CPS 470/570: Wireshark Lab TCP

due 11:55 PM, Wednesday, 3-22-2017 (100 pts)

Receive 5 bonus points if submit it without errors at least one day before deadline
Receive an *F* for this course if any academic dishonesty occurs

1. Purpose

The goal of this lab is to introduce you to Wireshark and observe TCP traces in Wireshark.

2. Description

2.1. Overview

In this assignment, you will connect to our course website and observe the network protocols in your computer "in action," interacting and exchanging messages with protocol entities executing elsewhere in the Internet. You will observe, and you will learn, by doing.

2.2. Details

a) Getting Wireshark

The basic tool for observing the messages exchanged between executing protocol entities is called a **packet sniffer**. As the name suggests, a packet sniffer captures ("sniffs") messages being sent/received from/by your computer; it will also typically store and/or display the contents of the various protocol fields in these captured messages. A packet sniffer itself is passive. It observes messages being sent and received by applications and protocols running on your computer, but never sends packets itself. Similarly, received packets are never explicitly addressed to the packet sniffer. Instead, a packet sniffer receives a *copy* of packets that are sent/received from/by application and protocols executing on your machine.

Figure 1 shows the structure of a packet sniffer. At the right of Figure 1 are the protocols (in this case, Internet protocols) and applications (such as a web browser or ftp client) that normally run on your computer. The packet sniffer, shown within the dashed rectangle in Figure 1 is an addition to the usual software in your computer, and consists of two parts. The **packet capture library** receives a copy of every packet that is sent from or received by your computer. Recall from the discussion from section 1.5 in the text (Figure 1.24) that messages exchanged by higher layer protocols such as HTTP, FTP, TCP, UDP, DNS, or IP all are eventually encapsulated in link-layer frames that are transmitted over physical media such as an Ethernet cable. In Figure 1, the assumed physical media is an Ethernet, and so all upper layer protocols are eventually encapsulated within an Ethernet frame. Capturing all link-layer frames thus gives you all messages sent/received from/by all protocols and applications executing in your computer. The second component of a packet sniffer is the **packet analyzer**, which displays the contents of all fields within a protocol message.

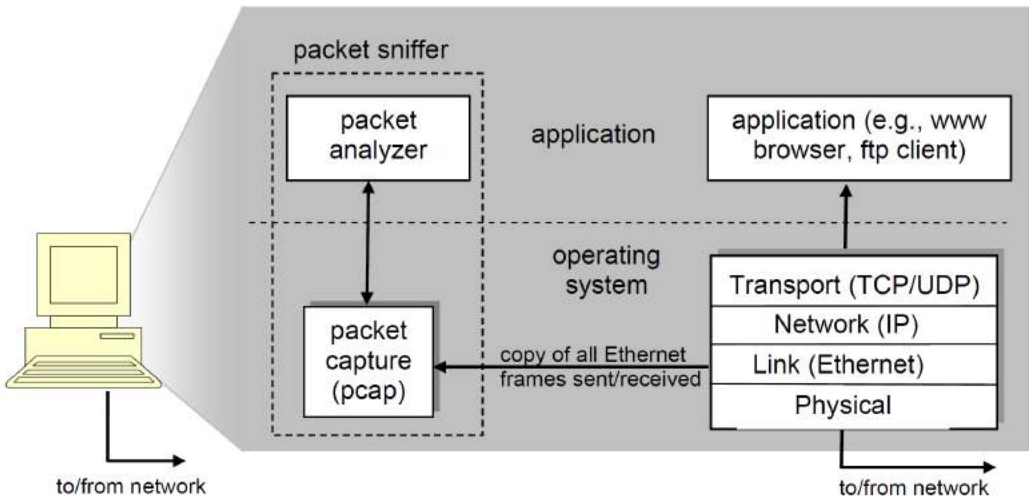


Figure 1. Packet sniffer structure

You will be using the **Wireshark** packet sniffer [http://www.wireshark.org/] for this assignment, allowing you to display the contents of messages being sent/received from/by protocols at different levels of the protocol stack. In order to run Wireshark, you will go to http://www.wireshark.org/download.html and download/install the Wireshark binary for your computer. The Wireshark FAQ has a number of helpful hints and interesting tidbits of information, particularly if you have trouble installing or running Wireshark.

b) Running Wireshark

When you run the Wireshark program, the Wireshark graphical user interface shown in **Figure 2** will be displayed. Initially, no data will be displayed in the various windows.

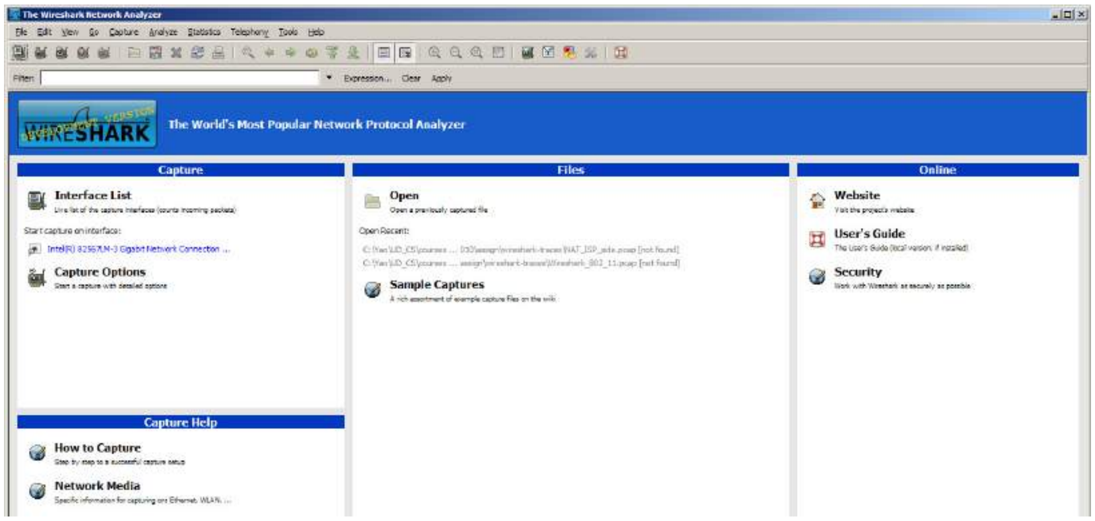

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter: Expression... Clear Apply
WIRESHARK
The World's Most Popular Network Protocol Analyzer
Capture
Interface List
Start capture on interface:
Capture Options
Files
Open
Open Recent:
Sample Captures
Online
Website
User's Guide
Security
Capture Help
How to Capture
Network Media

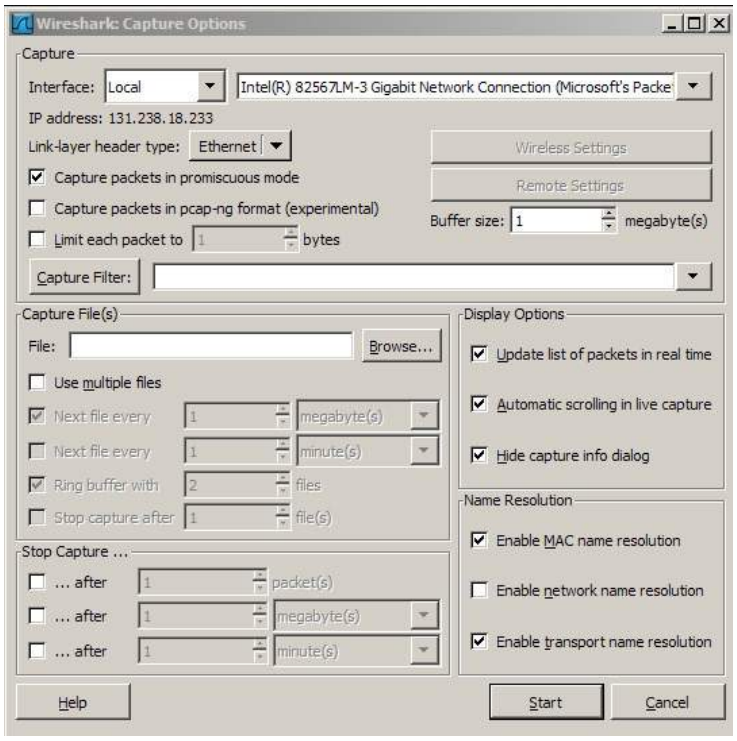
Figure 2. Wireshark Graphical User Interface

The best way to learn about any new piece of software is to try it out! We'll assume that your computer is connected to the Internet via a wired Ethernet interface. Do the following

1. Start up your favorite web browser, which will display your selected homepage.

2. Start up the Wireshark software. You will initially see a window similar to that shown in **Figure 2**.

3. To begin packet capture, select *Capture Options* under *Capture* (see Figure 2). This will cause the "Wireshark: Capture Options" window to be displayed, as shown in **Figure 3**.



