**Ulster University**

**COM 342 Networks**

**Lab Exercise - ARP**

**Student Name: Leo Groza**

**Student Number:**

Contents

[Objective & Insights 3](#_Toc500096231)

[Address Resolution Protocol 3](#_Toc500096232)

[Tools 4](#_Toc500096233)

[Setup 4](#_Toc500096234)

[Step One – Trace Capture 5](#_Toc500096235)

[Step Two – Inspect the Trace 7](#_Toc500096236)

[Table 11](#_Toc500096237)

[Bibliography 13](#_Toc500096238)

# Objective & Insights

The objective of the lab is to investigate the functionality of the Address Resolution Protocol (ARP).

Every device connected to the internet *must speak IP*, it must have at least one IP address to connect to the *grid.* Internet Protocol Address is an identifier assigned to each computer connected to a network, employed to identify and locate a communication node connected to different nodes on the network. Regrettably, the IP address is not sufficient to send data over the internet (Tanenbaum & Wetherall, 2011).

# Address Resolution Protocol

ARP is applied by the Internet Protocol to map an IP network address to the hardware address used by a data link protocol. Hardware address resolution denotes the process of finding the address of a computer on a network. Every machine working on the internet is using ARP.

The Address Resolution Protocol operates as part of the Open System Interconnection link layer and produces a result when IP is used over the Ethernet.

There are two types of addresses that are used to uniquely identify a host over the network:

- MAC Address = a unique six-byte address given by the card manufacturer;

- IP Address = (explained above) can be static or dynamically assigned by the Dynamic Host Configuration Protocol (DHCP).

The ARP is implemented to assign an IPv4 address (or logical address) to a physical address (or MAC address). The purpose of the ARP is to acquire the MAC address of a device located in your Local Area Network, for a corresponding IPv4 address, from where the network application is attempting to communicate (Omnisecu.com, 2008).

IPv4 is the fourth version of Internet Protocol, a protocol used on networks that implement packet-switching. It is one of the main protocols implemented on internetworking methods on the internet.

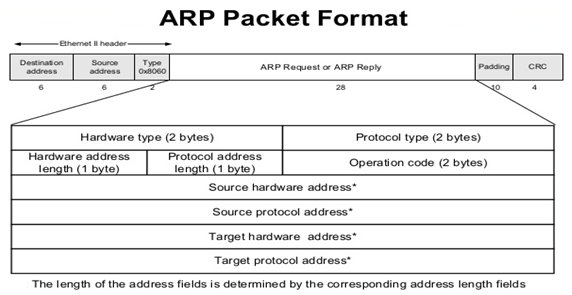
Technically, ARP exemplifies a table where the system keeps information about which IP addresses relate to MAC addresses. Prone to sending a packet to a designated IP address, the system will first inspect the ARP table to investigate if he already knows the MAC address. If there is a value, the ARP is not used, otherwise, the system will send a broadcast packet to the network, using the protocol of the ARP, to inquire who holds a specific IP address. All devices on the network will receive the packet, and the device with the corresponding IP will reply by providing the specific MAC address which can receive packets for its specific IP (tummy.com, ltd, 2013).

# Tools

The functionality of the ARP can be investigated with *Wireshark*. Wireshark is a very popular network packet analyser, applied to investigate application protocols and networks in a large variety of technologies. Wireshark is capable of saving, capture and importing packet files in a variety of formats. It can provide protocol information, has filtered capability, presents statistics that can help analyse and identify notable events in the network.

Developers make use of Wireshark to understand the routine in their code, investigate how the packets and data fields are managed in the application. Personnel in the network support make use of Wireshark to investigate, identify and resolve troubleshooting in connectivity, slow network and packet analysis (Baxter, 2014).

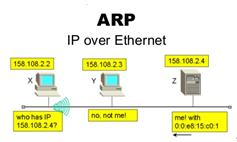
An *ARP Packet Format* tableis provided to be completed along the exercise with the specific details exposed by Wireshark. The fields contain details of the packet structure of ARP illustrating the case of IPv4 network running over the Ethernet.



# Setup

The purpose of the exercise is to observe the functionality of the ARP protocol. The typical example exploited to observe the functionality of the ARP is to observe the connection between a local IP linked with a router which connects the computer to with the internet.

