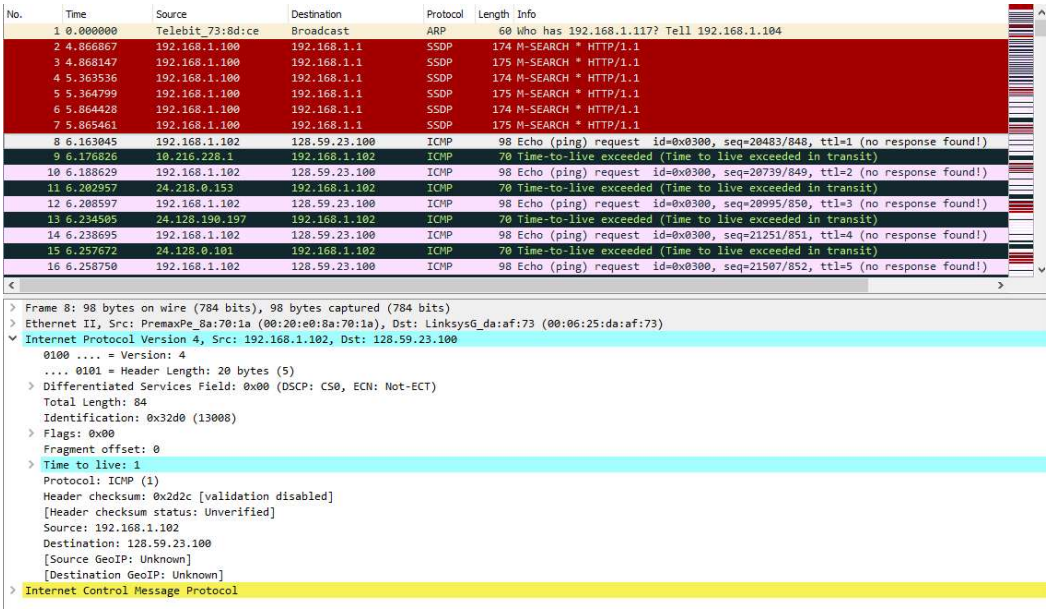


No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Telebit_73:8d:ce	Broadcast	ARP	60	Who has 192.168.1.117? Tell 192.168.1.104
2	4.866867	192.168.1.100	192.168.1.1	SSDP	174	M-SEARCH * HTTP/1.1
3	4.868147	192.168.1.100	192.168.1.1	SSDP	175	M-SEARCH * HTTP/1.1
4	5.363536	192.168.1.100	192.168.1.1	SSDP	174	M-SEARCH * HTTP/1.1
5	5.364799	192.168.1.100	192.168.1.1	SSDP	175	M-SEARCH * HTTP/1.1
6	5.864428	192.168.1.100	192.168.1.1	SSDP	174	M-SEARCH * HTTP/1.1
7	5.865461	192.168.1.100	192.168.1.1	SSDP	175	M-SEARCH * HTTP/1.1
8	6.163045	192.168.1.102	128.59.23.100	ICMP	98	Echo (ping) request id=0x0300, seq=20483/848, ttl=1 (no response found!)
9	6.176826	10.216.228.1	192.168.1.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
10	6.188629	192.168.1.102	128.59.23.100	ICMP	98	Echo (ping) request id=0x0300, seq=20739/849, ttl=2 (no response found!)
11	6.202957	24.218.0.153	192.168.1.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
12	6.208597	192.168.1.102	128.59.23.100	ICMP	98	Echo (ping) request id=0x0300, seq=20995/850, ttl=3 (no response found!)
13	6.234505	24.128.190.197	192.168.1.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
14	6.238695	192.168.1.102	128.59.23.100	ICMP	98	Echo (ping) request id=0x0300, seq=21251/851, ttl=4 (no response found!)
15	6.257672	24.128.0.101	192.168.1.102	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
16	6.258750	192.168.1.102	128.59.23.100	ICMP	98	Echo (ping) request id=0x0300, seq=21507/852, ttl=5 (no response found!)

```
> Frame 8: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: PremaxPe_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: 84
    Identification: 0x32d0 (13008)
  > Flags: 0x00
    Fragment offset: 0
  > Time to live: 1
    Protocol: ICMP (1)
    Header checksum: 0x2d2c [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.1.102
    Destination: 128.59.23.100
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
> Internet Control Message Protocol
```

