Student Information

Full Name: Doruk Gerçel
Id Number: 2310027

HTTP & DNS (70 Points)

I used Mozilla Firefox Private Browser as my web browser.

1. (8 Points)

From Figure 1 we can observe that 48 DNS queries were sent from my computer to the DNS Server. When we look at the right bottom corner of the figure, we can observe that there are 48 packets (DNS queries) on display when I apply my filter. 22 of these queries were type AAAA (IPv6) queries and 26 of these queries were type A (IPv4) queries. After my research I learned that, old DNS resolvers cannot handle AAAA type queries and returns error. Therefore to overcome this problem the hosts send two types of queries which are A and AAAA.

2. (10 Points)

192.168.1.1 is the address where the DNS queries were sent and answered. I found this address by looking at the destination address of the queries from Wireshark.

2. (Bonus) (10 Bonus Points)

The destination address where my queries are sent is actually the modem that I use in my home. I made this interpretation by observing the resemblance between the source and destination addresses. Source address is 192.168.1.26 and destination address is 192.168.1.1. Therefore I realized that as source address is my own computer, the destination address must be home modem as the home modem assigned my computer's IP address and assigned the first address that it can assign (192.168.1.1) to itself. My computer didn't use its own local DNS cache and directly asked to the home modem directly.

3. (15 Points)

As my computer received DNS response, I learned that the IP address of "http://ceng.metu.edu.tr" is 144.122.145.146. Therefore by checking the

request which is sent to this address and the response which is received from this address I can observe the first request-response pair. From Figure 2 we can observe that No. 51 is the first packet which is sent to the "http://ceng.metu.edu.tr" web server as a request and No. 56 is the first packet which is received as a response from the web server. They are a TCP request and response pair. They are not a HTTP pair because as we know HTTP is an application layer protocol and TCP is a transport layer protocol, and application layer uses the services of the transport layer. Therefore in-order to exchange application layer messages, first they need to initialize a transport layer connection. So this is why the first request and response pair was a TCP pair but no a HTTP pair.

4. (15 Points)

In the first HTTP request to the web server, a web cookie is sent. First, by applying the necessary filter (filter applied can be seen in the figure), I viewed the first HTTP request in Figure 3. Cookie can be find in the HTTP message part of the request and the cookie is highlighted in the Figure 4.

5a. (7 Points)

User-agent string is a string which is a part of HTTP protocol. This string is necessary for web server to identify some properties of the host which sent request to the web server. The name of the browser which sent the request, version of the browser and the operating system that the browser runs on are all included in the user-agent string. The user-agent string that my computer sent is "Mozilla/5.0 (X11; Ubuntu; Linux x86-64; rv:81.0) Gecko/20100101 Firefox/81.0"(Figure 5).

5b. (15 Points)

The user-agent string includes Firefox/81.0 which is the browser that I'm using and the version of this browser. No other browser is mentioned. Although my user-agent includes only a single browser, the other user-agents from other browsers include several browsers. For example all user-agents from different browsers include Mozilla in-order to inform the web servers that they are using a newer technology (actually newer than Mosaic browser) and that they can support frames. Therefore web servers send web pages with frames to these browsers.

HTTPS & TLS (30 Points)

1. (10 Points)

First I learned the IP address of the "https://odtuclass.metu.edu.tr" by checking the DNS response and learned that it is 144.122.145.167. Therefore I made the necessary filters (filter applied can be seen in the figure) and observed that the first request response pair is a TCP pair in the Figure 6. The time difference between the response and request query is 0.02 seconds. I set the request as a time reference to easily observe the time difference.

2. (10 Points)

In the info field of the first TLS request "Client Hello" is written. Also in the info field of the first TLS response "Server Hello, Change Cipher Spec, Application Data" is written (Figure 7). As both of them said "Hello" to each other I can conclude that they are trying to initialize a TLS connection and they are performing a handshake.

3. (10 Points)

There are 12 "Hello" messages sent from client and server (Figure 8). 6 of them are sent from the client and 6 of them are sent from the server. There are more than one pair of "Hello" messages, because the browser connects with the web server from different ports of the server, by this way the web server can handle multiple requests from the client in parallel. Therefore in-order to initialize connection between these different ports, different handshakes ("Hello" pairs) occur.

