NAME – INDRANIL BAIN

ENROLLMENT NO. – 2020CSB039

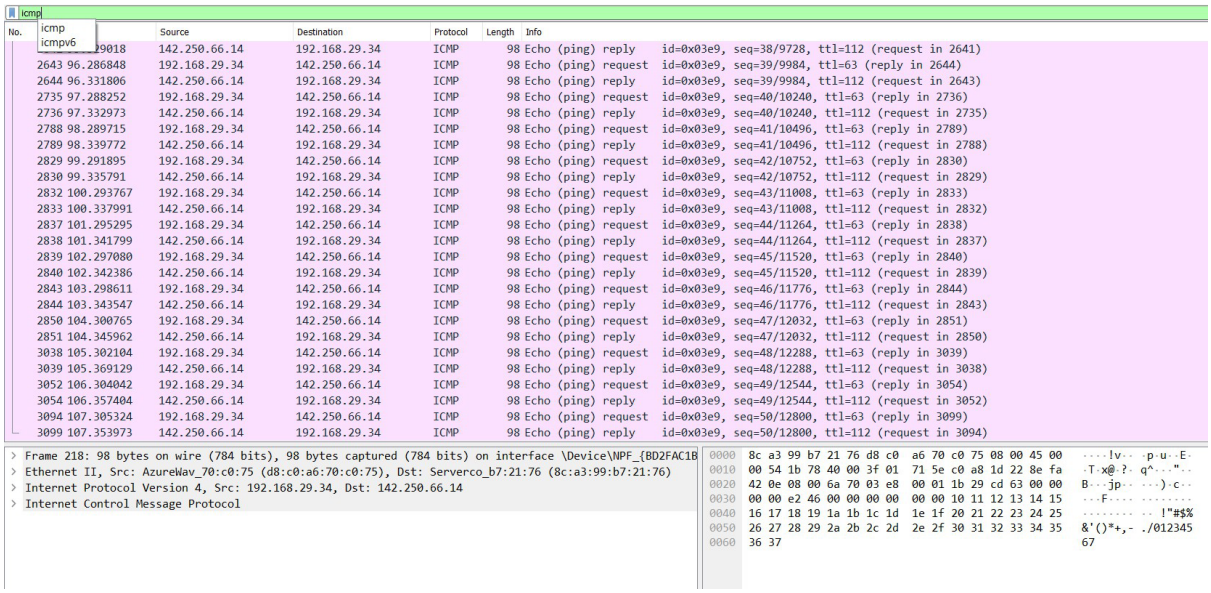
GROUP – GX

SUBJECT – COMPUTER NETWORK LAB

1. Analysing Packets for given commands –

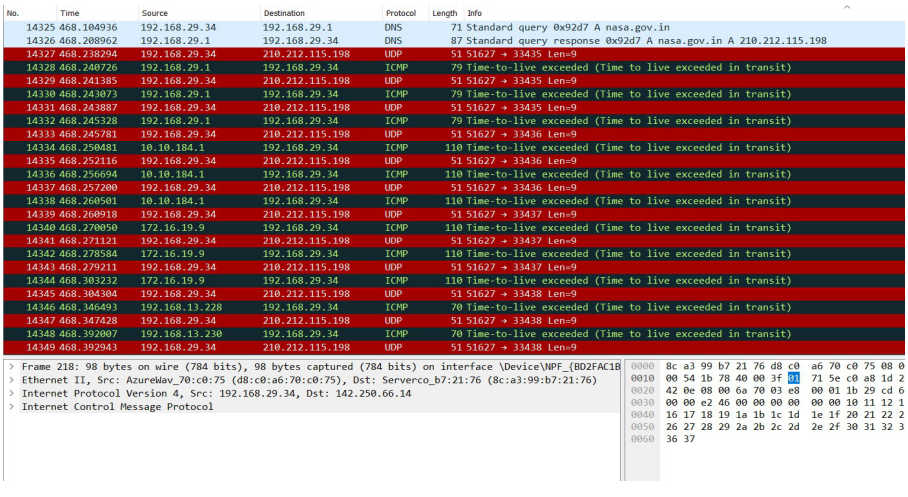
(i) **ping:**

I ran the command ping -c google.com to get results like below –



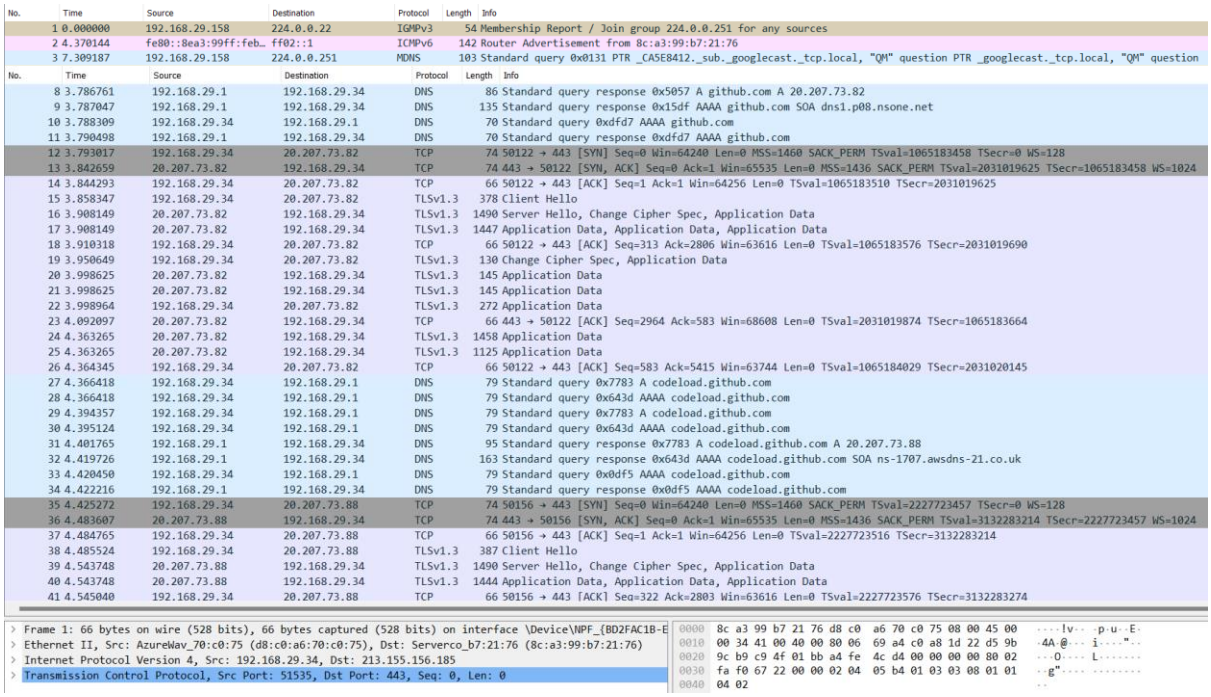
(ii) **traceroute:**

Running traceroute ‘isro.gov.in ’ gives the result –



(iii) **arp:**

Running arp gave the following result –



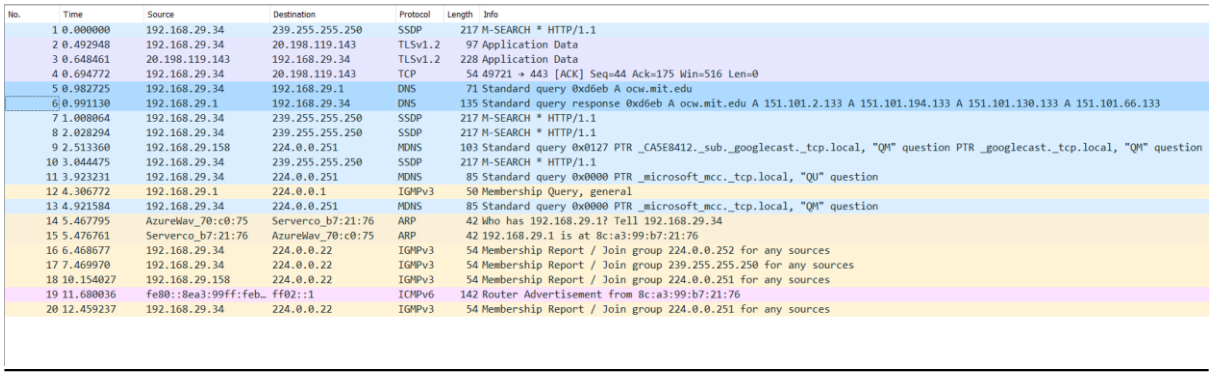
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.29.158	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
2	4.370144	fe80::8ea3:99ff:feb…	ff02::1	ICMPv6	142	Router Advertisement from 8c:a3:99:b7:21:76
3	7.309187	192.168.29.158	224.0.0.251	MDNS	103	Standard query 0x0131 PTR _CA5E8412._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp.local, "QM" question
8	3.786761	192.168.29.1	192.168.29.34	DNS	86	Standard query response 0x5057 A github.com A 20.207.73.82
9	3.787047	192.168.29.1	192.168.29.34	DNS	135	Standard query response 0x15df AAAA github.com SOA dns1.p08.nsone.net
10	3.788309	192.168.29.34	192.168.29.1	DNS	70	Standard query 0xdfd7 AAAA github.com
11	3.790498	192.168.29.1	192.168.29.34	DNS	70	Standard query response 0xdfd7 AAAA github.com
12	3.793017	192.168.29.34	20.207.73.82	TCP	74	50122 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1065183458 TSecr=0 WS=128
13	3.842659	20.207.73.82	192.168.29.34	TCP	74	443 → 50122 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1436 SACK_PERM TSval=2031019625 TSecr=1065183458 WS=1024
14	3.844293	192.168.29.34	20.207.73.82	TCP	66	50122 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1065183510 TSecr=2031019625
