

Packet Analyzer Report

Cedric Bone

Contents

1 Introduction

2 Description

2.1	Command-line Arguments
2.2	Comparison with Wireshark
2.3	Comparison Screenshots (First Few Packets)
2.4	Comparison Screenshots (Last Few Packets)

3 Filter Demonstrations

3.1	Host Filter
3.2	Port Filter
3.3	IP Filter
3.4	TCP Filter
3.5	UDP Filter
3.6	ICMP Filter
3.7	Network Filter

4 Conclusion

1 Introduction

This report summarizes the functionality of pktsniffer.py. The goal was to capture network traffic using Wireshark, parse the captured packets, and then display a breakdown of the packet headers.

2 Description

pktsniffer.py reads packets from a specified .pcap file. It then processes each packet to retrieve Ethernet, IP, and TCP/UDP/ICMP headers.

2.1 Command-line Arguments

Below is a summary of the main command-line arguments:

- **-r** : Path to the pcap file (required).
- **-c** : Limit the number of packets analyzed (optional).
- **-host** : Filter by host (source IP).
- **-port** : Filter by source/destination port.
- **-ip** : Filter by IP address (source or destination).
- **-tcp** : Filter TCP packets.
- **-udp** : Filter UDP packets.
- **-icmp** : Filter ICMP packets.
- **-net** : Filter by a substring that matches a network prefix.

2.2 Comparison with Wireshark

Network traffic was captured using Wireshark and saved to clasp_03_02_2025.pcap. Below are screenshots comparing packets in Wireshark versus pktsniffer.py.