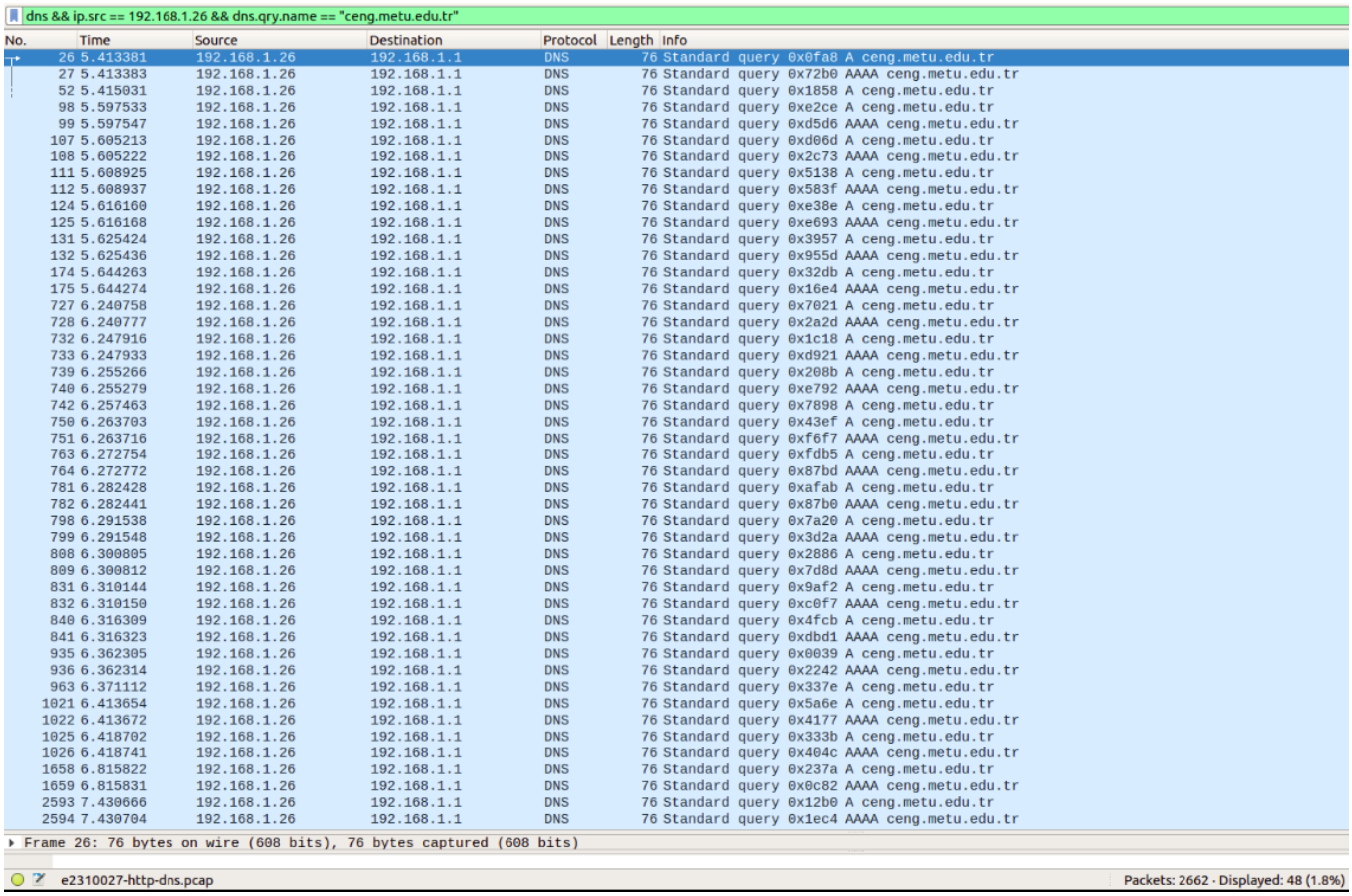

Figure 1: DNS Queries which Were Sent from My Computer

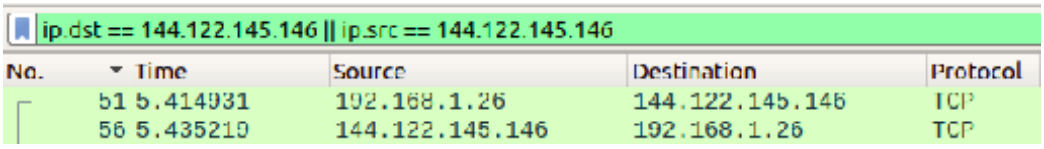

Figure 2: First Request Response Pair for ceng.metu.edu.tr

