**COMPUTER NETWORK**

**LAB ASSIGNMENT 4**

**Submitted By:**

**Name: AKSHAT SAHU**

**Roll No: U22CS034**

**Branch: CSE**

**Semester: 4th Sem**

**Division : A**

Department of Computer Science and Engineering



**SV NATIONAL INSTITUTE OF TECHNOLOGY**

**SURAT**

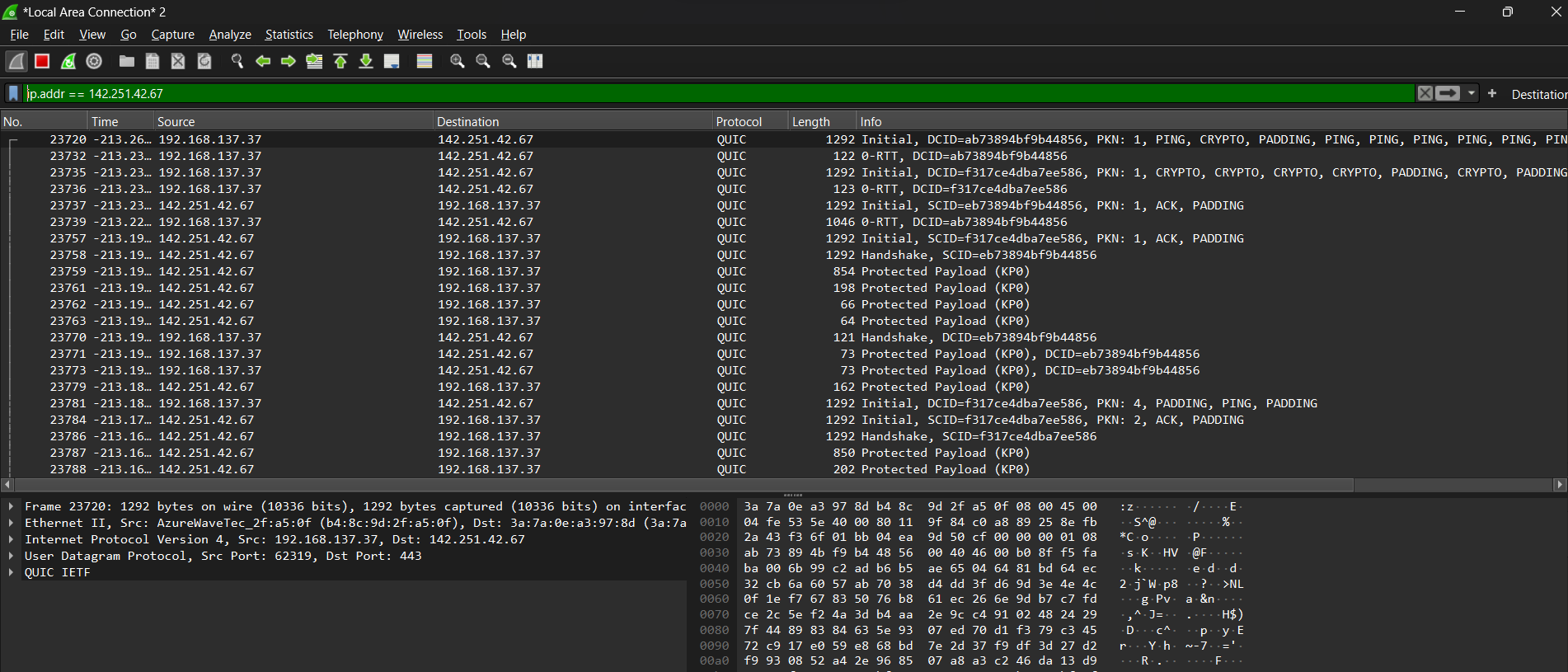
2024

Wireshark is a popular network protocol analyzer that allows you to capture and inspect data traveling back and forth on a network in real-time. Filters in Wireshark help you focus on specific packets of interest and analyze network traffic more effectively. Here are 20 Wireshark filters and their applications:

1. ip.addr == x.x.x.x

- Application: Filters packets based on source or destination IP address.

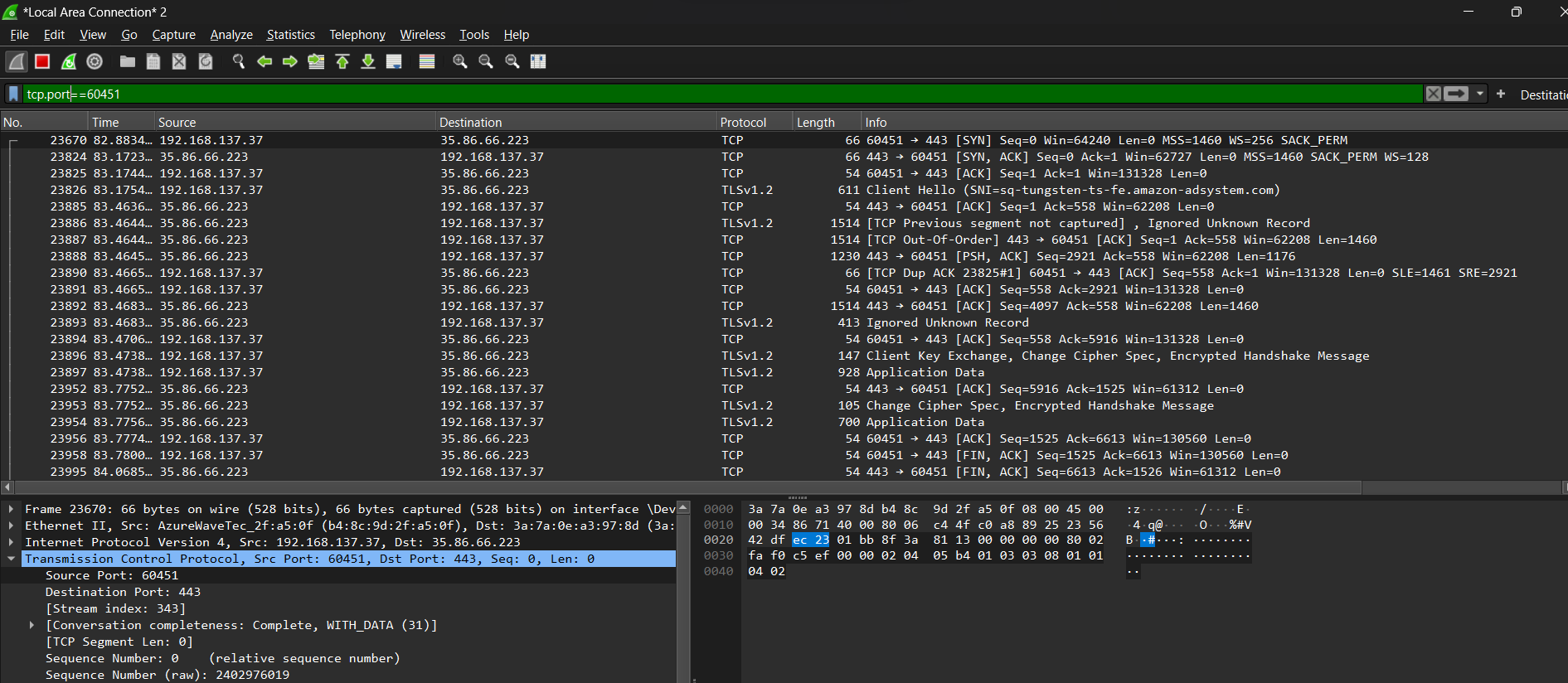
- Example: `ip.addr == 142.251.42.67`



2. tcp.port == xx

- Application: Filters TCP packets based on source or destination port number.