2.3 Comparison Screenshots (First Few Packets)

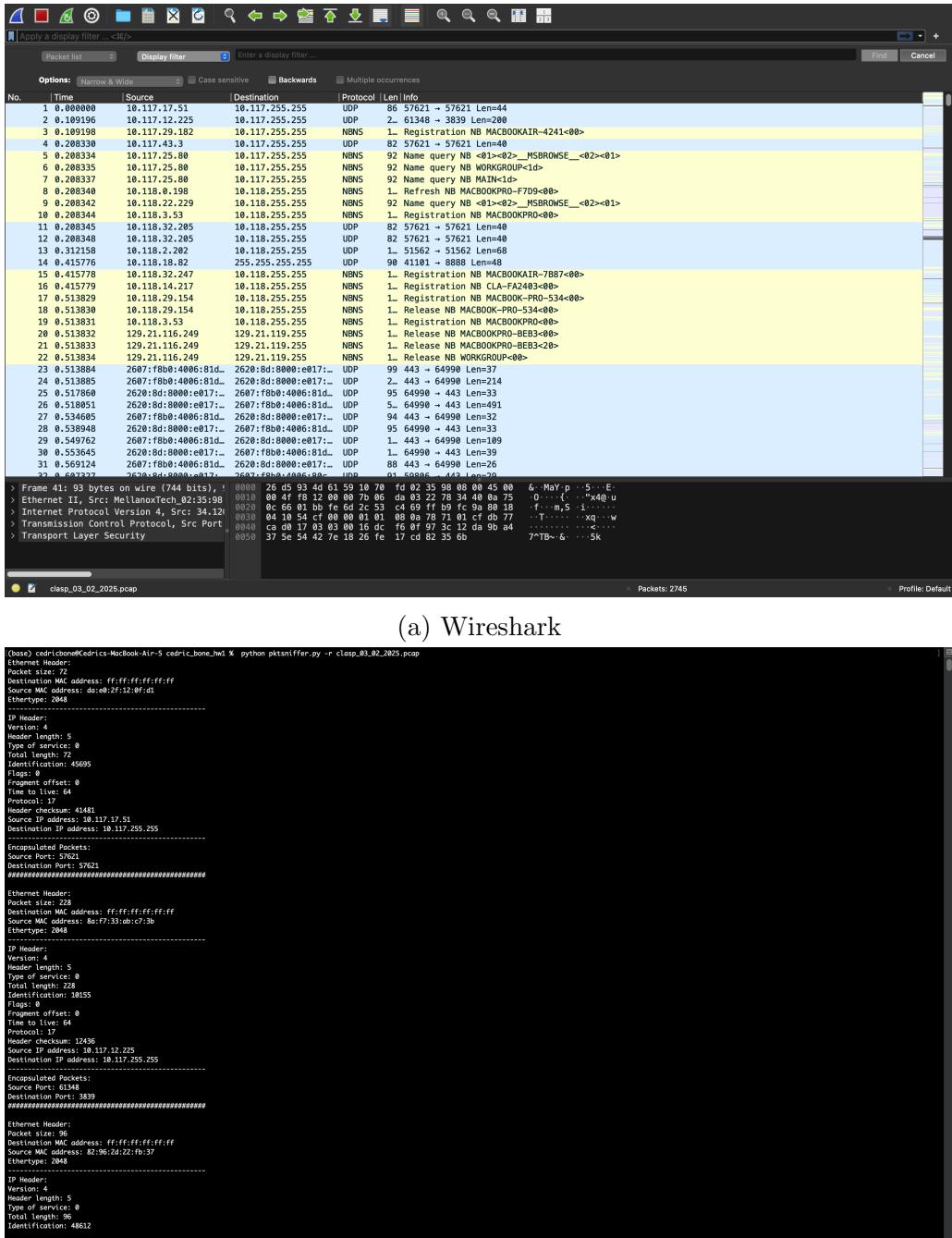


Figure 1: Wireshark vs. pktsniffer (first few packets).

2.4 Comparison Screenshots (Last Few Packets)

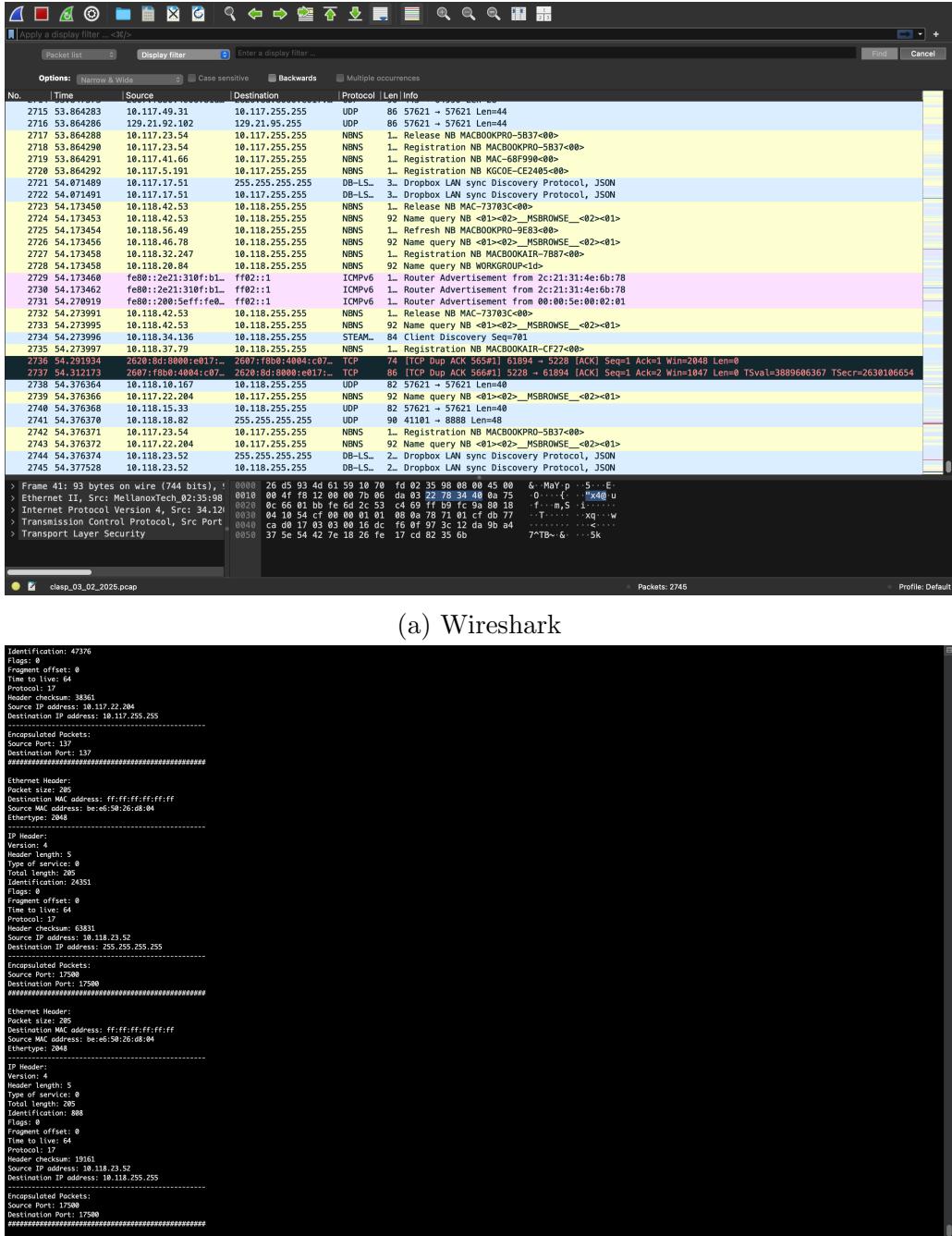


Figure 2: Wireshark vs. pktsniffer (last few packets).

