计算机网络实验报告

课程名称_____计算机网络_____成绩评定_____

实验项目名称_____TCP 协议_____指导教师____张伟____

实验项目编号__实验 6_____实验项目类型_____实验地点_____

学生姓名_____密语_____学号___202200460104_____

学院_____网络空间安全_____专业___网络空间安全_____

实验时间__2024__年__4__月__30__日

一、 实验目的

1. 熟悉 TCP 的协议格式。
2. 理解 TCP 对序列号和确认号的使用。
3. 理解 TCP 的流量控制算法和拥塞控制算法。

二、 实验步骤与结果

任务一：

1. 首先使用 ping 指令，我们可以得知 gaia.cs.umass.edu.com 对应的 IP 地址

```
C:\Users\86135>ping gaia.cs.umass.edu

正在 Ping gaia.cs.umass.edu [128.119.245.12] 具有 32 字节的数据:
来自 128.119.245.12 的回复: 字节=32 时间=285ms TTL=37
来自 128.119.245.12 的回复: 字节=32 时间=286ms TTL=37
请求超时。
来自 128.119.245.12 的回复: 字节=32 时间=298ms TTL=37

128.119.245.12 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 3, 丢失 = 1 (25% 丢失),
往返行程的估计时间(以毫秒为单位):
    最短 = 285ms, 最长 = 298ms, 平均 = 289ms
```

2. 然后使用 Wireshark 抓包得到计算机与 gaia.cs.umass.edu.com 之间的一系列 TCP 通信和 HTTP 通信:

74	2.992922	113.240.75.252	172.25.170.244	TCP	60	443 → 58769 [ACK] Seq=363 Ack=7408 Win=64128 Len=0
75	3.011903	128.119.245.12	172.25.170.244	TCP	66	80 → 58768 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=1
76	3.012005	172.25.170.244	128.119.245.12	TCP	54	58768 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
77	3.012232	172.25.170.244	128.119.245.12	TCP	66	58772 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
78	3.012371	172.25.170.244	128.119.245.12	HTTP	993	POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1
79	3.017741	128.119.245.12	172.25.170.244	TCP	60	80 → 58694 [ACK] Seq=1 Ack=2 Win=229 Len=0
80	3.049020	20.54.232.160	172.25.170.244	TLSv1.2	544	Application Data
81	3.095517	172.25.170.244	20.54.232.160	TCP	54	58767 → 443 [ACK] Seq=5922 Ack=9394 Win=131840 Len=0
82	3.095529	172.25.170.244	128.119.245.12	TCP	54	[TCP Retransmission] 58693 → 80 [FIN, ACK] Seq=1 Ack=1 Win=510 Len=0
83	3.263893	128.119.245.12	172.25.170.244	TCP	66	80 → 58772 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM WS=1
84	3.263997	172.25.170.244	128.119.245.12	TCP	54	58772 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
85	3.266013	128.119.245.12	172.25.170.244	TCP	60	80 → 58768 [ACK] Seq=1 Ack=940 Win=31104 Len=0
86	3.267941	128.119.245.12	172.25.170.244	HTTP	831	HTTP/1.1 200 OK (text/html)
87	3.322706	172.25.170.244	128.119.245.12	TCP	54	58768 → 80 [ACK] Seq=940 Ack=778 Win=130560 Len=0
88	3.704229	172.25.170.244	128.119.245.12	TCP	54	[TCP Retransmission] 58693 → 80 [FIN, ACK] Seq=1 Ack=1 Win=510 Len=0
89	3.727357	172.25.170.244	51.105.71.137	TLSv1.2	138	Application Data
90	3.727426	172.25.170.244	51.105.71.137	TLSv1.2	93	Application Data
91	3.727467	172.25.170.244	51.105.71.137	TLSv1.2	1464	Application Data
92	3.771489	172.25.170.244	111.30.169.83	TCP	174	58220 → 8080 [PSH, ACK] Seq=1 Ack=1 Win=517 Len=120
93	3.795046	111.30.169.83	172.25.170.244	TCP	134	8080 → 58220 [PSH, ACK] Seq=1 Ack=121 Win=755 Len=80
94	3.838377	172.25.170.244	111.30.169.83	TCP	54	58220 → 8080 [ACK] Seq=121 Ack=81 Win=517 Len=0
96	4.036296	51.105.71.137	172.25.170.244	TCP	60	443 → 58484 [ACK] Seq=1 Ack=124 Win=16381 Len=0
97	4.036296	51.105.71.137	172.25.170.244	TLSv1.2	93	Application Data
98	4.036296	51.105.71.137	172.25.170.244	TCP	60	443 → 58484 [ACK] Seq=40 Ack=1534 Win=16385 Len=0
99	4.092382	172.25.170.244	51.105.71.137	TCP	54	58484 → 443 [ACK] Seq=1534 Ack=40 Win=513 Len=0
101	4.378551	51.105.71.137	172.25.170.244	TLSv1.2	148	Application Data
102	4.379635	172.25.170.244	51.105.71.137	TLSv1.2	89	Application Data
104	4.707541	51.105.71.137	172.25.170.244	TCP	60	443 → 58484 [ACK] Seq=134 Ack=1569 Win=16384 Len=0
106	4.905515	172.25.170.244	128.119.245.12	TCP	54	[TCP Retransmission] 58693 → 80 [FIN, ACK] Seq=1 Ack=1 Win=510 Len=0
119	7.325943	172.25.170.244	128.119.245.12	TCP	54	[TCP Retransmission] 58693 → 80 [FIN, ACK] Seq=1 Ack=1 Win=510 Len=0
123	8.272200	128.119.245.12	172.25.170.244	TCP	60	80 → 58768 [FIN, ACK] Seq=778 Ack=940 Win=31104 Len=0
124	8.272262	172.25.170.244	128.119.245.12	TCP	54	58768 → 80 [ACK] Seq=940 Ack=779 Win=130560 Len=0

