

CS305 Lab7

Name: 胡玉斌

Student Id: 11712121

1. Select one UDP packet from your trace. From this packet,determine

- 1. how many fields there are in the UDP header.

Solution:

There are 4 fields in the UDP header

```
eveneko@DESKTOP-MMVJRV3 /mnt/c/Users/Eveneko dig @8.8.8.8 www.google.com

; <<>> DiG 9.11.3-1ubuntu1.9-Ubuntu <<>> @8.8.8.8 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37231
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.google.com.                IN      A

;; ANSWER SECTION:
www.google.com.                104     IN      A      31.13.64.1

;; Query time: 7 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Thu Oct 31 17:35:58 DST 2019
;; MSG SIZE rcvd: 48
```

No.	Time	Source	Destination	Protocol	Length	Info
16	2.183778	10.21.63.252	8.8.8.8	DNS	97	Standard query 0x916f A www.google.com OPT
17	2.190952	8.8.8.8	10.21.63.252	DNS	90	Standard query response 0x916f A www.google.com A 31.13.64.1
18	2.190953	8.8.8.8	10.21.63.252	DNS	90	Standard query response 0x916f A www.google.com A 31.13.77.55
20	2.243766	8.8.8.8	10.21.63.252	DNS	1...	Standard query response 0x916f A www.google.com A 172.217.160.100 OPT
495	25.820369	10.21.63.252	172.18.1.92	DNS	83	Standard query 0x5aba AAAA interface.music.163.com
496	25.823085	172.18.1.92	10.21.63.252	DNS	1...	Standard query response 0x5aba AAAA interface.music.163.com SOA ns4.nease.net
531	26.010943	10.21.63.252	172.18.1.92	DNS	76	Standard query 0x19c9 A p4.music.126.net
532	26.011258	10.21.63.252	172.18.1.92	DNS	76	Standard query 0xd9e5 AAAA p4.music.126.net
533	26.015809	172.18.1.92	10.21.63.252	DNS	3...	Standard query response 0x19c9 A p4.music.126.net CNAME p4.music.126.net.163jiasu.com CNAME p4.music.12...
534	26.015810	172.18.1.92	10.21.63.252	DNS	3...	Standard query response 0xd9e5 AAAA p4.music.126.net CNAME p4.music.126.net.163jiasu.com CNAME p4.music...
546	26.178335	10.21.63.252	172.18.1.92	DNS	76	Standard query 0x7c3e A m7.music.126.net
547	26.180141	172.18.1.92	10.21.63.252	DNS	5...	Standard query response 0x7c3e A m7.music.126.net CNAME m7.music.126.net.163jiasu.com CNAME m7.music.12...
1730	26.828801	10.21.63.252	172.18.1.92	DNS	73	Standard query 0xa595 AAAA music.163.com
1732	26.831034	172.18.1.92	10.21.63.252	DNS	1...	Standard query response 0xa595 AAAA music.163.com SOA ns4.nease.net
1849	26.955771	10.21.63.252	172.18.1.92	DNS	76	Standard query 0x74c7 A p3.music.126.net
1850	26.955950	10.21.63.252	172.18.1.92	DNS	76	Standard query 0xb1ca AAAA p3.music.126.net
1851	26.959222	172.18.1.92	10.21.63.252	DNS	3...	Standard query response 0x74c7 A p3.music.126.net CNAME p3.music.126.net.163jiasu.com CNAME p3.music.12...
1852	26.959222	172.18.1.92	10.21.63.252	DNS	3...	Standard query response 0xb1ca AAAA p3.music.126.net CNAME p3.music.126.net.163jiasu.com CNAME p3.music...

User Datagram Protocol, Src Port: 51924, Dst Port: 53 Source Port: 51924 Destination Port: 53 Length: 63 Checksum: 0xf851 [unverified] [Checksum Status: Unverified] [Stream index: 3] [Timestamps]	Domain Name System (query) Transaction ID: 0x916f Flags: 0x0120 Standard query Questions: 1
---	---

0000	40 71 83 ab 30 03 bc 83 85 eb 12 f8 08 00 45 00	@q...E..
0010	00 53 05 80 00 00 80 11 da f9 0a 15 3f fc 08 08	.S.....?...
0020	08 08 ca d4 00 35 00 3f f8 51 91 6f 01 20 00 01	...5.?..Q.o..
0030	00 00 00 00 00 01 03 77 77 77 06 67 6f 6f 67 6cw ww.googl
0040	65 03 63 6f 6d 00 00 01 00 01 00 00 29 10 00 00	e.com.....)
0050	00 00 00 00 0c 00 0a 00 08 61 14 d2 70 e1 6e 96a.p.n.
0060	86	.

