# CA169 Networks Assignment Two

# Answer Sheets

|  |  |
| --- | --- |
| STUDENT NAME: | James Fallon |
| STUDENT NUMBER: | 18394123 |
| PROJECT NUMBER: | 2 |
| MODULE CODE: | CA169 |
| DEGREE: {CA|EC|CPSSD|ECSA] | CA |
| LECTURER: | Brian Stone |

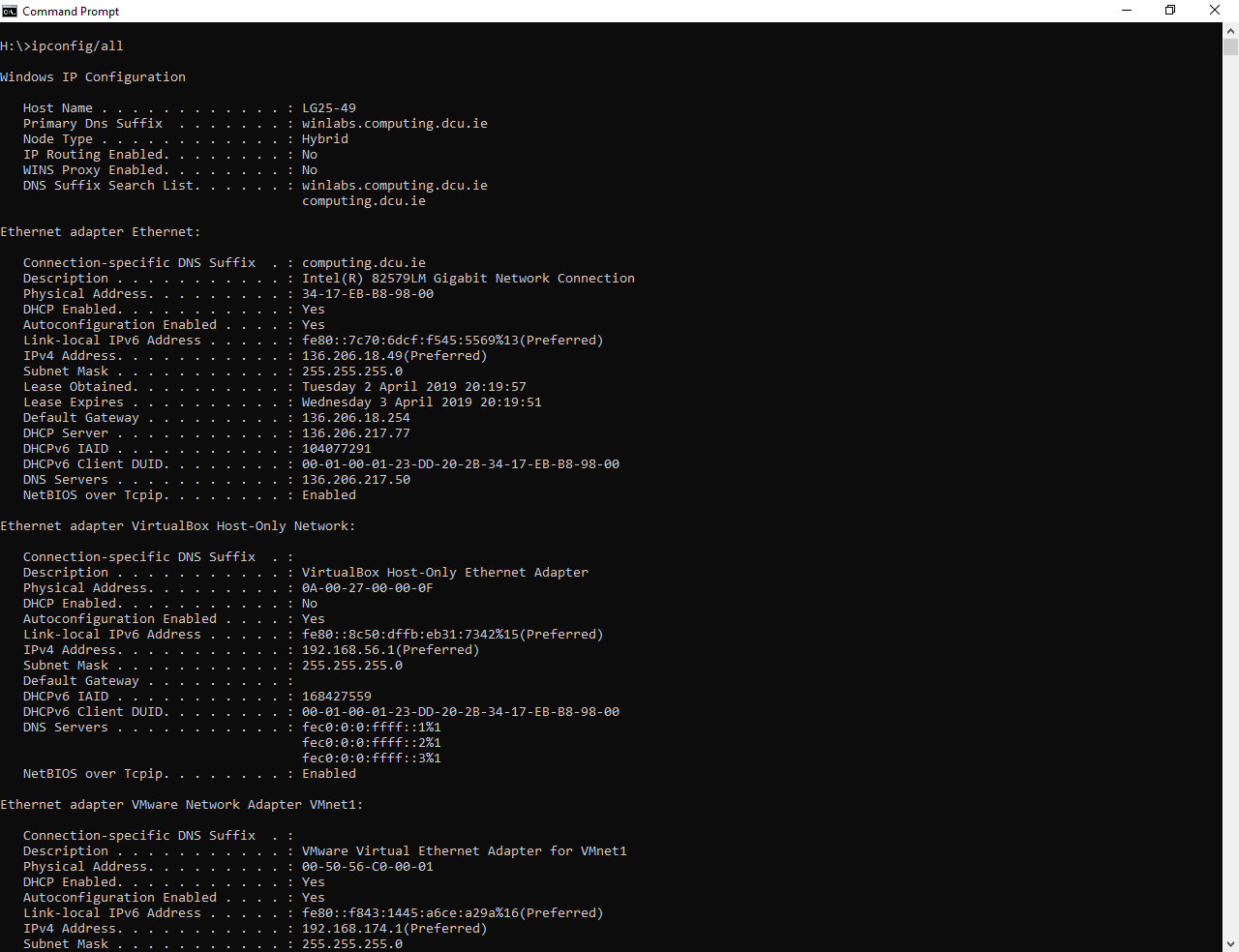
|  |
| --- |
| **Declaration**  ***In submitting this project, I declare that the project material, which I now submit, is my own work. Any assistance received by way of borrowing from the work of others has been cited and acknowledged within the work. I make this declaration in the knowledge that a breach of the rules pertaining to project submission may carry serious consequences.*** |