Wireshark: Capture Options

Capture
- Interface: Local — Intel(R) 82567LM-3 Gigabit Network Connection (Microsoft's Packe
- IP address: 131.238.18.233
- Link-layer header type: Ethernet
- [✓] Capture packets in promiscuous mode
- [ ] Capture packets in pcap-ng format (experimental)
- [ ] Limit each packet to 1 bytes
- Buffer size: 1 megabyte(s)
- Capture Filter:

Capture File(s)
- File:
- [ ] Use multiple files
- [✓] Next file every 1 megabyte(s)
- [ ] Next file every 1 minute(s)
- [✓] Ring buffer with 2 files
- [ ] Stop capture after 1 file(s)

Stop Capture ...
- [ ] ... after 1 packet(s)
- [ ] ... after 1 megabyte(s)
- [ ] ... after 1 minute(s)

Display Options
- [✓] Update list of packets in real time
- [✓] Automatic scrolling in live capture
- [✓] Hide capture info dialog

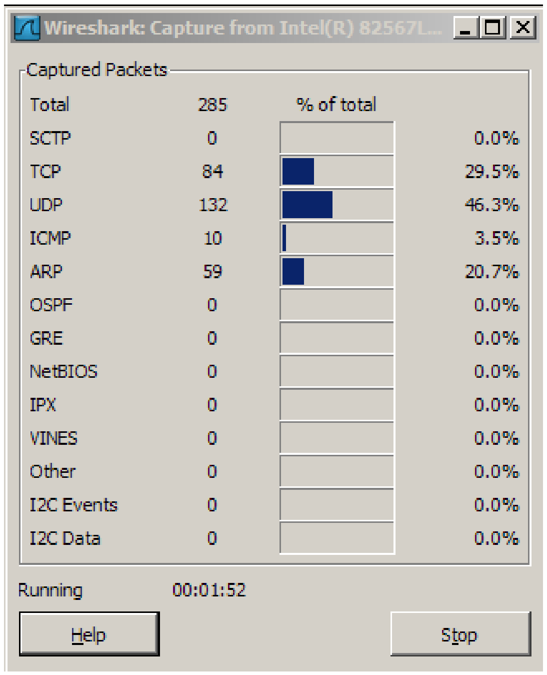
Name Resolution
- [✓] Enable MAC name resolution
- [ ] Enable network name resolution
- [✓] Enable transport name resolution

Figure 3. Wireshark Capture Options Window

4. You can use most of the default values in this window, but *uncheck* "*Hide capture info dialog*" under Display Options. In case your computer has more than one active network interface (e.g., if you have both a wireless and a wired Ethernet connection), you will need to select an interface that is being used to send and receive packets (mostly likely the wired interface). After selecting the network interface (or using the default interface chosen by Wireshark), click *Start*. Packet capture will now begin - all packets being sent/received from/by your computer are now being captured by Wireshark!

5. Once you begin packet capture, a packet capture summary window will appear, as shown in **Figure 4**. This window summarizes the number of packets of various types that are

being captured, and (importantly!) contains the *Stop* button that will allow you to stop packet capture. Don't stop packet capture yet.



Captured Packets

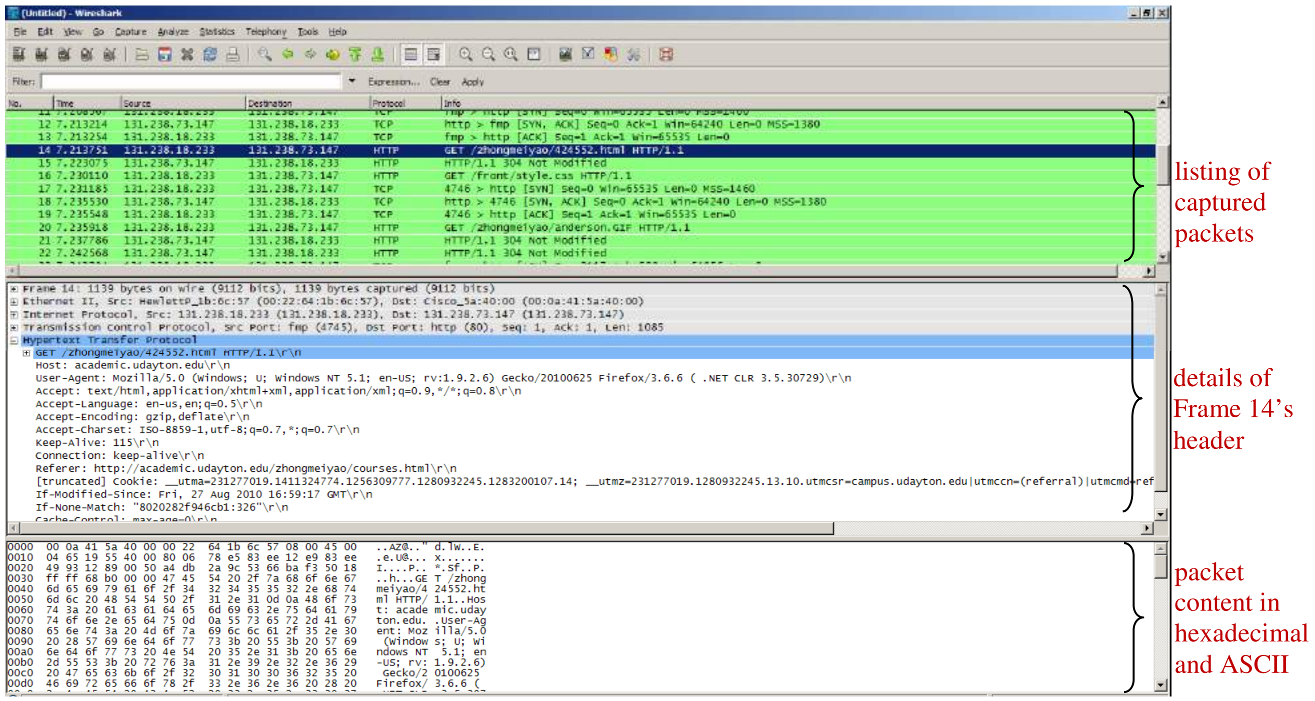
Total	285	% of total
SCTP	0	0.0%
TCP	84	29.5%
UDP	132	46.3%
ICMP	10	3.5%
ARP	59	20.7%
OSPF	0	0.0%
GRE	0	0.0%
NetBIOS	0	0.0%
IPX	0	0.0%
VINES	0	0.0%
Other	0	0.0%
I2C Events	0	0.0%
I2C Data	0	0.0%

Running 00:01:52

Figure 4. Wireshark Packet Capture Window

6. While Wireshark is running, enter the URL in your browser: http://academic.udayton.edu/zhongmeiyao/470.html and have that page displayed in your browser. In order to display this page, your browser will contact the HTTP server at academic.udayton.edu and exchange HTTP messages with the server in order to download this page. The Ethernet frames containing these HTTP messages will be captured by Wireshark.

7. After your browser has displayed the course web page, stop Wireshark packet capture by selecting stop in the Wireshark capture window (see Figure 4). This will cause the Wireshark capture window to disappear and the main Wireshark window to display all packets captured since you began packet capture. The main Wireshark window should now look similar to **Figure 5**. You now have live packet data that contains all protocol messages exchanged between your computer and other network entities! The HTTP message exchanges with the academic.udayton.edu web server should appear somewhere in the listing of packets captured. But there will be many other types of packets displayed as well (see, e.g., the many different protocol types shown in the *Protocol* column in **Figure 5**). Even though the only action you took was to download a web page, there were evidently many other protocols running on your computer that are unseen by the user. We'll learn much more about these protocols as we progress through the textbook! For now, you should just be aware that there is often much more going on than "meet's the eye"!


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
listing of captured packets
details of Frame 14's header
packet content in hexadecimal and ASCII