- 2. the name of each fields in the UDP header.

Solution:

There are

- Source Port
- Destination
- Length
- Checksum

- 3. the length (in bytes) of each fields in the UDP header.

Solution:

The length of UDP header is 8 bytes. So each fields in the UDP header is 2 bytes.

> Frame 16: 97 bytes on wire (776 bits), 97 bytes captured (776 bits) on interface 0
> Ethernet II, Src: Microsof_eb:12:f8 (bc:83:85:eb:12:f8), Dst: JuniperN_ab:30:03 (40:71:83:ab:30:03)
> Internet Protocol Version 4, Src: 10.21.63.252, Dst: 8.8.8.8
> User Datagram Protocol, Src Port: 51924, Dst Port: 53
Source Port: 51924
Destination Port: 53
Length: 63
Checksum: 0xf851 [unverified]
[Checksum Status: Unverified]
[Stream index: 3]
> [Timestamps]
> Domain Name System (query)

0010	00 53 05 80 00 00 80 11 da f9 0a 15 3f fc 08 08	.S.....?...
0020	08 08 ca d4 00 35 00 3f f8 51 91 6f 01 20 00 01	...5.?..Q.o..
0030	00 00 00 00 00 01 03 77 77 77 06 67 6f 6f 67 6cw ww.googl
0040	65 03 63 6f 6d 00 00 01 00 01 00 00 29 10 00 00	e.com.....)
0050	00 00 00 00 0c 00 0a 00 08 61 14 d2 70 e1 6e 96a.p.n.
0060	86	.

User Datagram Protocol (udp), 8 字节

- 4. What is the maximum number of bytes that can be included in a UDP payload? (Hint: the answer to this question can be determined by your answer to 3) above)

Solution:

There 2 bytes to record the length. Also the length of UDP header is 8 bytes, and the length of IP header is 20 bytes. The maximum number of bytes is $2^{16} - 1 - 8 - 20 = 65507$

- 5. What is the largest possible source port number? (Hint: same as the hint in 4) above)

Solution:

The largest possible source port number is $2^{16} - 1 = 65535$

- 6. What is the protocol number for UDP? (Give your answer in both hexadecimal and decimal notation.)

Solution:

The IP protocol number for UDP is 0x11 hex, which is 17 in decimal value.

```
> Frame 16: 97 bytes on wire (776 bits), 97 bytes captured (776 bits) on interface 0
> Ethernet II, Src: Microsof_eb:12:f8 (bc:83:85:eb:12:f8), Dst: JuniperN_ab:30:03 (40:71:83:ab:30:03)
✓ Internet Protocol Version 4, Src: 10.21.63.252, Dst: 8.8.8.8
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 83
  Identification: 0x0580 (1408)
  > Flags: 0x0000
  Time to live: 128
  Protocol: UDP (17)
  Header checksum: 0xdaf9 [validation disabled]
  [Header checksum status: Unverified]
  Source: 10.21.63.252
  Destination: 8.8.8.8
```

0000	40 71 83 ab 30 03 bc 83	85 eb 12 f8 08 00 45 00	@q..0... ..E.
0010	00 53 05 80 00 00 80 11	da f9 0a 15 3f fc 08 08	.S.....?...
0020	08 08 ca d4 00 35 00 3f	f8 51 91 6f 01 20 00 015? .Q.o. .
0030	00 00 00 00 00 01 03 77	77 77 06 67 6f 6f 67 6cw ww.googl
0040	65 03 63 6f 6d 00 00 01	00 01 00 00 29 10 00 00	e.com... ..)
0050	00 00 00 00 0c 00 0a 00	08 61 14 d2 70 e1 6e 96a.p.n.
0060	86		.

