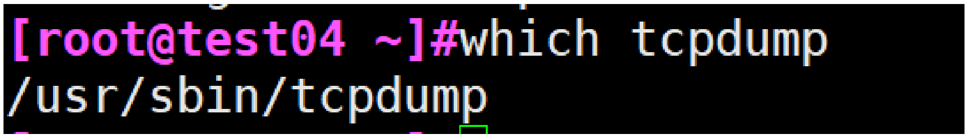
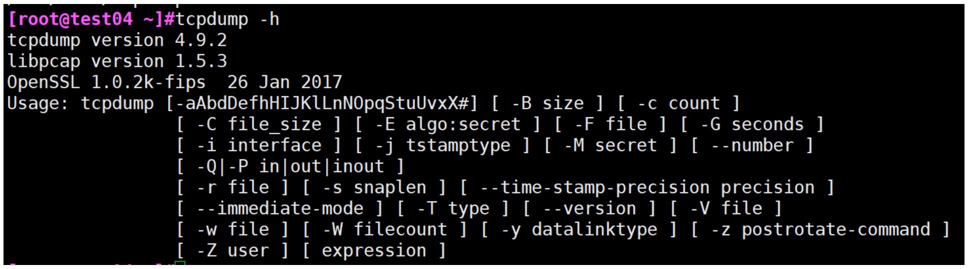
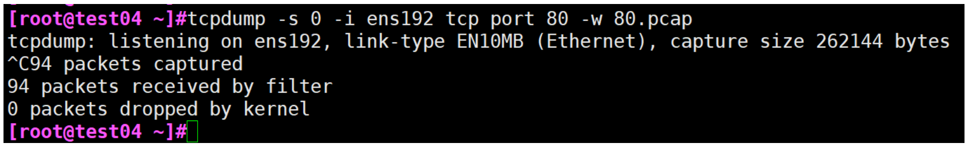
001【TCPIP协议原理揭秘】教案

序号	环节	步骤
0		整个演示环境中用到的文件 80.pcap
1	tcpdump工具	 [root@test04 ~]#which tcpdump /usr/sbin/tcpdump --- 出现上面的结果表示已经安装了tcpdump包

2	tcpdump
	帮助文档
		
		[root@test04 ~]#tcpdump -h
		tcpdump version 4.9.2
		libpcap version 1.5.3
		OpenSSL 1.0.2k-fips 26 Jan 2017
		Usage: tcpdump [-aAbdDefhHIJKlLnNOpqStuUvxX#] [-B size] [-c count]
		                [-C file_size] [-E algo:secret] [-F file] [-G seconds]
		                [-i interface] [-j tstamptype] [-M secret] [--number]
		                [-Q|-P in|out|inout]
		                [-r file] [-s snaplen] [--time-stamp-precision precision]
		                [--immediate-mode] [-T type] [--version] [-V file]
		                [-w file] [-W filecount] [-y datalinktype] [-z postrotate-command]
		                [-Z user] [expression]

3	使用方式1	tcpdump -s 0 -i　接口名　　　　备注：接口名通过ifconfig -a 获得】
4	使用方式2 增加限制端口	tcpdump -s 0 -i 接口名　tcp port 80 表示报文里的全部是跟80端口相关的【可以是源端口80 也可以是目的端口80】

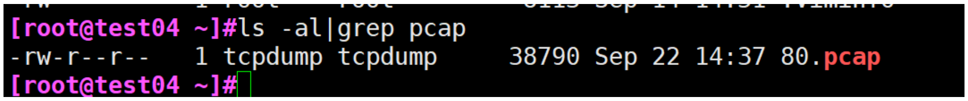
<table>
<tr>
<td>5</td>
<td>使用方式3
保存到pcap文件</td>
<td>



```
[root@test04 ~]#tcpdump -s 0 -i ens192 tcp port 80 -w 80.pcap
tcpdump: listening on ens192, link-type EN10MB (Ethernet), capture size 262144 bytes
^C94 packets captured
94 packets received by filter
0 packets dropped by kernel
[root@test04 ~]#
```

在另外一个窗口执行　wget www.baidu.com

回到第1个界面，ctrl+c就可以退出，然后看到本地有个80.pcap文件如下



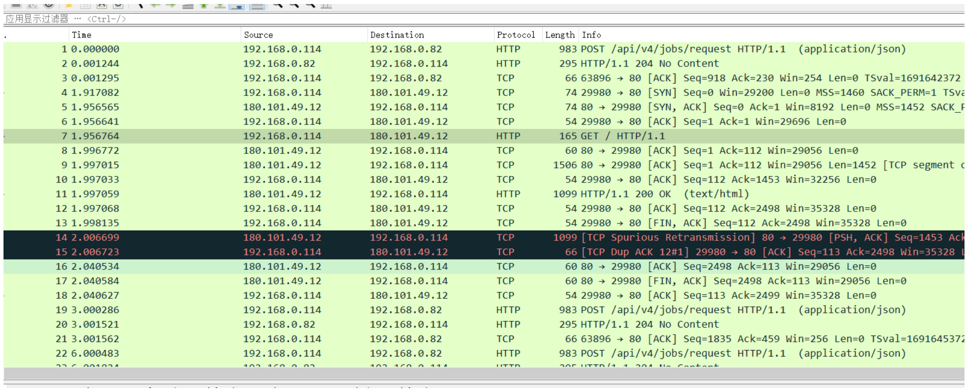
```
[root@test04 ~]#ls -al|grep pcap
-rw-r--r--    1 tcpdump tcpdump     38790 Sep 22 14:37 80.pcap
[root@test04 ~]#
```

</td>
</tr>
<tr>
<td colspan="3">休息1分钟</td>
</tr>
<tr>
<td>6</td>
<td>拷贝到本地</td>
<td>sz 80.pcap　　　　　　　　　　　如果没有这个命令，则执行yum -y install lrzsz</td>
</tr>
<tr>
<td>7</td>
<td>使用wireshark</td>
<td>