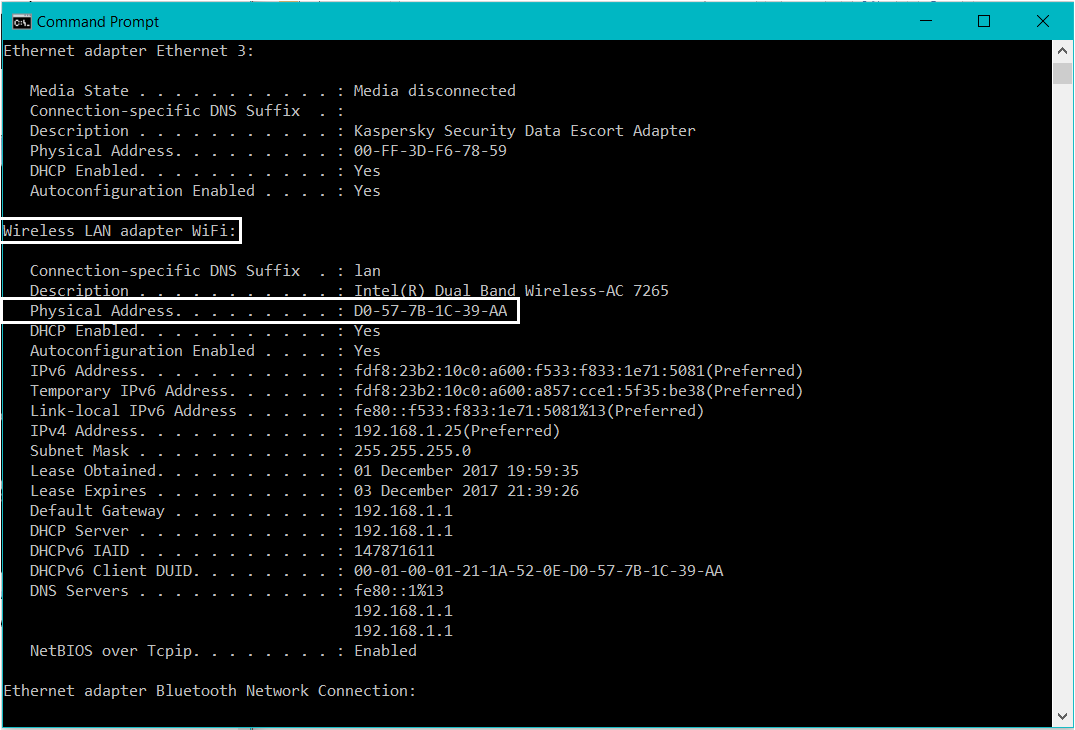
The setup from a computer’s point of view is presented in the image below.

(slideshare.net, n.d.)

# Step One – Trace Capture

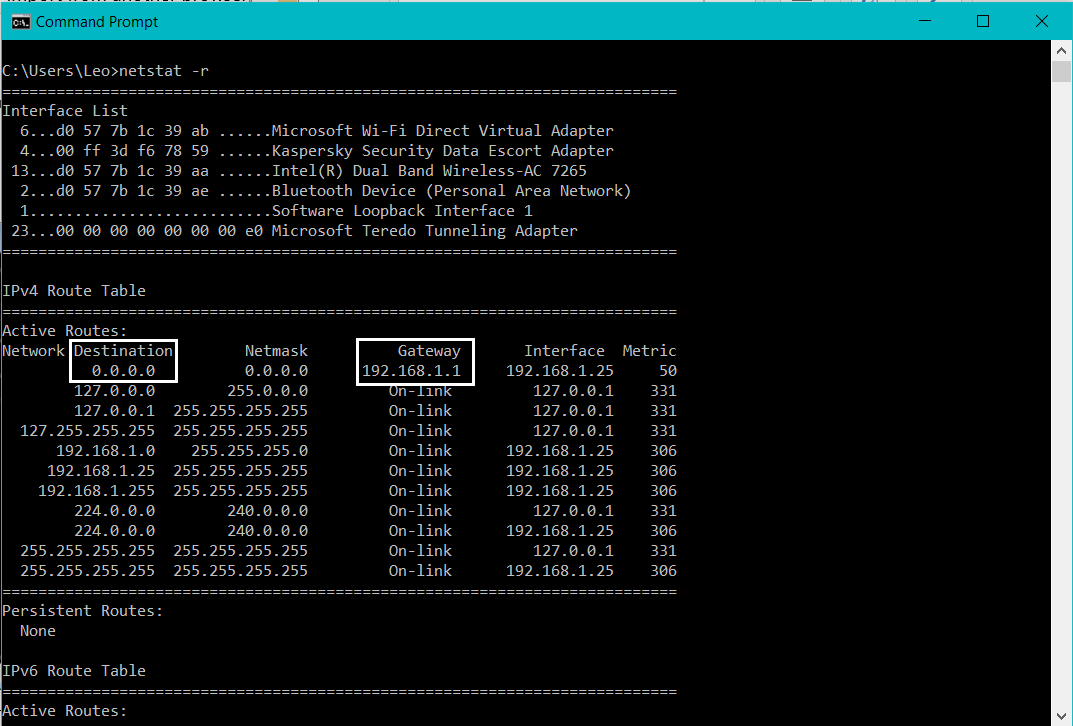
To gather data, the computer will be manipulated to send traffic to the router when it’s not aware of the Ethernet address of the router. The computer will make use of the ARP to unveil the Ethernet address.