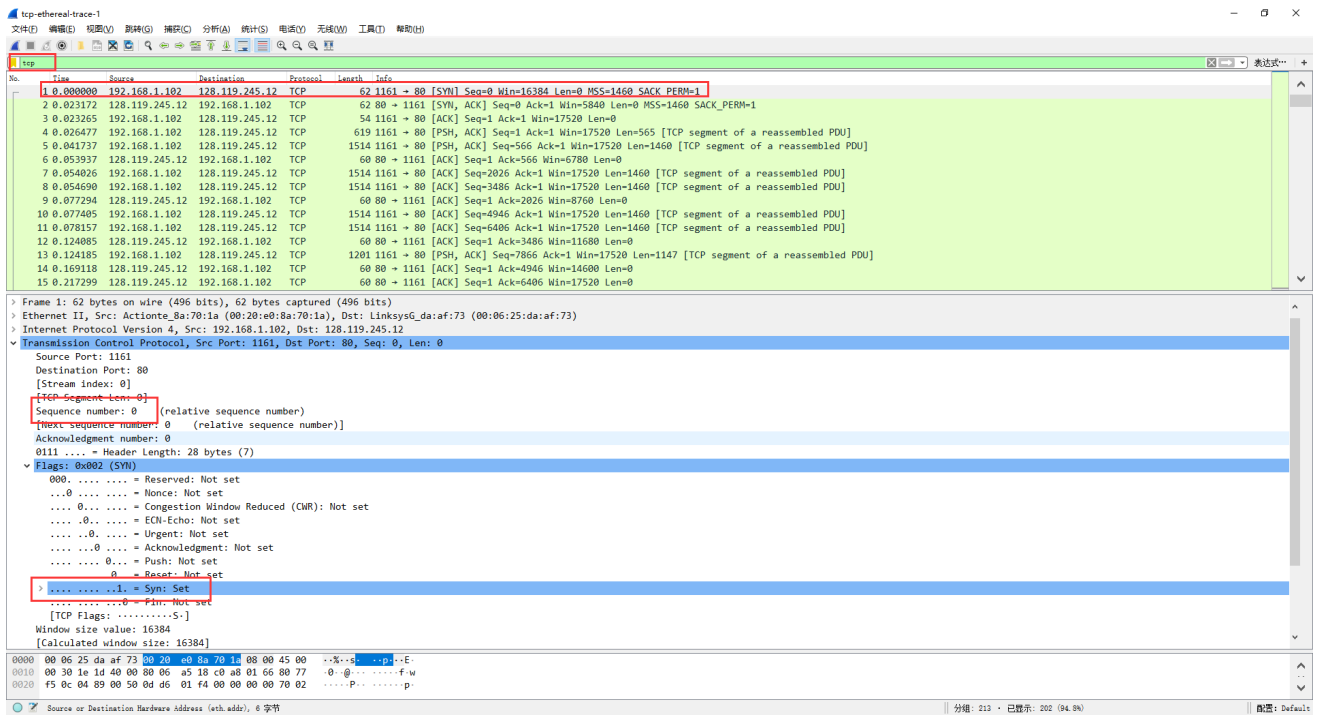
2. Wireshark Lab: TCP v7.0

- 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?

Solution:

Sequence number of the TCP SYN segment is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu. The value is 0.

The SYN segment flag is set 1 in the segment that identifies the segment as a SYN segment.