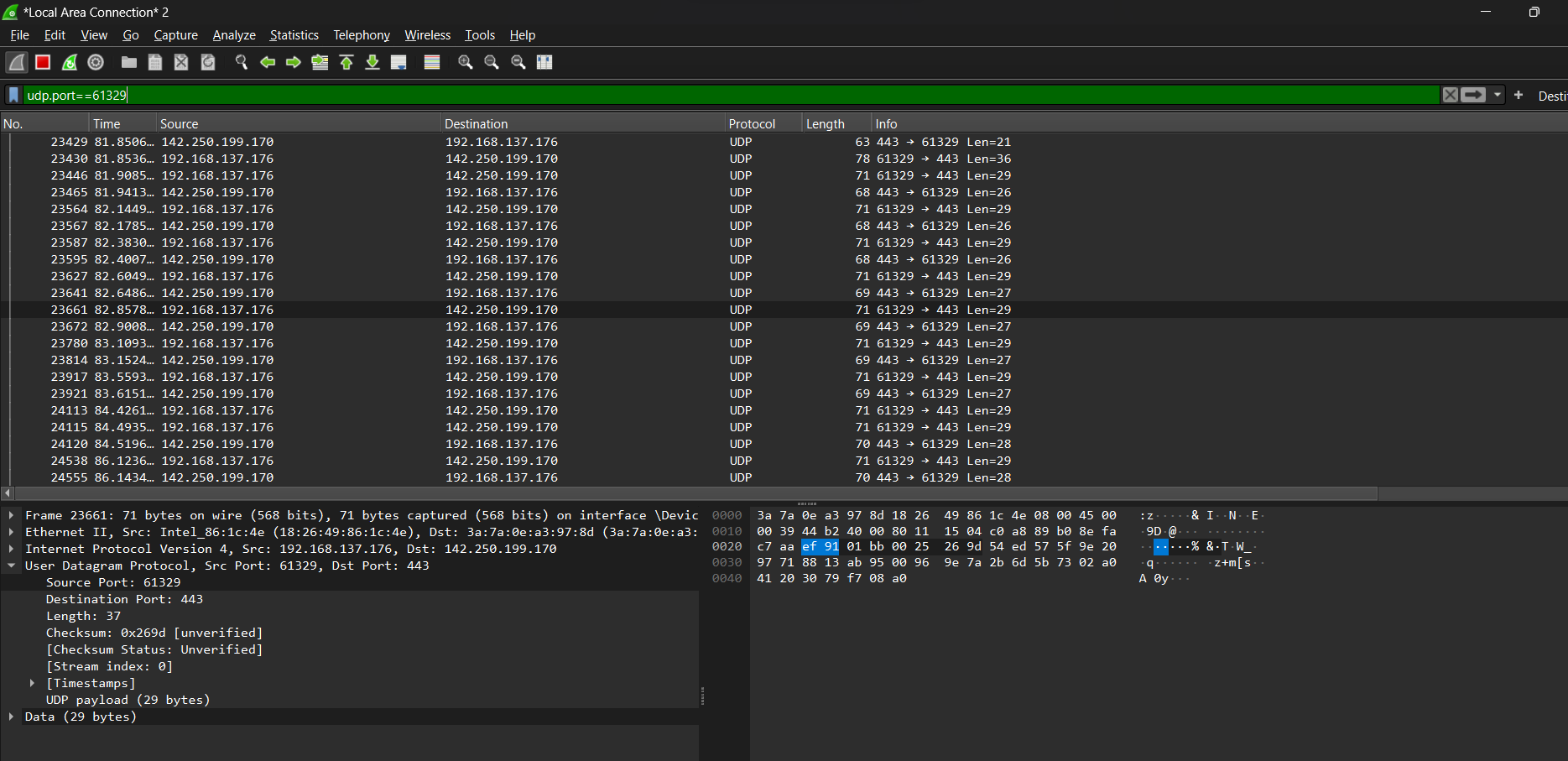
- Example: `tcp.port == 80`



3. udp.port == xx

- Application: Filters UDP packets based on source or destination port number.

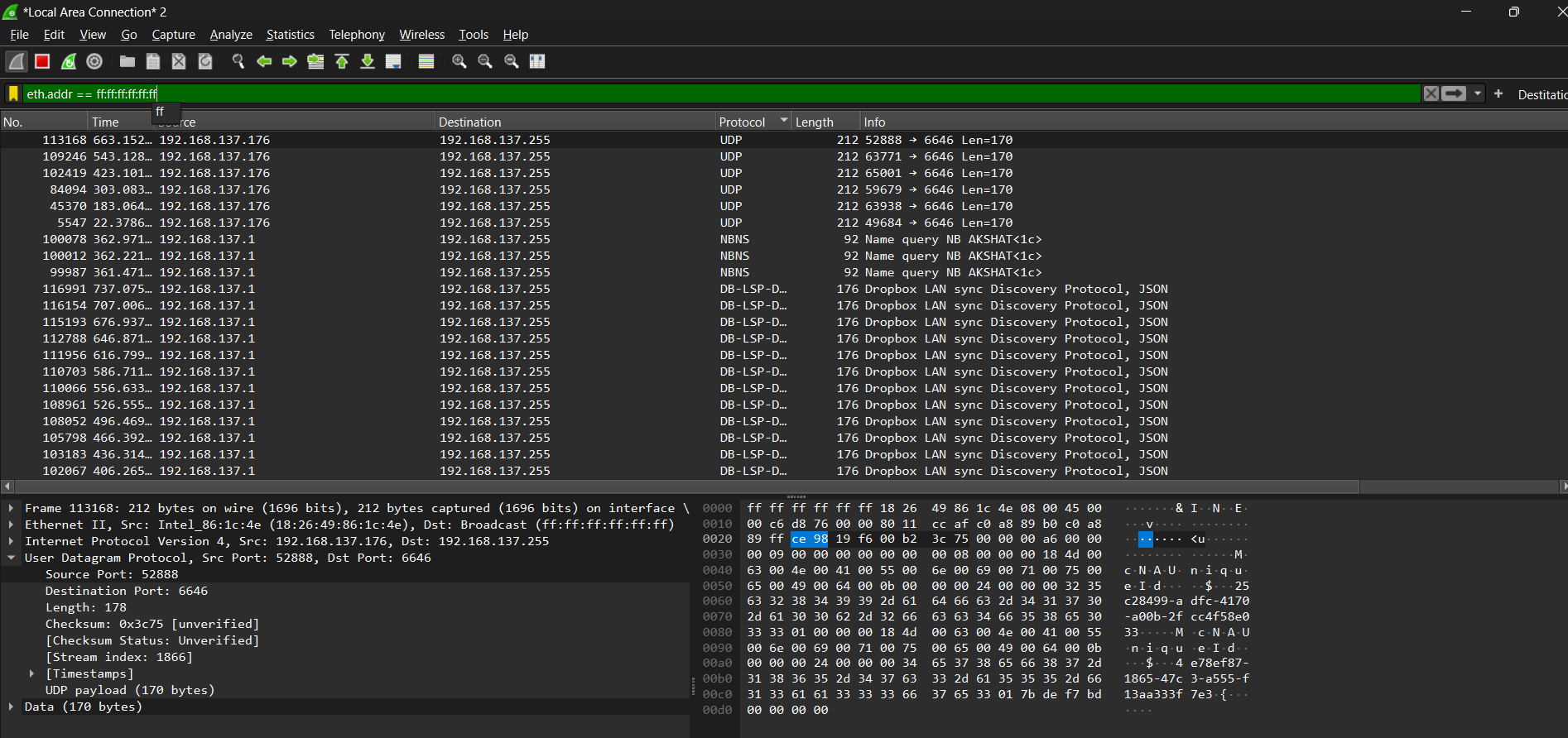
- Example: `udp.port == 53`



4. eth.addr == xx:xx:xx:xx:xx:xx

- Application: Filters packets based on Ethernet MAC address.