1. To begin the exercise, first, the Ethernet address of the main network interface must be acquired. To acquire the Ethernet address, in the command line-shell, the *ipconfig /all* command will be inputted. The ipconfig */all* command is operated in Windows operating systems to investigate the network interface of the computer. The computer operated for this exercise only has Wifi capabilities, and the information needed will be found on the *Wireless LAN adapter Wifi* block. For Mac and Linux O.S. the alternative command is *ifconfig.*



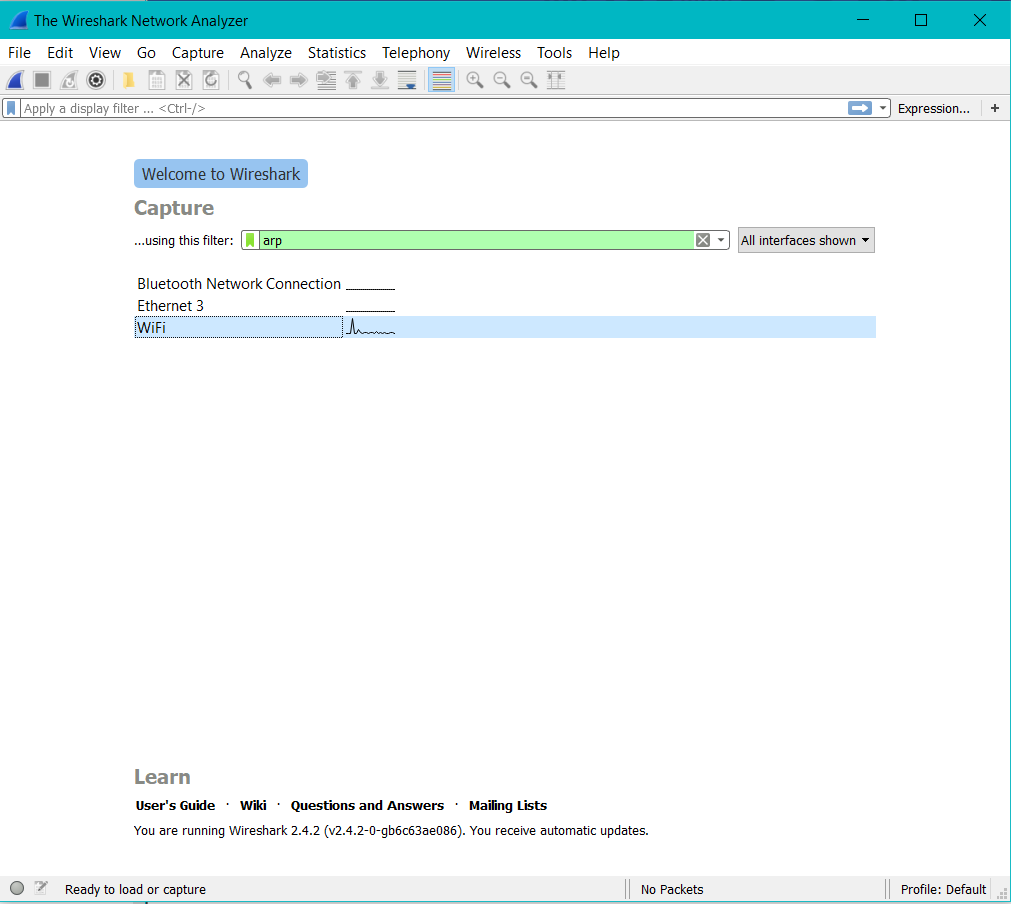
Finding the Ethernet address using *ipconfig* (Windows OS)

2. The second step is to acquire the IP address of the local router. An alternative option is to find the default gateway worked by the computer to contact the internet. To complete the step, in the command line-shell the *netstat -r* (Windows O.S.) command will be inputted to inspect the paths used by the computer to access the internet. The command line for Mac and Linux O.S. is identical, but extra command line parameters are needed to be used with the command line.



