Report: Analyzing Web Traffic with Wireshark

Introduction

This report documents the process of capturing and analyzing web traffic using Wireshark, a network protocol analyzer. The primary objective of this exercise was to capture network traffic while visiting specific websites and then filter and analyze the captured data to list only HTTP and HTTPS packets while excluding packets related to the "cygwin.com" website.

Methodology

1. Preparations

Before initiating the packet capture process, the following preparations were made:

The cache in the Firefox browser was cleared to ensure that the captured packets would represent fresh requests and responses.

Wireshark was opened and configured to capture packets on the Ethernet interface.

2. Packet Capture

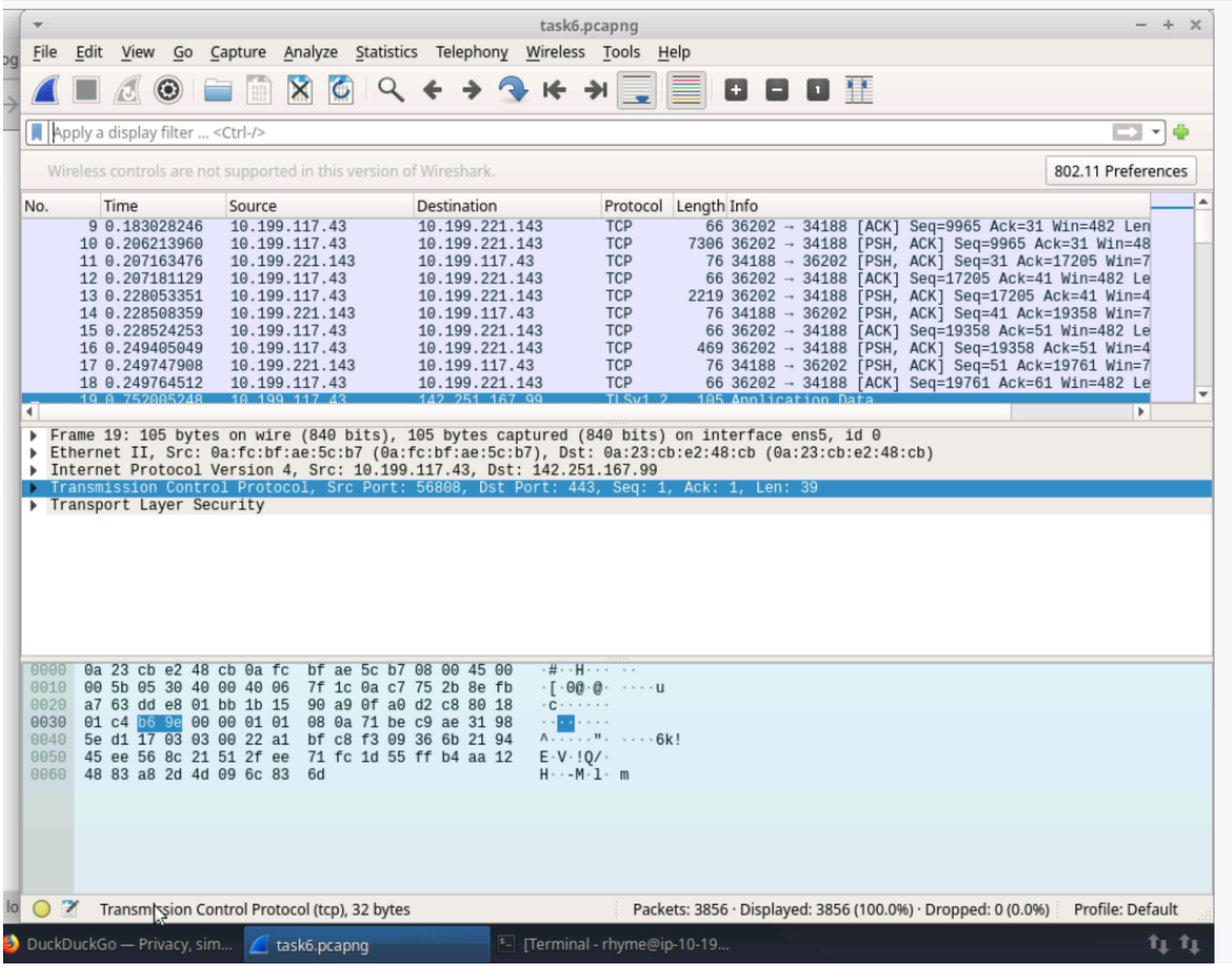
The packet capture process involved visiting three different websites:

a. Google.com
b. Duckduckgo.com
c. http://cygwin.com

While visiting these websites, Wireshark was actively capturing network traffic on the Ethernet interface.

3. Stopping and Saving Capture

After visiting the specified websites and capturing network traffic, the packet capture process was stopped in Wireshark, and the capture file was saved for further analysis.


task6.pcapng
File
Edit
View
Go
Capture
Analyze
Statistics
Telephony
Wireless
Tools
Help
Apply a display filter ... <Ctrl-/>
Wireless controls are not supported in this version of Wireshark.
802.11 Preferences
DuckDuckGo — Privacy, sim...
task6.pcapng
[Terminal - rhyme@ip-10-19...

```
No.   Time          Source           Destination      Protocol  Length Info
    9 0.183028246   10.199.117.43    10.199.221.143   TCP         66 36202 → 34188 [ACK] Seq=9965 Ack=31 Win=482 Len
   10 0.206213960   10.199.117.43    10.199.221.143   TCP       7306 36202 → 34188 [PSH, ACK] Seq=9965 Ack=31 Win=48
   11 0.207163476   10.199.221.143   10.199.117.43    TCP         76 34188 → 36202 [PSH, ACK] Seq=31 Ack=17205 Win=7
   12 0.207181129   10.199.117.43    10.199.221.143   TCP         66 36202 → 34188 [ACK] Seq=17205 Ack=41 Win=482 Le
   13 0.228053351   10.199.117.43    10.199.221.143   TCP       2219 36202 → 34188 [PSH, ACK] Seq=17205 Ack=41 Win=4
   14 0.228508359   10.199.221.143   10.199.117.43    TCP         76 34188 → 36202 [PSH, ACK] Seq=41 Ack=19358 Win=7
   15 0.228524253   10.199.117.43    10.199.221.143   TCP         66 36202 → 34188 [ACK] Seq=19358 Ack=51 Win=482 Le
   16 0.249405049   10.199.117.43    10.199.221.143   TCP        469 36202 → 34188 [PSH, ACK] Seq=19358 Ack=51 Win=4
   17 0.249747908   10.199.221.143   10.199.117.43    TCP         76 34188 → 36202 [PSH, ACK] Seq=51 Ack=19761 Win=7
   18 0.249764512   10.199.117.43    10.199.221.143   TCP         66 36202 → 34188 [ACK] Seq=19761 Ack=61 Win=482 Le
   19 0.752005248   10.199.117.43    142.251.167.99   TLSv1.2     105 Application Data
```

```
▸ Frame 19: 105 bytes on wire (840 bits), 105 bytes captured (840 bits) on interface ens5, id 0
▸ Ethernet II, Src: 0a:fc:bf:ae:5c:b7 (0a:fc:bf:ae:5c:b7), Dst: 0a:23:cb:e2:48:cb (0a:23:cb:e2:48:cb)
▸ Internet Protocol Version 4, Src: 10.199.117.43, Dst: 142.251.167.99
▸ Transmission Control Protocol, Src Port: 56808, Dst Port: 443, Seq: 1, Ack: 1, Len: 39
▸ Transport Layer Security
```

```
0000  0a 23 cb e2 48 cb 0a fc  bf ae 5c b7 08 00 45 00   ·#··H··· ··
0010  00 5b 05 30 40 00 40 06  7f 1c 0a c7 75 2b 8e fb   ·[·0@·@· ····u
0020  a7 63 dd e8 01 bb 1b 15  90 a9 0f a0 d2 c8 80 18   ·c······
0030  01 c4 b6 9e 00 00 01 01  08 0a 71 be c9 ae 31 98   ········
0040  5e d1 17 03 03 00 22 a1  bf c8 f3 09 36 6b 21 94   ^·····"· ····6k!
0050  45 ee 56 8c 21 51 2f ee  71 fc 1d 55 ff b4 aa 12   E·V·!Q/·
0060  48 83 a8 2d 4d 09 6c 83  6d                        H··-M·l· m
```