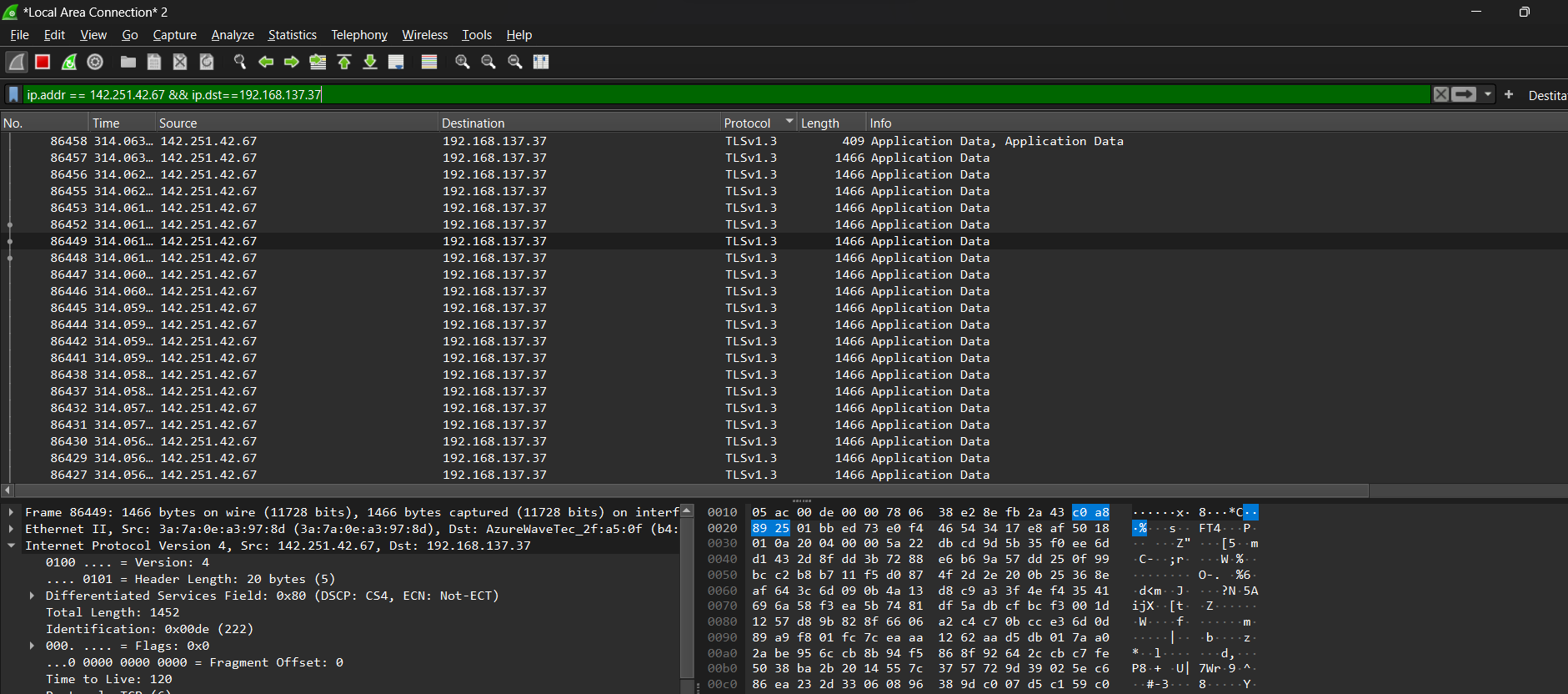
- Example: `eth.addr == 00:1a:2b:3c:4d:5e`



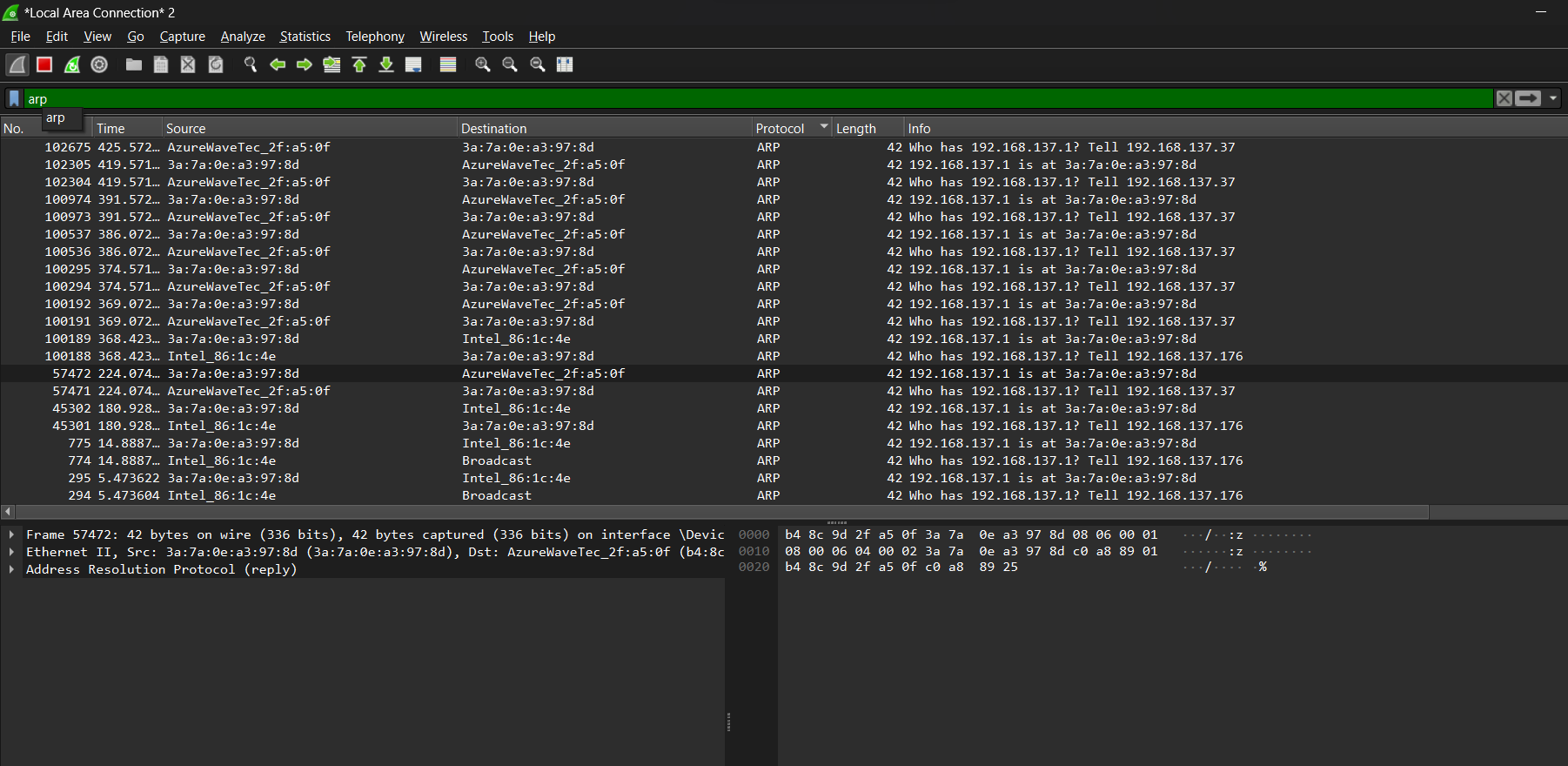
5. ip.src == x.x.x.x && ip.dst == y.y.y.y

- Application: Filters packets based on both source and destination IP addresses.