## Part 1: DHCP traffic

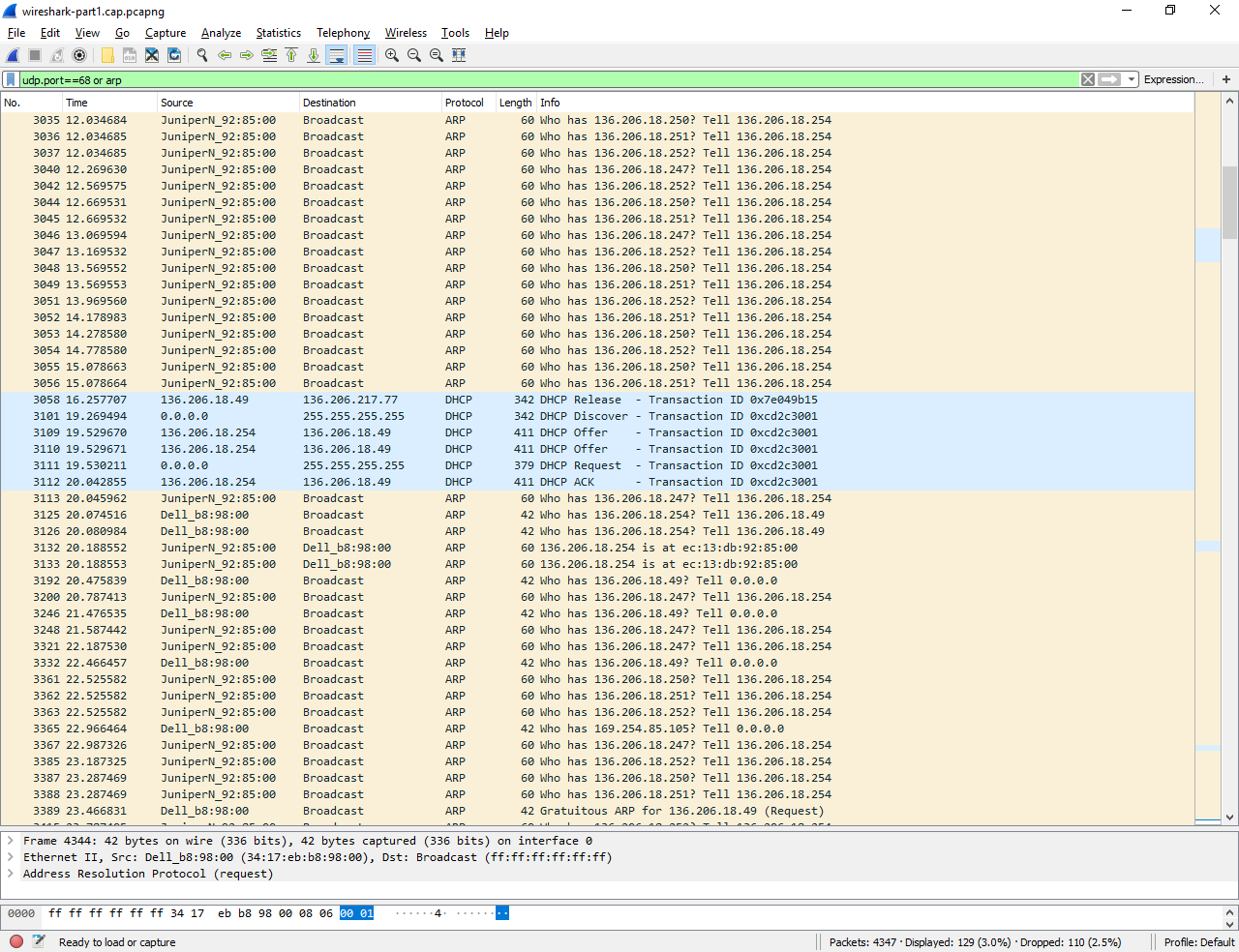
Your IP & MAC address for this experiment (use ipconfig)