Finding the default gateway IP address with netstat -r

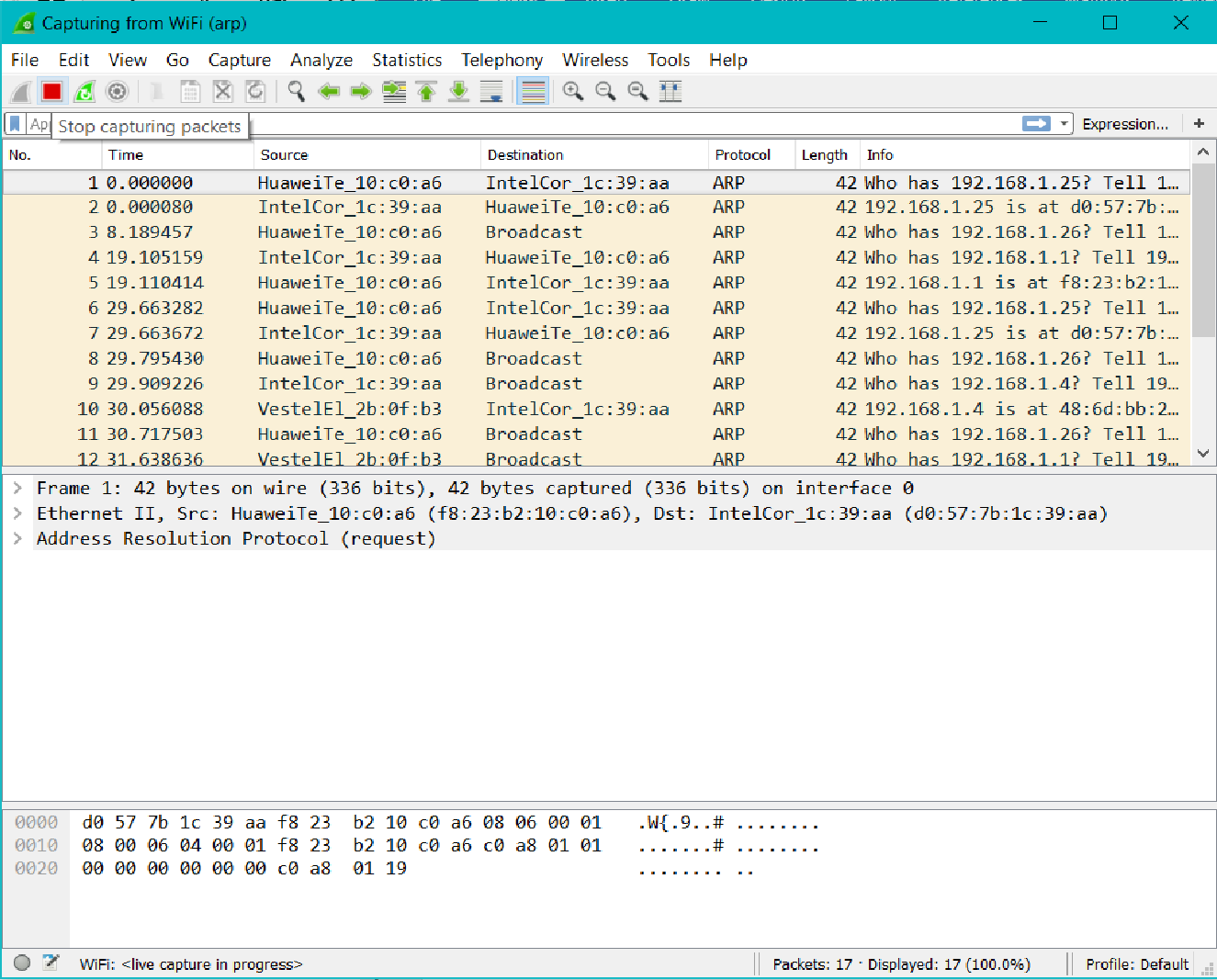
3. Engage Wireshark and start to capture data. To focus only on the essential data, a filter titled *arp* will be called for*.*  The command line investigates the cache data used by the ARP protocol. It is installed on Windows, Mac and Linux as part of the operating system. At the start of the capture, the *arp* command will clear the default gateway from the ARP cache.



Start the capture using an *arp* filter.

4. Following the clearance of the ARP cache the later step implies fetching a remote page with the web browser. The ARP protocol will be empowered to acquire the default gateway of the Ethernet address to direct the packets. Wireshark will capture the ARP packets. If additional computers make use of the same network, Wireshark will display different ARP packets conducted by other computers. The delay might extend to 30 seconds, taking in account the slow process of the ARP.

