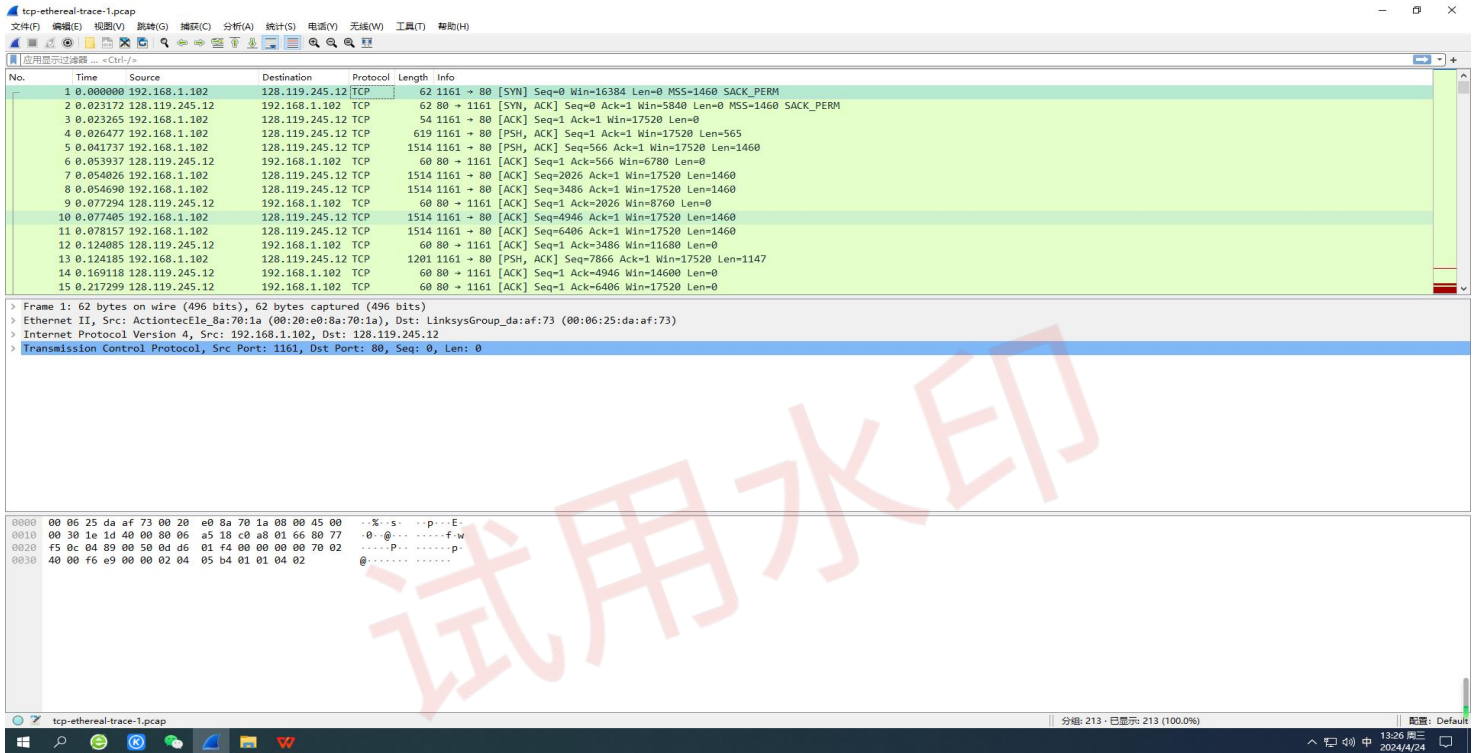


1. TCP_LAB

问题回答：

1. 源使用的 IP 地址和TCP 端口号是什么？

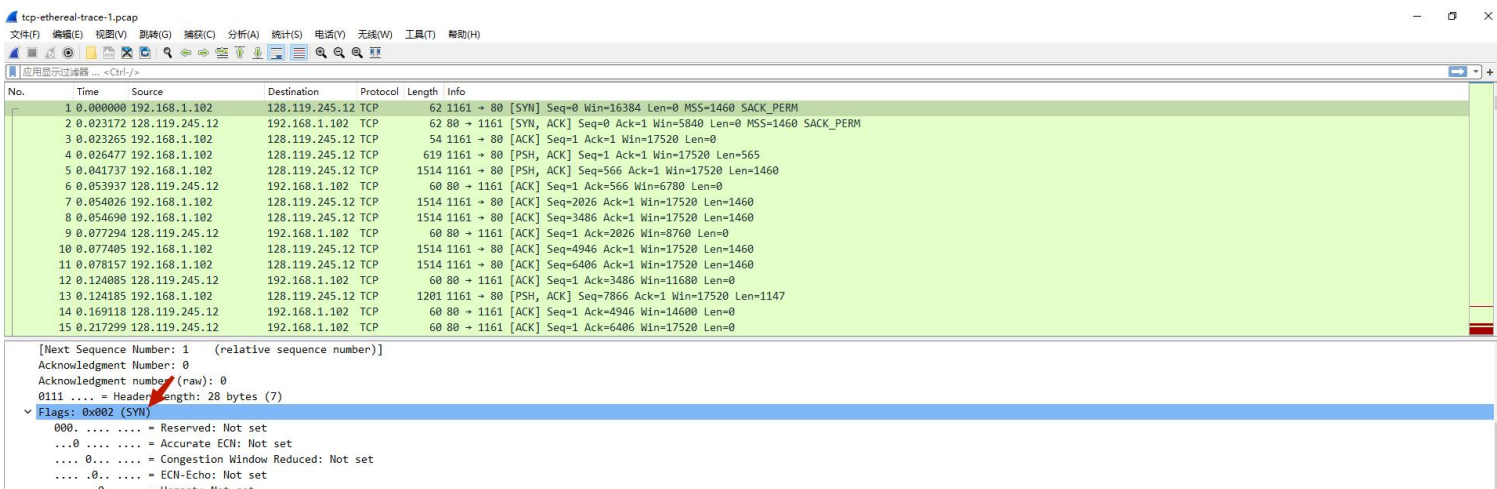
192.168.1.102 端口1161



2. gai a. 