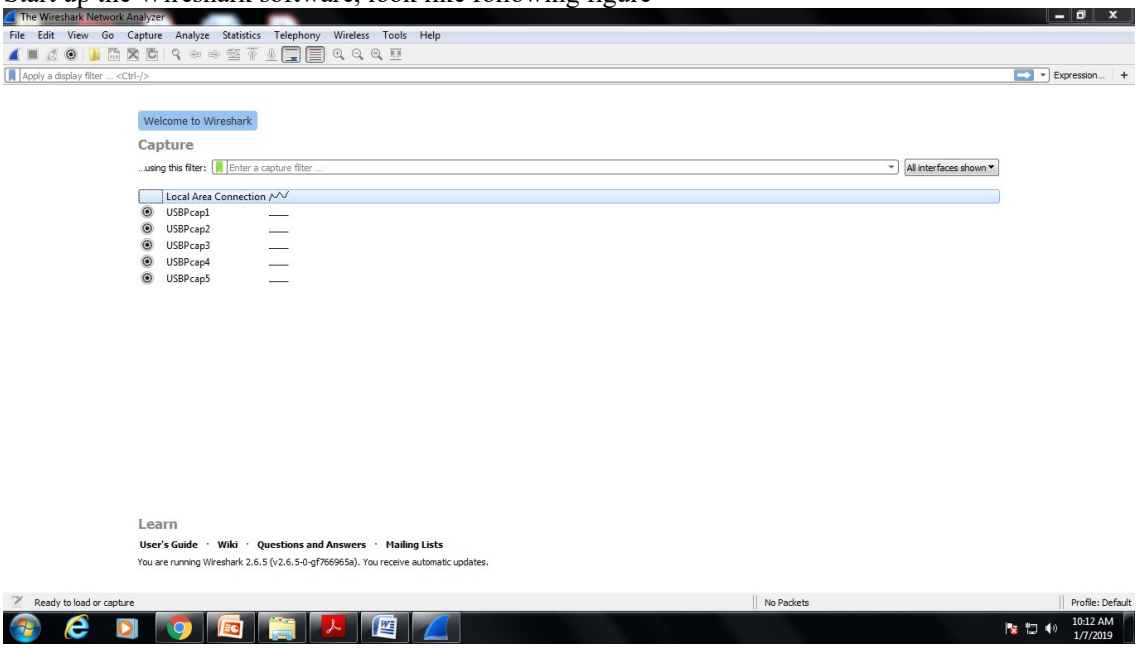
Computer Networks (CN)

"Network Analysis using Wireshark"

The best way to learn wireshark is to try it out! We'll assume that your computer is connected to the Internet via a wired Ethernet interface. If you use a laptop that only has wireless connection, this will also be fine with the following experiments. Do the following:

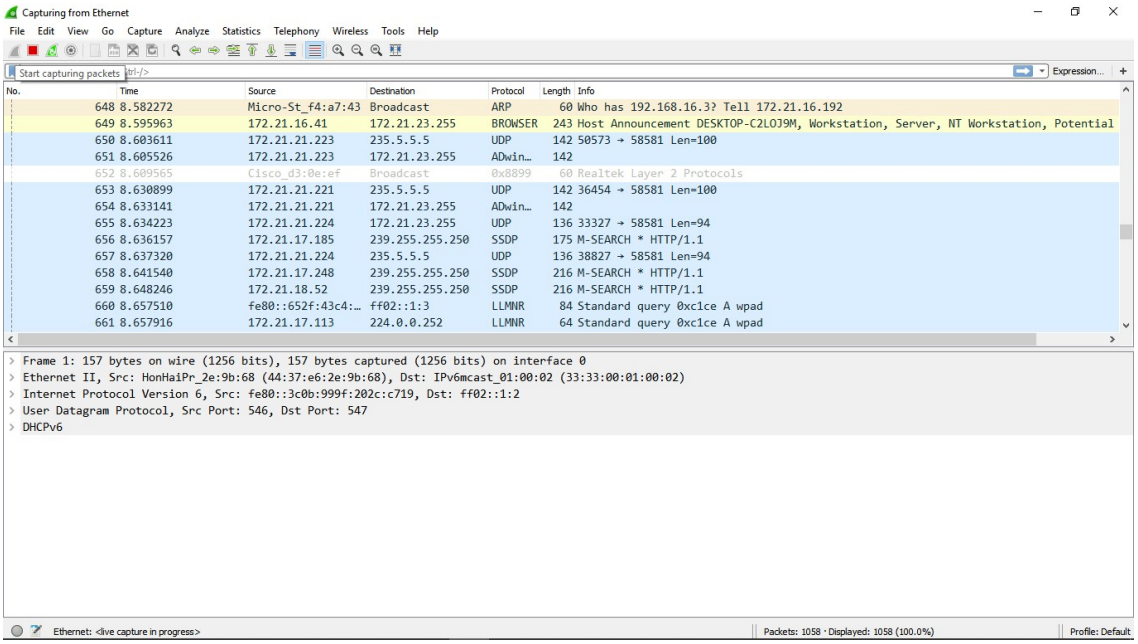
1. Start up your favorite web browser, which will display your selected homepage.
2. If you are using a proxy (especially a host-based one), disable it if possible to examine uncached network traffic.
3. Also better to clear browser cache, cookies if you have previously displayed this page.
4. Disable anti-virus protection software before your own IP address will show up in captured data.
5. Start up the Wireshark software, look like following figure

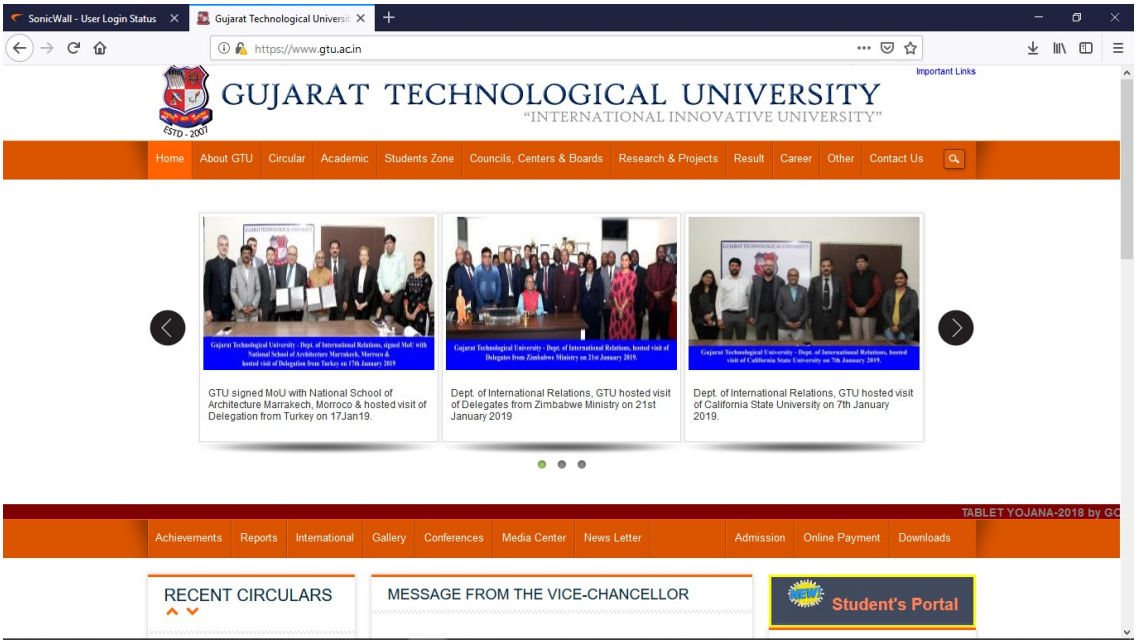


6. To begin packet capture, select the Capture pull down menu and select Options. This will cause the "Wireshark: Capture Options" window to be displayed
7. You can use most of the default values in this window. The network interfaces (i.e., the physical connections) that your computer has to the network will be shown in the Interface pull down menu at the top of the Capture Options window. 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 live packets being sent/received from/by your computer are now being captured by Wireshark!

8. Once you begin packet capture, a packet capture summary window will appear. 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.
9. While Wireshark is running, enter the URL: http://gtu.ac.in
and have that page displayed in your browser. In order to display this page, your browser will contact the HTTP server at http://gtu.ac.in 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.

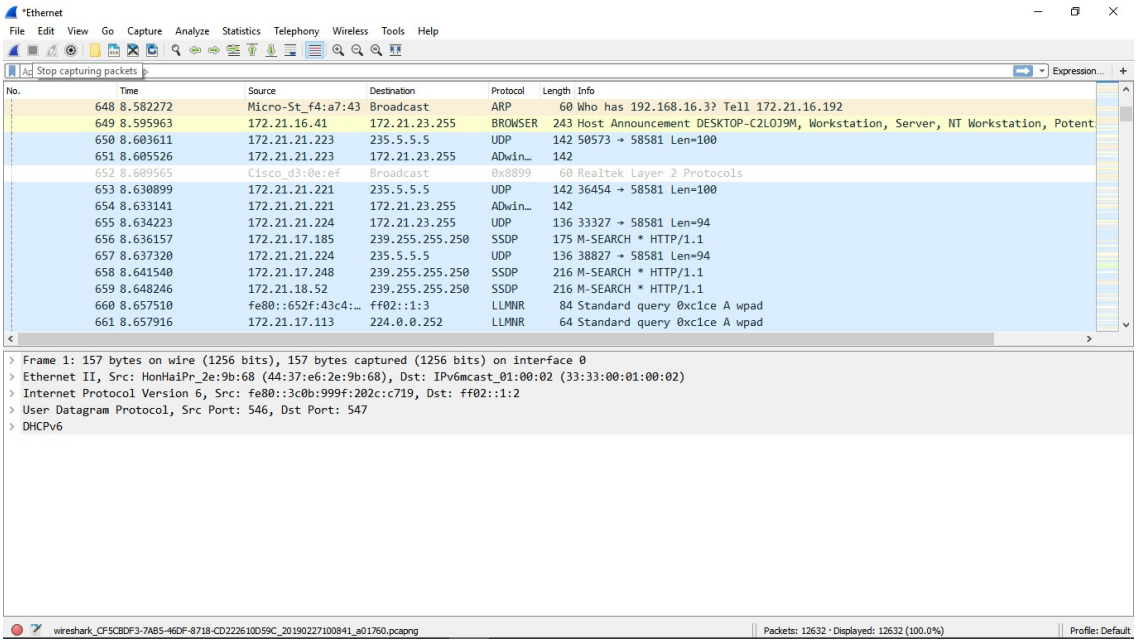
Ans:





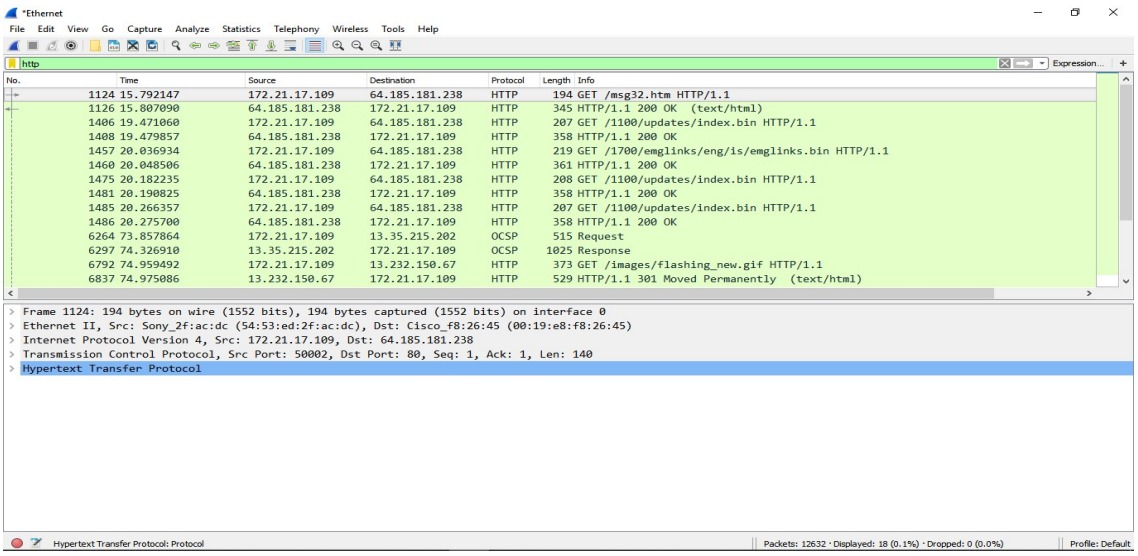
10. After your browser has displayed the web page, stop Wireshark packet capture by selecting stop in the Wireshark capture window. This will cause the Wireshark capture window to disappear and the main Wireshark window to display all packets captured since you began packet capture. You now have live packet data that contains all protocol messages exchanged between your computer and other network entities! The HTTP message exchanges with the http://gtu.ac.in should appear somewhere in the listing of packets captured. But there will be many other types of packets displayed as well. 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.

Ans:



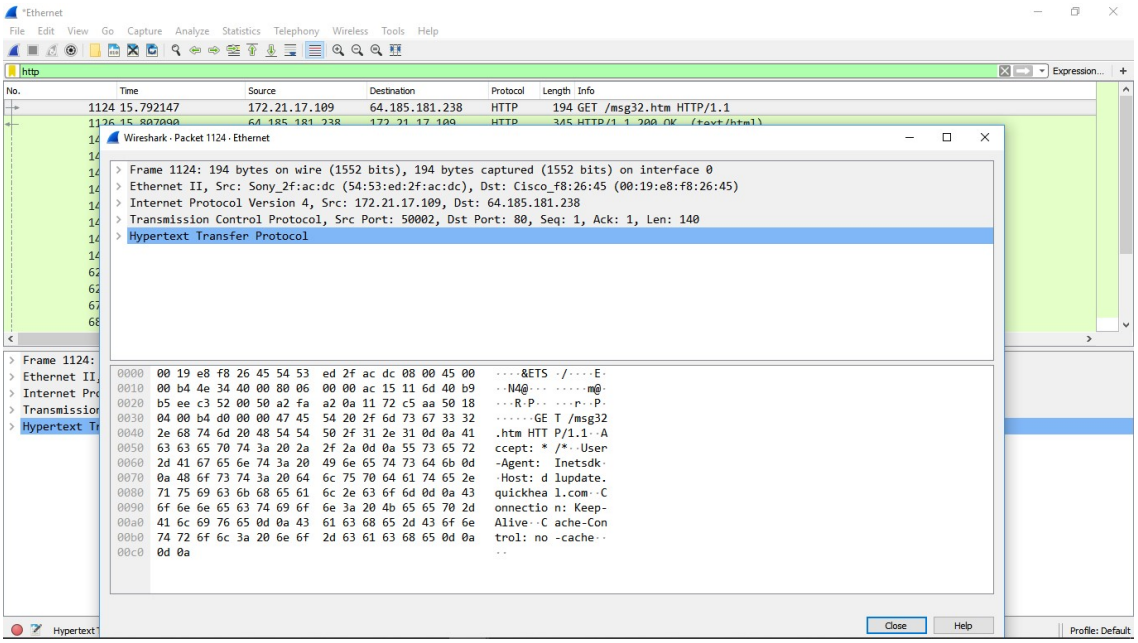
11. Type in “http” (without the quotes, and in lower case – all protocol names are in lower case in Wireshark) into the display filter specification window at the top of the main Wireshark window. Then select Apply (to the right of where you entered “http”). This will cause only HTTP message to be displayed in the packet-listing window.

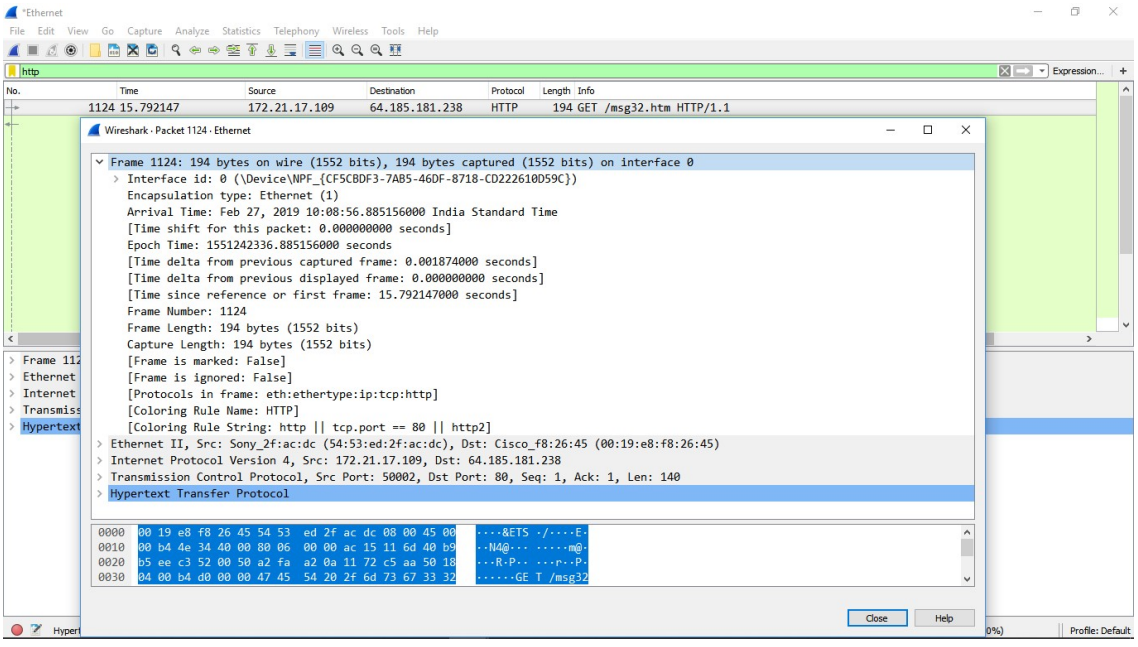
Ans:

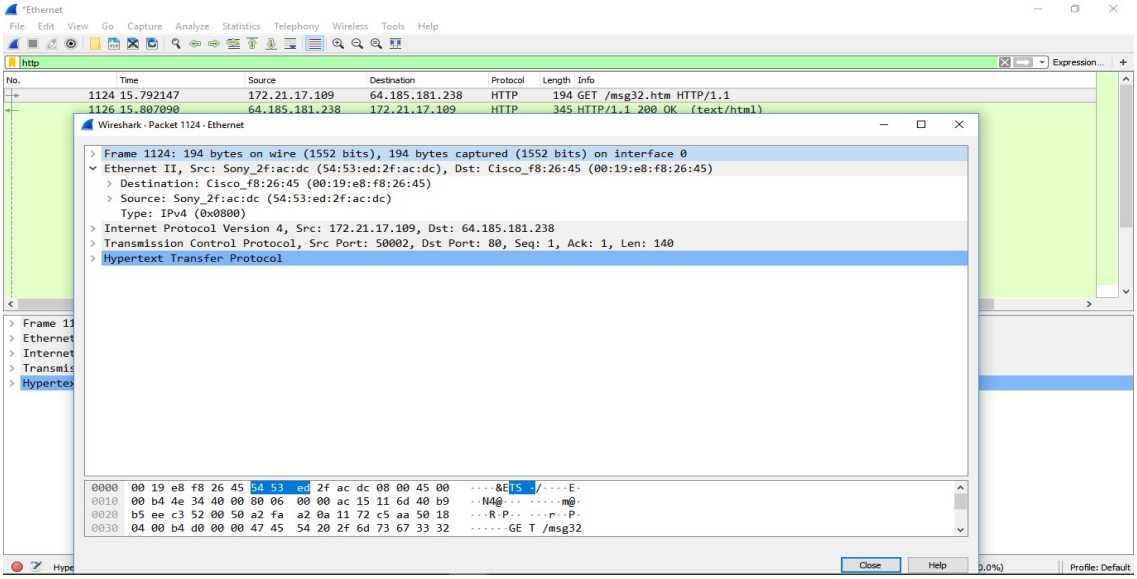


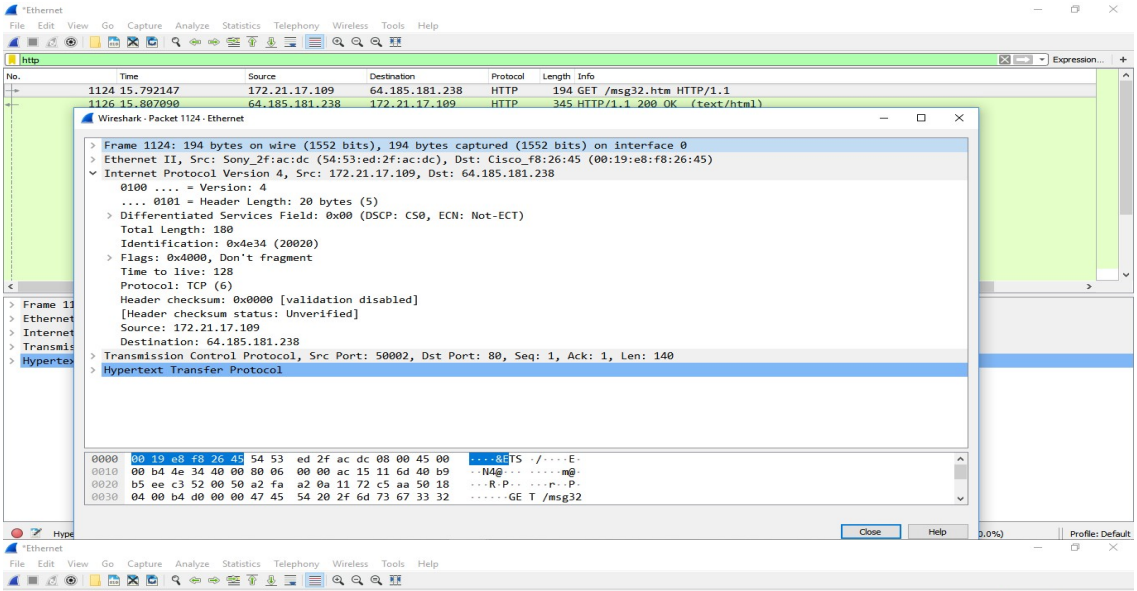
12. Select the first http message shown in the packet-listing window. This should be the HTTP GET message that was sent from your computer to the **www.gtu.ac.in HTTP server or www.google.com HTTP server or any other server**. When you select the HTTP GET message, the Ethernet frame, IP datagram, TCP segment, and HTTP message header information will be displayed in the packet-header window. Maximize the amount information displayed about the HTTP protocol.