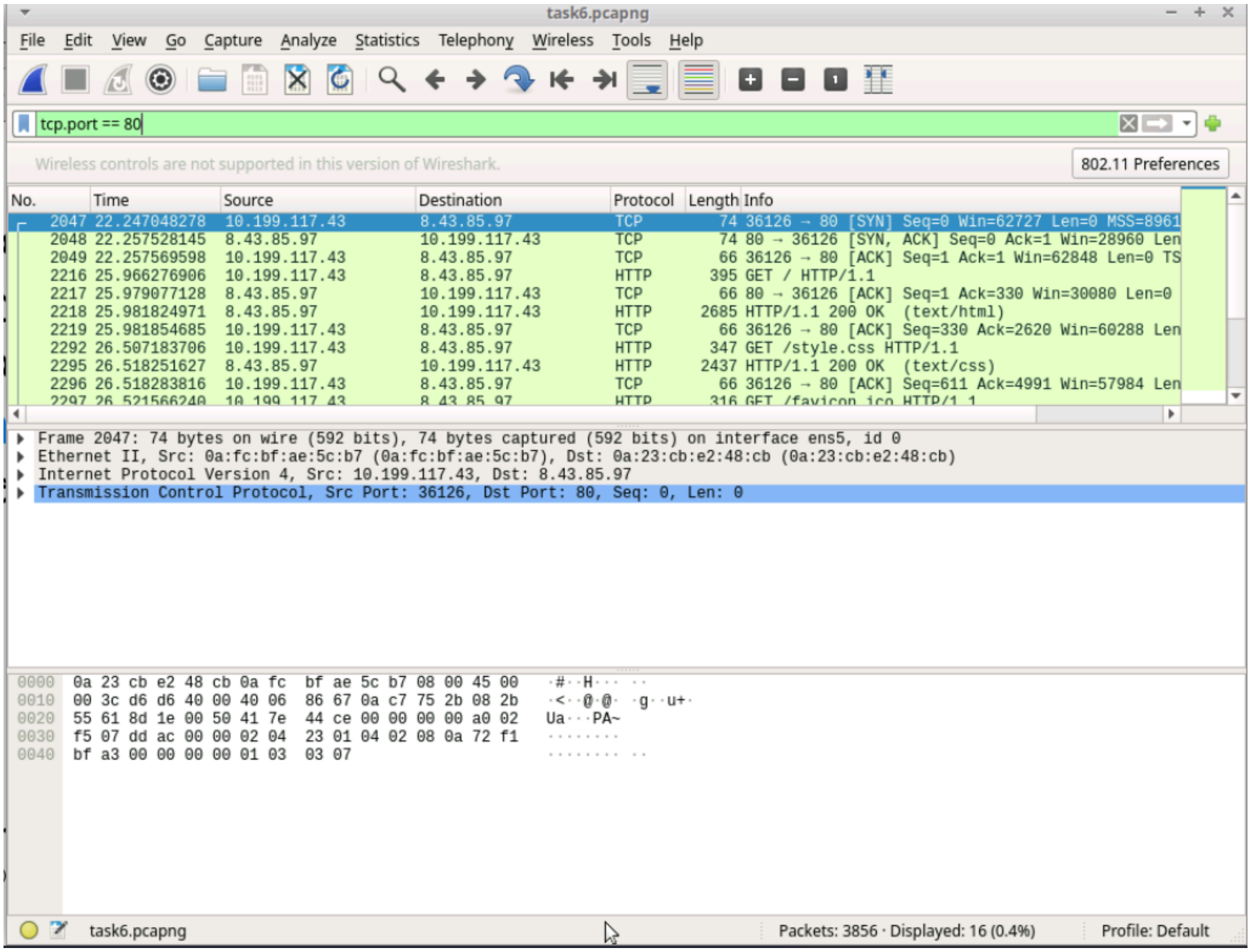
```
Transmission Control Protocol (tcp), 32 bytes    Packets: 3856 · Displayed: 3856 (100.0%) · Dropped: 0 (0.0%)    Profile: Default
```