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 区段？

128.119.254.32 端口80

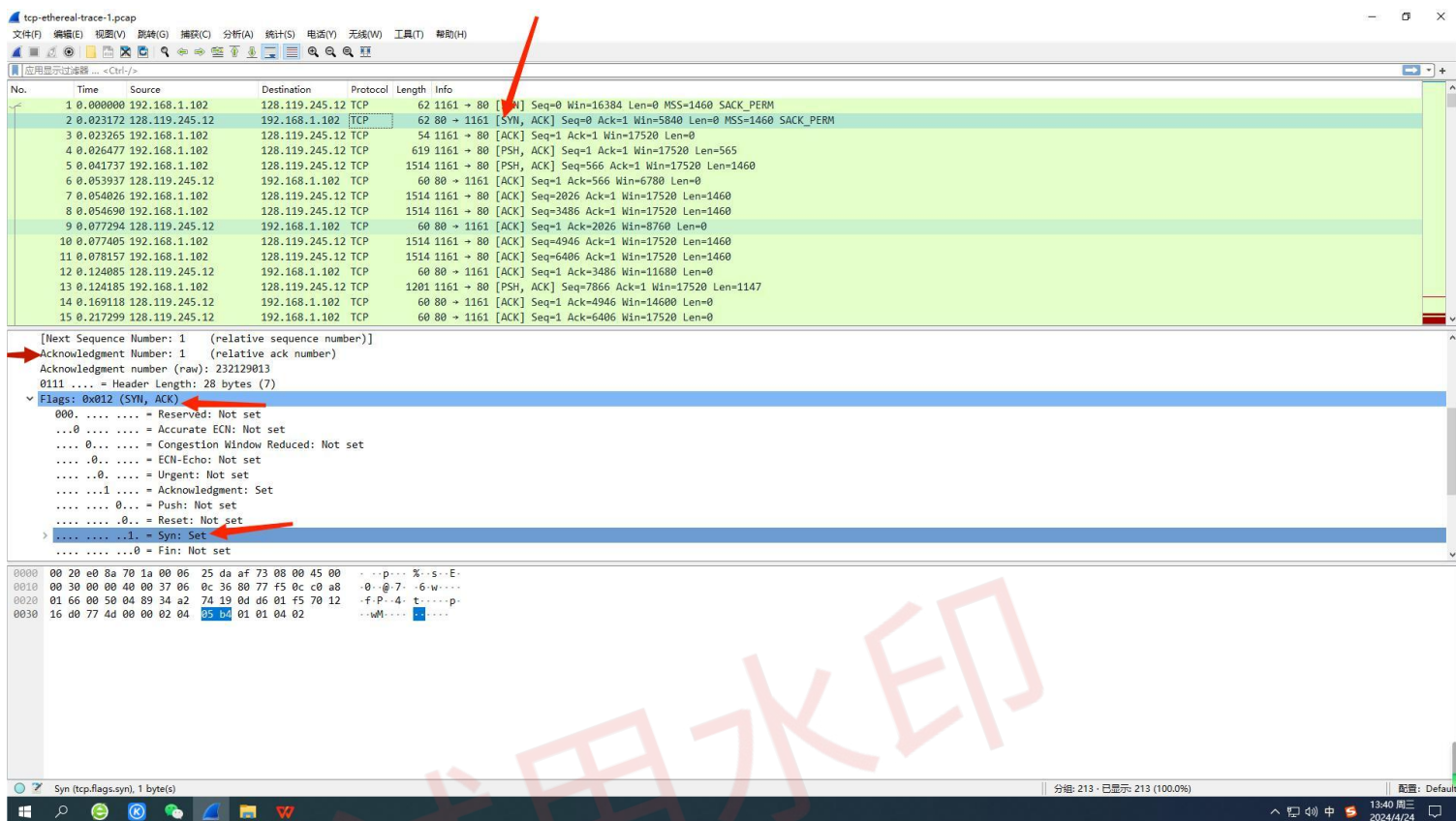
3. 用于在客户端计算机和 `gai a.cs.umass.edu` 之间启动 TCP 连接的 TCP SYN 区段的序列号是什么？将区段标识为 SYN 区段的区段有什么功能？



