Q4.

1a.

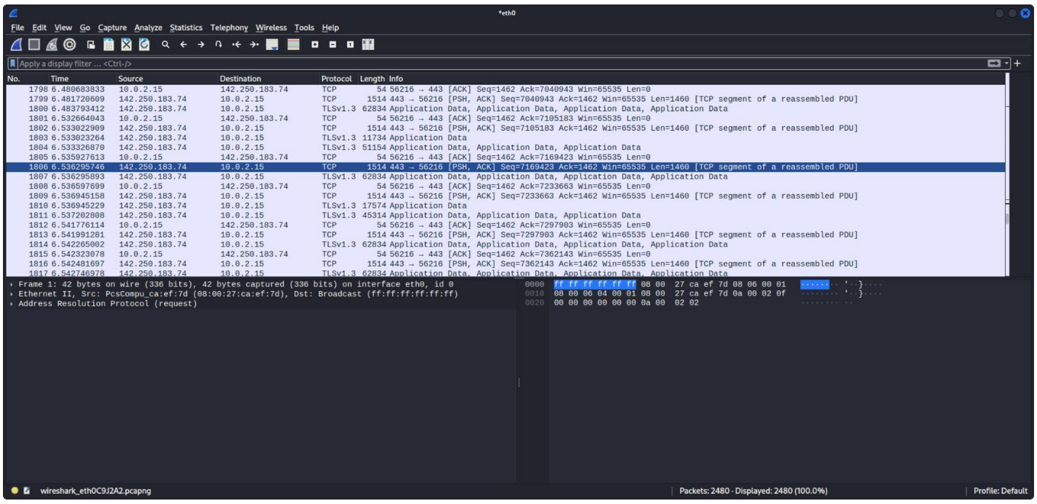
1.TLSv1.3 - The latest version of the Transport Layer Security (TLS) protocol provides secure communication between two endpoints over the internet. It is designed to prevent eavesdropping, tampering, and message forgery. TLSv1.3 improves upon previous protocol versions by eliminating obsolete cryptographic algorithms, enhancing security, and encrypting as much of the handshake as possible2. TLSv1.3 is **RFC 8446.**

2. User Datagram Protocol (UDP): UDP is an alternative communication protocol to TCP for creating loss-tolerating and low-latency connections between different applications. UDP is documented in **RFC 768**.

3. Network Time Protocol (NTP): Network Time Protocol (NTP) is a protocol that synchronizes the clocks of computer systems over data networks. NTP permits network devices to synchronize their time settings with the NTP server. NTP is **RFC 5905.**

4. Dynamic Host Control Protocol (DHCP): Dynamic Host Control Protocol (DHCP) uses a server to allocate an IP address and other configuration information to network devices and is documented in **RFC 3456**. As a result, the device is getting a permission slip from the DHCP server to use the network.

5. Routing Information Protocol (RIP): It constrains the number of hops permitted in a path on a network from the source device to the destination. The maximum number of hops permitted for RIP is fifteen1. It is a routing protocol used to exchange routing information1. It figures the best route based on the hop count. RIP is **RFC 1058.**



