**CSC 335 Data Communications and Networking**

**Examining the Internet with Wireshark.**

**Student’s name: Tyler Prehl**

**Date of the Experiment:** 6/30/21

**1. Objectives**

The goal of this lab is to investigate the functionality of the Internet using network protocol analyzer Wireshark.

**Please answer all questions and attach required screenshots to this lab report and then submit it to D2L.**

**2. Introduction and Background**

The Wireshark network protocol analyzer (former Wireshark) is a tool for capturing, displaying, and analyzing the frames, packets, and messages that are exchanged in a network. The Wireshark package can be downloaded from <http://www.wireshark.org/download.html>. Download the latest version. Note that in some computing environments, such as MS Windows, it is necessary to install a separate file capture utility (WinPcap for MS Windows). This utility is included to the latest version of the Wireshark installation package.

One’s understanding of network protocols can often be greatly deepened by “seeing protocols in action” and by “playing around with protocols” – observing the sequence of messages exchanged between two protocol entities, investigating the details of protocol operation, and causing protocols to perform certain actions and then observing these actions and their consequences. In this lab, you’ll be running various network applications in different scenarios using a computer on your desk, at home, or in a lab. You’ll observe the network protocols in your computer “in action,” interacting and exchanging messages with protocol entities executing elsewhere in the Internet.

In this lab we will investigate the following protocols:

|  |  |
| --- | --- |
| Protocol | Objective |
| DNS | Investigation and analysis of DNS query and  response messages. |
| ARP | Observation of sending and receiving ARP  messages |
| ICMP | Exploration of two applications of the Internet  Control Message Protocol (ICMP):  1. *Ping* to determine whether a host is reachable  2. *Traceroute* to allow users to determine the route that a packet takes from a local host to a remote host. |
| IP, UDP,  TCP | Analysis of the layered structure of network  protocols. |

**3. Lab Procedures**

**3.1 Installation**

Download and install Wireshark on your computer. Do not forget to install WinPcap utility, if you use MS Windows. You can use any operating system but this manual is written for MS Windows.

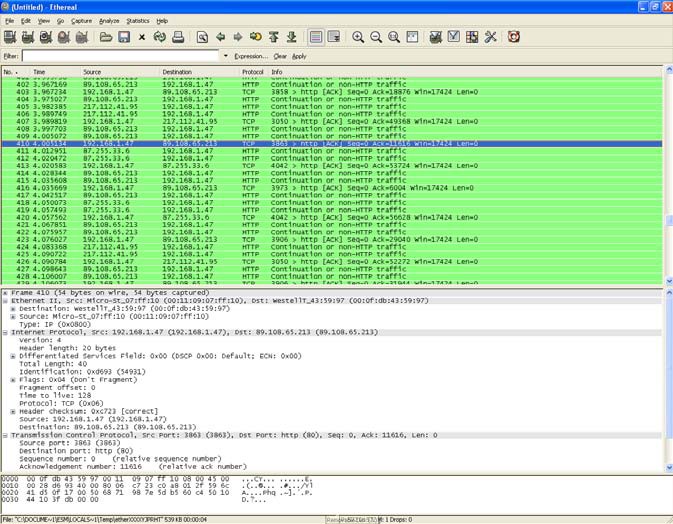
**3.2 Using Wireshark**

Wireshark user manual can be found here: <http://www.wireshark.org/docs/>

If you have any questions about this software, you can refer to this document.

When you run the Wireshark program, the Wireshark graphical user interface will be displayed. Creating a packet capture file is straightforward. Once the Wireshark application (and packet capture utility) is installed, you simply start Wireshark and select the “Capture” menu option. Be sure that the interface option is set to whichever interface your computer uses, if more than one is listed.

