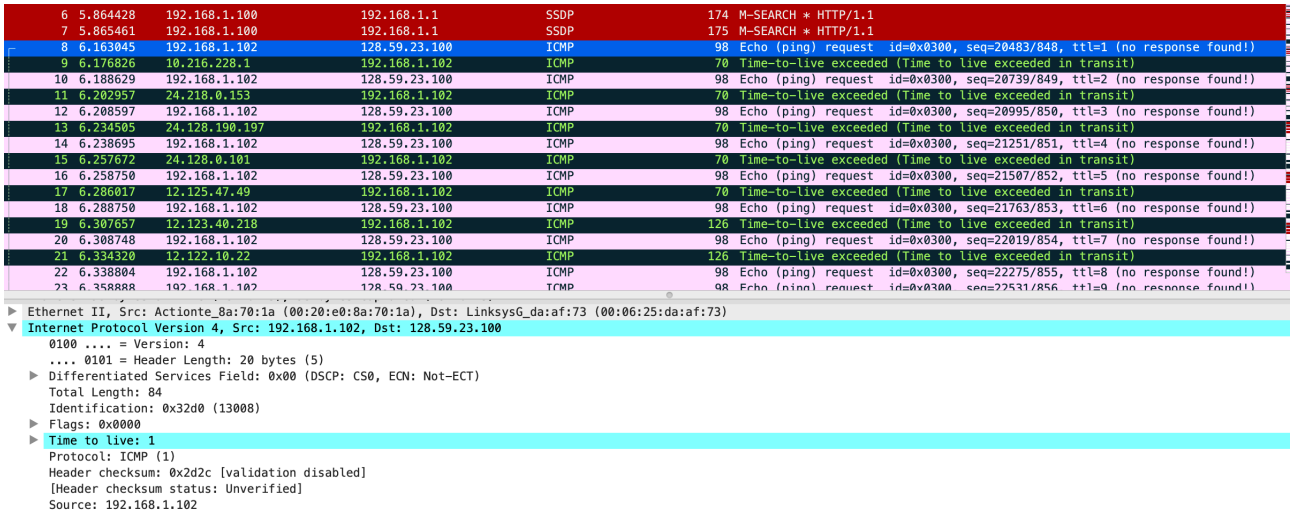
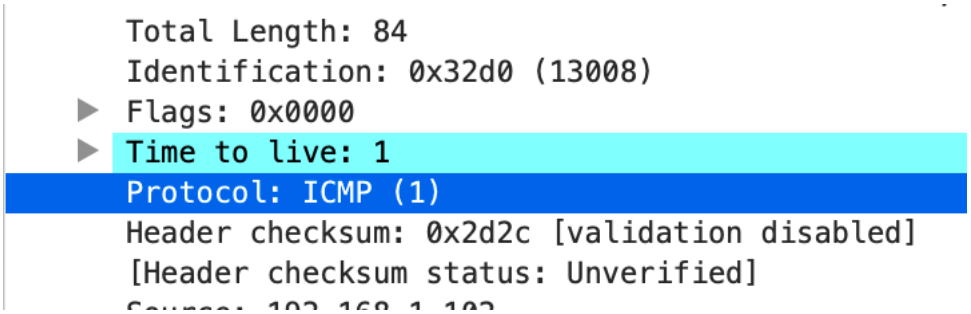
1. Select the first ICMP Echo Request message sent by your computer, and expand the Internet Protocol part of the packet in the packet details window. What is the IP address of your computer?

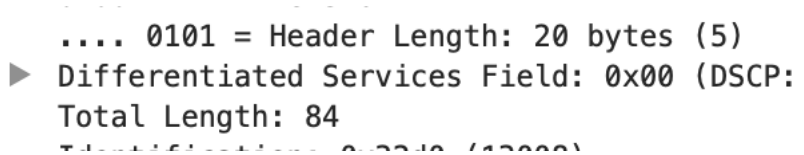



2. Within the IP packet header, what is the value in the upper layer protocol field?

-> ICMP(1)



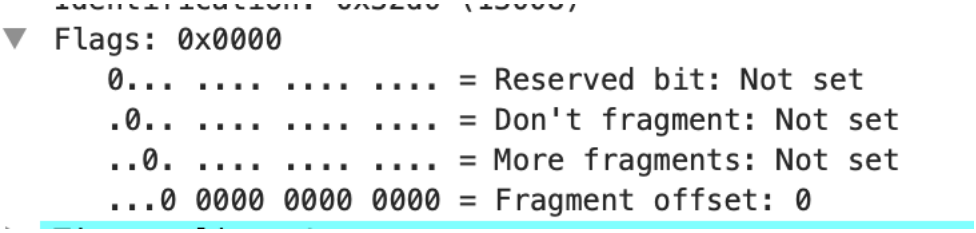

3. How many bytes are in the IP header? How many bytes are in the payload *of the IP datagram*? Explain how you determined the number of payload bytes.




