are contained in the payload (data) field of this TCP segment? Did all of the data in the transferred file alice.txt fit into this single segment?

754bytes；不在一个段里

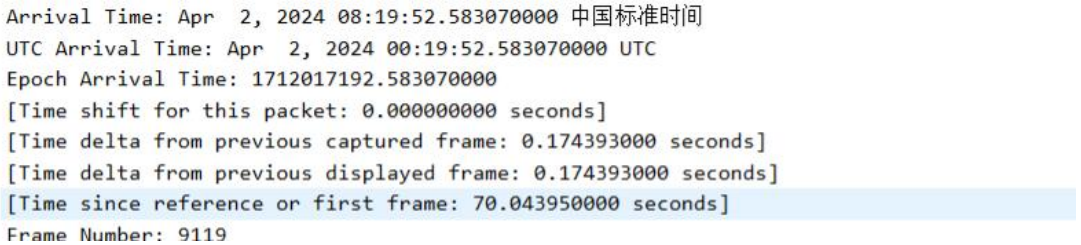



6. At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent?

69.714823s

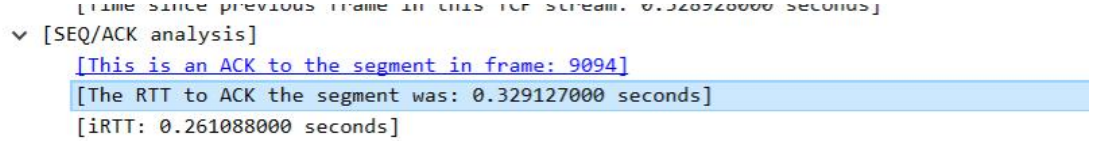

At what time was the ACK for this first data-containing segment received?

70.04395s



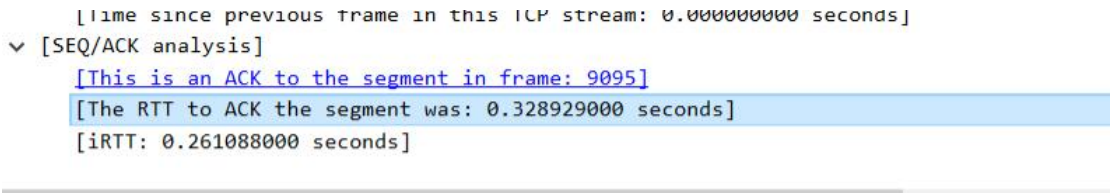

What is the RTT for this first data-containing segment?

0.329127s



What is the RTT value the second data-carrying TCP segment and its ACK?

**0.261088s**




What is the EstimatedRTT value (see Section 3.5.3, in the text) after the ACK for the second data-carrying segment is received? α = 0.125

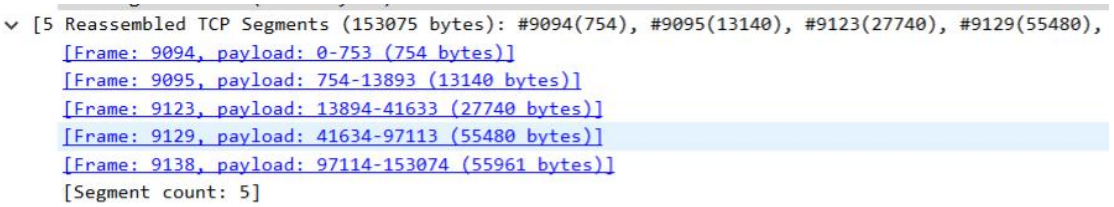
**EstimatedRTT = (1 - α) × CurrentEstimatedRTT + α × SampleRTT**