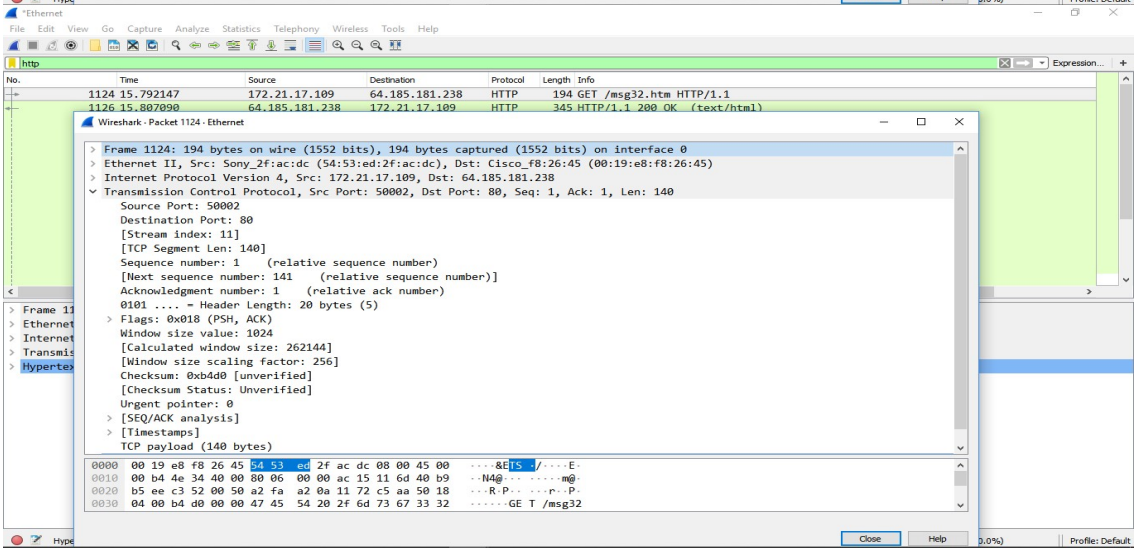
Ans:

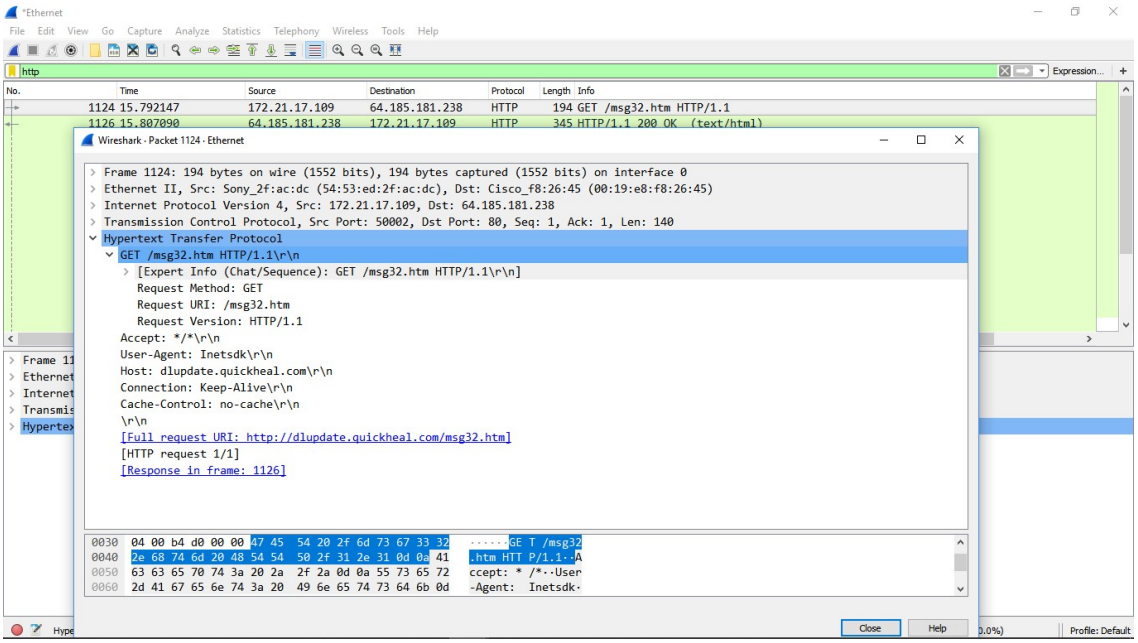





*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No. Time Source Destination Protocol Length Info
1124 15.792147 172.21.17.109 64.185.181.238 HTTP 194 GET /msg32.htm HTTP/1.1
1126 15.807090 64.185.181.238 172.21.17.109 HTTP 345 HTTP/1.1 200 OK (text/html)
Wireshark · Packet 1124 · Ethernet
Frame 1124: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits) on interface 0
Ethernet II, Src: Sony_2f:ac:dc (54:53:ed:2f:ac:dc), Dst: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Destination: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Source: Sony_2f:ac:dc (54:53:ed:2f:ac:dc)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 172.21.17.109, Dst: 64.185.181.238
Transmission Control Protocol, Src Port: 50002, Dst Port: 80, Seq: 1, Ack: 1, Len: 140
Hypertext Transfer Protocol
0000 00 19 e8 f8 26 45 54 53 ed 2f ac dc 08 00 45 00 ····&ETS ·/····E·
0010 00 b4 4e 34 40 00 80 06 00 00 ac 15 11 6d 40 b9 ··N4@··· ·····m@·
0020 b5 ee c3 52 00 50 a2 fa a2 0a 11 72 c5 aa 50 18 ···R·P·· ···r··P·
0030 04 00 b4 d0 00 00 47 45 54 20 2f 6d 73 67 33 32 ·····GE T /msg32
Close
Help
Profile: Default

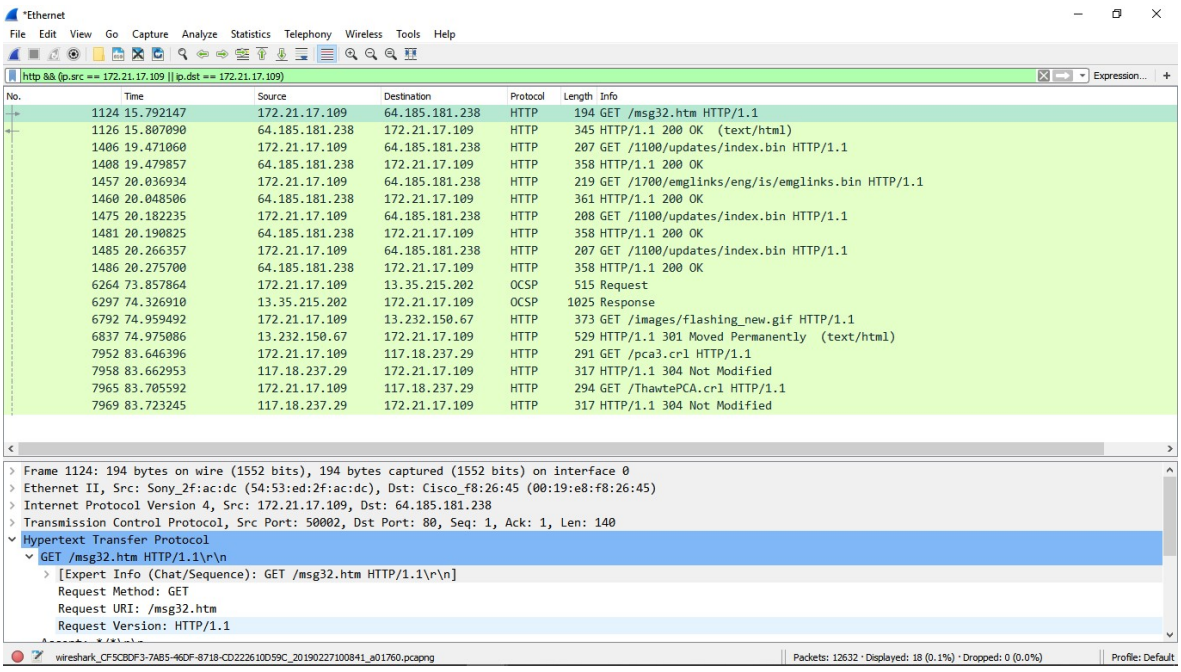

*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No. Time Source Destination Protocol Length Info
1124 15.792147 172.21.17.109 64.185.181.238 HTTP 194 GET /msg32.htm HTTP/1.1
1126 15.807090 64.185.181.238 172.21.17.109 HTTP 345 HTTP/1.1 200 OK (text/html)
Wireshark · Packet 1124 · Ethernet
Frame 1124: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits) on interface 0
Ethernet II, Src: Sony_2f:ac:dc (54:53:ed:2f:ac:dc), Dst: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Internet Protocol Version 4, Src: 172.21.17.109, Dst: 64.185.181.238
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 180
Identification: 0x4e34 (20020)
Flags: 0x4000, Don't fragment
Time to live: 128
Protocol: TCP (6)
Header checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
Source: 172.21.17.109
Destination: 64.185.181.238
Transmission Control Protocol, Src Port: 50002, Dst Port: 80, Seq: 1, Ack: 1, Len: 140
Hypertext Transfer Protocol
0000 00 19 e8 f8 26 45 54 53 ed 2f ac dc 08 00 45 00 ····&ETS ·/····E·
0010 00 b4 4e 34 40 00 80 06 00 00 ac 15 11 6d 40 b9 ··N4@··· ·····m@·
0020 b5 ee c3 52 00 50 a2 fa a2 0a 11 72 c5 aa 50 18 ···R·P·· ···r··P·
0030 04 00 b4 d0 00 00 47 45 54 20 2f 6d 73 67 33 32 ·····GE T /msg32
Close
Help
Profile: Default


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No. Time Source Destination Protocol Length Info
1124 15.792147 172.21.17.109 64.185.181.238 HTTP 194 GET /msg32.htm HTTP/1.1
1126 15.807090 64.185.181.238 172.21.17.109 HTTP 345 HTTP/1.1 200 OK (text/html)
Wireshark · Packet 1124 · Ethernet
Frame 1124: 194 bytes on wire (1552 bits), 194 bytes captured (1552 bits) on interface 0
Ethernet II, Src: Sony_2f:ac:dc (54:53:ed:2f:ac:dc), Dst: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Internet Protocol Version 4, Src: 172.21.17.109, Dst: 64.185.181.238
Transmission Control Protocol, Src Port: 50002, Dst Port: 80, Seq: 1, Ack: 1, Len: 140
Source Port: 50002
Destination Port: 80
[Stream index: 11]
[TCP Segment Len: 140]
Sequence number: 1 (relative sequence number)
[Next sequence number: 141 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window size value: 1024
[Calculated window size: 262144]
[Window size scaling factor: 256]
Checksum: 0xb4d0 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
[SEQ/ACK analysis]
[Timestamps]
TCP payload (140 bytes)
0000 00 19 e8 f8 26 45 54 53 ed 2f ac dc 08 00 45 00 ····&ETS ·/····E·
0010 00 b4 4e 34 40 00 80 06 00 00 ac 15 11 6d 40 b9 ··N4@··· ·····m@·
0020 b5 ee c3 52 00 50 a2 fa a2 0a 11 72 c5 aa 50 18 ···R·P·· ···r··P·
0030 04 00 b4 d0 00 00 47 45 54 20 2f 6d 73 67 33 32 ·····GE T /msg32
Close
Help
Profile: Default



