Analyzing IP protocol using Wireshark

By – Naga Jayadeep Akula
Course – CPE/NIS 654

Traceroute

- Traceroute is a process of tracking the packet route across the network and measure delays of packet across IP network.
- Execution of traceroute

  a.Windows – in command prompt type "tracert"

  b.Linux/Unix – in command prompt type "traceroute"

  In linux you can set the size of datagram by using this command

  "traceroute gaia.cs.umass.edu 2000" where 200 is the size of

  datagram(size should be in bytes).
- Another way to traceroute is use the software "pingplotter"

Pingplotter

- The software can be downloaded from http://www.pingplotter.com
- Pingplotter is used to track the IP packets sent through a network.
- We can choose number of hops of a packet by giving the number.
- Pingplotter gives a easier and visualization of tracing packet across the network.
- It is suggested to use a pingplotter over command prompt as it only gives exclusive details of only the IP packet compared to command promt which gives details of all packets passing through the network

Running the IP protocol

- Start wireshark and begin capture.
- Start pingplotter and enter the IP address of the destination in ”address to trace window”
- Enter 3 in the “number of times to trace” as we don’t want to gather too much data.’
- Define the packet size if have changed it by going to edit->advanced options->packet options
- Stop wireshark tracing.

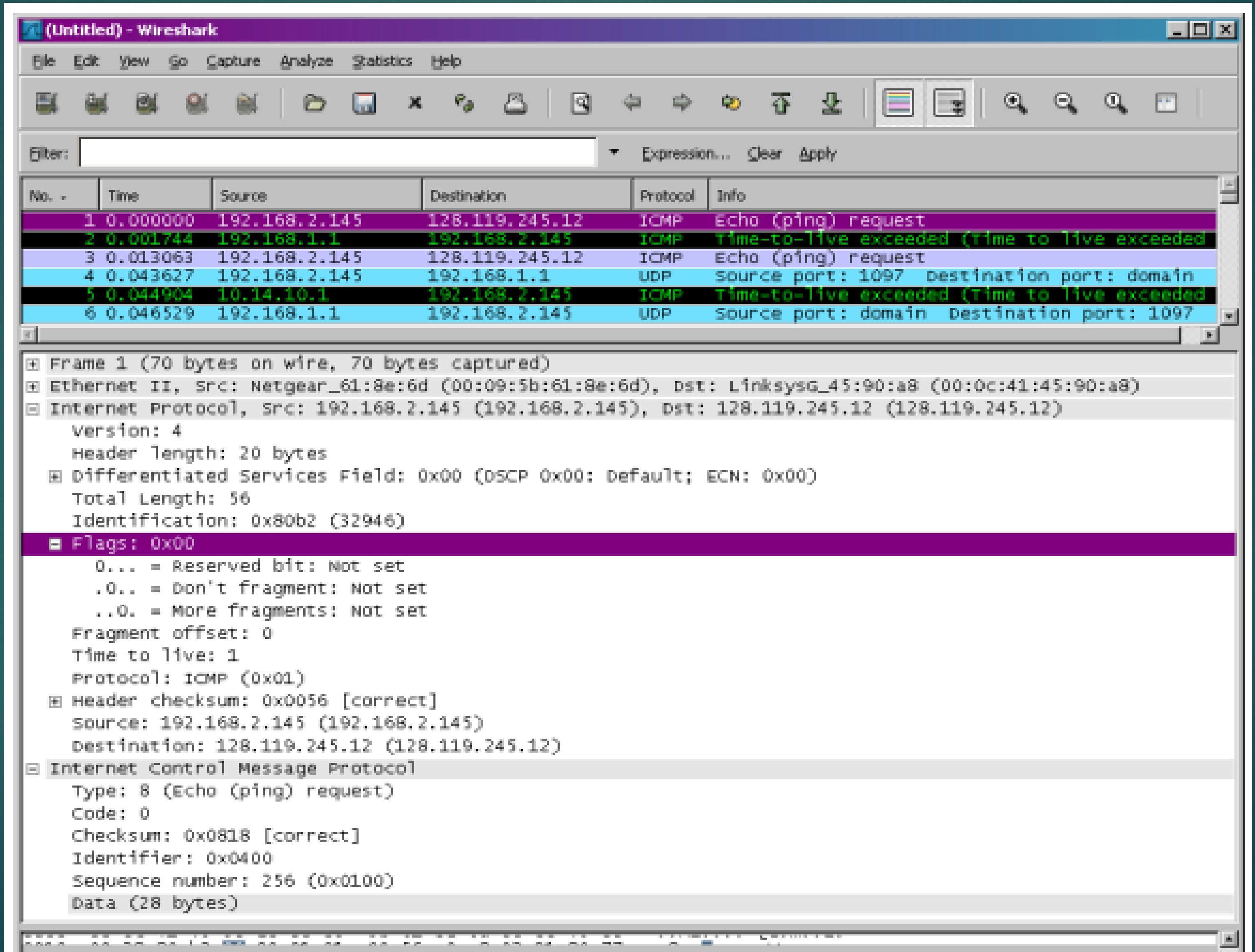
Analysis of IP protocol packet

- The packets for IP protocol are differentiated as ICMP(windows) and UDP(linux) under protocol column in wireshark.
- Now select or filter the IP protocol packets to analyze it.
- When we select a packet, all of its details are given in description box below.
- When we click on the IP protocol, it will create a drop box showing all the information about the packet.

Sample results on IP protocol packets

- We can find the IP address of computer.
- Size of the packet is given in total length.
- Fragmentation of the bits.
- Checksum value.
- Check the TTL of packets.
- Flag information about the packet.
- Number of packets.

Sample workspace


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.2.145	128.119.245.12	ICMP	Echo (ping) request
2	0.001744	192.168.1.1	192.168.2.145	ICMP	Time-to-live exceeded (Time to live exceeded
3	0.013063	192.168.2.145	128.119.245.12	ICMP	Echo (ping) request