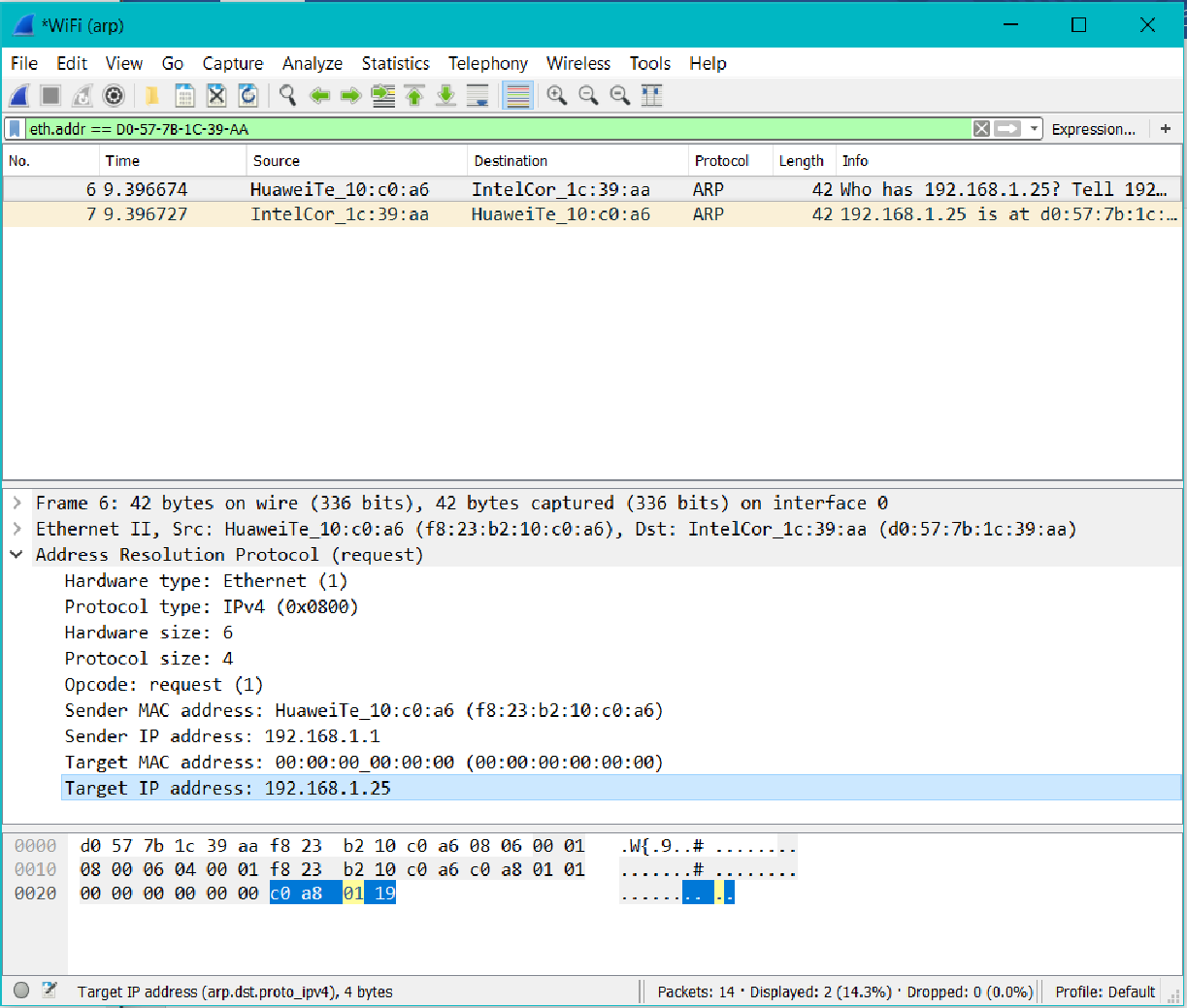
5. Shortly as Wireshark has captured data; the following step is to stop the capture. The trace and the Ethernet address of the computer, uncovered earlier will be managed to complete the succeeding steps.



Stop the capture of packets.

# Step Two – Inspect the Trace

The ARP exchange can be seen at this point. Perhaps several ARP packets will be exposed by Wireshark and it is required to narrow the view, only at the packets sent from, or, to your computer. This can be ensured by keying an expression in the *Filter box*, the expression will enclose the Ethernet address of the computer. If the Ethernet address is ‘*D0:57:7B:1C:39:AE’* the expression has to be ‘*eth.addr== D0-57-7B-1C-39-AE’.* Once the filter has been applied, the output is similar to the following image.



Details of a request.

The next phase is to select one of the ARP and inspect the fields.

There are two types of ARP packets:

*- Request;*

*- Reply.*

The line for the request starts as following: “Who has….00.00.00.00”; where “00.00.00.00.00” is your computer’s default gateway.

Expanding the Address Resolution Protocol block, the program will display the following fields:

- Hardware and Protocols; informs the Ethernet is the hardware type and IP is the protocol;

- The size of the Hardware and Protocol are set to 6 and 4;

- Opcode informs that is a request;

- The final four fields are completed as ample as possible, they display the following:

- MAC address of the sender;

- IP address of the sender;