**= (1-0.125) * 0.329127 + 0.125 * 0.261088 = 0.320622125s**

7. What is the length (header plus payload) of each of the first four data-carrying TCP segments?

**Header=20;**

**754+20=774; 13140+20=13160; 27740+20=27760; 55480+20=55500**

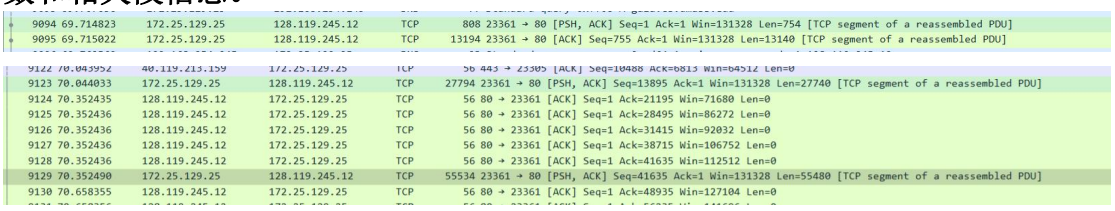



8. What is the minimum amount of available buffer space advertised to the client by gaia.cs.umass.edu among these first four data-carrying TCP segments ? Does the lack of receiver buffer space ever throttle the sender for these first four data- carrying segments?

**513*256=131328，最小可用缓冲空间为 131328；空间不足会限制段的使用**


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

**没有重传段。检查是否有[TCP Retransmission]标记，在 Wireshark 中，右键点击感兴趣的 TCP 流，选择“Analyze” -> “TCP Stream”，在弹出的对话框中查看“Retransmissions”区域，如果有重传发生，这里会列出具体的重传次数和相关段信息。**




10. How much data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the client to gaia.cs.umass.edu? Can you identify cases where the receiver is ACKing

every other received segment(see Table 3.2 in the text) among these first ten data-carrying segments?

大约 6000 左右数据

```
   808 23361 → 80 [PSH, ACK] Seq=1 Ack=1 Win=131328 Len=754 [TCP segment of a reassembled PDU]
 13194 23361 → 80 [ACK] Seq=755 Ack=1 Win=131328 Len=13140 [TCP segment of a reassembled PDU]
    56 80 → 23361 [ACK] Seq=1 Ack=755 Win=30720 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=8055 Win=45312 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=13895 Win=57088 Len=0
 27794 23361 → 80 [PSH, ACK] Seq=13895 Ack=1 Win=131328 Len=27740 [TCP segment of a reassembled PDU]
    56 80 → 23361 [ACK] Seq=1 Ack=21195 Win=71680 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=28495 Win=86272 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=31415 Win=92032 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=38715 Win=106752 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=41635 Win=112512 Len=0
 55534 23361 → 80 [PSH, ACK] Seq=41635 Ack=1 Win=131328 Len=55480 [TCP segment of a reassembled PDU]
    56 80 → 23361 [ACK] Seq=1 Ack=48935 Win=127104 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=56235 Win=141696 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=63535 Win=156288 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=70835 Win=170880 Len=0
    56 80 → 23361 [ACK] Seq=1 Ack=78135 Win=179584 Len=0
```

```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 428255011
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 755    (relative ack number)
Acknowledgment number (raw): 2715343146
0101 .... = Header Length: 20 bytes (5)
```

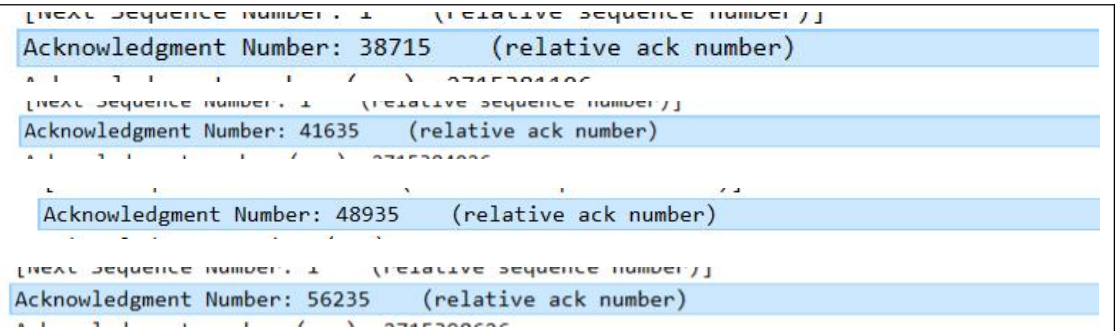
```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 428255011
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 8055    (relative ack number)
Acknowledgment number (raw): 2715350446
0101 .... = Header Length: 20 bytes (5)
```

```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 428255011
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 13895    (relative ack number)
Acknowledgment number (raw): 2715356286
```

```
[TCP Segment Len: 0]
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 428255011
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 21195    (relative ack number)
Acknowledgment number (raw): 2715363586
0101 .... = Header Length: 20 bytes (5)
```

