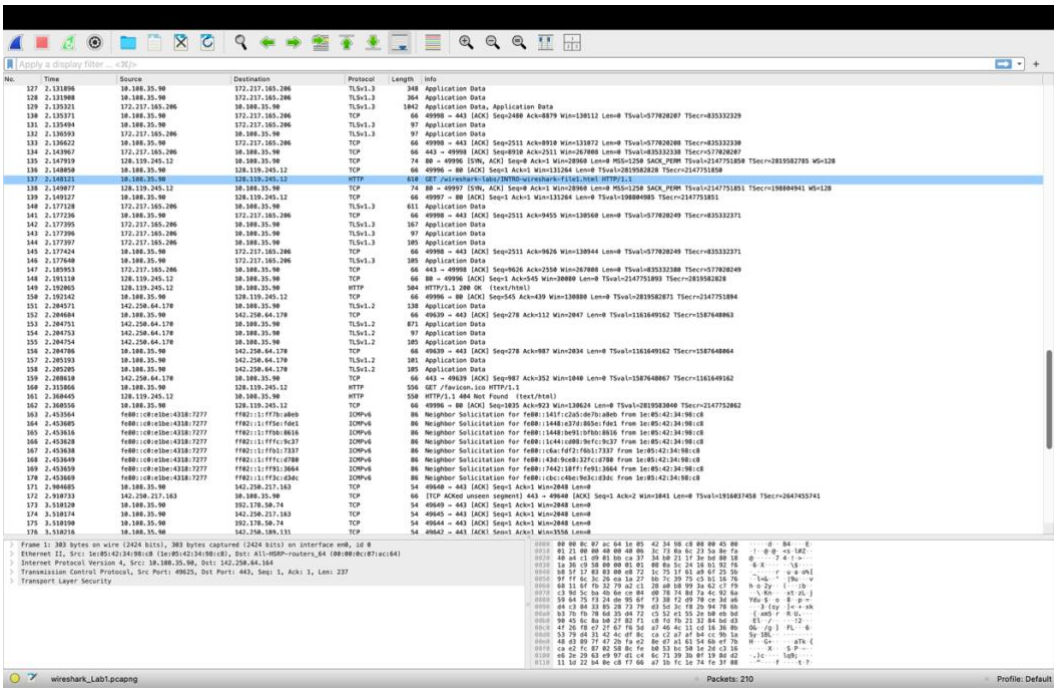
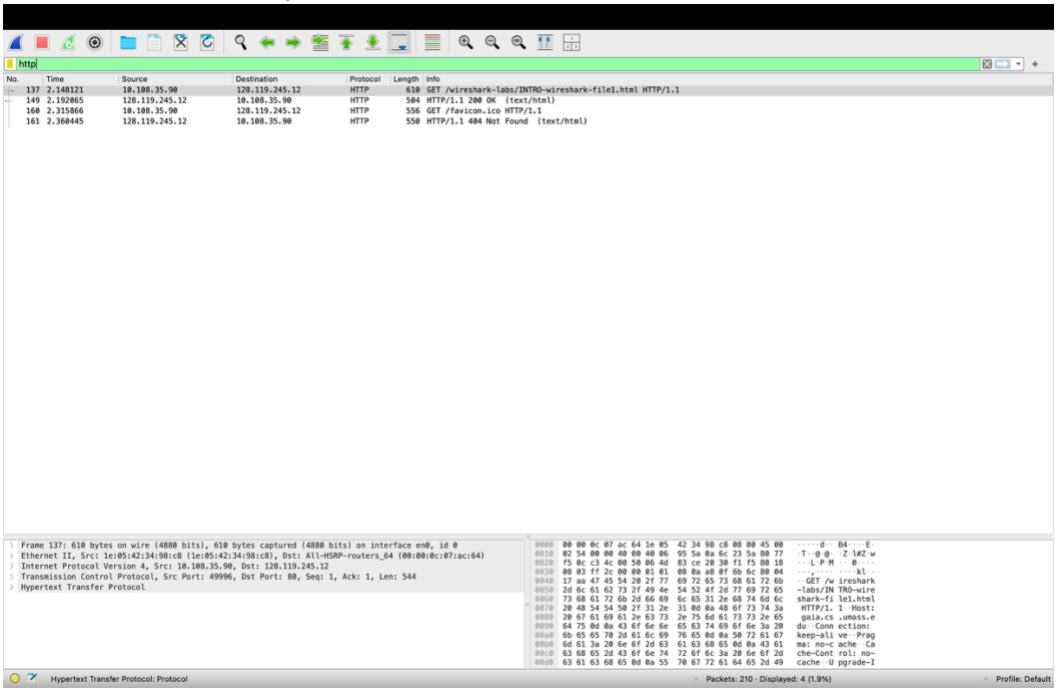
Catalina Cisneros