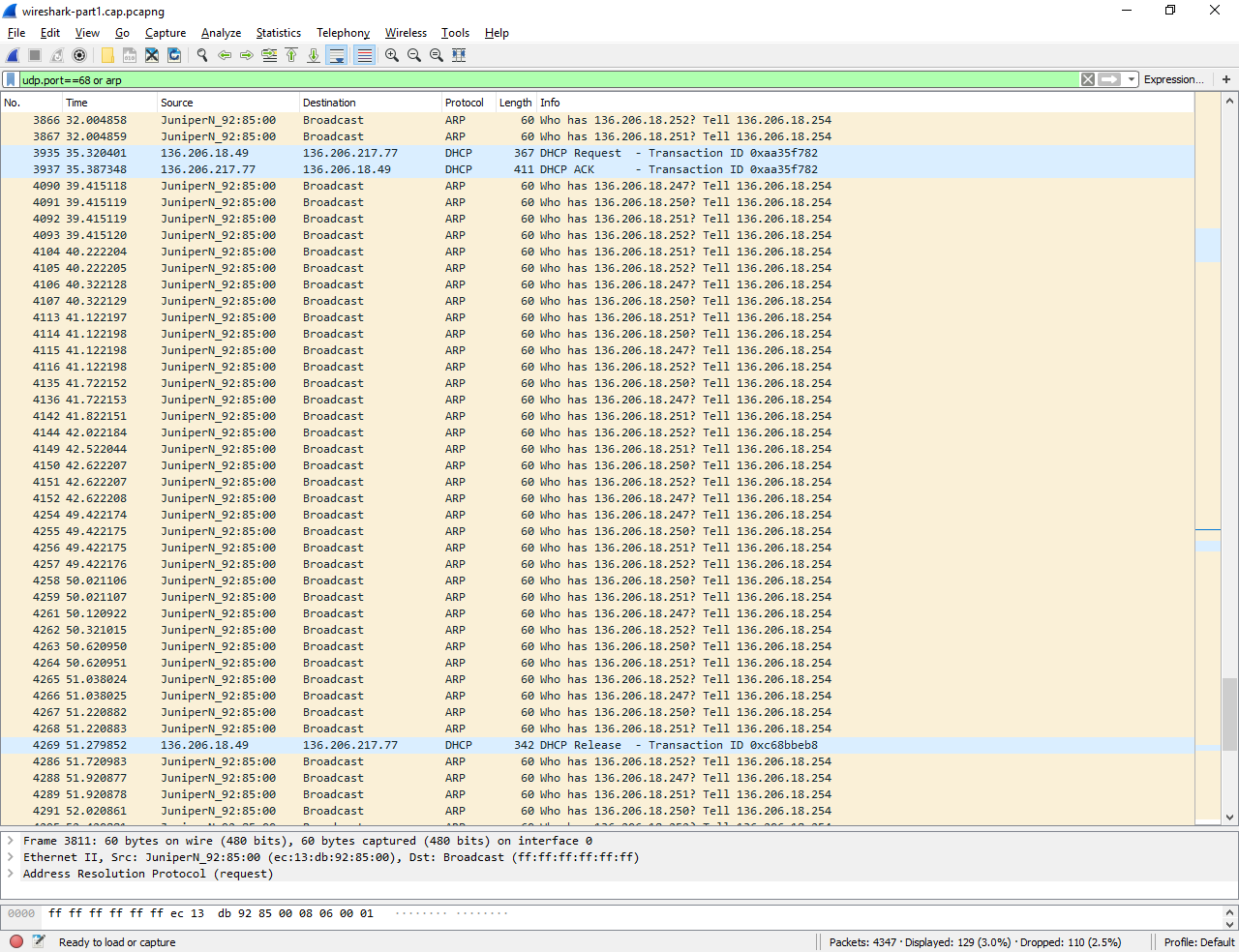
|  |  |
| --- | --- |
| 136.206.18.49 | 34-17-EB-B8-98-00 |

Screen capture: ipconfig information **cmd** window



Screen capture of Wireshark with DHCP and all ARP packets shown.





Packet numbers relevant to the DHCP interaction:

1. DHCP DISCOVER – Packet 3101
2. DHCP OFFER – Packet 3109, 3110
3. DHCP Request – Packets 3111, 3935
4. DHCP Acknowledgement – Packets 3112, 3937
5. DHCP Release (if you release using ipconfig /release) – Packets 3058, 4269
6. All ARP packets used - 3125, 3126, 3192, 3246, 3332, 3365, 3389, 3935, 3937, 4269, 4344

Function of each packet

1. DHCP DISCOVER – DHCP client sends a DHCP Discover broadcast on the network for finding a DHCP server. If there is no response from a DHCP server, the client assigns itself a new IP address.
2. DHCP OFFER – DHCP servers on a network that receive a DHCP Discover message respond with a DHCP offer message. This offers the computer an IP address.
3. DHCP Request – The computer accepts the first offer received by broadcasting a DHCP request message for the offered IP address.
4. DHCP Acknowledgement - The server accepts the request by sending the computer a DHCP acknowledgement message.
5. DHCP Release (if you release using ipconfig /release) – A DHCP client sends a DHCP Release packet to the server to release the IP address and cancel any remaining lease. The DHCP server may now lease this address to other clients.
6. ARP – ARP Packets get the MAC address from the machine that is pinged. It is then broadcasted across the network. It asks who has the IP address contained in the packet. If there is a machine with the IP address that was in the ARP request message, it sends an ARP Reply packet which contains its MAC address so that the machines may communicate.