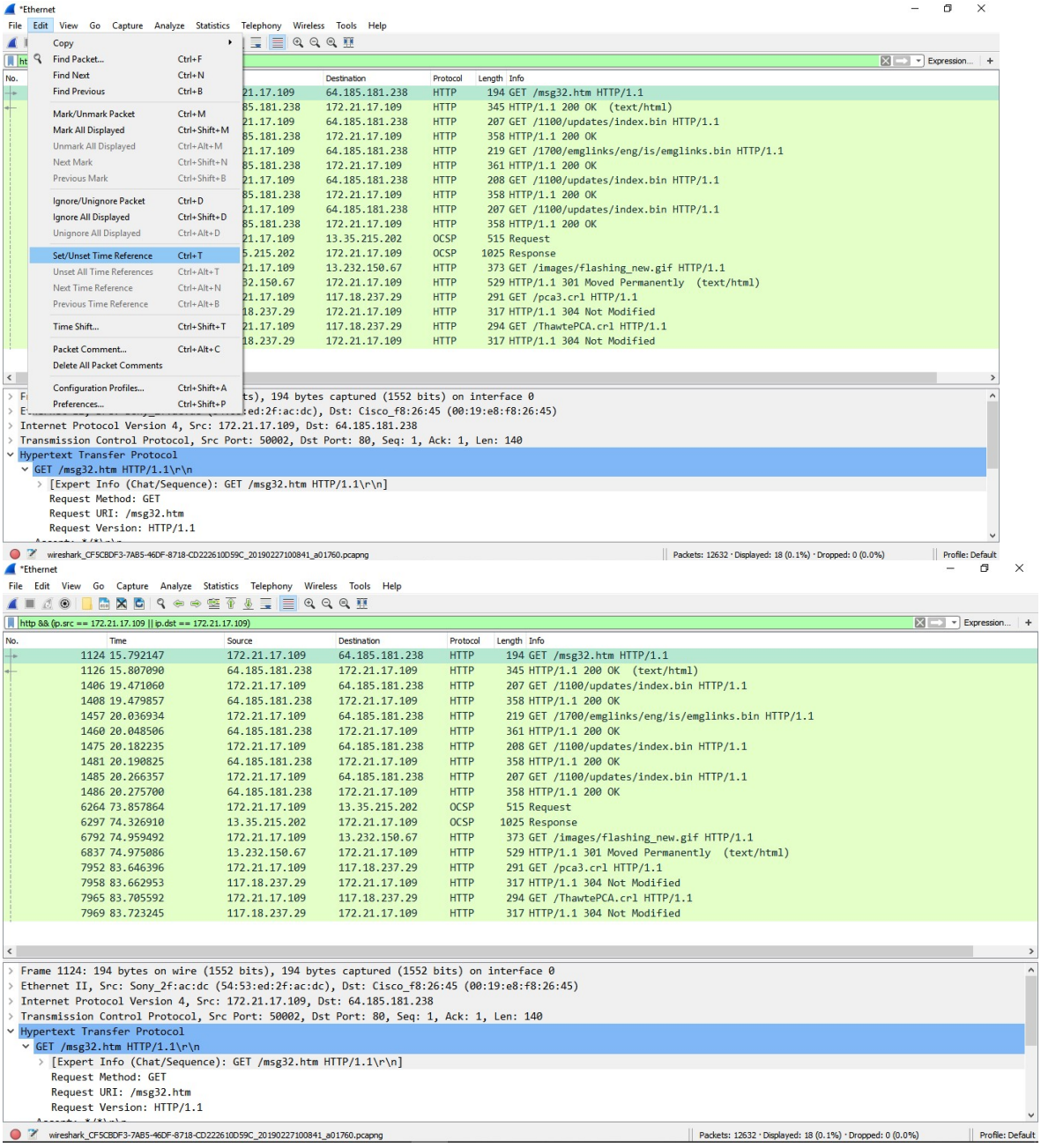
13. Type in http into the display filter, select “Apply” in the filter toolbar. This will cause only HTTP message to be displayed in the packet-listing window. Add the filter ip.src == <your IP address> || ip.dst == <your IP address> to filter out traffic that is going to or from your computer. This will keep other people’s traffic private and get rid of lots of HTTP exchanges from other computers that you don’t care about. Filters are combined with C operators. For example, if your IP address is 169.1.19.87, then your filter should be http && (ip.src == 169.1.19.87 || ip.dst == 169.1.19.87). You can also use a more English-like term to describe the same operators. For instance, ip.src eq 169.1.19.87 or ip.dst eq 169.1.19.87.

Ans:



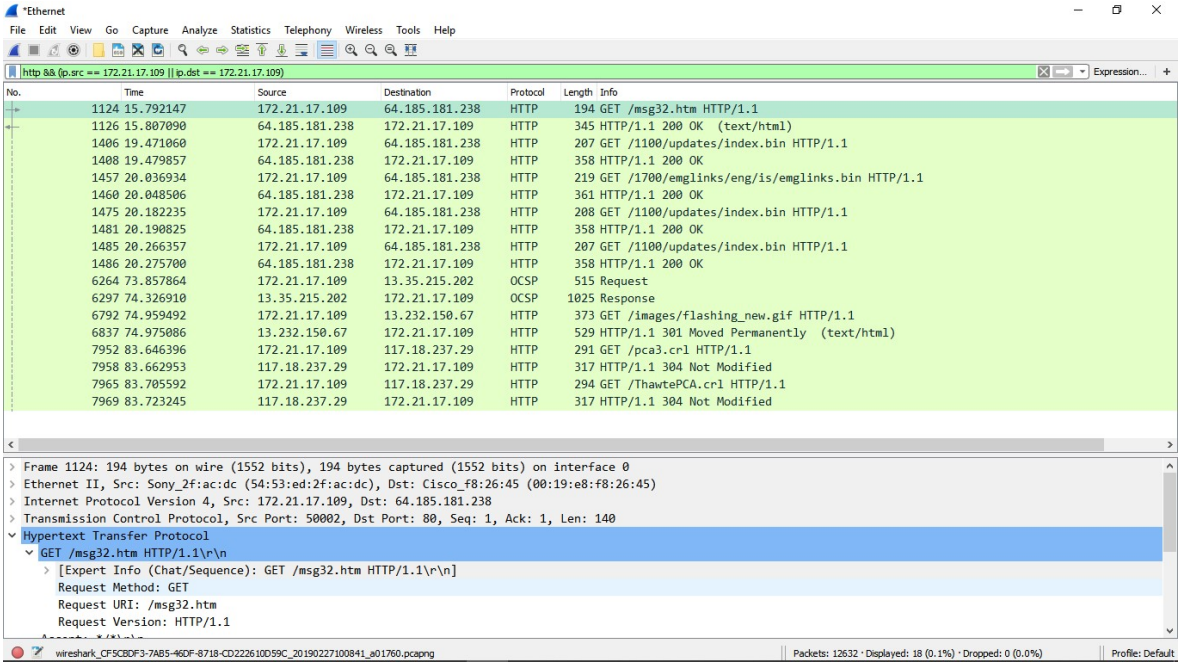
14. The capture time of each packet is quite important, so is displayed in the packet listing area as the second column. By default, this time is "number of seconds since the beginning of capture." However, you have control over what is displayed. Explore the View → Time Display Format menu to see display formats as well as precision choices. Also of interest is the ability to change the time reference so that all times are displayed relative to the capture time of a chosen packet. First, chose a packet from the display list by clicking on it. Then, go to the Edit → Set/Unset Time Reference, which will toggle your choice to use the chosen packet as the reference.

Ans:



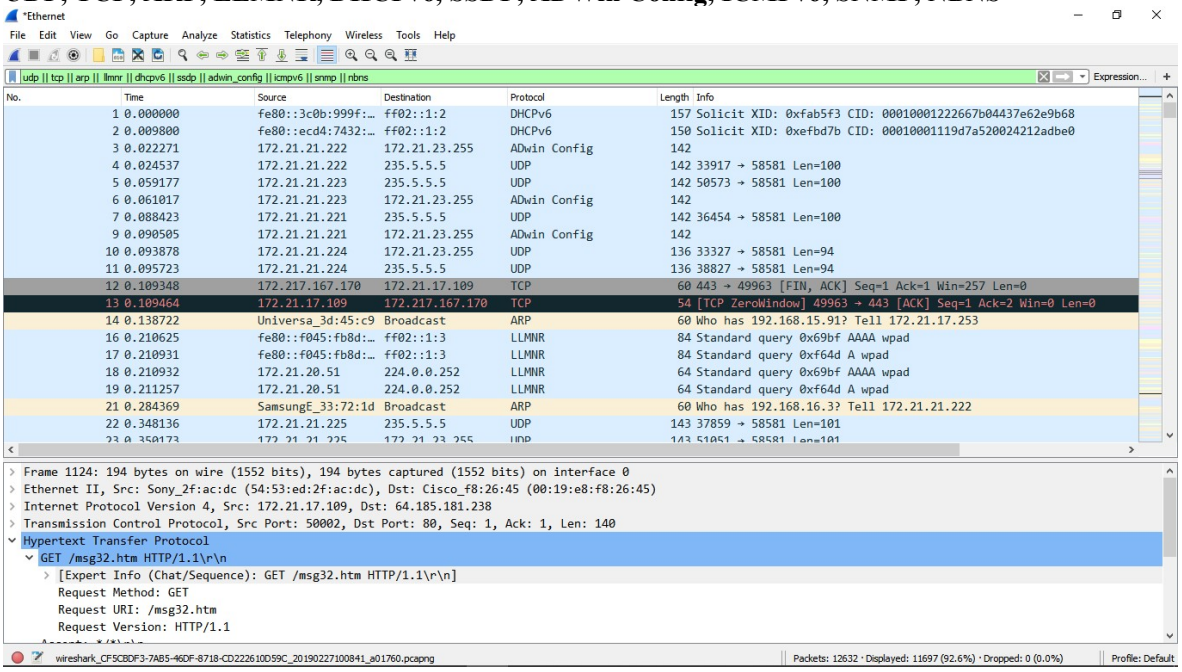
15. Color coding: Wireshark uses colors to identify the types of particular packets in the user interface. You will see packet highlighted in green, blue and black. By default Green is TCP traffic, dark blue is DNS traffic, light blue is UDP traffic and black identifies TCP packets with problem.