4	0.043627	192.168.2.145	192.168.1.1	UDP	Source port: 1097 Destination port: domain
5	0.044904	10.14.10.1	192.168.2.145	ICMP	Time-to-live exceeded (Time to live exceeded
6	0.046529	192.168.1.1	192.168.2.145	UDP	Source port: domain Destination port: 1097

```
⊞ Frame 1 (70 bytes on wire, 70 bytes captured)
⊞ Ethernet II, Src: Netgear_61:8e:6d (00:09:5b:61:8e:6d), Dst: LinksysG_45:90:a8 (00:0c:41:45:90:a8)
⊟ Internet Protocol, Src: 192.168.2.145 (192.168.2.145), Dst: 128.119.245.12 (128.119.245.12)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 56
    Identification: 0x80b2 (32946)
  ⊟ Flags: 0x00
      0... = Reserved bit: Not set
      .0.. = Don't fragment: Not set
      ..0. = More fragments: Not set
    Fragment offset: 0
    Time to live: 1
    Protocol: ICMP (0x01)
  ⊞ Header checksum: 0x0056 [correct]
    Source: 192.168.2.145 (192.168.2.145)
    Destination: 128.119.245.12 (128.119.245.12)
⊟ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x0818 [correct]
    Identifier: 0x0400
    Sequence number: 256 (0x0100)
    Data (28 bytes)
```

Exercise

- By using pingplotter, track the IP packet and answer the following questions for any IP protocol packet.

1.What is the IP address of your computer?

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

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.

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

Exercise

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?

6. Which fields stay constant? Which of the fields must stay constant? Which fields must change? Why?

7. Describe the pattern you see in the values in the Identification field of the IP datagram

8. What is the value in the Identification field and the TTL field?

9. Do these values remain unchanged for all of the ICMP TTL-exceeded replies sent to your computer by the nearest (first hop) router? Why?

Exercise

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? [Note: if you find your packet has not been fragmented, you should download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the ipethereal-trace-1packet trace. If your computer has an Ethernet interface, a packet size of 2000 should cause fragmentation.3]

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?

Exercise

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 the more fragments? How can you tell?

13. What fields change in the IP header between the first and second fragment?

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