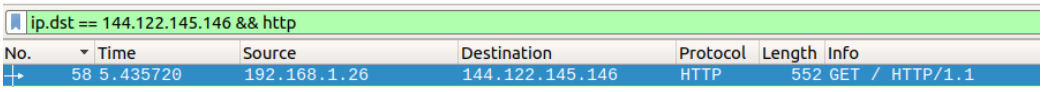

Figure 3: First HTTP Request to the Web Server

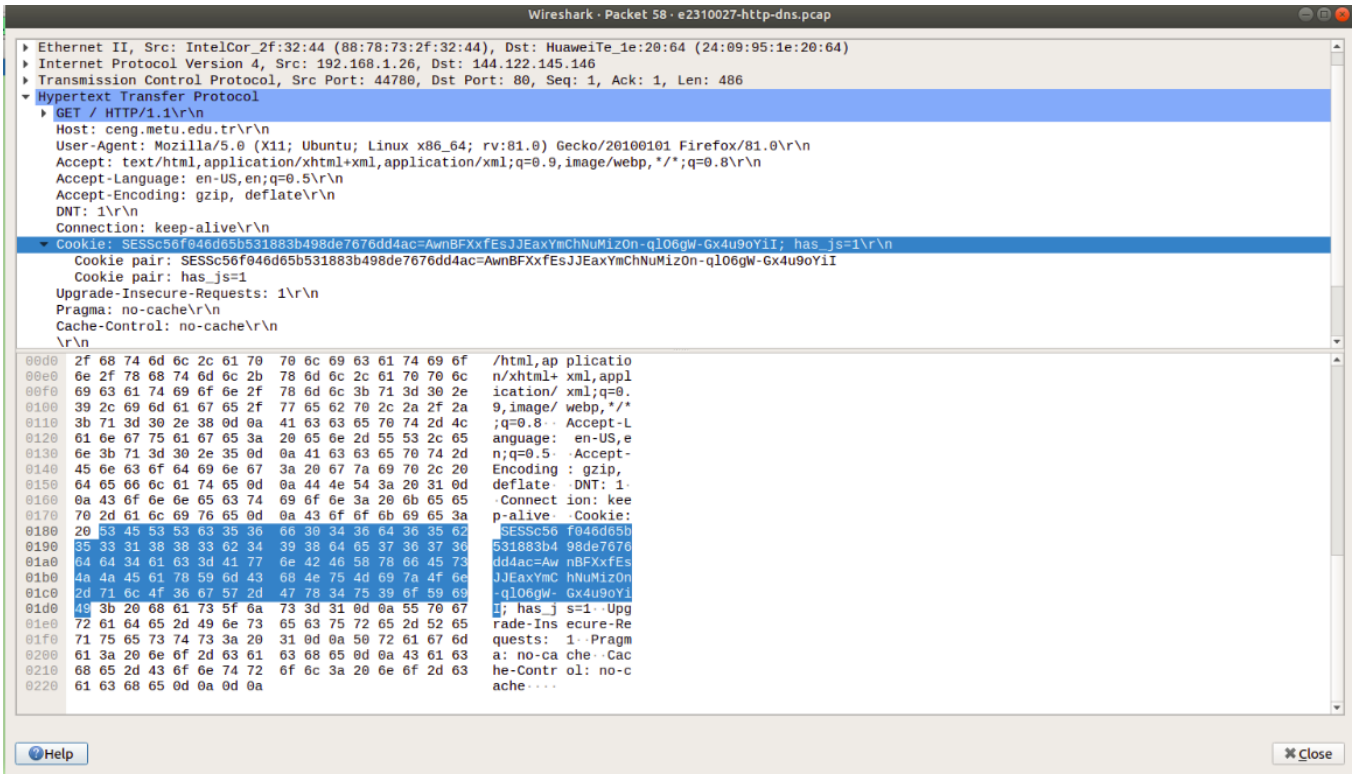

Figure 4: HTTP Request (Cookie is Highlighted)