备注：如果无法识别，则安装wireshark软件,比如通过360软件管家

双击这个文件,

结果如图所示：

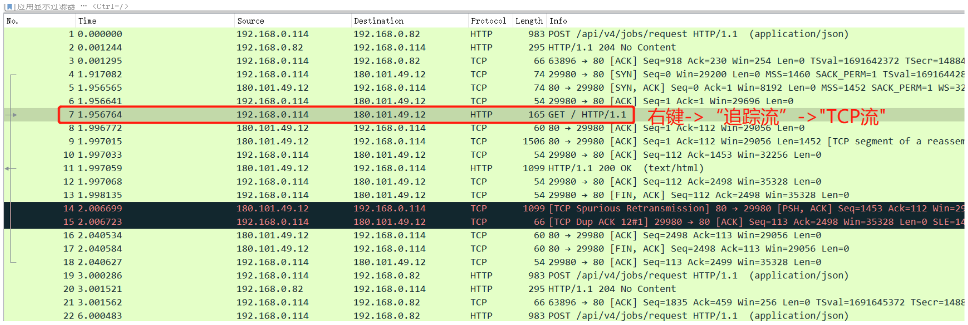


.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.114	192.168.0.82	HTTP	983	POST /api/v4/jobs/request HTTP/1.1 (application/json)
2	0.001244	192.168.0.82	192.168.0.114	HTTP	295	HTTP/1.1 204 No Content
3	0.001295	192.168.0.114	192.168.0.82	TCP	66	63896 → 80 [ACK] Seq=918 Ack=230 Win=254 Len=0 TSval=1691642372
4	1.917082	192.168.0.114	180.101.49.12	TCP	74	29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSva
5	1.956565	180.101.49.12	192.168.0.114	TCP	74	80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_P
6	1.956641	192.168.0.114	180.101.49.12	TCP	54	29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
7	1.956764	192.168.0.114	180.101.49.12	HTTP	165	GET / HTTP/1.1
8	1.996772	180.101.49.12	192.168.0.114	TCP	60	80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
9	1.997015	180.101.49.12	192.168.0.114	TCP	1506	80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment o
10	1.997033	192.168.0.114	180.101.49.12	TCP	54	29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
11	1.997059	180.101.49.12	192.168.0.114	HTTP	1099	HTTP/1.1 200 OK (text/html)
12	1.997068	192.168.0.114	180.101.49.12	TCP	54	29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
13	1.998135	192.168.0.114	180.101.49.12	TCP	54	29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
14	2.006699	180.101.49.12	192.168.0.114	TCP	1099	[TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack
15	2.006723	192.168.0.114	180.101.49.12	TCP	66	[TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 L
16	2.040534	180.101.49.12	192.168.0.114	TCP	60	80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
17	2.040584	180.101.49.12	192.168.0.114	TCP	60	80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
18	2.040627	192.168.0.114	180.101.49.12	TCP	54	29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0
19	3.000286	192.168.0.114	192.168.0.82	HTTP	983	POST /api/v4/jobs/request HTTP/1.1 (application/json)
20	3.001521	192.168.0.82	192.168.0.114	HTTP	295	HTTP/1.1 204 No Content
21	3.001562	192.168.0.114	192.168.0.82	TCP	66	63896 → 80 [ACK] Seq=1835 Ack=459 Win=256 Len=0 TSval=169164537
22	6.000483	192.168.0.114	192.168.0.82	HTTP	983	POST /api/v4/jobs/request HTTP/1.1 (application/json)

</td>
</tr>
</table>

8

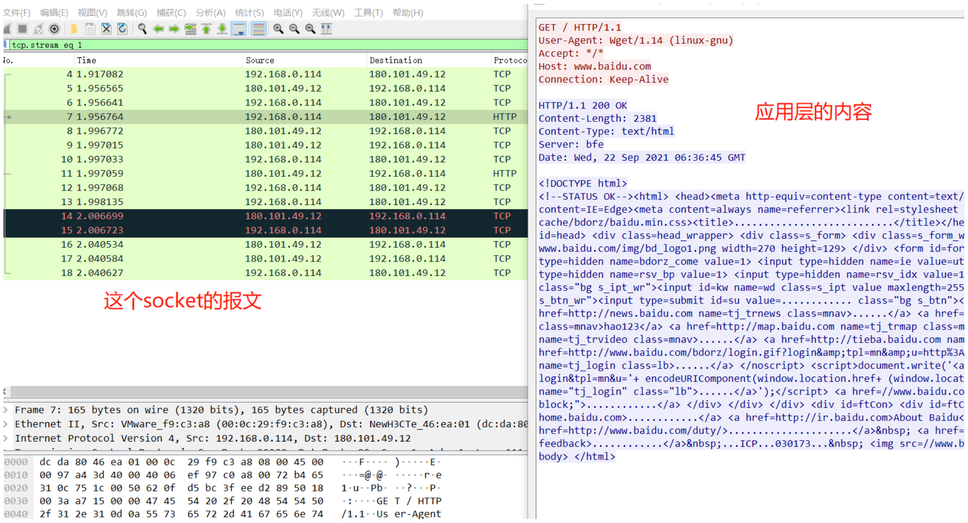
过滤出自己要的单一socket



```
No.   Time      Source         Destination    Protocol Length Info
    1 0.000000  192.168.0.114  192.168.0.82   HTTP      983 POST /api/v4/jobs/request HTTP/1.1  (application/json)
    2 0.001244  192.168.0.82   192.168.0.114  HTTP      295 HTTP/1.1 204 No Content
    3 0.001295  192.168.0.114  192.168.0.82   TCP        66 63896 → 80 [ACK] Seq=918 Ack=230 Win=254 Len=0 TSval=1691642372 TSecr=14884
    4 1.917082  192.168.0.114  180.101.49.12  TCP        74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=169164428
    5 1.956565  180.101.49.12  192.168.0.114  TCP        74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 WS=32
    6 1.956641  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764  192.168.0.114  180.101.49.12  HTTP      165 GET / HTTP/1.1
    8 1.996772  180.101.49.12  192.168.0.114  TCP        60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015  180.101.49.12  192.168.0.114  TCP      1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassem
   10 1.997033  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
   11 1.997059  180.101.49.12  192.168.0.114  HTTP     1099 HTTP/1.1 200 OK  (text/html)
   12 1.997068  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
   13 1.998135  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
   14 2.006699  180.101.49.12  192.168.0.114  TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29
   15 2.006723  192.168.0.114  180.101.49.12  TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=14
   16 2.040534  180.101.49.12  192.168.0.114  TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
   17 2.040584  180.101.49.12  192.168.0.114  TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
   18 2.040627  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0
   19 3.000286  192.168.0.114  192.168.0.82   HTTP      983 POST /api/v4/jobs/request HTTP/1.1  (application/json)
   20 3.001521  192.168.0.82   192.168.0.114  HTTP      295 HTTP/1.1 204 No Content
   21 3.001562  192.168.0.114  192.168.0.82   TCP        66 63896 → 80 [ACK] Seq=1835 Ack=459 Win=256 Len=0 TSval=1691645372 TSecr=1488
   22 6.000483  192.168.0.114  192.168.0.82   HTTP      983 POST /api/v4/jobs/request HTTP/1.1  (application/json)
```

右键-> “追踪流” ->"TCP流"

———


文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)
tcp.stream eq 1

```
No.   Time      Source         Destination    Protoco
    4 1.917082  192.168.0.114  180.101.49.12  TCP
    5 1.956565  180.101.49.12  192.168.0.114  TCP
    6 1.956641  192.168.0.114  180.101.49.12  TCP
    7 1.956764  192.168.0.114  180.101.49.12  HTTP
    8 1.996772  180.101.49.12  192.168.0.114  TCP
    9 1.997015  180.101.49.12  192.168.0.114  TCP
   10 1.997033  192.168.0.114  180.101.49.12  TCP
   11 1.997059  180.101.49.12  192.168.0.114  HTTP
   12 1.997068  192.168.0.114  180.101.49.12  TCP
   13 1.998135  192.168.0.114  180.101.49.12  TCP
   14 2.006699  180.101.49.12  192.168.0.114  TCP
   15 2.006723  192.168.0.114  180.101.49.12  TCP
   16 2.040534  180.101.49.12  192.168.0.114  TCP
   17 2.040584  180.101.49.12  192.168.0.114  TCP
   18 2.040627  192.168.0.114  180.101.49.12  TCP
```