- Example: `ip.src == 192.168.1.1 && ip.dst == 192.168.1.2`

****

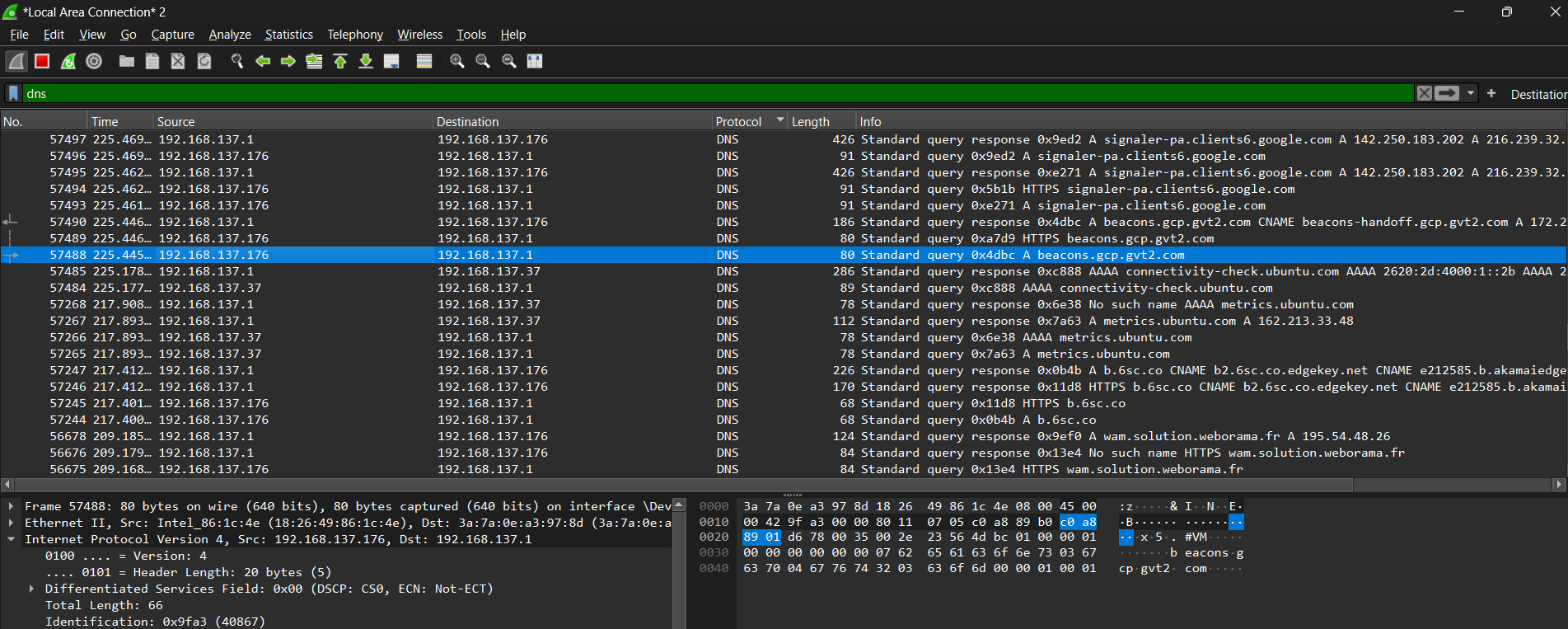
6.arp



7. dns

- Application: Filters DNS protocol packets.

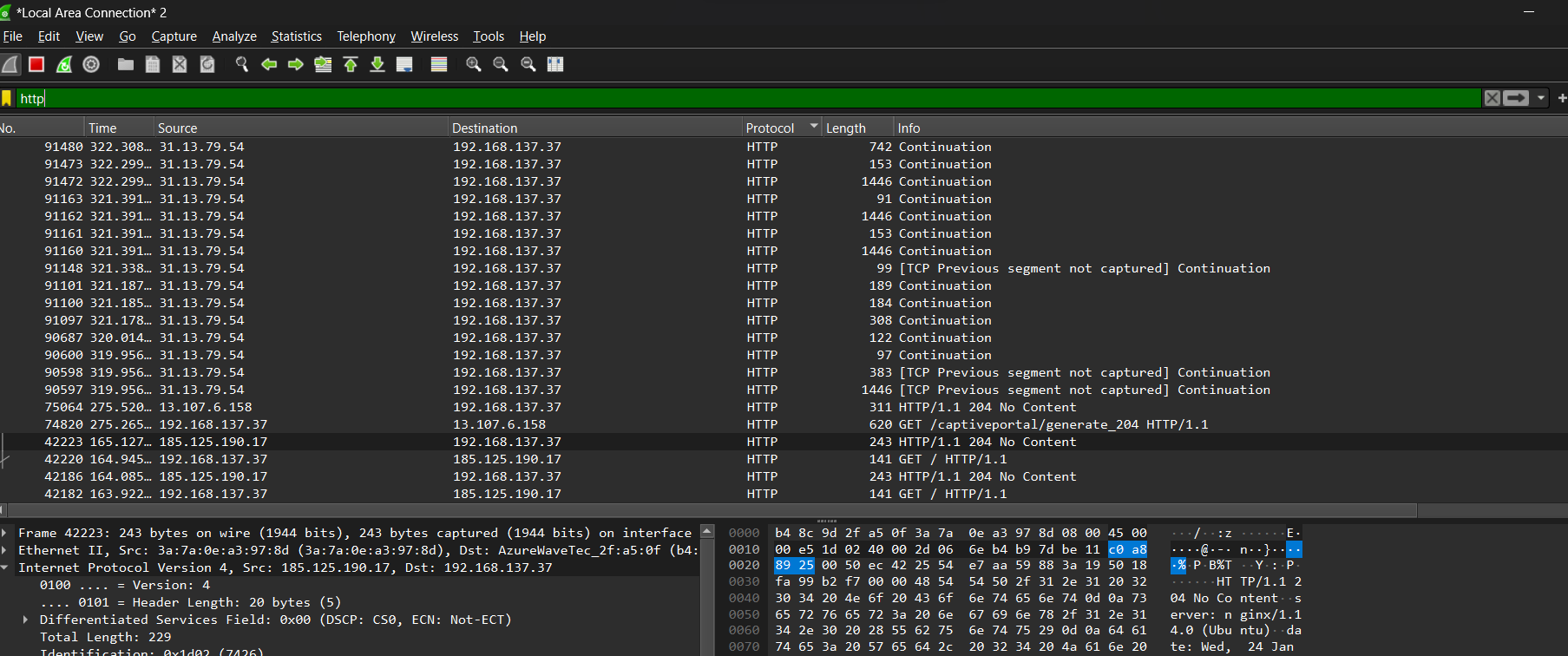
- Example: `dns`



8. http

- Application: Filters HTTP protocol packets.

