

ASSIGNMENT NO:01

Q1) Briefly explain HTTP protocol. Demonstrate a comparison of HTTP request and response messages as learned from the textbook with the Wireshark output.

ANS:

HTTP protocol:

- 1) Hypertext Transfer Protocol is what it stands for.
- 2) It is a set of explicit guidelines for communication between a client (the network resource asking for data or services) and a server (the resource that receives and responds to the request).
- 3) The guidelines for resource requests and answers between web clients and servers are laid out in the HTTP protocol.
- 4) In the seven-layer OSI networking model, HTTP is an application layer protocol that standardizes communication between computing or telecommunications systems, regardless of underlying internal structure and technology.
- 5) An multinational group known as the Internet Engineering Task Force is now in charge of the defining and continuous development of this protocol (IETF).

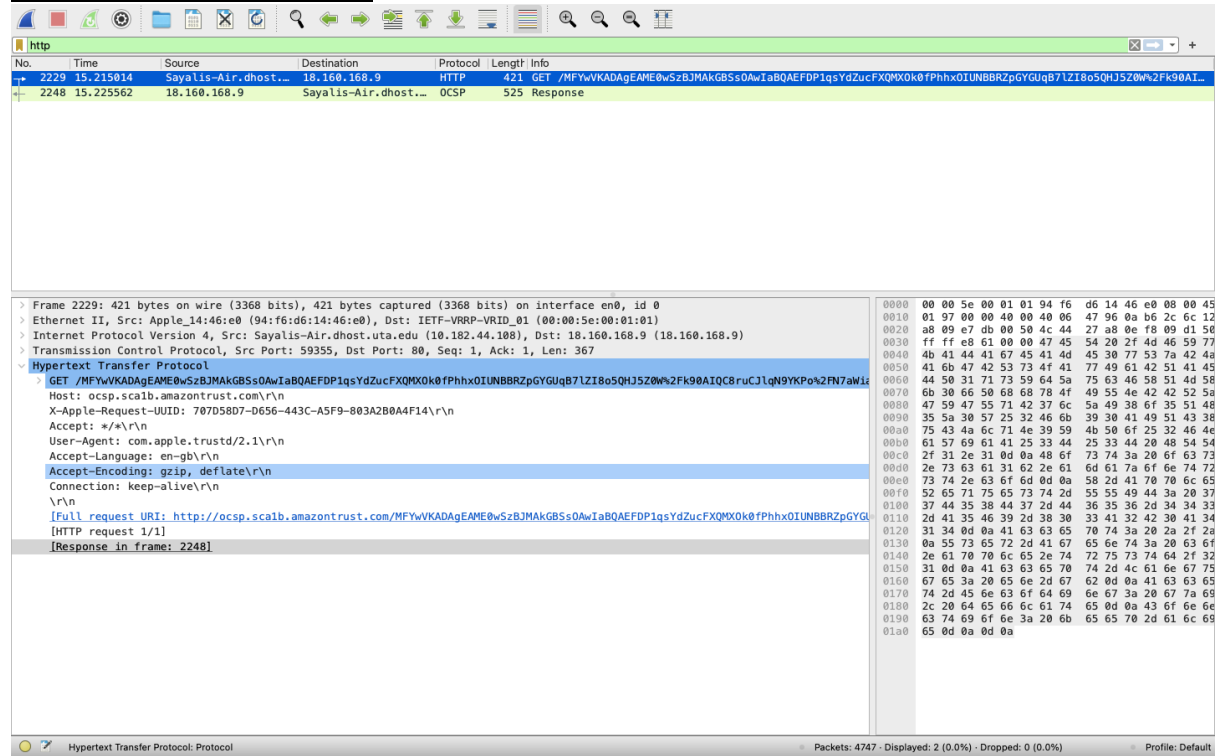
Comparison of HTTP request and response messages:**HTTP request:**

- 1) HTTP requests are messages that the client sends to the server to start an action. Three components make up their starting point:
- 2) An HTTP method is a verb or noun (such as GET, PUT, or POST) that specifies the action to be taken. For instance, the terms GET and POST denote when a resource should be fetched from and when data should be posted to a server, respectively (creating or modifying a resource, or generating a temporary document to send back).
The request context often identifies the request target, which is either a URL or the absolute path of the protocol, port, and domain. This request target's format differs depending on the HTTP method.
When connected to a proxy, GET typically uses the absolute form of a URL, often known as the entire URL.
The domain name and, optionally, the port make up the authority form, which is a part of a URL. When creating an HTTP tunnel, it is only used in conjunction with CONNECT. JOIN developer.mozilla.org at port 80 HTTP/1.1
A basic asterisk (*) is used with OPTIONS to represent the server as a whole in the asterisk form.
- 3) The HTTP version serves as a signal of the anticipated version to use for the response by defining the structure of the remaining message.

HTTP response:

- 1) The status line, which appears at the beginning of an HTTP response, includes the following data:
- 2) often HTTP/1.1, the protocol version, a code that represents the request's success or failure. Status codes 200, 404, or 302 are frequently used.
- 3) an update message. A succinct, primarily informative text explanation of the status code that aids in understanding the HTTP message by humans.

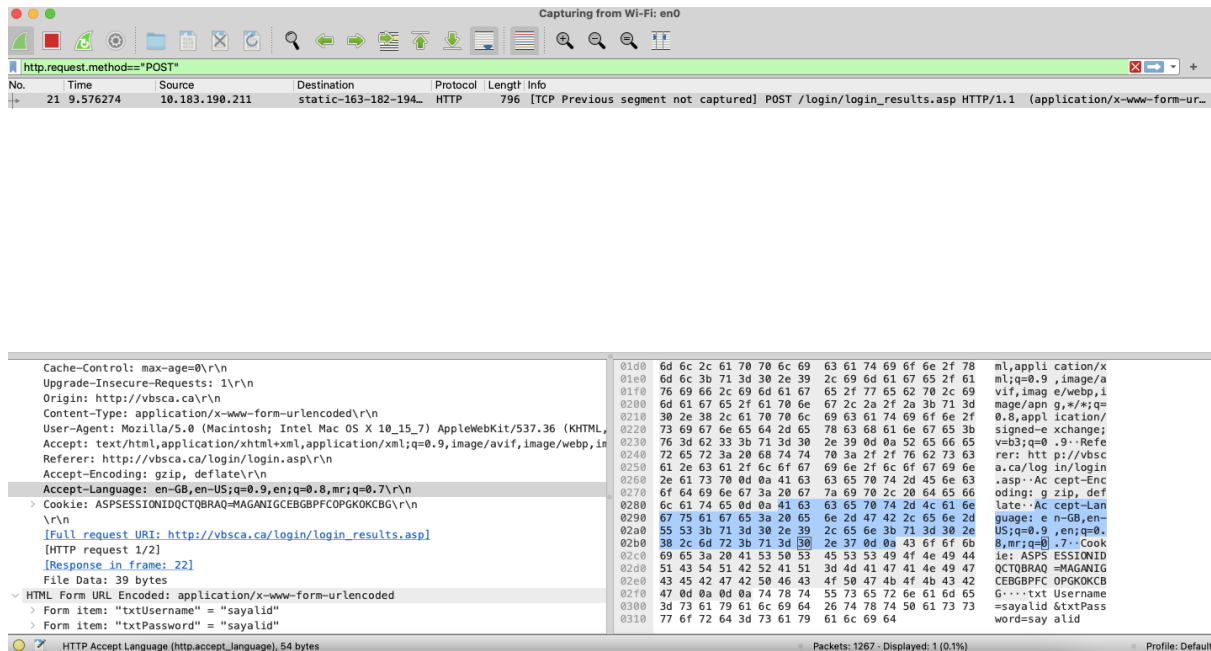
Screenshot from Wireshark:



Q2) Go to any unsecured website (Example: <http://vbsca.ca/login/login.asp>). You will be prompted to enter login and password. Once you are done with that, use Wireshark to check if the password is encrypted or not. You must be able to find the entered username and password as plain text among the packets exchanged shown on Wireshark. Provide a screenshot of the username and password seen on Wireshark.

ANS:

Screenshot of the username and password seen on Wireshark.



Q3)Using Wireshark demonstrate TCP three-way handshake.

ANS:

TCP three-way handshake

- 1)A TCP/IP network connection procedure known as the 3-Way handshake links the server and client. Both the client and the server must exchange synchronization and acknowledgment packets before the actual data transmission can start.
- 2)Prior to data transmission, the 3-way handshake process is intended to allow both communication ends to simultaneously establish and negotiate the network TCP socket connection specifications. It enables the simultaneous transport of a large number of TCP socket connections in both directions.

Screenshots:

Wireshark packet capture analysis showing network traffic details, including packet list, packet details, and packet bytes.

Packet List:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::c5e:ad27:697...	ff02::2	ICMPv6	70	Router Solicitation from 4e:36:93:2a:b3:85
17	2.802225	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	78	56769 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2157016246 TSecr=0 SACK_PERM
20	2.869569	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	74	443 → 56769 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 MSS=1406 SACK_PERM TSval=2395435707 TSecr=215
21	2.869569	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=1 Ack=1 Win=132416 Len=0 TSval=2157016313 TSecr=2395435707
23	3.072090	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	66	443 → 56769 [ACK] Seq=1 Ack=518 Win=28160 Len=0 TSval=2395435777 TSecr=2157016314
25	3.073051	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1460	443 → 56769 [ACK] Seq=1395 Ack=518 Win=28160 Len=1394 TSval=2395435777 TSecr=2157016314 [TCP seg
26	3.073176	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=518 Ack=2789 Win=129600 Len=0 TSval=2157016514 TSecr=2395435777
27	3.073303	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1460	443 → 56769 [ACK] Seq=2789 Ack=518 Win=28160 Len=1394 TSval=2395435777 TSecr=2157016314 [TCP seg
29	3.074431	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=518 Ack=5407 Win=128448 Len=0 TSval=2157016516 TSecr=2395435777
30	3.075360	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1290	[TCP Spurious Retransmission] 443 → 56769 [PSH, ACK] Seq=1395 Ack=518 Win=28160 Len=1224 TSval=2
31	3.075463	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	78	[TCP Window Update] 56769 → 443 [ACK] Seq=518 Ack=5407 Win=131072 Len=0 TSval=2157016517 TSecr=2
34	3.277040	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=644 Ack=5458 Win=131008 Len=0 TSval=2157016718 TSecr=2395435993
36	3.277672	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=644 Ack=5527 Win=130944 Len=0 TSval=2157016718 TSecr=2395435993
44	3.482129	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	66	443 → 56769 [ACK] Seq=5527 Ack=747 Win=28160 Len=0 TSval=2395436182 TSecr=2157016719
46	3.482989	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=1644 Ack=5565 Win=131008 Len=0 TSval=2157016921 TSecr=2395436182

Packet Details:

Frame 17: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface eth0
Ethernet II, Src: Apple_14:46:e0 (94:f6:d6:14:46:e0), Dst: IETF-VRRP-VRID_01 (08:00:00:00:00:00)
Internet Protocol Version 4, Src: Sayalis-Air.dhost.uta.edu (10.182.44.108), Dst: femetrics.grammarly.io (10.182.214.201)
Transmission Control Protocol, Src Port: 56769, Dst Port: 443, Seq: 0, Len: 0

Packet Bytes:

0000 00 00 5e 00 01 01 94 f6 d6 14 46 e0 08 00 45 00 ...F...E...
0010 00 40 00 00 40 00 00 06 1a 00 0a b6 2c 6c 12 cd ...@...k...
0020 d6 c9 dd c1 01 bb 74 c7 79 44 00 00 00 00 b0 02 ...t...y0...
0030 ff ff 53 73 00 00 02 04 05 b4 01 03 03 06 01 01 ...S...t...
0040 08 0a 80 91 74 b6 00 00 00 04 02 00 00 00 ...t...t...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::c5e:ad27:697...	ff02::2	ICMPv6	70	Router Solicitation from 4e:36:93:2a:b3:85
17	2.802225	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	78	56769 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2157016246 TSecr=0 SACK_PERM
20	2.869569	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	74	443 → 56769 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 MSS=1406 SACK_PERM TSval=2395435707 TSecr=215
21	2.869699	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=1 Ack=1 Win=132416 Len=0 TSval=2157016313 TSecr=2395435707
23	3.072090	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	66	443 → 56769 [ACK] Seq=1 Ack=518 Win=28160 Len=0 TSval=2395435777 TSecr=2157016314
25	3.073051	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1460	443 → 56769 [ACK] Seq=1395 Ack=518 Win=28160 Len=1394 TSval=2395435777 TSecr=2157016314 [TCP seg
26	3.073176	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=518 Ack=2789 Win=129600 Len=0 TSval=2157016514 TSecr=2395435777
27	3.074303	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1460	443 → 56769 [ACK] Seq=2789 Ack=518 Win=28160 Len=1394 TSval=2395435777 TSecr=2157016314 [TCP seg
29	3.074431	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=518 Ack=5407 Win=128448 Len=0 TSval=2157016516 TSecr=2395435777
30	3.075360	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	1290	[TCP Spurious Retransmission] 443 → 56769 [PSH, ACK] Seq=1395 Ack=518 Win=28160 Len=1224 TSval=2
31	3.075463	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	78	[TCP Window Update] 56769 → 443 [ACK] Seq=518 Ack=5407 Win=131072 Len=0 TSval=2157016517 TSecr=2
34	3.277040	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=644 Ack=5458 Win=131008 Len=0 TSval=2157016718 TSecr=2395435993
36	3.277672	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=644 Ack=5527 Win=130944 Len=0 TSval=2157016718 TSecr=2395435993
44	3.482129	femetrics.grammarL...	Sayalis-Air.dhost...	TCP	66	443 → 56769 [ACK] Seq=5527 Ack=747 Win=28160 Len=0 TSval=2395436182 TSecr=2157016719
46	3.482989	Sayalis-Air.dhost...	femetrics.grammarL...	TCP	66	56769 → 443 [ACK] Seq=1644 Ack=5565 Win=131008 Len=0 TSval=2157016921 TSecr=2395436182

