Lab 6 Analyzing IP by WireShark

In this lab, we'll investigate the IP protocol, focusing on the IP datagram. We'll do so by analyzing a trace of IP datagrams sent and received by executing the `traceroute` program.

Capturing Packets by Executing the traceroute

If you are using Windows

A nicer Windows `traceroute` program is called *pingplotter*, available both in free version and shareware versions at http://www.pingplotter.com. Download and install *pingplotter*, and test it 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->Packet Options* 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*.

If you are using Linux/Unix/MacOS

With the Unix/MacOS `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 2000 bytes towards gaia.cs.umass.edu, the command would be:

```
%traceroute gaia.cs.umass.edu 2000
```

Do the following:

- Start Wireshark and begin packet.
- If you are using a Windows platform, start the *pingplotter* and enter the name of a target destination in the "Address to Trace Window." Enter 3 in the "# of times to Trace" field, so you don't gather too much data. Select the menu item *Edit >Advanced Options->Packet Options* and enter a value of 56 in the *Packet Size* field and then press OK. Then press the Trace button.
- Send a set of datagrams with a larger length, by selecting *Edit->Advanced Options->Packet Options* and enter a value of 2000 in the *Packet Size* field and then press OK. Then press the Resume button.
- Send a set of datagrams with an even larger length by selecting *Edit>Advanced Options->Packet Options* and enter a value of 3500 in the *Packet Size* field and then press OK. Then press the Resume button.
- Stop Wireshark

If you are using a Unix or Mac platform, enter three `traceroute` commands, one with a length of 56 bytes, one with a length of 2000 bytes, and one with a length of 3500 bytes.

The Captured Trace

If you are using desktops in EPS 254, please open/import the trace file (ip-*trace.pcap*) provided on

D2L. In your trace file, you should see a set of ICMP Echo Requests in the case of Windows machine, or UDP segments in the case of Unix, sent by your computer, and the ICMP TTL-exceeded messages returned to your computer.

In the questions below, we'll assume you are using a Windows machine; the corresponding questions for the case of Unix machine should be clear. If you are using your own tracefile, screenshots should be provided to indicate how to identify the answers.

1. Select the first ICMP Echo Request message sent by your computer, what is the IP address of your computer? (1pt)

2. Within the IP packet header, what is the value in the upper layer protocol field? (1pt)

3. 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. (2pts)

4. Has this IP datagram been fragmented? Justify your answer. (1pt)

Sort the traced packets according to IP source address. 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 below this first ICMP.

5. Which fields in the IP datagram *always* change from one datagram to the next within this series of ICMP messages sent by your computer? (1pt)

6. Which fields stay constant? Which of these fields *must* stay constant? Which fields must change? Justify your answer. (2pts)

7. Describe the pattern you see in the values of the *Identification* field of the IP datagram. (1pt)

Find the series of ICMP TTL-exceeded replies sent to your computer by the first-hop router.

8. What is the value in the *Identification* field and the *TTL* field? (1pt)

9. Do these values remain unchanged for all ICMP TTL-exceeded replies sent to your computer by the first-hop router? Justify your answer. (1pt)

Fragmentation

Sort the packet listing according to time 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 2000. Has that message been fragmented across more than one IP datagram? (1pt)

11. Look at 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? What is the length of this IP datagram? (2pts)

12. Look at 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? Justify your answer. (3pts)

13. What fields change in the IP header between the first and second fragments? (1pt)

Now find the first ICMP Echo Request message that was sent by your computer after you changed the *Packet Size* in *pingplotter* to be 3500.

14. How many fragments were created from the original datagram? (1pt)
15. What fields change in the IP header among the fragments? (1pt)