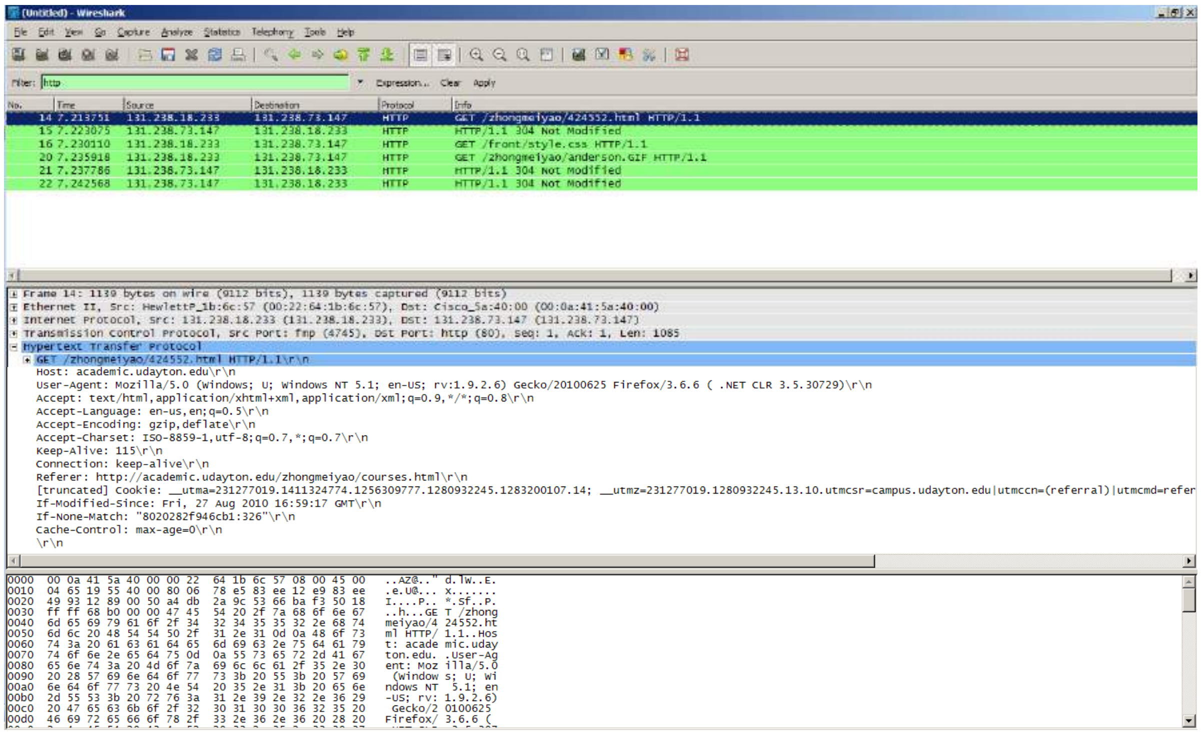
```
No.   Time      Source          Destination     Protocol Info
12 7.213214  131.238.73.147  131.238.18.233  TCP   http > fmp [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1380
13 7.213254  131.238.18.233  131.238.73.147  TCP   fmp > http [ACK] Seq=1 Ack=1 Win=65535 Len=0
14 7.213751  131.238.18.233  131.238.73.147  HTTP  GET /zhongmeiyao/424552.html HTTP/1.1
15 7.223075  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
16 7.230110  131.238.18.233  131.238.73.147  HTTP  GET /front/style.css HTTP/1.1
17 7.231185  131.238.18.233  131.238.73.147  TCP   4746 > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460
18 7.235530  131.238.73.147  131.238.18.233  TCP   http > 4746 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1380
19 7.235548  131.238.18.233  131.238.73.147  TCP   4746 > http [ACK] Seq=1 Ack=1 Win=65535 Len=0
20 7.235918  131.238.18.233  131.238.73.147  HTTP  GET /zhongmeiyao/anderson.GIF HTTP/1.1
21 7.237786  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
22 7.242568  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
```

```
⊞ Frame 14: 1139 bytes on wire (9112 bits), 1139 bytes captured (9112 bits)
⊞ Ethernet II, Src: HewlettP_1b:6c:57 (00:22:64:1b:6c:57), Dst: Cisco_5a:40:00 (00:0a:41:5a:40:00)
⊞ Internet Protocol, Src: 131.238.18.233 (131.238.18.233), Dst: 131.238.73.147 (131.238.73.147)
⊞ Transmission Control Protocol, Src Port: fmp (4745), Dst Port: http (80), Seq: 1, Ack: 1, Len: 1085
⊟ Hypertext Transfer Protocol
  ⊞ GET /zhongmeiyao/424552.html HTTP/1.1\r\n
    Host: academic.udayton.edu\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.6) Gecko/20100625 Firefox/3.6.6 ( .NET CLR 3.5.30729)\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 115\r\n
    Connection: keep-alive\r\n
    Referer: http://academic.udayton.edu/zhongmeiyao/courses.html\r\n
    [truncated] Cookie: __utma=231277019.1411324774.1256309777.1280932245.1283200107.14; __utmz=231277019.1280932245.13.10.utmcsr=campus.udayton.edu|utmccn=(referral)|utmcmd=ref
    If-Modified-Since: Fri, 27 Aug 2010 16:59:17 GMT\r\n
    If-None-Match: "8020282f946cb1:326"\r\n
```

```
0000  00 0a 41 5a 40 00 00 22  64 1b 6c 57 08 00 45 00   ..AZ@.." d.lW..E.
0010  04 65 19 55 40 00 80 06  78 e5 83 ee 12 e9 83 ee   .e.U@... x.......
0020  49 93 12 89 00 50 a4 db  2a 9c 53 66 ba f3 50 18   I....P.. *.Sf..P.
0030  ff ff 68 b0 00 00 47 45  54 20 2f 7a 68 6f 6e 67   ..h...GE T /zhong
0040  6d 65 69 79 61 6f 2f 34  32 34 35 35 32 2e 68 74   meiyao/4 24552.ht
0050  6d 6c 20 48 54 54 50 2f  31 2e 31 0d 0a 48 6f 73   ml HTTP/ 1.1..Hos
0060  74 3a 20 61 63 61 64 65  6d 69 63 2e 75 64 61 79   t: acade mic.uday
0070  74 6f 6e 2e 65 64 75 0d  0a 55 73 65 72 2d 41 67   ton.edu. .User-Ag
0080  65 6e 74 3a 20 4d 6f 7a  69 6c 6c 61 2f 35 2e 30   ent: Moz illa/5.0
0090  20 28 57 69 6e 64 6f 77  73 3b 20 55 3b 20 57 69    (Window s; U; Wi
00a0  6e 64 6f 77 73 20 4e 54  20 35 2e 31 3b 20 65 6e   ndows NT  5.1; en
00b0  2d 55 53 3b 20 72 76 3a  31 2e 39 2e 32 2e 36 29   -US; rv: 1.9.2.6)
00c0  20 47 65 63 6b 6f 2f 32  30 31 30 30 36 32 35 20    Gecko/2 0100625
00d0  46 69 72 65 66 6f 78 2f  33 2e 36 2e 36 20 28 20   Firefox/ 3.6.6 (
```

Figure 5. A snapshot of Wireshark lab

8. As shown in **Figure 6**, type in "http" (without the quotes, and in lower case – all protocol names are in lower case in Wireshark) into the display **filter** specification window at the top of the main Wireshark window. Then select *Apply* (to the right of where you entered "http"). This will cause only HTTP message to be displayed in the packet-listing window.


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter: http
Expression... Clear Apply

```
No.   Time      Source          Destination     Protocol Info
14 7.213751  131.238.18.233  131.238.73.147  HTTP  GET /zhongmeiyao/424552.html HTTP/1.1
15 7.223075  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
16 7.230110  131.238.18.233  131.238.73.147  HTTP  GET /front/style.css HTTP/1.1
20 7.235918  131.238.18.233  131.238.73.147  HTTP  GET /zhongmeiyao/anderson.GIF HTTP/1.1
21 7.237786  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
22 7.242568  131.238.73.147  131.238.18.233  HTTP  HTTP/1.1 304 Not Modified
```

```
⊞ Frame 14: 1139 bytes on wire (9112 bits), 1139 bytes captured (9112 bits)
⊞ Ethernet II, Src: HewlettP_1b:6c:57 (00:22:64:1b:6c:57), Dst: Cisco_5a:40:00 (00:0a:41:5a:40:00)
⊞ Internet Protocol, Src: 131.238.18.233 (131.238.18.233), Dst: 131.238.73.147 (131.238.73.147)
⊞ Transmission Control Protocol, Src Port: fmp (4745), Dst Port: http (80), Seq: 1, Ack: 1, Len: 1085
⊟ Hypertext Transfer Protocol
  ⊞ GET /zhongmeiyao/424552.html HTTP/1.1\r\n
    Host: academic.udayton.edu\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.6) Gecko/20100625 Firefox/3.6.6 ( .NET CLR 3.5.30729)\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 115\r\n
    Connection: keep-alive\r\n
    Referer: http://academic.udayton.edu/zhongmeiyao/courses.html\r\n
    [truncated] Cookie: __utma=231277019.1411324774.1256309777.1280932245.1283200107.14; __utmz=231277019.1280932245.13.10.utmcsr=campus.udayton.edu|utmccn=(referral)|utmcmd=refer
    If-Modified-Since: Fri, 27 Aug 2010 16:59:17 GMT\r\n
    If-None-Match: "8020282f946cb1:326"\r\n
    Cache-Control: max-age=0\r\n
    \r\n
```

```
0000  00 0a 41 5a 40 00 00 22  64 1b 6c 57 08 00 45 00   ..AZ@.." d.lW..E.
0010  04 65 19 55 40 00 80 06  78 e5 83 ee 12 e9 83 ee   .e.U@... x.......
0020  49 93 12 89 00 50 a4 db  2a 9c 53 66 ba f3 50 18   I....P.. *.Sf..P.
0030  ff ff 68 b0 00 00 47 45  54 20 2f 7a 68 6f 6e 67   ..h...GE T /zhong
0040  6d 65 69 79 61 6f 2f 34  32 34 35 35 32 2e 68 74   meiyao/4 24552.ht
0050  6d 6c 20 48 54 54 50 2f  31 2e 31 0d 0a 48 6f 73   ml HTTP/ 1.1..Hos
0060  74 3a 20 61 63 61 64 65  6d 69 63 2e 75 64 61 79   t: acade mic.uday
0070  74 6f 6e 2e 65 64 75 0d  0a 55 73 65 72 2d 41 67   ton.edu. .User-Ag
0080  65 6e 74 3a 20 4d 6f 7a  69 6c 6c 61 2f 35 2e 30   ent: Moz illa/5.0
0090  20 28 57 69 6e 64 6f 77  73 3b 20 55 3b 20 57 69    (Window s; U; Wi
00a0  6e 64 6f 77 73 20 4e 54  20 35 2e 31 3b 20 65 6e   ndows NT  5.1; en
00b0  2d 55 53 3b 20 72 76 3a  31 2e 39 2e 32 2e 36 29   -US; rv: 1.9.2.6)
00c0  20 47 65 63 6b 6f 2f 32  30 31 30 30 36 32 35 20    Gecko/2 0100625
00d0  46 69 72 65 66 6f 78 2f  33 2e 36 2e 36 20 28 20   Firefox/ 3.6.6 (
```

Figure 6. Wireshark display after step 8

9. Select the first http message shown in the packet-listing window. This should be the HTTP GET message that was sent from your computer to the academic.udayton.edu HTTP server. When you select the HTTP GET message, the Ethernet frame, IP datagram, TCP segment, and HTTP message header information will be displayed in the packet-header window. By clicking on right-pointing and down-pointing arrowsheads to the left side of the packet details window, *minimize* the amount of Frame, Ethernet, Internet Protocol, and Transmission Control Protocol information displayed. *Maximize* the amount information displayed about the HTTP protocol. (Note, in particular, the minimized amount of protocol information for all protocols except HTTP, and the maximized amount of protocol information for HTTP in the packet-header window).

10. Exit Wireshark

Congratulations! You've now understand how to use wireshark.