> Frame 21: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface en0, id 0
 > Ethernet II, Src: Apple_14:46:e0 (94:f6:d6:14:46:e0), Dst: IETF-VRRP-VRID_01 (00:00:5e:00:01:01)
 > Internet Protocol Version 4, Src: Sayalis-Air.dhost.uta.edu (10.182.44.188), Dst: femetrics.gram
 > Transmission Control Protocol, Src Port: 56769, Dst Port: 443, Seq: 1, Ack: 1, Len: 0

wireshark_Wi-FiB5AVT1.pcapng Packets: 793 · Displayed: 270 (34.0%) · Dropped: 0 (0.0%) Profile: Default

Q4) Explain the Wireshark output of following filters

ANS:

TCP interactions are considered complete when they have both the opening and closing handshakes, regardless of any data transfer.

For instance, the filter "tcp.completeness==7" will detect a conversation that just involves a three-way handshake

The lengthier filter will identify a conversation that also involves data transfer

FIN or RST packets, or even both, can be used to signify the closing of a connection

'tcp.completeness==31.

a) tcp.completeness == 7

1) Here in Transmission Control Protocol, It uses the source port = 61413 and destination port 443 and regarding conversation completeness, it shows incomplete on data(15).

tcp.completeness == 7

No.	Time	Source	Destination	Protocol	Length	Info
33	13.089686	10.219.142.81	doh.opendns.com	TLSv1	618	Client Hello
60	13.185087	10.219.142.81	52.114.128.92	TLSv1	571	Client Hello
63	13.206990	10.219.142.81	52.114.128.91	TLSv1	571	Client Hello
121	13.639017	10.219.142.81	52.113.194.132	TLSv1	647	Client Hello
139	13.669493	10.219.142.81	52.113.194.132	TLSv1	654	Client Hello
186	13.717758	10.219.142.81	onedscolprdcus01.c...	TLSv1	571	Client Hello
342	43.034282	10.219.142.81	ec2-23-21-6-92.c...	TLSv1	583	Client Hello
380	47.000842	10.219.142.81	doh.opendns.com	TLSv1	618	Client Hello
399	47.127990	10.219.142.81	52.112.95.99	TLSv1	583	Client Hello

> Frame 33: 618 bytes on wire (4944 bits), 618 bytes captured (4944 bits) on interface en0, id 0
 > Ethernet II, Src: Apple_14:46:e0 (94:f6:d6:14:46:e0), Dst: Cisco_27:00:00 (00:25:83:27:00:00)
 > Internet Protocol Version 4, Src: 10.219.142.81 (10.219.142.81), Dst: doh.opendns.com (146.112.41.2)
 > Transmission Control Protocol, Src Port: 61413, Dst Port: 443, Seq: 1, Ack: 1, Len: 552
 > Source Port: 61413
 > Destination Port: 443
 > [Stream index: 9]
 > [Conversation completeness: Incomplete, DATA (15)]
 > [TCP Segment Len: 552]
 > Sequence Number: 1 (relative sequence number)
 > Sequence Number (raw): 1400398339
 > [Next Sequence Number: 553 (relative sequence number)]
 > Acknowledgment Number: 1 (relative ack number)
 > Acknowledgment number (raw): 2340880083
 > 1000 = Header Length: 32 bytes (8)
 > Flags: 0x018 (PSH, ACK)
 > Window: 2061
 > [Calculated window size: 131904]
 > [Window size scaling factor: 64]
 > Checksum: 0x6d41 [unverified]
 > [Checksum Status: Unverified]
 > Urgent Pointer: 0
 > Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
 > [Timestamps]
 > [SEQ/ACK analysis]
 > TCP payload (552 bytes)
 > Transport Layer Security