1) **Select the first ICMP Echo Request message sent by your computer, and expand the Internet Protocol part of the packet in the packet details window. What is the IP Address of your computer?**

```
v Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.59.23.100
```

192.168.1.102

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

```
Protocol: ICMP (1)
```

The value in the upper layer protocol field is ICMP (1)

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.**

```
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: 84
```

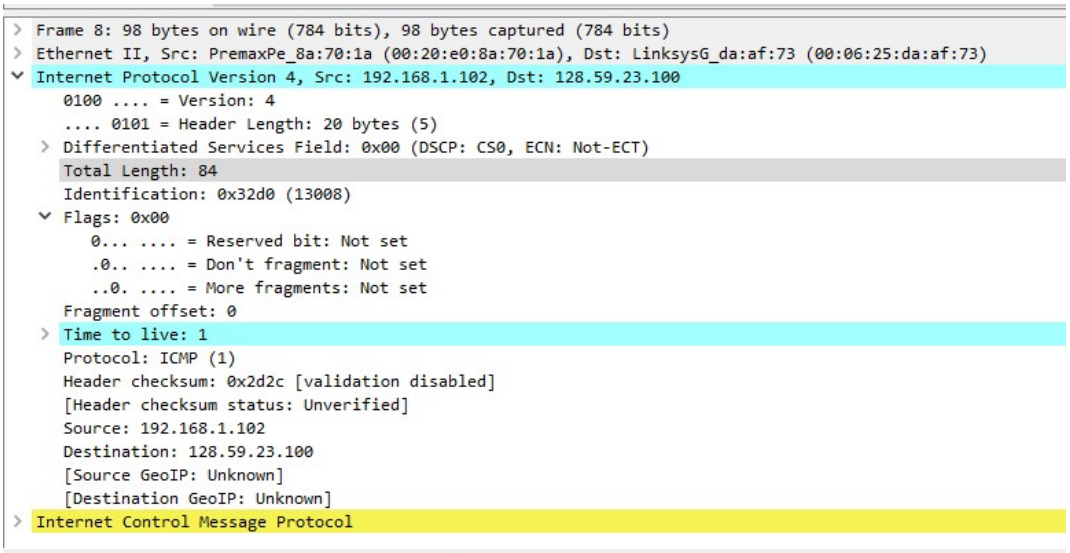
There are 20 bytes in the header, and 64 bytes in the payload of the IP datagram. I determined this because the total length is 84, and 20 of this is the header. 84-20 = 64.

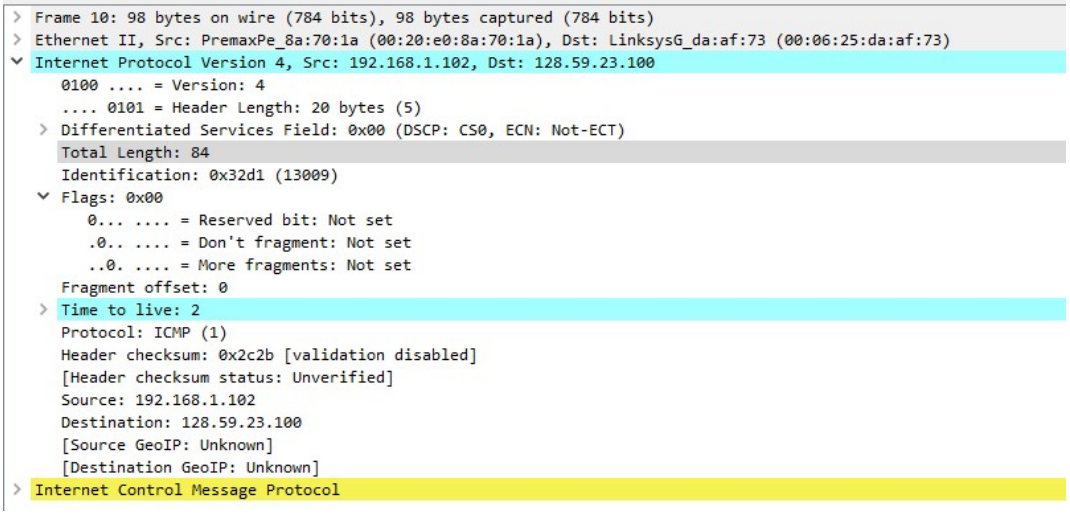
4) **Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.**

```
v Flags: 0x00
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
  Fragment offset: 0
```