2.3. TCP

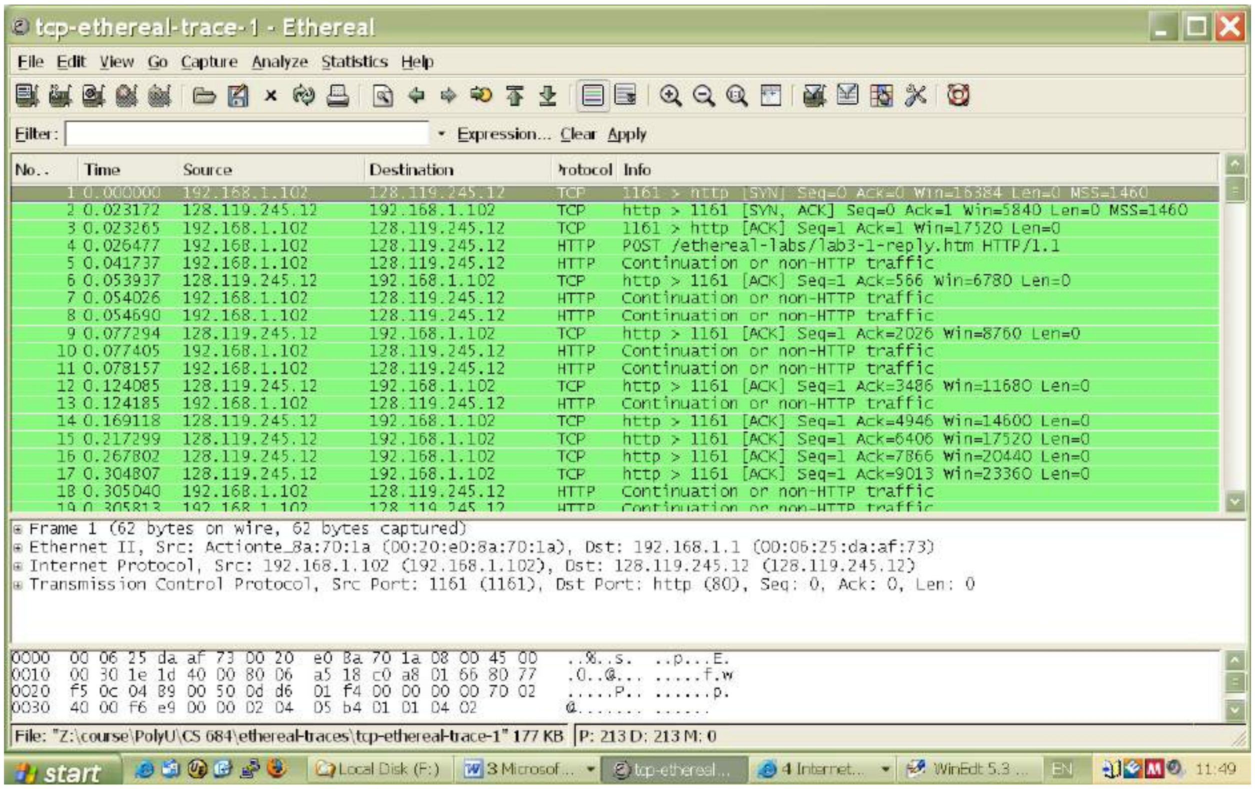
The following questions are based on the trace file *tcp-ethereal-trace-1* in in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip

Answer the following questions for the TCP segments:

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

2. What is the IP address and port number used by gaia.cs.umass.edu to receive the file.

3. If you did this problem on your own computer, you'll have your own solution



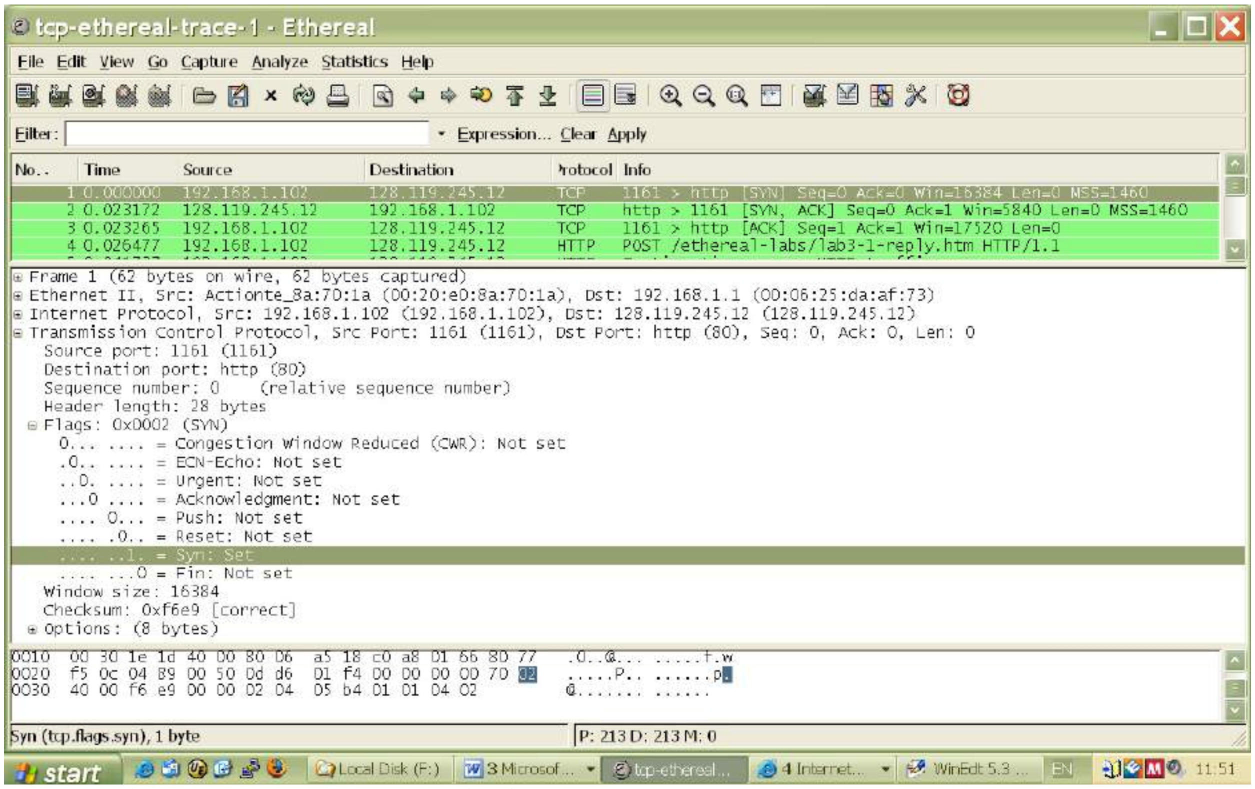
tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply
start
Local Disk (F:)
3 Microsof...
tcp-ethereal...
4 Internet...
WinEdt 5.3 ...
EN
11:49

```
No. .  Time       Source            Destination       Protocol Info
     1 0.000000   192.168.1.102     128.119.245.12    TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
     2 0.023172   128.119.245.12    192.168.1.102     TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
     3 0.023265   192.168.1.102     128.119.245.12    TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
     4 0.026477   192.168.1.102     128.119.245.12    HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
     5 0.041737   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
     6 0.053937   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
     7 0.054026   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
     8 0.054690   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
     9 0.077294   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
    10 0.077405   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
    11 0.078157   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
    12 0.124085   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
    13 0.124185   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
    14 0.169118   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
    15 0.217299   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
    16 0.267802   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
    17 0.304807   128.119.245.12    192.168.1.102     TCP      http > 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
    18 0.305040   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic
    19 0.305813   192.168.1.102     128.119.245.12    HTTP     Continuation or non-HTTP traffic

⊞ Frame 1 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: 192.168.1.1 (00:06:25:da:af:73)
⊞ Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
⊞ Transmission Control Protocol, Src Port: 1161 (1161), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0

0000  00 06 25 da af 73 00 20  e0 8a 70 1a 08 00 45 00   ..%..s.  ..p...E.
0010  00 30 1e 1d 40 00 80 06  a5 18 c0 a8 01 66 80 77   .0..@... .....f.w
0020  f5 0c 04 89 00 50 0d d6  01 f4 00 00 00 00 70 02   .....P.. ......p.
0030  40 00 f6 e9 00 00 02 04  05 b4 01 01 04 02         @....... ......

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 0
```

Figure 1: IP addresses and TCP port numbers of the client computer (source) and gaia.cs.umass.edu

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?

The SYN flag is set to 1 and it indicates that this segment is a SYN segment.



```
No..  Time       Source           Destination      Protocol Info
    1 0.000000   192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
    2 0.023172   128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
    3 0.023265   192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
    4 0.026477   192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
```

```
⊞ Frame 1 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: 192.168.1.1 (00:06:25:da:af:73)
⊞ Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
⊟ Transmission Control Protocol, Src Port: 1161 (1161), Dst Port: http (80), Seq: 0, Ack: 0, Len: 0
    Source port: 1161 (1161)
    Destination port: http (80)
    Sequence number: 0    (relative sequence number)
    Header length: 28 bytes
  ⊟ Flags: 0x0002 (SYN)
      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
    Window size: 16384
    Checksum: 0xf6e9 [correct]
  ⊞ Options: (8 bytes)
```