3 Filter Demonstrations

Tested filter flags to ensure correct functionality.

3.1 Host Filter

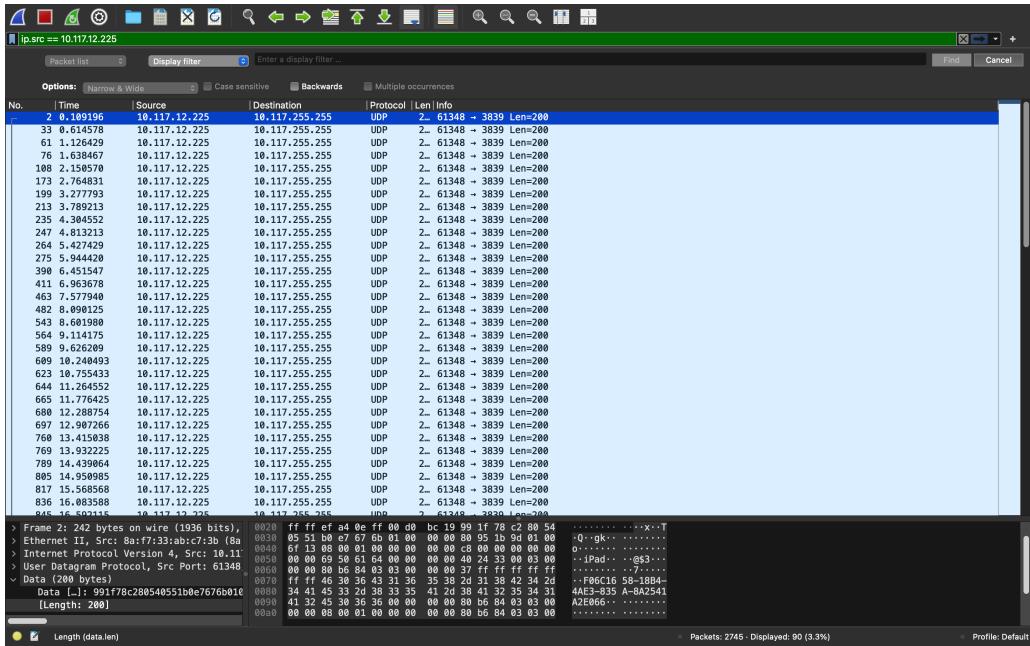
Only packets from 10.117.12.225 are shown.

pktsniffer command:

```
python pktsniffer.py -r clasp_03_02_2025.pcap -host 10.117.12.225
```

Wireshark filter:

```
ip.src == 10.117.12.225
```



(a) Wireshark Host Filter

```
(base) cedricbone@Cedrics-MacBook-Air-5 cedric_bone_hw1 % python pktsniffer.py -r clasp_03_02_2025.pcap -host 10.117.12.225
Ethereal 1.10.0 (Build 20090720)
Packet size: 228
Destination MAC address: ff:ff:ff:ff:ff:ff
Source MAC address: 80:f7:33:ab:c7:3b
Ethernet Header:
-----IP Header:
Version: 4
Header length: 5
Type of service: 0
Total length: 228
Identification: 36155
Flags: 0
Fragment offset: 0
Time to live: 64
Protocol: 17
Header checksum: 12436
Source IP address: 10.117.12.225
Destination IP address: 10.117.255.255
Encapsulated Packets:
Source Port: 9100
Destination Port: 3839
#####
Ethernet Header:
Packet size: 228
Destination MAC address: ffffff:ffff:ffff:ffff:ffff:ffff
Source MAC address: 80:f7:33:ab:c7:3b
Ether-type: 2048
IP Header:
Version: 4
Header length: 5
Type of service: 0
Total length: 228
Identification: 58995
Flags: 0
Fragment offset: 0
Time to live: 64
Protocol: 17
Header checksum: 29131
Source IP address: 10.117.12.225
Destination IP address: 10.117.255.255
Encapsulated Packets:
Source Port: 61348
Destination Port: 3839
#####
Ethernet Header:
Packet size: 228
Destination MAC address: ff:ff:ff:ff:ff:ff
Source MAC address: 80:f7:33:ab:c7:3b
Ether-type: 2048
IP Header:
Version: 4
Header length: 5
Type of service: 0
Total length: 228
Identification: 23694
```

(b) pktsniffer Host Filter

Figure 3: Host filter demonstration.