Lab: Wireshark Intro

Screenshot of the whole packet capture in Wireshark:

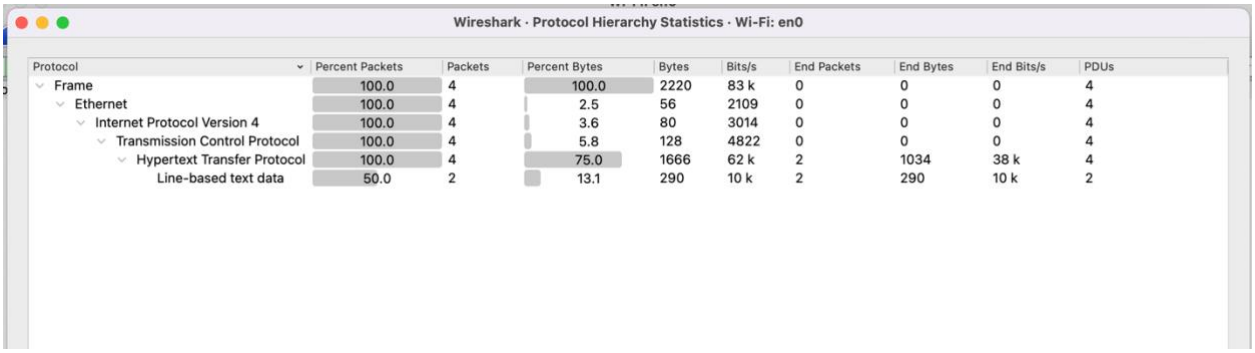


HTTP packets to analyze:



1. Which of the following protocols are shown as appearing (i.e., are listed in the Wireshark "protocol" column) in your trace file: TCP, QUIC, HTTP, DNS, UDP, TLSv1.2?

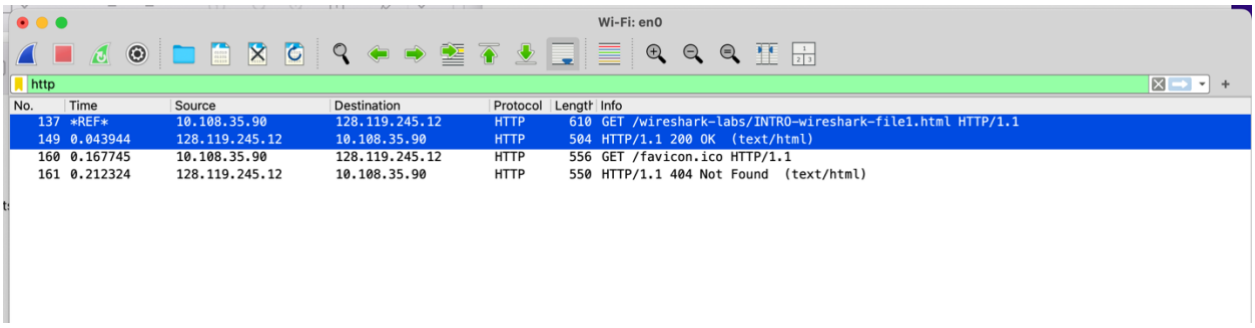
   → Protocols shown: I observed **TCP** and **HTTP**. Others, like QUIC, DNS, UDP, and TLSv1.2 did not appear.