This IP datagram has not been fragmented. We can determine this because The 'More Fragments' flag is set to 0.

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?**





The fields that change from IP datagram to the next include:
Identification, Time to live, Header checksum.

6) **Which fields stay constant? Which of the fields must stay constant? Which fields must change? Why?**

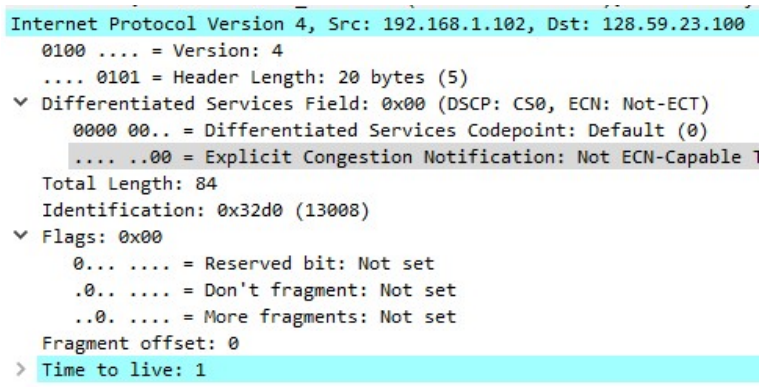
The fields that stay constant include:
Version – can't change version number mid packet
Source IP – the IP address of my PC should not change
Destination IP – the IP address of my destination should not change
Header Length – all header lengths should be the same length
Differentiated Services Field – all of these packets are on ICMP, thus they must not change
Protocol – again, all of these packets are ICMP, thus they should all use ICMP

The fields that *must* stay constant include:
Version – see above
Source IP– see above
Destination IP– see above
Header Length– see above
Differentiated Services Field– see above
Protocol– see above

The fields that must change include:
Identification – every packet must have a unique ID
Time to live – traceroute program will increment each packet in sequence
Header Checksum – we utilize this checksum to ensure our packets are arriving correctly, must change every packet.

7) **Describe the pattern you see in the values in the Identification field of the IP datagram.**
(See above pictures). The Identification field is incrementing by one in between each packet.
0x32d0 -> 0x32d1

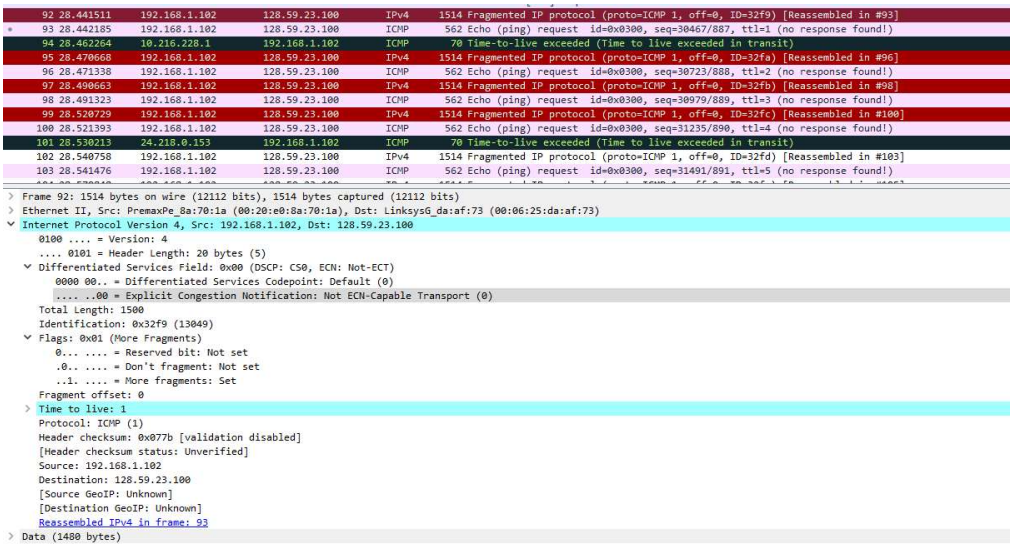
8) **What is the value in the Identification field and the TTL field?**