任务二：

1. 将文件传输到 gaia.cs.umass.edu 的客户端计算机（源）使用的 IP 地址和 TCP 端口号是什么？

从 wireshark 抓包信息中可以看到，源 IP 地址是 172.25.170.244，TCP 端口号为 60305

```
> Frame 206: 535 bytes on wire (4280 bits), 535 bytes captured (4280 bits) on interface \Device\NPF_{8E4
> Ethernet II, Src: HaiyingZhili_3f:32:a0 (08:26:ae:3f:32:a0), Dst: JuniperNetwo_f6:12:a0 (28:a2:4b:f6:1
> Internet Protocol Version 4, Src: 172.25.170.244, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 60305, Dst Port: 80, Seq: 152595, Ack: 1, Len: 481
> [ [truncated]106 Reassembled TCP Segments (153075 bytes): #48(754), #49(1460), #50(1460), #51(1460), #
```

2. gaia.cs.umass.edu 的 IP 地址是什么？在哪个端口号上发送和接收此连接的 TCP 报文段？

由上图和上面 ping 的信息可以得知，gaia.cs.umass.edu.com 的 IP 地址是 128.119.245.12，接收端接口号为 80

任务三：

1. **用于在客户端计算机和 gaia.cs.umass.edu 之间启动 TCP 连接的 TCP SYN 报文段的序列号（sequence number）是什么？TCP SYN 报文段有什么作用？**