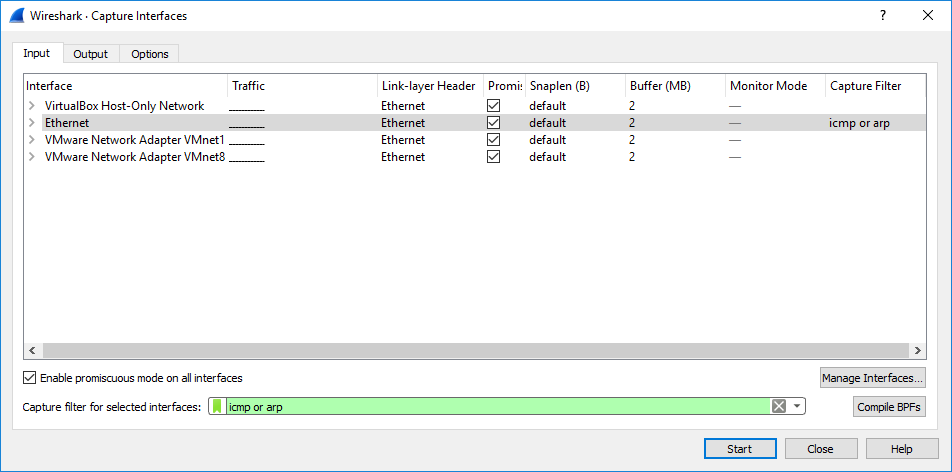
Packet 3389 is a Gratuitous ARP packet. This updates the ARP cache.

## Part 2: ping traffic

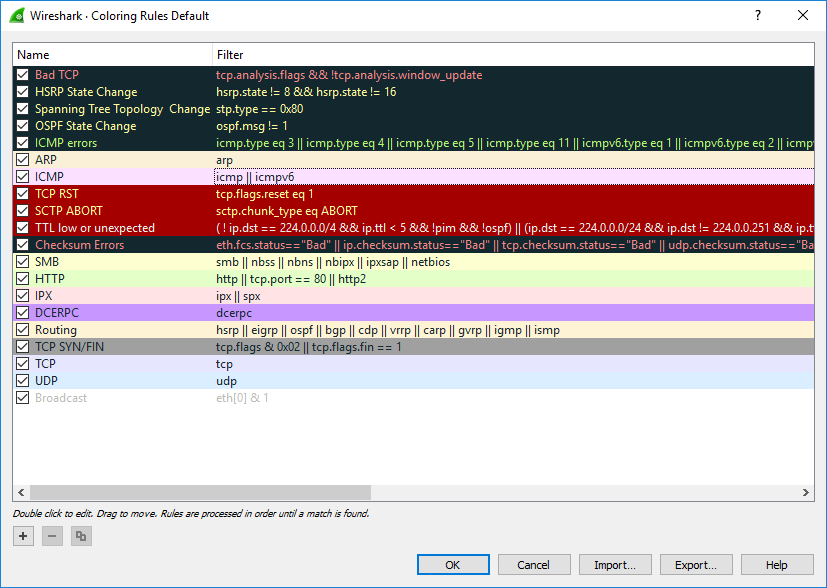
Your IP & MAC address for this experiment (use ipconfig)