Ans:

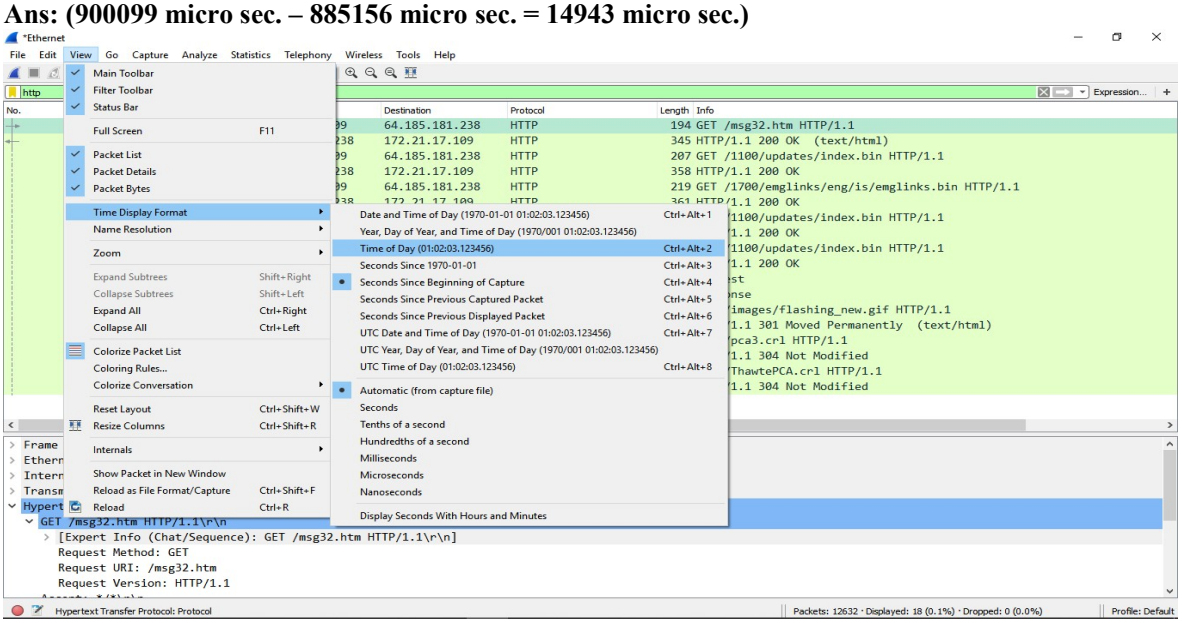


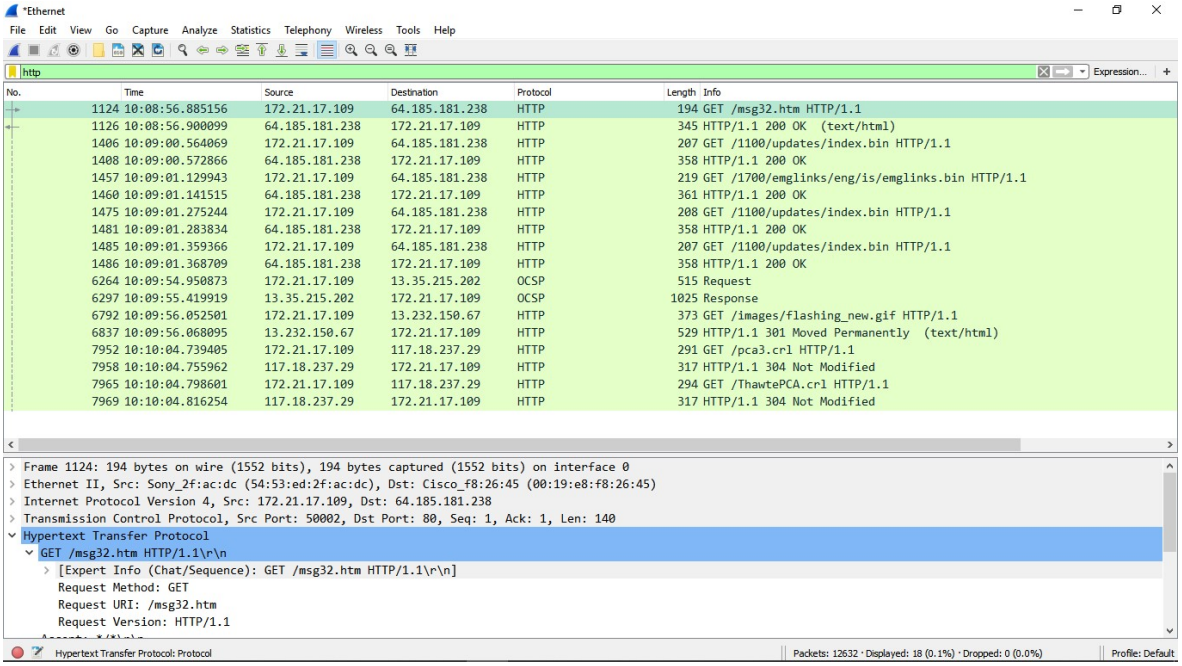
Experiment:

1. List up to 10 different protocols that appear in the protocol column in the unfiltered packet-listing window.
   Ans:
   UDP, TCP, ARP, LLMNR, DHCPv6, SSDP, ADWin Config, ICMPv6, SNMP, NBNS



2. How long did it take from when the HTTP GET message was sent until the HTTP OK reply was received? (By default, the value of the Time column in the packet listing window is the amount of time, in seconds, since Wireshark tracing began. To display the Time field in time-of-day format, select the Wireshark View pull down menu, then select Time Display Format, then select Time-of-day. For now you don't need to understand HTTP GET and OK, but reading the textbook may be helpful if you are curious on how they work.).
   Ans: (900099 micro sec. – 885156 micro sec. = 14943 micro sec.)