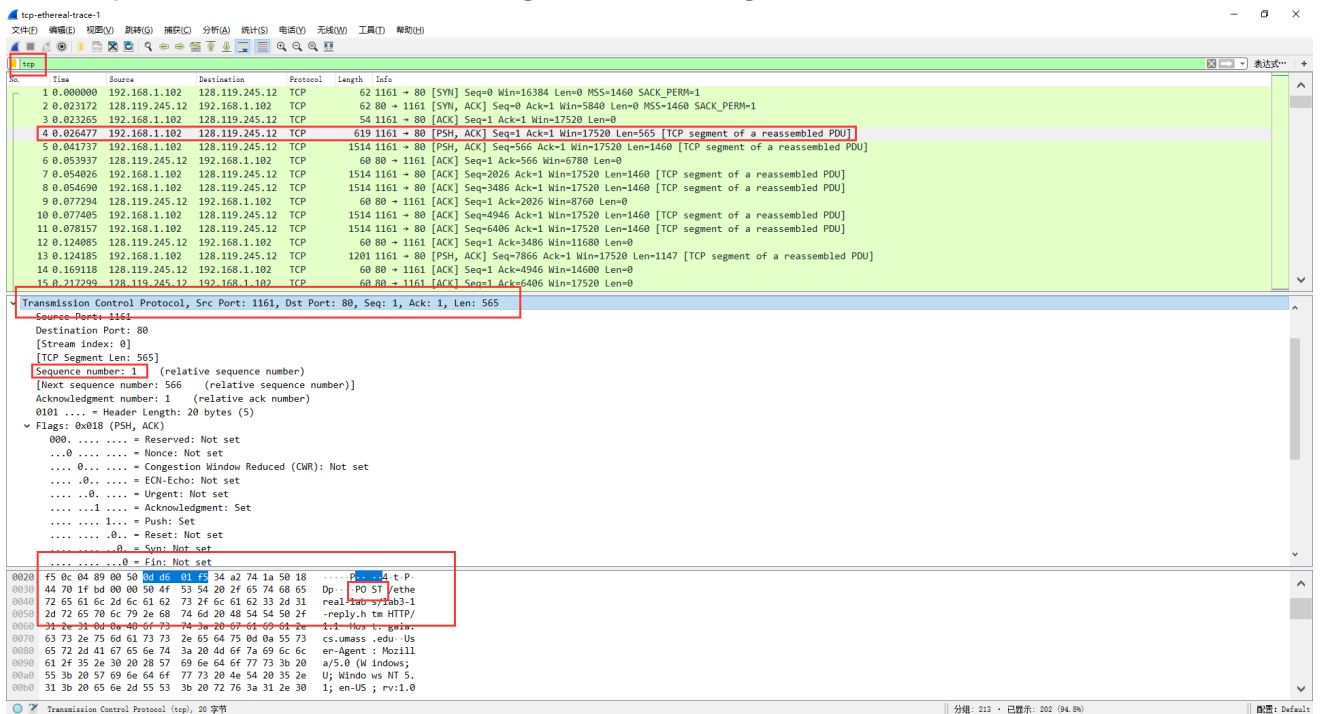


6. What is the sequence number of the TCP segment containing the HTTP POST command?
- Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

Solution:

The 4th segment is the TCP segment containing the HTTP POST command.

The sequence number of the TCP segment containing the HTTP POST command is 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? 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.

Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the “listing of captured packets” window that is being sent from the client to the gaia.cs.umass.edu server. Then select: Statistics->TCP Stream Graph->Round Trip Time Graph.

Solution:

The first six segments in the TCP connection is 4th, 5th, 7th, 8th, 10th, 11th.

	Segment 1	Segment 2	Segment 3	Segment 4	Segment 5	Segment 6
sequence number	1	566	2026	3486	4946	6406
send time	0.026477	0.041737	0.054026	0.054690	0.077405	0.078157
ACK receive time	0.053937	0.077294	0.124085	0.169118	0.217299	0.267802
RTT	0.02746	0.035557	0.070059	0.11443	0.13989	0.18964
EstimatedRTT	0.02746	0.0285	0.0337	0.0438	0.0558	0.0725

$$\text{EstimatedRTT} = 0.875 * \text{EstimatedRTT} + 0.125 * \text{SampleRTT}$$

$$\text{EstimatedRTT}_1 = \text{RRT}_1 = 0.02746 \text{ s}$$

$$\text{EstimatedRTT}_2 = 0.875 * \text{EstimatedRTT}_1 + 0.125 * \text{RTT}_2 = 0.875 * 0.02746 + 0.125 * 0.035557 \approx 0.0285 \text{ s}$$

$$\text{EstimatedRTT}_3 = 0.875 * \text{EstimatedRTT}_2 + 0.125 * \text{RTT}_3 = 0.875 * 0.0285 + 0.125 * 0.070059 \approx 0.0337 \text{ s}$$

$$\text{EstimatedRTT}_4 = 0.875 * \text{EstimatedRTT}_3 + 0.125 * \text{RTT}_4 = 0.875 * 0.0337 + 0.125 * 0.11442 \approx 0.0438 \text{ s}$$

$$\text{EstimatedRTT}_5 = 0.875 * \text{EstimatedRTT}_4 + 0.125 * \text{RTT}_5 = 0.875 * 0.0438 + 0.125 * 0.13989 \approx 0.0558 \text{ s}$$

$$\text{EstimatedRTT}_6 = 0.875 * \text{EstimatedRTT}_5 + 0.125 * \text{RTT}_6 = 0.875 * 0.0558 + 0.125 * 0.18964 \approx 0.0725 \text{ s}$$

