CS536 Lab3: A Synthesis Example via Wireshark

Name: Mansi Shinde

PUID: 0034784153

1. Part A: Use Wireshark to capture packets over Wi-Fi.

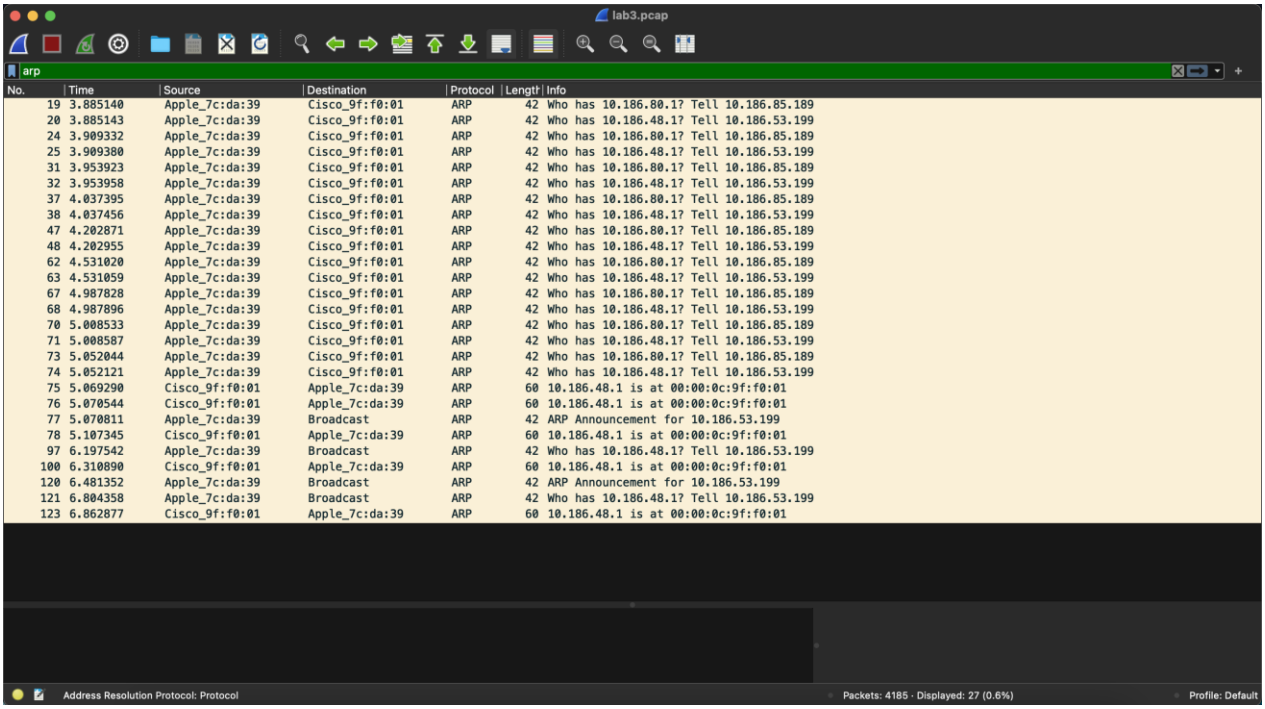
Please see the attached lab3.pcap file

2. Part B: Please use Wireshark to check the trace captured in Part A and then answer the following questions.

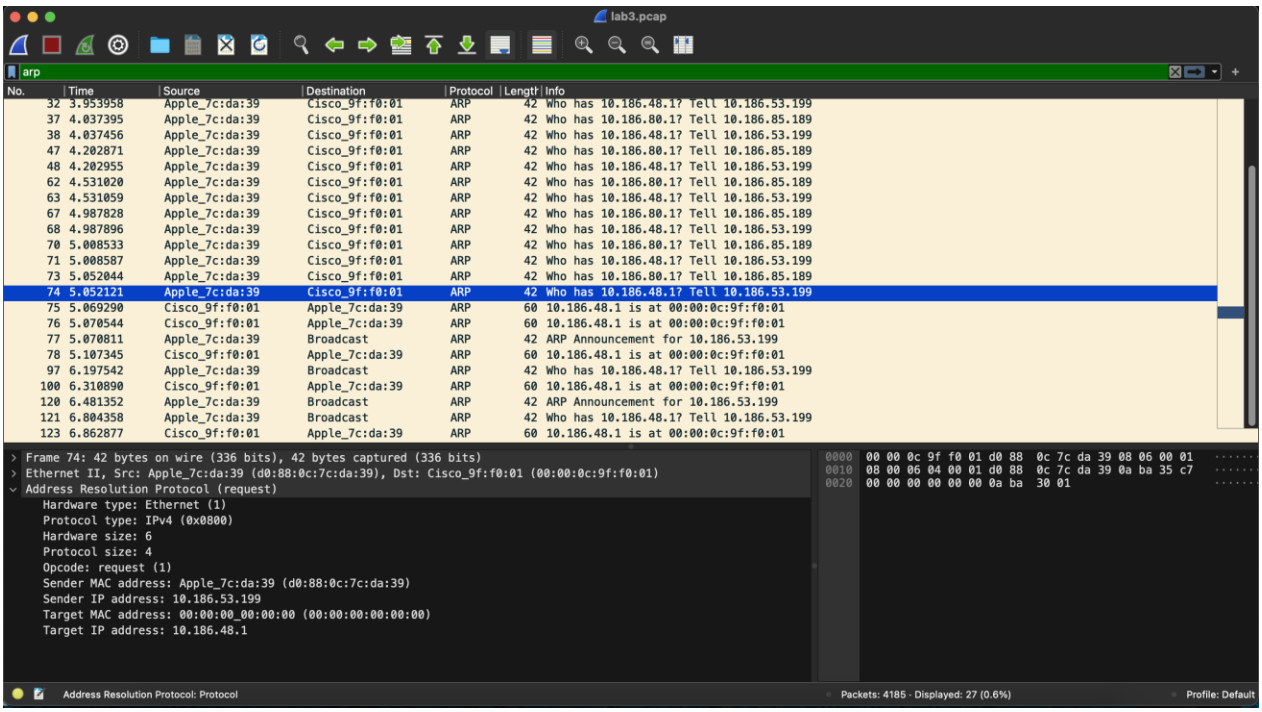
a. In the trace captured by you, how did it obtain the IP and MAC address of a gateway router? Which address was obtained first? Please show the involved packets or screenshot of these involved packets.