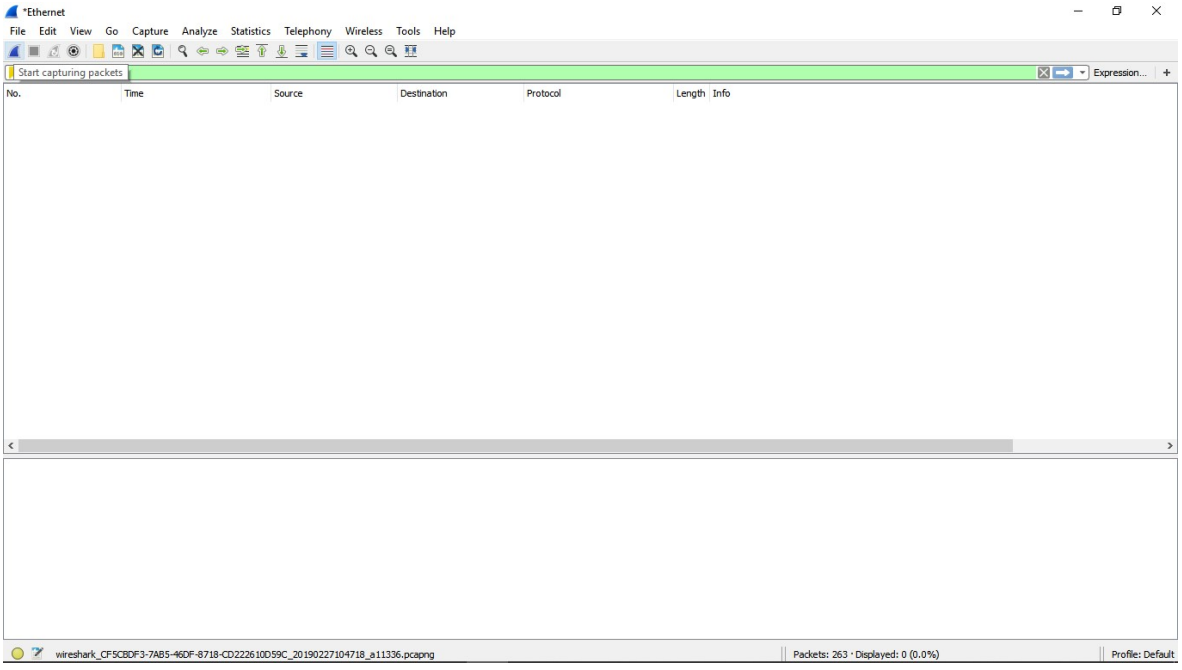


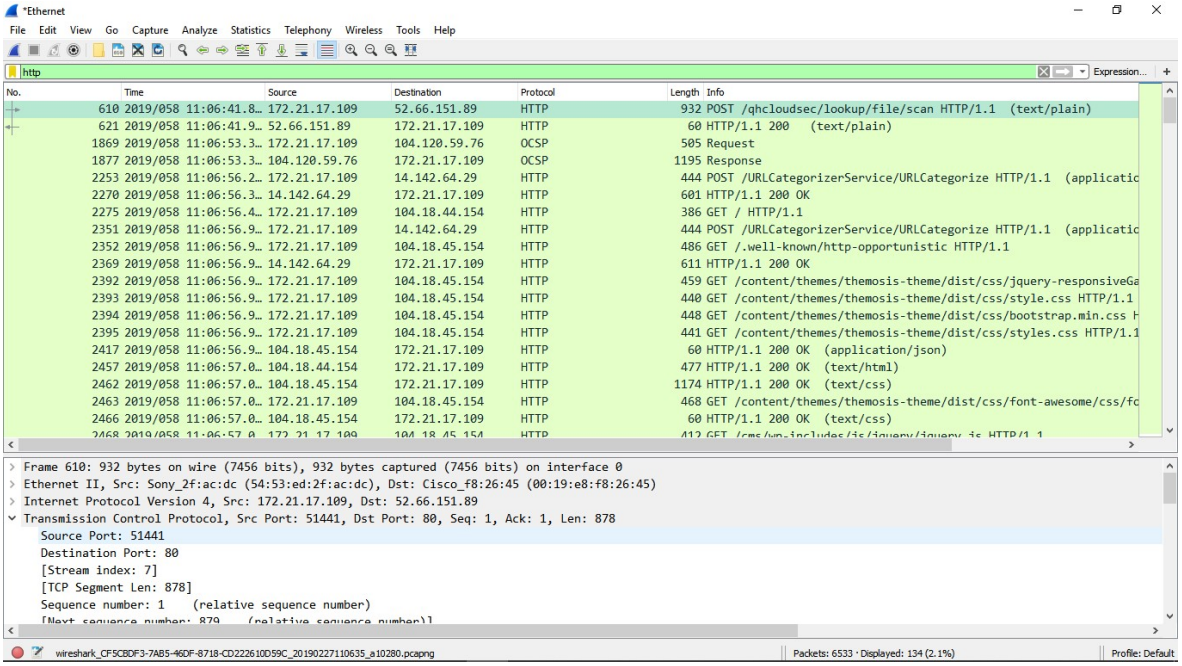


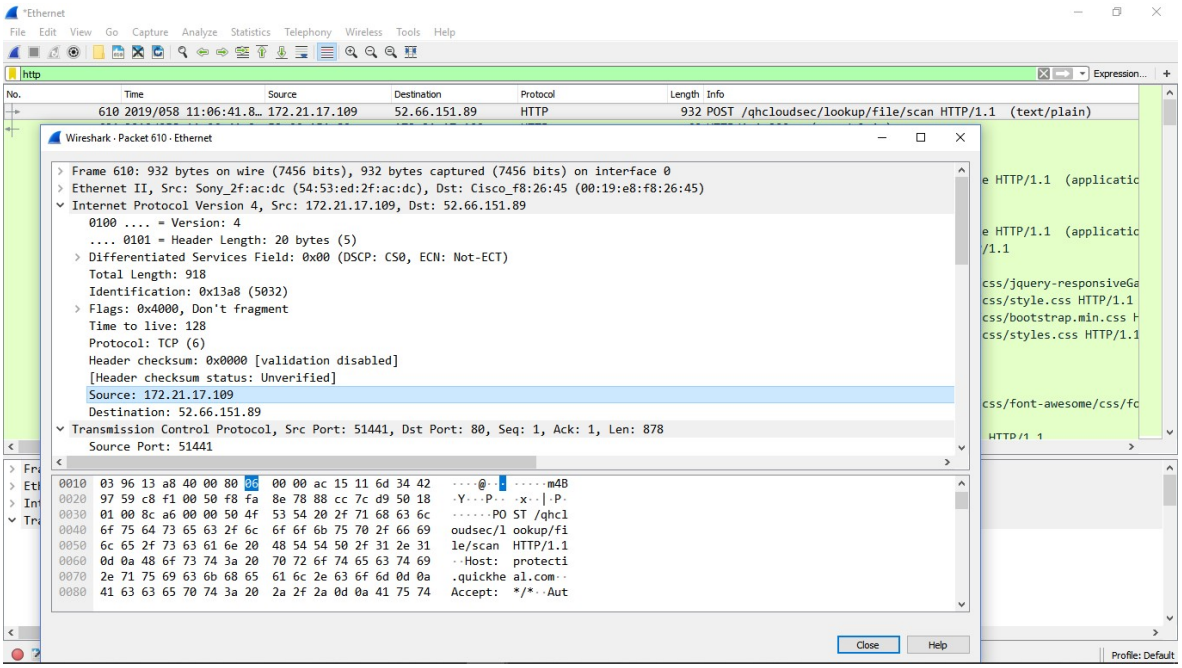
3. What is the Internet address of the www.google.com or www.sandesh.com ? What is the Internet address of your computer? Include a screenshot and describe where you got the data to answer this question.

Ans:

My IP address is 172.21.17.109 and the server's (www.sandesh.com) is 52.66.151.89

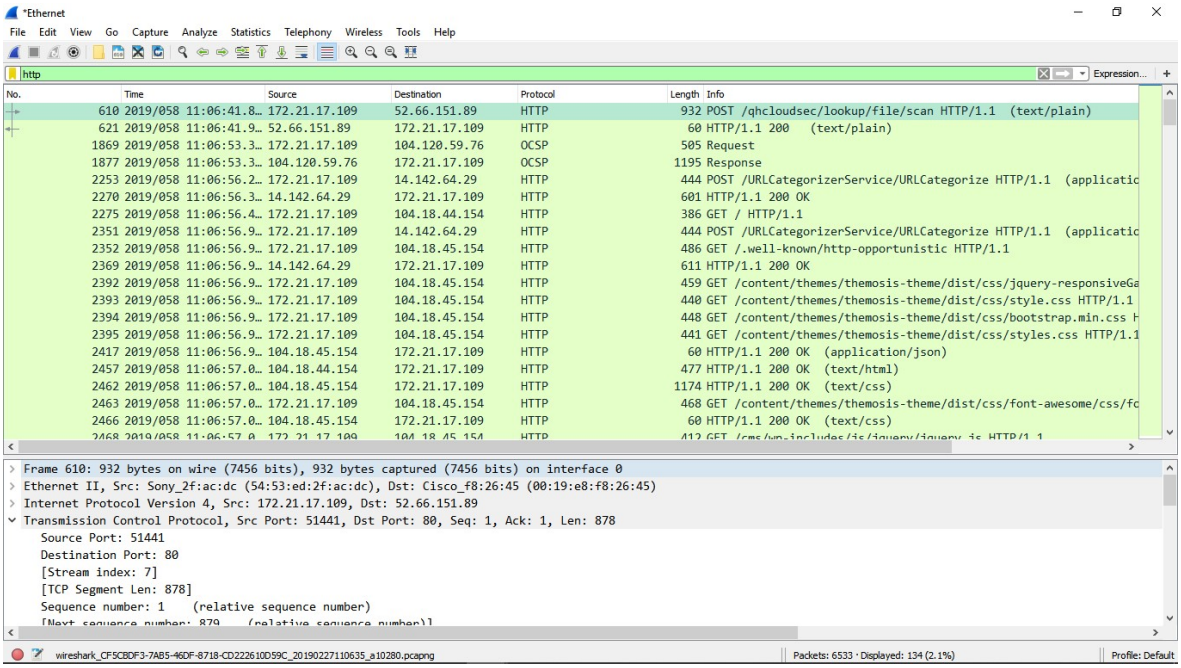



*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No. Time Source Destination Protocol Length Info
610 2019/058 11:06:41.8... 172.21.17.109 52.66.151.89 HTTP 932 POST /qhcloudsec/lookup/file/scan HTTP/1.1 (text/plain)
621 2019/058 11:06:41.9... 52.66.151.89 172.21.17.109 HTTP 60 HTTP/1.1 200 (text/plain)
1869 2019/058 11:06:53.3... 172.21.17.109 104.120.59.76 OCSP 505 Request
1877 2019/058 11:06:53.3... 104.120.59.76 172.21.17.109 OCSP 1195 Response
2253 2019/058 11:06:56.2... 172.21.17.109 14.142.64.29 HTTP 444 POST /URLCategorizerService/URLCategorize HTTP/1.1 (applicatic
2270 2019/058 11:06:56.3... 14.142.64.29 172.21.17.109 HTTP 601 HTTP/1.1 200 OK
2275 2019/058 11:06:56.4... 172.21.17.109 104.18.44.154 HTTP 386 GET / HTTP/1.1
2351 2019/058 11:06:56.9... 172.21.17.109 14.142.64.29 HTTP 444 POST /URLCategorizerService/URLCategorize HTTP/1.1 (applicatic
2352 2019/058 11:06:56.9... 172.21.17.109 104.18.45.154 HTTP 486 GET /.well-known/http-opportunistic HTTP/1.1
2369 2019/058 11:06:56.9... 14.142.64.29 172.21.17.109 HTTP 611 HTTP/1.1 200 OK
2392 2019/058 11:06:56.9... 172.21.17.109 104.18.45.154 HTTP 459 GET /content/themes/themosis-theme/dist/css/jquery-responsiveGa
2393 2019/058 11:06:56.9... 172.21.17.109 104.18.45.154 HTTP 440 GET /content/themes/themosis-theme/dist/css/style.css HTTP/1.1
2394 2019/058 11:06:56.9... 172.21.17.109 104.18.45.154 HTTP 448 GET /content/themes/themosis-theme/dist/css/bootstrap.min.css H
2395 2019/058 11:06:56.9... 172.21.17.109 104.18.45.154 HTTP 441 GET /content/themes/themosis-theme/dist/css/styles.css HTTP/1.1
2417 2019/058 11:06:56.9... 104.18.45.154 172.21.17.109 HTTP 60 HTTP/1.1 200 OK (application/json)
2457 2019/058 11:06:57.0... 104.18.44.154 172.21.17.109 HTTP 477 HTTP/1.1 200 OK (text/html)
2462 2019/058 11:06:57.0... 104.18.45.154 172.21.17.109 HTTP 1174 HTTP/1.1 200 OK (text/css)
2463 2019/058 11:06:57.0... 172.21.17.109 104.18.45.154 HTTP 468 GET /content/themes/themosis-theme/dist/css/font-awesome/css/fc
2466 2019/058 11:06:57.0... 104.18.45.154 172.21.17.109 HTTP 60 HTTP/1.1 200 OK (text/css)
Frame 610: 932 bytes on wire (7456 bits), 932 bytes captured (7456 bits) on interface 0
Ethernet II, Src: Sony_2f:ac:dc (54:53:ed:2f:ac:dc), Dst: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Internet Protocol Version 4, Src: 172.21.17.109, Dst: 52.66.151.89
Transmission Control Protocol, Src Port: 51441, Dst Port: 80, Seq: 1, Ack: 1, Len: 878
Source Port: 51441
Destination Port: 80
[Stream index: 7]
[TCP Segment Len: 878]
Sequence number: 1 (relative sequence number)
wireshark_CF5CBDF3-7AB5-46DF-8718-CD222610D59C_20190227110635_a10280.pcapng
Packets: 6533 · Displayed: 134 (2.1%)
Profile: Default


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No. Time Source Destination Protocol Length Info
610 2019/058 11:06:41.8... 172.21.17.109 52.66.151.89 HTTP 932 POST /qhcloudsec/lookup/file/scan HTTP/1.1 (text/plain)
Wireshark · Packet 610 · Ethernet
Frame 610: 932 bytes on wire (7456 bits), 932 bytes captured (7456 bits) on interface 0
Ethernet II, Src: Sony_2f:ac:dc (54:53:ed:2f:ac:dc), Dst: Cisco_f8:26:45 (00:19:e8:f8:26:45)
Internet Protocol Version 4, Src: 172.21.17.109, Dst: 52.66.151.89
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 918
Identification: 0x13a8 (5032)
Flags: 0x4000, Don't fragment
Time to live: 128
Protocol: TCP (6)
Header checksum: 0x0000 [validation disabled]
[Header checksum status: Unverified]
Source: 172.21.17.109
Destination: 52.66.151.89
Transmission Control Protocol, Src Port: 51441, Dst Port: 80, Seq: 1, Ack: 1, Len: 878
Source Port: 51441
0010 03 96 13 a8 40 00 80 06 00 00 ac 15 11 6d 34 42 ····@··· ·····m4B
0020 97 59 c8 f1 00 50 f8 fa 8e 78 88 cc 7c d9 50 18 ·Y···P·· ·x··|·P·
0030 01 00 8c a6 00 00 50 4f 53 54 20 2f 71 68 63 6c ·····PO ST /qhcl
0040 6f 75 64 73 65 63 2f 6c 6f 6f 6b 75 70 2f 66 69 oudsec/l ookup/fi
0050 6c 65 2f 73 63 61 6e 20 48 54 54 50 2f 31 2e 31 le/scan HTTP/1.1
0060 0d 0a 48 6f 73 74 3a 20 70 72 6f 74 65 63 74 69 ··Host: protecti
0070 2e 71 75 69 63 6b 68 65 61 6c 2e 63 6f 6d 0d 0a .quickhe al.com··
0080 41 63 63 65 70 74 3a 20 2a 2f 2a 0d 0a 41 75 74 Accept: */*··Aut
Close
Help
Profile: Default

