Middle East Technical University

Department of Computer Engineering

CENG 435

Data Communications and Networking

Fall 2020–2021

Socket Programming Take Home Exam 3

Deadline: 2020

1 Preamble

Follow along this instruction sheet and fill the template we have provided on our ODTUClass page. Upload the report (on its own) and the requested `.pcap` files (archived) to ODTUClass, to their respective modules. Please include the screenshots in the correct positions.[1] Each question is worth 10 points.

In this Wireshark assignment we will heavily rely on the `traceroute` program. You can refer to the manual of the program either locally (`$ man traceroute`) or online. Complete this assignment on a Linux distribution, use a virtual machine if you have to.

The following is from the `traceroute` manual.

> This program attempts to trace the route an IP packet would follow to some internet host by launching probe packets with a small **ttl** (time to live) then listening for an **ICMP** "time exceeded" reply from a gateway. We start our probes with a ttl of one and increase by one until we get an ICMP "port unreachable" (or TCP reset), which means we got to the "host", or hit a max (which defaults to 30 hops). Three probes (by default) are sent at each ttl setting and a line is printed showing the ttl, address of the gateway and round trip time of each probe. The address can be followed by additional information when requested. If the probe answers come from different gateways, the address of each responding system will be printed. If there is no response within a certain timeout, an "*" (asterisk) is printed for that probe.

[1] The template *should* place them in the correct place, do not dump every screenshot at the end of the file or they will be ignored. If you need LaTeX typsetting help (after giving your best effort) feel free to reach out

2 Assignment

Close as many programs as you can that connect to the internet to grab a clean capture

First, observe the usual operation of the `traceroute` program; open up Wireshark and capture the traffic generated by the command below, stop the capture after the traffic is finished;[2]

```
$ traceroute metu.edu.tr
```

Question 1

Looking at the *traceroute* output, have you been able to see the *whole path* to the metu.edu.tr? Include a screenshot or include the output of the *traceroute* program, explain your reasoning.

Question 2

Referring to the Wireshark capture and the `traceroute` manual, what is the default method of route tracing?

You can discard the previous capture. Now observe the traffic generated by the following command (you might need root privileges);

```
# traceroute metu.edu.tr –I
```

Question 3

Referring to the Wireshark capture and the `traceroute` manual, what have we changed using the `-I` flag? Why would you get a different route trace/Wireshark capture using this flag? Explain in detail, you can refer to external sources as well.

Pick two university websites, one from Argentina and one from Malaysia. You can use the `dig` command to learn the IP addresses of the university websites you have chosen to ensure that your `traceroute` command is able to reach to the server. *Rule of thumb:* you can try to navigate to the final URL you reach in the `traceroute`. You might see the actual website but reaching the university network is good enough.

Question 4

Write the websites/universities you have chosen alongside their IP addresses that you can reach using the `traceroute` command.

(Bonus) (20 pts) If you have found a university website that you cannot reach using the traceroute commands given above, what's the "closest" you can get to the actual server using different `traceroute` options? Read the manual and experiment with different settings, then write your method down and explain your reasoning.

Start with the website you have chosen from Argentina. Using the `-I` flag for the `traceroute` command, prepare the following command;

```
# traceroute <university_from_argentina> –I 92
```

[2] your ability to answer the following questions is highly dependant on a clean and correct capture, make sure you perfect it first to avoid problems later

Start the Wireshark capture and analyse the traffic. Save this capture as `<student_id>_icmp92.pcap`.

Inspect the first ICMP packet sent by your computer;

Question 5

What is the value of the IPv4 protocol field? Include a screenshot with the relevant line highlighted.

Question 6

How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain your reasoning.

Now sort the fields by source address (click on the column header) so that all the packets that report TTL exceeded are clustered at the bottom of the capture. Pick the topmost such packet;

Question 7

What is the value in the Identification field and the TTL field? Does this value change among other TTL exceeded packets?

Prepare the following command with the website you have chosen and tested from Malaysia.

```
# traceroute <university_from_malaysia> -I 3200
```

Start the Wireshark capture and analyse the traffic. Save this capture as `<student_id>_icmp3200.pcap`

Inspect the first **ICMP Echo request** sent from your machine, this packet should be *fragmented*. Find the first fragment of this IP datagram and answer the questions below;

Question 8

By looking at the packet information (IP Header), how can you tell this datagram has been fragmented? Explain your reasoning and include a screenshot with the relevant part highlighted as well.

Question 9

By looking at the packet information (IP Header) of the first datagram, can you tell how many fragments have been created by the fragmentation? Explain your reasoning and include a screenshot with relevant parts highlighted.

Question 10

Look at the next fragments (datagrams), which fields in the *header* change between the first datagram and the following ones?