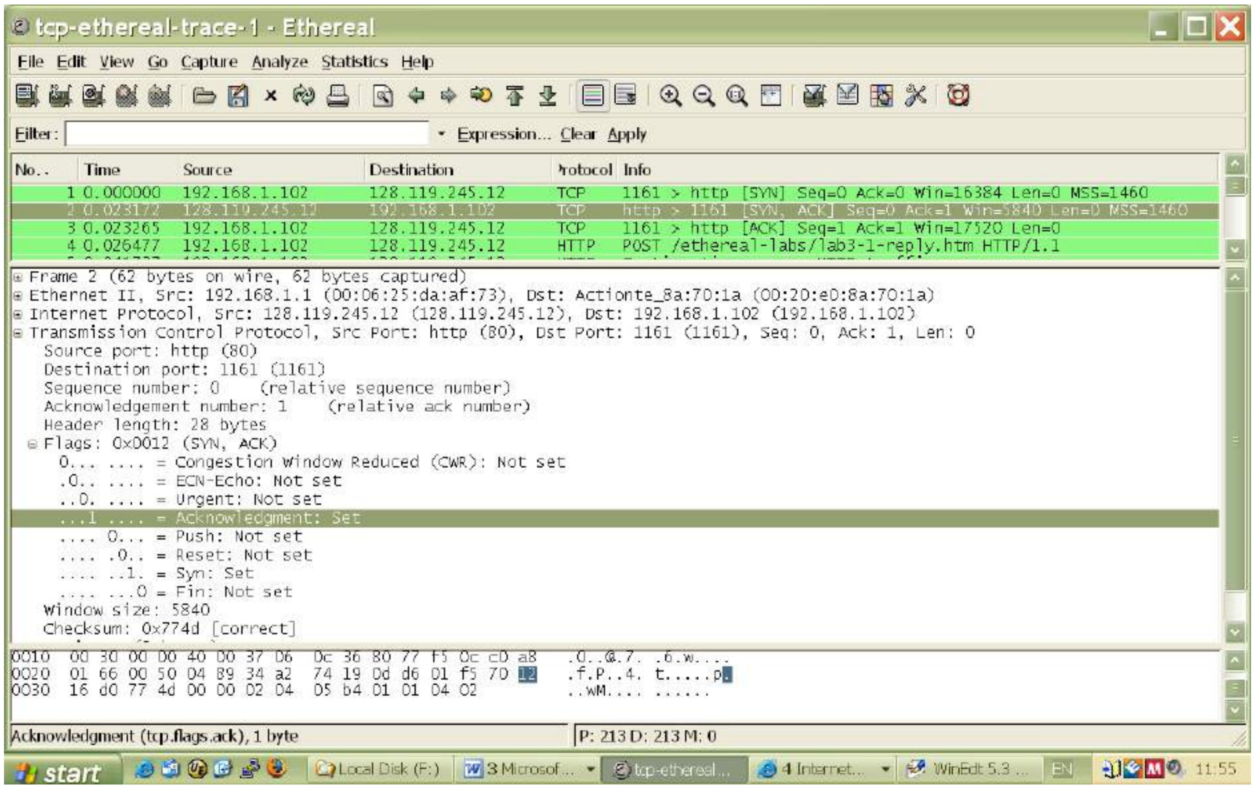
```
0010  00 30 1e 1d 40 00 80 06  a5 18 c0 a8 01 66 80 77   .0..@... .....f.w
0020  f5 0c 04 89 00 50 0d d6  01 f4 00 00 00 00 70 02   .....P.. ......p.
0030  40 00 f6 e9 00 00 02 04  05 b4 01 01 04 02         @....... ......
```

Syn (tcp.flags.syn), 1 byte   P: 213 D: 213 M: 0

tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter: Expression... Clear Apply
start
Local Disk (F:)
3 Microsof...
tcp-ethereal...
4 Internet...
WinEdt 5.3 ...
EN
11:51

Figure 2: Sequence number of the TCP SYN segment

5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the ACKnowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?


tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply

```
No..  Time      Source           Destination      Protocol Info
    1 0.000000  192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
    2 0.023172  128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
    3 0.023265  192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
    4 0.026477  192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
```

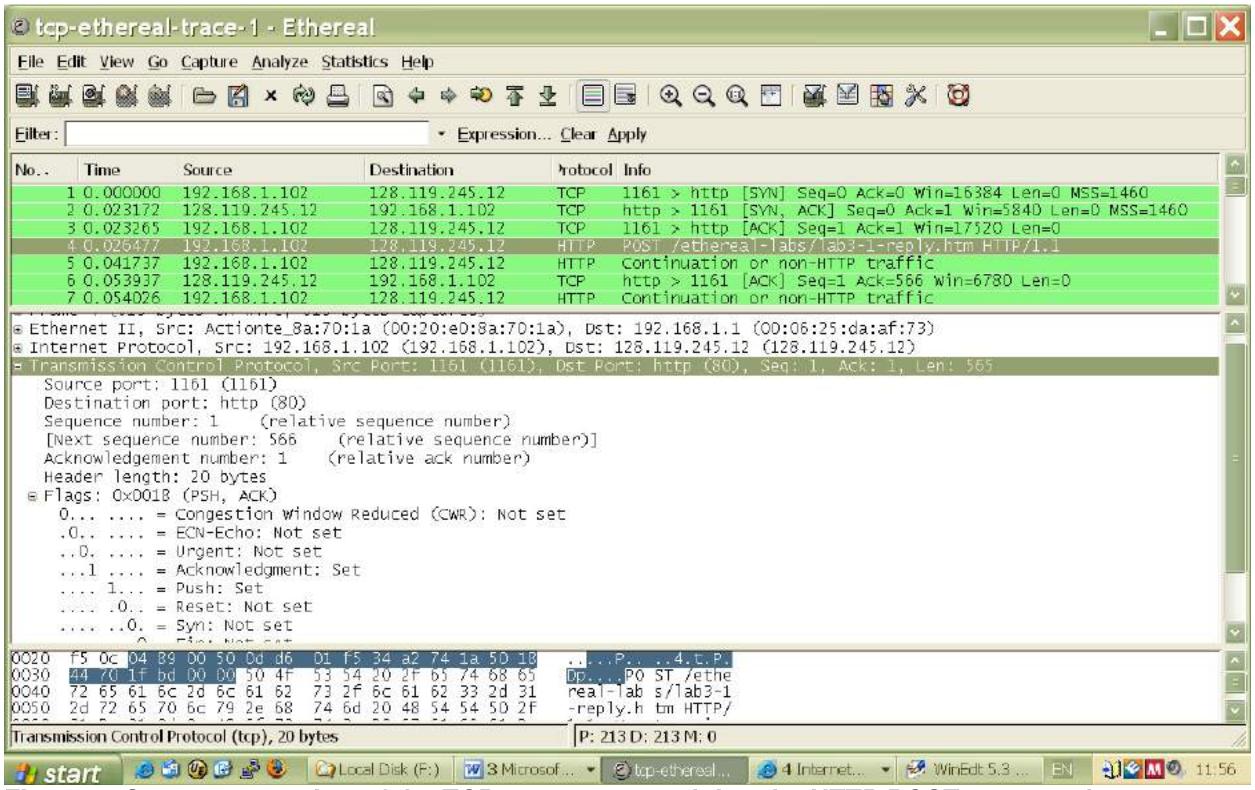
```
⊞ Frame 2 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: 192.168.1.1 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
⊞ Internet Protocol, Src: 128.119.245.12 (128.119.245.12), Dst: 192.168.1.102 (192.168.1.102)
⊟ Transmission Control Protocol, Src Port: http (80), Dst Port: 1161 (1161), Seq: 0, Ack: 1, Len: 0
    Source port: http (80)
    Destination port: 1161 (1161)
    Sequence number: 0    (relative sequence number)
    Acknowledgement number: 1    (relative ack number)
    Header length: 28 bytes
  ⊟ Flags: 0x0012 (SYN, ACK)
      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
    Window size: 5840
    Checksum: 0x774d [correct]
```

```
0010  00 30 00 00 40 00 37 06  0c 36 80 77 f5 0c c0 a8   .0..@.7. .6.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 d0 77 4d 00 00 02 04  05 b4 01 01 04 02         ..wM.... ......
```

```
Acknowledgment (tcp.flags.ack), 1 byte        P: 213 D: 213 M: 0
```

Figure 3: Sequence number and Acknowledgement number of the SYNACK 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.



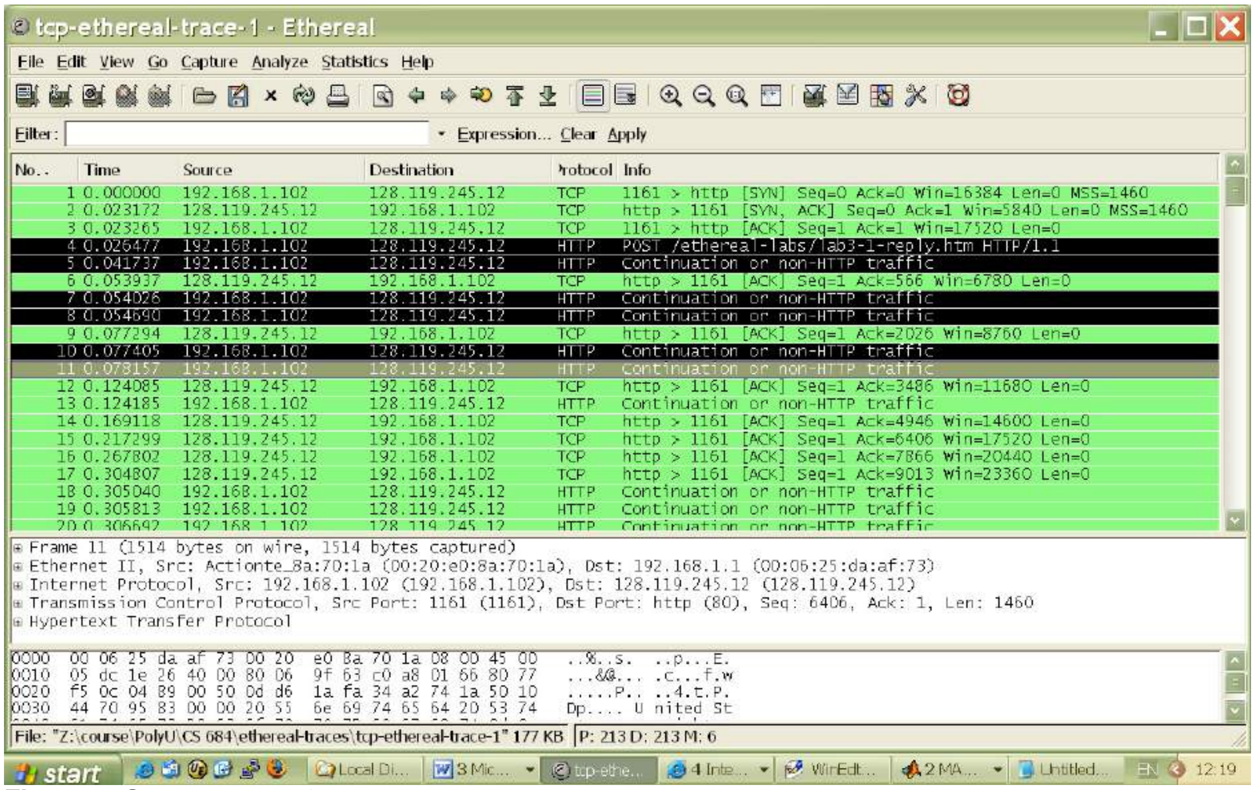
tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter: Expression... Clear Apply
start
Local Disk (F:)
3 Microsof...
tcp-ethereal...
4 Internet...
WinEdt 5.3 ...
EN
11:56

```
No..  Time      Source           Destination      Protocol Info
    1 0.000000  192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
    2 0.023172  128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
    3 0.023265  192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
    4 0.026477  192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
    5 0.041737  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    6 0.053937  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
    7 0.054026  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic

⊞ Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: 192.168.1.1 (00:06:25:da:af:73)
⊞ Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
⊟ Transmission Control Protocol, Src Port: 1161 (1161), Dst Port: http (80), Seq: 1, Ack: 1, Len: 565
    Source port: 1161 (1161)
    Destination port: http (80)
    Sequence number: 1    (relative sequence number)
    [Next sequence number: 566    (relative sequence number)]
    Acknowledgement number: 1    (relative ack number)
    Header length: 20 bytes
  ⊟ Flags: 0x0018 (PSH, ACK)
      0... .... = Congestion Window Reduced (CWR): Not set
      .0.. .... = ECN-Echo: Not set
      ..0. .... = Urgent: Not set
      ...1 .... = Acknowledgment: Set
      .... 1... = Push: Set
      .... .0.. = Reset: Not set
      .... ..0. = Syn: Not set

0020  f5 0c 04 b9 00 50 0d d6  01 f5 34 a2 74 1a 50 18   .....P.. ..4.t.P.
0030  44 70 1f bd 00 00 50 4f  53 54 20 2f 65 74 68 65   Dp....PO ST /ethe
0040  72 65 61 6c 2d 6c 61 62  73 2f 6c 61 62 33 2d 31   real-lab s/lab3-1
0050  2d 72 65 70 6c 79 2e 68  74 6d 20 48 54 54 50 2f   -reply.h tm HTTP/

Transmission Control Protocol (tcp), 20 bytes    P: 213 D: 213 M: 0
```