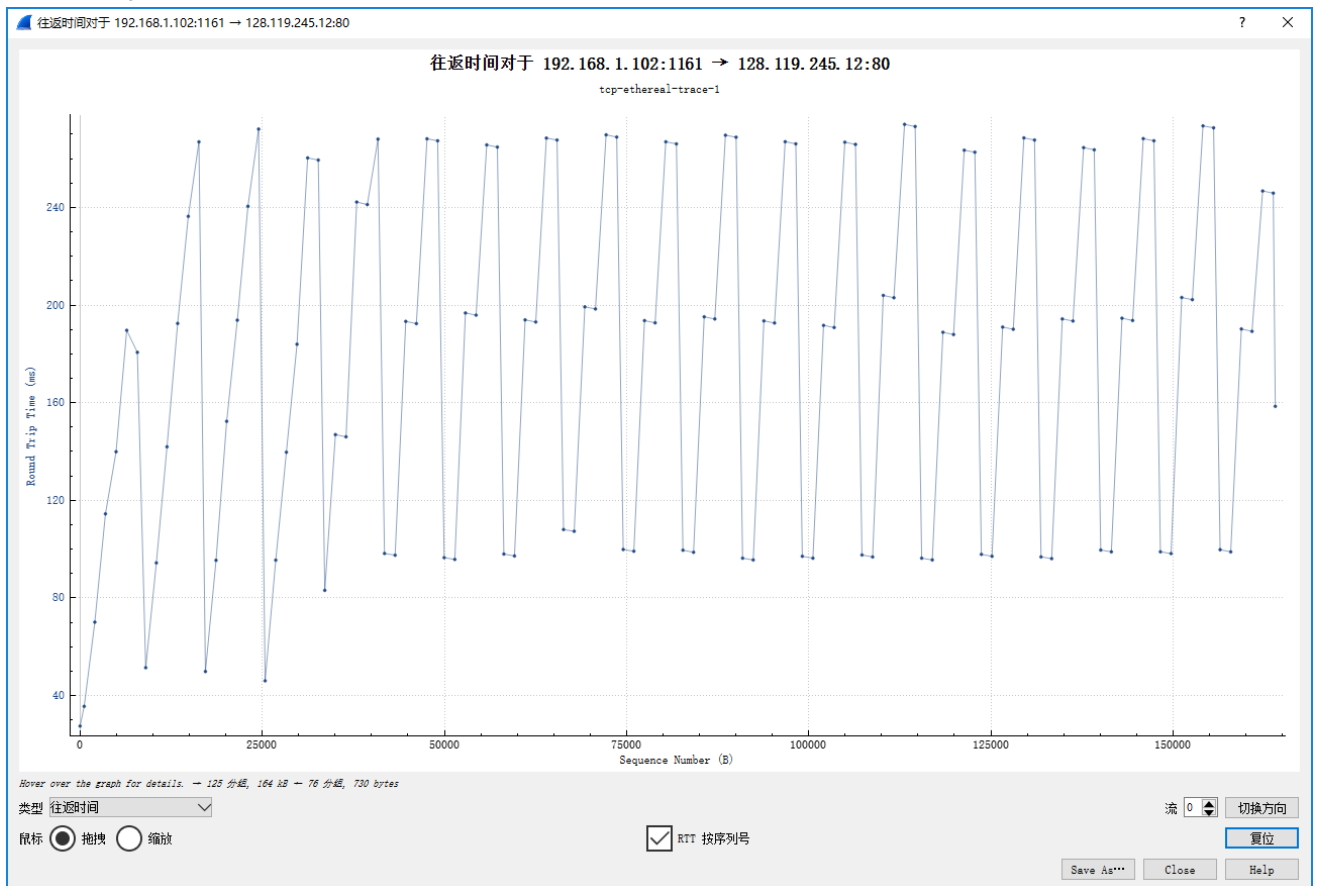
Segment 1-6:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	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=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
14	0.169118	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0

ACK 1-6:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	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=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
14	0.169118	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267882	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=9813 Win=23360 Len=0
18	0.304809	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=9813 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]

RTT Graph



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?