15	3.858347	192.168.29.34	20.207.73.82	TLSv1.3	378	Client Hello
16	3.908149	20.207.73.82	192.168.29.34	TLSv1.3	1490	Server Hello, Change Cipher Spec, Application Data
17	3.908149	20.207.73.82	192.168.29.34	TLSv1.3	1447	Application Data, Application Data, Application Data
18	3.910318	192.168.29.34	20.207.73.82	TCP	66	50122 → 443 [ACK] Seq=313 Ack=2806 Win=63616 Len=0 TSval=1065183576 TSecr=2031019690
19	3.950649	192.168.29.34	20.207.73.82	TLSv1.3	130	Change Cipher Spec, Application Data
20	3.998625	20.207.73.82	192.168.29.34	TLSv1.3	145	Application Data
21	3.998625	20.207.73.82	192.168.29.34	TLSv1.3	145	Application Data
22	3.998964	192.168.29.34	20.207.73.82	TLSv1.3	272	Application Data
23	4.092097	20.207.73.82	192.168.29.34	TCP	66	443 → 50122 [ACK] Seq=2964 Ack=583 Win=68608 Len=0 TSval=2031019874 TSecr=1065183664
24	4.363265	20.207.73.82	192.168.29.34	TLSv1.3	1458	Application Data
25	4.363265	20.207.73.82	192.168.29.34	TLSv1.3	1125	Application Data
26	4.364345	192.168.29.34	20.207.73.82	TCP	66	50122 → 443 [ACK] Seq=583 Ack=5415 Win=63744 Len=0 TSval=1065184029 TSecr=2031020145
27	4.366418	192.168.29.34	192.168.29.1	DNS	79	Standard query 0x7783 A codeload.github.com
28	4.366418	192.168.29.34	192.168.29.1	DNS	79	Standard query 0x643d AAAA codeload.github.com
29	4.394357	192.168.29.34	192.168.29.1	DNS	79	Standard query 0x7783 A codeload.github.com
30	4.395124	192.168.29.34	192.168.29.1	DNS	79	Standard query 0x643d AAAA codeload.github.com
31	4.401765	192.168.29.1	192.168.29.34	DNS	95	Standard query response 0x7783 A codeload.github.com A 20.207.73.88
32	4.419726	192.168.29.1	192.168.29.34	DNS	163	Standard query response 0x643d AAAA codeload.github.com SOA ns-1707.awsdns-21.co.uk