Sol: The **IP Address of gateway router was obtained first, which is 10.186.48.1** in this case via DHCP protocol. Next, after getting the gateway router's IP address, to get its MAC address, ARP request packet is broadcasted asking for its MAC address. The gateway router receives the packet and responses with its MAC address. Therefore, **MAC address of gateway router is 00:00:0c:9f:f0:01**

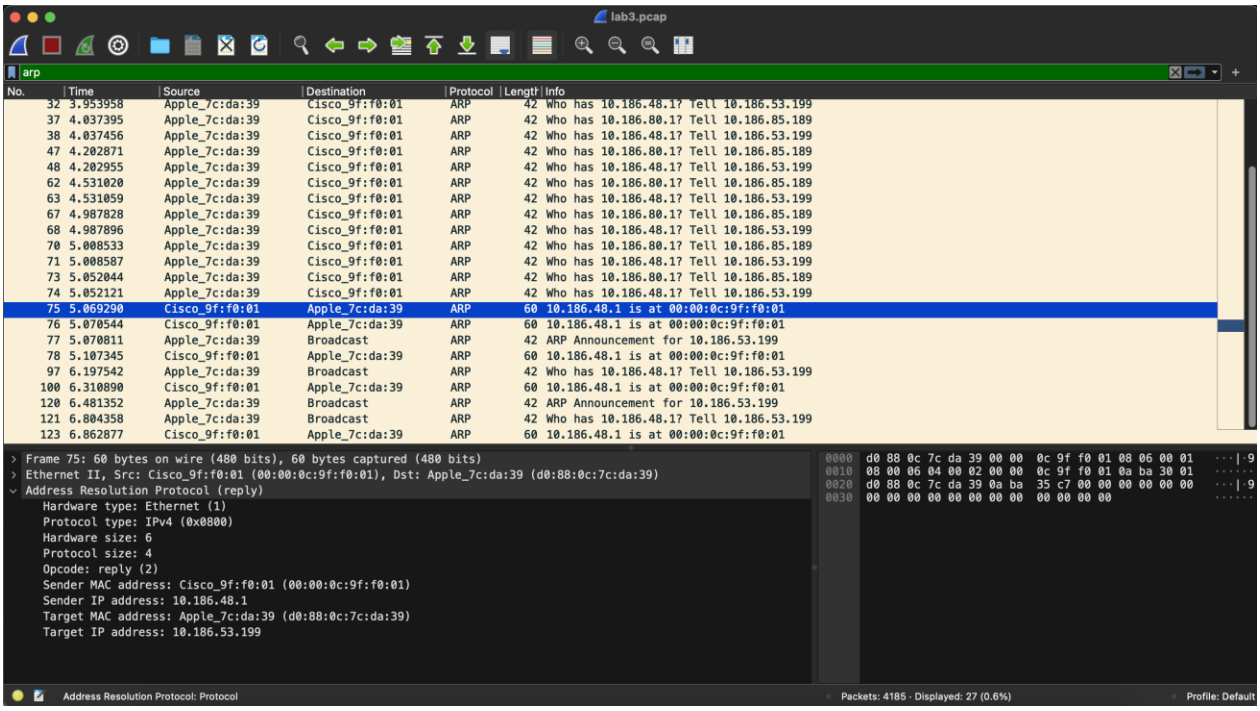
Screenshot of the involved packets is:



ARP Request:



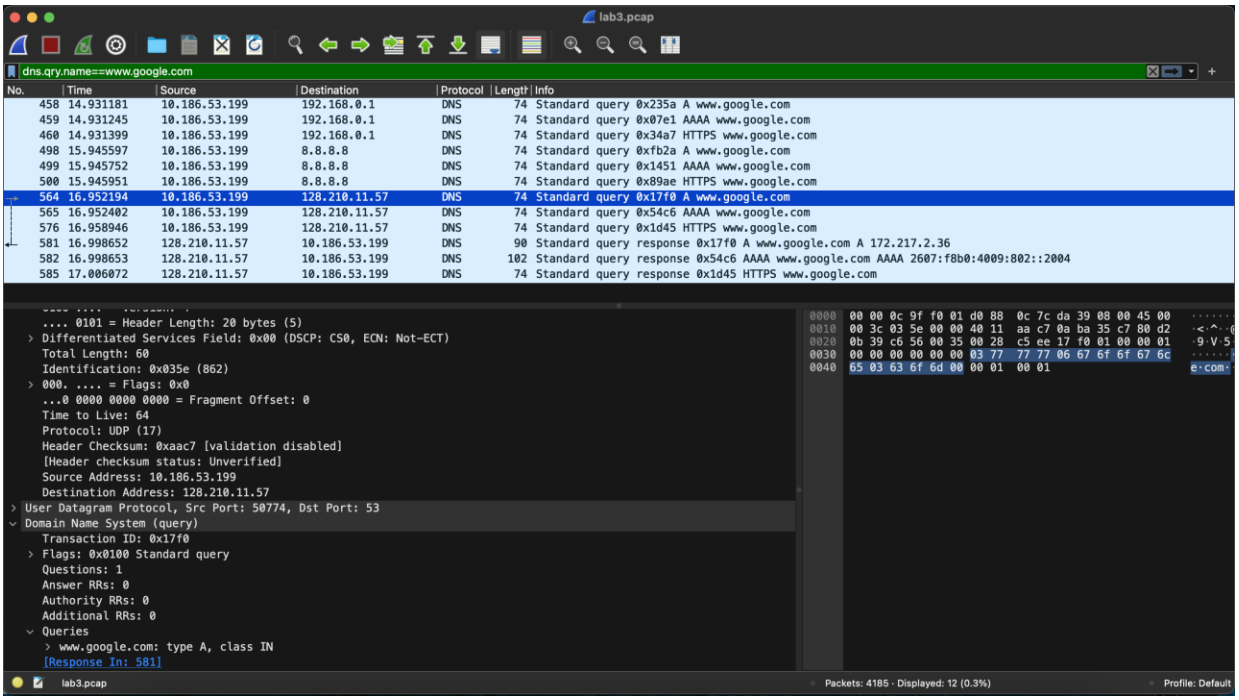
ARP Response:



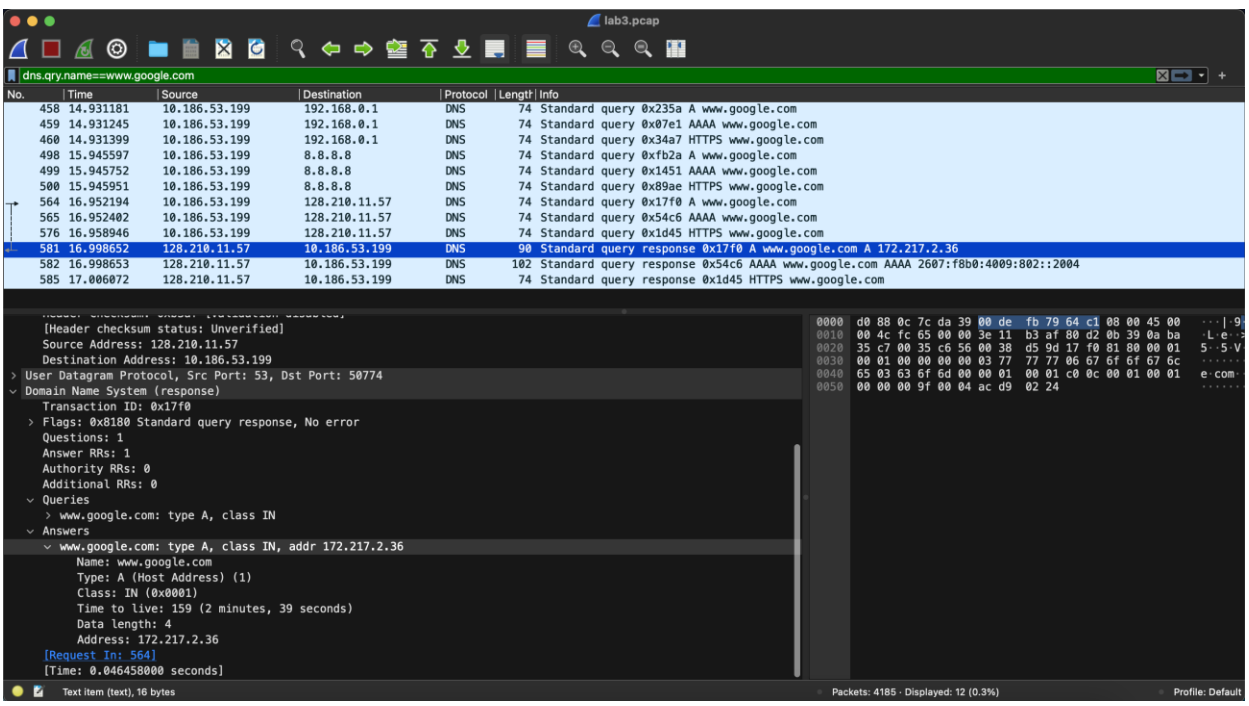
b. In the trace captured by you, what was the IP address used by google.com? How did you get it? Please show the involved packets or screenshot of these involved packets.

Sol: In the captured trace, the IP address used by **google.com is 172.217.2.36** . We got it using DNS resolution request and response packets.

1. Filtered the captured packets by “dns”
2. Looked for DNS query packet. The packet will have DNS layer with Query Type set to ‘A’ (for IPV4 addresses) and Query Name set to “google.com”



3. Looked for DNS response packet. The packet will have DNS layer with Response section. The IP address will be listed as “Answer” to the DNS query.



c. In the trace captured by you, can you see any packets relevant to intra-AS or inter-AS routing? If yes, please show the involved packets or screenshot of these involved packets. If no, please explain why.