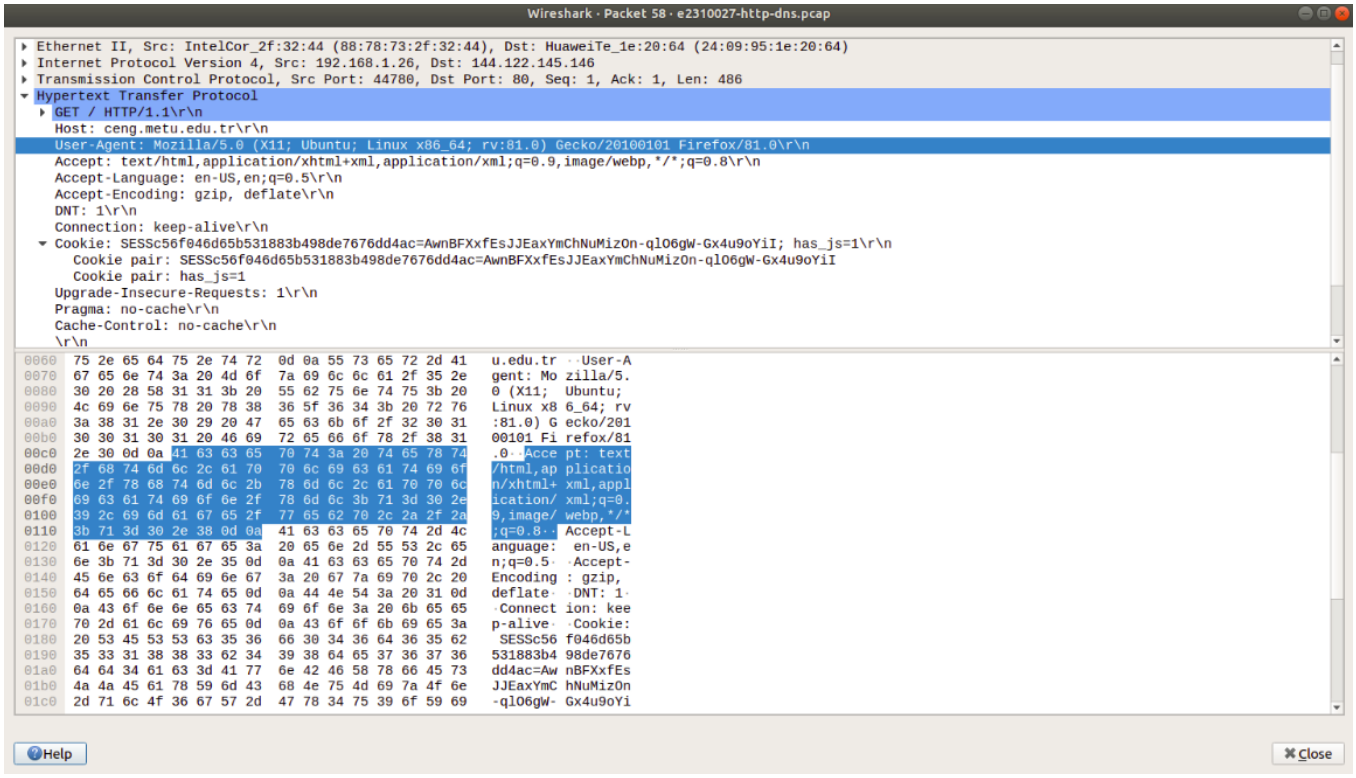

Figure 5: HTTP Request (User-Agent is Highlighted)

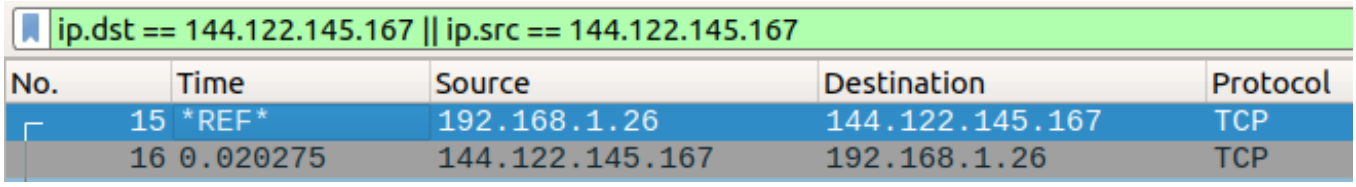

Figure 6: First Request Response Pair for odtuclass.metu.edu.tr

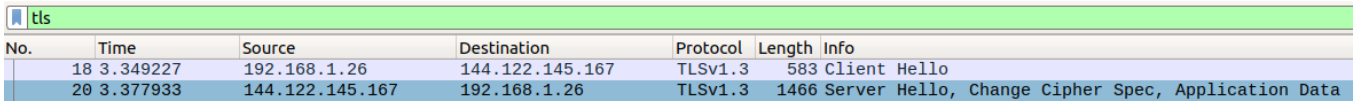

Figure 7: First TLS Request Response Pair for odtuclass.metu.edu.tr

tls.handshake && (ip.src == 144.122.145.167 || ip.dst == 144.122.145.167)

No.	Time	Source	Destination	Protocol	Length	Info
18	3.349227	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
20	3.377933	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data
89	3.672133	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
98	3.683672	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
101	3.689737	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
104	3.692908	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data
108	3.695686	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
116	3.704203	192.168.1.26	144.122.145.167	TLSv1.3	583	Client Hello
117	3.706092	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data
132	3.712172	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data
162	3.716776	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data
212	3.732686	144.122.145.167	192.168.1.26	TLSv1.3	1466	Server Hello, Change Cipher Spec, Application Data

Figure 8: TLS "Hello" Messages