|  |  |
| --- | --- |
| 136.206.18.49 | 34-17-EB-B8-98-00 |

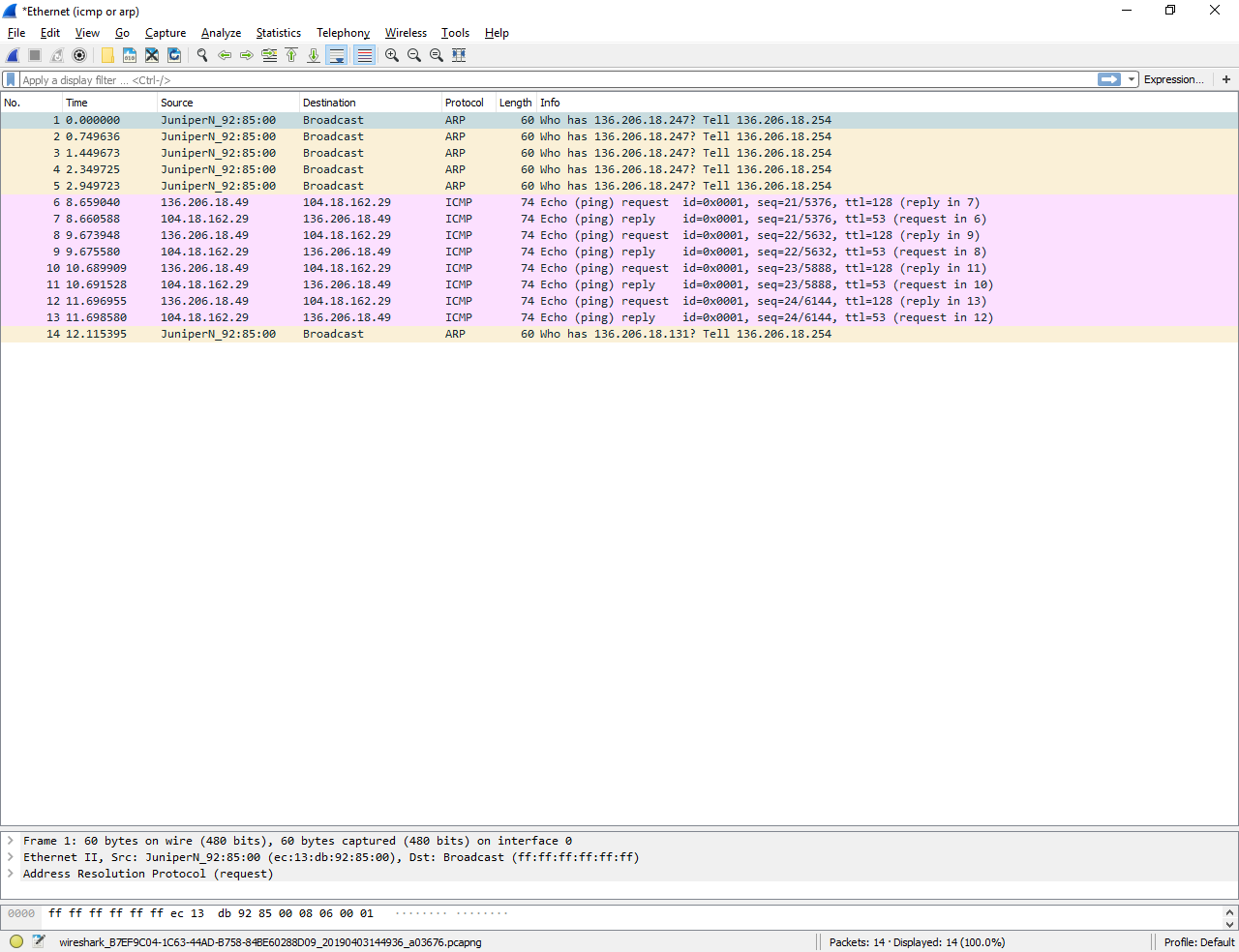
Screen capture of Wireshark filter utilised.



Screen capture of Wireshark colouring rules applied



Screen capture of Wireshark packet trace showing all relevant ping generated traffic, including ARP and ICMP traffic.



Packet numbers relevant to the experiment:

Packets 6 – 13 are relevant.

Explanation for each packet

I pinged [www.rte.ie](http://www.rte.ie) for this experiment.

* Function

Despite having cleared my ARP cache and carrying out this experiment multiple times, there were no ARP packets associated with my machine. However, I found out that ARP Packets get the MAC address from the machine that is pinged. It is then broadcasted across the network. It asks who has the IP address contained in the packet.

Packet 6 is an Echo Request. When I pinged [www.rte.ie](http://www.rte.ie), this packet was sent. It has a length of 74 bytes. The Echo Request uses the ICMP protocol. This protocol is used to check if one machine can communicate with another. It also has a TTL (Time to Live) which will tell us if there is any issues with the network.

Packets 7, 9, 11 and 13 are Echo Replies repeated. These are all sent from 104.18.162.29 ([www.rte.ie](http://www.rte.ie)). When my machine received these packets, it then knew that it could communicate with [www.rte.ie](http://www.rte.ie).

Packets 8, 10 and 12 are Echo Requests repeated.