syn区段功能是实现tcp连接的三次握手

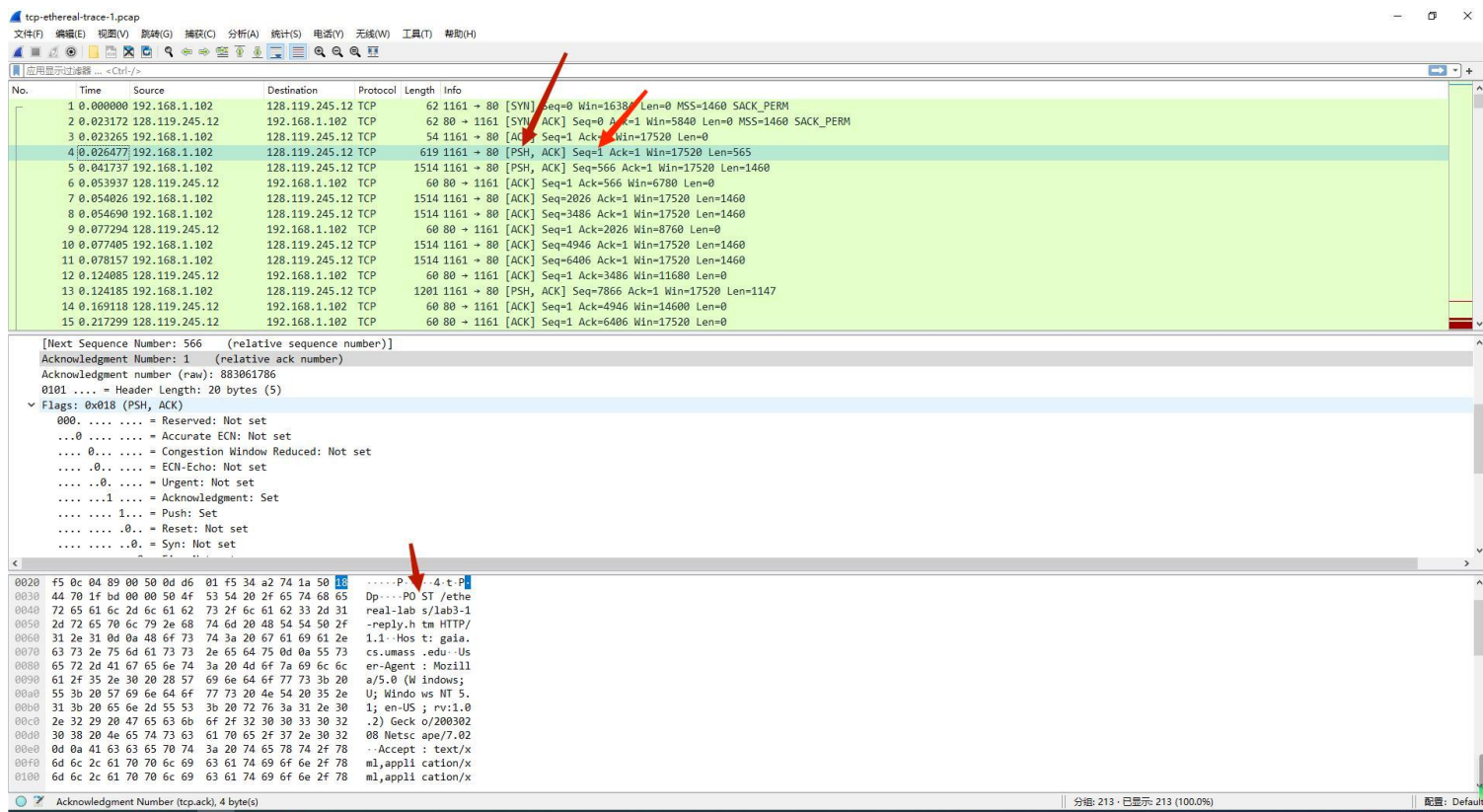
4. gai a. cs. umass. edu 发送给客户端计算机以回复 SYN 的 SYNACK 区段的序列号是多少？SYNACK 区段中的Acknowledgment 栏位的值是多少？

Gai a. cs. umass. edu 是如何确定此 Acknowledgment 的数值的，在将区段标识为SYNACK区段的区段在连线中有什么功能？



gai a. cs. umass. edu 是根据源发起TCP连接的tcp分组seq字段+1得来的Acknowledgment
SYNACK区段是完成三次握手中的第二步，表示服务器已接受源发起TCP连接