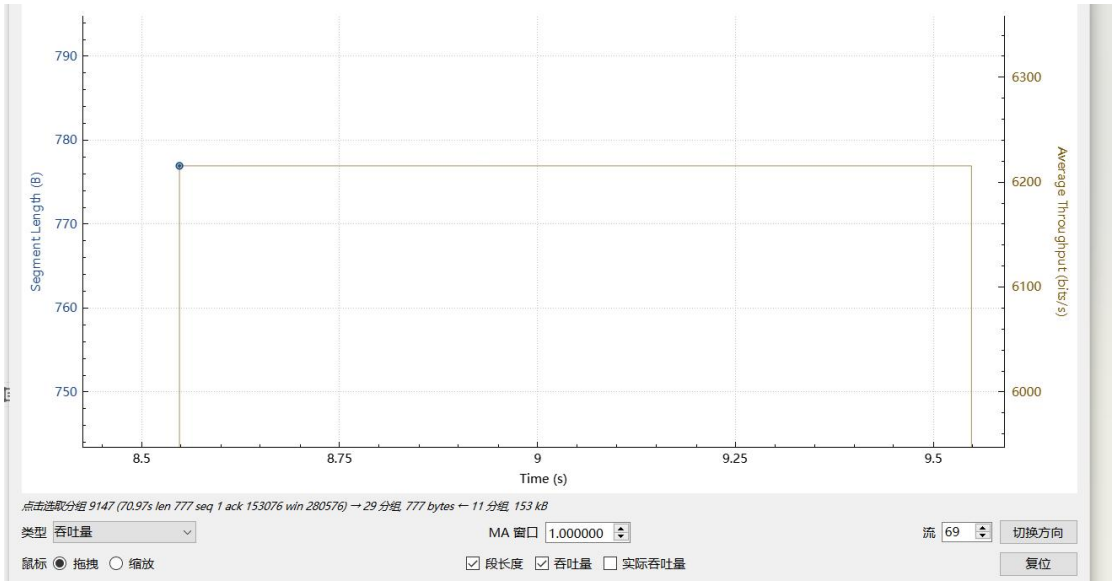
```
Sequence Number: 1    (relative sequence number)
Sequence Number (raw): 428255011
[Next Sequence Number: 1    (relative sequence number)]
Acknowledgment Number: 28495    (relative ack number)
Acknowledgment number (raw): 2715370886
0101 .... = Header Length: 20 bytes (5)
```