Analysis

The analysis of the captured network traffic was carried out in several steps as follows:

1. Filtering Port 80 TCP Data

A filter was applied to the captured packets to display only those related to port 80, which is commonly associated with HTTP traffic. This step was performed to isolate HTTP packets from other types of network traffic. All traffic from "cygwin.com" will come from port 80 in this report.



tcp.port == 80

```
No.    Time          Source          Destination     Protocol Length Info
  2047 22.247048278  10.199.117.43   8.43.85.97      TCP        74 36126 → 80 [SYN] Seq=0 Win=62727 Len=0 MSS=8961
  2048 22.257528145  8.43.85.97      10.199.117.43   TCP        74 80 → 36126 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len
  2049 22.257569598  10.199.117.43   8.43.85.97      TCP        66 36126 → 80 [ACK] Seq=1 Ack=1 Win=62848 Len=0 TS
  2216 25.966276906  10.199.117.43   8.43.85.97      HTTP      395 GET / HTTP/1.1
  2217 25.979077128  8.43.85.97      10.199.117.43   TCP        66 80 → 36126 [ACK] Seq=1 Ack=330 Win=30080 Len=0
  2218 25.981824971  8.43.85.97      10.199.117.43   HTTP     2685 HTTP/1.1 200 OK  (text/html)
  2219 25.981854685  10.199.117.43   8.43.85.97      TCP        66 36126 → 80 [ACK] Seq=330 Ack=2620 Win=60288 Len
  2292 26.507183706  10.199.117.43   8.43.85.97      HTTP      347 GET /style.css HTTP/1.1
  2295 26.518251627  8.43.85.97      10.199.117.43   HTTP     2437 HTTP/1.1 200 OK  (text/css)
  2296 26.518283816  10.199.117.43   8.43.85.97      TCP        66 36126 → 80 [ACK] Seq=611 Ack=4991 Win=57984 Len
  2297 26.521566240  10.199.117.43   8.43.85.97      HTTP      316 GET /favicon.ico HTTP/1.1
```

```
▸ Frame 2047: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface ens5, id 0
▸ Ethernet II, Src: 0a:fc:bf:ae:5c:b7 (0a:fc:bf:ae:5c:b7), Dst: 0a:23:cb:e2:48:cb (0a:23:cb:e2:48:cb)
▸ Internet Protocol Version 4, Src: 10.199.117.43, Dst: 8.43.85.97
▸ Transmission Control Protocol, Src Port: 36126, Dst Port: 80, Seq: 0, Len: 0
```

```
0000  0a 23 cb e2 48 cb 0a fc  bf ae 5c b7 08 00 45 00   ·#··H···  ··
0010  00 3c d6 d6 40 00 40 06  86 67 0a c7 75 2b 08 2b   ·<··@·@·  ·g··u+·
0020  55 61 8d 1e 00 50 41 7e  44 ce 00 00 00 00 a0 02   Ua···PA~
0030  f5 07 dd ac 00 00 02 04  23 01 04 02 08 0a 72 f1   ········
0040  bf a3 00 00 00 00 01 03  03 07                     ········  ··
```