-> 헤더길이는 20바이트이고, 전체길이가 84바이트이기 때문에 payload의 길이는 헤더 바이트의 길이를 뺀 64바이트이다.

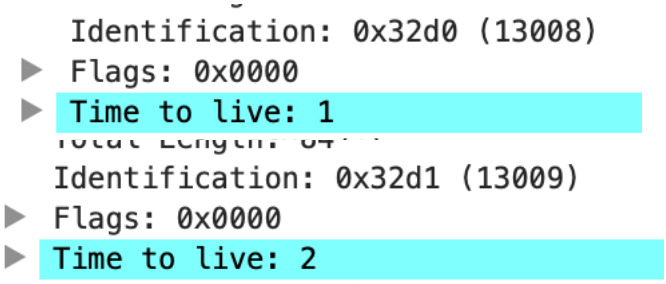
4. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.

-> 파편화 되어 있지 않다. 더 긴 길이의 데이터 그램을 보내게 되면 쪼개질 것이다. 더 긴 길이의 기준은 MTU값이다. 이 값보다 긴 길이의 데이터를 전송하게 되면 fragmented되어 보내진다. (IPv4인 경우에만)




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?

-> 아래 사진에서 보이듯, Identification과 Time to lice 영역의 값들은 항상 변한다.




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

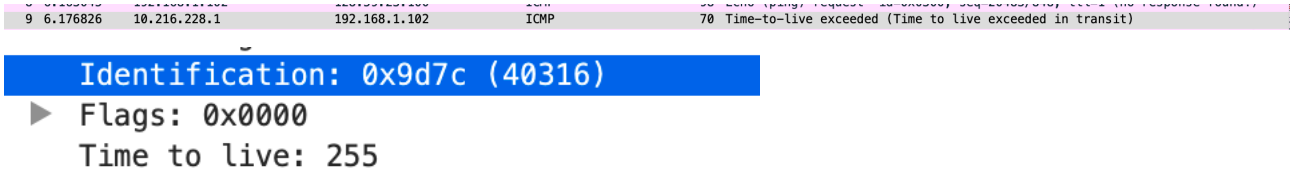
-> 다음 데이터 그램이 전송될 때 앞선 데이터 그램의 필드 값과 같아야 하는 부분은 다음과 같다. >>> version, type of service, upper layer.

그리고 나머지 부분은 변하게 된다.

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

-> 1씩 증가한다.

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?

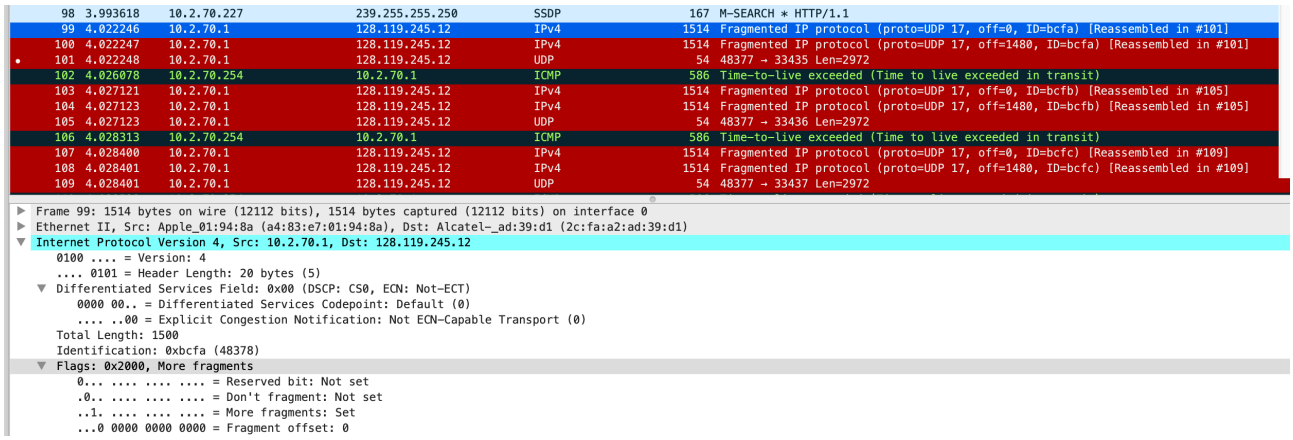
-> 변하지 않는다. TTL값은 1홉 이동할 때 마다 값이 변하는데, 같은 라우터에서 온 응답 메세지 이기때문에 TTL값이 같을 수 밖에 없다.