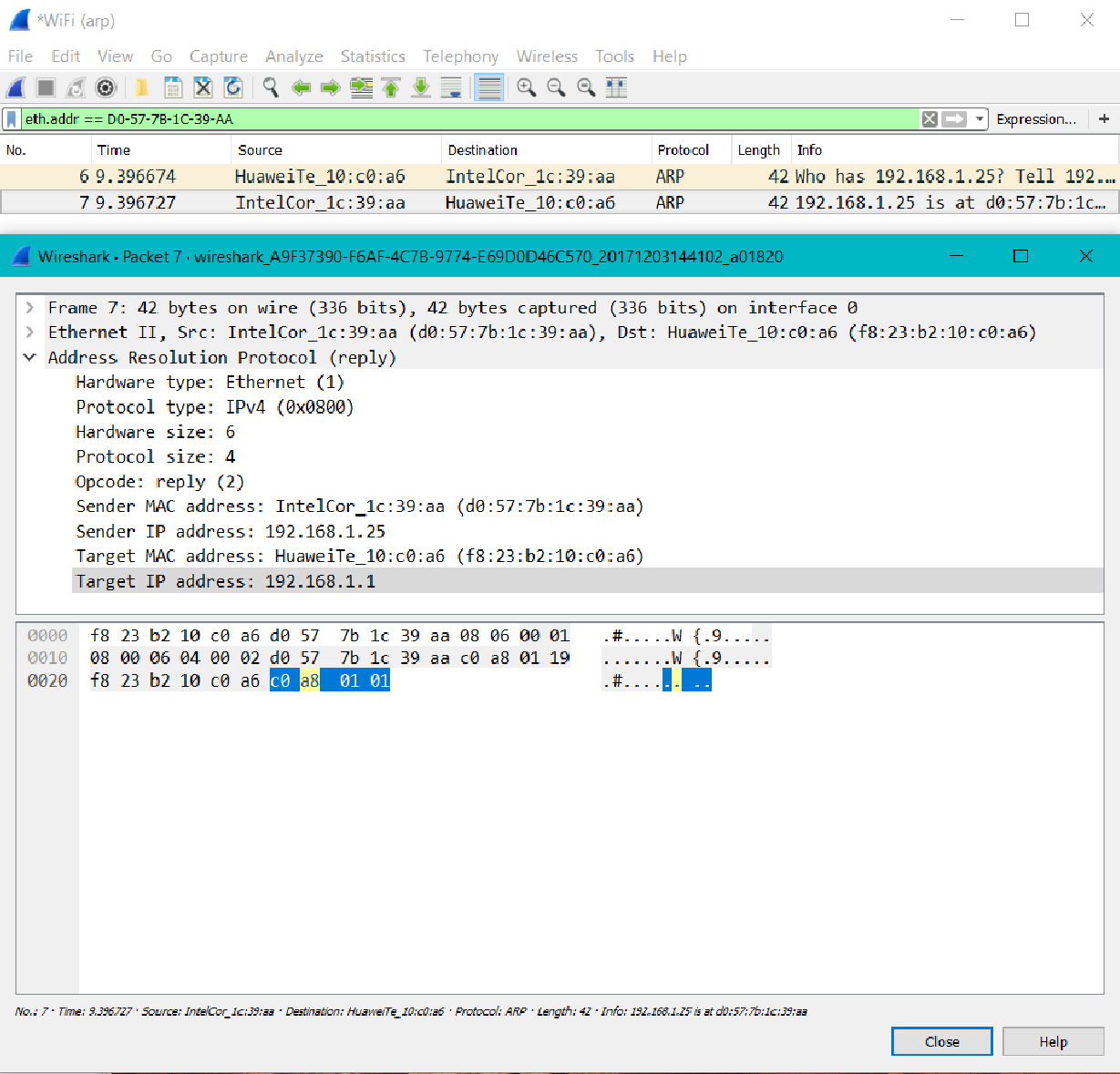
- IP and MAC address of the target.

The line following the *request*, which starts with “00.00.00.00” is the reply.

- Hardware and Protocol are the same type and size, and are set as per the *request* field;

- Opcode informs that it is a *reply*;

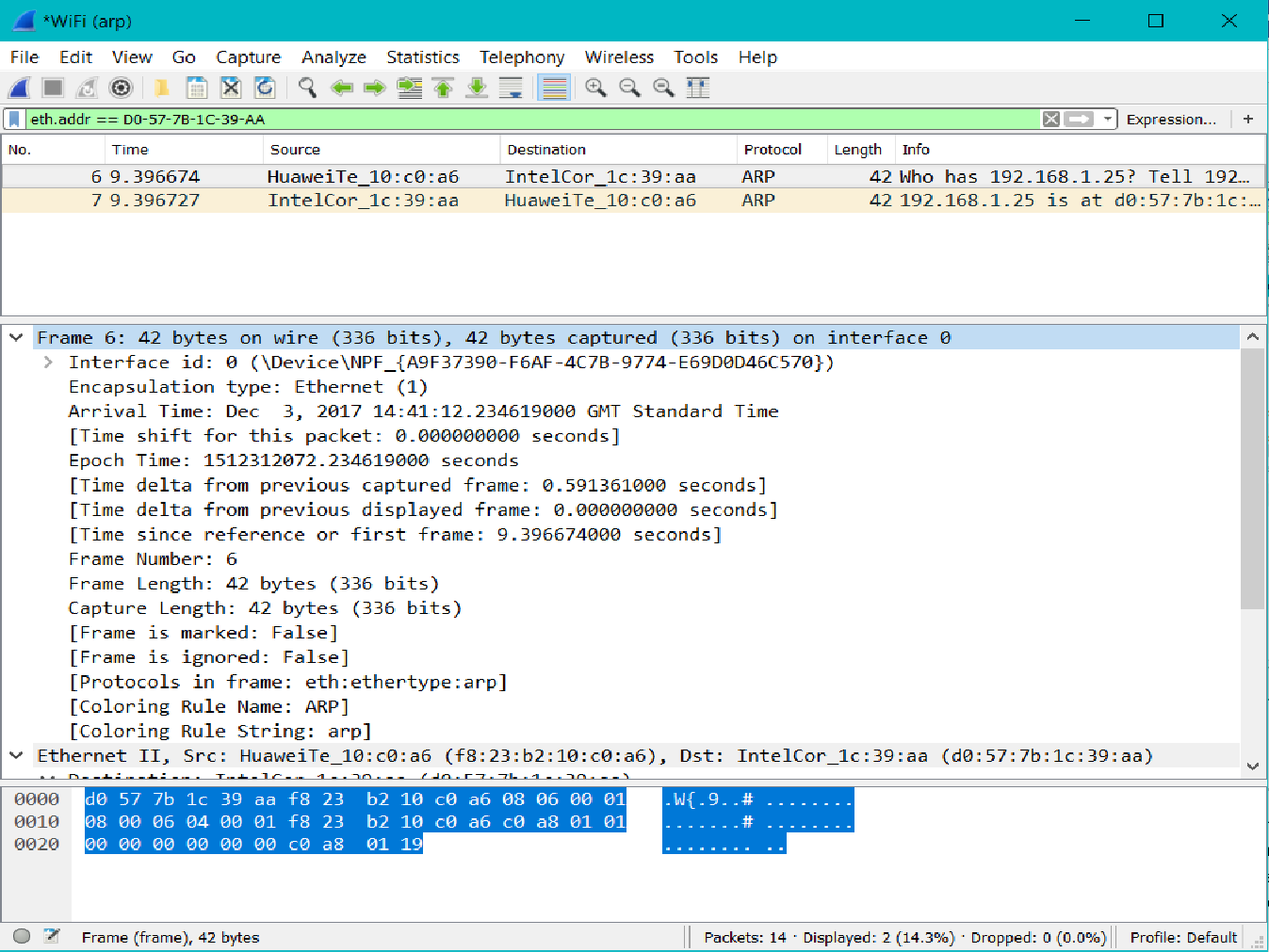
- The key fields are now reversed in contrast with the request, the target is now the sender. The fields are occupied at this point succeeding the address switch of the computers.