Solution:

The minimum amount of available buffer space at the receiver for the entire trace is 5840. It is showed in the first acknowledgement from the server. The server does not throttle the sender due to the lack of receiver buffer space. Because the window size is gradually increased from 5840 to 62780, the window size is always larger than the capacity of the packet sent by the sender.

tcp-ethereal-trace-1

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

tcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	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=1

... 0... .. = Congestion Window Reduced (CWR): Not set
0... .. = ECH-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....]
 Window size value: 5840
 [calculated window size: 5840]
 Checksum: 0x774d [unverified]
 [Checksum Status: Unverified]
 Urgent pointer: 0
 > Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
 > [SEQ/ACK analysis]
 > [Timestamps]

0000 00 20 e0 8a 70 1a 00 06 25 da af 73 08 00 45 00 ...p...X...s:E
 0010 00 30 00 00 00 37 06 0c 36 00 77 f5 0c c0 a8 ...@ 7...6w...
 0020 01 66 00 50 04 89 34 a2 74 19 0d d6 01 f5 70 12 f.p..4..t...p:
 0030 16 05 77 4d 00 00 02 04 05 b4 01 01 04 02 ...M.....

The window size value from the TCP header (tcp.window_size_value), 2 字节

分组: 213 · 已显示: 202 (94.8%)

配置: Default

tcp-ethereal-trace-1

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

tcp

No.	Time	Source	Destination	Protocol	Length	Info
37	0.672796	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=20125 Win=43800 Len=0
38	0.730684	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=21585 Win=46720 Len=0
39	0.772990	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=23045 Win=49640 Len=0
40	0.820622	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=24505 Win=52560 Len=0
41	0.853186	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=25397 Win=52560 Len=0
42	0.853405	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=25397 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
43	0.854076	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=26857 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
44	0.855936	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=28317 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
45	0.855978	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=29777 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
46	0.855802	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=31237 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
47	0.857683	128.119.245.12	192.168.1.102	TCP	946	1161 → 80 [PSH, ACK] Seq=32697 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
48	0.899423	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=26857 Win=55480 Len=0
49	0.940545	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=28317 Win=58400 Len=0
50	0.994715	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=29777 Win=61320 Len=0
51	1.035820	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=31237 Win=62780 Len=0
52	1.117097	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=33589 Win=62780 Len=0
53	1.117333	128.119.245.12	192.168.1.102	TCP	1514	1161 → 80 [ACK] Seq=33589 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]

... 0... .. = Congestion Window Reduced (CWR): Not set
0... .. = ECH-Echo: Not set
0... .. = Urgent: Not set
1... .. = Acknowledgment: Set
0... .. = Push: Not set
0... .. = Reset: Not set
0... .. = Syn: Not set
0... .. = Fin: Not set
 [TCP Flags:A....]
 Window size value: 62780
 [calculated window size: 62780]
 [window size scaling factor: ∞ (no window scaling used)]
 Checksum: 0x4ba0 [unverified]
 [Checksum Status: Unverified]
 Urgent pointer: 0
 > [SEQ/ACK analysis]
 > [Timestamps]

0000 00 20 e0 8a 70 1a 00 06 25 da af 73 08 00 45 00 ...p...X...s:E
 0010 00 28 58 88 40 00 37 06 b3 b5 80 77 f5 0c c0 a8 (X @ 7...w...
 0020 01 66 00 50 04 89 34 a2 74 19 0d d6 01 f5 70 12 f.p..4..t...p:
 0030 16 05 77 4d 00 00 11 0b 0b 00 cd fd ...M.....