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	4	100.0	2220	83 k	0	0	0	4
Ethernet	100.0	4	2.5	56	2109	0	0	0	4
Internet Protocol Version 4	100.0	4	3.6	80	3014	0	0	0	4
Transmission Control Protocol	100.0	4	5.8	128	4822	0	0	0	4
Hypertext Transfer Protocol	100.0	4	75.0	1666	62 k	2	1034	38 k	4
Line-based text data	50.0	2	13.1	290	10 k	2	290	10 k	2

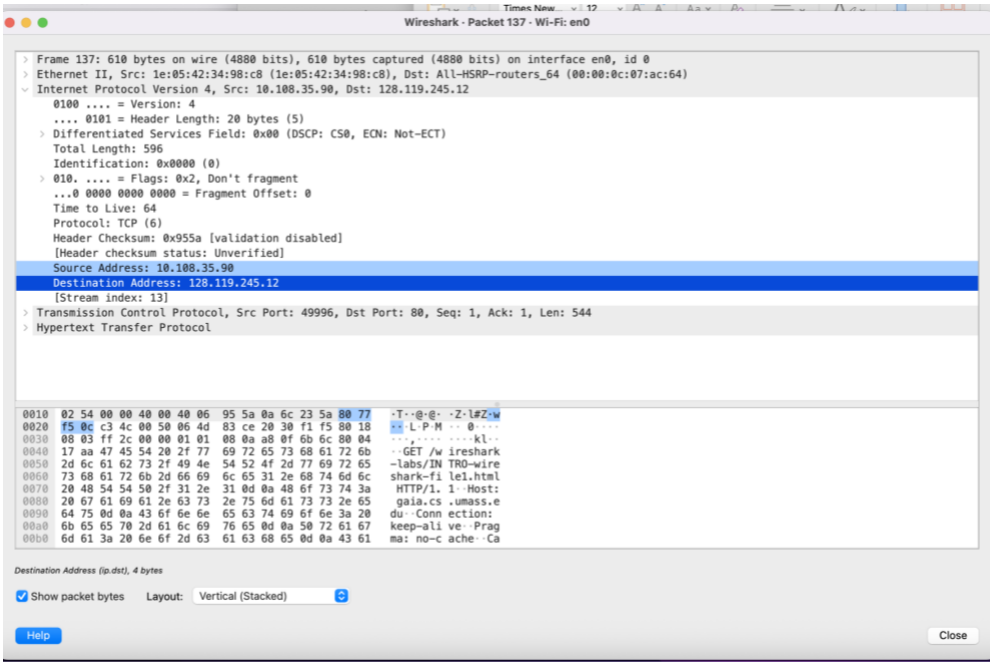
2. How long did it take from when the HTTP GET message was sent until the HTTP OK reply was received? (By default, the value of the Time column in the packet-listing window is the amount of time, in seconds, since Wireshark tracing began. (If you want to display the Time field in time-of-day format, select the Wireshark *View* pull down menu, then select Time *Display Format*, then select *Time-of-day*.

   → The time from the HTTP GET to the HTTP 200 OK was **0.043944 seconds**, using Time since reference.