Details of a reply.

The Ethernet frames are the ones carrying the ARP packets. The values of the Ethernet header fields are selected to sustain ARP.

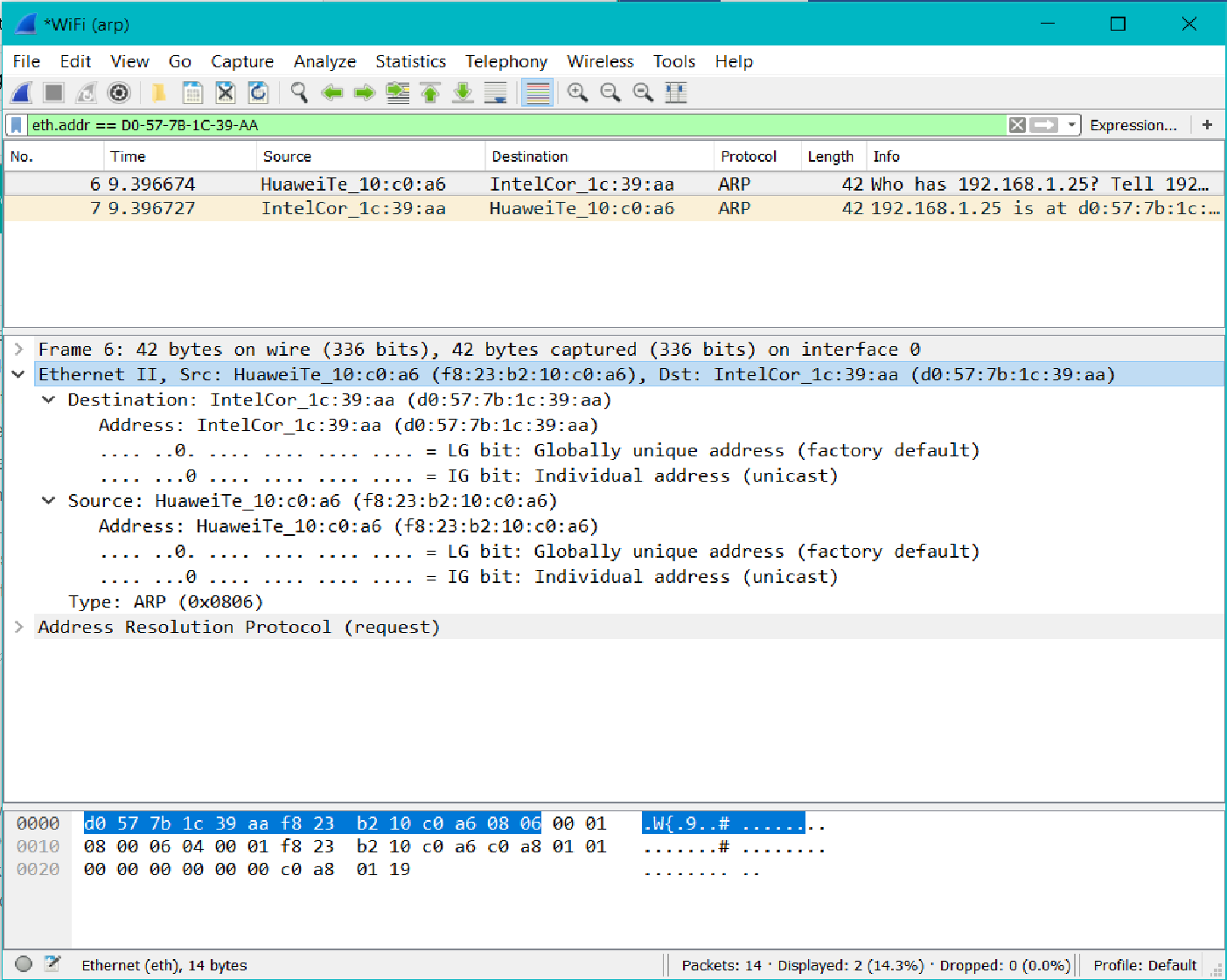
A frame stands for digital data transmission in telecommunications and networking. As a standard, a frame includes synchronisation frame features composed of symbols that indicate the beginning, the end and the payload data to the receiver. If a receiver connects to the system in the middle of a frame transmission, the data will be ignored until a new frame synchronisation is detected. The frame detail for a request is presented in the image below.