33	4.420450	192.168.29.34	192.168.29.1	DNS	79	Standard query 0x0df5 AAAA codeload.github.com
34	4.422216	192.168.29.1	192.168.29.34	DNS	79	Standard query response 0x0df5 AAAA codeload.github.com
35	4.425272	192.168.29.34	20.207.73.88	TCP	74	50156 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=2227723457 TSecr=0 WS=128
36	4.483607	20.207.73.88	192.168.29.34	TCP	74	443 → 50156 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1436 SACK_PERM TSval=3132283214 TSecr=2227723457 WS=1024
37	4.484765	192.168.29.34	20.207.73.88	TCP	66	50156 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2227723516 TSecr=3132283214
38	4.485524	192.168.29.34	20.207.73.88	TLSv1.3	387	Client Hello
39	4.543748	20.207.73.88	192.168.29.34	TLSv1.3	1490	Server Hello, Change Cipher Spec, Application Data
40	4.543748	20.207.73.88	192.168.29.34	TLSv1.3	1444	Application Data, Application Data, Application Data
41	4.545040	192.168.29.34	20.207.73.88	TCP	66	50156 → 443 [ACK] Seq=322 Ack=2803 Win=63616 Len=0 TSval=2227723576 TSecr=3132283274

```
> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{BD2FAC1B-E
> Ethernet II, Src: AzureWav_70:c0:75 (d8:c0:a6:70:c0:75), Dst: Serverco_b7:21:76 (8c:a3:99:b7:21:76)
> Internet Protocol Version 4, Src: 192.168.29.34, Dst: 213.155.156.185
> Transmission Control Protocol, Src Port: 51535, Dst Port: 443, Seq: 0, Len: 0
```

```
0000  8c a3 99 b7 21 76 d8 c0  a6 70 c0 75 08 00 45 00   ····!v·· ·p·u··E·
0010  00 34 41 00 40 00 80 06  69 a4 c0 a8 1d 22 d5 9b   ·4A·@··· i····"··
0020  9c b9 c9 4f 01 bb a4 fe  4c d4 00 00 00 00 80 02   ···O···· L·······
0030  fa f0 67 22 00 00 02 04  05 b4 01 03 03 08 01 01   ··g"···· ········
0040  04 02                                             ··
```