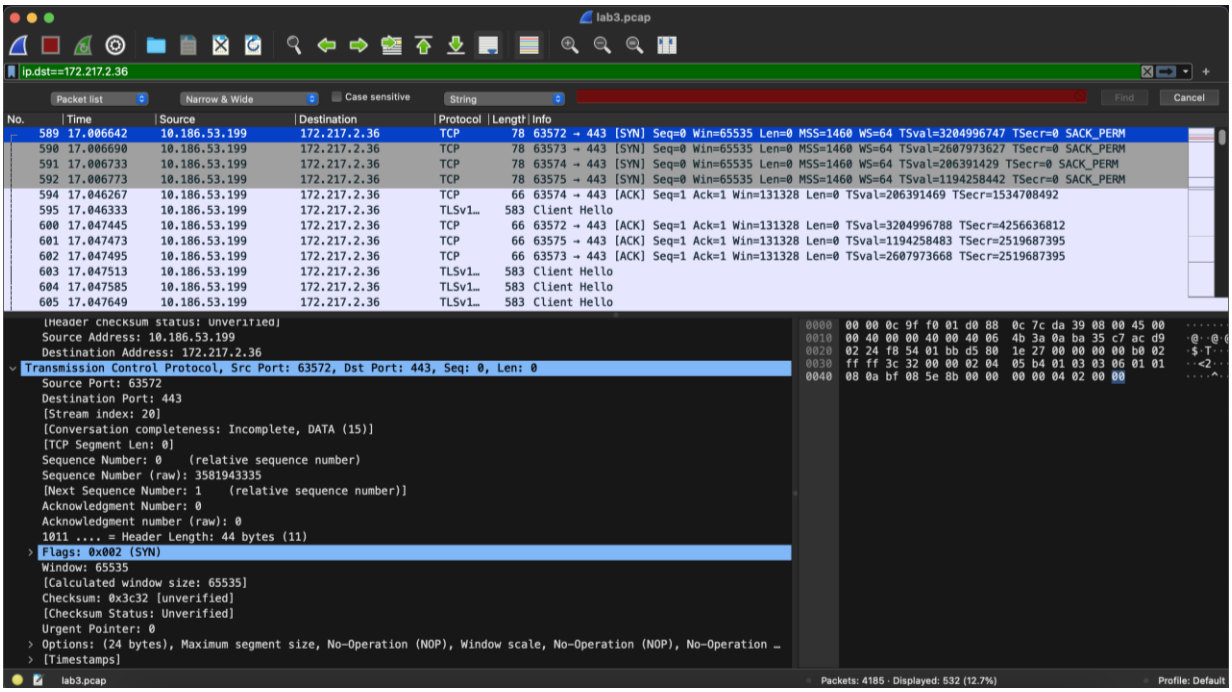
Sol: In the captured trace, **we cannot see the packets relevant to intra-AS or inter-AS routing.** This is because, Exterior routing protocols such as Border Gateway Protocol (BGP), RIP etc. are used by routers to exchange information about the best routes to reach other networks or destinations outside their own network. However, if the routers in the network already have the necessary routing information to reach a specific destination, such as www.google.com, there will be no need for exterior routing protocol packets to be exchanged.

d. In the trace captured by you, please find the packets for TCP three-way handshaking. Please screenshot of these involved packets.

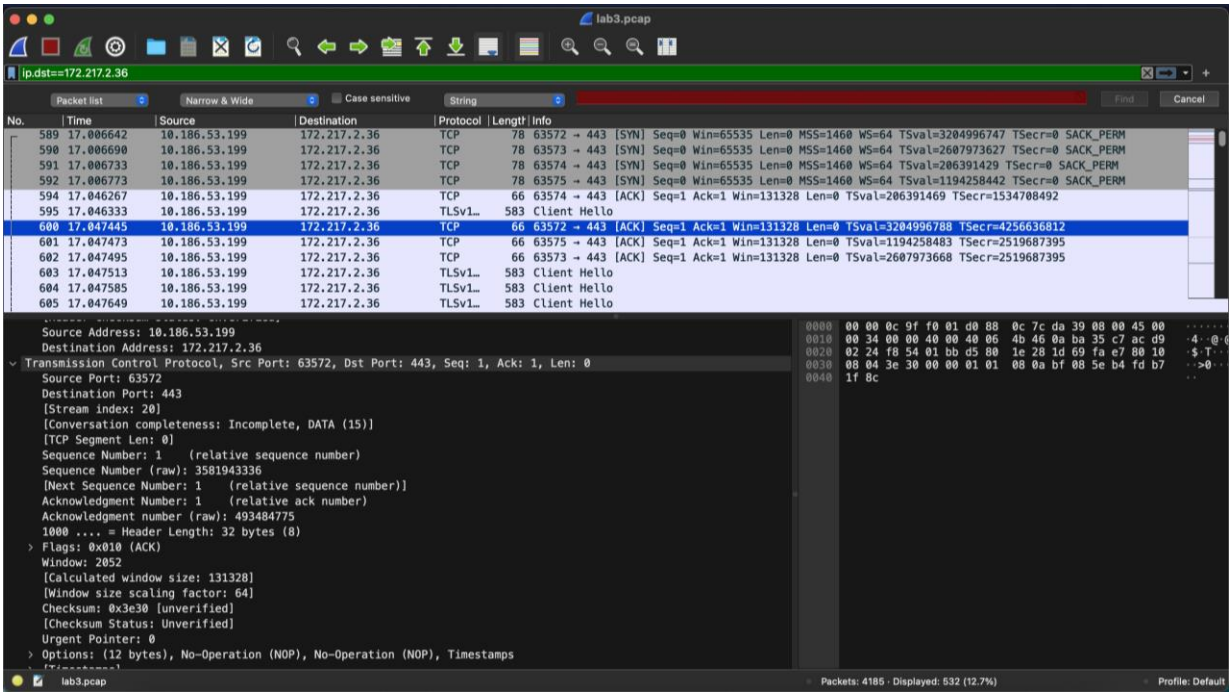
Sol: Following are the screenshot of the involved packets:

Example 1:

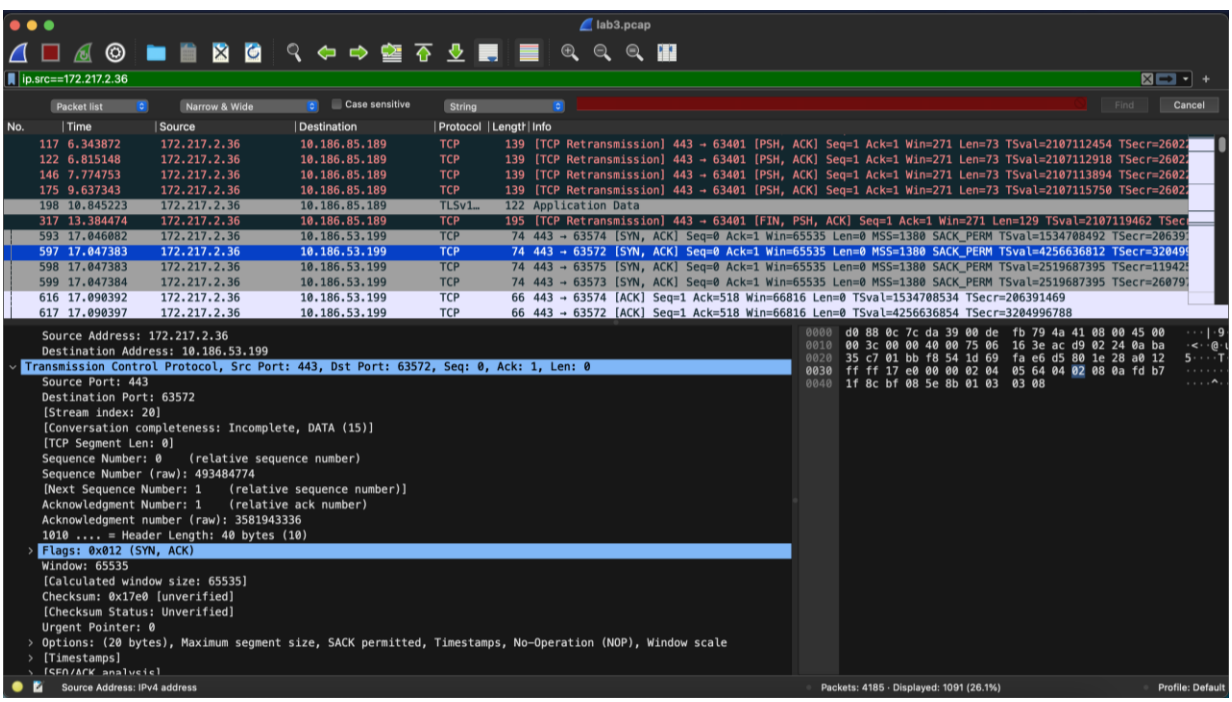
SYN Packet



SYN-ACK Packet


lab3.pcap
ip.dst==172.217.2.36
Packet list
Narrow & Wide
Case sensitive
String
Find
Cancel
No. Time Source Destination Protocol Length Info
589 17.006642 10.186.53.199 172.217.2.36 TCP 78 63572 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=3204996747 TSecr=0 SACK_PERM
590 17.006690 10.186.53.199 172.217.2.36 TCP 78 63573 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2607973627 TSecr=0 SACK_PERM
591 17.006733 10.186.53.199 172.217.2.36 TCP 78 63574 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=206391429 TSecr=0 SACK_PERM
592 17.006773 10.186.53.199 172.217.2.36 TCP 78 63575 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=1194258442 TSecr=0 SACK_PERM
594 17.046267 10.186.53.199 172.217.2.36 TCP 66 63574 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=206391469 TSecr=1534708492
595 17.046333 10.186.53.199 172.217.2.36 TLSv1… 583 Client Hello
600 17.047445 10.186.53.199 172.217.2.36 TCP 66 63572 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=3204996788 TSecr=4256636812
601 17.047473 10.186.53.199 172.217.2.36 TCP 66 63575 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=1194258483 TSecr=2519687395
602 17.047495 10.186.53.199 172.217.2.36 TCP 66 63573 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=2607973668 TSecr=2519687395
603 17.047513 10.186.53.199 172.217.2.36 TLSv1… 583 Client Hello
604 17.047585 10.186.53.199 172.217.2.36 TLSv1… 583 Client Hello
605 17.047649 10.186.53.199 172.217.2.36 TLSv1… 583 Client Hello
Source Address: 10.186.53.199
Destination Address: 172.217.2.36
Transmission Control Protocol, Src Port: 63572, Dst Port: 443, Seq: 1, Ack: 1, Len: 0
Source Port: 63572
Destination Port: 443
[Stream index: 20]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 1 (relative sequence number)
Sequence Number (raw): 3581943336
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 493484775
1000 .... = Header Length: 32 bytes (8)
Flags: 0x010 (ACK)
Window: 2052
[Calculated window size: 131328]
[Window size scaling factor: 64]
Checksum: 0x3e30 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
0000 00 00 0c 9f f0 01 d0 88 0c 7c da 39 08 00 45 00
0010 00 34 00 00 40 00 40 06 4b 46 0a ba 35 c7 ac d9
0020 02 24 f8 54 01 bb d5 80 1e 28 1d 69 fa e7 80 10
0030 08 04 3e 30 00 00 01 01 08 0a bf 08 5e b4 fd b7
0040 1f 8c
lab3.pcap
Packets: 4185 · Displayed: 532 (12.7%)
Profile: Default

ACK Packet


lab3.pcap
ip.src==172.217.2.36
Packet list
Narrow & Wide
Case sensitive
String
Find
Cancel
No. Time Source Destination Protocol Length Info