When you finish capturing packets, the information about the captured traffic will be shown on your screen:



The upper part of the screen shows the information about all packets captured by the system. You can use filters to display only specified patterns. When a packet is highlighted in the upper pane of the main window, the lower panes will show you more detailed information about a given packet. It will show each protocol layer of the selected packet: the physical

layer frame, the Ethernet frame and its headers, the Internet Protocol datagram and its headers, Transport layer protocol datagram and its headers, and the Hypertext Transfer Protocol (HTTP) notify message. For each protocol, you can expand the information even further. For example, if you expand the IP Layer, you can see each field in the IP header including version, the header length, etc. The lowest part of the main window shows each byte of the data contained in the packet.

**3.3 Address Resolution protocol (ARP)**

In this section we will investigate the ARP protocol. Recall that the ARP protocol typically maintains a cache of IP-to-Ethernet address translation pairs on your computer The *arp* command (in both Windows and Linux/Unix) is used to view and manipulate the contents of this cache. Since the *arp* command and the ARP protocol have the same name, it’s easy to confuse them. But keep in mind that they are different - the *arp* command is used to view and manipulate the ARP cache contents, while the ARP protocol defines the format and meaning of the messages sent and received, and defines the actions taken on message transmission and receipt.

In order to observe your computer sending and receiving ARP messages, it is necessary to clear the ARP cache, since otherwise your computer is likely to find a needed IP-Ethernet address translation pair in its cache and consequently not need to send out an ARP message.

 Run command terminal: Start -> Run -> cmd (Windows)

 Clear your ARP cache (*arp –d \** command). Sometimes this command does not remove the ARP address of the default gateway, so you can use the command *arp –d*

*<IP address>* to remove it from the cache.

If you have Windows Vista, please read this information about how to clean the cache:

<http://www.mydigitallife.info/2007/06/20/clear-delete-and-refresh-arp-cache-entry/>

 Type *arp –a* and make sure your ARP cache is empty

 Make sure your browser’s cache is empty. (For Internet Explorer, select *Tools →*

*Internet Options → Delete Files)*

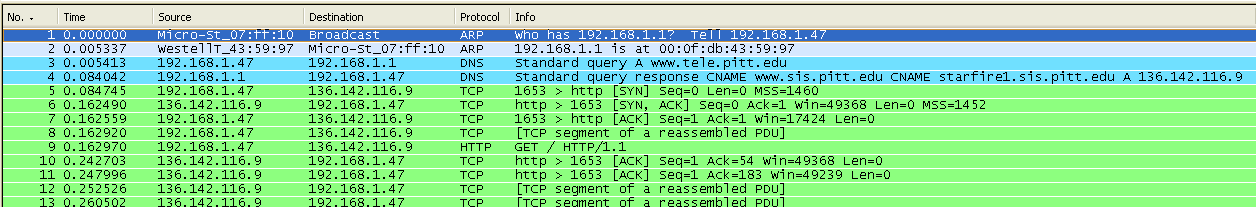
 Use *ipconfig* to empty the DNS cache in your host (command ipconfig /flushdns).

 Switch off any applications running on your computer, which generate the Internet traffic (downloaders, bit torrent client, etc). This will eliminate the capturing of unwanted traffic. Don’t use “Promiscuous mode” in the Wireshark. In this mode the program will capture all traffic in your network.

 Start up the capturing of packets

 Using your browser, visit any Web page. For example, [www.wcupa.edu.](http://www.wcupa.edu.)

 Stop Wireshark packet capture.



 Save the captured information to a file. You will use it in you future analysis.

Find two frames which contain ARP messages (use filter if necessary). The “Info” section of packets description should contain messages similar to:



Of course, you will have different IP/MAC addresses.

**Questions:**

1. Provide a screenshot with ARP Request and ARP Reply messages

2. Is ARP Request Multicast or Unicast packet?

ARP requests are Unicast packets.

