- School of Computer Science and Technology, Shandong University

- Lab Report on Computer Networking

Student Number	Name	Class	Lab Title	Period	Date
201900170249	李阳	智能19	IP v8.0	4h	June 1, 2021

Hardware Environment

- Lenovo Legion Y7000P 2020H(Intel Core i7-10750H, 16GB DDR4)
- Windows 10 Home, Chinese Version

Software Environment

- Wireshark-win64-3.44

Purpose

- Analyzing a trace of IP datagrams sent and received by an execution of the traceroute program.
- Investigate the various fields in the IP datagram, and study IP fragmentation in detail.

Experimental Records

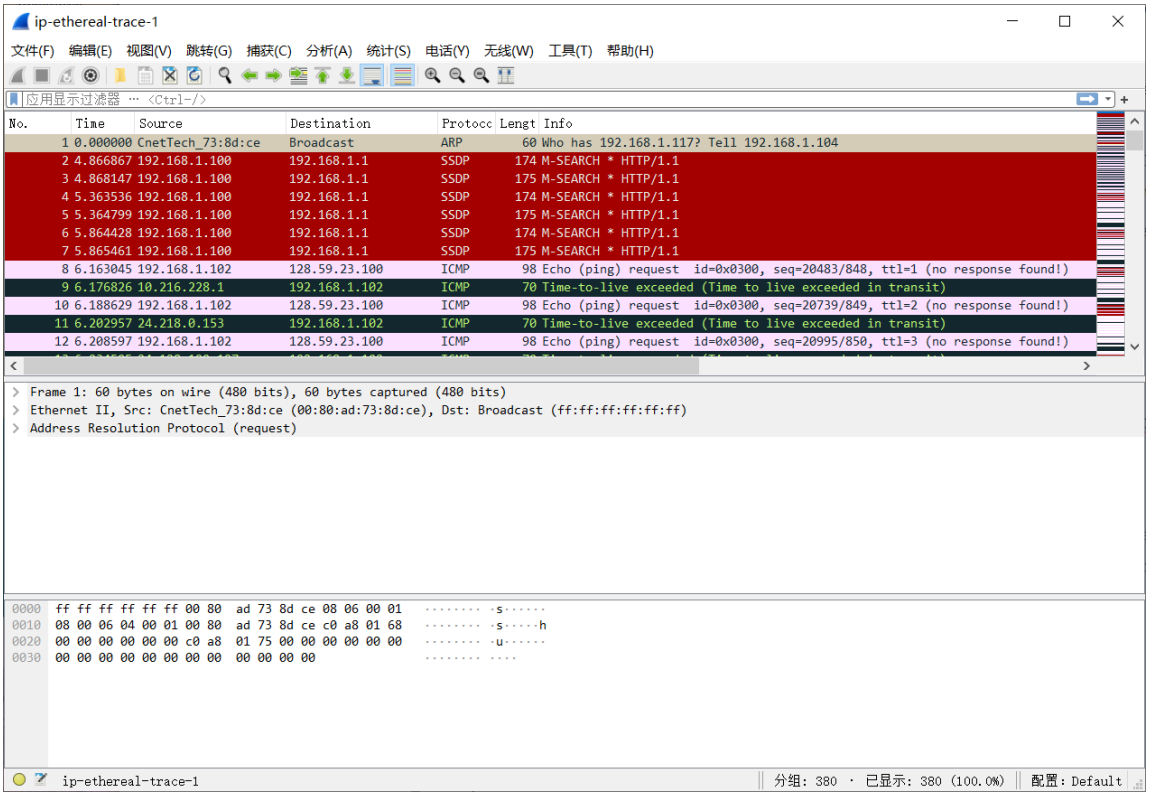
Capturing packets from an execution of traceroute

- Start up Wireshark and begin packet capture and then press OK on the Wireshark Packet Capture Options screen.
- Start up pingplotter and enter the name of a target destination in the "Address to Trace Window".
- Select the menu item Edit->Advanced Options->Packet Options and enter a value of 56 in the Packet Size field and then press OK.
- Press the Trace button, send a set of datagrams with a longer 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 a longer 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 tracing.