Figure 4: Sequence number of the TCP segment containing the HTTP POST command

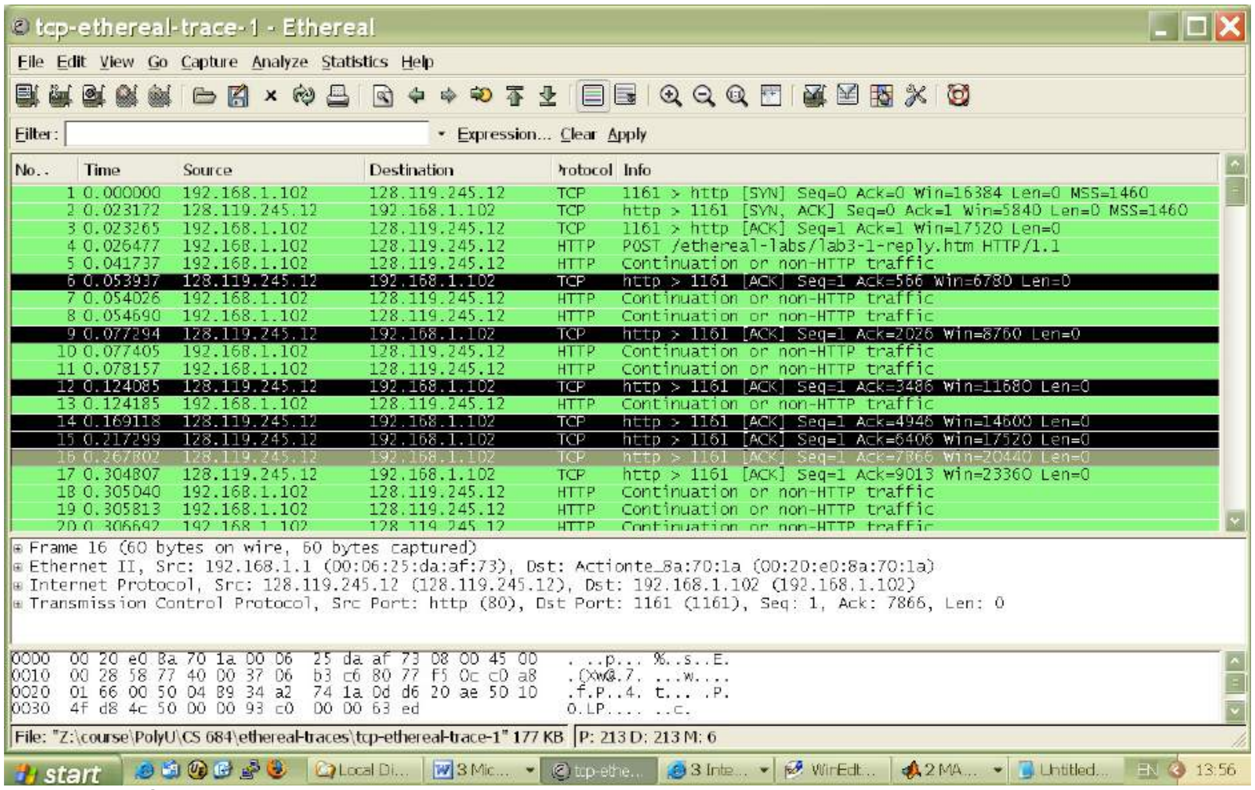
7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? 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 page 237 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 237 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.



```
No. .  Time       Source           Destination      Protocol Info
     1 0.000000   192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
     2 0.023172   128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
     3 0.023265   192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
     4 0.026477   192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
     5 0.041737   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     6 0.053937   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
     7 0.054026   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     8 0.054690   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     9 0.077294   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
    10 0.077405   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    11 0.078157   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    12 0.124085   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
    13 0.124185   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    14 0.169118   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
    15 0.217299   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
    16 0.267802   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
    17 0.304807   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
    18 0.305040   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    19 0.305813   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    20 0.306692   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic

⊞ Frame 11 (1514 bytes on wire, 1514 bytes captured)
⊞ Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: 192.168.1.1 (00:06:25:da:af:73)
⊞ Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
⊞ Transmission Control Protocol, Src Port: 1161 (1161), Dst Port: http (80), Seq: 6406, Ack: 1, Len: 1460
⊞ Hypertext Transfer Protocol

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 34 a2 74 1a 50 10   .....P.. ..4.t.P.
0030  44 70 95 83 00 00 20 55  6e 69 74 65 64 20 53 74   Dp.... U nited St

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 6
```

Figure 5: Segments 1 – 6



```
No. .  Time       Source           Destination      Protocol Info
     1 0.000000   192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
     2 0.023172   128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
     3 0.023265   192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
     4 0.026477   192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
     5 0.041737   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     6 0.053937   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
     7 0.054026   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     8 0.054690   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
     9 0.077294   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
    10 0.077405   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    11 0.078157   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    12 0.124085   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
    13 0.124185   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    14 0.169118   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
    15 0.217299   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
    16 0.267802   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
    17 0.304807   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
    18 0.305040   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    19 0.305813   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    20 0.306692   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic

⊞ Frame 16 (60 bytes on wire, 60 bytes captured)
⊞ Ethernet II, Src: 192.168.1.1 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
⊞ Internet Protocol, Src: 128.119.245.12 (128.119.245.12), Dst: 192.168.1.102 (192.168.1.102)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 1161 (1161), Seq: 1, Ack: 7866, Len: 0

0000  00 20 e0 8a 70 1a 00 06  25 da af 73 08 00 45 00   . ..p... %..s..E.
0010  00 28 58 77 40 00 37 06  b3 c6 80 77 f5 0c c0 a8   .(Xw@.7. ...w....
0020  01 66 00 50 04 89 34 a2  74 1a 0d d6 20 ae 50 10   .f.P..4. t... .P.
0030  4f d8 4c 50 00 00 93 c0  00 00 63 ed               O.LP.... ..c.

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 6
```

