Wireshark – Webpage Traffic

Host IP: 192.138.1.3

Subnet Mask: 255.255.255.0

Default Gateway: 192.138.1.2

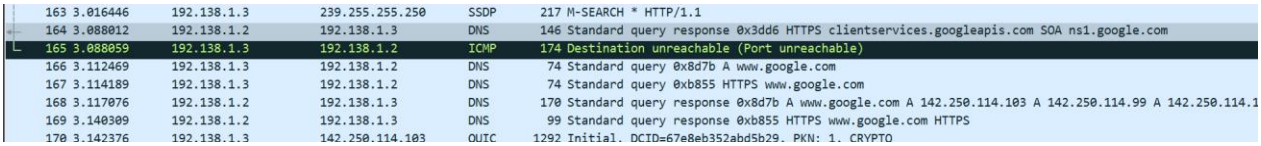
DNS Server: 192.138.1.2

A capture is started on Ethernet0 interface.

Navigate to google.com.

Once the page fully loads, the capture is ended.

When examining the capture, a DNS Query was made by the host to the DNS server.



This query is made for the host to resolve the URL “google.com” to an IP address that the browser can navigate to.

The DNS Server then sends a response packet.

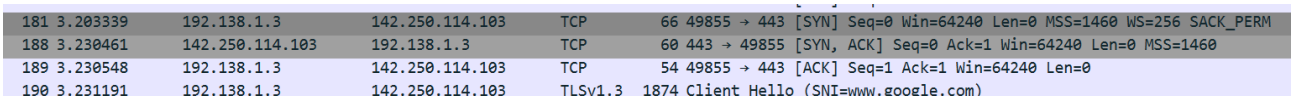
When clicking the packet and investigating the packet details, a collection of IP Addresses can be found as answers to the query.

```
> Internet Protocol Version 4, Src: 192.138.1.2, Dst: 192.138.1.3
> User Datagram Protocol, Src Port: 53, Dst Port: 50462
v Domain Name System (response)
    Transaction ID: 0x8d7b
  > Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 6
    Authority RRs: 0
    Additional RRs: 0
  > Queries
  v Answers
    > www.google.com: type A, class IN, addr 142.250.114.103
    > www.google.com: type A, class IN, addr 142.250.114.99
    > www.google.com: type A, class IN, addr 142.250.114.147
    > www.google.com: type A, class IN, addr 142.250.114.104
    > www.google.com: type A, class IN, addr 142.250.114.106
    > www.google.com: type A, class IN, addr 142.250.114.105
    [Request In: 166]
    [Time: 0.004607000 seconds]
```

When entering each of these IP Addresses directly into the browser, we confirm that all 6 of these addresses are assigned to google.com.

When examining the capture further, multiple TCP packets can be found being sent between the host and the first IP Address in DNS Response.

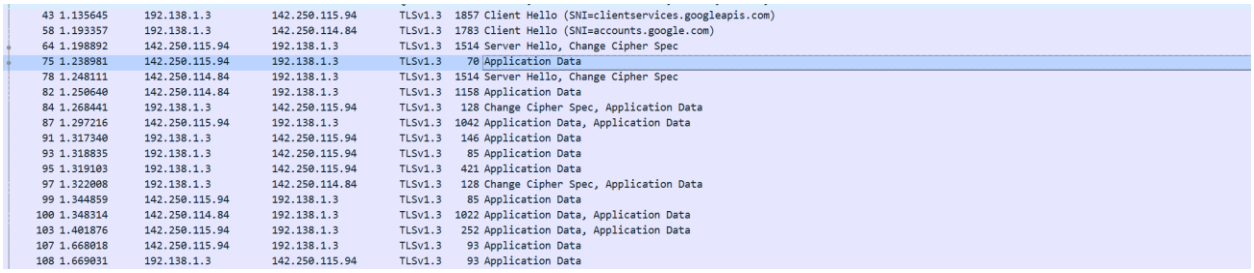
The capture is filtered for TCP packets for easier analysis.



The next 3 packets are evidence of the host attempting to make a connection with google.com with a 3-Way Handshake, which is completed.

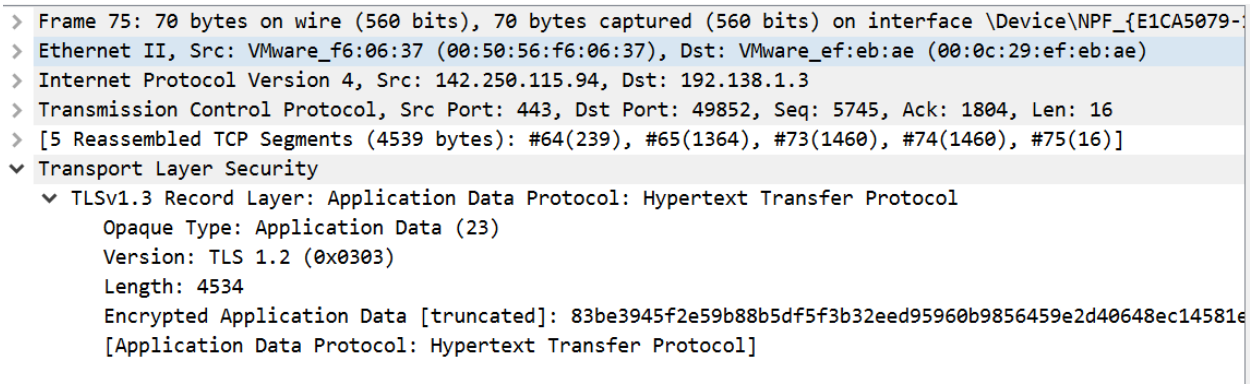
The next packets in the capture are sent using TLS.

The capture is filtered for TLS packets for easier analysis.



These packets are evidence of a TLS Handshake between google.com and the host in order to establish a secure connection.

When attempting to investigate the Application data sent by the Google server, we find that it is cyphertext.



HTTPS uses TLS to encrypt Application Layer data and maintain confidentiality if network traffic is captured.

The host has now established a secure connection with google.com.

A new capture in started on Ethernet0.

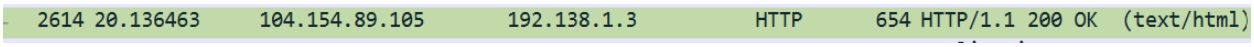
Navigate to http://http.badssl.com/.

Once the page fully loads, the capture is ended.

The packets captured are mostly similar to those from Google, but there are some key differences.

When examining the DNS packets, the browser attempts to relay the host to the website using HTTPS instead of HTTP for a secure connection.

Unfortunately, the DNS server can only respond with the IP Address of the website using HTTP.



Because of this, the Application Layer data is visible in complete plaintext in the packet details.