A look at the captured trace

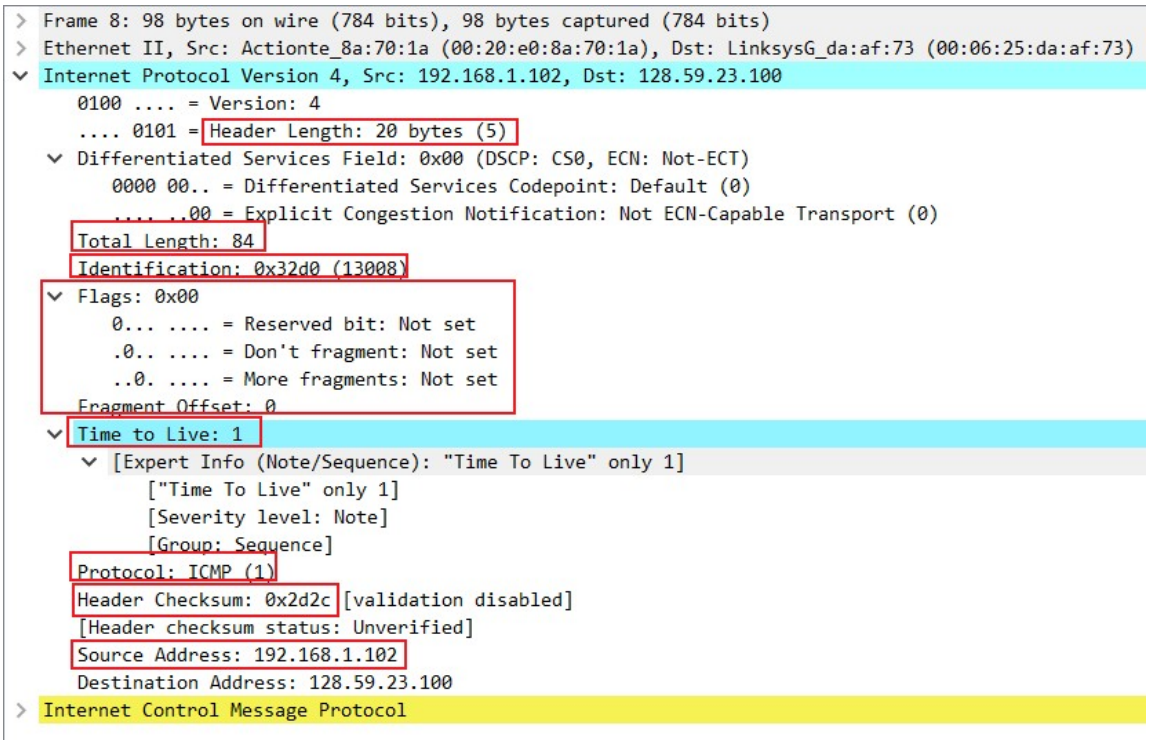
- In the trace, we should be able to see the series of ICMP Echo Request sent by computer and the ICMP TTL-exceeded messages returned to computer by the intermediate routers.



- Sort the traced packets according to IP source address by clicking on the Source column header; Select the first ICMP Echo Request message sent by computer, and expand the Internet Protocol portion in the "details of selected packet header" window.
- Find the series of ICMP TTL-exceeded replies sent to computer by the nearest (first hop) router.

Answer to Questions

1. 192.168.1.102
2. ICMP
3. 20 bytes are in the IP header, the length of the payload of the IP datagram is 84 - 20 = 64 bytes.
4. This IP datagram have NOT been fragmented because the value of "Fragment Offet" is 0.
5. Identification, Time to Live and Header Checksum always change from one datagram to the next within this series of ICMP messages sent by computer.