这个socket的报文

```
> Frame 7: 165 bytes on wire (1320 bits), 165 bytes captured (1320 bits)
> Ethernet II, Src: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8), Dst: NewH3CTe_46:ea:01 (dc:da:80
> Internet Protocol Version 4, Src: 192.168.0.114, Dst: 180.101.49.12
0000  dc da 80 46 ea 01 00 0c  29 f9 c3 a8 08 00 45 00   ···F···· )·····E·
0010  00 97 a4 3d 40 00 40 06  ef 97 c0 a8 00 72 b4 65   ···=@·@· ·····r·e
0020  31 0c 75 1c 00 50 62 0f  d5 bc 3f ee d2 89 50 18   1·u··Pb· ··?···P·
0030  00 3a a7 15 00 00 47 45  54 20 2f 20 48 54 54 50   ·:····GE T / HTTP
0040  2f 31 2e 31 0d 0a 55 73  65 72 2d 41 67 65 6e 74   /1.1··Us er-Agent
```

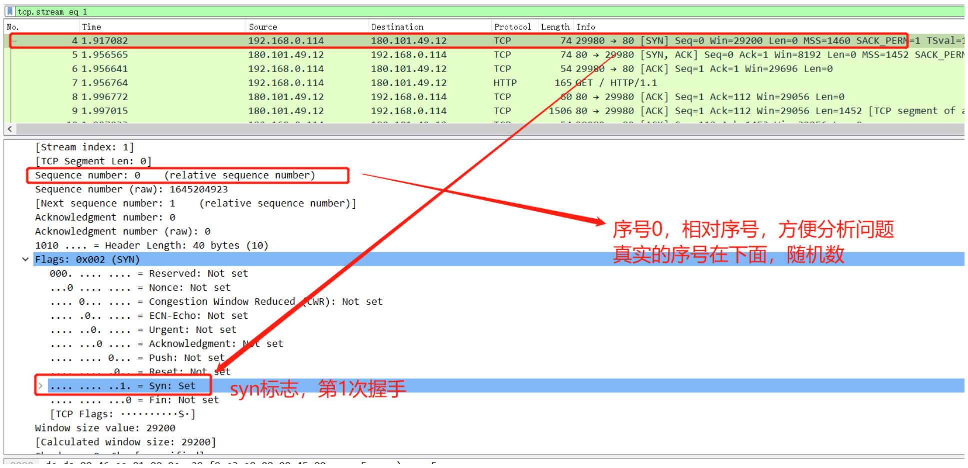
```
GET / HTTP/1.1
User-Agent: Wget/1.14 (linux-gnu)
Accept: */*
Host: www.baidu.com
Connection: Keep-Alive

HTTP/1.1 200 OK
Content-Length: 2381
Content-Type: text/html
Server: bfe
Date: Wed, 22 Sep 2021 06:36:45 GMT

<!DOCTYPE html>
<!--STATUS OK--><html> <head><meta http-equiv=content-type content=text/
content=IE=Edge><meta content=always name=referrer><link rel=stylesheet
cache/bdorz/baidu.min.css><title>.........................</title></he
id=head> <div class=head_wrapper> <div class=s_form> <div class=s_form_w
www.baidu.com/img/bd_logo1.png width=270 height=129> </div> <form id=for
type=hidden name=bdorz_come value=1> <input type=hidden name=ie value=ut
type=hidden name=rsv_bp value=1> <input type=hidden name=rsv_idx value=1
class="bg s_ipt_wr"><input id=kw name=wd class=s_ipt value maxlength=255
s_btn_wr"><input type=submit id=su value=........... class="bg s_btn"><
href=http://news.baidu.com name=tj_trnews class=mnav>......</a> <a href=
class=mnav>hao123</a> <a href=http://map.baidu.com name=tj_trmap class=m
name=tj_trvideo class=mnav>......</a> <a href=http://tieba.baidu.com nam
href=http://www.baidu.com/bdorz/login.gif?login&amp;tpl=mn&amp;u=http%3A
name=tj_login class=lb>......</a> </noscript> <script>document.write('<a
login&tpl=mn&u='+ encodeURIComponent(window.location.href+ (window.locat
name="tj_login" class="lb">......</a>');</script> <a href=//www.baidu.co
block;">...........</a> </div> </div> </div> <div id=ftCon> <div id=ftC
home.baidu.com>...........</a> <a href=http://ir.baidu.com>About Baidu<
href=http://www.baidu.com/duty/>....................</a>&nbsp; <a href=
feedback>...........</a>&nbsp;...ICP...030173...&nbsp; <img src=//www.b
body> </html>
```

应用层的内容

休息1分钟

9

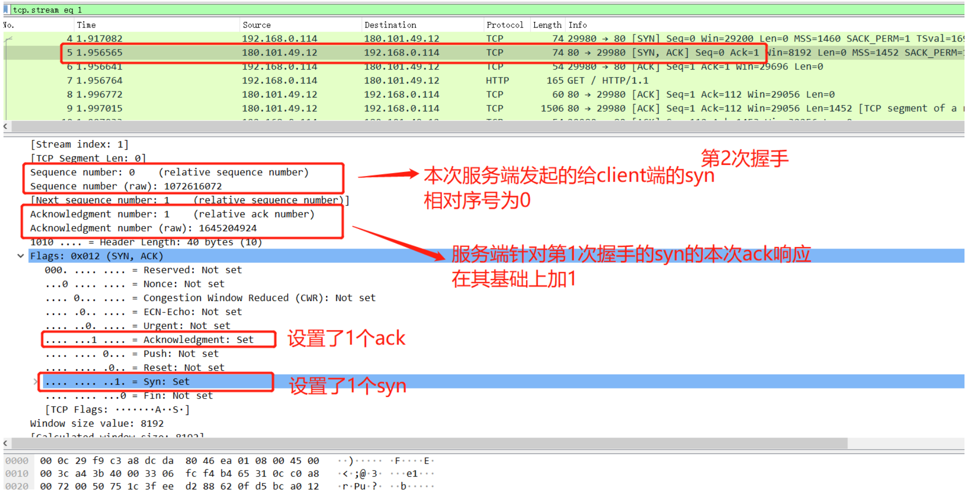
第1次握手



```
No.   Time      Source         Destination    Protocol Length Info
    4 1.917082  192.168.0.114  180.101.49.12  TCP        74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1
    5 1.956565  180.101.49.12  192.168.0.114  TCP        74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM
    6 1.956641  192.168.0.114  180.101.49.12  TCP        54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764  192.168.0.114  180.101.49.12  HTTP      165 GET / HTTP/1.1
    8 1.996772  180.101.49.12  192.168.0.114  TCP        60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015  180.101.49.12  192.168.0.114  TCP      1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a

    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Sequence number (raw): 1645204923
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 0
    Acknowledgment number (raw): 0
    1010 .... = Header Length: 40 bytes (10)
  v Flags: 0x002 (SYN)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): 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
      .... .... ...0 = Fin: Not set
      [TCP Flags: ··········S·]
    Window size value: 29200
    [Calculated window size: 29200]
```