The window size value from the TCP header (tcp.window_size_value), 2 字节

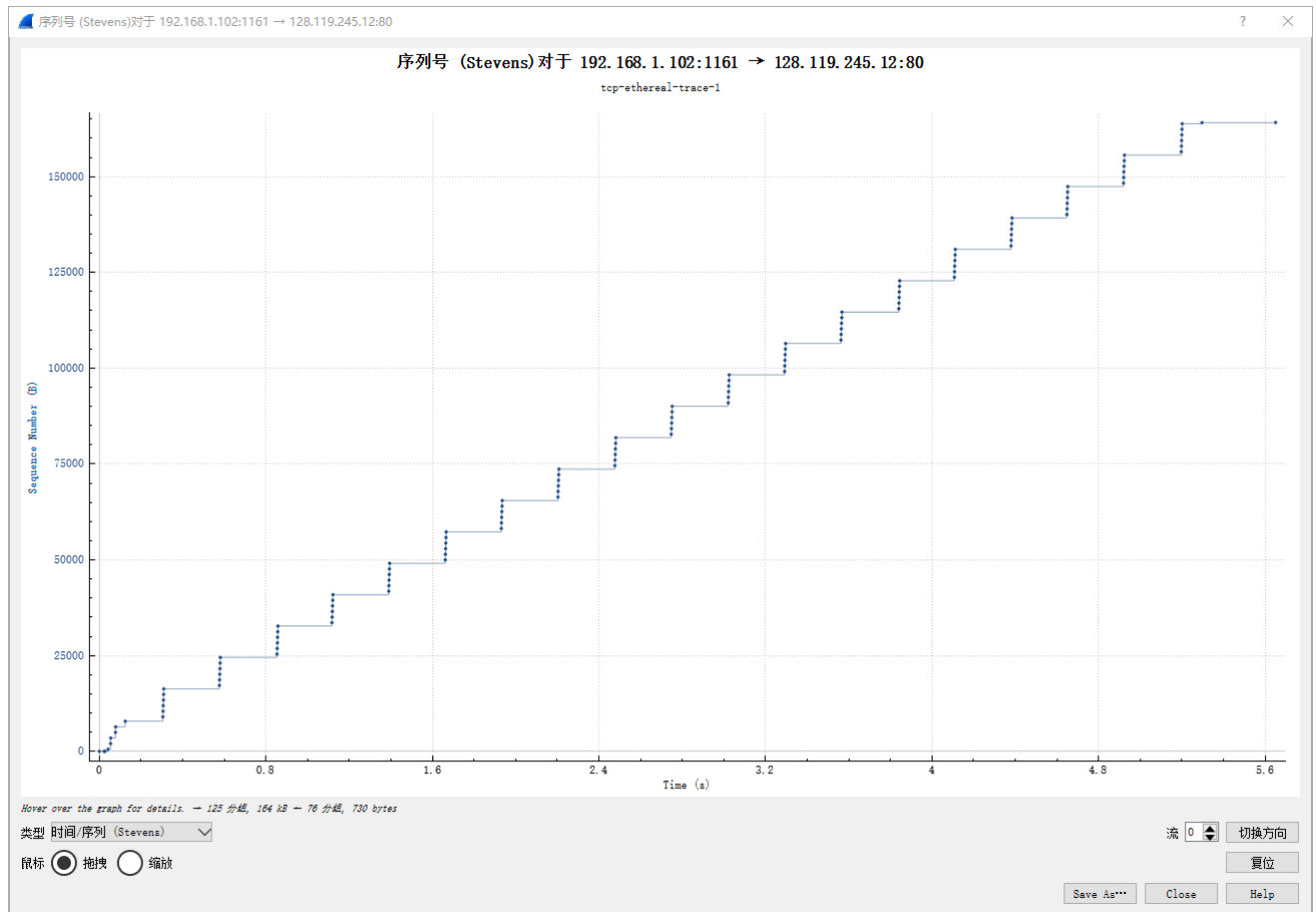
分组: 213 · 已显示: 202 (94.8%)

配置: Default

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

Solution:

No, no retransmitted segment. And the above conclusion can be drawn from the serial number of the TCP segment. It can be seen from the Time-Sequence-Graph (Stevens) that the sequence number sent from the source to the destination is gradually increased. If there is a retransmitted segment, the sequence number should have a packet smaller than its adjacent packet sequence number. Such a packet is not seen in the figure, so there is no packet that is retransmitted.



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

Solution:

The total amount of data transferred is the difference between the sequence number of the first TCP segment (1 byte of the 4th segment) and the ACK of the last sequence number (164091 bytes of the 202nd segment). Therefore, the total data is $164091 - 1 = 164090$ bytes. The whole transmission time is the difference of the time instant of the first TCP segment (0.026477 second for 4th segment) and the time instant of the last ACK (5.455830 second for 202th segment). Therefore, the total transmission time is $5.455830 - 0.026477 = 5.4294$ seconds.

Therefore, the throughput for TCP connection is the ratio between the total amount data and the total time. Throughput = $164090 \text{ bytes} / 5.4294 \text{ seconds} \approx 30222.49 \text{ Byte/second}$

tcp-ethereal-trace-1

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

tcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	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=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041797	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=560 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
14	0.169118	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	0.306000	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]

> Frame 4: 619 bytes on wire (4952 bits), 619 bytes captured (4952 bits)

> Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: Linksys6_da:af:73 (00:06:25:da:af:73)

> Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12

✓ Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565

Source Port: 1161

Destination Port: 80

[Stream index: 0]

[TCP Segment Len=565]

Sequence number: 1 (relative sequence number)

[Next sequence number: 566 (relative sequence number)]

