COMP/ELEC 416 : Computer Networks

Project #3 : Part 1

Due: April 28, 2020, 11.59pm (Late submissions will not be accepted.)

Note: This project must be done individually.

Investigating Network Layer Protocols with WireShark

In this project, the aim is to practice with and investigate various **network layer protocol** concepts. You'll experiment and learn more about the network layer concepts related to IP datagram, ICMP, fragmentation, and DHCP. For this purpose, you'll use the **WireShark network protocol analyzer tool**. You have 20 questions to answer individually.

Recall that WireShark is the world's foremost network protocol analyzer, and is the de facto standard across many industries and educational institutions. WireShark allows users to trace the network activity by capturing all the packets that hit to your network interface. It tags the information of each layer by parsing the given byte stream according to the corresponding protocol.

Capturing packets from an execution of traceroute:

In order to generate a trace of IP datagrams, you'll use the traceroute program to send datagrams of different sizes towards some destination, X. Recall that traceroute operates by first sending one or more datagrams with the time-to-live (TTL) field in the IP header set to 1; it then sends a series of one or more datagrams towards the same destination with a TTL value of 2; it then sends a series of datagrams towards the same destination with a TTL value of 3; and so on. Recall that a router must decrement the TTL in each received datagram by 1 (actually, RFC 791 says that the router must decrement the TTL by at least one). If the TTL reaches 0, the router returns an ICMP message (type 11 – TTL-exceeded) to the sending host. As a result of this behavior, a datagram with a TTL of 1 (sent by the host executing traceroute) will cause the router one hop away from the sender to send an ICMP TTL-exceeded message back to the sender; the datagram sent with a TTL of 2 will cause the router two hops away to send an ICMP message back to the sender; the datagram sent with a TTL of 3 will cause the router three hops away to send an ICMP message back to the sender; and so on. In this manner, the host executing traceroute can learn the identities of the routers between itself and destination X by looking at the source IP addresses in the datagrams containing the ICMP TTL-exceeded messages.

First, you are asked to run ***traceroute*** and have it send datagrams of various packet sizes.

• **Windows:** The ***tracert*** program provided with Windows does not allow one to change the size of the **ICMP echo request (ping)** message sent by the **tracert program**. A nicer Windows traceroute program is **pingplotter**, available both in free version and shareware versions at **http://www.pingplotter.com.** Download and install pingplotter, and test it out by performing a few traceroutes to your favorite sites. The size of the ICMP echo request message can be explicitly set in **pingplotter** by selecting the menu item **Edit -> Options -> Default Settings (Engine)** and then filling in the Packet Size field. The default packet size is 56 bytes. Once pingplotter has sent a series of packets with the increasing TTL values, it restarts the sending process again with a TTL of 1, after

waiting Trace Interval amount of time. The value of Trace Interval and the number of intervals can be explicitly set in pingplotter.

• **Linux/Unix:** With the Unix **traceroute** command, the size of the UDP datagram sent towards the destination can be explicitly set by indicating the number of bytes in the datagram; this value is entered in the traceroute command line immediately after the name or address of the destination. For example, to send traceroute datagrams of 2400 bytes towards ku.edu.tr, the command would be: **%traceroute ku.edu.tr 2400**.

Do the following:

- Start up Wireshark
- Select Capture --> Options (see Figure 1)
    - Choose your network interface (Ethernet, wireless, etc.).
    - Uncheck the *promiscuous* packet capturing to decrease the amount of packets captured.
    - Set your filter to "icmp", in order to capture ICMP packets only.
- Click on *Start* button to start capturing packets.