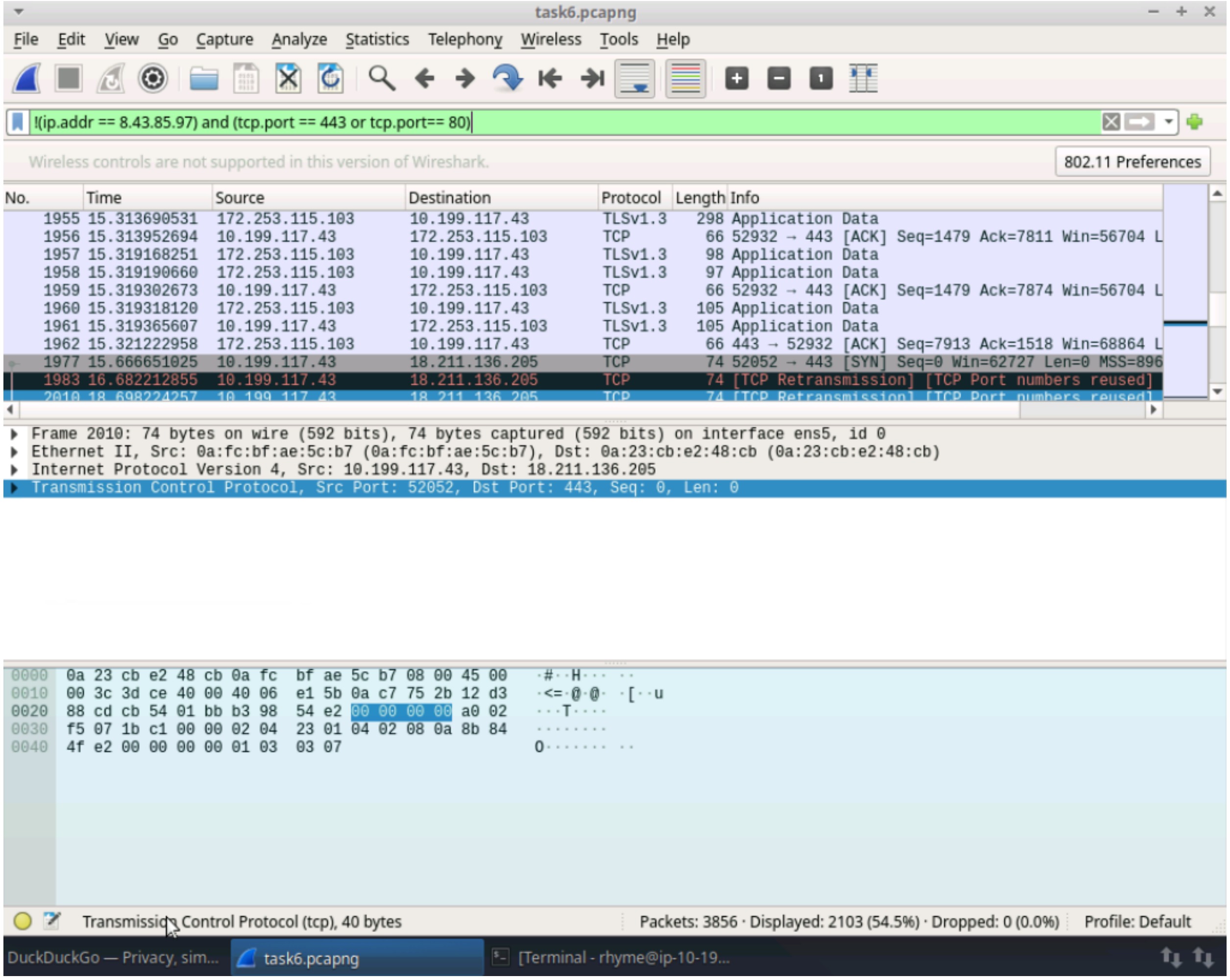
task6.pcapng   Packets: 3856 · Displayed: 16 (0.4%)   Profile: Default

2. Filtering HTTP and HTTPS Packets

Next, a filter was created to display only HTTP and HTTPS packets. This was achieved by applying a filter that identifies packets with the HTTP and HTTPS protocols.