TCP SYN 报文段的序列号是 232129012

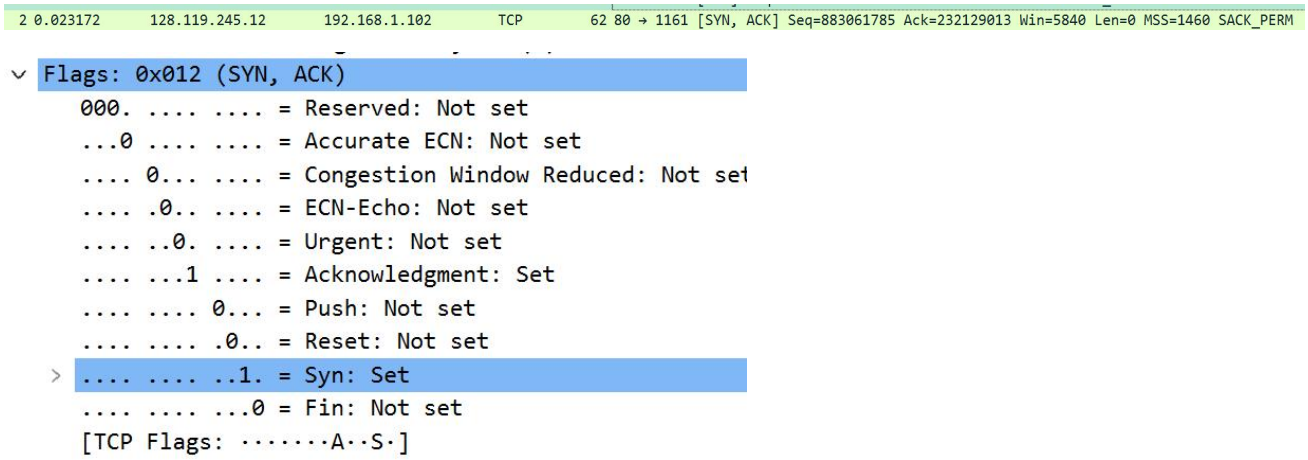
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM

SYN 报文段表示客户端发起连接，是三次握手的第一次握手。

2. gaia.cs.umass.edu 发送给客户端计算机以回复 SYN 的 SYN-ACK 报文段的序列号是多少？SYN ACK 报文段中的 Acknowledgment number 栏位的值是多少？gaia.cs.umass.edu 是如何确定此 Acknowledgment number 的数值的？TCP SYN-ACK 报文段有什么作用？

gaia.cs.umass.edu 发送给客户端计算机以回复 SYN 的 SYN-ACK 报文段的序列号是 883061785，SYN ACK 报文段中的 Acknowledgment number 栏位的值是 1，从 flags 中可以看出。gaia.cs.umass.edu 通过客户端发送的 SYN 的 Acknowledgment number 值加一得到此 Acknowledgment number 的值。

SYN ACK 区段是 TCP 三次握手中的第二次握手，表明服务器接受连接。



3. 包含 HTTP POST 命令的 TCP 报文段的序列号是多少？请注意，为了找到 POST 命令，你需要深入了解 Wireshark 窗口底部的数据包内容字段，在其 DATA 栏位中查找带有“POST”的报文段。

包含 HTTP POST 命令的 TCP 报文段的序列号为 232129013。

```
44 70 1f bd 00 00 50 4f  53 54 20 2f 65 74 68 65   Dp····PO ST /ethe
72 65 61 6c 2d 6c 61 62  73 2f 6c 61 62 33 2d 31  real-lab s/lab3-1
2d 72 65 70 6c 79 2e 68  74 6d 20 48 54 54 50 2f  -reply.h tm HTTP/
31 2e 31 0d 0a 48 6f 73  74 3a 20 67 61 69 61 2e  1.1··Hos t: gaia.
63 73 2e 75 6d 61 73 73  2e 65 64 75 0d 0a 55 73  cs.umass .edu··Us
65 72 2d 41 67 65 6e 74  3a 20 4d 6f 7a 69 6c 6c  er-Agent : Mozill
61 2f 35 2e 30 20 28 57  69 6e 64 6f 77 73 3b 20  a/5.0 (W indows;
55 3b 20 57 69 6e 64 6f  77 73 20 4e 54 20 35 2e  U; Windo ws NT 5.
31 3b 20 65 6e 2d 55 53  3b 20 72 76 3a 31 2e 30  1; en-US ; rv:1.0
2e 32 29 20 47 65 63 6b  6f 2f 32 30 30 33 30 32  .2) Geck o/200302
30 38 20 4e 65 74 73 63  61 70 65 2f 37 2e 30 32  08 Netsc ape/7.02
0d 0a 41 63 63 65 70 74  3a 20 74 65 78 74 2f 78  ··Accept : text/x
```