117 6.343872 172.217.2.36 10.186.85.189 TCP 139 [TCP Retransmission] 443 → 63401 [PSH, ACK] Seq=1 Ack=1 Win=271 Len=73 TSval=2107112454 TSecr=2602
122 6.815148 172.217.2.36 10.186.85.189 TCP 139 [TCP Retransmission] 443 → 63401 [PSH, ACK] Seq=1 Ack=1 Win=271 Len=73 TSval=2107112918 TSecr=2602
146 7.774753 172.217.2.36 10.186.85.189 TCP 139 [TCP Retransmission] 443 → 63401 [PSH, ACK] Seq=1 Ack=1 Win=271 Len=73 TSval=2107113894 TSecr=2602
175 9.637343 172.217.2.36 10.186.85.189 TCP 139 [TCP Retransmission] 443 → 63401 [PSH, ACK] Seq=1 Ack=1 Win=271 Len=73 TSval=2107115750 TSecr=2602
198 10.845223 172.217.2.36 10.186.85.189 TLSv1… 122 Application Data
317 13.384474 172.217.2.36 10.186.85.189 TCP 195 [TCP Retransmission] 443 → 63401 [FIN, PSH, ACK] Seq=1 Ack=1 Win=271 Len=129 TSval=2107119462 TSec
593 17.046082 172.217.2.36 10.186.53.199 TCP 74 443 → 63574 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=1534708492 TSecr=20639
597 17.047383 172.217.2.36 10.186.53.199 TCP 74 443 → 63572 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=4256636812 TSecr=32049
598 17.047383 172.217.2.36 10.186.53.199 TCP 74 443 → 63575 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=2519687395 TSecr=11942
599 17.047384 172.217.2.36 10.186.53.199 TCP 74 443 → 63573 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1380 SACK_PERM TSval=2519687395 TSecr=26079
616 17.090392 172.217.2.36 10.186.53.199 TCP 66 443 → 63574 [ACK] Seq=1 Ack=518 Win=66816 Len=0 TSval=1534708534 TSecr=206391469
617 17.090397 172.217.2.36 10.186.53.199 TCP 66 443 → 63572 [ACK] Seq=1 Ack=518 Win=66816 Len=0 TSval=4256636854 TSecr=3204996788
Source Address: 172.217.2.36
Destination Address: 10.186.53.199
Transmission Control Protocol, Src Port: 443, Dst Port: 63572, Seq: 0, Ack: 1, Len: 0
Source Port: 443
Destination Port: 63572
[Stream index: 20]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 0]
Sequence Number: 0 (relative sequence number)
Sequence Number (raw): 493484774
[Next Sequence Number: 1 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 3581943336
1010 .... = Header Length: 40 bytes (10)
Flags: 0x012 (SYN, ACK)
Window: 65535
[Calculated window size: 65535]
Checksum: 0x17e0 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
[Timestamps]
0000 d0 88 0c 7c da 39 00 de fb 79 4a 41 08 00 45 00
0010 00 3c 00 00 40 00 75 06 16 3e ac d9 02 24 0a ba
0020 35 c7 01 bb f8 54 1d 69 fa e6 d5 80 1e 28 a0 12
0030 ff ff 17 e0 00 00 02 04 05 64 04 02 08 0a fd b7
0040 1f 8c bf 08 5e 8b 01 03 03 08
Source Address: IPv4 address
Packets: 4185 · Displayed: 1091 (26.1%)
Profile: Default