0000 00 25 83 27 00 00 94 f6 d6 14 46 e0 00 00 45
 0010 02 5c 00 00 40 00 00 06 e3 fd 0a d0 8e 51 92
 0020 29 02 ef e5 01 bb 53 78 62 03 8b 86 fe d3 80
 0030 08 0d 6d 41 00 00 01 01 08 0a 90 b8 14 70 71
 0040 48 88 16 03 01 02 23 01 00 02 1f 03 03 03 dc
 0050 ac 52 a7 ce 1a da 0c 31 0e 1b aa 92 91 3b c2
 0060 4d 63 62 48 0f 67 51 9c 5a 2a 98 f2 c6 20 7b
 0070 4d 18 08 5e a0 10 9a cd 14 30 cc c7 d2 30 bf
 0080 0f ff 1e 68 03 b1 ea 64 5c 68 e9 31 2d c3 00
 0090 3a 3a 13 01 13 02 13 03 c0 2b c0 2f c0 2c c0
 00a0 cc a9 cc a8 c0 13 c0 14 00 9c 00 9d 00 2f 00
 00b0 01 00 01 b6 2a 2a 00 00 00 00 14 00 12 00
 00c0 0f 64 6f 68 2e 6f 70 05 6e 64 6e 73 2e 63 6f
 00d0 00 17 00 00 ff 01 00 01 00 00 0a 00 0a 00 00
 00e0 6a 00 1d 00 17 00 18 00 0b 00 02 01 00 00 23
 00f0 00 00 10 00 0e 00 0c 02 68 32 08 68 74 74 70
 0100 31 2e 31 00 05 00 05 01 00 00 00 00 0d 00 00
 0110 00 10 04 03 00 04 04 01 05 03 00 05 05 01 00
 0120 06 01 00 12 00 00 00 33 00 2b 00 29 6a 6a 00
 0130 00 00 1d 00 20 25 09 47 c7 33 b4 a4 c9 f6 c0
 0140 f6 8d aa 1f 10 0e a8 b4 9b 56 6f 7a c7 d5 c1
 0150 cf 53 4c c4 6a 00 2d 00 02 01 01 00 2b 00 07
 0160 3a 3a 03 04 03 03 00 1b 00 03 02 00 02 44 69
 0170 05 00 03 02 68 32 6a 6a 00 01 00 00 29 00 eb
 0180 c6 00 c0 70 09 34 79 a8 11 70 fc 4d cb 2a 88
 0190 93 7d 96 62 73 6e 0d a7 3e 01 b5 38 f1 2e 75
 01a0 c9 58 cf 0a c1 6a e7 f2 41 0a b7 f8 00 c5 a3
 01b0 d8 c8 b7 3c 59 d4 51 be b1 0a 62 84 bd 88 31
 01c0 bb 3d 87 4d 9b ba d6 76 13 8c be 00 18 47 7e
 01d0 64 92 80 d6 90 dc 49 47 af 05 a0 bf 98 18 4d
 01e0 e0 fd bd 2c e3 23 56 a5 04 f5 c8 90 ae 22 46
 01f0 2f f6 3a be 00 55 16 d3 68 44 1e de 07 52 e2
 0200 ac 01 d5 69 26 29 e7 17 1b 35 11 3f 1b ba 56
 0210 e2 91 05 8d 69 92 fa 65 c5 ed 69 1d c4 c7 05

Wi-Fi: en0: <live capture in progress> Packets: 534 - Displayed: 9 (1.7%) Profile: Default

b)tcp.completeness == 31

1)Here in Transmission Control Protocol, It uses the source port = 56895 and destination port 443 and regarding conversion completeness, it shows complete with data(31).

tcp.completeness == 31

No.	Time	Source	Destination	Protocol	Length	Info
100	16.179100	Sayalis-Air.dhost...	s.s002.iad.prod.t2...	TCP	66	56895 → 443 [ACK] Seq=611 Ack=4776 Win=131072 Len=0 TSval=2822309635 TSecr=1447337874

> Frame 100: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface en0, id 0
 > Ethernet II, Src: Apple_14:46:e0 (94:f6:d6:14:46:e0), Dst: IETF-VRRP-VRID_01 (00:00:5e:00:01:01)
 > Internet Protocol Version 4, Src: 10.182.44.108 (10.182.44.108), Dst: s.s002.iad.prod.t200.ter.int.amaz
 > Transmission Control Protocol, Src Port: 56895, Dst Port: 443, Seq: 611, Ack: 4776, Len: 0
 > Source Port: 56895
 > Destination Port: 443
 > [Stream index: 3]
 > [Conversation completeness: Complete, WITH_DATA (31)]
 > [TCP Segment Len: 0]
 > Sequence Number: 611 (relative sequence number)
 > Sequence Number (raw): 1577075699
 > [Next Sequence Number: 611 (relative sequence number)]
 > Acknowledgment Number: 4776 (relative ack number)
 > Acknowledgment number (raw): 189979791
 > 1000 = Header Length: 32 bytes (8)
 > Flags: 0x010 (ACK)
 > Window: 2048
 > [Calculated window size: 131072]
 > [Window size scaling factor: 64]
 > Checksum: 0x767b [unverified]
 > [Checksum Status: Unverified]
 > Urgent Pointer: 0
 > Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
 > [Timestamps]
 > [SEQ/ACK analysis]

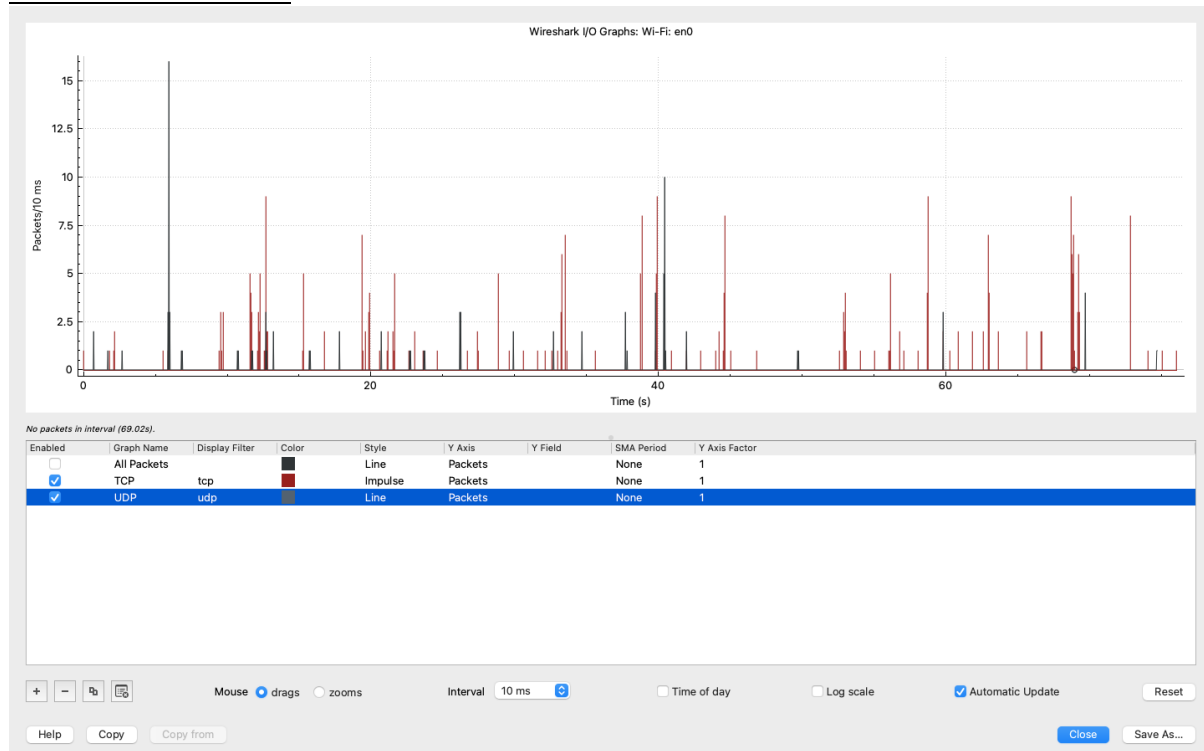
0000 00 00 5e 00 01 01 94 f6 d6 14 46 e0 00 00 45 00
 0010 00 34 00 00 40 00 00 06 51 66 0a b6 2c 6c 36 9f
 0020 7b 9d de 3f 01 bb 5e 00 43 f3 0b 52 dc 8f 00 10
 0030 08 00 76 7b 00 00 01 01 08 0a a8 39 07 03 56 44
 0040 9f 92

wireshark_Wi-Fi3BQT1.pcapng Packets: 123 - Displayed: 1 (0.8%) Profile: Default

Q5)Using Wireshark observe packets that follows DNS and TCP. Plot a graph for the same (in Wireshark) and provide screenshot for an interval of per 10ms.