Figure 6: ACKs of segments 1 – 6

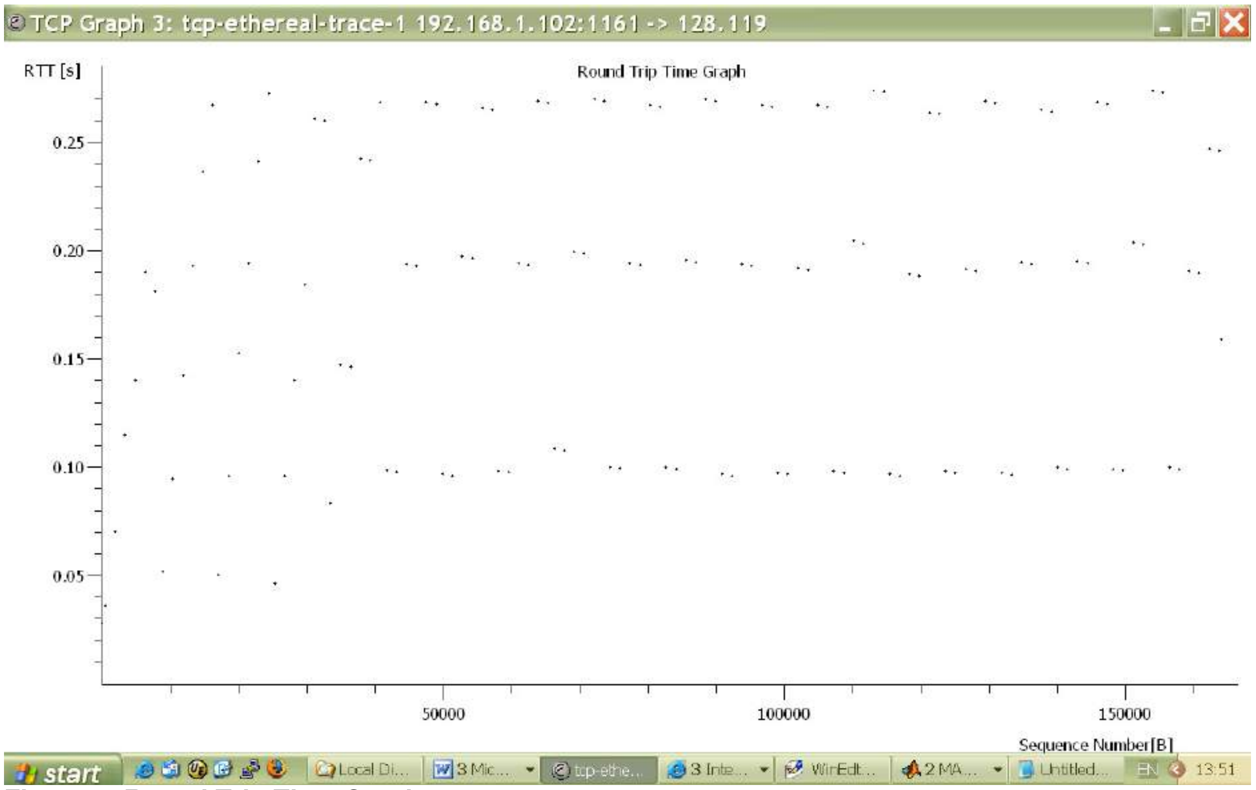
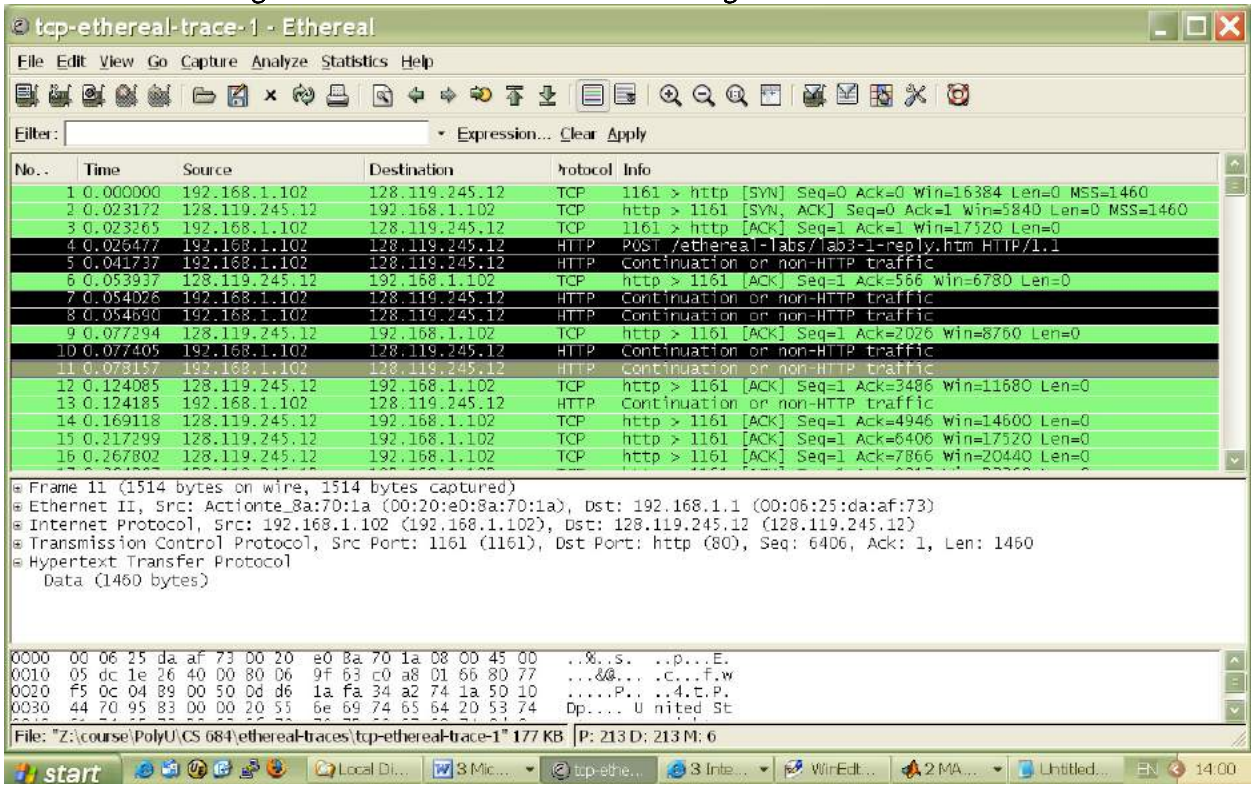


Figure 7: Round Trip Time Graph

8. What is the length of each of the first six TCP segments?


tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply
start
Local Di...
3 Mic...
tcp-ethe...
3 Inte...
WinEdt...
2 MA...
Untitled...
EN
14:00

```
No..  Time       Source           Destination      Protocol Info
    1 0.000000   192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
    2 0.023172   128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
    3 0.023265   192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
    4 0.026477   192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
    5 0.041737   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    6 0.053937   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
    7 0.054026   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    8 0.054690   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
    9 0.077294   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
   10 0.077405   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   11 0.078157   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   12 0.124085   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
   13 0.124185   192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   14 0.169118   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
   15 0.217299   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
   16 0.267802   128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
```

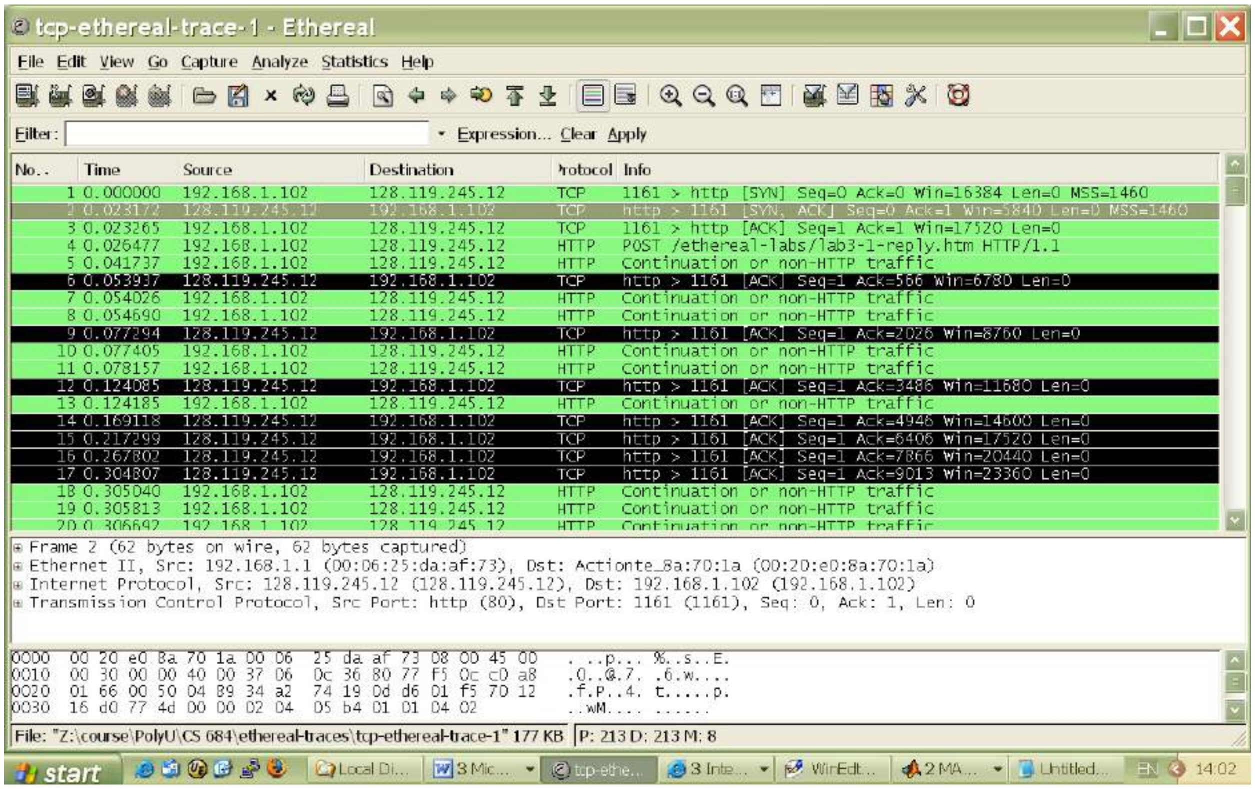
```
⊞ Frame 11 (1514 bytes on wire, 1514 bytes captured)
⊞ Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: 192.168.1.1 (00:06:25:da:af:73)
⊞ Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
⊞ Transmission Control Protocol, Src Port: 1161 (1161), Dst Port: http (80), Seq: 6406, Ack: 1, Len: 1460
⊟ Hypertext Transfer Protocol
    Data (1460 bytes)
```

```
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 34 a2 74 1a 50 10   .....P.. ..4.t.P.
0030  44 70 95 83 00 00 20 55  6e 69 74 65 64 20 53 74   Dp.... U nited St
```