Frame details.

Ethernet is a link protocol that defines in what way a device connected to a network can create data for transmission to another device on the same network segment. It defines frames and packets as units of transmission. The frame includes, besides the data to be transmitted, the MAC (Media Access Control) address from both the sender and the receiver and information about error correction to identify eventual problems in the transmission.

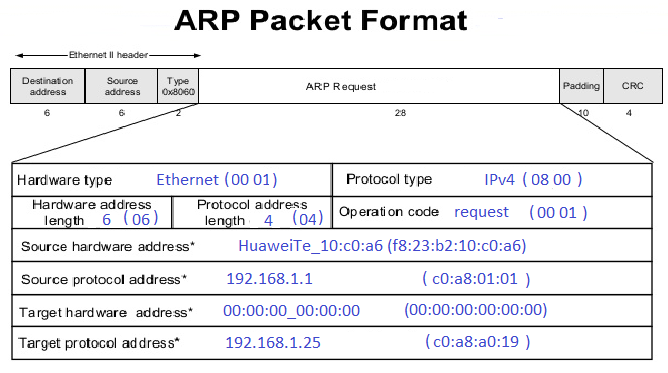
The Ethernet details specific to our exercise are presented below.



Ethernet details.

# Table

The ARP Packet Format table, for request, according to our exercise is the following:



In the Ethernet block, the destination address, the source address and the type of the operation can be found as:

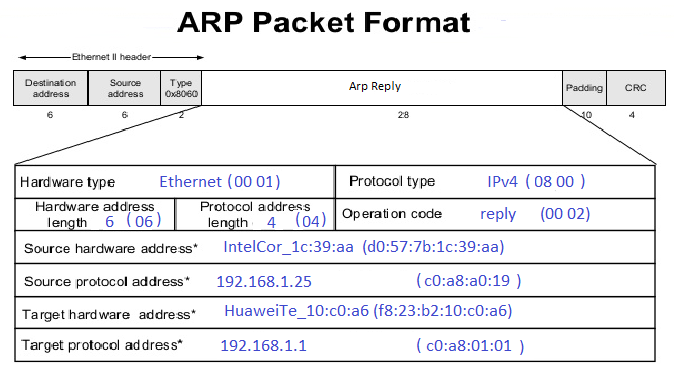
- Destination: IntelCor\_1c:39:aa (d0:57:7b:1c:39:aa);

- Source: HuaweiTe\_10:c0:a6 (f8:23:b2:10:c0:a6);

- Type: ARP (0x0806).

The information is included in the Ethernet II header.

The ARP Packet Format table, for a *reply*, relates to the one for the *request*.



At a *reply* all the fields are completed with the addresses of both the sender and the receiver.

In the Ethernet header, the destination address, the source address, are the same but in inverse order; at a reply, the sender becomes the receiver and the receiver becomes the sender, as follows:

- Destination: HuaweiTe\_10:c0:a6 (f8:23:b2:10:c0:a6);

- Source: IntelCor\_1c:39:aa (d0:57:7b:1c:39:aa);

- Type: ARP (0x0806).

# Bibliography

Baxter, J. H., 2014. *Wireshark Essentials.* s.l.:Packt Publishing.

Omnisecu.com, 2008. *Address Resolution Protocol Tutorial How ARP work, ARP message format.* [Online]   
Available at: http://www.omnisecu.com/tcpip/address-resolution-protocol-arp.php  
[Accessed 01 12 2017].

slideshare.net, n.d. *Address Resolution Protocol ARP Table.* [Online]   
Available at: https://www.bing.com/images/search?view=detailV2&ccid=ahS0TVY3&id=854323169CF65BD3D23E1EA247B3BF87E8D422D7&thid=OIP.ahS0TVY311x8gKPPXmoPWwEsDh&q=network+setup+to+study+arp&simid=607997178647216680&selectedIndex=0&ajaxhist=0  
[Accessed 03 December 2017].

Tanenbaum, A. S. & Wetherall, D. J., 2011. *Computer Networks.* 5th Edition ed. s.l.:s.n.

tummy.com, ltd, 2013. *Networks Basics: How ARP Works.* [Online]   
Available at: https://www.tummy.com/articles/networking-basics-how-arp-works/  
[Accessed 01 12 2017].