在第四条流量中发现 POST 命令。

4. 观察编号（No.）3 和 4 的 TCP 报文段的序列号，你有什么发现？请解释这个现象的原因。

```
3 0.023265    192.168.1.102    128.119.245.12    TCP    54 1161 → 80 [ACK] Seq=232129013 Ack=883061786 Win=17520 Len=0
4 0.026477    192.168.1.102    128.119.245.12    TCP    619 1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565
```

编号（No.）3 和 4 的 TCP 报文段的序列号是一样的，都是 232129013

原因：3 号报文的 len 为 0。

5. 将包含 HTTP POST 的 TCP 报文段视为 TCP 连接中的第一个报文段。在这个 TCP 连接中前六个用于数据发送的 TCP 报文段是那些（列出编号，即查看“No.”这一列）？序列号分别是多少（包括包含 HTTP POST 的报文段）？这六个报文段发送的时间是什么时候?

NO	序列号	发送时间
4	232129013	0.026477
5	232129578	0.041737
7	232131038	0.054026
8	232132498	0.054690
10	232133958	0.077405
11	232135418	0.078157

6. 收到的对应前六个数据发送 TCP 报文段的确认 ACK 分别是在那些 TCP 报文段里（列出编号，即查看“No.”这一列）？是什么时候收到的？

NO	收到时间
6	0.053937
9	0.077294
12	0.124085

7. 鉴于发送每个 TCP 报文段的时间与收到确认的时间之间的差异，前六个数据发送 TCP 报文段中每个报文段的往返时间（RTT）是多少？加权平均往返时延（RTTS）是多少？

用收到时间减发送时间得到往返时间 RTT

NO	RTT	RTTS
4	0.026477	0.026477
5	0.035557	0.028472125
7	0.070059	0.033670484375
8	0.114428	0.043765173828125
10	0.139894	0.05578127709960937
11	0.189645	0.07251424246215821

8. 前六个数据发送 TCP 报文段的长度是多少？

```
  4 0.026477    192.168.1.102    128.119.245.12   TCP     619 1161 → 80 [PSH, ACK] Seq=232129013 Ack=883061786 Win=17520 Len=565
  5 0.041737    192.168.1.102    128.119.245.12   TCP    1514 1161 → 80 [PSH, ACK] Seq=232129578 Ack=883061786 Win=17520 Len=1460
  6 0.053937    128.119.245.12   192.168.1.102    TCP      60 80 → 1161 [ACK] Seq=883061786 Ack=232129578 Win=6780 Len=0
  7 0.054026    192.168.1.102    128.119.245.12   TCP    1514 1161 → 80 [ACK] Seq=232131038 Ack=883061786 Win=17520 Len=1460
  8 0.054690    192.168.1.102    128.119.245.12   TCP    1514 1161 → 80 [ACK] Seq=232132498 Ack=883061786 Win=17520 Len=1460
  9 0.077294    128.119.245.12   192.168.1.102    TCP      60 80 → 1161 [ACK] Seq=883061786 Ack=232131038 Win=8760 Len=0
 10 0.077405    192.168.1.102    128.119.245.12   TCP    1514 1161 → 80 [ACK] Seq=232133958 Ack=883061786 Win=17520 Len=1460
 11 0.078157    192.168.1.102    128.119.245.12   TCP    1514 1161 → 80 [ACK] Seq=232135418 Ack=883061786 Win=17520 Len=1460
```

由图可知，长度为 565、1460、1460、1460、1460、1460

9. 对于整个抓包过程，收到的服务器声明的最小可用接收缓冲区空间（接收窗口）大小是

多少？整个过程中，声明的接收缓冲区空间是否限制了发送方传送 TCP 报文段?