5. 包含 HTTP POST 命令的 TCP 区段的序列号是多少？



6. 将包含 HTTP POST 的 TCP 区段视为 TCP 连接中的第一个区段。在这个 TCP 连线中前六个 TCP 区段的序列号是什么（包括包含 HTTP POST 的区段）？每区段发送的时间是什么时候？收到的每个区段的 ACK 是什么时候？鉴于发送每个 TCP 区段的时间与收到确认的时间之间的差异，六个区段中每个区段的 RTT 值是多少？收到每个 ACK 后，EstimatedRTT 值（参见本节中的第 3.5.3 节，第 242 页）是什么？假设第一个 EstimatedRTT 的值等于第一个区段的测量 RTT，然后使用课本第 242 页的 EstimatedRTT 公式

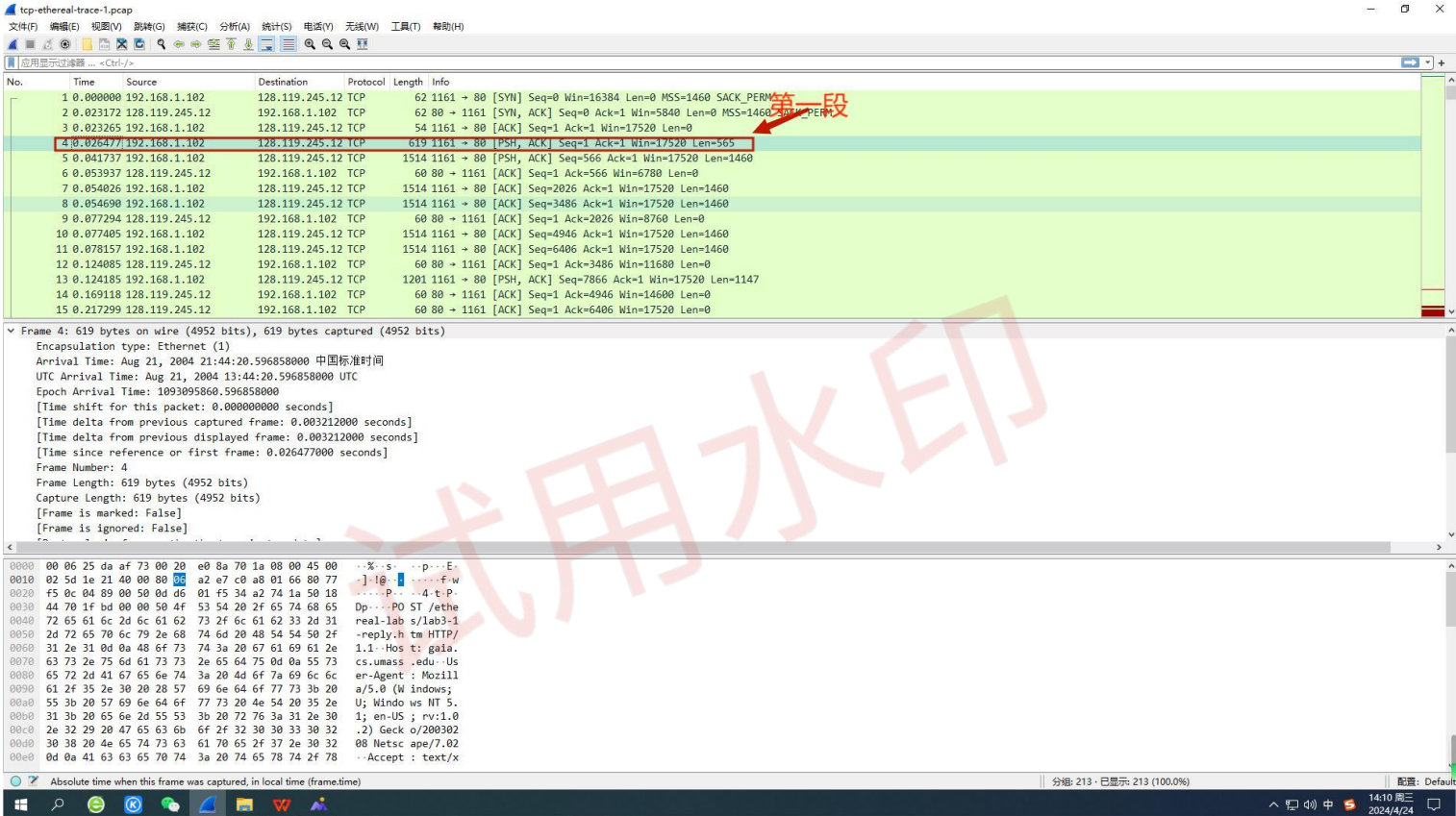
第一段

序号：1

发送时间：Aug 21, 2004 21:44:20.596858000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.624318000 中国标准时间
RTT：0.02746