序号0，相对序号，方便分析问题
真实的序号在下面，随机数

syn标志，第1次握手

10	第2次握手	

```
tcp.stream eq 1
No.   Time        Source          Destination     Protocol  Length Info
    4 1.917082    192.168.0.114   180.101.49.12   TCP       74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=16
    5 1.956565    180.101.49.12   192.168.0.114   TCP       74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=
    6 1.956641    192.168.0.114   180.101.49.12   TCP       54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764    192.168.0.114   180.101.49.12   HTTP     165 GET / HTTP/1.1
    8 1.996772    180.101.49.12   192.168.0.114   TCP       60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015    180.101.49.12   192.168.0.114   TCP     1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a

    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Sequence number (raw): 1072616072
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    Acknowledgment number (raw): 1645204924
    1010 .... = Header Length: 40 bytes (10)
  v Flags: 0x012 (SYN, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): 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·]
    Window size value: 8192
    [Calculated window size: 8192]

0000  00 0c 29 f9 c3 a8 dc da  80 46 ea 01 08 00 45 00   ··)····· ·F····E·
0010  00 3c a4 3b 40 00 33 06  fc f4 b4 65 31 0c c0 a8   ·<·;@·3· ···e1···
0020  00 72 00 50 75 1c 3f ee  d2 88 62 0f d5 bc a0 12   ·r·Pu·?· ··b·····
```

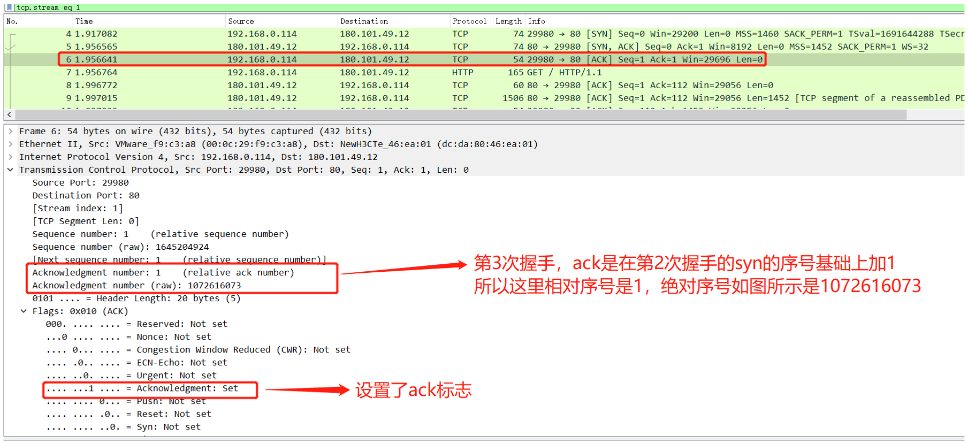
第2次握手

本次服务端发起的给client端的syn
相对序号为0

服务端针对第1次握手的syn的本次ack响应
在其基础上加1

设置了1个ack

设置了1个syn

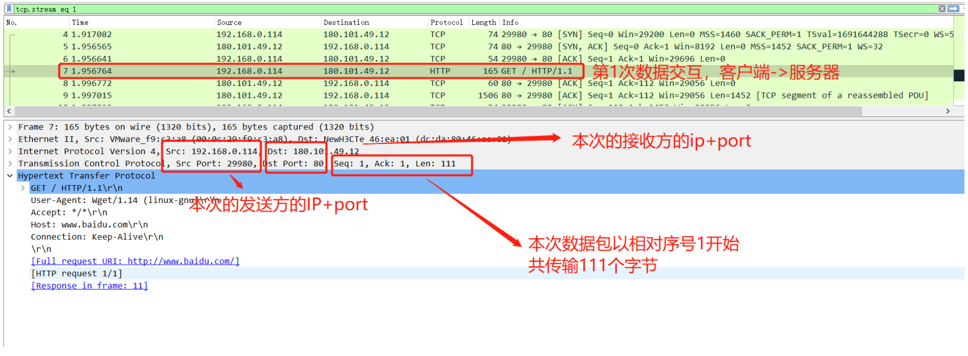
11	第3次握手	

```
tcp.stream eq 1
No.   Time        Source          Destination     Protocol  Length Info
    4 1.917082    192.168.0.114   180.101.49.12   TCP       74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1691644288 TSecr
    5 1.956565    180.101.49.12   192.168.0.114   TCP       74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 WS=32
    6 1.956641    192.168.0.114   180.101.49.12   TCP       54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764    192.168.0.114   180.101.49.12   HTTP     165 GET / HTTP/1.1
    8 1.996772    180.101.49.12   192.168.0.114   TCP       60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015    180.101.49.12   192.168.0.114   TCP     1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassembled PD

> Frame 6: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
> Ethernet II, Src: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8), Dst: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01)
> Internet Protocol Version 4, Src: 192.168.0.114, Dst: 180.101.49.12
v Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
    Source Port: 29980
    Destination Port: 80
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Sequence number (raw): 1645204924
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    Acknowledgment number (raw): 1072616073
    0101 .... = Header Length: 20 bytes (5)
  v Flags: 0x010 (ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
```

第3次握手，ack是在第2次握手的syn的序号基础上加1
所以这里相对序号是1，绝对序号如图所示是1072616073

设置了ack标志

休息1分钟

12	第1次数据交互	

```
tcp.stream eq 1
No.   Time        Source          Destination     Protocol  Length Info
    4 1.917082    192.168.0.114   180.101.49.12   TCP       74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1691644288 TSecr=0 WS=5
    5 1.956565    180.101.49.12   192.168.0.114   TCP       74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 WS=32
    6 1.956641    192.168.0.114   180.101.49.12   TCP       54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764    192.168.0.114   180.101.49.12   HTTP     165 GET / HTTP/1.1
    8 1.996772    180.101.49.12   192.168.0.114   TCP       60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015    180.101.49.12   192.168.0.114   TCP     1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassembled PDU]

> Frame 7: 165 bytes on wire (1320 bits), 165 bytes captured (1320 bits)
> Ethernet II, Src: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8), Dst: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01)
> Internet Protocol Version 4, Src: 192.168.0.114, Dst: 180.101.49.12
> Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 1, Ack: 1, Len: 111
v Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    User-Agent: Wget/1.14 (linux-gnu)\r\n
    Accept: */*\r\n
    Host: www.baidu.com\r\n
    Connection: Keep-Alive\r\n
    \r\n
    [Full request URI: http://www.baidu.com/]
    [HTTP request 1/1]
    [Response in frame: 11]
```