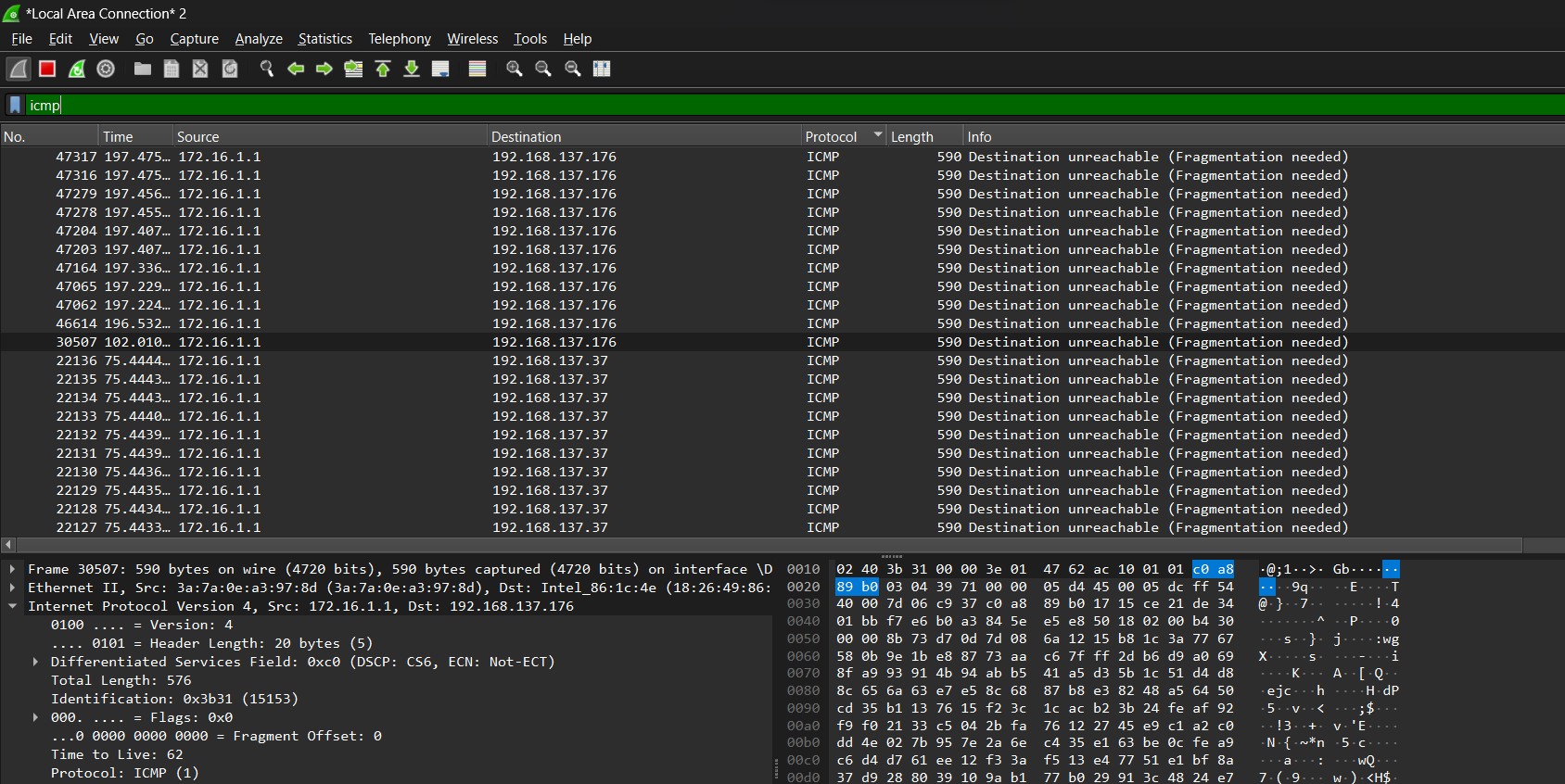
- Example: `http`



9. icmp

- Application: Filters ICMP (Internet Control Message Protocol) packets.

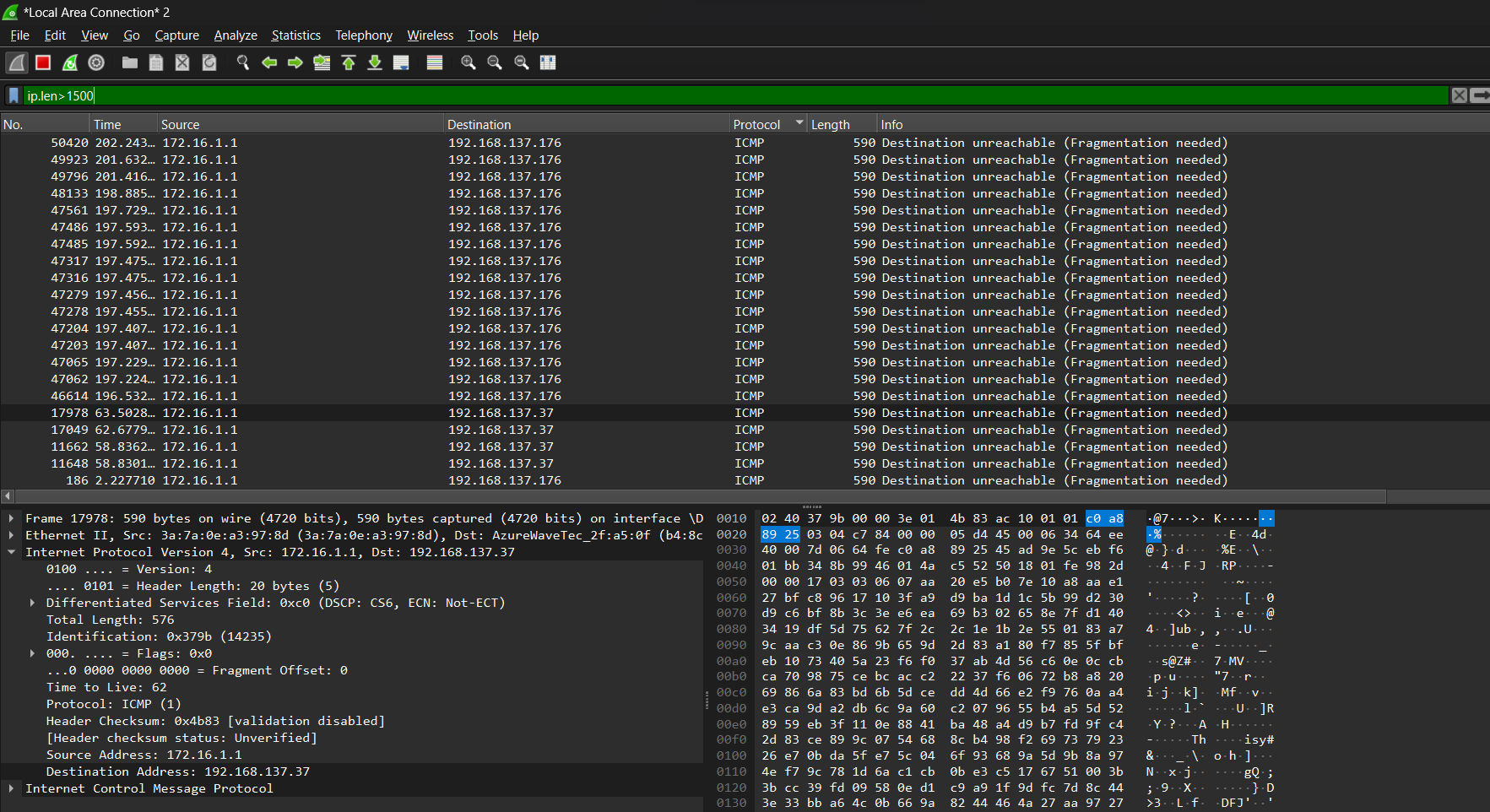
- Example: `icmp`



10. ip.len > 1500

- Application: Filters packets with an IP length greater than 1500 bytes.