ID field: 0x32d0 (130008)
TTL field: 1

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?**
The ID field will change with every single packet. ID must be unique.
TTL does not change for a first hop.

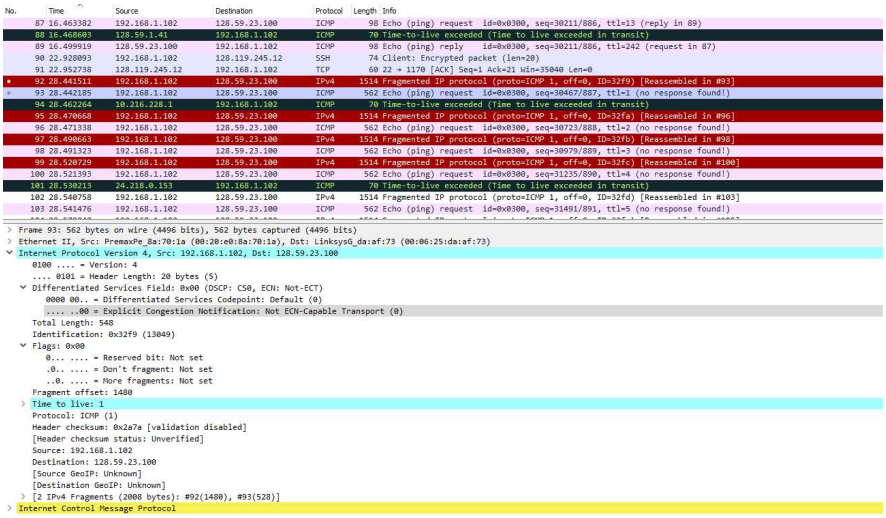
10) **Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size in pingplotter to be 2000. Has that message been fragmented across more than one IP datagram? [Note: if you find your packet has not been fragmented, you should download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the ipethereal-trace-1packet trace. If your computer has an Ethernet interface, a packet size of 2000 should cause fragmentation.]**



Yes, it has been fragmented.