3. Eliminating Cygwin Site Visits

To exclude packets related to the "cygwin.com" website, a filter was applied to eliminate packets associated with this specific domain. This step helps in focusing the analysis on packets related to the other visited websites (i.e., google.com and duckduckgo.com).



```
!(ip.addr == 8.43.85.97) and (tcp.port == 443 or tcp.port== 80)
```

Wireless controls are not supported in this version of Wireshark.

```
No.  Time         Source          Destination     Protocol Length Info
1955 15.313690531 172.253.115.103 10.199.117.43   TLSv1.3  298 Application Data
1956 15.313952694 10.199.117.43   172.253.115.103 TCP       66 52932 → 443 [ACK] Seq=1479 Ack=7811 Win=56704 L
1957 15.319168251 172.253.115.103 10.199.117.43   TLSv1.3   98 Application Data
1958 15.319190660 172.253.115.103 10.199.117.43   TLSv1.3   97 Application Data
1959 15.319302673 10.199.117.43   172.253.115.103 TCP       66 52932 → 443 [ACK] Seq=1479 Ack=7874 Win=56704 L
1960 15.319318120 172.253.115.103 10.199.117.43   TLSv1.3  105 Application Data
1961 15.319365607 10.199.117.43   172.253.115.103 TLSv1.3  105 Application Data
1962 15.321222958 172.253.115.103 10.199.117.43   TCP       66 443 → 52932 [ACK] Seq=7913 Ack=1518 Win=68864 L
1977 15.666651025 10.199.117.43   18.211.136.205  TCP       74 52052 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=896
1983 16.682212855 10.199.117.43   18.211.136.205  TCP       74 [TCP Retransmission] [TCP Port numbers reused]
2010 18.698224257 10.199.117.43   18.211.136.205  TCP       74 [TCP Retransmission] [TCP Port numbers reused]
```

```
▸ Frame 2010: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface ens5, id 0
▸ Ethernet II, Src: 0a:fc:bf:ae:5c:b7 (0a:fc:bf:ae:5c:b7), Dst: 0a:23:cb:e2:48:cb (0a:23:cb:e2:48:cb)
▸ Internet Protocol Version 4, Src: 10.199.117.43, Dst: 18.211.136.205
▸ Transmission Control Protocol, Src Port: 52052, Dst Port: 443, Seq: 0, Len: 0
```

```
0000  0a 23 cb e2 48 cb 0a fc  bf ae 5c b7 08 00 45 00   ·#··H···  ··
0010  00 3c 3d ce 40 00 40 06  e1 5b 0a c7 75 2b 12 d3   ·<=·@·@·  ·[··u
0020  88 cd cb 54 01 bb b3 98  54 e2 00 00 00 00 a0 02   ···T····
0030  f5 07 1b c1 00 00 02 04  23 01 04 02 08 0a 8b 84   ········
0040  4f e2 00 00 00 00 01 03  03 07                     O·······  ··
```

Transmission Control Protocol (tcp), 40 bytes

Packets: 3856 · Displayed: 2103 (54.5%) · Dropped: 0 (0.0%)

Profile: Default

Results

The results of the analysis are as follows:

The packet capture process successfully captured network traffic while visiting "google.com," "duckduckgo.com," and "http://cygwin.com."
Filtering port 80 TCP data isolated HTTP-related packets.
Further filtering to display only HTTP and HTTPS packets provided a list of packets related to web traffic.
By eliminating packets related to "cygwin.com," the analysis focused on HTTP and HTTPS traffic excluding the specified website.

Conclusion

This exercise demonstrated the use of Wireshark to capture and analyze network traffic related to specific websites. The process allowed for the isolation of HTTP and HTTPS packets while excluding packets associated with the "cygwin.com" website. This type of analysis can be valuable for troubleshooting network issues and understanding web traffic patterns.