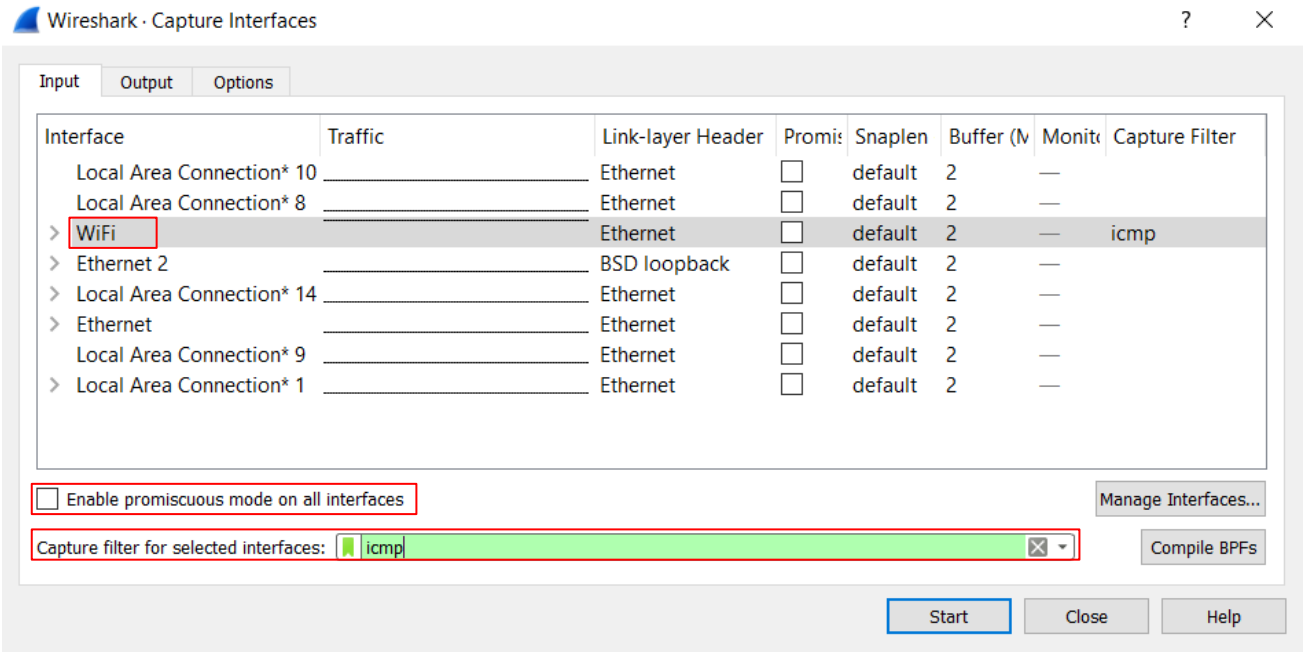

Figure 1- Wireshark Capture->Options menu

- If you are using a **Windows platform**, start **pingplotter** and enter the name of the target destination (**destination address for each student is in file destination.pdf, you should use the address assigned to you.**) in the "Target name" field. Set the **Trace Interval to 1 second**. Select the menu item Edit -> Options -> Default Settings (Engine) and enter a value of 56 in the Packet Size field and then press OK. Then press the Trace button. You should see a pingplotter window that looks something like this:

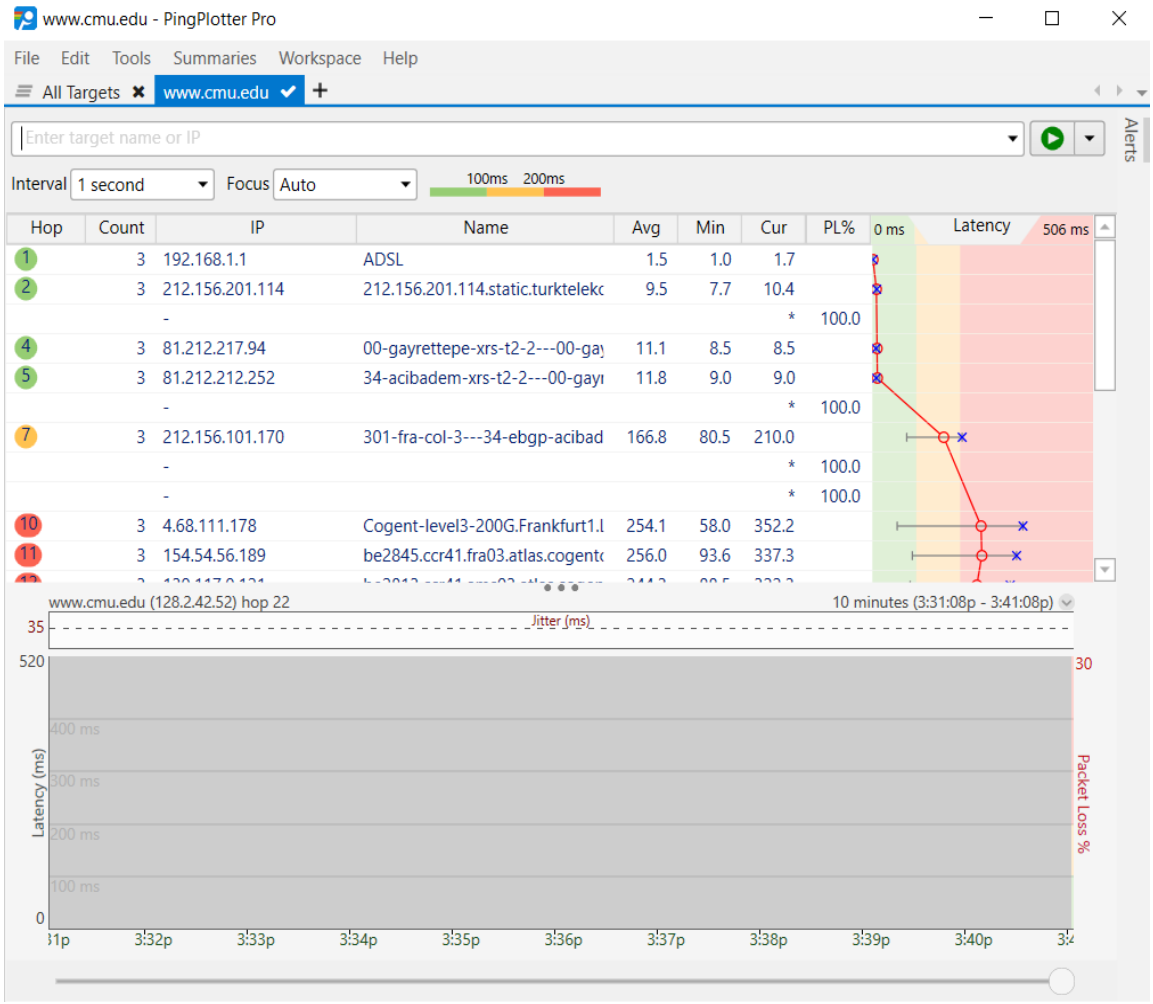

Figure 2- A sample Ping Plotter output

You should see that Wireshark captured the sent and received ICMP packets.

Next, send a set of datagrams with a larger size, by entering a value of 2400 in the Packet Size field and then press OK. Then press the Resume button.

Finally, send a set of datagrams with larger size, by entering a value of 3450 in the Packet Size field and then press OK. Then press the Resume button. Stop Wireshark tracing.

- If you are using a **Unix platform**, enter three traceroute commands for the target destination (**destination address for each student is in file destination.pdf, you should use the address assigned to you.**), one with a packet size of 56 bytes, one with a size of 2400 bytes, and one with a size of 3450 bytes. Stop Wireshark tracing

A look at the captured trace:

In your trace, you should be able to see the series of ICMP Echo Request (in the case of Windows machine) or the UDP segment (in the case of Unix) sent by your computer and the ICMP TTL-exceeded messages returned to your computer by the intermediate routers. In the questions below, we assume you are using a Windows machine; the corresponding questions for the case of a Unix machine should be clear. When answering a question you should provide an output of the packet(s) within the trace that you used to answer the question. Annotate the output to explain your answer. To print a packet, use **File->Print**, choose selected packet only, choose Packet summary line, and select the minimum amount of packet detail that you need to answer the question.

QUESTIONS:

You should explain your answers and provide your WireShark outputs to get credit.

1. Match each part (indicated in Figure 3) with its protocol stack (Physical / Data Link / Network / Transport / Application). (For instance, is Internet Control Message Protocol (ICMP) a physical layer protocol, or a data link layer protocol, or a network layer protocol? Likewise; Internet protocol, Ethernet II, and Frame1?)

2. What is ICMP used for? Select the first ICMP Echo Request message sent by your computer (**you will also use that message to answer question 4, 5 and 6**), and expand the Internet Protocol part (and all subparts) of the packet in the packet details window. (Print the expanded window so that we can evaluate your answers from that output figure.) For this message, draw and fill the fields of IP datagram. (A typical IPv4 datagram format is available in the course book (Computer Networking: A Top-Down Approach, 6/E), Chapter 4)

3. Why is it that an ICMP packet does not have source and destination port numbers? Examine one of the ping request packets sent by your host. What are the ICMP type and code numbers? What other fields does this ICMP packet have? How many bytes are the checksum, sequence number and identifier fields?

4. What is the IP address of your computer? What is the IP address of the nearest router?

5. 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.

6. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented

7. If ICMP sent UDP packets instead (as in Unix/Linux), would the IP protocol number still be 01 for the probe packets? If not, what would it be? 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?