e. In the trace captured by you, can you see any packets for HTTP GET messages and HTTP response messages? If yes, please show the first HTTP GET message (to www.google.com) and its HTTP response message or screenshot of them. If no, please explain why.

Sol: In the captured trace, we cannot see any packets for HTTP GET messages and HTTP response messages.

This is because the application data is TLS-encrypted, therefore, Wireshark cannot see it. Because of this, TLS and TLS version are used in the protocol column by Wireshark instead of HTTPS.

Part C: Please write your own code to analyze the captured pcap file and extract the packets needed in Part B. Please feel free to use C or Python3. Please run packet analysis in the following cases:

• CASE A: Extract the IP address and MAC address of a gateway router.

• CASE B: Extract the IP address of the destination website (here, www.google.com).

• CASE C: Extract the packets for TCP three-way handshaking for the TCP connection between your computer and the destination website (here, www.google.com).

Sol:

Part A:

python lab3.py A lab3.pcap www.google.com

```
(base) mansishinde@Mansis-MacBook-Air Lab3_34784153 % python lab3.py A lab3.pcap www.google.com
CASE A:
IPAddr:  10.186.48.1
MACAddr:  00:00:0c:9f:f0:01
(base) mansishinde@Mansis-MacBook-Air Lab3_34784153 %
```

Part B:

python lab3.py B lab3.pcap www.google.com

```
(base) mansishinde@Mansis-MacBook-Air Lab3_34784153 % python lab3.py B lab3.pcap www.google.com
CASE B:
IPAddr-DEST:  172.217.2.36
(base) mansishinde@Mansis-MacBook-Air Lab3_34784153 %
```

Part C:

python lab3.py C lab3.pcap www.google.com

```
(base) mansishinde@pal-nat186-60-70 Lab3_34784153 % python lab3.py C lab3.pcap www.google.com
CASE C:
IPAddr-SRC:  10.186.53.199
IPAddr-DEST:  172.217.2.36
Port-SRC:  443
SYN:  1
ACK:  0
IPAddr-SRC:  172.217.2.36
IPAddr-DEST:  10.186.53.199
Port-SRC:  443
SYN:  1
ACK:  1
IPAddr-SRC:  10.186.53.199
IPAddr-DEST:  172.217.2.36
Port-SRC:  443
SYN:  0
ACK:  1
(base) mansishinde@pal-nat186-60-70 Lab3_34784153 %
```

All:

python lab3.py ALL lab3.pcap www.google.com

```
(base) mansishinde@pal-nat186-60-70 Lab3_34784153 % python lab3.py ALL lab3.pcap www.google.com
CASE A:
IPAddr:  10.186.48.1
MACAddr:  00:00:0c:9f:f0:01
CASE B:
IPAddr-DEST:  172.217.2.36
CASE C:
IPAddr-SRC:  10.186.53.199
IPAddr-DEST:  172.217.2.36
Port-SRC:  443
SYN:  1
ACK:  0
IPAddr-SRC:  172.217.2.36
IPAddr-DEST:  10.186.53.199
Port-SRC:  443
SYN:  1
ACK:  1
IPAddr-SRC:  10.186.53.199
IPAddr-DEST:  172.217.2.36
Port-SRC:  443
SYN:  0
ACK:  1
(base) mansishinde@pal-nat186-60-70 Lab3_34784153 %
```