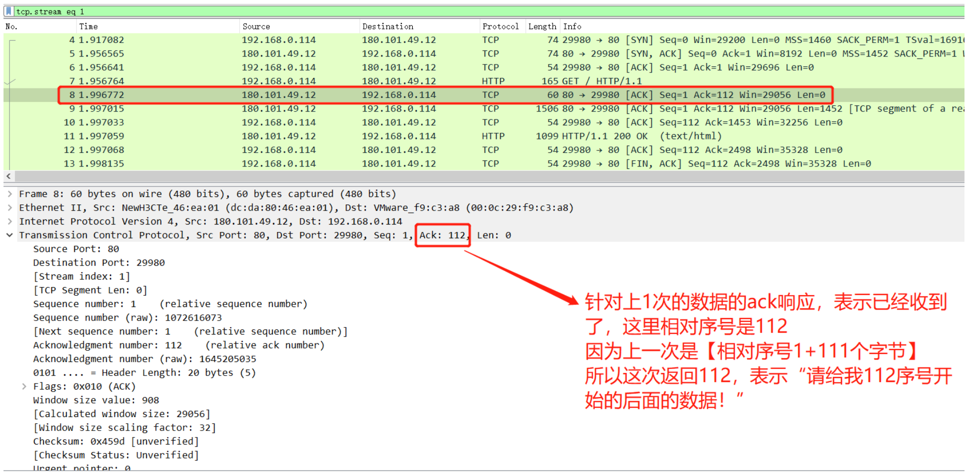
第1次数据交互，客户端->服务器

本次的接收方的ip+port

本次的发送方的IP+port

本次数据包以相对序号1开始
共传输111个字节

<table>
<tr>
<td>13</td>
<td>对方的响应</td>
<td>

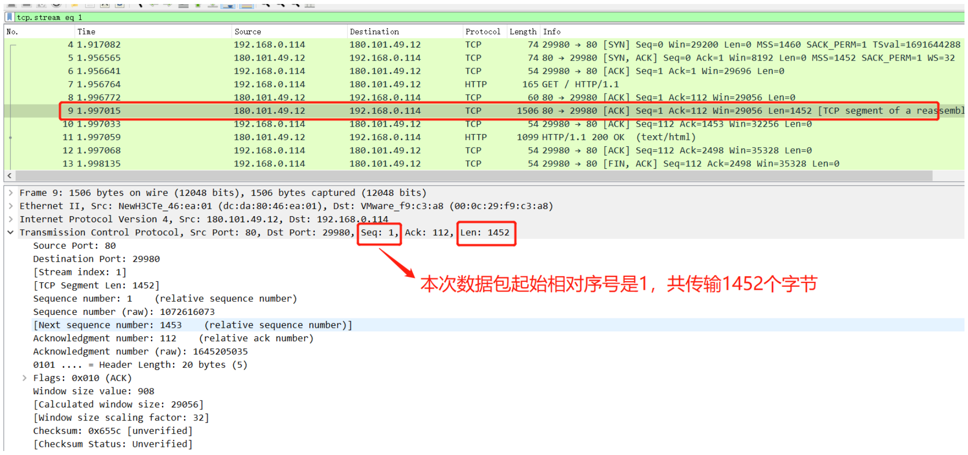


```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol  Length  Info
     4 1.917082    192.168.0.114    180.101.49.12    TCP         74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1691
     5 1.956565    180.101.49.12    192.168.0.114    TCP         74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 W
     6 1.956641    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
     7 1.956764    192.168.0.114    180.101.49.12    HTTP       165 GET / HTTP/1.1
     8 1.996772    180.101.49.12    192.168.0.114    TCP         60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
     9 1.997015    180.101.49.12    192.168.0.114    TCP       1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a rea
    10 1.997033    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
    11 1.997059    180.101.49.12    192.168.0.114    HTTP      1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0

> Frame 8: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
> Ethernet II, Src: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01), Dst: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8)
> Internet Protocol Version 4, Src: 180.101.49.12, Dst: 192.168.0.114
v Transmission Control Protocol, Src Port: 80, Dst Port: 29980, Seq: 1, Ack: 112, Len: 0
    Source Port: 80
    Destination Port: 29980
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Sequence number (raw): 1072616073
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 112    (relative ack number)
    Acknowledgment number (raw): 1645205035
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 908
    [Calculated window size: 29056]
    [Window size scaling factor: 32]
    Checksum: 0x459d [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
```

针对上1次的数据的ack响应，表示已经收到了，这里相对序号是112
因为上一次是【相对序号1+111个字节】
所以这次返回112，表示“请给我112序号开始的后面的数据！”

</td>
</tr>
<tr>
<td>14</td>
<td>服务器开始返回数据</td>
<td>

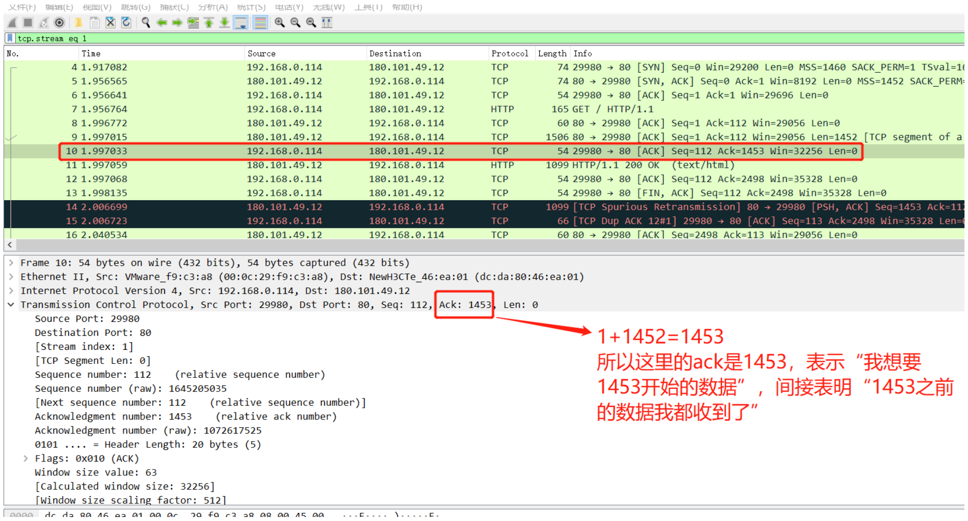
因为是http请求，所以此时会等待服务端发回响应内容，如图所示



```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol  Length  Info
     4 1.917082    192.168.0.114    180.101.49.12    TCP         74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1691644288
     5 1.956565    180.101.49.12    192.168.0.114    TCP         74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 WS=32
     6 1.956641    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
     7 1.956764    192.168.0.114    180.101.49.12    HTTP       165 GET / HTTP/1.1
     8 1.996772    180.101.49.12    192.168.0.114    TCP         60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
     9 1.997015    180.101.49.12    192.168.0.114    TCP       1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassembl
    10 1.997033    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
    11 1.997059    180.101.49.12    192.168.0.114    HTTP      1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0

> Frame 9: 1506 bytes on wire (12048 bits), 1506 bytes captured (12048 bits)
> Ethernet II, Src: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01), Dst: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8)
> Internet Protocol Version 4, Src: 180.101.49.12, Dst: 192.168.0.114
v Transmission Control Protocol, Src Port: 80, Dst Port: 29980, Seq: 1, Ack: 112, Len: 1452
    Source Port: 80
    Destination Port: 29980
    [Stream index: 1]
    [TCP Segment Len: 1452]
    Sequence number: 1    (relative sequence number)
    Sequence number (raw): 1072616073
    [Next sequence number: 1453    (relative sequence number)]
    Acknowledgment number: 112    (relative ack number)
    Acknowledgment number (raw): 1645205035
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 908
    [Calculated window size: 29056]
    [Window size scaling factor: 32]
    Checksum: 0x655c [unverified]
    [Checksum Status: Unverified]
```