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 6

Figure 8: Lengths of segments 1 – 6

9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?



```
No..  Time      Source           Destination      Protocol Info
   1 0.000000  192.168.1.102    128.119.245.12   TCP      1161 > http [SYN] Seq=0 Ack=0 Win=16384 Len=0 MSS=1460
   2 0.023172  128.119.245.12   192.168.1.102    TCP      http > 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
   3 0.023265  192.168.1.102    128.119.245.12   TCP      1161 > http [ACK] Seq=1 Ack=1 Win=17520 Len=0
   4 0.026477  192.168.1.102    128.119.245.12   HTTP     POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
   5 0.041737  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   6 0.053937  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
   7 0.054026  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   8 0.054690  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
   9 0.077294  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
  10 0.077405  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
  11 0.078157  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
  12 0.124085  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
  13 0.124185  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
  14 0.169118  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
  15 0.217299  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
  16 0.267802  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
  17 0.304807  128.119.245.12   192.168.1.102    TCP      http > 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
  18 0.305040  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
  19 0.305813  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic
  20 0.306692  192.168.1.102    128.119.245.12   HTTP     Continuation or non-HTTP traffic

⊞ Frame 2 (62 bytes on wire, 62 bytes captured)
⊞ Ethernet II, Src: 192.168.1.1 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
⊞ Internet Protocol, Src: 128.119.245.12 (128.119.245.12), Dst: 192.168.1.102 (192.168.1.102)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 1161 (1161), Seq: 0, Ack: 1, Len: 0

0000  00 20 e0 8a 70 1a 00 06  25 da af 73 08 00 45 00   . ..p... %..s..E.
0010  00 30 00 00 40 00 37 06  0c 36 80 77 f5 0c c0 a8   .0..@.7. .6.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 d0 77 4d 00 00 02 04  05 b4 01 01 04 02         ..wM.... ......

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 8
```

tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply
start
Local Di...
3 Mic...
tcp-ethe...
3 Inte...
WinEdt...
2 MA...
Untitled...
EN
14:02

Figure 9: Minimum receive window advertised at gaia.cs.umass.edu (packet No. 2)

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

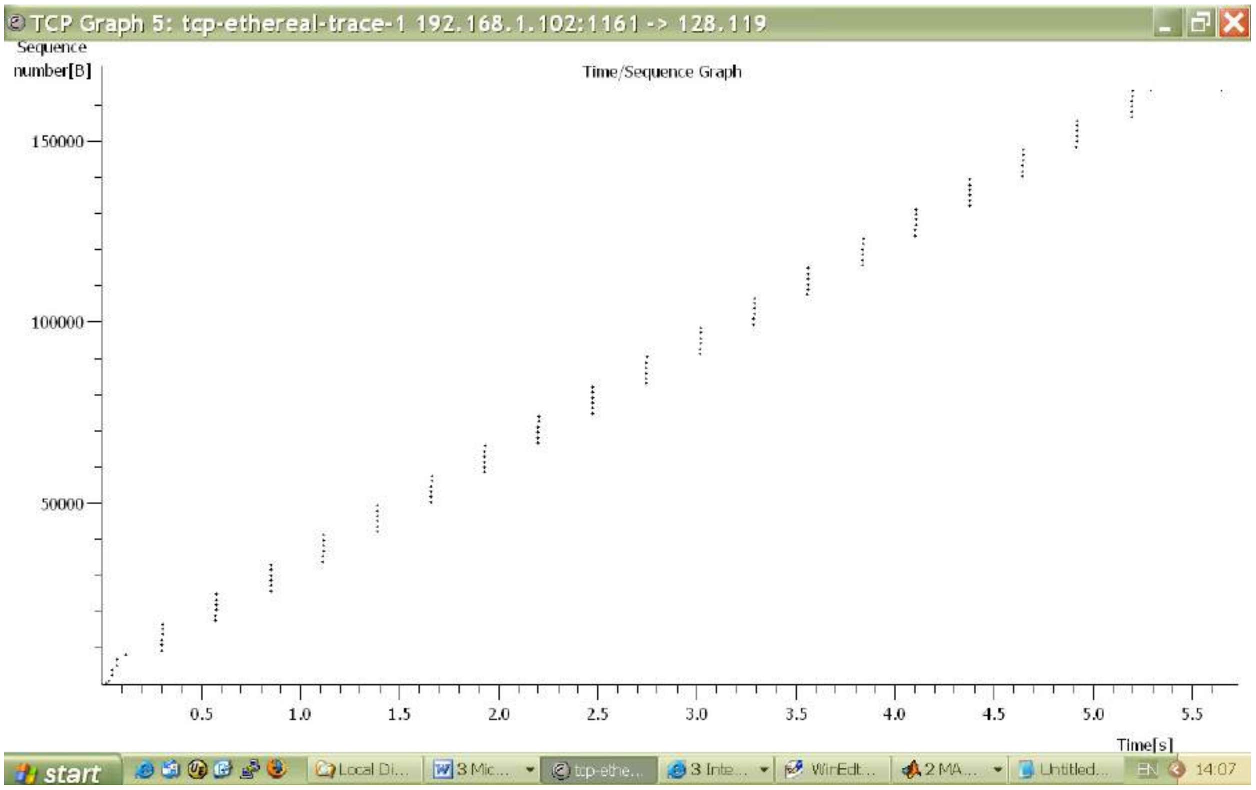
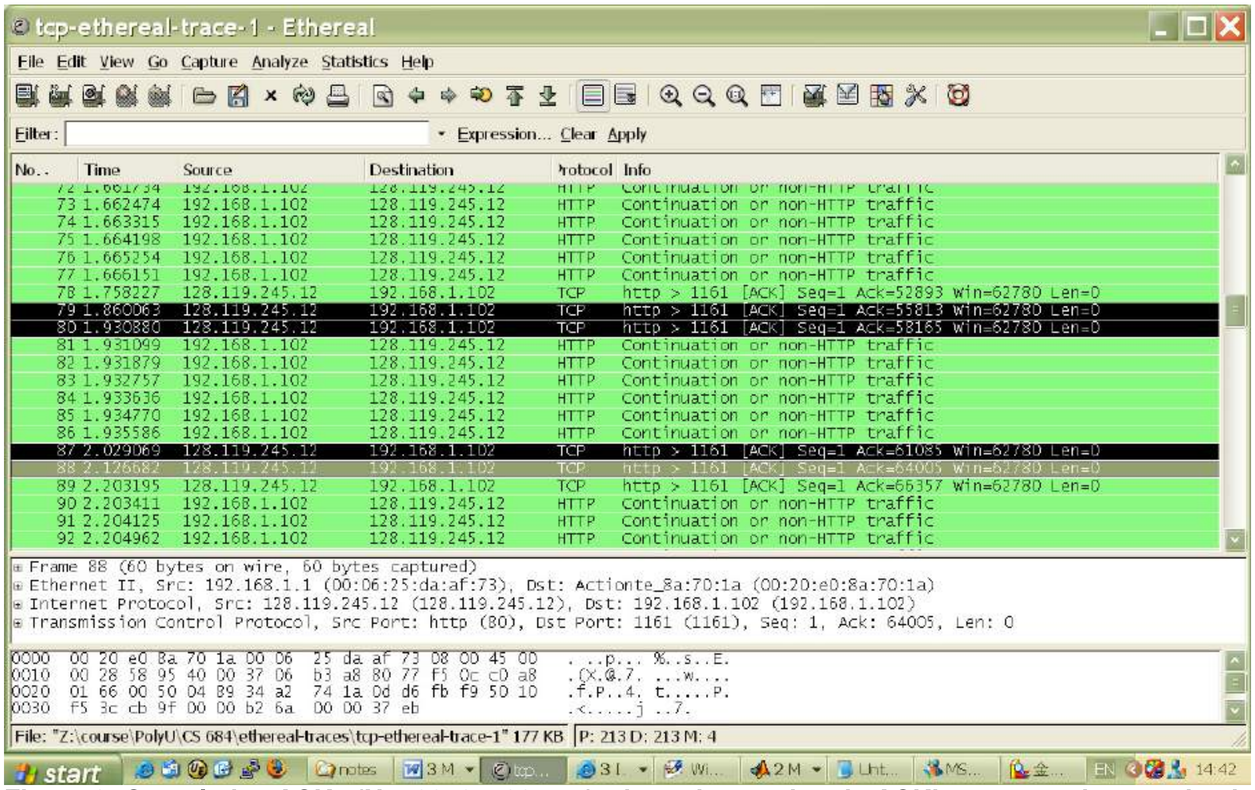


Figure 10: Sequence numbers of the segments from the source (`192.168.1.102`) to the destination (`128.119.245.12`)

11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 247 in the text).


tcp-ethereal-trace-1 - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter: Expression... Clear Apply

```
No. . Time      Source          Destination     Protocol Info
   72 1.661734  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   73 1.662474  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   74 1.663315  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   75 1.664198  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   76 1.665254  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   77 1.666151  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   78 1.758227  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=52893 Win=62780 Len=0
   79 1.860063  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=55813 Win=62780 Len=0
   80 1.930880  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=58165 Win=62780 Len=0
   81 1.931099  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   82 1.931879  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   83 1.932757  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   84 1.933636  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   85 1.934770  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   86 1.935586  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   87 2.029069  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=61085 Win=62780 Len=0
   88 2.126682  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=64005 Win=62780 Len=0
   89 2.203195  128.119.245.12  192.168.1.102   TCP      http > 1161 [ACK] Seq=1 Ack=66357 Win=62780 Len=0
   90 2.203411  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   91 2.204125  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic
   92 2.204962  192.168.1.102   128.119.245.12  HTTP     Continuation or non-HTTP traffic

⊞ Frame 88 (60 bytes on wire, 60 bytes captured)
⊞ Ethernet II, Src: 192.168.1.1 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
⊞ Internet Protocol, Src: 128.119.245.12 (128.119.245.12), Dst: 192.168.1.102 (192.168.1.102)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 1161 (1161), Seq: 1, Ack: 64005, Len: 0

0000  00 20 e0 8a 70 1a 00 06  25 da af 73 08 00 45 00   . ..p... %..s..E.
0010  00 28 58 95 40 00 37 06  b3 a8 80 77 f5 0c c0 a8   .(X.@.7. ...w....
0020  01 66 00 50 04 89 34 a2  74 1a 0d d6 fb f9 50 10   .f.P..4. t.....P.
0030  f5 3c cb 9f 00 00 b2 6a  00 00 37 eb              .<.....j ..7.

File: "Z:\course\PolyU\CS 684\ethereal-traces\tcp-ethereal-trace-1" 177 KB   P: 213 D: 213 M: 4
```

Figure 8: Cumulative ACKs (No. 80, 87, 88, etc) where the receiver is ACKing every other received segment.

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