(iv) **dig:**

Running dig ocw.mit.edu gives me the following result –



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.29.34	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
2	0.492948	192.168.29.34	20.198.119.143	TLSv1.2	97	Application Data
3	0.648461	20.198.119.143	192.168.29.34	TLSv1.2	228	Application Data
4	0.694772	192.168.29.34	20.198.119.143	TCP	54	49721 → 443 [ACK] Seq=44 Ack=175 Win=516 Len=0
5	0.982725	192.168.29.34	192.168.29.1	DNS	71	Standard query 0xd6eb A ocw.mit.edu
6	0.991130	192.168.29.1	192.168.29.34	DNS	135	Standard query response 0xd6eb A ocw.mit.edu A 151.101.2.133 A 151.101.194.133 A 151.101.130.133 A 151.101.66.133
7	1.008064	192.168.29.34	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
8	2.028294	192.168.29.34	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
9	2.513360	192.168.29.158	224.0.0.251	MDNS	103	Standard query 0x0127 PTR _CA5E8412._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp.local, "QM" question
10	3.044475	192.168.29.34	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11	3.923231	192.168.29.34	224.0.0.251	MDNS	85	Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QU" question
12	4.306772	192.168.29.1	224.0.0.1	IGMPv3	50	Membership Query, general
13	4.921584	192.168.29.34	224.0.0.251	MDNS	85	Standard query 0x0000 PTR _microsoft_mcc._tcp.local, "QM" question
14	5.467795	AzureWav_70:c0:75	Serverco_b7:21:76	ARP	42	Who has 192.168.29.1? Tell 192.168.29.34
15	5.476761	Serverco_b7:21:76	AzureWav_70:c0:75	ARP	42	192.168.29.1 is at 8c:a3:99:b7:21:76
16	6.468677	192.168.29.34	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.252 for any sources
17	7.469970	192.168.29.34	224.0.0.22	IGMPv3	54	Membership Report / Join group 239.255.255.250 for any sources
18	10.154027	192.168.29.158	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
19	11.680036	fe80::8ea3:99ff:feb…	ff02::1	ICMPv6	142	Router Advertisement from 8c:a3:99:b7:21:76
20	12.459237	192.168.29.34	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources

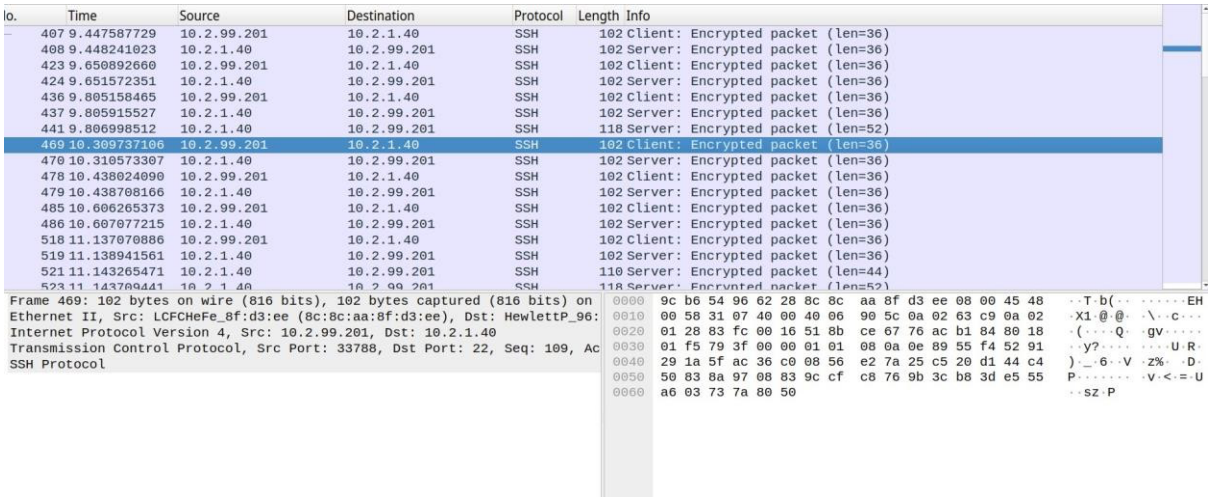
(v) **wget:**

I downloaded a **github** repository using **wget**. The packet capture is shown below –

2. [This question cannot be solved because telnet daemon is not available anymore.]

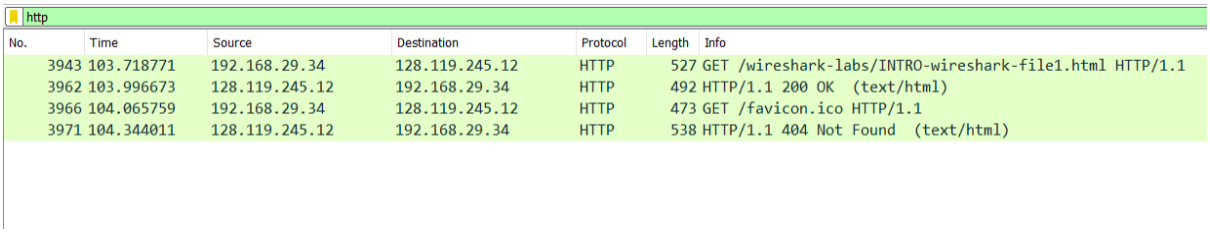
3. While sending SSH request to the local server of IIEST (hamsa) and analysing the network traffic using Wireshark tool, we would be able to see all the packets that were transferred during this session. These packets include the initial connection request, the exchange of authentication information (such as username and password), and the subsequent data transfer between client and server. The data is encrypted hence not human-readable, but we can view the different packets and their corresponding size.

   The packet data is shown below-



4. I entered the URL http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html in Chrome browser and started packet capture. Once the file download confirmation was shown, I stopped capturing packets.