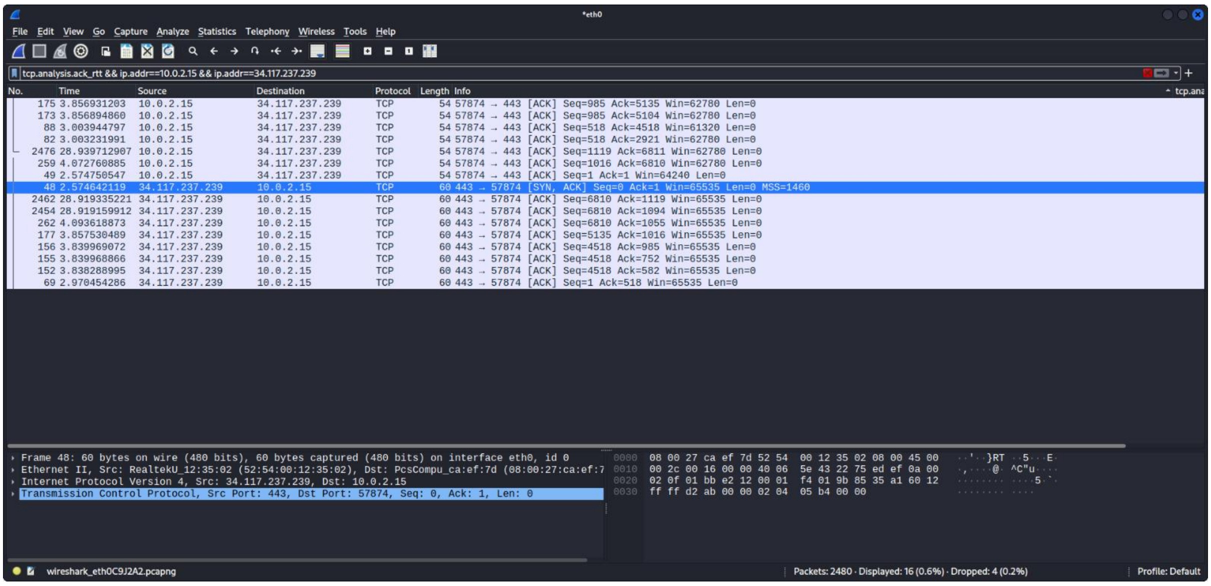
```
No.     Time        Source          Destination     Protocol Length Info
   1798 6.480683833 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7040943 Win=65535 Len=0
   1799 6.481720609 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7040943 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1800 6.483793412 142.250.183.74  10.0.2.15       TLSv1.3 62834 Application Data, Application Data, Application Data, Application Data
   1801 6.532664043 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7105183 Win=65535 Len=0
   1802 6.533022909 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7105183 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1803 6.533023264 142.250.183.74  10.0.2.15       TLSv1.3 11734 Application Data
   1804 6.533326870 142.250.183.74  10.0.2.15       TLSv1.3 51154 Application Data, Application Data, Application Data
   1805 6.535927613 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7169423 Win=65535 Len=0
   1806 6.536295746 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7169423 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1807 6.536295893 142.250.183.74  10.0.2.15       TLSv1.3 62834 Application Data, Application Data, Application Data, Application Data
   1808 6.536597699 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7233663 Win=65535 Len=0
   1809 6.536945158 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7233663 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1810 6.536945229 142.250.183.74  10.0.2.15       TLSv1.3 17574 Application Data
   1811 6.537202808 142.250.183.74  10.0.2.15       TLSv1.3 45314 Application Data, Application Data, Application Data
   1812 6.541776114 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7297903 Win=65535 Len=0
   1813 6.541991281 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7297903 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1814 6.542265002 142.250.183.74  10.0.2.15       TLSv1.3 62834 Application Data, Application Data, Application Data, Application Data
   1815 6.542323078 10.0.2.15       142.250.183.74  TCP        54 56216 → 443 [ACK] Seq=1462 Ack=7362143 Win=65535 Len=0
   1816 6.542481697 142.250.183.74  10.0.2.15       TCP      1514 443 → 56216 [PSH, ACK] Seq=7362143 Ack=1462 Win=65535 Len=1460 [TCP segment of a reassembled PDU]
   1817 6.542746978 142.250.183.74  10.0.2.15       TLSv1.3 62834 Application Data, Application Data, Application Data, Application Data
```

```
› Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
› Ethernet II, Src: PcsCompu_ca:ef:7d (08:00:27:ca:ef:7d), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
› Address Resolution Protocol (request)
```

```
0000  ff ff ff ff ff ff 08 00  27 ca ef 7d 08 06 00 01   ········ '··}····
0010  08 00 06 04 00 01 08 00  27 ca ef 7d 0a 00 02 0f   ········ '··}····
0020  00 00 00 00 00 00 0a 00  02 02                     ········ ··
```

wireshark_eth0C9J2A2.pcapng
Packets: 2480 · Displayed: 2480 (100.0%)
Profile: Default

1b.