3.2 Port Filter

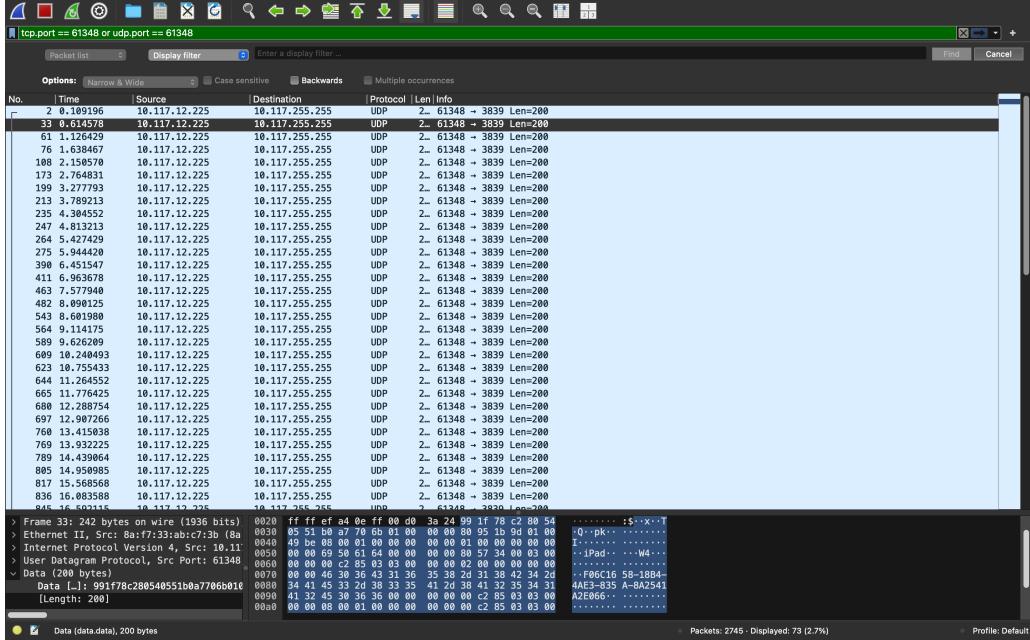
Only packets on port 61348 are shown.

pktsniffer command:

```
python pktsniffer.py -r clasp_03_02_2025.pcap -port 61348
```

Wireshark filter:

```
tcp.port == 61348 or udp.port == 61348
```



(a) Wireshark Port Filter

```
(base) cedricbone@Cedrics-MacBook-Air-5 cedric_bone_hw1 % python pktsniffer.py -r clasp_03_02_2025.pcap -port 61348
Ethernet Header:
  Destination MAC address: ff:ff:ff:ff:ff:ff
  Source MAC address: 8a:f7:33:ab:c7:3b
  EtherType: 0x0800
  IP Header:
    Version: 4
    Header length: 5
    Type of service: 0
    Total length: 228
    Identification: 36155
    Flags: 0
    Fragment offset: 0
    Time to live: 64
    Protocol: 17
    Header checksum: 12436
  Source IP address: 10.117.12.225
  Destination IP address: 10.117.255.255
  Encapsulated Packets:
    Source IP address: 10.117.12.225
    Destination Port: 3839
  #####
Ethernet Header:
  Destination MAC address: ffff:ff:ff:ff:ff:ff
  Source MAC address: 8a:f7:33:ab:c7:3b
  EtherType: 0x0806
  IP Header:
    Version: 4
    Header length: 5
    Type of service: 0
    Total length: 228
    Identification: 58995
    Flags: 0
    Fragment offset: 0
    Time to live: 64
    Protocol: 17
    Header checksum: 29311
  Source IP address: 10.117.12.225
  Destination IP address: 10.117.255.255
  Encapsulated Packets:
    Source Port: 61348
    Destination Port: 3839
  #####
Ethernet Header:
  Destination MAC address: ff:ff:ff:ff:ff:ff
  Source MAC address: 8a:f7:33:ab:c7:3b
  EtherType: 0x0804
  IP Header:
    Version: 4
    Header length: 5
    Type of service: 0
    Total length: 228
    Identification: 23694
```

(b) pktsniffer Port Filter

Figure 4: Port filter demonstration.