* Explain why it is generated

The ARP messages are generated because we wiped the ARP cache and this information needed to be retrieved. ARP finds a MAC address that matches the IP address. It looks in the ARP cache and if it finds the address, provides it to the machine. In this case, no entry will be found because I cleared the ARP cache. ARP then broadcasts a request packet to all machines on the LAN to see if one machine knows that it has that IP address associated with it. ARP will update the ARP cache for future reference and then sends the packet to the Mac address that replied. The Echo Replies and Echo requests are generated and repeated because statistics can therefore be provided based on the connections.

* Explain the data contained in the packet

Echo Reply and Echo request contain data such as TTL, Round Trip time, and packet loss. ARP packets contain the source IP address and the target IP address. It also contains the source MAC and target addresses.

## Part 3:

Your IP & MAC address for this experiment (use ipconfig)

|  |  |
| --- | --- |
| 136.206.18.49 | 34-17-EB-B8-98-00 |

Filter to show only traffic concerning the test machine

|  |  |
| --- | --- |
| Filter | ip.addr==136.206.18.49 |

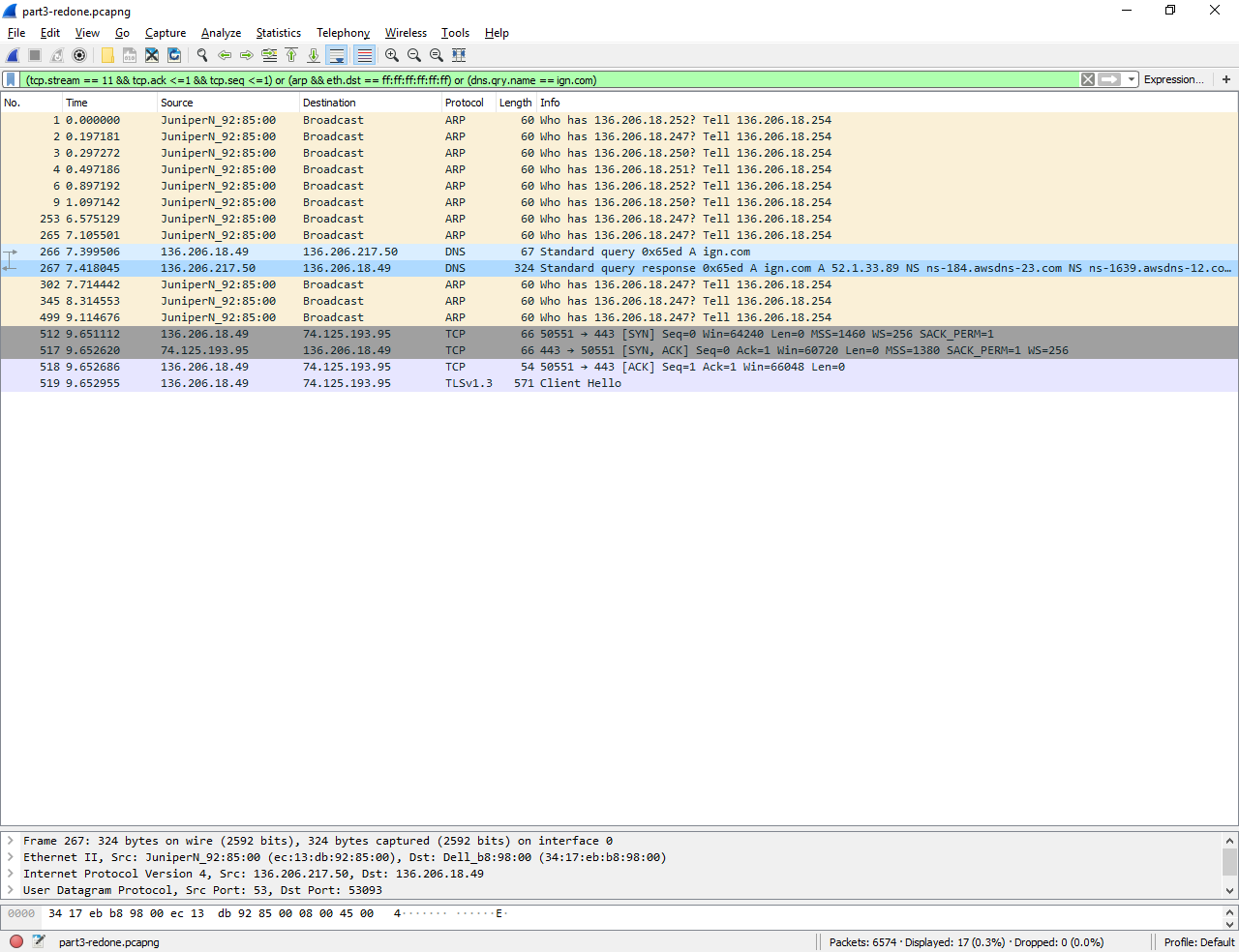
Explain how you found the start of the interaction between your PC and the website.