3. Is ARP Reply Multicast or Unicast packet?

ARP replies are Unicast packets

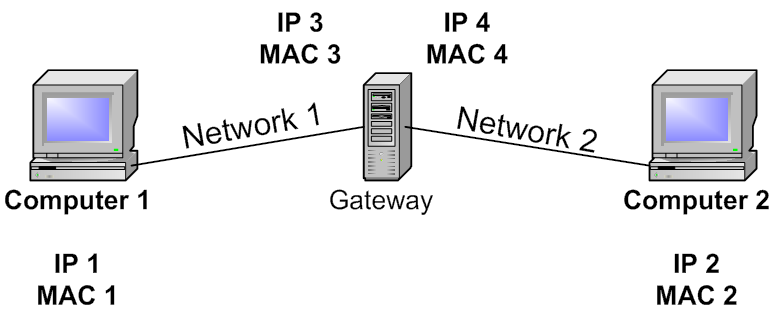
4. What is the 48-bit Ethernet address (MAC address) of your computer?

My MAC address is 00-E0-4C-68-2B-1B

5. What is the 48-bit destination address in the Ethernet frame? Is this the Ethernet address of www.wcupa.edu (if you typed this URL in your browser)? (Hint: the answer is *no*). What device has this as its Ethernet address? Explain.

The MAC address belongs to my home router since that is the next machine destination for where my datagrams are headed.

6. Let’s assume that you have the following network architecture:



The computer 1, which has IP1 and MAC1, sends a packet to the computer 2 with IP2 and MAC2. The computer 2 is located in a different subnet, so the packet crosses a gateway, which has two interfaces (IP3, MAC3 and IP4, MAC4).

Describe the structure of the packet (IP and MAC addresses), when a) the packet is in the subnet 1

b) the packet is in the subnet 2

To answer this question correctly, you have to understand very clearly a difference between

IP and MAC addressing.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Sender IP** | **Sender MAC** | **Target IP** | **Target MAC** |
| Network 1 (subnet 1?) | IP1 | MAC1 | IP2 | MAC3 |
| Network 2  (subnet 2?) | IP1 | MAC4 | IP2 | MAC2 |

**3.4 Domain Name System (DNS)**

When you typed a name of the web-site in your browser, this name was translated to its IP address. The Domain Name System (DNS) translates hostnames to IP addresses, fulfilling a critical role in the Internet infrastructure.

In the previous section of the lab you visited a web-site in the Internet and captured the information about packet exchanges in the network. Use this information to answer the following questions.

**Questions:**

7. Provide a screenshot with the DNS query and response messages.



8. Locate and examine the DNS query and response messages. Are they sent over

UDP or TCP?

They are sent over UDP.

9. Is DNS Query Multicast or Unicast packet?

Multicast

10. Is DNS Response Multicast or Unicast packet?

Multicast

11. What is the destination port for the DNS query message? What is the source port of

DNS response message?

The destination port for the DNS query message is 59785, and the source port of the DNS response message is 53.

12. To what IP address is the DNS query message sent? Use ipconfig *(ipconfig /all or ipconfig <interface>)* to determine the IP address of your local DNS server. Are these two IP addresses the same?

It is sent to 192.168.1.1, and it is the same as my IP address on my local DNS server since my home router uses NAT.

13. Examine the DNS query message. What query information is contained in the packet?

The domain name is contained in the packet (c.bing.com).

14. Examine the DNS response message. What is the IP address of the URL you typed in your browser?

It either came from 13.107.21.200 or 204.79.197.200, I’m not sure which is correct to be honest since there are two. But the listed “source” is actually 192.168.1.1, which is my home router’s IP.

15. This web page contains images. Before retrieving each image, does your host issue new DNS queries? Explain your answer.

It does not issue new DNS queries because the data is now stored in my machine’s cache, so it references the cache for the IP address of the website.

**3.5 Internet Control Message Protocol (ICMP)**

In this section we will investigate ICMP protocol. We will explore two applications of this protocol:

1. Ping command to determine whether a host is reachable or not.

2. Traceroute to allow users to determine the route that a packet takes from a local host to a remote host.

 Start up the Wireshark packet analyzer and begin Wireshark packet capture.