EstimatedRTT: (1-)EstimatedRTT+ *sampl eRTT=0.02746



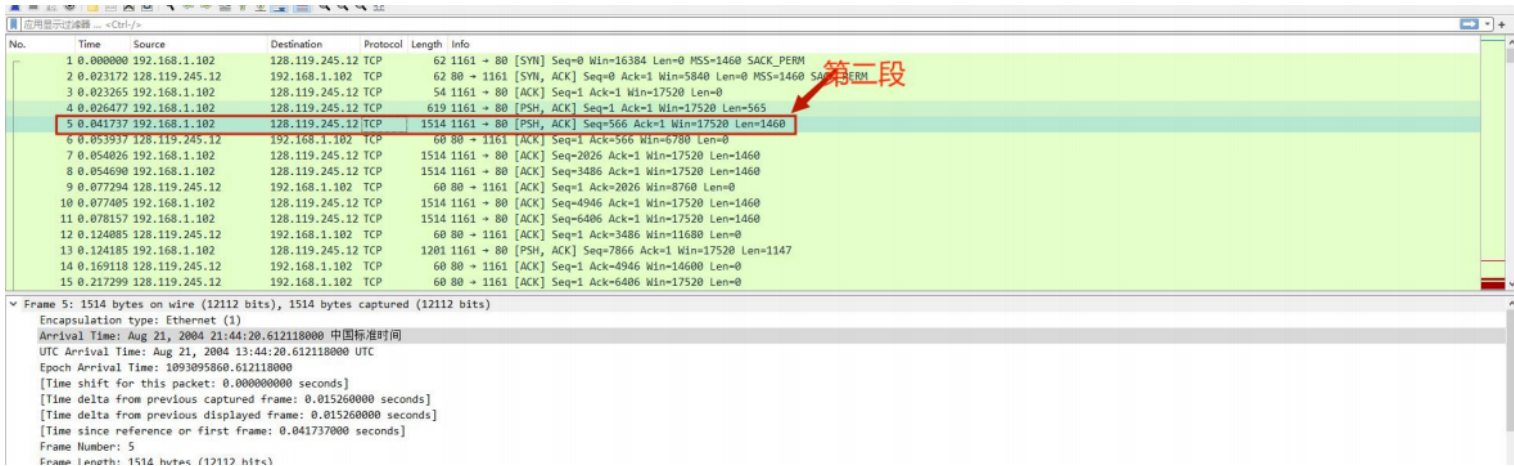
第二段

序号：566

发送时间：Aug 21, 2004 21:44:20.612118000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.647675000 中国标准时间
RTT：0.03556