Acknowledgment number: 1 (relative ack number)

0101 = Header Length: 20 bytes (5)

✓ Flags: 0x018 (PSH, ACK)

000. = Reserved: Not set

...0 = Nonce: Not set

....0 = Congestion Window Reduced (CWR): Not set

....0 = ECH-Echo: Not set

0000 1f bd 00 00 50 4f 53 54 20 2f 65 74 68 65 30PO ST /ethe

0000 72 65 61 6c 20 6c 61 62 73 2f 6c 62 33 20 31 real-lab s/lab3-1

0000 2d 72 65 70 6c 79 2e 68 74 6d 20 48 54 54 50 2f -reply.htm HTTP/

0000 31 2e 31 0d 0a 48 6f 73 74 3a 20 67 61 69 61 2e 1.1 -Host t: gala.

0000 63 73 2e 75 6d 61 73 73 2e 65 64 75 0d 0a 55 73 cs.umass.edu: Us

0000 65 72 2d 41 67 65 9e 74 3a 20 4d 6f 7a 09 6c 6c en-Agent : Mozilla

0000 61 2f 35 2e 30 20 28 57 69 6e 64 6f 77 73 3b 20 a/5.0 (Windows;

0000 55 3b 20 57 69 6e 64 6f 77 73 20 4e 54 20 35 2e U; Windo ws NT 5.

0000 31 3b 20 65 6e 2d 55 53 3b 20 72 76 3a 31 2e 30 1; en-US ; rv:1.0

0000 2e 32 29 20 47 65 63 6b 6f 2f 32 30 30 33 30 32 .2) Gecko o/200302

The window size value from the TCP header (tcp.window_size_value), 2 字节

分帧: 213 · 已显示: 202 (94.8%)

配置: Default

tcp-ethereal-trace-1

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

tcp

No.	Time	Source	Destination	Protocol	Length	Info
186	5.019189	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=151197 Win=62780 Len=0
190	5.125019	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=154117 Win=62780 Len=0
191	5.197286	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=156469 Win=62780 Len=0
192	5.197508	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=156469 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
193	5.198388	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=157929 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
194	5.199275	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=159389 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
195	5.200252	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=160849 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
196	5.201150	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=162309 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
197	5.202024	192.168.1.102	128.119.245.12	TCP	326	1161 → 80 [PSH, ACK] Seq=163769 Ack=1 Win=17520 Len=272 [TCP segment of a reassembled PDU]
198	5.292257	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=159389 Win=62780 Len=0
199	5.297341	192.168.1.102	128.119.245.12	HTTP	104	POST /etherreal-lab3/lab3-1-reply.htm HTTP/1.1 (text/plain)
200	5.389471	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=162309 Win=62780 Len=0
201	5.447887	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=164091 Win=62780 Len=0
202	5.455830	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=164091 Win=62780 Len=0
203	5.461175	128.119.245.12	192.168.1.102	HTTP	784	HTTP/1.1 200 OK (text/html)
206	5.651141	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=164091 Ack=731 Win=16790 Len=0
213	7.595557	192.168.1.102	199.2.53.206	TCP	62	1162 → 631 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1

✓ Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 1, Ack: 164091, Len: 0

Source Port: 80

Destination Port: 1161

[Stream index: 0]

[TCP Segment Len: 0]

Sequence number: 1 (relative sequence number)

[Next sequence number: 1 (relative sequence number)]

Acknowledgment number: 164091 (relative ack number)

0101 = Header Length: 20 bytes (5)

✓ Flags: 0x010 (ACK)

000. = Reserved: Not set

...0 = Nonce: Not set

....0 = Congestion Window Reduced (CWR): Not set

....0 = ECH-Echo: Not set

....0 = Urgent: Not set

....1 = Acknowledgment: Set

....0 = Push: Not set

0000 00 20 e0 8a 70 1a 00 06 25 da af 73 08 00 45 00 ...p...%...E

0010 00 28 58 bb 40 00 37 06 b3 82 80 77 f5 0c a0 a8 (X@7...m...

0020 01 66 00 50 04 09 34 a2 74 1a 0d a0 82 ef 50 10 f.P..4...E...P.

0030 f5 3f 44 a8 00 00 e5 e7 00 00 07 fb 80

The window size value from the TCP header (tcp.window_size_value), 2 字节

分帧: 213 · 已显示: 202 (94.8%)

配置: Default