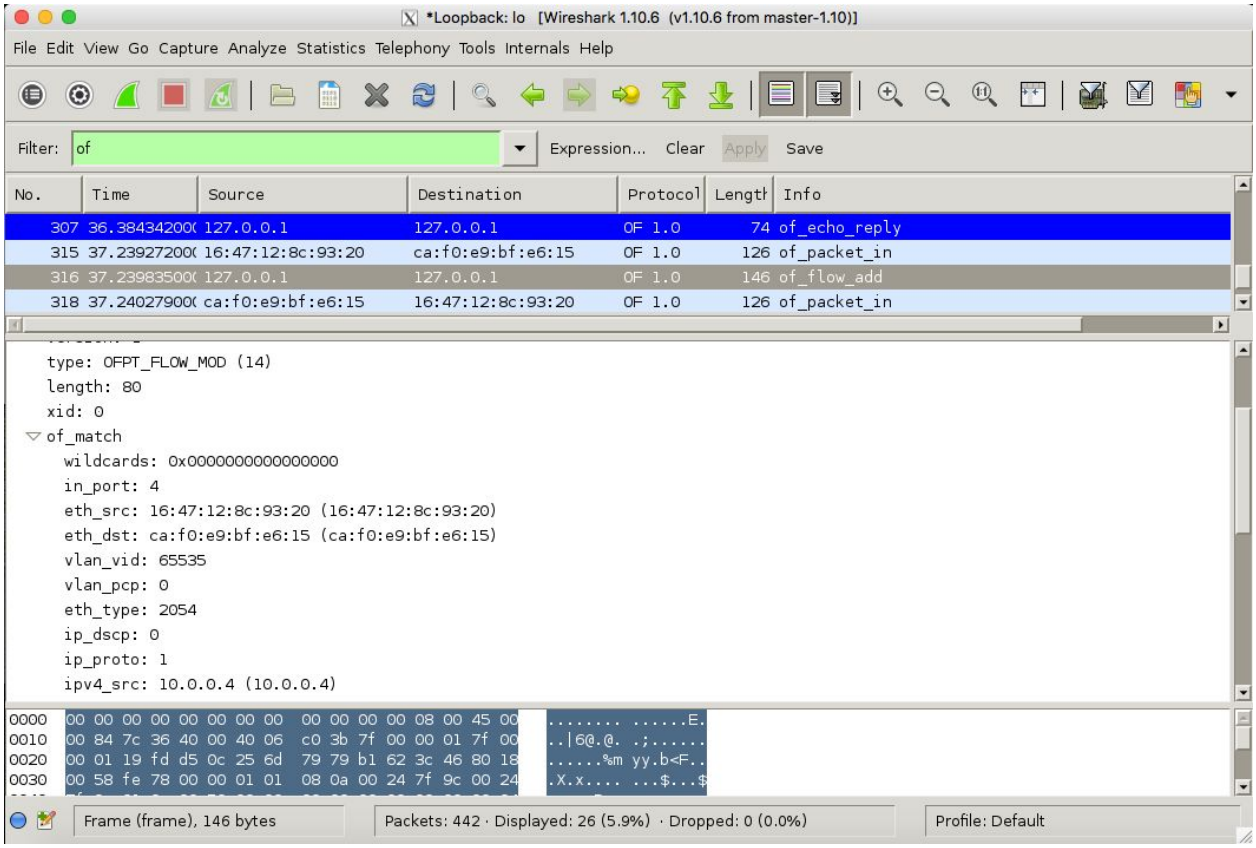
a.) Replicate the example from the video and take a screenshot of wireshark showing a flow modification message.




(my screen wasn't big enough to get the entire packet info for the flow mod, hopefully this is enough)

b.) Show the flow table on switch s1. (Hint: Remember how from the walkthrough.) There are separate entries for flows from h1 and from h4. Why is there not a single bi-directional flow table entry instead.

Because the flows might expire later. If there is a single flow between the two, and no packets that match the flow table entry are coming from h1 for example, then the flow entry would be deleted. But then, if h4 tried to send a packet to h1 that matched the previous flow table entry, there would not be an entry to match, and it would take longer for the packet to reach h1. Therefore, it is better in this scenario to have two separate flows for each host, so that if one host's flow entry expires, the other host still has an entry.

c.) After how long and under what conditions do the flow table entries at the switch expire?

According to Wireshark, the idle timeout is 60 seconds. So, after 60 seconds, a flow entry is deleted from the flow table and the hardware if no packets match the entry. Since the hard timeout is set to 0, it is not initialized and will not be used in this situation.

d.) After the flow table entries have expired run:

```
Mininet> h1 ping -c5 h4
```

Why do the latencies tend to go down over time?

The latencies go down because the packets are being sent before any flow table entries are deleted. Since the destination of the packets matches entries in the flow table, the controller already knows where to send the packets by looking at the flow table. This allows it to send packets faster since it already knows where they go.