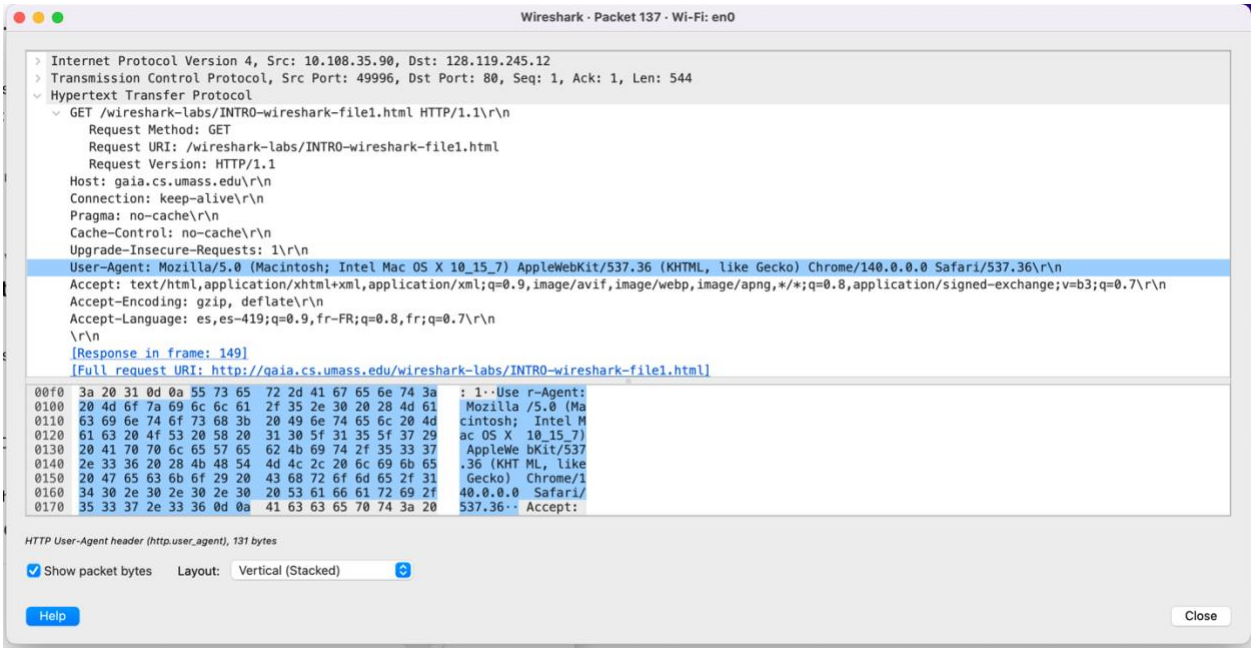
3. What is the Internet address of the gaia.cs.umass.edu (also known as www-net.cs.umass.edu)? What is the Internet address of your computer or (if you are using the trace file) the computer that sent the HTTP GET message?

   → Destination IP for gaia.cs.umass.edu is **128.119.245.12** and my host IP is **10.108.35.90**, taken from the IPv4 header of the GET.



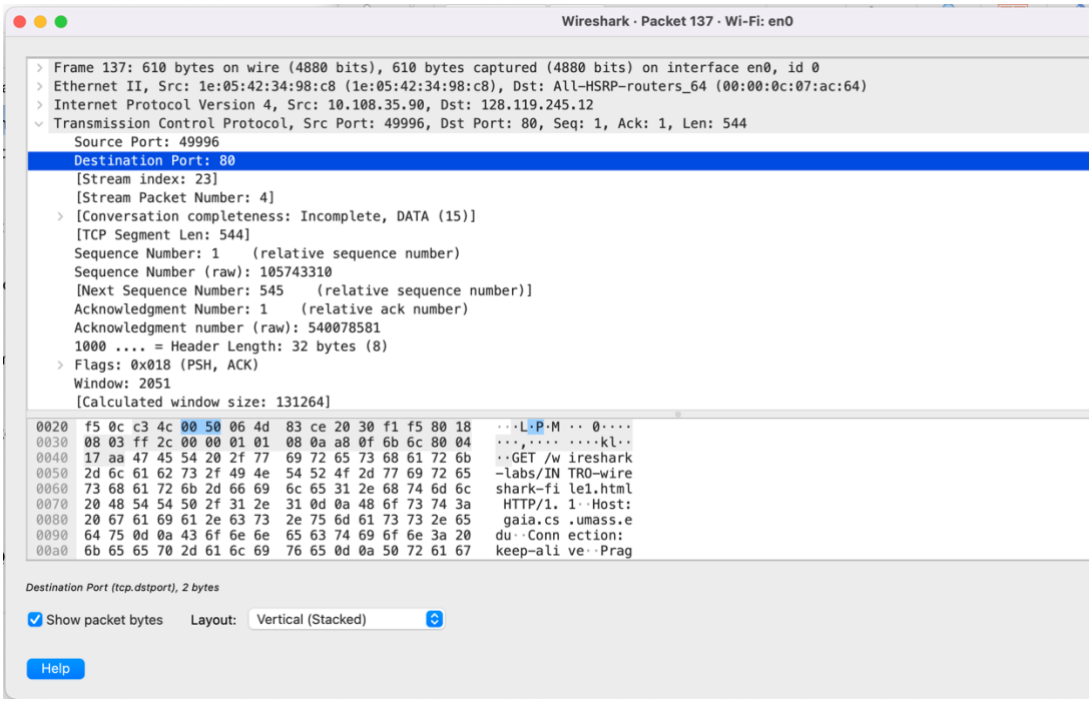
4. Expand the information on the HTTP message in the Wireshark “Details of selected packet” window (see Figure 3 above) so you can see the fields in the HTTP GET request message. What type of Web browser issued the HTTP request? The answer is shown at the right end of the information following the “User-Agent:” field in the expanded HTTP message display. [This field value in the HTTP message is how a web server learns what type of browser you are using.] Firefox, Safari, Microsoft Internet Edge, Other