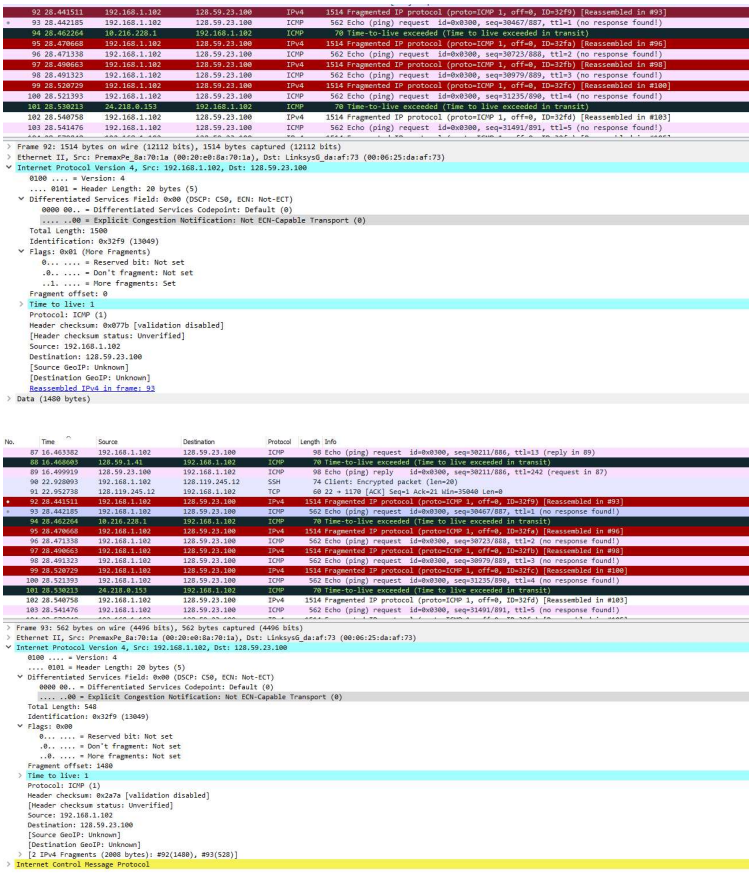
11) **Print out the first fragment of the fragmented IP datagram. What information in the IP header indicates that the datagram been fragmented? What information in the IP header indicates whether this is the first fragment versus a latter fragment? How long is this IP datagram?**
(See above picture).
The flag for More Fragments has been set to 1. This indicates the datagram has been fragmented. We can conclude that this is the first fragment because the Fragment offset value is 0. The total length of this IP datagram is 1500 bytes.

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?**



The fragment offset value is 1480, indicating that this is not the first fragment (which offset would be 0). We can tell that there are no more fragments because now the More Fragments flag value is 0.

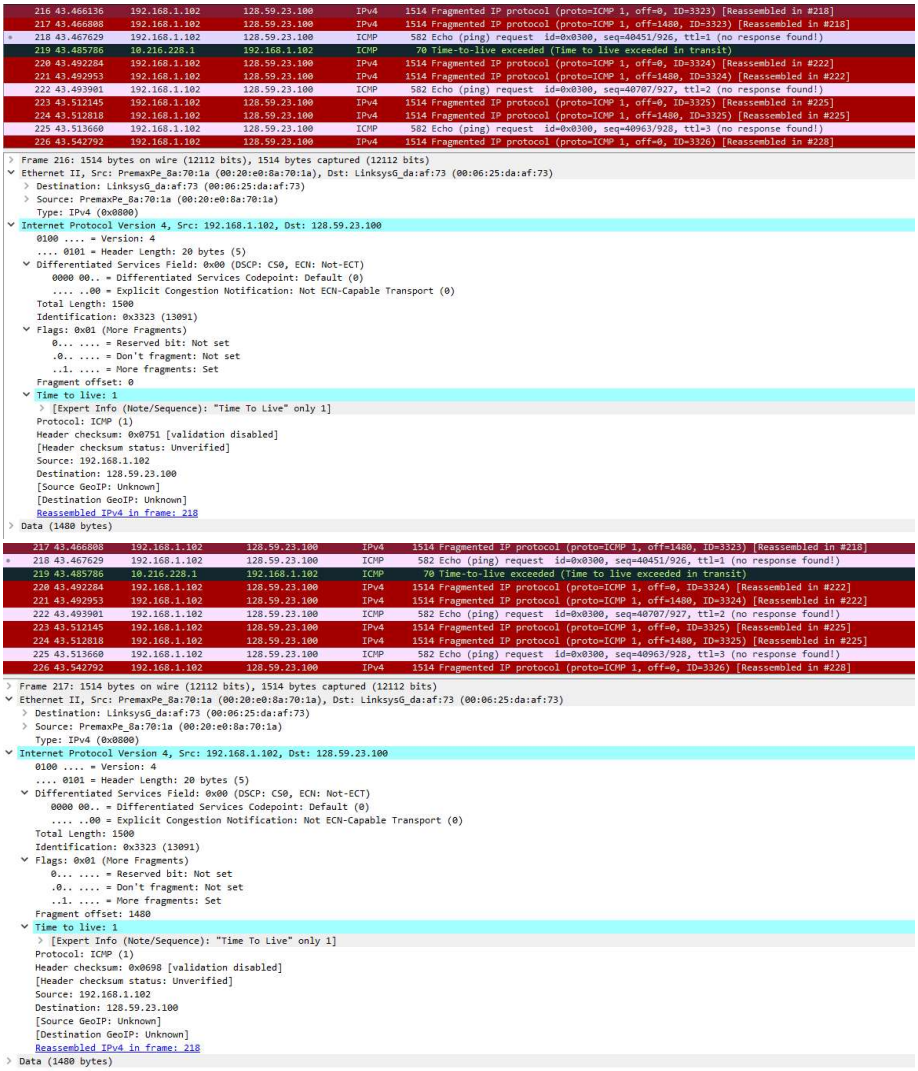
13) **What fields change in the IP header between the first and second fragment?**