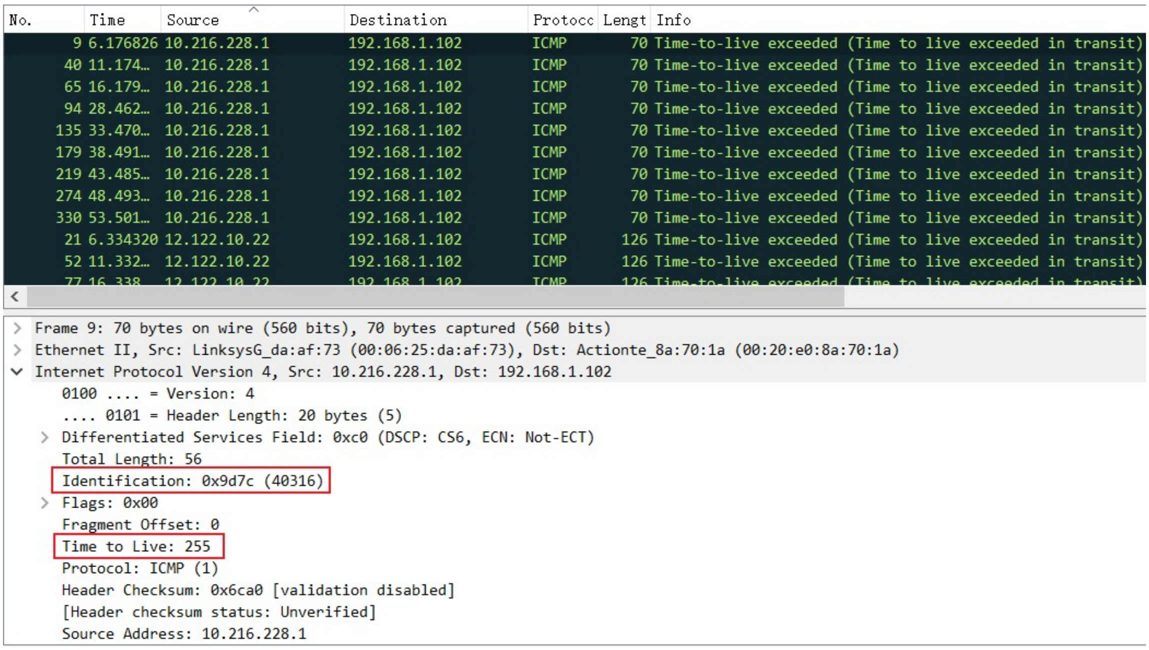


6. Stay constant:
    - Version
    - Header Length
    - Differentiated Services Field
    - Flags
    - Protocol 5
    - Source Address
    - Destination Address