EstimatedRTT: (1-)EstimatedRTT+ *sampl eRTT=0.0671045



第三段

序号：2026

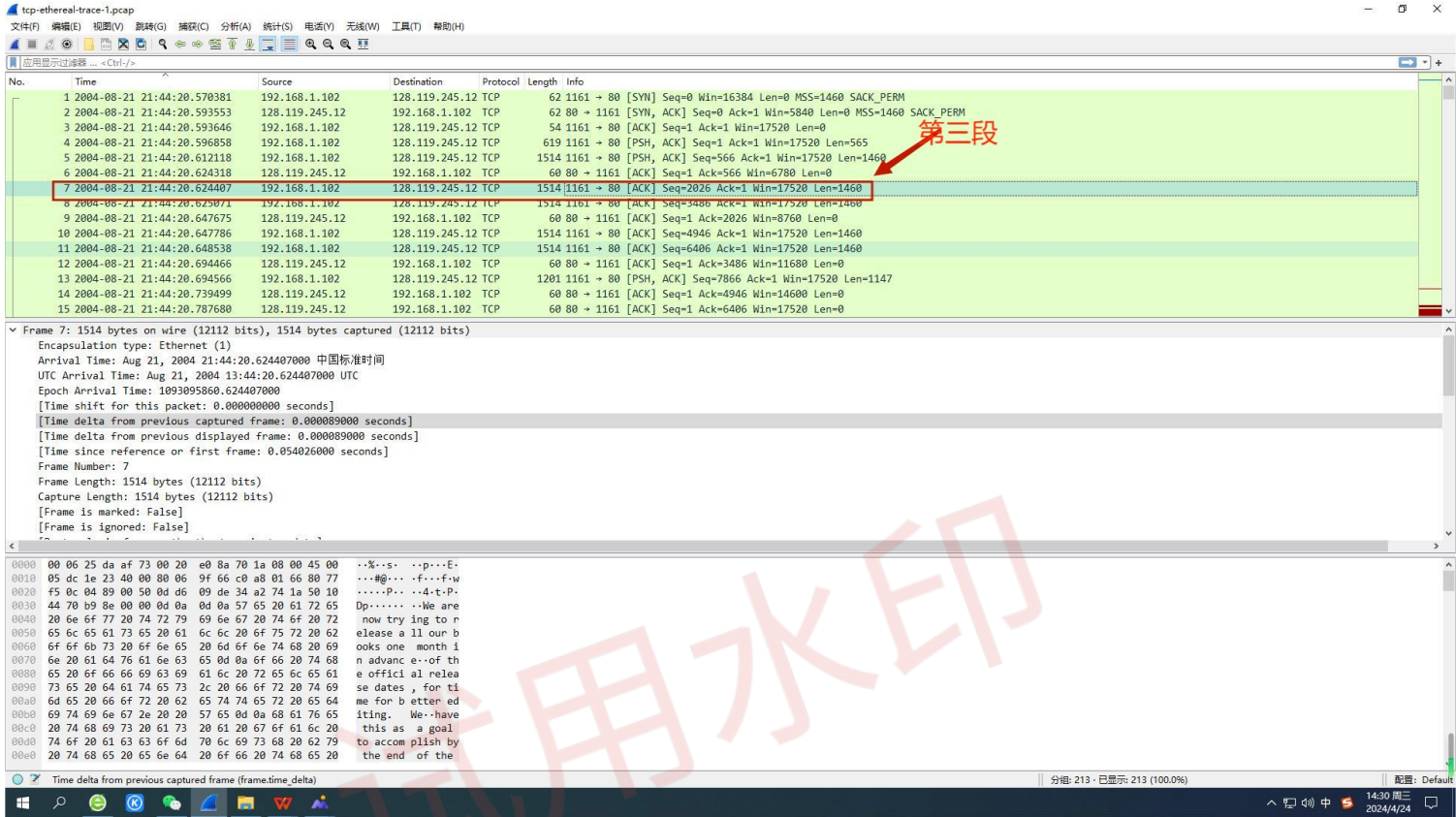
长度：1460

发送时间：Aug 21, 2004 21:44:20.624407000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.694466000 中国标准时间

RTT：0.07006

EstimatedRTT: (1-)EstimatedRTT+ *sampl eRTT=0.0641187125



第四段

序号：3486

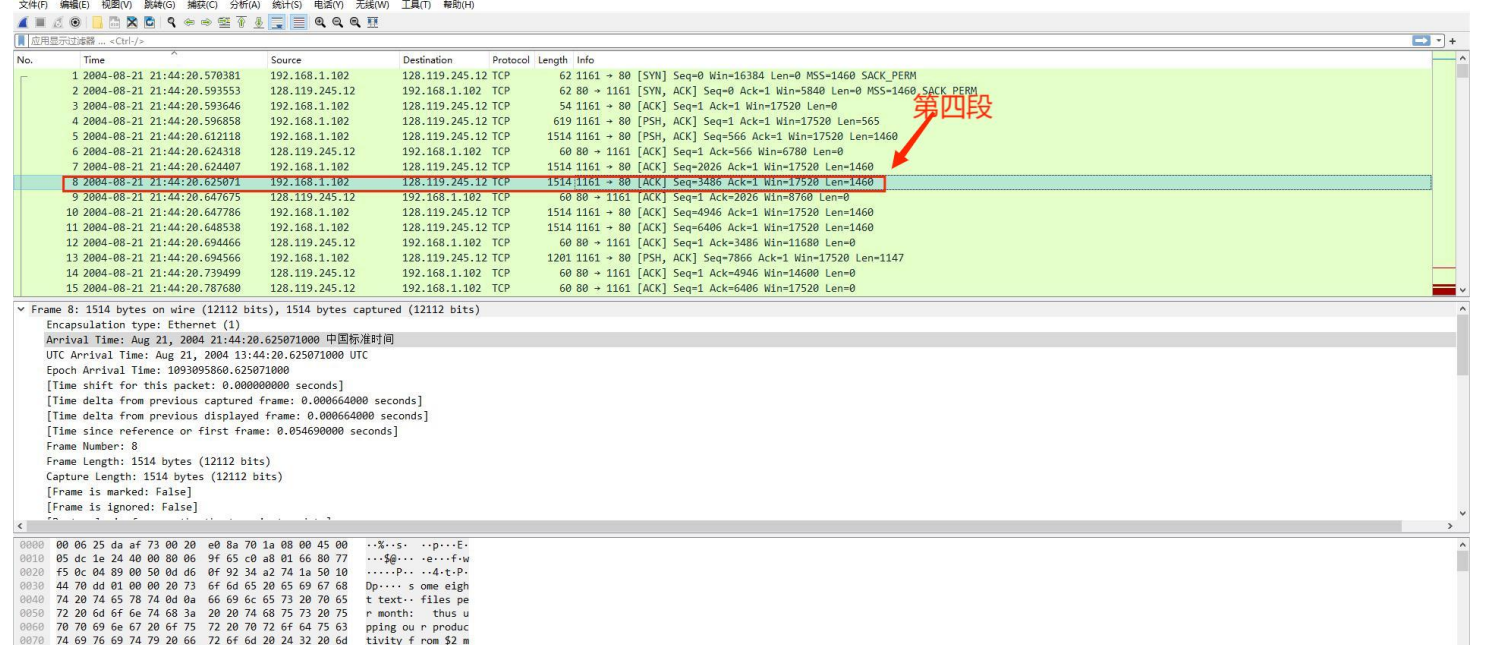
长度：1460

发送时间：Aug 21, 2004 21:44:20.625071000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.739499000 中国标准时间