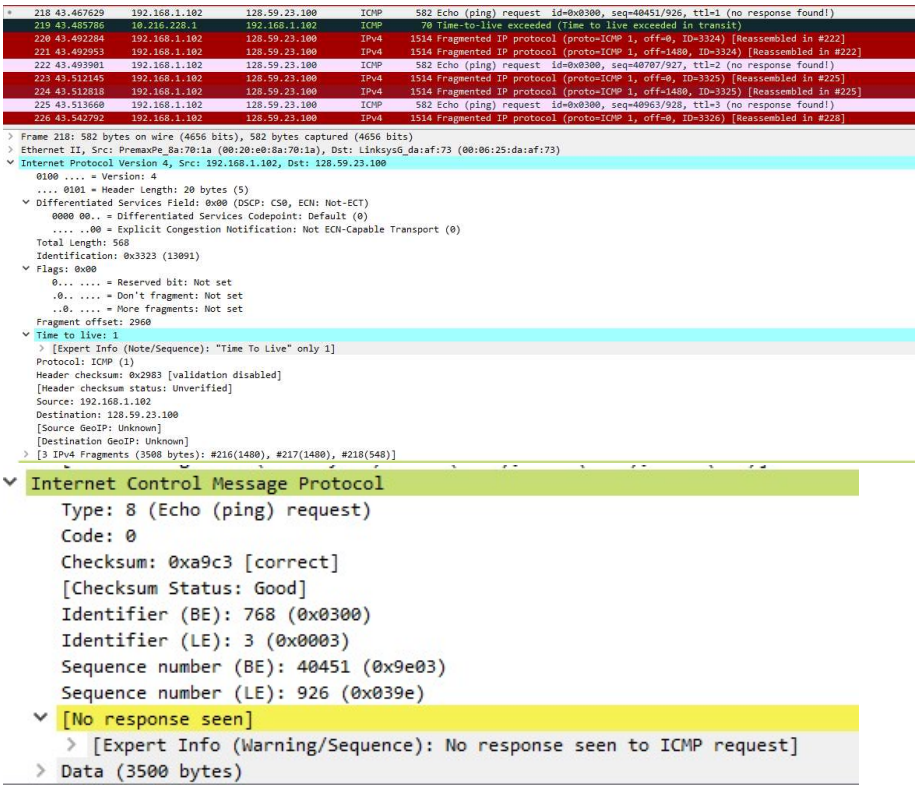


The field values that change in between the two fragments include:
the Total Length field in the Differentiated Services Field section
the Flag, specifically the More Fragments flag
the Fragment offset
and the Header Checksum.

Identification stays the same since this is the same IP Datagram.

14) **How many fragments were created from the original datagram?**





Three fragments were created. On frames 216, 217 and 218.

15) **What fields change in the IP header among the fragments?**

(See above pictures)

Between the first two fragments, Fragment offset changes from 0 to 1480, and between the second and third fragment from 1480 to 2960.

Between the first two fragments, the More Fragments flag is 1, stating that there are more fragments.

Between the second and third fragment, the More Fragments flag will change from 1 to 0, indicating that the third fragment is the last one.