    Must change:
    - Identification
    - Time to Live
    - Checksum

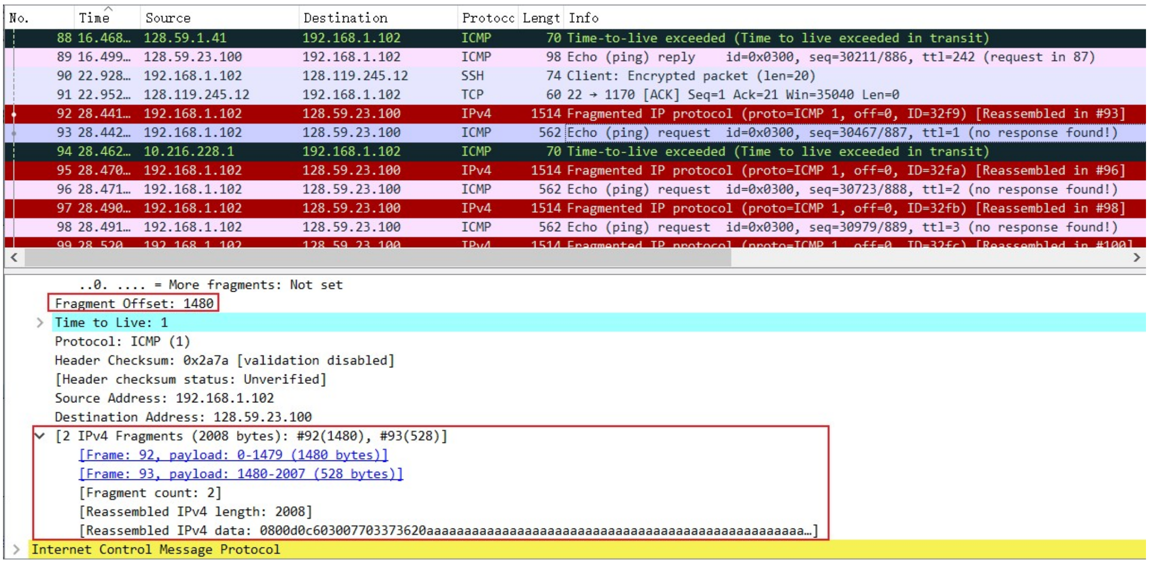
> “
>
> 发送方每发送一个datagram，Identification++；TTL++；Header Checksum随datagram中其他字节变而变

7. Because IP protocol is unreliable, packets cannot be received sequentially and values in the Identification field can identifies whether multiple fragments belong to the same packet.
8. The value in the Identification field is 0x9d7c, the value in the TTL field is 255.



9. These values remain unchanged for all of the ICMP TTL-exceeded replies sent to the computer by the nearest router, because the value of TTL will minus one when the IP packet pass a router.

10. That message been fragmented across two IP datagram.



11. "Flags" and "Fragment Offset" indicate that the datagram been fragmented.

If it is the first fragment, the value of Flags is 1 and the value of Fragment Offset is 0.

If it is the last fragment, the value of Flags is 1 and the value of Fragment Offset isn't 0.

The length of IP data gram is 1500.

> `Flag` 和 `Fragment offset` 可以判断是否被分片：
> 最后一位为1，代表还有“more fragment”。
> 最后一位为0，再看fragment offset是否为0，若也为0，则未被分片；若非0，则代表datagram被分片，且当前包是分片的最后一片。

```
> Frame 92: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)
> Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
v Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 1500
      Identification: 0x32f9 (13049)
    v Flags: 0x20, More fragments
          0... .... = Reserved bit: Not set
          .0.. .... = Don't fragment: Not set
          ..1. .... = More fragments: Set
      Fragment Offset: 0
    > Time to Live: 1
      Protocol: ICMP (1)
```

12. The value of Fragment Offset is 1480 means it isn't the first datagram fragment, and the value of Flags is 0 means there is no fragment later, so it is the last datagram fragment.

```
> Frame 93: 562 bytes on wire (4496 bits), 562 bytes captured (4496 bits)
> Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
v Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 548
      Identification: 0x32f9 (13049)
    v Flags: 0x00
          0... .... = Reserved bit: Not set
          .0.. .... = Don't fragment: Not set
          ..0. .... = More fragments: Not set
      Fragment Offset: 1480
    > Time to Live: 1
      Protocol: ICMP (1)
```

13. Total Length, Flags and Fragment Offset.

14. Three fragments were created from the original datagram.

```
216 43.466…  192.168.1.102   128.59.23.100   IPv4   1514 Fragmented IP protocol (proto=ICMP 1, off=0, ID=3323) [Reassembled in #218]
217 43.466…  192.168.1.102   128.59.23.100   IPv4   1514 Fragmented IP protocol (proto=ICMP 1, off=1480, ID=3323) [Reassembled in #21
218 43.467…  192.168.1.102   128.59.23.100   ICMP    582 Echo (ping) request  id=0x0300, seq=40451/926, ttl=1 (no response found!)
```

15. "Flags" of each fragment is 1, 1, 0.

    "Fragment Offset" of each fragment is 0, 1480, 2960.