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 *ip- ethereal-trace-1*packet trace. If your computer has an Ethernet interface, a packet size of 2000 *should* cause fragmentation. ]

-> 단편화 되었다.

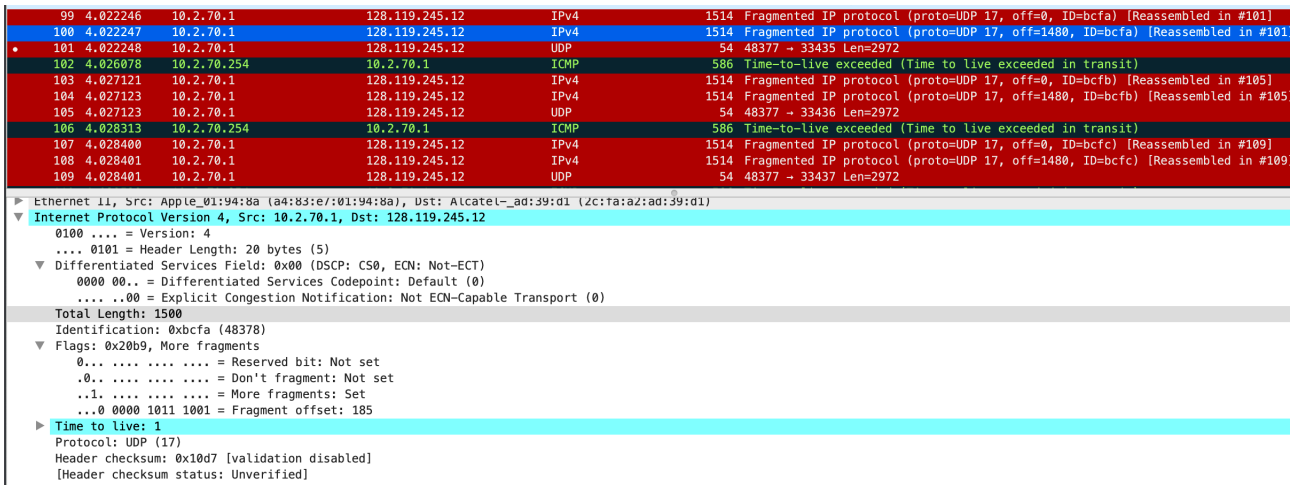
11. Print out the first fragment of the fragmented IP datagram. What information in the IP header indicates that the datagram been fragmented? What information in the IP header indicates whether this is the first fragment versus a latter fragment? How long is this IP datagram?

-> More fragments를 보면 알 수 있다. Fragment offset 값이 0이면 첫번째 조각이다. 또한 전체 길이가 1500바이트이고, 헤더가 20바이트이므로 IP datagram의 길이는 1480임을 알 수 있다.



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?

-> Fragment offset 값이 0이 아니다. More fragments 비트가 set 되어 있기 때문에 조각이 더 있다고 할 수 있다.

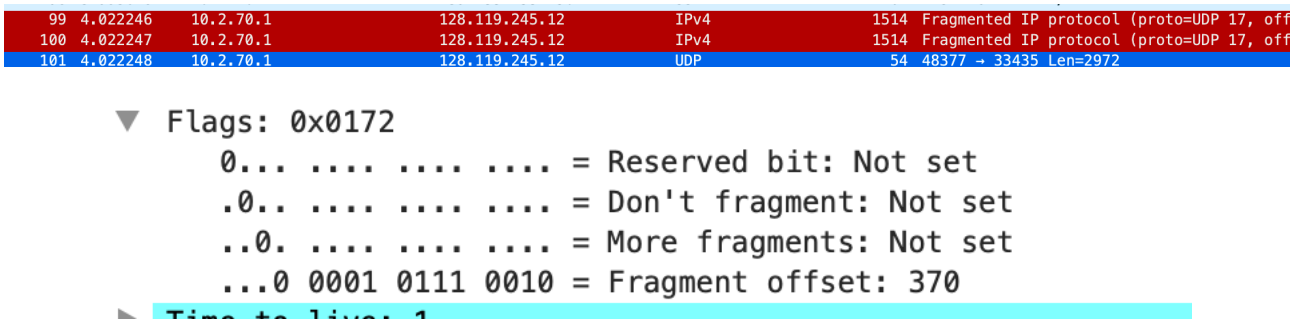



13. What fields change in the IP header between the first and second fragment?

-> Fragment offset 부분이 변경되었다.

14. How many fragments were created from the original datagram?

-> 3개의 조각이 만들어졌다. More fragments가 Not set으로 바뀌는 데이터그램이 마지막 조각이다.




15. What fields change in the IP header among the fragments?

-> Fragment offset 부분과 More fragments부분이 변경되었다.