By observing the packet that follows DNS and TCP here we are plotting a graph in Wireshark for an interval of per 10ms.

GRAPH SCREENSHOT:

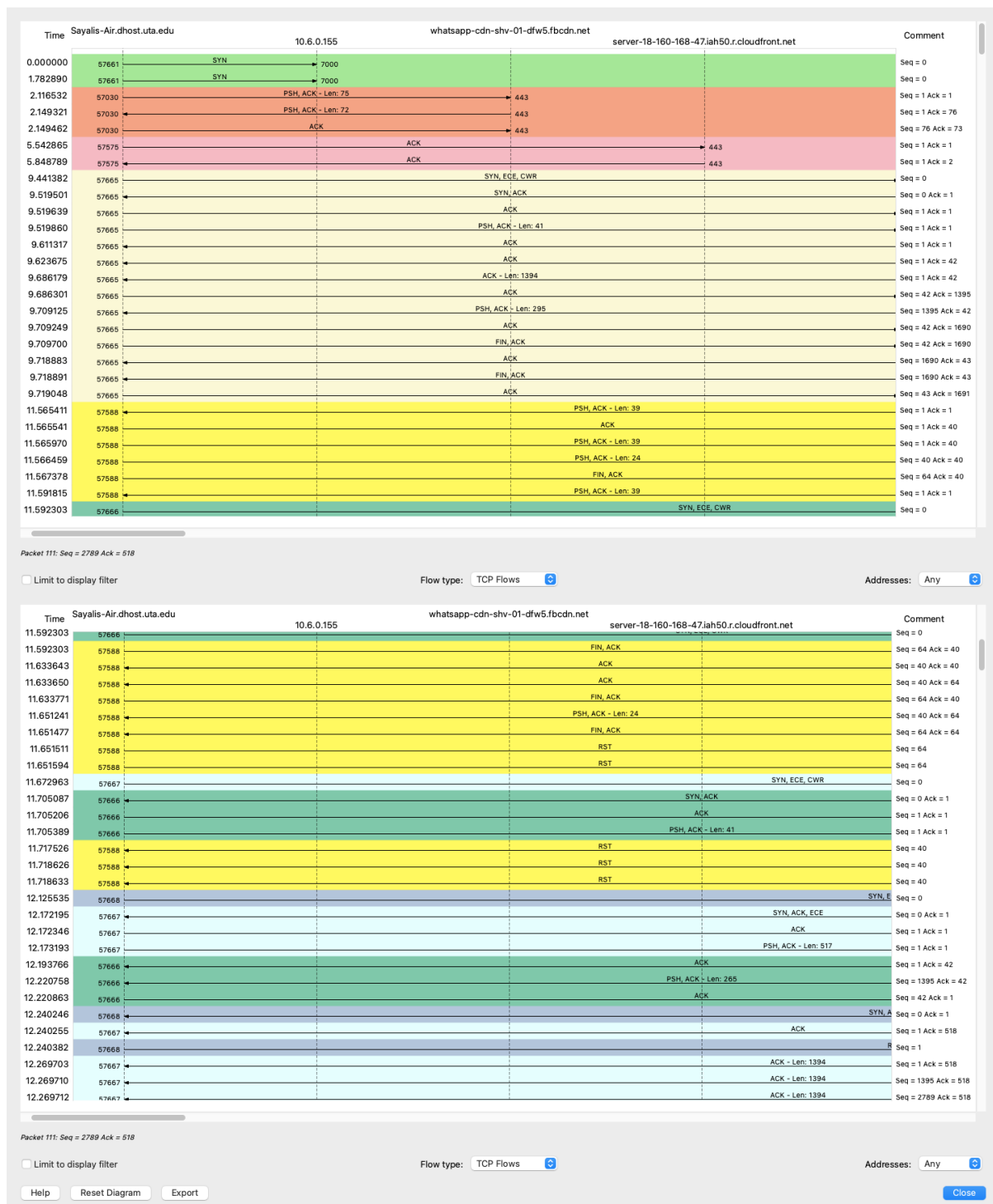


Q6)For a request, capture the TCP flow using the Wireshark Flow chart that highlights the [SYN], [SYN,ACK], [ACK] and [FIN,ACK]. Explain what it does.

TCP Flows:

- 1) When a client tries to establish a TCP connection with a server, SYN packets are typically created, and the client and server exchange a series of messages that typically go like this
- 2)By communicating with the server via a SYN (synchronize) message, the client seeks a connection.
- 3)This request is acknowledged by the server by returning SYN-ACK to the client. After receiving an ACK from the client, the connection is established.
- 4) Acknowledgment Code, or ACK, refers to a service that is offered by mail companies to inform the sender of a letter that the recipient has received the delivery. It is usually a form signed by the receiver and then delivered to the sender. This gives proof to the sender that the letter has been received.
- 5) The sender sends TCP FIN to the receiver for an outgoing stream. The packet has a FIN flag set as another type of TCP message. The packet has a sequence number, the receiver sends the FIN Ack with one more sequence number received in the FIN. Now the connection is closed in one direction.

Screenshots:



Q7) Briefly explain the function of DNS? Provide a screenshot of Wireshark that includes the Source Port and Destination port for the DNS queried message

ANS:

Function of DNS:

- 1) A hostname is transformed into an IP address that computers can understand as part of the DNS resolution process.
- 2) Each Internet-connected device has a unique IP address, which is required to identify the right item, just as a street address is required to identify a certain residence.

3)When a user requests a webpage to load, a translation between what they type into their web browser and the machine-friendly address required to find the webpage must take place.