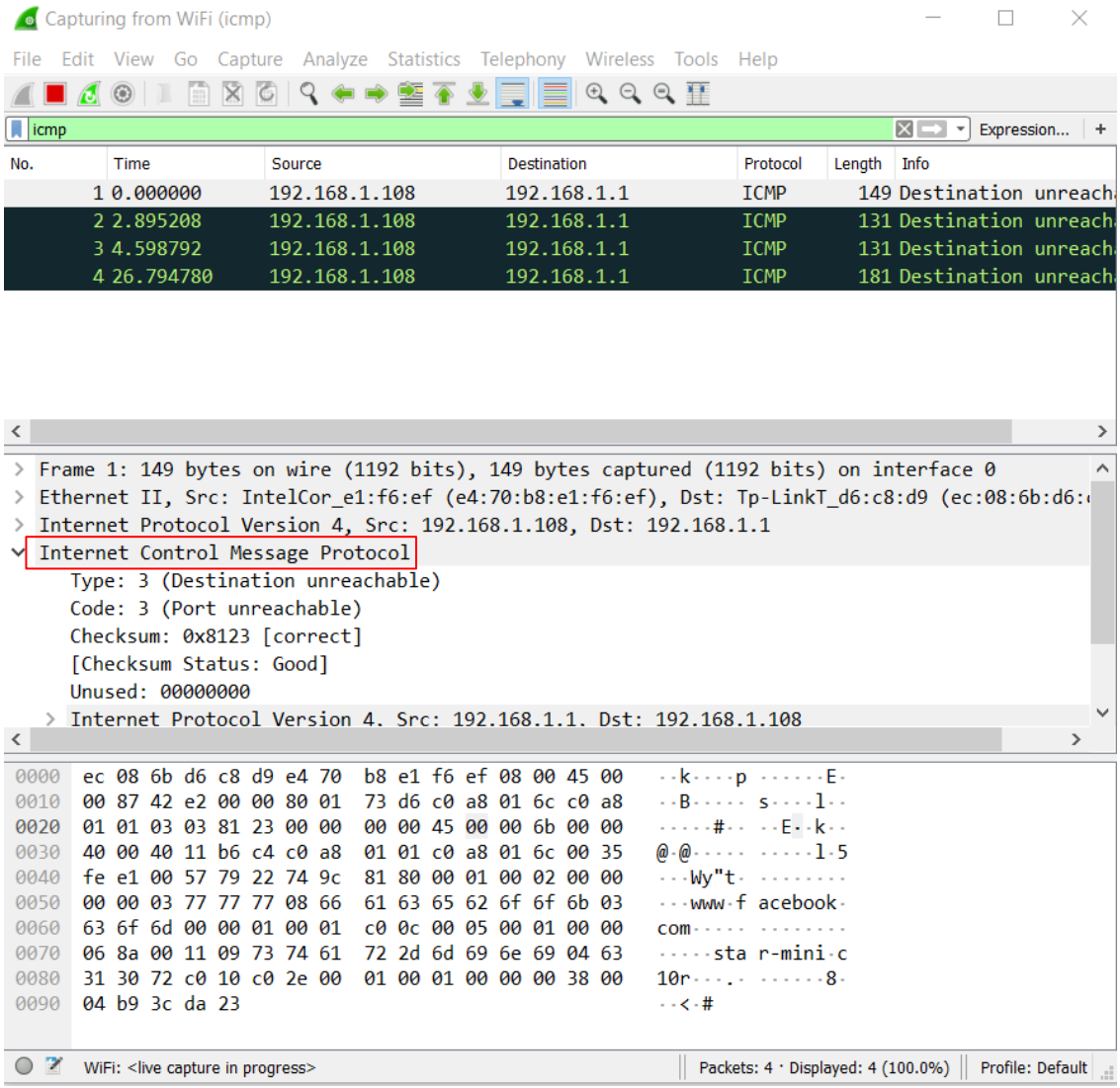

Figure 3 An example of captured packet information

Next, sort the traced packets according to IP source address by clicking on the Source column header; a small downward pointing arrow should appear next to the word ‘Source’. If the arrow points up, click on the Source column header again. Select the first ICMP Echo Request message sent by your computer, and expand the Internet Protocol portion in the “details of selected packet header” window. In the “listing of captured packets” window, you should see all of the subsequent ICMP messages (perhaps with additional interspersed packets sent by other protocols running on your computer) below this first ICMP. Use the down arrow on your keyboard to move through the ICMP messages sent by your computer.

8. Which fields in the IP datagram always change from one datagram to the next within this series of ICMP messages sent by your computer? Which fields stay constant? Explain the reasons.