 Send ten ping packets to any web-server in the Internet. For example,

[*ping –n 10 www.alcatel.fr*](http://www.alcatel.fr/)

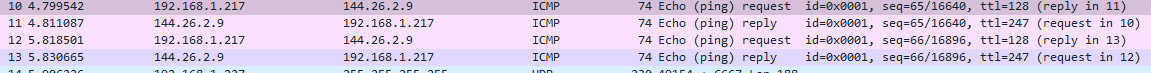
 Stop Wireshark packet capture.

 Examine the structure of ICMP Request and ICMP Reply packets.

16. Why is it that an ICMP packet does not have source and destination port numbers?

It is not actually trying to establish a connection, it just tests if the machine is reachable (it only has network-layer information, not application layer information).

17. Provide a screenshot with ICMP messages.



Let’s now continue the investigation of ICMP protocol by capturing the packets generated by the Traceroute program. This program can be used to figure out the path a packet takes from a source to a destination. Traceroute is implemented in different ways in Unix/Linux and in Windows. In Unix/Linux, the source sends a series of UDP packets to the target destination; in Windows, the source sends a series of ICMP packets to the target destination. Windows uses the command “*tracert*” to run this program, Unix uses “*traceroute*” command.

 Start up the Wireshark packet analyzer and begin Wireshark packet capture.

 Analyze a route to some European web-site. In the terminal (Start->Run->cmd) type:

*tracer*[*t www.alcatel.fr*](http://www.alcatel.fr/)

 Stop Wireshark packet capture.

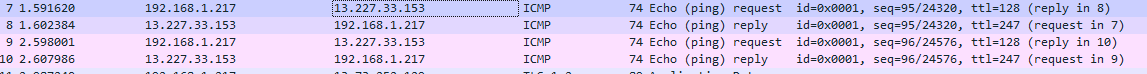
 Examine the structure of ICMP packets

18. Provide a screenshot of the terminal (Start->Run->cmd) with IP addresses of all intermediate nodes between your computer and the selected web-site.



19. Examine the ICMP echo packet in your screenshot. Is this different from the ICMP

ping query packets in the first half of this lab? If yes, how so?



I actually think they are exactly the same, other than the dest/src IP address of the desired website.

20. Examine the ICMP error packet in your screenshot. What is included in its fields?

I did not receive any ICMP error packet.

21. Examine the last three ICMP packets received by the source host. How are these packets different from the ICMP error packets? Why are they different?

Since I did not receive any ICMP error packets, I have nothing to compare my packets to.

**3.6 TCP protocols**

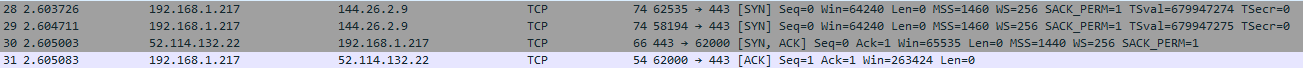
TCP or the Transmission Control Protocol is the dominant transport layer protocol in the Internet. It provides a reliable, in-order stream of data between two applications, even if the applications are running and separated by a network that can drop, reorder or corrupt the packets. TCP provides this reliable data stream by detecting if packets are lost, delayed or changed in transit and retransmits them. Please read a theoretical material about the operation and structure of this protocol.

In this section you can use the same packet capturing statistics, which was used in the sections 3.3 and 3.4. You can also capture new packets starting Wireshark and visiting any web-site in the Internet.

First, we have to analyze the connection establishment process used by TCP. TCP uses three messages to do it. The first message is a TCP segment with no data and the SYN bit in the header set to one. This message is often called the SYN packet. The sequence number in this segment may be set to any value. If the server process is listening and accepts the incoming connection, then it responds with a segment, which is usually called SYNACK packet. Finally the client completes the three-way handshake acknowledges the server SYNACK and confirms that the communications session is established.



22. Provide a screenshot demonstrating the connection establishment between your computer and the remote server. Show the lines containing this information. Explore the structure of these packets.