4)DNS Server has 4 name server and they plays a role:

1. DNS recursor - The recursor is comparable to a librarian who is asked to look for a specific book in a library. The DNS recursor is a server made to take requests from client machines using programs like web browsers. The recursor is typically thereafter in charge of submitting further queries to respond to the client's DNS query.

2. The root nameserver: IT is the first stage in converting human readable host names into IP addresses (resolving). It can be compared to an index that directs readers to certain book racks in a library; often, it acts as a guide to other, more precise locations.

3. TLD nameserver - A top-level domain (TLD) server might be compared to a particular shelf of books in a library. This nameserver, which hosts the final part of a hostname (in the case of example.com, the TLD server is "com"), is the next stage in the process of locating a specific IP address.

4. Authoritative nameserver - This last nameserver can be compared to a dictionary on a shelf of books, allowing one to look up a specific name and get its definition. In the nameserver inquiry, the authoritative nameserver is the last stop. If the authoritative name server has access to the requested record, it will provide the DNS Recursor (the librarian) with the IP address for the requested hostname.

Source Port and Destination port screenshot :

The screenshot displays a Wireshark capture of DNS traffic. The packet list on the left shows a series of queries and responses between Sayalis-Air.dhost.uta.edu and erdhcwpapp01.ad.uta.edu. The packet details pane on the right shows the structure of a DNS query (Standard query) for PTR 108.44.182.10. The packet bytes pane at the bottom shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
2	0.678052	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0x2186 PTR 108.44.182.10.in-addr.arpa
3	0.678212	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	83	Standard query 0xd768 PTR 155.0.6.10.in-addr.arpa
4	0.680724	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	125	Standard query response 0x2186 PTR 108.44.182.10.in-addr.arpa PTR Sayalis-Air.dhost.uta.edu
5	0.681915	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	168	Standard query response 0xd768 No such name PTR 155.0.6.10.in-addr.arpa SOA erdhcwpapp01.ad.uta.edu
6	1.677599	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	83	Standard query 0xad94 PTR 3.0.247.10.in-addr.arpa
7	1.680667	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	121	Standard query response 0xad94 PTR 3.0.247.10.in-addr.arpa PTR erdhcwpapp01.ad.uta.edu
12	2.678503	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0x5a22 PTR 53.19.240.157.in-addr.arpa
13	2.693606	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	134	Standard query response 0x5a22 PTR 53.19.240.157.in-addr.arpa PTR whatsapp-cdn-shv-01-dfw5.fbcdn.net
16	5.885274	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	79	Standard query 0x7ebd A clients4.google.com
17	5.885447	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	79	Standard query 0xe412 HTTPS clients4.google.com
18	5.891530	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	153	Standard query response 0xe412 HTTPS clients4.google.com CNAME clients.l.google.com SOA ns1.google.com
19	5.891757	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	119	Standard query response 0x7ebd A clients4.google.com CNAME clients.l.google.com A 142.251.33.46
55	6.813964	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0x639a PTR 47.168.160.18.in-addr.arpa
56	6.855604	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	143	Standard query response 0x639a PTR 47.168.160.18.in-addr.arpa PTR server-18-160-168-47.iah50.r.cloudfront.net
71	10.677724	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	85	Standard query 0x5357 PTR 59.44.182.10.in-addr.arpa
72	10.753603	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	126	Standard query response 0x5357 PTR 59.44.182.10.in-addr.arpa PTR Hardiks-Air-2.dhost.uta.edu
90	11.671379	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	82	Standard query 0xc6b7 A femetrics.grammarly.io
98	11.746734	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	210	Standard query response 0xc6b7 A femetrics.grammarly.io A 52.206.199.142 A 34.202.76.136 A 54.157.140.136
130	12.677417	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	85	Standard query 0xc69c PTR 53.23.182.10.in-addr.arpa
131	12.677558	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0xb5cf PTR 103.154.134.3.in-addr.arpa
132	12.677675	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	87	Standard query 0xeca8 PTR 187.126.225.34.in-addr.arpa
136	12.766250	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	125	Standard query response 0xc69c PTR 53.23.182.10.in-addr.arpa PTR macbooke-pro.dhost.uta.edu
140	13.220820	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	149	Standard query response 0xb5cf PTR 103.154.134.3.in-addr.arpa PTR ec2-3-134-154-103.us-east-2.compute.amazonaws.com
141	13.224727	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	143	Standard query response 0xeca8 PTR 187.126.225.34.in-addr.arpa PTR ec2-34-225-126-187.compute-1.amazonaws.com
158	15.728590	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0xf0d3 PTR 120.46.182.10.in-addr.arpa
159	15.731282	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	129	Standard query response 0xf0d3 PTR 120.46.182.10.in-addr.arpa PTR DESKTOP-0Q805GF.dhost.uta.edu
163	17.772436	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	85	Standard query 0xdd3a PTR 12.92.115.52.in-addr.arpa
164	17.775467	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	171	Standard query response 0xdd3a No such name PTR 12.92.115.52.in-addr.arpa SOA ns1-02.azure-dns.com
189	20.688713	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	85	Standard query 0x9b8b PTR 88.44.182.10.in-addr.arpa
190	20.688713	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0xb176 PTR 27.185.183.10.in-addr.arpa
191	20.690507	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	126	Standard query response 0xb176 PTR 27.185.183.10.in-addr.arpa PTR Galaxy-Note9.guest.uta.edu
192	20.690843	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	128	Standard query response 0x9b8b PTR 88.44.182.10.in-addr.arpa PTR LAPTOP-6V03H1I1.dhost.uta.edu
213	22.676763	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	86	Standard query 0x9a54 PTR 123.35.104.34.in-addr.arpa
214	22.727131	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	138	Standard query response 0x9a54 PTR 123.35.104.34.in-addr.arpa PTR 123.35.104.34.bc.googleusercontent.com
219	23.676790	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	87	Standard query 0x3755 PTR 205.133.114.52.in-addr.arpa
220	23.717831	Sayalis-Air.dhost.uta.edu	erdhcwpapp01.ad.uta.edu	DNS	161	Standard query response 0x3755 No such name PTR 205.133.114.52.in-addr.arpa SOA ns1-201.azure-dns.com

Frame 2: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface en0, id 1
Ethernet II, Src: Sayalis-Air.dhost.uta.edu (94:f6:d6:14:46:e8), Dst: IETF-VRRP-VRID_01
Internet Protocol Version 4, Src: Sayalis-Air.dhost.uta.edu (10.182.44.108), Dst: erdhcwpapp01.ad.uta.edu (10.182.44.108)
User Datagram Protocol, Src Port: 56371, Dst Port: 53
Domain Name System (query)