Next (with the packets still sorted by source address) find the series of ICMP TTL-exceeded replies sent to your computer by the nearest (first hop) router.

9. What is the value in the Identification field and the TTL field? Do these values remain unchanged for all of the ICMP TTL-exceeded replies sent to your computer by the nearest (first hop) router? Explain the reason.

Fragmentation :

Sort the packet listing according to time again by clicking on the Time column.

10. Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size (in pingplotter) to be 2400. Has that message been fragmented across more than one IP datagram?

11. Print out the first fragment of the fragmented IP datagram. What information in the IP header indicates that the datagram been fragmented? What information in the IP header indicates whether this is the first fragment versus a latter fragment? How long is this IP datagram?

12. Print out the second fragment of the fragmented IP datagram. What information in the IP header indicates that this is not the first datagram fragment? Are there more fragments? How can you tell?

13. Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size to 3450. How many fragments were created from the original datagram? What fields change in the IP header among the fragments?

DHCP Experiment:

Recall that DHCP (Section 4.4.2 of the textbook) is used extensively in corporate, university and home-network wired and wireless LANs to dynamically assign IP addresses to hosts (as well as to configure other network configuration information).

In order to observe DHCP in action, you are asked to perform several DHCP-related commands and capture the DHCP messages exchanged as a result of executing these commands. Do the following:

- Begin by opening the Windows Command Prompt application (which can be found in your Accessories folder). Enter command ***"ipconfig /release"***. The executable for ipconfig is in C:\windows\system32. This command releases your current IP address, so that your host's IP address becomes 0.0.0.0.
- Start up the Wireshark packet sniffer, and begin Wireshark packet capture.
- Now go back to the Windows Command Prompt and enter ***"ipconfig /renew"***. This instructs your host to obtain a network configuration, including a new IP address.
- Wait until the ***"ipconfig /renew"*** has terminated. Then enter the same "ipconfig /renew" command again.
- When the second ***"ipconfig /renew"*** terminates, enter the command "ipconfig/release" to release the previously-allocated IP address to your computer.
- Finally, enter ***"ipconfig /renew"*** to again be allocated an IP address for your computer.
- Stop Wireshark packet capture.

Now take a look at the resulting Wireshark window. To see only the DHCP packets, enter into the filter field “bootp”. (DHCP derives from an older protocol called BOOTP. Both BOOTP and DHCP use the same port numbers, 67 and 68. To see DHCP packets in the last version of Wireshark, you need to enter “bootp” and not “dhcp” in the filter.)

Answer the following questions, and make sure that you provide related screenshots with proper discussion.

14. Are DHCP messages sent over UDP or TCP? What is the link-layer (e.g., Ethernet) address of your host?

15. What is the value of the Transaction-ID in each of the first four (Discover/Offer/Request/ ACK) DHCP messages? What are the values of the TransactionID in the second set (Request/ACK) set of DHCP messages? What is the purpose of the Transaction-ID field?

16. A host uses DHCP to obtain an IP address, among other things. But a host’s IP address is not confirmed until the end of the four-message exchange! If the IP address is not set until the end of the four-message exchange, then what values are used in the IP datagrams in the four-message exchange? For each of the four DHCP messages (Discover/Offer/Request/ACK DHCP), indicate the source and destination IP addresses that are carried in the encapsulating IP datagram.

17. Explain the purpose of the router and subnet mask lines in the DHCP offer message.

18. Explain the purpose of the lease time. How long is the lease time in your experiment?

19. What is the purpose of the DHCP release message? Does the DHCP server issue an acknowledgement of receipt of the client’s DHCP request? What would happen if the client’s DHCP release message is lost?

20. Clear the ***bootp*** filter from your Wireshark window. Were any ARP packets sent or received during the DHCP packet-exchange period? If so, explain the purpose of those ARP packets.

Project Deliverables:

- You should submit your project report, in PDF format, that documents and explains all the steps you have performed in order to answer the questions. In your report, include the corresponding Wireshark screenshots to illustrate the steps and provide clear, precise textual descriptions as well.
- You should also submit the saved capture file from Wireshark.

The name of your project file must be your <surname>.pdf

The **report acts as a proof of work to assert your contributions to this project.** Figures in your report should be scaled to be visible and clear enough. All figures should have captions, should be numbered according to their order of appearance in the report, and should be referenced and described clearly in your text. All pages should be numbered. All reports would be analyzed for plagiarism. Please be aware of the KU Statement on Academic Honesty.

Good Luck!