RTT：0.11442

EstimatedRTT: (1-)EstimatedRTT+ *sampl eRTT=0.06720043781



第五段

序号：4946

长度：1460

发送时间：Aug 21, 2004 21:44:20.647786000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.787680000 中国标准时间

RTT：0.1399

EstimatedRTT: (1-)EstimatedRTT+ *sampleRTT=0.07292786119

tcp-ethereal-trace-1.pcap

文件(F) 编辑(E) 视图(V) 跟踪(G) 捕获(C) 分析(A) 统计(S) 电话(V) 无线(W) 工具(T) 帮助(H)

应用显示过滤器: <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
1	2004-08-21 21:44:20.570381	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	2004-08-21 21:44:20.593553	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	2004-08-21 21:44:20.593666	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	2004-08-21 21:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
5	2004-08-21 21:44:20.612118	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
6	2004-08-21 21:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	2004-08-21 21:44:20.624407	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
8	2004-08-21 21:44:20.625071	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
9	2004-08-21 21:44:20.647675	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	2004-08-21 21:44:20.647786	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
11	2004-08-21 21:44:20.648538	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
12	2004-08-21 21:44:20.694466	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	2004-08-21 21:44:20.694566	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147
14	2004-08-21 21:44:20.739499	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	2004-08-21 21:44:20.787680	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0

Frame 10: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)

Encapsulation type: Ethernet (1)

Arrival Time: Aug 21, 2004 21:44:20.647786000 中国标准时间

UTC Arrival Time: Aug 21, 2004 13:44:20.647786000 UTC

Epoch Arrival Time: 1093095860.647786000

[Time shift for this packet: 0.000000000 seconds]

[Time delta from previous captured frame: 0.000111000 seconds]

[Time delta from previous displayed frame: 0.000111000 seconds]

[Time since reference or first frame: 0.077405000 seconds]

Frame Number: 10

Frame Length: 1514 bytes (12112 bits)

Capture Length: 1514 bytes (12112 bits)

[Frame is marked: False]

[Frame is ignored: False]

0000 00 06 25 da af 73 00 20 e0 8a 70 1a 08 00 45 00 ...s...p...E-
0010 05 dc 1e 25 40 00 80 06 9f 64 c0 a8 01 66 80 77 ...&@...c...f-w
0020 f5 0c 04 89 00 50 0d d6 15 46 3a a2 74 1a 50 10P...4-t-P-
0030 44 70 95 83 00 00 20 55 6a 69 74 65 64 20 53 74 Dp... United St
0040 61 74 65 73 20 63 6f 70 79 72 69 67 68 74 0d 0a ates cop yright..
0050 6f 6e 20 6f 72 20 66 6f 72 20 68 69 73 20 77 on or fo r this w
0060 6f 72 6b 2c 20 73 6f 20 74 68 65 20 50 72 6f 6a ork, so the Proj
0070 65 63 74 20 28 61 6e 64 20 79 6f 75 21 29 20 63 ect (and you) c
0080 61 6e 20 63 6f 70 79 20 61 6e 64 0d 0a 64 69 73 an copy and-dis
0090 74 72 69 62 75 74 65 20 69 74 20 69 6e 20 74 68 tribute it in th
00a0 65 20 55 6e 69 74 65 64 20 53 74 61 74 65 73 20 e United States
00b0 77 69 74 68 6f 75 74 20 70 65 72 6d 69 73 73 69 without permissi
00c0 6f 6e 20 61 6e 64 0d 0a 77 69 74 68 6f 75 74 20 on and.. without
00d0 70 61 79 69 6e 67 20 63 6f 70 79 72 69 67 68 74 payin c opyright
00e0 20 72 6f 79 61 6c 74 69 65 73 2e 20 53 70 75 65 royalti es. Spe

Absolute time when this frame was captured, in local time (frame.time)

分帧: 213 / 已显示: 213 (100.0%)

配置: Defau

14:48 周三 2024/4/24

第六段

序号：6406

长度：1460

发送时间：Aug 21, 2004 21:44:20.648538000 中国标准时间

收到ACK时间：Aug 21, 2004 21:44:20.694566000 中国标准时间

RTT：0.04603

EstimatedRTT: (1-)EstimatedRTT+ *sampleRTT=0.06591923548

tcp-ethereal-trace-1.pcap

文件(F) 编辑(E) 视图(V) 跟踪(G) 捕获(C) 分析(A) 统计(S) 电话(V) 无线(W) 工具(T) 帮助(H)