4. Provide a screenshot showing http protocol only with Wireshark running on your computer.

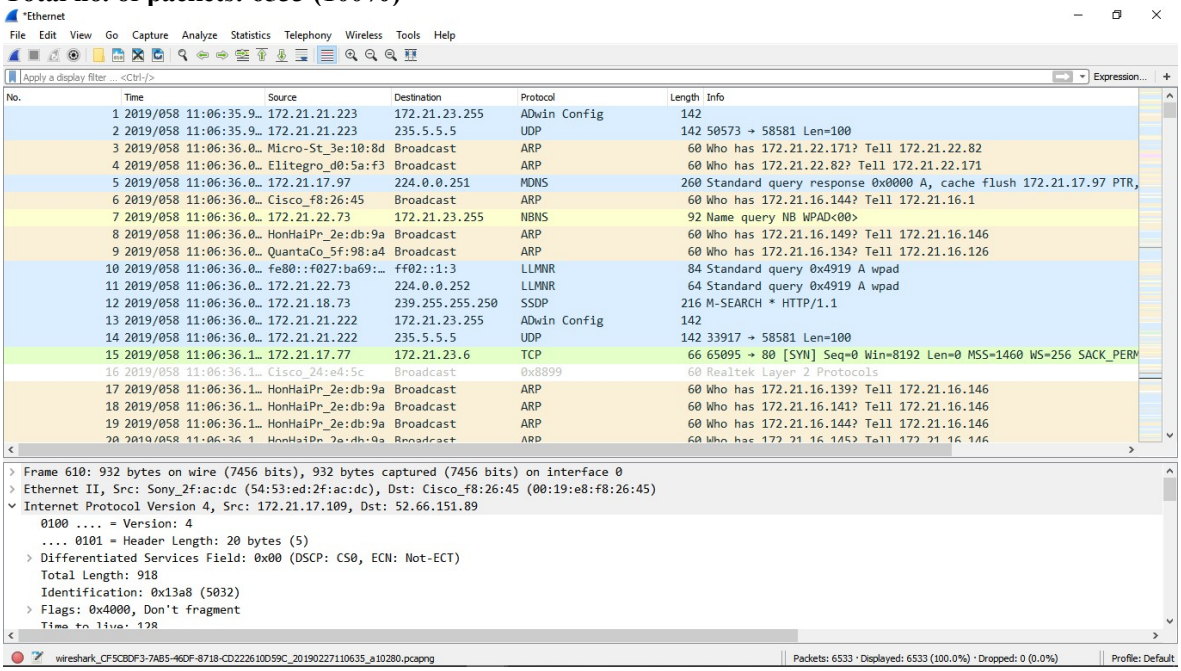
Ans:



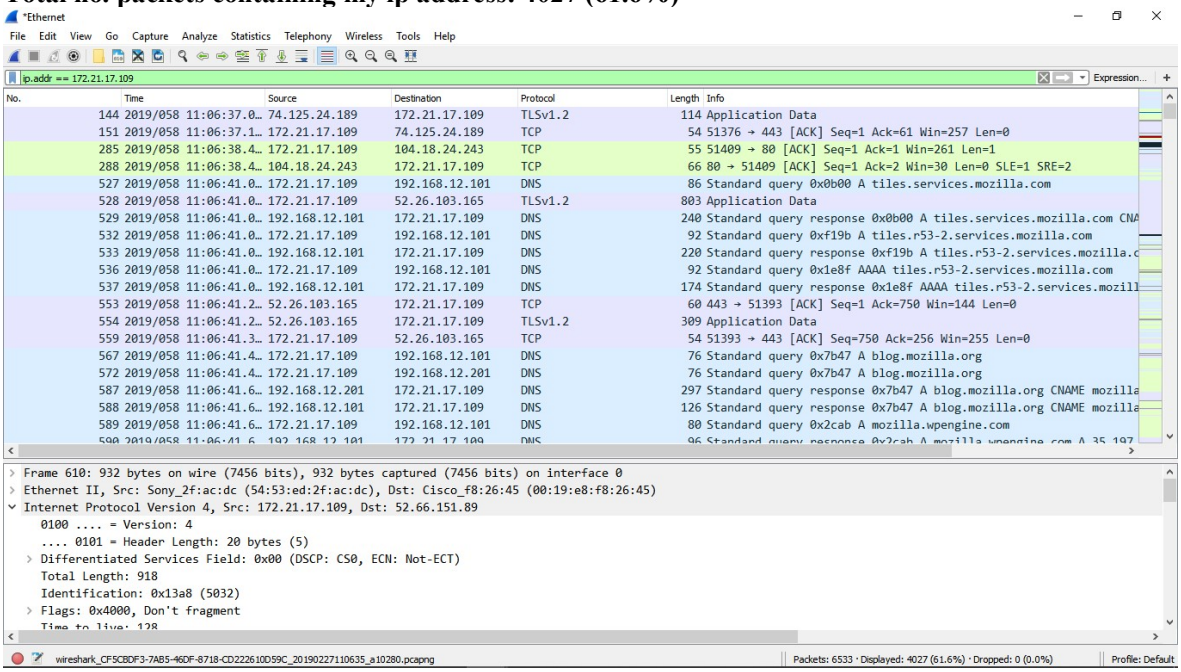
5. How many packets did you capture (total of all protocols, not just HTTP)? Now, use display filters to determine how many packets contain your ip address (hint: Use ip.addr instead of the clumsy ip.src or ip.dst format). What is this filter you used? Now, reverse the filter to determine how many packets don't contain your ip address.

Ans:

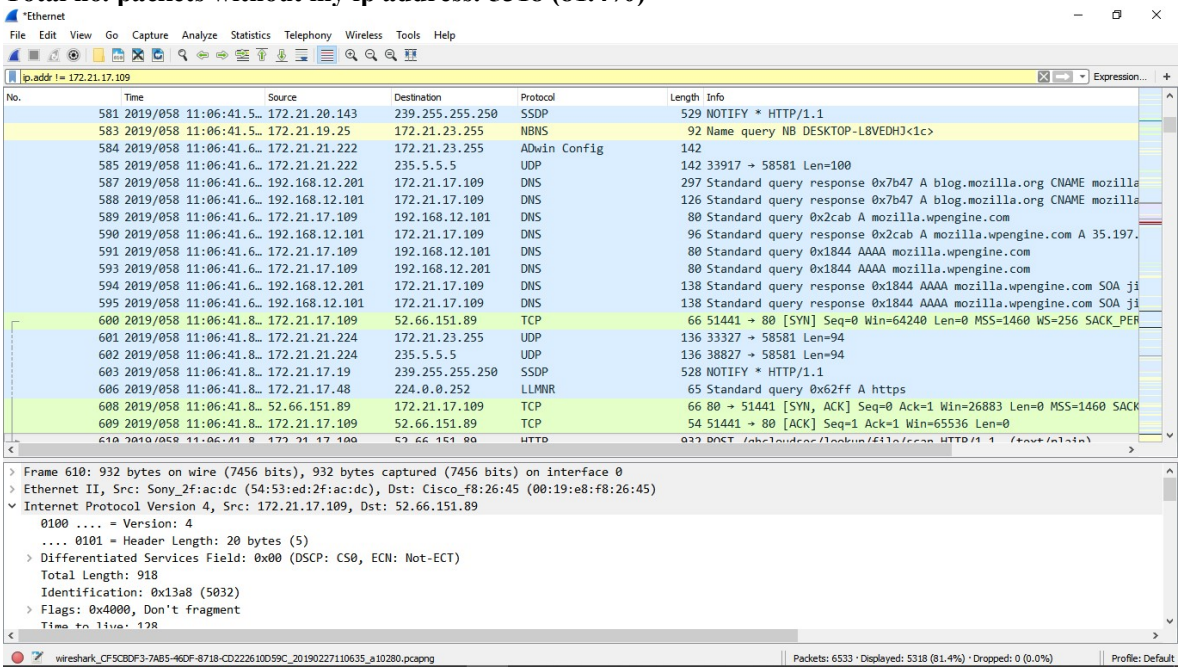
Total no. of packets: 6533 (100%)



Total no. packets containing my ip address: 4027 (61.6%)



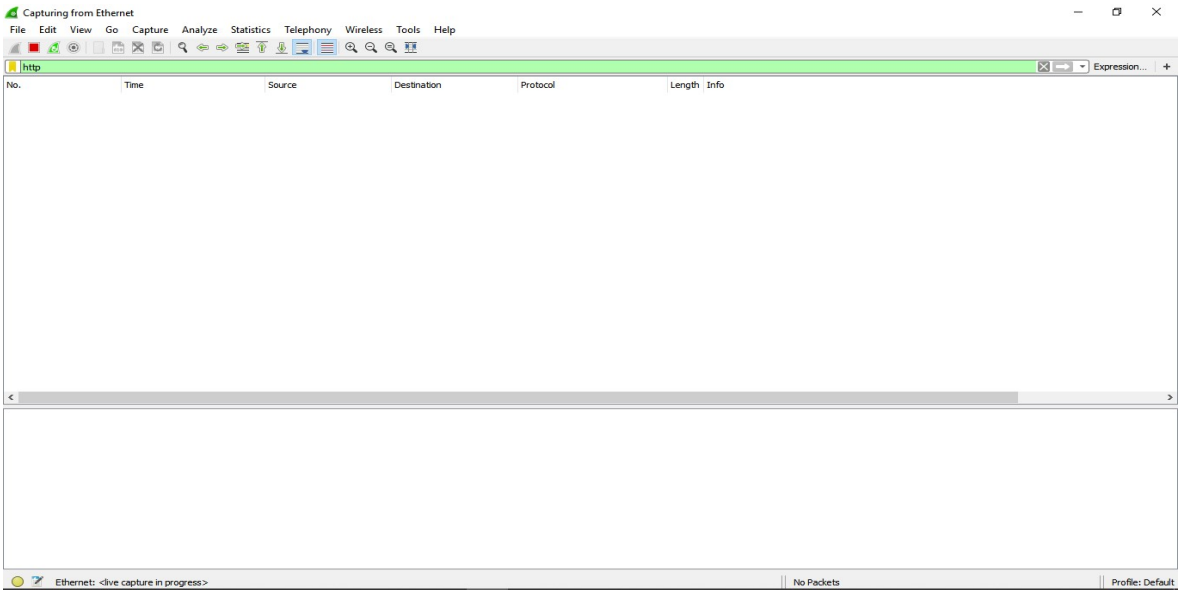
Total no. packets without my ip address: 5318 (81.4%)



HTTP Conditional GET/Response Interaction

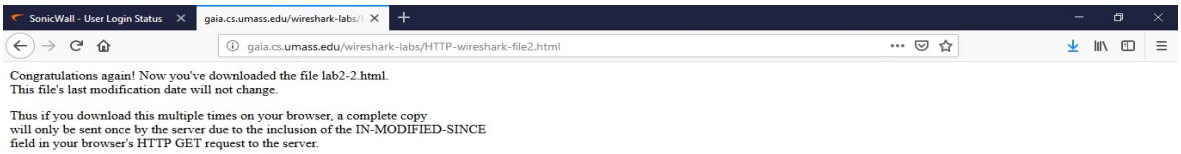
- We know that most web browsers perform object caching and thus perform the conditional GET when retrieving HTTP objects. Before performing the steps below, make sure that your browser's cache is empty.
- Start up your web browser, and make sure your browser's cache is cleared.
- Start up the Wireshark packet sniffer, and make sure that "http" is in the displayfilter, so that only captured HTTP messages will be displayed in the packet-list pane.

Ans:



- Enter the following URL into your browser:
  http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html
  Your browser should display a very simple HTML file.