23. What is the TCP port number used by the remote web-server.

The TCP port number is 443 (https protocol).

24. What is the TCP port number used by your computer

The TCP port number used by my computer is 62535, then 58194, then 62000.

Look at the structure of other TCP packets exchanging the data between your computer and the remote web-server.

25. Choose any captured TCP packet. What is the length of each TCP segment

(including both the header and the payload)? Explain.



The length of each TCP segment in these packets is 1514 bytes (found in the more info underneath as “1514 bytes on wire”).

26. What is the length of data in this TCP segments? Explain.

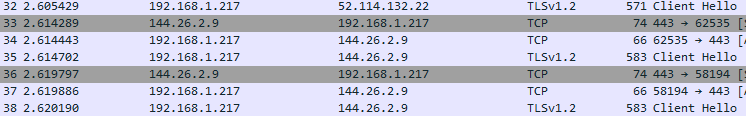
The length of the data itself is 1448 bytes (the actual “len” of the payload).

27. What is the length of a TCP segment carrying the acknowledgments?



The length of the TCP segment carrying the acknowledgments is hard to determine since sometimes it’s 1448 or 1460, and sometimes it’s 0. I assume the correct answer is 0 since when establishing the connection, no actual data is transferred yet.

28. Are there any retransmitted segments in your trace file? What did you check for (in the trace) in order to answer this question?



Yes, there are. I checked for times when there were several of the same type of TCP segments in a row (picture shown of several “Client Hello”s with no server hello).

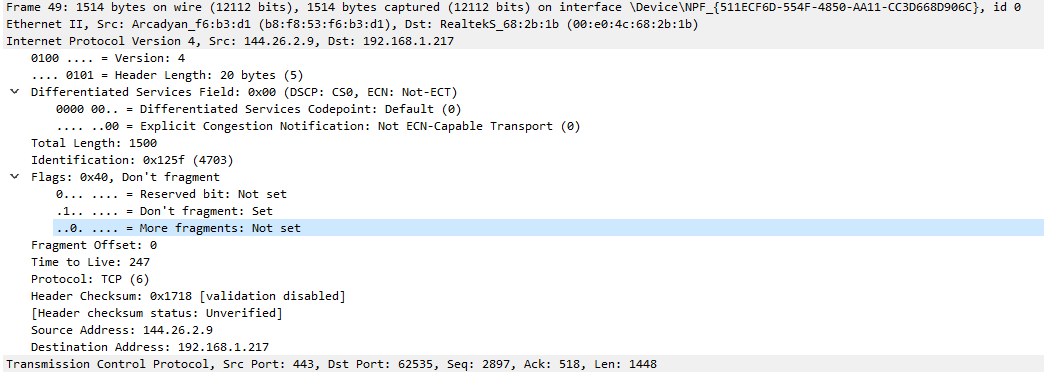
**3.7 IP protocol**

Take any TCP or UDP packet and expand the Internet Protocol part of the packet in the packet details window. Answer the following questions.

29. Within the IP packet header, what is the value in the upper (according to the OSI

model) layer protocol field? Show this information on a screenshot of the packet.

The value in the upper layer protocol field is Ethernet II, which I assume is the link layer.



30. How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes. Show this information on a screenshot of the packet.

Using the picture above, there are 20 bytes in the IP header, and 1448 bytes in the actual datagram since in the transport layer (TCP), the length of the payload is shown as 1448 bytes.

31. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented. Show this information on a screenshot of the packet.

Using the picture above, no it has not been fragmented, which can be seen because in the Flags, it specifically says “don’t fragment,” and under More Fragments it says “not set,” so there are no more fragments to follow.

32. What does “Time to live” field means? Does the value of this field the same for

TCP packets and their ACKs?

The time to live field shows how many hops the packet has left before it is dropped by whatever machine currently holds it. The value of this field is exactly the same for all TCP packets and their ACKS (247 for these packets).