The packet capture is shown below –



http

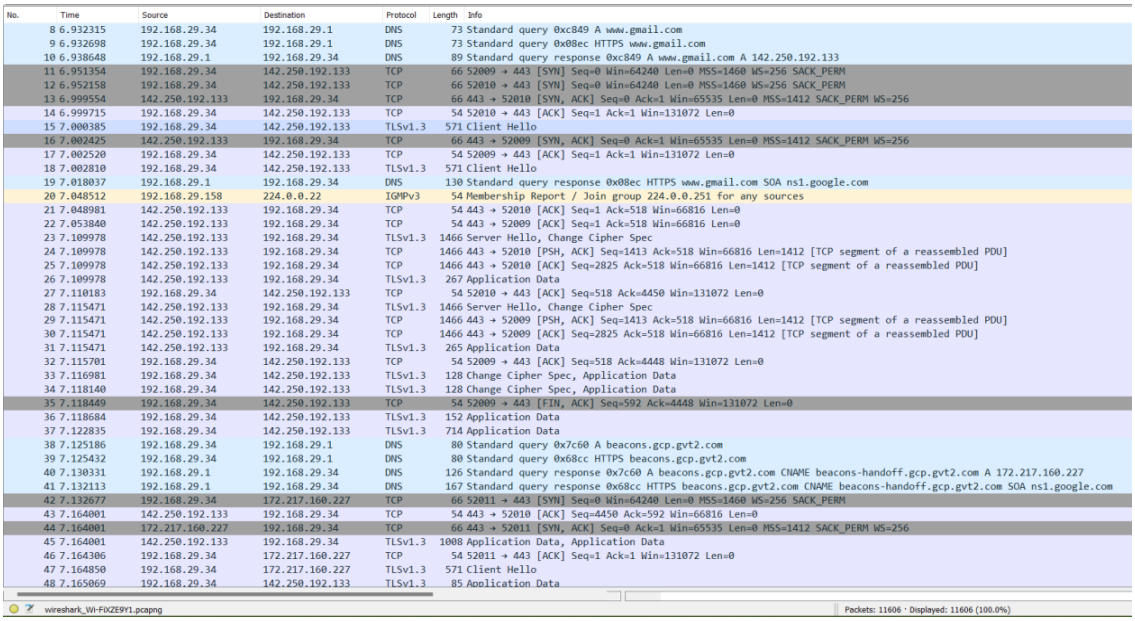
No.	Time	Source	Destination	Protocol	Length	Info
3943	103.718771	192.168.29.34	128.119.245.12	HTTP	527	GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
3962	103.996673	128.119.245.12	192.168.29.34	HTTP	492	HTTP/1.1 200 OK (text/html)
3966	104.065759	192.168.29.34	128.119.245.12	HTTP	473	GET /favicon.ico HTTP/1.1
3971	104.344011	128.119.245.12	192.168.29.34	HTTP	538	HTTP/1.1 404 Not Found (text/html)

a) We can see from the screenshot that HTTP GET was sent at Time 103.718771 and HTTP OK was received at Time 103.996673. So, it took 277.902 milliseconds.

b) The IP address of gaia.cs.umass.edu is 128.119.245.12 whereas the IP address of my device is 192.168.29.34. This is evident from the above screenshot.

5. I started the Wireshark packet capture service and then I typed the URL https://www.gmail.com in my browser. I stopped the packet capture after I have given my username and password to log on to Gmail.