应用显示过滤器: <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
1	2004-08-21 21:44:20.570381	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	2004-08-21 21:44:20.593553	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	2004-08-21 21:44:20.593666	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	2004-08-21 21:44:20.596858	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565
5	2004-08-21 21:44:20.612118	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460
6	2004-08-21 21:44:20.624318	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	2004-08-21 21:44:20.624407	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460
8	2004-08-21 21:44:20.625071	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460
9	2004-08-21 21:44:20.647675	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	2004-08-21 21:44:20.647786	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460
11	2004-08-21 21:44:20.648538	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460
12	2004-08-21 21:44:20.694466	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	2004-08-21 21:44:20.694566	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147
14	2004-08-21 21:44:20.739499	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	2004-08-21 21:44:20.787680	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0

Frame 11: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)

Encapsulation type: Ethernet (1)

Arrival Time: Aug 21, 2004 21:44:20.648538000 中国标准时间

UTC Arrival Time: Aug 21, 2004 13:44:20.648538000 UTC

Epoch Arrival Time: 1093095860.648538000

[Time shift for this packet: 0.000000000 seconds]

[Time delta from previous captured frame: 0.000752000 seconds]

[Time delta from previous displayed frame: 0.000752000 seconds]

[Time since reference or first frame: 0.078157000 seconds]

Frame Number: 11

Frame Length: 1514 bytes (12112 bits)

Capture Length: 1514 bytes (12112 bits)

[Frame is marked: False]

[Frame is ignored: False]

0000 00 06 25 da af 73 00 20 e0 8a 70 1a 08 00 45 00 ...s...p...E-
0010 05 dc 1e 26 40 00 80 06 9f 63 c0 a8 01 66 80 77 ...&@...c...f-w
0020 f5 0c 04 89 00 50 0d d6 1a fa 3a a2 74 1a 50 10P...4-t-P-
0030 44 70 95 83 00 00 20 55 6a 69 74 65 64 20 53 74 Dp... United St
0040 61 74 65 73 20 63 6f 70 79 72 69 67 68 74 0d 0a ates cop yright..
0050 6f 6e 20 6f 72 20 66 6f 72 20 68 69 73 20 77 on or fo r this w
0060 6f 72 6b 2c 20 73 6f 20 74 68 65 20 50 72 6f 6a ork, so the Proj
0070 65 63 74 20 28 61 6e 64 20 79 6f 75 21 29 20 63 ect (and you) c
0080 61 6e 20 63 6f 70 79 20 61 6e 64 0d 0a 64 69 73 an copy and-dis
0090 74 72 69 62 75 74 65 20 69 74 20 69 6e 20 74 68 tribute it in th
00a0 65 20 55 6e 69 74 65 64 20 53 74 61 74 65 73 20 e United States
00b0 77 69 74 68 6f 75 74 20 70 65 72 6d 69 73 73 69 without permissi
00c0 6f 6e 20 61 6e 64 0d 0a 77 69 74 68 6f 75 74 20 on and.. without
00d0 70 61 79 69 6e 67 20 63 6f 70 79 72 69 67 68 74 payin c opyright
00e0 20 72 6f 79 61 6c 74 69 65 73 2e 20 53 70 75 65 royalti es. Spe

Absolute time when this frame was captured, in local time (frame.time)

分帧: 213 / 已显示: 213 (100.0%)

配置: Defau

14:51 周三 2024/4/24

7. 对于整个跟踪包，收到的最小可用缓冲区空间量是多少？缺少接收器缓冲区空间是否会限制发送方传送 TCP 区段？

最小的缓存空间5840，当win=0时，发送方会定时发送一个字节数据的分组询问接收方win的值

8. 在跟踪文件中是否有重传的区段？为了回答这个问题，您检查了什么（在跟踪包中）？

此次实验未发现有重传，通过检查是否有重复的seq和ACK字段

9. 接收器通常在 ACK 中确认多少数据？您是否可以识别接收方每隔一个接收到的区段才发送确认的情况

接收方ACK的值是接受到分组中数据的首部字节编号，TCP使用的是延迟累积确认，TCP标准规定，最大延迟确认时间不应该超过0.5秒，如果收到连续的最大一串数据的分组，必须每隔一个分组发送ACK。延迟累积确认的好处是可以降低网络流量，不需要每个收到分组发送ACK。

10. 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。

可以通过吞吐量=总传输数据量/总传输时间计算

11. 使用时序图（Stevens）绘图工具查看从客户端发送到 gai a.cs.umass.edu 服务器的区段的序列号与时间关系图。您能否确定 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？评论测量数据与我们在文本中研究的 TCP 的理想化行为的不同之处。



结束位置无法确定，因为没发现慢启动的转折点（发生丢包）。

与书本上理想情况不同点：

1. 慢启动开始时Congwin不是一个MSS而是2个
2. 此次实验中收到ACK后没有指数增加Congwin，而是在一定时间保持2个MSS的窗口长度
3. 后续也没有一直增加Congwin，而是保持6个MSS的长度