For this experiment, I searched [www.ign.com](http://www.ign.com)

I found the start of the interaction by using the filter (tcp.stream == 11 && tcp.ack <=1 && tcp.seq <=1) or (arp && eth.dst == ff:ff:ff:ff:ff:ff) or (dns.qry.name == ign.com).

This filtered through the unwanted traffic and left me with the ARP packets, DNS packets and the TCP 3-way handshake. The DNS packets contained ign.com and this is the website I contacted. The TCP packets are the first sent in this interaction. The ARP packets contain the MAC address.

Wireshark window showing the start of the interaction (should show ARP, DNS and TCP 3-way handshake)



Write down the numbers of the packets with the 3-way handshake.

Explain what is happening with these 3 packets.

Packet 512:

This is a SYN packet. A client node sends a SYN data packet over an IP network to a server on the same or an external network. The objective of this packet is to ask if the server is open for new connections.

Packet 517:

This is a SYN/ACK packet. The target server must have open ports that can accept and initiate new connections. When the server receives the SYN packet from the client node, it responds and returns a confirmation receipt, the ACK packet or SYN/ACK packet.

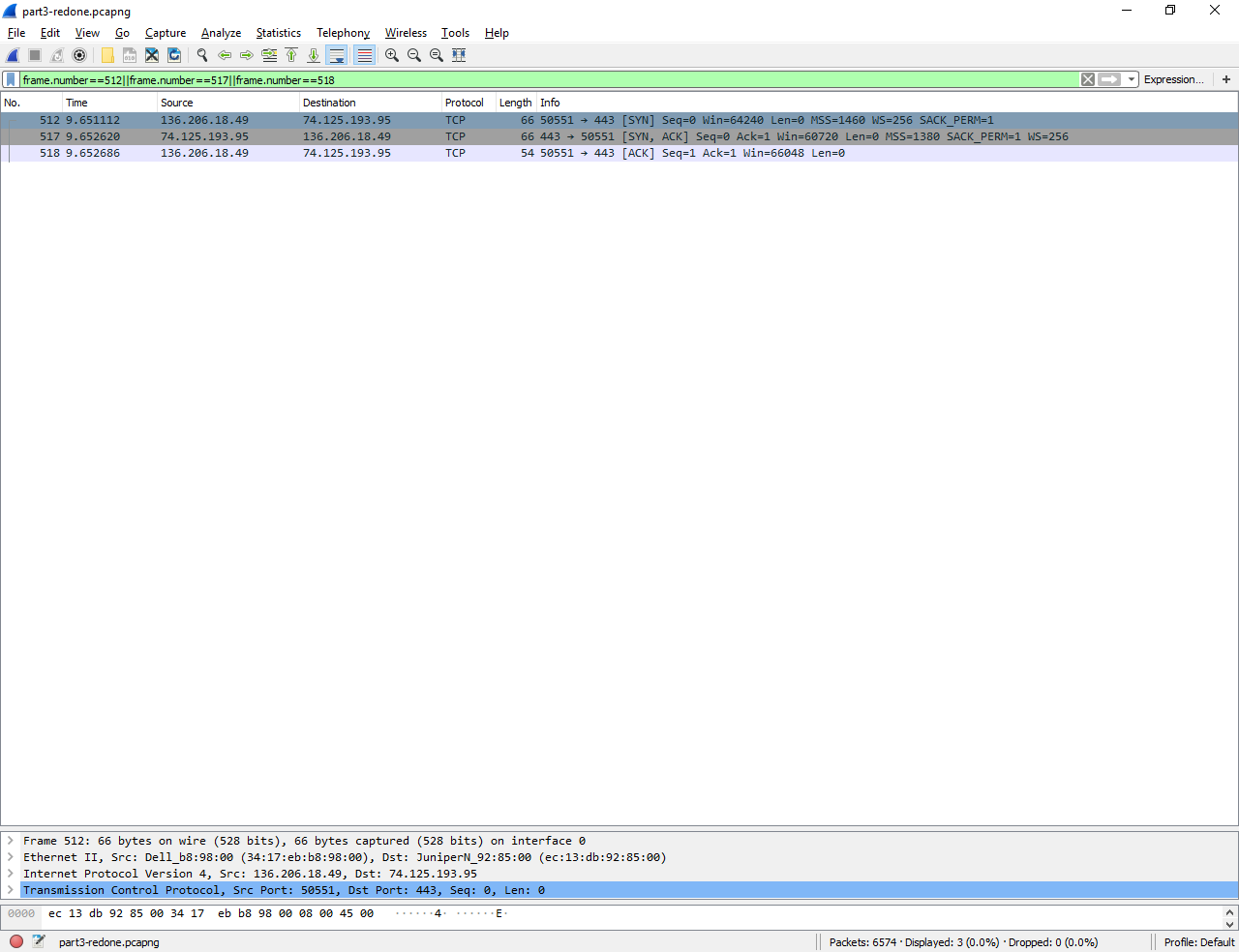
Packet 518:

This is an ACK packet. The client node receives the SYN/ACK from the server and responds with the ACK packet.

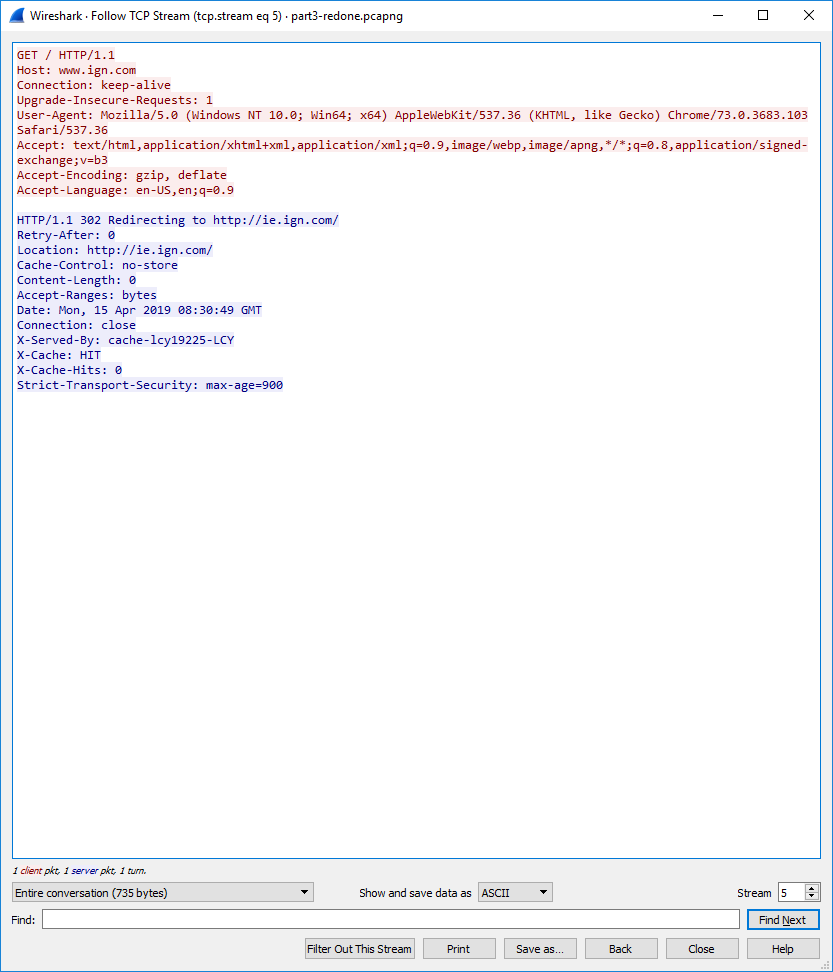
Write down a filter to show only these three-way-handshake packets

|  |  |
| --- | --- |
| Filter | frame.number==512||frame.number==517||frame.number==518 |

Wireshark window for the 3-way-handshake



Show the **Follow TCP Stream** window here.



Your notes on…

1. The GET requests made

I found that the GET request was packet 295. I used the filter http.request.method==”GET” to find this information. This packet contained the host site name. The Get request is used to request data from a specified resource.

1. The responses from the server

The first response from the server was the packet 285 HTTP /1.1 302 redirecting. This was followed by TLS handshakes. TLS is an encryption protocol designed to secure internet communications. During the TLS handshake, the two communicating sides exchange messages to acknowledge each other, verify each-other, establish the encryption algorithms they will use, and agree on session keys.

1. The HTTP response codes used in the interaction and what they mean (look them up yourself on the Web)

The HTTP response codes used in this interaction is packet 285, the HTTP/1.1 302 redirecting code. A 302 redirect means that the page was temporarily moved. It tells the browser reading a page to go somewhere else and load another page.