本次数据包起始相对序号是1，共传输1452个字节

</td>
</tr>
<tr>
<td>15</td>
<td>客户端的ack响应</td>
<td>

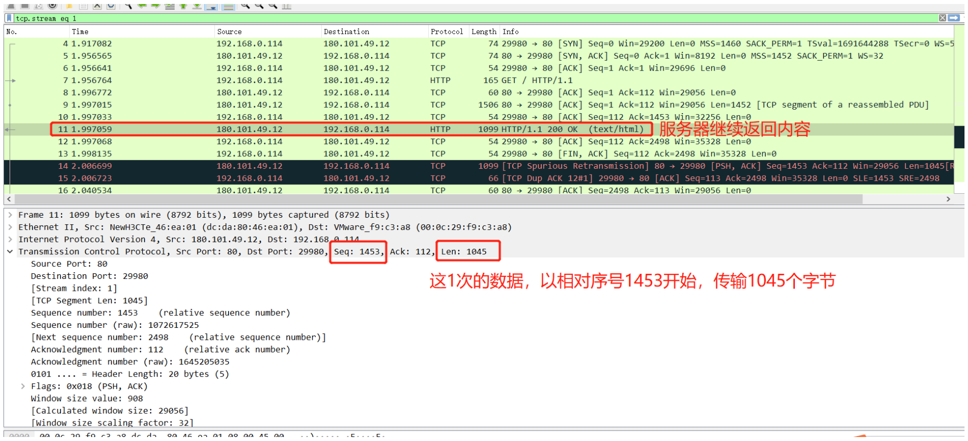

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)

```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol  Length  Info
     4 1.917082    192.168.0.114    180.101.49.12    TCP         74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=16
     5 1.956565    180.101.49.12    192.168.0.114    TCP         74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=
     6 1.956641    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
     7 1.956764    192.168.0.114    180.101.49.12    HTTP       165 GET / HTTP/1.1
     8 1.996772    180.101.49.12    192.168.0.114    TCP         60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
     9 1.997015    180.101.49.12    192.168.0.114    TCP       1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a
    10 1.997033    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
    11 1.997059    180.101.49.12    192.168.0.114    HTTP      1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP         54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
    14 2.006699    180.101.49.12    192.168.0.114    TCP       1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=11
    15 2.006723    192.168.0.114    180.101.49.12    TCP         66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=
    16 2.040534    180.101.49.12    192.168.0.114    TCP         60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0

> Frame 10: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
> Ethernet II, Src: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8), Dst: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01)
> Internet Protocol Version 4, Src: 192.168.0.114, Dst: 180.101.49.12
v Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 112, Ack: 1453, Len: 0
    Source Port: 29980
    Destination Port: 80
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 112    (relative sequence number)
    Sequence number (raw): 1645205035
    [Next sequence number: 112    (relative sequence number)]
    Acknowledgment number: 1453    (relative ack number)
    Acknowledgment number (raw): 1072617525
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 63
    [Calculated window size: 32256]
    [Window size scaling factor: 512]
0000  dc da 80 46 ea 01 00 0c  29 f9 c3 a8 08 00 45 00   ...F.... ).....E.
```

1+1452=1453
所以这里的ack是1453，表示“我想要1453开始的数据”，间接表明“1453之前的数据我都收到了”

</td>
</tr>
</table>

16 服务器继续发送

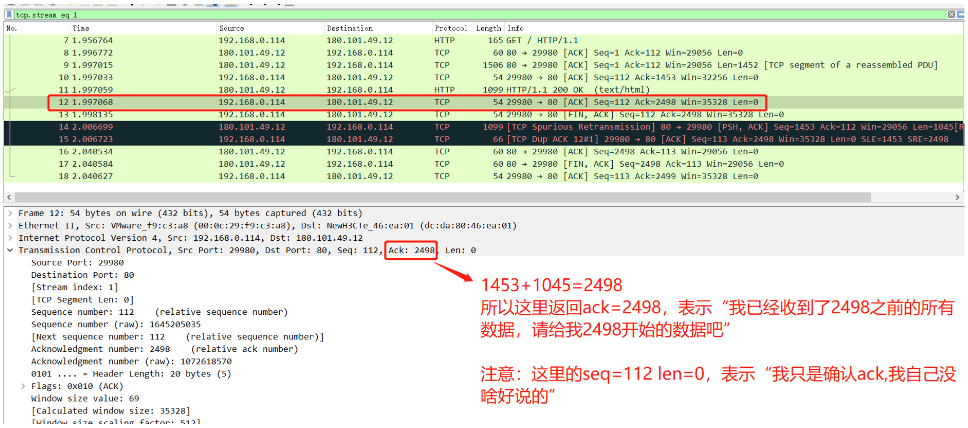


```
tcp.stream eq 1
No.   Time       Source          Destination     Protocol Length Info
    4 1.917082   192.168.0.114   180.101.49.12   TCP        74 29980 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1691644288 TSecr=0 WS=5
    5 1.956565   180.101.49.12   192.168.0.114   TCP        74 80 → 29980 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1452 SACK_PERM=1 WS=32
    6 1.956641   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0
    7 1.956764   192.168.0.114   180.101.49.12   HTTP      165 GET / HTTP/1.1
    8 1.996772   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015   180.101.49.12   192.168.0.114   TCP      1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassembled PDU]
   10 1.997033   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
   11 1.997059   180.101.49.12   192.168.0.114   HTTP     1099 HTTP/1.1 200 OK  (text/html)
   12 1.997068   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
   13 1.998135   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
   14 2.006699   180.101.49.12   192.168.0.114   TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[R
   15 2.006723   192.168.0.114   180.101.49.12   TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
   16 2.040534   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0

> Frame 11: 1099 bytes on wire (8792 bits), 1099 bytes captured (8792 bits)
> Ethernet II, Src: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01), Dst: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8)
> Internet Protocol Version 4, Src: 180.101.49.12, Dst: 192.168.0.114
v Transmission Control Protocol, Src Port: 80, Dst Port: 29980, Seq: 1453, Ack: 112, Len: 1045
    Source Port: 80
    Destination Port: 29980
    [Stream index: 1]
    [TCP Segment Len: 1045]
    Sequence number: 1453    (relative sequence number)
    Sequence number (raw): 1072617525
    [Next sequence number: 2498    (relative sequence number)]
    Acknowledgment number: 112    (relative ack number)
    Acknowledgment number (raw): 1645205035
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
    Window size value: 908
    [Calculated window size: 29056]
    [Window size scaling factor: 32]
```