2	0.023172	128.119.245.12	192.168.1.102	TCP	62 80 → 1161 [SYN, ACK] Seq=883061785 Ack=232129013 Win=5840 Len=0 MSS=1460 SACK_PERM

由上图，最小的接收窗口大小为 5840，没有限制发送方传送 TCP 报文段，因为接收窗口的大小远大于发送的报文的数量。

10. 在抓包文件中是否有重传的报文段？为了回答这个问题，你检查了什么（在抓包文件中）？

使用 ip.src==192.168.1.102，检查了主机发送的所有的报文，发现序号（NO.）一直是增加的，证明没有重传的报文段。

11. 接收方通常在 ACK 中确认多少数据（确认收到的多少个报文段）？你是否可以识别接收方每隔一个接收到的报文段才发送确认 ACK 的情况?

ACK 通常确认 1048 字节数据，跟每个报文段发送的数据一致。

78	1.758227	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232181905 Win=62780 Len=0
79	1.860063	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232184825 Win=62780 Len=0
80	1.930880	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232187177 Win=62780 Len=0
81	1.931099	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=232187177 Ack=883061786 Win=17520 Len=1460
82	1.931879	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=232188637 Ack=883061786 Win=17520 Len=1460
83	1.932757	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=232190097 Ack=883061786 Win=17520 Len=1460
84	1.933636	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=232191557 Ack=883061786 Win=17520 Len=1460
85	1.934770	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=232193017 Ack=883061786 Win=17520 Len=1460
86	1.935586	192.168.1.102	128.119.245.12	TCP	946 1161 → 80 [PSH, ACK] Seq=232194477 Ack=883061786 Win=17520 Len=892
87	2.029069	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232190097 Win=62780 Len=0
88	2.126682	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232193017 Win=62780 Len=0
89	2.203195	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=883061786 Ack=232195369 Win=62780 Len=0

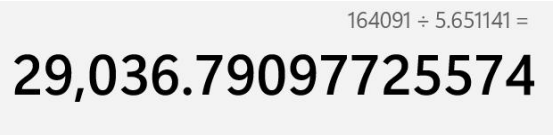
图中情况即为隔一个报文段才确认。

12. TCP 连接的吞吐量（每单位时间传输的字节数）是多少？解释你如何计算这个值。

1	0.000000	192.168.1.102	128.119.245.12	TCP	62 1161 → 80 [SYN] Seq=232129012 Win=16384 Len=0 MSS=1460 SACK_PERM
206	5.651141	192.168.1.102	128.119.245.12	TCP	54 1161 → 80 [ACK] Seq=232293103 Ack=883062516 Win=16790 Len=0