The screenshot below shows the capture –



No.	Time	Source	Destination	Protocol	Length	Info
8	6.932315	192.168.29.34	192.168.29.1	DNS	73	Standard query 0xc849 A www.gmail.com
9	6.932698	192.168.29.34	192.168.29.1	DNS	73	Standard query 0x08ec HTTPS www.gmail.com
10	6.938648	192.168.29.1	192.168.29.34	DNS	89	Standard query response 0xc849 A www.gmail.com A 142.250.192.133
11	6.951354	192.168.29.34	142.250.192.133	TCP	66	52009 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
12	6.952158	192.168.29.34	142.250.192.133	TCP	66	52010 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
13	6.999554	142.250.192.133	192.168.29.34	TCP	66	443 → 52010 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
14	6.999715	192.168.29.34	142.250.192.133	TCP	54	52010 → 443 [ACK] Seq=1 Ack=1 Win=131072 Len=0
15	7.000385	192.168.29.34	142.250.192.133	TLSv1.3	571	Client Hello
16	7.002425	142.250.192.133	192.168.29.34	TCP	66	443 → 52009 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
17	7.002520	192.168.29.34	142.250.192.133	TCP	54	52009 → 443 [ACK] Seq=1 Ack=1 Win=131072 Len=0
18	7.002810	192.168.29.34	142.250.192.133	TLSv1.3	571	Client Hello
19	7.018037	192.168.29.1	192.168.29.34	DNS	130	Standard query response 0x08ec HTTPS www.gmail.com SOA ns1.google.com
20	7.048512	192.168.29.158	224.0.0.22	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
21	7.048981	142.250.192.133	192.168.29.34	TCP	54	443 → 52010 [ACK] Seq=1 Ack=518 Win=66816 Len=0
22	7.053840	142.250.192.133	192.168.29.34	TCP	54	443 → 52009 [ACK] Seq=1 Ack=518 Win=66816 Len=0
23	7.109978	142.250.192.133	192.168.29.34	TLSv1.3	1466	Server Hello, Change Cipher Spec
24	7.109978	142.250.192.133	192.168.29.34	TCP	1466	443 → 52010 [PSH, ACK] Seq=1413 Ack=518 Win=66816 Len=1412 [TCP segment of a reassembled PDU]
25	7.109978	142.250.192.133	192.168.29.34	TCP	1466	443 → 52010 [ACK] Seq=2825 Ack=518 Win=66816 Len=1412 [TCP segment of a reassembled PDU]
26	7.109978	142.250.192.133	192.168.29.34	TLSv1.3	267	Application Data
27	7.110183	192.168.29.34	142.250.192.133	TCP	54	52010 → 443 [ACK] Seq=518 Ack=4450 Win=131072 Len=0
28	7.115471	142.250.192.133	192.168.29.34	TLSv1.3	1466	Server Hello, Change Cipher Spec
29	7.115471	142.250.192.133	192.168.29.34	TCP	1466	443 → 52009 [PSH, ACK] Seq=1413 Ack=518 Win=66816 Len=1412 [TCP segment of a reassembled PDU]
30	7.115471	142.250.192.133	192.168.29.34	TCP	1466	443 → 52009 [ACK] Seq=2825 Ack=518 Win=66816 Len=1412 [TCP segment of a reassembled PDU]
31	7.115471	142.250.192.133	192.168.29.34	TLSv1.3	265	Application Data
32	7.115701	192.168.29.34	142.250.192.133	TCP	54	52009 → 443 [ACK] Seq=518 Ack=4448 Win=131072 Len=0
33	7.116981	192.168.29.34	142.250.192.133	TLSv1.3	128	Change Cipher Spec, Application Data
34	7.118140	192.168.29.34	142.250.192.133	TLSv1.3	128	Change Cipher Spec, Application Data
35	7.118449	192.168.29.34	142.250.192.133	TCP	54	52009 → 443 [FIN, ACK] Seq=592 Ack=4448 Win=131072 Len=0
36	7.118684	192.168.29.34	142.250.192.133	TLSv1.3	152	Application Data
37	7.122835	192.168.29.34	142.250.192.133	TLSv1.3	714	Application Data
38	7.125186	192.168.29.34	192.168.29.1	DNS	80	Standard query 0x7c60 A beacons.gcp.gvt2.com
39	7.125432	192.168.29.34	192.168.29.1	DNS	80	Standard query 0x68cc HTTPS beacons.gcp.gvt2.com
40	7.130331	192.168.29.1	192.168.29.34	DNS	126	Standard query response 0x7c60 A beacons.gcp.gvt2.com CNAME beacons-handoff.gcp.gvt2.com A 172.217.160.227
41	7.132113	192.168.29.1	192.168.29.34	DNS	167	Standard query response 0x68cc HTTPS beacons.gcp.gvt2.com CNAME beacons-handoff.gcp.gvt2.com SOA ns1.google.com
42	7.132677	192.168.29.34	172.217.160.227	TCP	66	52011 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
43	7.164001	142.250.192.133	192.168.29.34	TCP	54	443 → 52010 [ACK] Seq=4450 Ack=592 Win=66816 Len=0
44	7.164001	172.217.160.227	192.168.29.34	TCP	66	443 → 52011 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
45	7.164001	142.250.192.133	192.168.29.34	TLSv1.3	1008	Application Data, Application Data
46	7.164306	192.168.29.34	172.217.160.227	TCP	54	52011 → 443 [ACK] Seq=1 Ack=1 Win=131072 Len=0
47	7.164850	192.168.29.34	172.217.160.227	TLSv1.3	571	Client Hello
48	7.165069	192.168.29.34	142.250.192.133	TLSv1.3	85	Application Data

wireshark_Wi-FiXZE9Y1.pcapng — Packets: 11606 · Displayed: 11606 (100.0%)

a) Yes, the current application layer protocol is TLS. A primary use case of TLS is encrypting the communication between web applications and servers, such as web-browsers loading a website.
But, in the previous case, it was HTTP.

b) Previously, the protocol used was simple HTTP. As a result, sniffers like Wireshark could decode the data. But https://www.gmail.com is using HTTPS with TLS. So, the data is encrypted and Wireshark cannot decode it.

-:-