服务器继续返回内容

这1次的数据，以相对序号1453开始，传输1045个字节

17 客户端的ack



```
tcp.stream eq 1
No.   Time       Source          Destination     Protocol Length Info
    7 1.956764   192.168.0.114   180.101.49.12   HTTP      165 GET / HTTP/1.1
    8 1.996772   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=0
    9 1.997015   180.101.49.12   192.168.0.114   TCP      1506 80 → 29980 [ACK] Seq=1 Ack=112 Win=29056 Len=1452 [TCP segment of a reassembled PDU]
   10 1.997033   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=112 Ack=1453 Win=32256 Len=0
   11 1.997059   180.101.49.12   192.168.0.114   HTTP     1099 HTTP/1.1 200 OK  (text/html)
   12 1.997068   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
   13 1.998135   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
   14 2.006699   180.101.49.12   192.168.0.114   TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[R
   15 2.006723   192.168.0.114   180.101.49.12   TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
   16 2.040534   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
   17 2.040584   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
   18 2.040627   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0

> Frame 12: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
> Ethernet II, Src: VMware_f9:c3:a8 (00:0c:29:f9:c3:a8), Dst: NewH3CTe_46:ea:01 (dc:da:80:46:ea:01)
> Internet Protocol Version 4, Src: 192.168.0.114, Dst: 180.101.49.12
v Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 112, Ack: 2498, Len: 0
    Source Port: 29980
    Destination Port: 80
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 112    (relative sequence number)
    Sequence number (raw): 1645205035
    [Next sequence number: 112    (relative sequence number)]
    Acknowledgment number: 2498    (relative ack number)
    Acknowledgment number (raw): 1072618570
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 69
    [Calculated window size: 35328]
    [Window size scaling factor: 512]
```

1453+1045=2498
所以这里返回ack=2498，表示“我已经收到了2498之前的所有数据，请给我2498开始的数据吧”

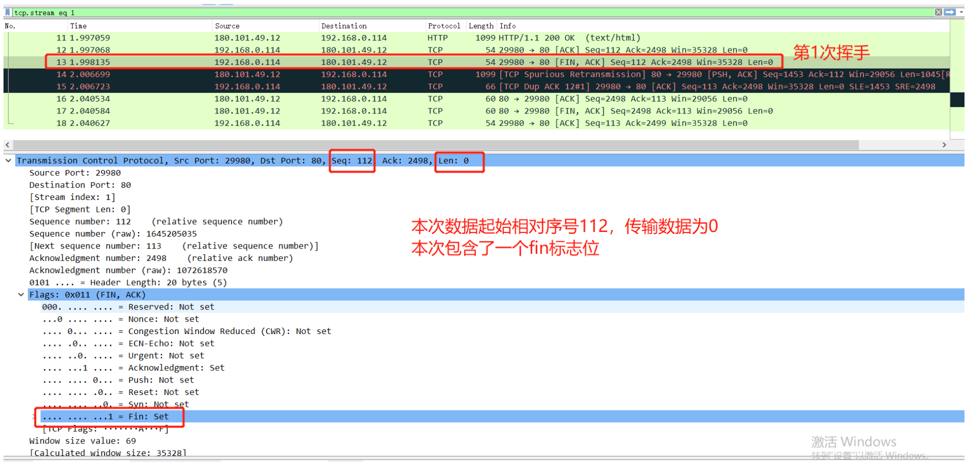
注意：这里的seq=112 len=0，表示“我只是确认ack,我自己没啥好说的”

休息1分钟，注意：

ack是对对方的消息的确认，seq+len是本次自己发送的数据

18 4次挥手的

第1次挥手



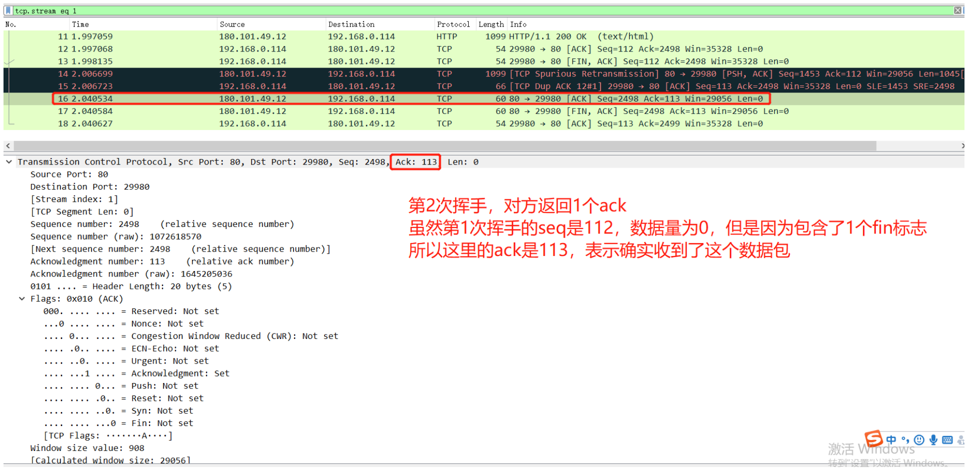
```
tcp.stream eq 1
No.   Time       Source          Destination     Protocol Length Info
   11 1.997059   180.101.49.12   192.168.0.114   HTTP     1099 HTTP/1.1 200 OK  (text/html)
   12 1.997068   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
   13 1.998135   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
   14 2.006699   180.101.49.12   192.168.0.114   TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[R
   15 2.006723   192.168.0.114   180.101.49.12   TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
   16 2.040534   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
   17 2.040584   180.101.49.12   192.168.0.114   TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
   18 2.040627   192.168.0.114   180.101.49.12   TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0

v Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 112, Ack: 2498, Len: 0
    Source Port: 29980
    Destination Port: 80
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 112    (relative sequence number)
    Sequence number (raw): 1645205035
    [Next sequence number: 113    (relative sequence number)]
    Acknowledgment number: 2498    (relative ack number)
    Acknowledgment number (raw): 1072618570
    0101 .... = Header Length: 20 bytes (5)
  v Flags: 0x011 (FIN, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...1 = Fin: Set
      [TCP Flags: ·······A···F]
    Window size value: 69
    [Calculated window size: 35328]
```

第1次挥手

本次数据起始相对序号112，传输数据为0
本次包含了一个fin标志位

4次挥手的

第2次挥手

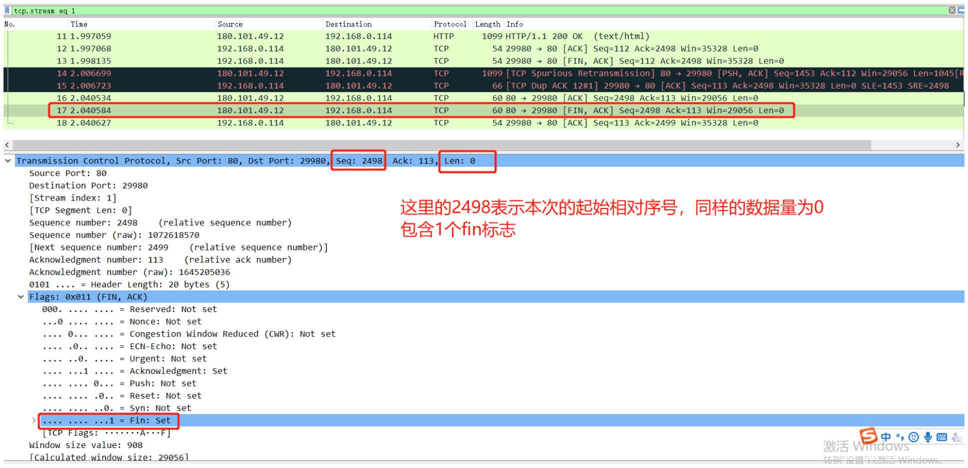


```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol Length Info
    11 1.997059    180.101.49.12    192.168.0.114    HTTP     1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
    14 2.006699    180.101.49.12    192.168.0.114    TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[
    15 2.006723    192.168.0.114    180.101.49.12    TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
    16 2.040534    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
    17 2.040584    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
    18 2.040627    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0

Transmission Control Protocol, Src Port: 80, Dst Port: 29980, Seq: 2498, Ack: 113, Len: 0
    Source Port: 80
    Destination Port: 29980
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 2498    (relative sequence number)
    Sequence number (raw): 1072618570
    [Next sequence number: 2498    (relative sequence number)]
    Acknowledgment number: 113    (relative ack number)
    Acknowledgment number (raw): 1645205036
    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.. .... = ECN-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: 908
    [Calculated window size: 29056]
```