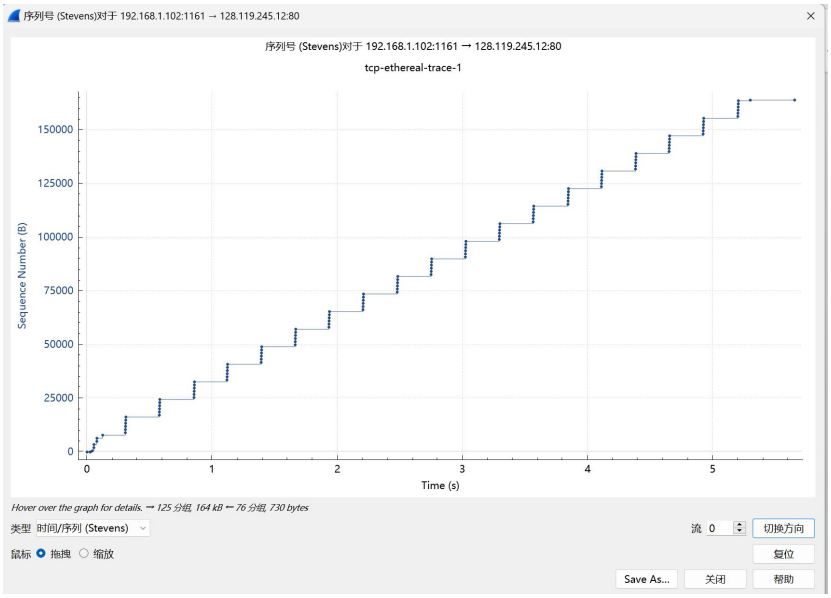
发送的字节数为 164091，时间为 5.651141s

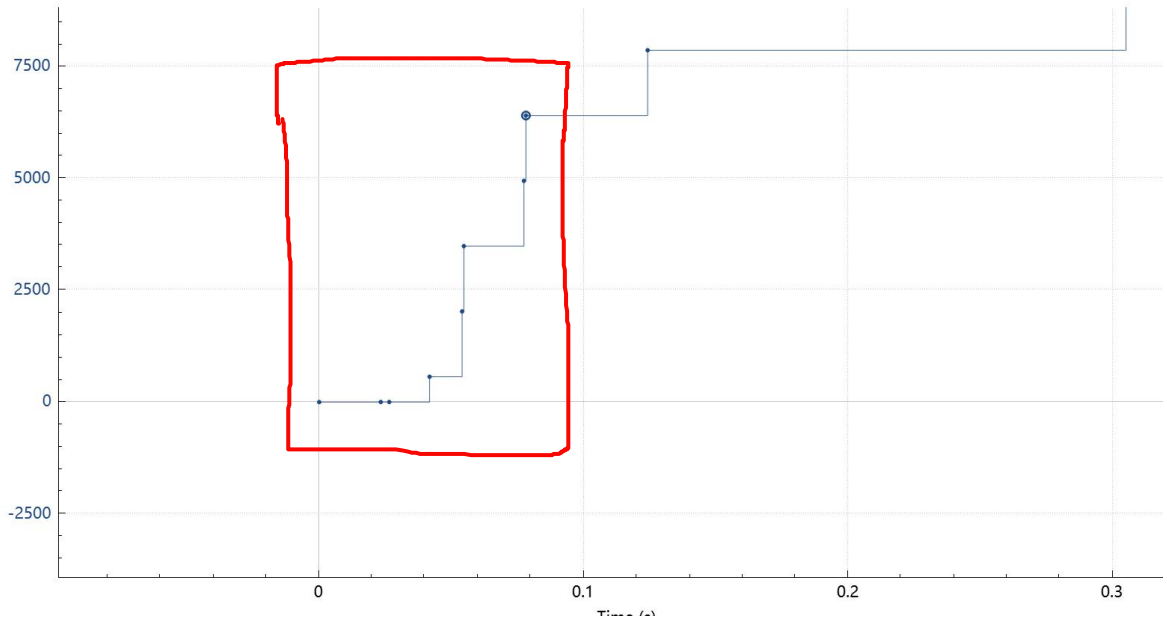
故，吞吐量为：29.036 KByte/s



任务四：

1. 使用时间序列图（Stevens）绘图工具查看从客户端发送到 gaia.cs.umass.edu 服务器的报文段的序列号与时间关系图。你能否确定 TCP 的慢启动阶段的开始和结束位置，以及拥塞避免接管的位置？





红框是 TCP 的慢启动阶段，之后是拥塞避免接管的位置。

2. 评论测量数据与我们在书本中研究的 TCP 的理想化行为的不同之处。

在 TCP 的理想化行为中，拥塞窗口应该会增大，但是在测量数据中，拥塞窗口保持不变。