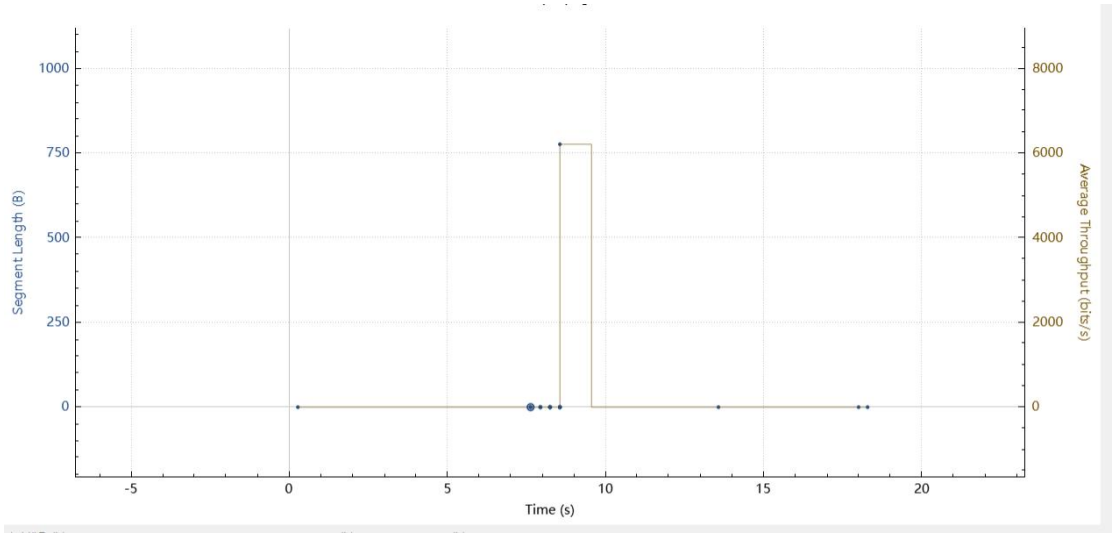
```
Acknowledgment Number: 31415    (relative ack number)
```




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

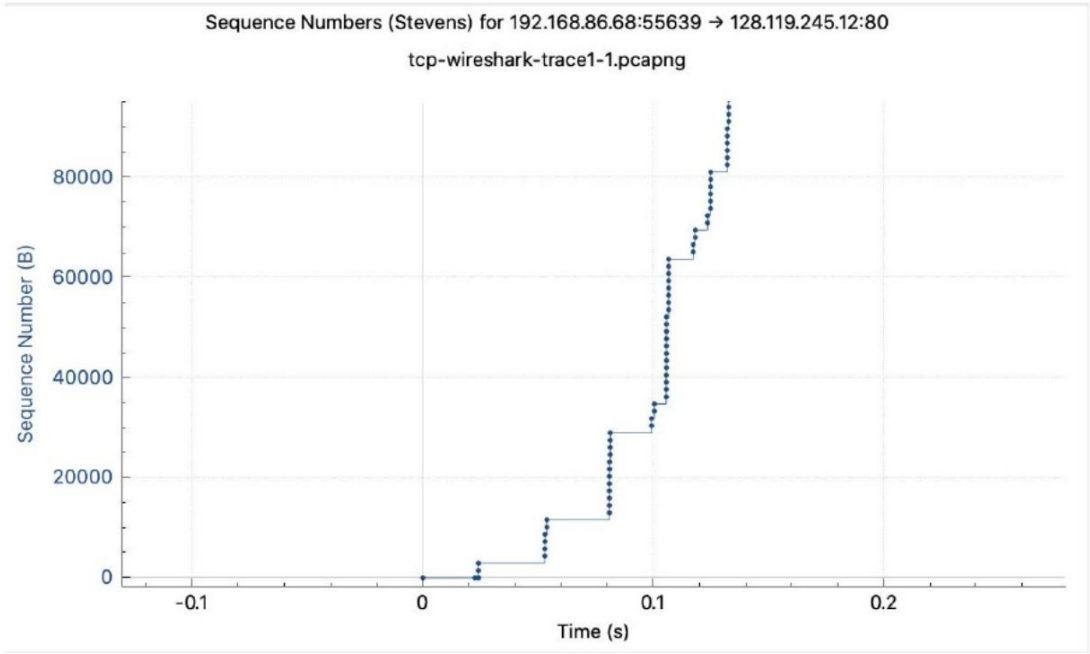
**吞吐率 = 总字节数 / 时间单位 = 777bytes / 1s = 6216bits/s**




12. 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.Consider the “fleets” of packets sent around t = 0.025, t = 0.053, t = 0.082 and t = 0.1.