第2次挥手，对方返回1个ack
虽然第1次挥手的seq是112，数据量为0，但是因为包含了1个fin标志
所以这里的ack是113，表示确实收到了这个数据包

4次挥手的

第3次挥手

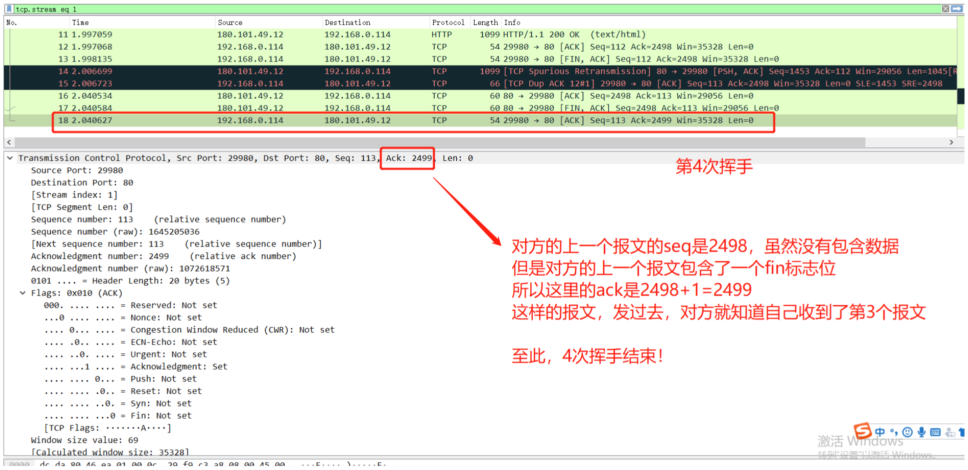


```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol Length Info
    11 1.997059    180.101.49.12    192.168.0.114    HTTP     1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
    14 2.006699    180.101.49.12    192.168.0.114    TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[R
    15 2.006723    192.168.0.114    180.101.49.12    TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
    16 2.040534    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
    17 2.040584    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
    18 2.040627    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0

Transmission Control Protocol, Src Port: 80, Dst Port: 29980, Seq: 2498, Ack: 113, Len: 0
    Source Port: 80
    Destination Port: 29980
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 2498    (relative sequence number)
    Sequence number (raw): 1072618570
    [Next sequence number: 2499    (relative sequence number)]
    Acknowledgment number: 113    (relative ack number)
    Acknowledgment number (raw): 1645205036
    0101 .... = Header Length: 20 bytes (5)
    Flags: 0x011 (FIN, ACK)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...1 .... = Acknowledgment: Set
        .... .... 0... = Push: Not set
        .... .... .0.. = Reset: Not set
        .... .... ..0. = Syn: Not set
      > .... .... ...1 = Fin: Set
        [TCP Flags: ·······A···F]
    Window size value: 908
    [Calculated window size: 29056]
```

这里的2498表示本次的起始相对序号，同样的数据量为0
包含1个fin标志

4次挥手的

第4次挥手



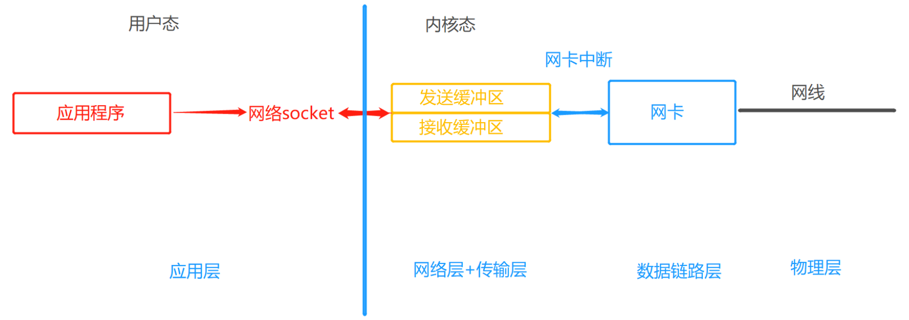
```
tcp.stream eq 1
No.    Time        Source           Destination      Protocol Length Info
    11 1.997059    180.101.49.12    192.168.0.114    HTTP     1099 HTTP/1.1 200 OK  (text/html)
    12 1.997068    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=112 Ack=2498 Win=35328 Len=0
    13 1.998135    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [FIN, ACK] Seq=112 Ack=2498 Win=35328 Len=0
    14 2.006699    180.101.49.12    192.168.0.114    TCP      1099 [TCP Spurious Retransmission] 80 → 29980 [PSH, ACK] Seq=1453 Ack=112 Win=29056 Len=1045[R
    15 2.006723    192.168.0.114    180.101.49.12    TCP        66 [TCP Dup ACK 12#1] 29980 → 80 [ACK] Seq=113 Ack=2498 Win=35328 Len=0 SLE=1453 SRE=2498
    16 2.040534    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [ACK] Seq=2498 Ack=113 Win=29056 Len=0
    17 2.040584    180.101.49.12    192.168.0.114    TCP        60 80 → 29980 [FIN, ACK] Seq=2498 Ack=113 Win=29056 Len=0
    18 2.040627    192.168.0.114    180.101.49.12    TCP        54 29980 → 80 [ACK] Seq=113 Ack=2499 Win=35328 Len=0

Transmission Control Protocol, Src Port: 29980, Dst Port: 80, Seq: 113, Ack: 2499, Len: 0
    Source Port: 29980
    Destination Port: 80
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 113    (relative sequence number)
    Sequence number (raw): 1645205036
    [Next sequence number: 113    (relative sequence number)]
    Acknowledgment number: 2499    (relative ack number)
    Acknowledgment number (raw): 1072618571
    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.. .... = ECN-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: 69
    [Calculated window size: 35328]
0000  dc da 80 46 ea 01 00 0c  29 f9 c3 a8 08 00 45 00   ···F···· )·····E·
```

第4次挥手

对方的上一个报文的seq是2498，虽然没有包含数据
但是对方的上一个报文包含了一个fin标志位
所以这里的ack是2498+1=2499
这样的报文，发过去，对方就知道自己收到了第3个报文

至此，4次挥手结束！

总结



用户态
内核态
网卡中断
应用程序
网络socket
发送缓冲区
接收缓冲区
网卡
网线
应用层
网络层+传输层
数据链路层
物理层