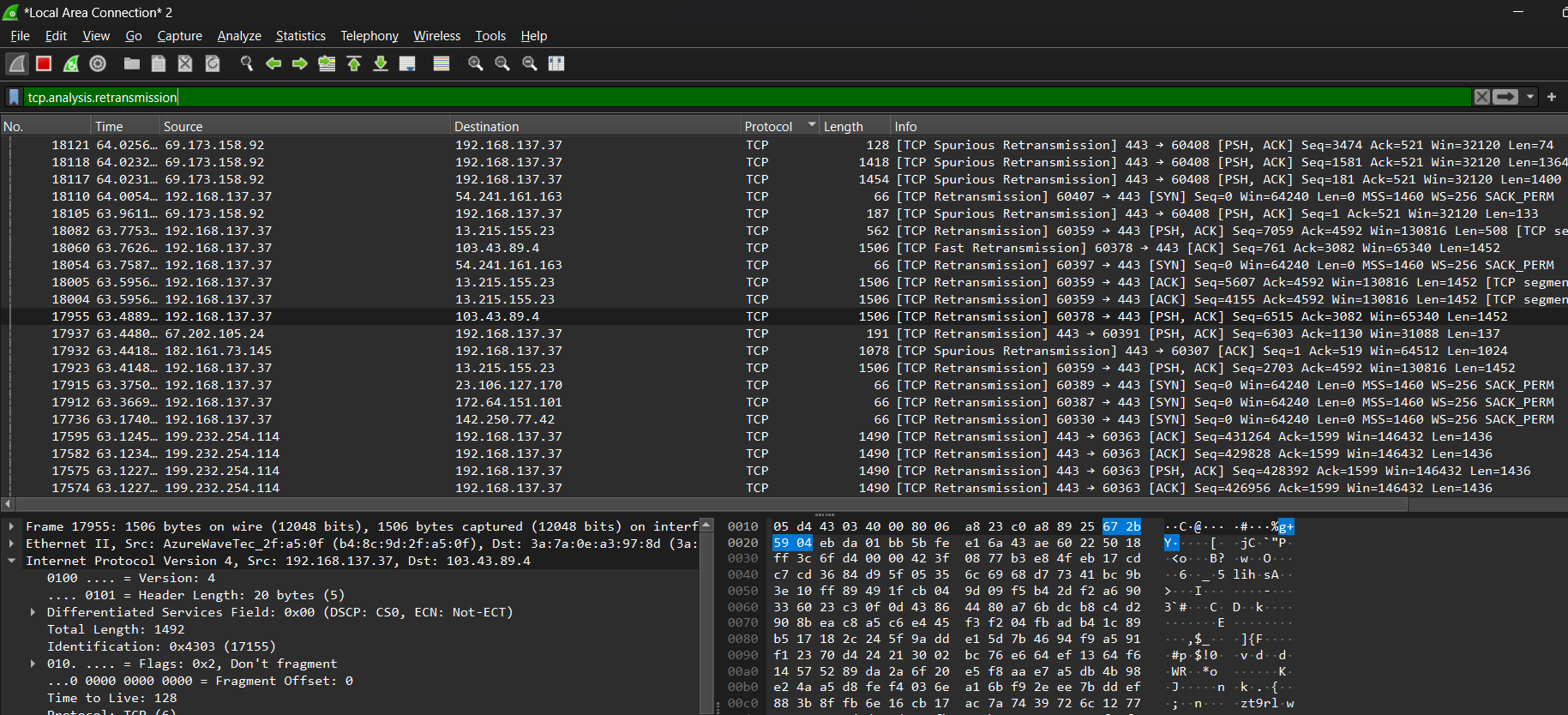
- Example: `ip.len > 1500`



11. tcp.analysis.retransmission

- Application: Filters retransmitted TCP packets.

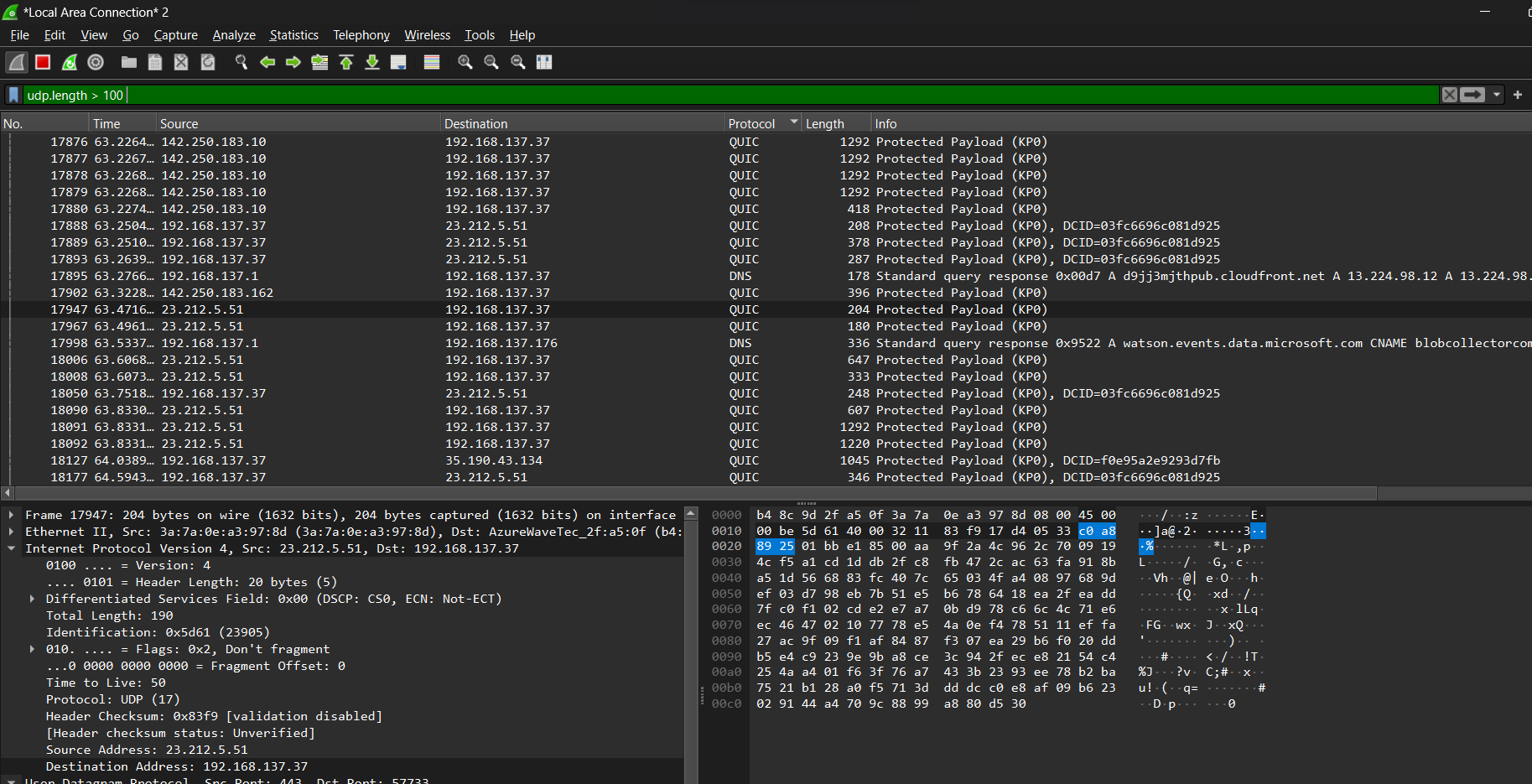
- Example: `tcp.analysis.retransmission`



12. udp.length > 100

- Application: Filters UDP packets with payload length greater than 100 bytes.