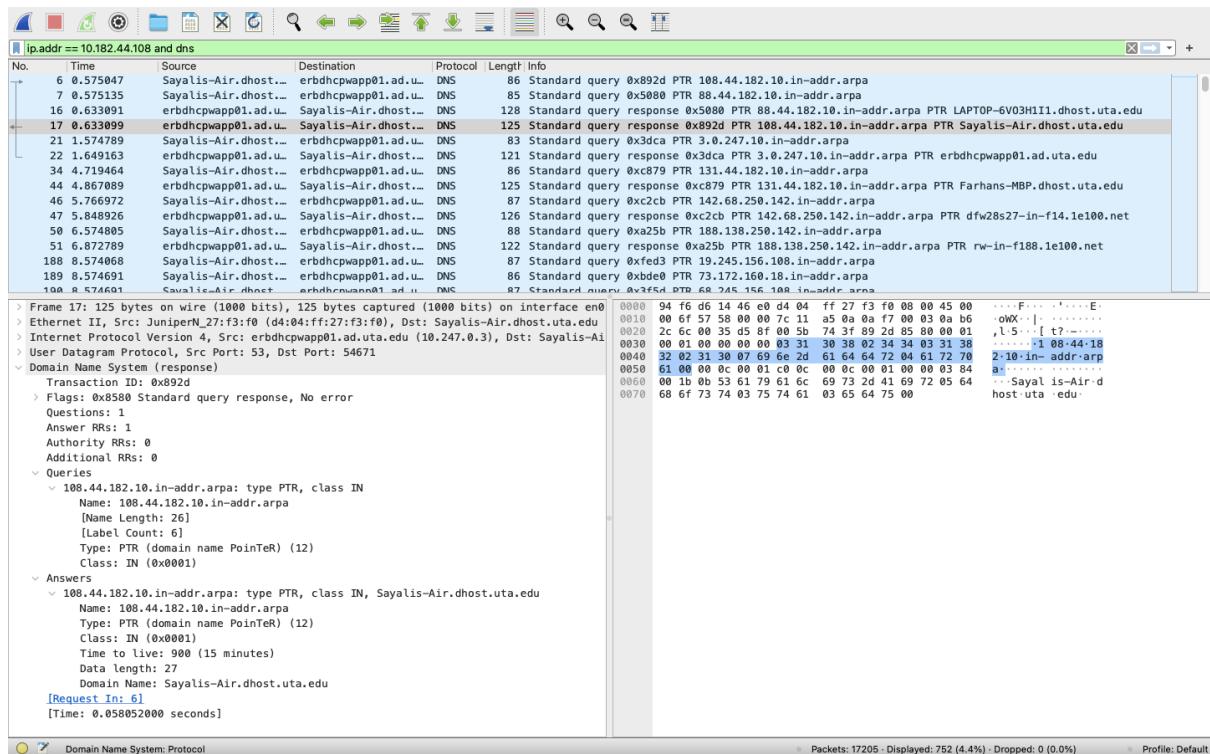
Packets: 582 · Displayed: 72 (12.4%) · Dropped: 0 (0.0%) · Profile: Default

Q8) Locate the DNS query and response messages. Are they sent over UDP or TCP?

ANS:

Packets are sent over UDP.

Screenshot of Query and response message:



Q9) What are the different types of DNS records? Using Wireshark examine the DNS query message and write the “TYPE” of the DNS record.

ANS:

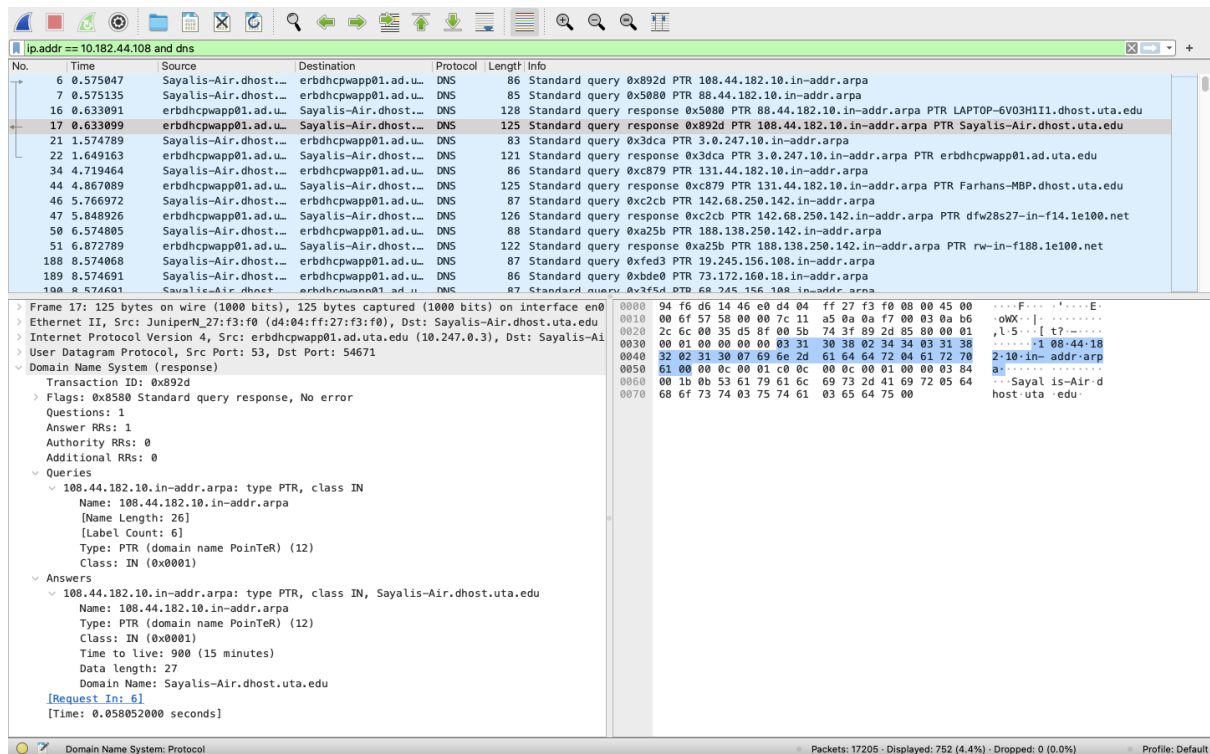
types of DNS records:

- A record
- AAAA record
- CNAME record
- Nameserver (NS) record
- Mail exchange (MX) record
- SOA record
- TXT record
- PTR record
- SRV record
- CERT record
- DCHID
- DNAME

Screenshot of the type and query message:

Type in the snapshot is PTR:

A pointer (PTR) record provides a domain name for reverse lookup. It's the opposite of an A record as it provides the domain name linked to an IP address instead of the IP address for a domain.