*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.analysis.ack_rtt && ip.addr==10.0.2.15 && ip.addr==34.117.237.239

No.   Time          Source          Destination     Protocol Length Info
  175 3.856931203   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=985 Ack=5135 Win=62780 Len=0
  173 3.856894860   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=985 Ack=5104 Win=62780 Len=0
   88 3.003944797   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=518 Ack=4518 Win=61320 Len=0
   82 3.003231991   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=518 Ack=2921 Win=62780 Len=0
 2476 28.939712907  10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=1119 Ack=6811 Win=62780 Len=0
  259 4.072760885   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=1016 Ack=6810 Win=62780 Len=0
   49 2.574750547   10.0.2.15       34.117.237.239  TCP      54 57874 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
   48 2.574642119   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
 2462 28.919335221  34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=6810 Ack=1119 Win=65535 Len=0
 2454 28.919159912  34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=6810 Ack=1094 Win=65535 Len=0
  262 4.093618873   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=6810 Ack=1055 Win=65535 Len=0
  177 3.857530489   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=5135 Ack=1016 Win=65535 Len=0
  156 3.839969072   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=4518 Ack=985 Win=65535 Len=0
  155 3.839968866   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=4518 Ack=752 Win=65535 Len=0
  152 3.838288995   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=4518 Ack=582 Win=65535 Len=0
   69 2.970454286   34.117.237.239  10.0.2.15       TCP      60 443 → 57874 [ACK] Seq=1 Ack=518 Win=65535 Len=0

› Frame 48: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0
› Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_ca:ef:7d (08:00:27:ca:ef:7
› Internet Protocol Version 4, Src: 34.117.237.239, Dst: 10.0.2.15
› Transmission Control Protocol, Src Port: 443, Dst Port: 57874, Seq: 0, Ack: 1, Len: 0

0000  08 00 27 ca ef 7d 52 54  00 12 35 02 08 00 45 00   ··'··}RT ··5···E·
0010  00 2c 00 16 00 00 40 06  5e 43 22 75 ed ef 0a 00   ·,····@· ^C"u····
0020  02 0f 01 bb e2 12 00 01  f4 01 9b 85 35 a1 60 12   ········ ····5·`·
0030  ff ff d2 ab 00 00 02 04  05 b4 00 00               ········ ····

wireshark_eth0C9J2A2.pcapng    Packets: 2480 · Displayed: 16 (0.6%) · Dropped: 4 (0.2%)    Profile: Default
```

Here, the RTT between the source and destination is 2.57s. This is done using the ip.addr property in the search bar.

2.

Application layer protocols used in:

- netflix.com
  - HTTP/3
  - HTTP/1.1
  - HTTP/2
- google.com
  - HTTP/3
  - HTTP/1.1
  - HTTP/2
- github.com
  - HTTP/2
  - HTTP/1.1

HTTP/1.1, **HTTP/2**, and **HTTP/3** are three generations of the Hypertext Transfer Protocol, each with unique features and improvements. **HTTP/1.1**, the oldest, follows a simple request/response model but lacks efficient header compression and multiplexing, which can result in performance issues, especially in high-latency scenarios. **HTTP/2**, introduced as a binary protocol, addresses these concerns with improved header compression and multiplexing support, allowing multiple requests and responses to occur simultaneously on a single connection. Additionally, it introduces prioritization for resource loading.

HTTP/3 represents a significant leap forward. It's built atop the **QUIC** transport protocol, designed for faster and more reliable connections, especially in challenging network conditions. **HTTP/3** retains header compression and multiplexing but does so more efficiently. It's connectionless, eliminating the need for dedicated connection setup/teardown, and emphasizes security by encrypting all traffic by default.

In summary, **HTTP/2** and **HTTP/3** significantly enhance web performance compared to **HTTP/1.1**, thanks to header compression, multiplexing, and reduced latency. **HTTP/3**, with its **QUIC** foundation, takes performance and security to the next level, making it the most advanced choice. However, the selection among these protocols should consider factors like server/client support and application requirements. Upgrading to HTTP/2 or HTTP/3 can substantially boost web performance and security.

3.

The cookies found are as follows:
_ga_ESHC2E4D52=GS1.1.1668848448.1.0.1668848451.0.0.0;
_ga=GA1.1.1403880787.1668848449;
_ga_9JPLGQPDX3=GS1.1.1694194300.5.1.1694194300.60.0.0;
PHPSESSID=qjubungj9rhdoobq5q33ovve54; AIT=55b097c3d2ca71cfa468e033bd31379d

1._ga, _ga_ESHC2E4D52, and _ga_9JPLGQPDX3 are Google Analytics cookies. They are used to distinguish unique users by assigning a randomly generated number as a client identifier. They are included in each page request in a site and used to calculate visitor, session, and campaign data for the site's analytics reports1.
2.PHPSESSID is a cookie generated by applications based on the PHP language. It is used to maintain user session variables, such as logged-in status or shopping cart contents. The name of the cookie is a reference to PHP's default session handling function, session_start()1.