   → User-Agent line shows *Chrome/140.0.0.0* so it is **Chrome**



5. Expand the information on the Transmission Control Protocol for this packet in the Wireshark "Details of selected packet" window (see Figure 3 in the lab writeup) so you can see the fields in the TCP segment carrying the HTTP message. What is the destination port number (the number following "Dest Port:" for the TCP segment containing the HTTP request) to which this HTTP request is being sent?

   → TCP destination port for the HTTP request is **80**.



   And finally ...

6. Print the two HTTP messages (GET and OK) referred to in question 2 above. To do so, select *Print* from the Wireshark *File* command menu, and select the "*Selected Packet Only*" and *"Print as displayed"* radial buttons, and then click OK.

   → PDFs attached

/var/folders/gh/4pct5jrs0dlcfcgzgw3fnzyc0000gn/T/wireshark_Wi-Fi1VX7C3.pcapng 210 total packets, 1 shown

```
No.     Time           Source                Destination           Protocol Length Info
    137 *REF*          10.108.35.90          128.119.245.12        HTTP     610    GET /wireshark-
labs/INTRO-wireshark-file1.html HTTP/1.1
Frame 137: 610 bytes on wire (4880 bits), 610 bytes captured (4880 bits) on interface en0, id 0
Ethernet II, Src: 1e:05:42:34:98:c8 (1e:05:42:34:98:c8), Dst: All-HSRP-routers_64 (00:00:0c:07:ac:
64)
Internet Protocol Version 4, Src: 10.108.35.90, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 49996, Dst Port: 80, Seq: 1, Ack: 1, Len: 544
    Source Port: 49996
    Destination Port: 80
    [Stream index: 23]
    [Stream Packet Number: 4]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 544]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 105743310
    [Next Sequence Number: 545    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 540078581
    1000 .... = Header Length: 32 bytes (8)
    Flags: 0x018 (PSH, ACK)
    Window: 2051
    [Calculated window size: 131264]
    [Window size scaling factor: 64]
    Checksum: 0xff2c [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
    Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    [Timestamps]
    [SEQ/ACK analysis]
    TCP payload (544 bytes)
Hypertext Transfer Protocol
```

/var/folders/gh/4pct5jrs0dlcfcgzgw3fnzyc0000gn/T/wireshark_Wi-Fi1VX7C3.pcapng 210 total packets, 1 shown

```
No.     Time           Source                Destination           Protocol Length Info
    149 0.043944       128.119.245.12        10.108.35.90          HTTP     504    HTTP/1.1 200 OK
(text/html)
Frame 149: 504 bytes on wire (4032 bits), 504 bytes captured (4032 bits) on interface en0, id 0
Ethernet II, Src: Cisco_54:9c:43 (8c:60:4f:54:9c:43), Dst: 1e:05:42:34:98:c8 (1e:05:42:34:98:c8)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.108.35.90
Transmission Control Protocol, Src Port: 80, Dst Port: 49996, Seq: 1, Ack: 545, Len: 438
    Source Port: 80
    Destination Port: 49996
    [Stream index: 23]
    [Stream Packet Number: 6]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 438]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 540078581
    [Next Sequence Number: 439    (relative sequence number)]
    Acknowledgment Number: 545    (relative ack number)
    Acknowledgment number (raw): 105743854
    1000 .... = Header Length: 32 bytes (8)
    Flags: 0x018 (PSH, ACK)
    Window: 235
    [Calculated window size: 30080]
    [Window size scaling factor: 128]
    Checksum: 0x5f3a [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
    Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    [Timestamps]
    [SEQ/ACK analysis]
    TCP payload (438 bytes)
Hypertext Transfer Protocol
Line-based text data: text/html (3 lines)
```