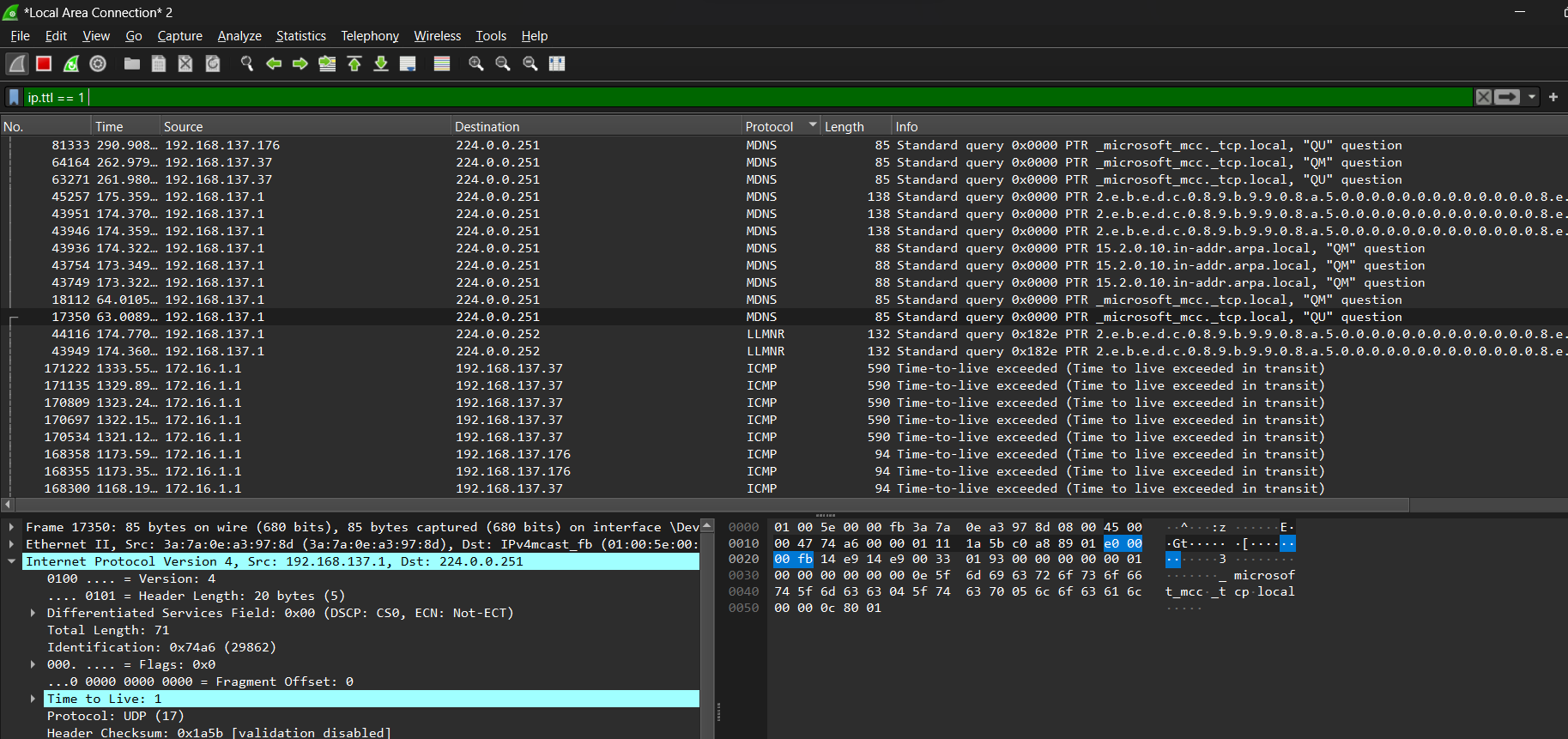
- Example: `udp.length > 100`



13. ip.ttl == 1

- Application: Filters packets with a Time-to-Live (TTL) value of 1.

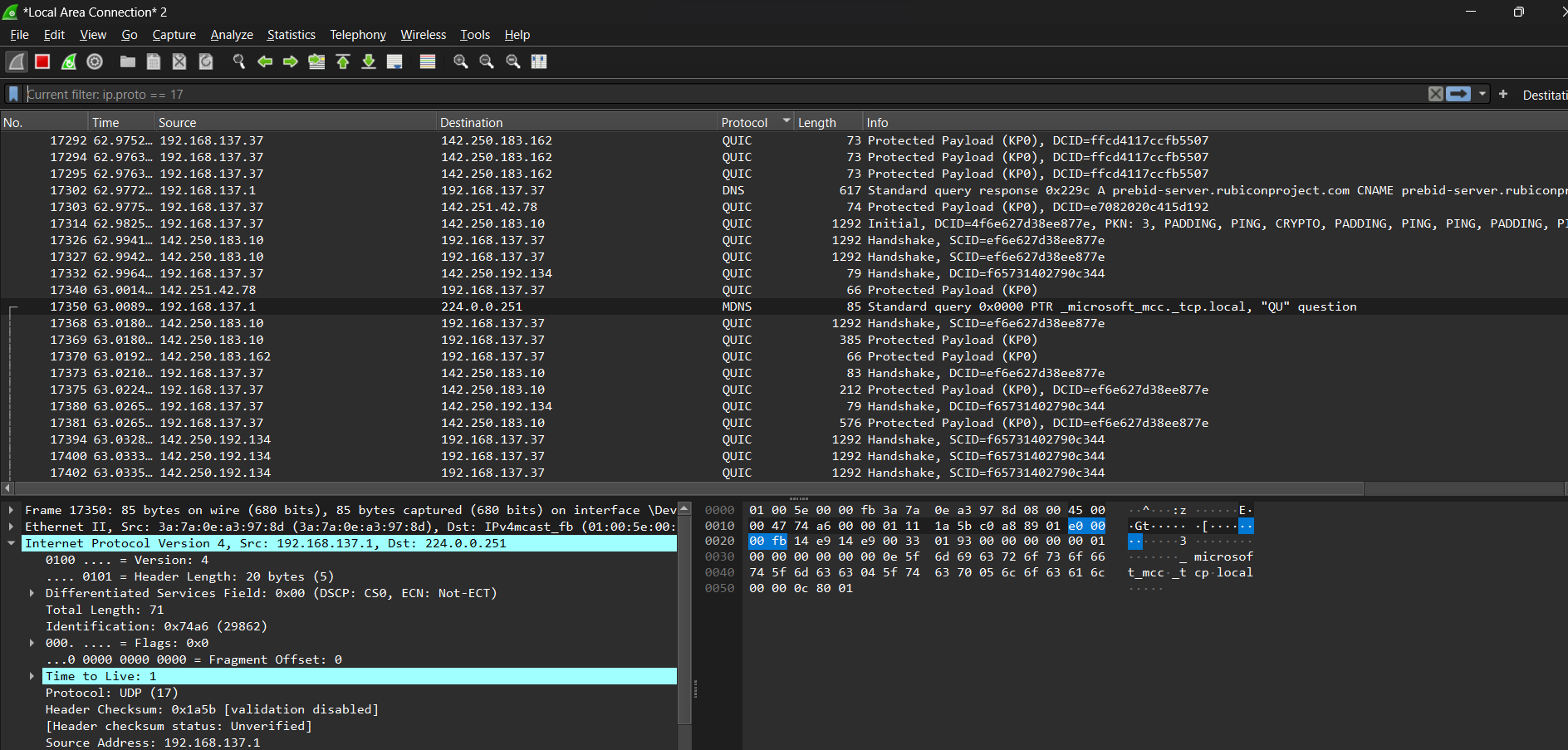
- Example: `ip.ttl == 1`



14. ip.proto == 17

- Application: Filters packets with a specific IP protocol number (17 for UDP).

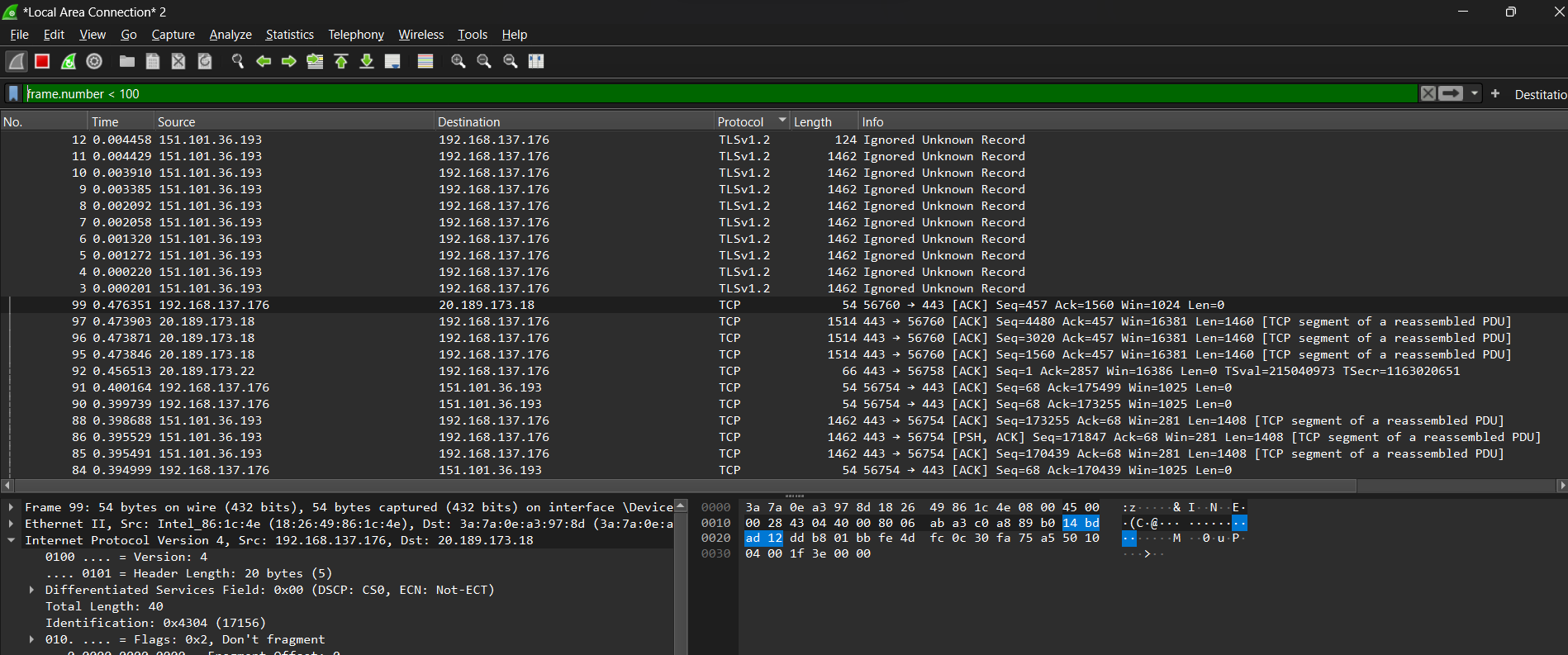
- Example: `ip.proto == 17`



15. frame.number < 100

- Application: Filters the first 100 frames in the capture.