Ans:

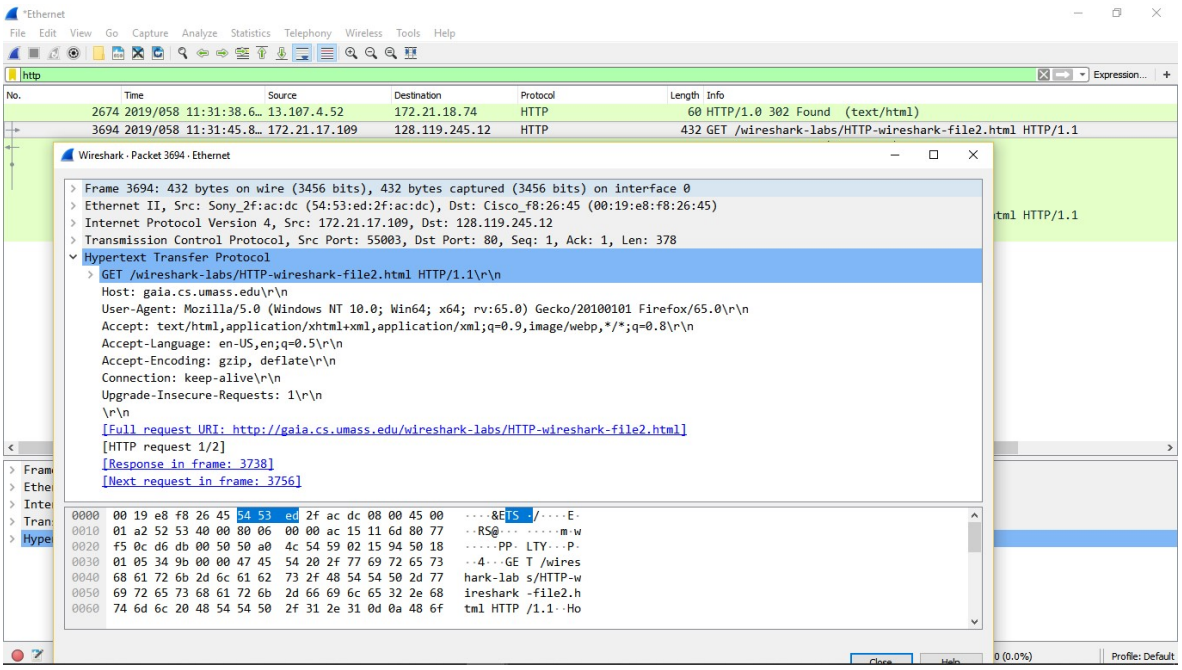


- Quickly enter the same URL into your browser again (or simply select the refresh button on your browser).
- Stop Wireshark packet capture.

Answer the following questions. Please explain how you can find the answers.

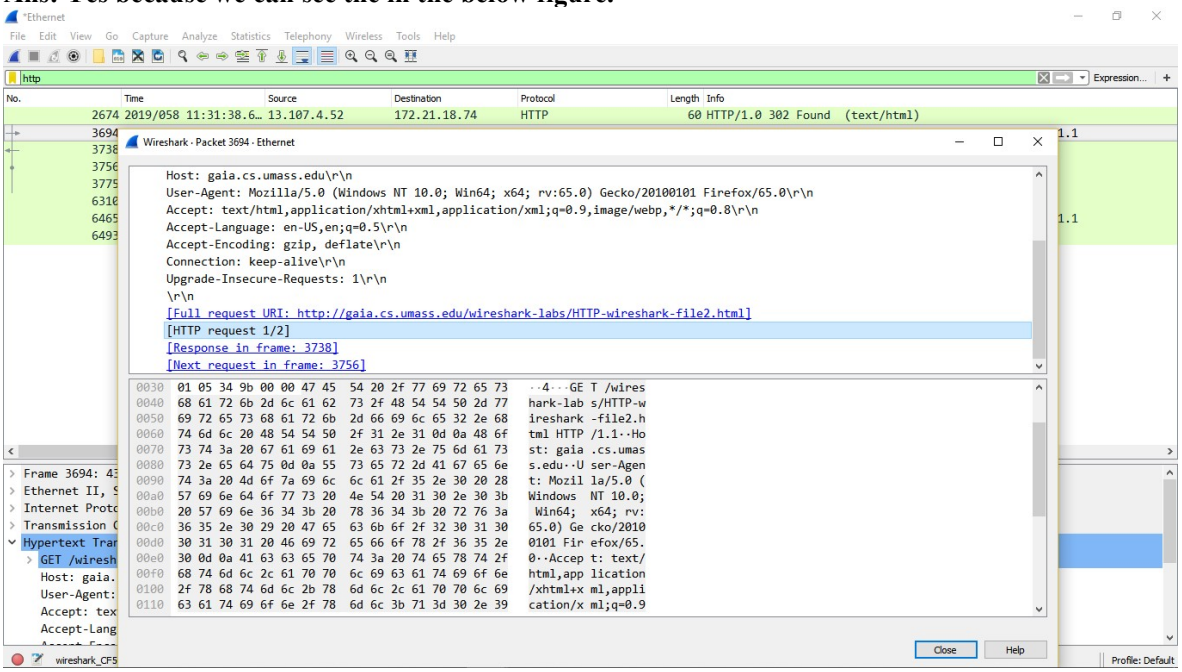
1. Inspect the contents of the first HTTP GET request from your browser to the server. Is there an "IF-MODIFIED-SINCE" header line in the HTTP GET message? Why or why not?

Ans: No

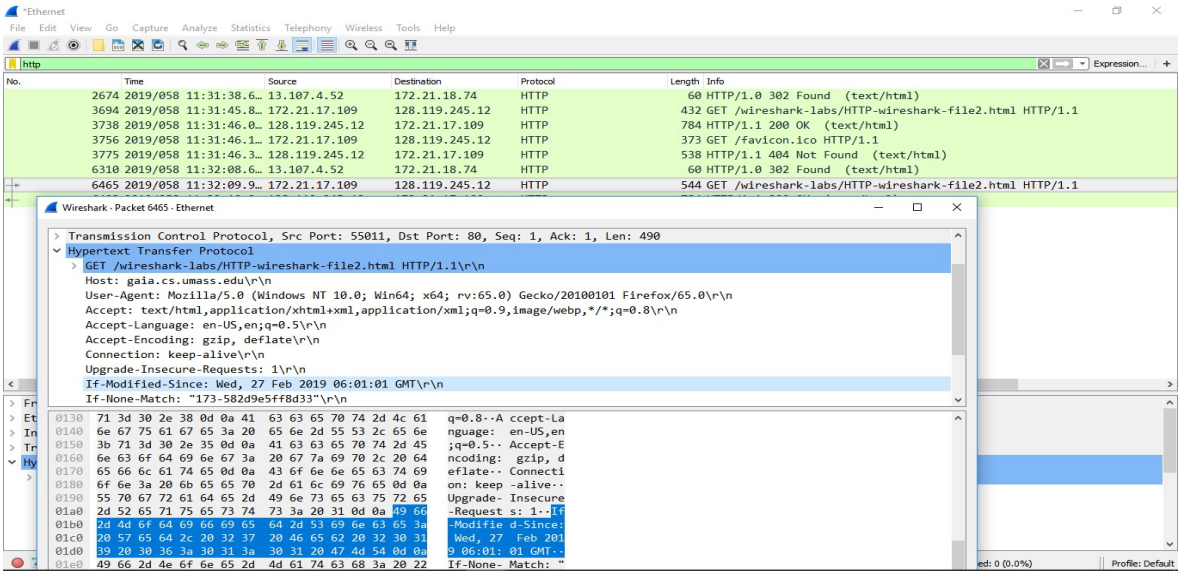


2. Inspect the contents of the server response. Has the server explicitly returned the contents of the file? How can you tell? (Hint: Search for http response if it is 1/1 or ½ then you can say server explicitly returned the contents of the file)

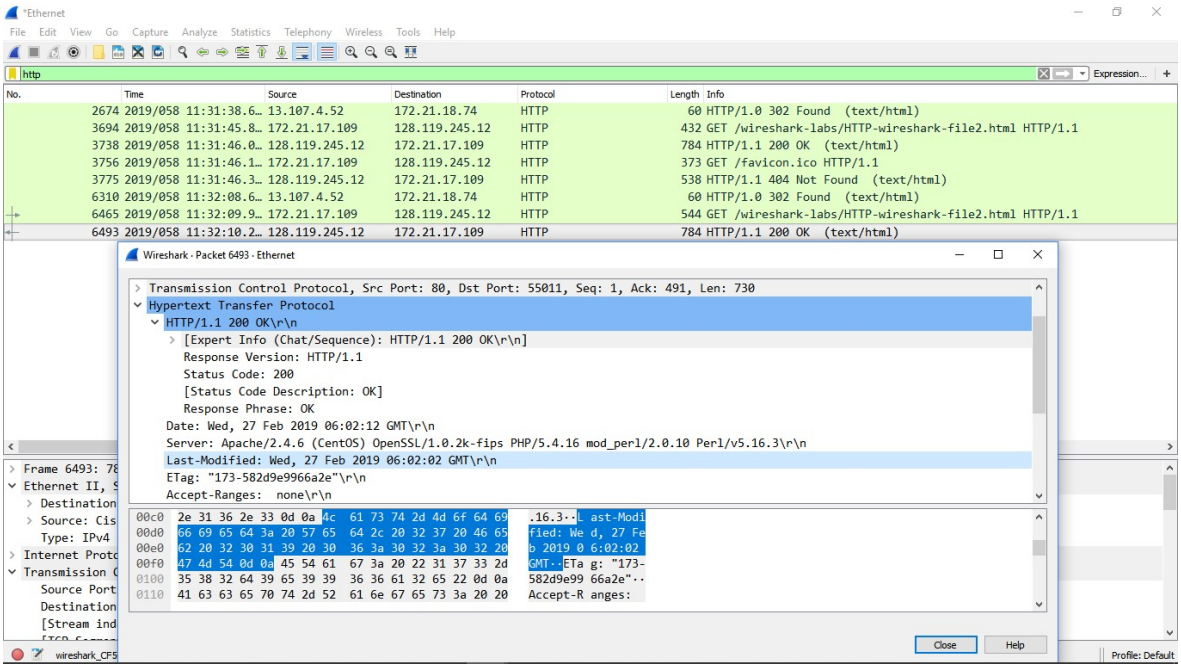
Ans: Yes because we can see the in the below figure.



3. Now inspect the contents of the second HTTP GET request from your browser to the server. Is there an "IF-MODIFIED-SINCE:" header line in the HTTP GET message? If so, what information follows the "IF-MODIFIED-SINCE:" header line?
Ans: Yes. The information followed is: Wed, 27 Feb 2019 06:01:01 GMT\r\n which is the date of the last modification of the file from the previous get request



4. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Has the server explicitly returned the contents of the file? Explain. (Hint: The status code and phrase returned from the server is ***HTTP/1.1 304 Not Modified***. The server didn't return the contents of the file since the browser loaded it from its cache (cookies))
Ans: The status code and phrase returned from the server is HTTP/1.1 304 Not Modified. The server didn't return the contents of the file since the browser loaded it from its cache.



5. Which packet in the trace contains the status code and phrase associated with the response to the HTTP GET request? What is the status code and phrase in the response?
Ans: Packet no. is 3738. 200 is the status code and OK is the phrase.