Q10) What are Authoritative and Recursive nameservers? Demonstrate either of them using Wireshark.

ANS:

- 1) When attempting to connect to a website, your queries will be processed by one of two different types of servers.
- 2) Authoritative and Recursive DNS servers are the ones that reply to your requests and maintain the canonical data that specifies which IP address corresponds to which domain.
- 3) The "mappings" of your domain names to IP addresses are kept on Authoritative DNS servers, to put it briefly. System administrators typically configure this domain name to IP mapping. When someone visits a website, Recursive DNS servers are contacted for lookups. Recursive DNS servers then inquire about the solution from the required Authoritative Name Server. The individual who requests the information will then receive this response from the Recursive name server.
- 4) The mainstays of the DNS lookup process are recursive servers. In order to provide the correct IP for the inquiring client, they frequently need to perform several DNS lookups. These servers are often run by an ISP (Internet Service Provider) or specialized DNS providers. For instance, Google manages its own public recursive DNS servers.

Screenshot showing Recursive nameservers:

No.	Time	Source	Destination	Protocol	Length	Info
3638	28.357241	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	171	Standard query response 0xb340 No such name PTR 12.92.115.52.in-addr.arpa SOA ns1-02.azure-dns.com
3649	29.347838	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	85	Standard query 0xc0e2 PTR 78.30.18.104.in-addr.arpa
3650	29.349243	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	147	Standard query response 0xc0e2 No such name PTR 78.30.18.104.in-addr.arpa SOA cruz.ns.cloudflare.com
4667	43.583890	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	82	Standard query 0xebad A femetrics.grammarly.io
4668	43.584586	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	83	Standard query 0x9c6c HTTPS treatment.grammarly.com
4669	43.584734	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	83	Standard query 0xf296 A treatment.grammarly.com
4672	43.589452	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	232	Standard query response 0x9c6c HTTPS treatment.grammarly.com CNAME public-treatment.prod-experimentation.grammarlyaws.com
4673	43.591028	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	114	Standard query 0x8af0 HTTPS public-treatment.prod-experimentation.grammarlyaws.com
4674	43.591663	Sayalis-Air.dhost...	erbdhpcwapp01.ad.u...	DNS	114	Standard query 0x1ad0 A public-treatment.prod-experimentation.grammarlyaws.com
4675	43.596959	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	210	Standard query response 0xebad A femetrics.grammarly.io A 52.206.199.142 A 54.157.213.17 A 34.234.1
4676	43.596965	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	276	Standard query response 0xf296 A treatment.grammarly.com CNAME public-treatment.prod-experimentation.grammarlyaws.com
4677	43.618614	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	198	Standard query response 0x8af0 HTTPS public-treatment.prod-experimentation.grammarlyaws.com SOA ns1-02.azure-dns.com
4678	43.633281	erbdhpcwapp01.ad.u...	Sayalis-Air.dhost...	DNS	242	Standard query response 0x1ad0 A public-treatment.prod-experimentation.grammarlyaws.com A 44.194.1

<p>> Frame 7: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface en0, id 0</p> <p>> Ethernet II, Src: Apple14:46:e0 (94:f6:d6:14:46:e0), Dst: IETF-VRRP-VRID_01 (00:00:5e:00:01:01)</p> <p>> Internet Protocol Version 4, Src: Sayalis-Air.dhost.uta.edu (10.182.44.108), Dst: erbdhpcwapp01.ad.uta.edu (10.247.0.3)</p> <p>> User Datagram Protocol, Src Port: 51903, Dst Port: 53</p> <p>> Domain Name System (query)</p> <p>Transaction ID: 0x91e3</p> <p>Flags: 0x0100 Standard query</p> <p>0... .. = Response: Message is a query</p> <p>.000 0... .. = Opcode: Standard query (0)</p> <p>... .. = Truncated: Message is not truncated</p> <p>... .. = Recursion desired: Do query recursively</p> <p>... .. = Z: reserved (0)</p> <p>... .. = Non-authenticated data: Unacceptable</p> <p>Questions: 1</p> <p>Answer RRs: 0</p> <p>Authority RRs: 0</p> <p>Additional RRs: 0</p> <p>Queries</p> <p>108.44.182.10.in-addr.arpa: type PTR, class IN</p> <p>Name: 108.44.182.10.in-addr.arpa</p> <p>[Name Length: 26]</p> <p>[Label Count: 6]</p> <p>Type: PTR (domain name Pointer) (12)</p> <p>Class: IN (0x0001)</p> <p>[Response In: 9]</p>	<pre> 0000 00 00 5e 00 01 01 94 f6 d6 14 46 e0 00 00 45 0010 00 48 c1 b7 00 00 40 11 76 d2 0a b6 2c 6c 0a 0020 00 03 ca bf 00 35 00 34 4e 3a 91 e3 01 00 00 0030 00 00 00 00 00 00 03 31 30 38 02 34 34 03 31 0040 32 02 31 30 07 69 6e 2d 61 64 64 72 04 61 72 0050 61 00 00 0c 00 01 </pre>
--	---

References:

<https://youtu.be/bEXEEfbNADs>

https://www.youtube.com/watch?v=C0cX_AqMul8

https://www.wireshark.org/docs/wsug_html_chunked/ChAdvTCPAnalysis.html

<https://www.youtube.com/watch?v=qYh6k-S5xC4>

[https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Web/HTTP/Messages#:~:text=HTTP%20messages%20are%20how%20data,the%20answer%20from%20the%20server)

[US/docs/Web/HTTP/Messages#:~:text=HTTP%20messages%20are%20how%20data,the%20](https://developer.mozilla.org/en-US/docs/Web/HTTP/Messages#:~:text=HTTP%20messages%20are%20how%20data,the%20answer%20from%20the%20server)

[answer%20from%20the%20server](https://www.site24x7.com/learn/dns-record-types.html)

<https://www.site24x7.com/learn/dns-record-types.html>

https://developer.mozilla.org/en-US/docs/Glossary/TCP_handshake

<https://www.cspsprotocol.com/tcp-connection-termination/>