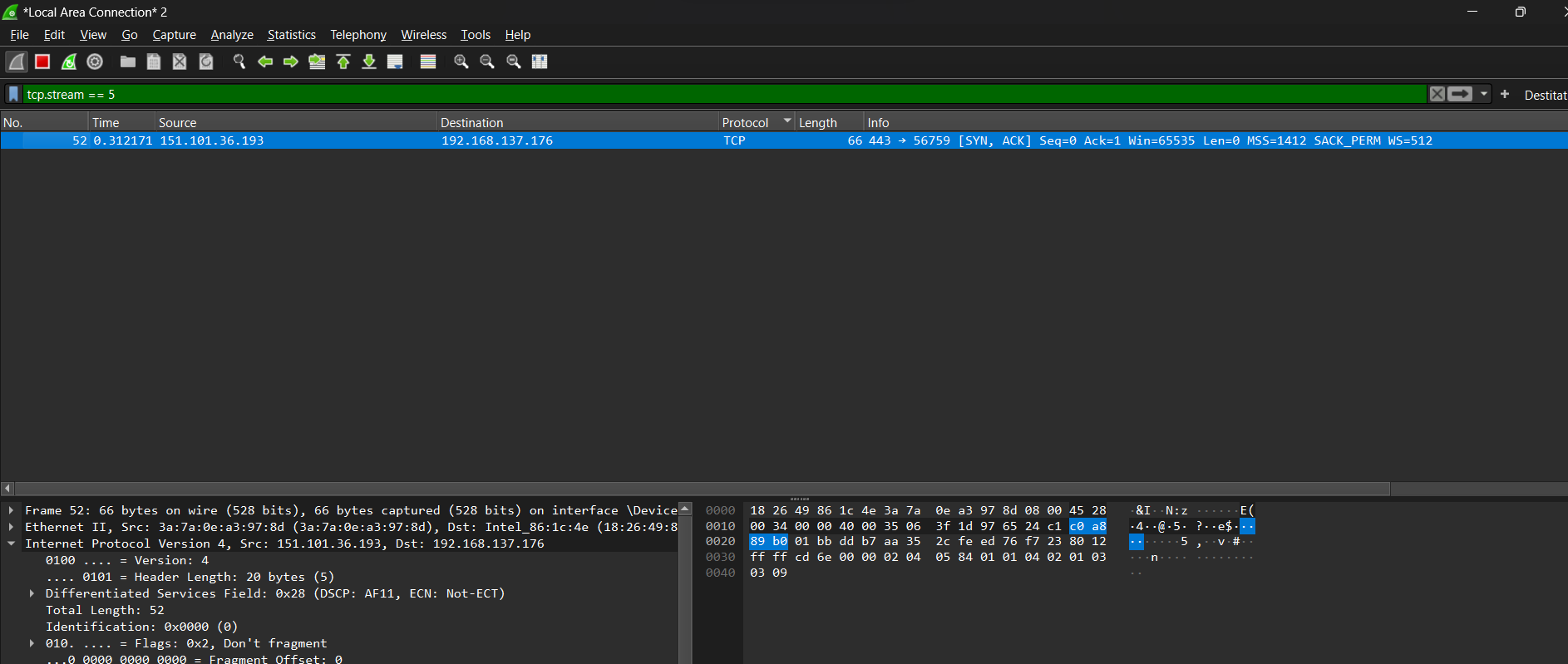
- Example: `frame.number < 100`



17. tcp.stream == x

- Application: Filters packets belonging to a specific TCP stream.

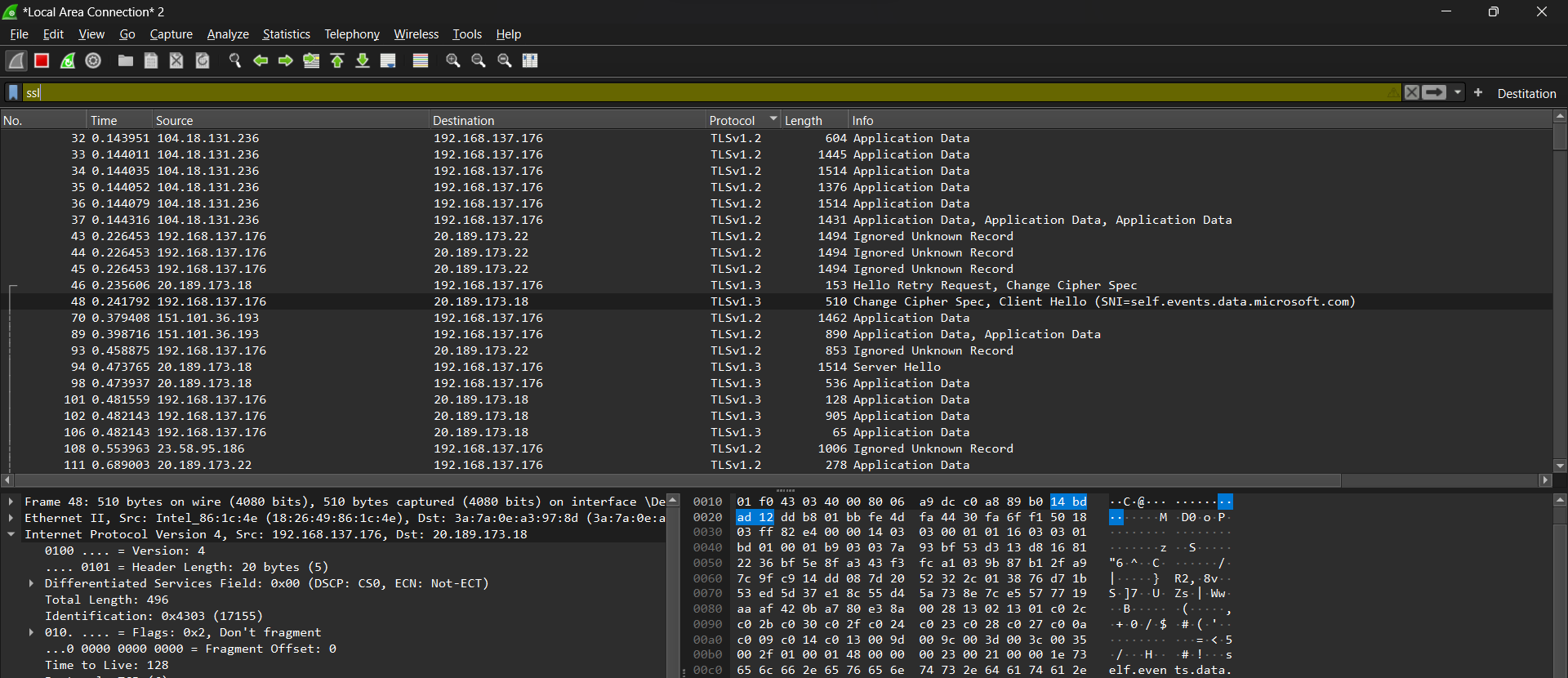
- Example: `tcp.stream == 5`



18. ssl

- Application: Filters SSL/TLS encrypted packets.

- Example: `ssl`



19. ip.addr == x.x.x.x && http.request.method == "POST"

- Application: Filters HTTP POST requests from a specific IP address.

- Example: `ip.addr == 192.168.1.1 && http.request.method == "POST"`

20. frame.time\_delta > 0.1

- Application: Filters packets with a time difference between frames greater than 0.1 seconds.

- Example: `frame.time\_delta > 0.1`