**根据题设给出的图可以看出 0.1 之前的传输间隔时间在变短而传输的数据在成倍变多，处于缓慢启动阶段。**

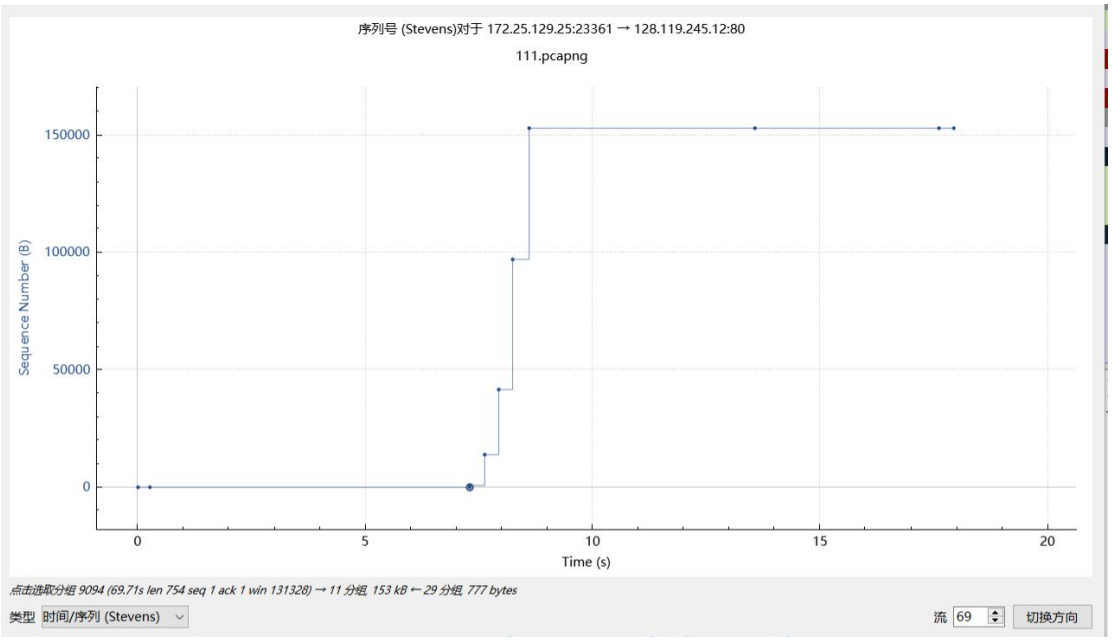



13. These “fleets” of segments appear to have some periodicity. What can you say about the period?

0.1s 前在数据成倍增长，而 0.14s 后数据从一个较高值大幅度减半，这通常意味着网络严重拥塞已被检测到，TCP 进入慢启动阶段重新开始增长，即拥塞发生与拥塞窗口减半阶段。可以看出这个周期就是从慢启动到拥塞一个不断重复的过程。

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

（12）在 7.3s-8.2s 可以看出慢启动，传输的数据成倍增加。




（13） 在 8.2s 之后也就是最后一个传输增幅很小，进入了拥塞避免阶段，直到传输结束，可以看出每段传输间隔时间相同。

结论：
1.三次握手进行 TCP 传输（SYN 段，SYN ACK 段，ACK 段）
2.吞吐率 = 总字节数 / 时间单位
3.大文件会进行多次 TCP 传输
4.拥塞控制
（1）慢启动阶段：当 TCP 连接刚建立或者经历过重置后，通常从一个较小的值开始增长。在慢启动阶段，每次收到一个确认 ACK 时，cwnd 大小翻倍。
（2）拥塞避免阶段：当 cwnd 达到某个阈值时，TCP 进入拥塞避免阶段。在此阶段，cwnd 的增长不再是指数级，而是按照每个往返时间（RTT）增加 1 个 MSS 的线性速度增长。序列号间隔有规律地、小幅度地递增。
（3）快速重传与快速恢复：若在图中观察到序列号突然跳跃回退，随后 cwnd 有所减小并重新开始增长，这表明网络中出现了短暂的拥塞，TCP 正在执行快速重传与恢复机制。
（4）拥塞发生与拥塞窗口减半：若序列号增长突然大幅下降，且后续 cwnd 从一个较高值大幅度减半，这通常意味着网络严重拥塞已被检测到，TCP 进入慢启动阶段重新开始增长。