3.3 IP Filter

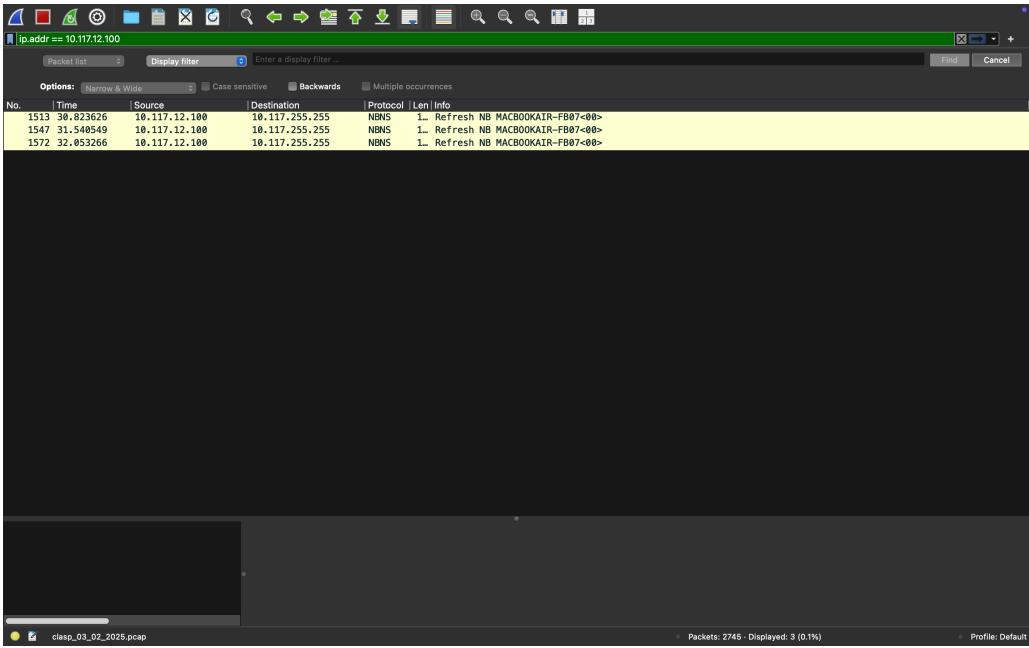
Shows all packets where the source or destination IP is 10.117.12.100.

pktsniffer command:

```
python pktsniffer.py -r clasp_03_02_2025.pcap -ip 10.117.12.100
```

Wireshark filter:

```
ip.addr == 10.117.12.100
```



(a) Wireshark IP Filter

```
(base) cedribonneKedrics-MacBook-Air-5 cedric_bone_hel% python ptksniff.py -r ./captures/2025.pcap -ip 10.117.12.100
[...]
[Ethernet Header: Version: 2 Protocol: 0x0800 Length: 54 Destination MAC address: ff:ff:ff:ff:ff:ff Source MAC address: ce:77:fd:49:c8:54 EtherType: 2048]
[IP Header: Version: 4 Version: 0x0 Protocol: 17 Length: 5 Type of service: 0 Identification: 1548 Identification: 1548 Fragment offset: 0 TTL: 64 Protocol: 17 Header checksum: 3836 Header checksum: 3836 Destination IP address: 10.117.12.100 Destination IP address: 10.117.12.100 Encapsulated Packets:
[...]
Destination Port: 137
#####
[Ethernet Header: Version: 2 Protocol: 0x0800 Length: 54 Destination MAC address: ff:ff:ff:ff:ff:ff Source MAC address: ce:77:fd:49:c8:54 EtherType: 2048]
[IP Header: Version: 4 Version: 0x0 Protocol: 17 Length: 5 Type of service: 0 Identification: 52876 Identification: 52876 Fragment offset: 0 TTL: 64 Protocol: 17 Header checksum: 9387 Header checksum: 9387 Destination IP address: 10.117.12.100 Destination IP address: 10.117.12.100 Encapsulated Packets:
[...]
Destination Port: 137
#####
[Ethernet Header: Version: 2 Protocol: 0x0800 Length: 54 Destination MAC address: ff:ff:ff:ff:ff:ff Source MAC address: ce:77:fd:49:c8:54 EtherType: 2048]
[IP Header: Version: 4 Version: 0x0 Protocol: 17 Length: 5 Type of service: 0 Identification: 57776 Identification: 57776 Fragment offset: 0 TTL: 64 Protocol: 17 Header checksum: 38631 Header checksum: 38631 Destination IP address: 10.117.12.100 Destination IP address: 10.117.12.100 Encapsulated Packets:
[...]
Destination Port: 137
#####
(base) cedribonneKedrics-MacBook-Air-5 cedric_bone_hel% ]
```

(b) pktsniffer IP Filter

Figure 5: IP filter demonstration.

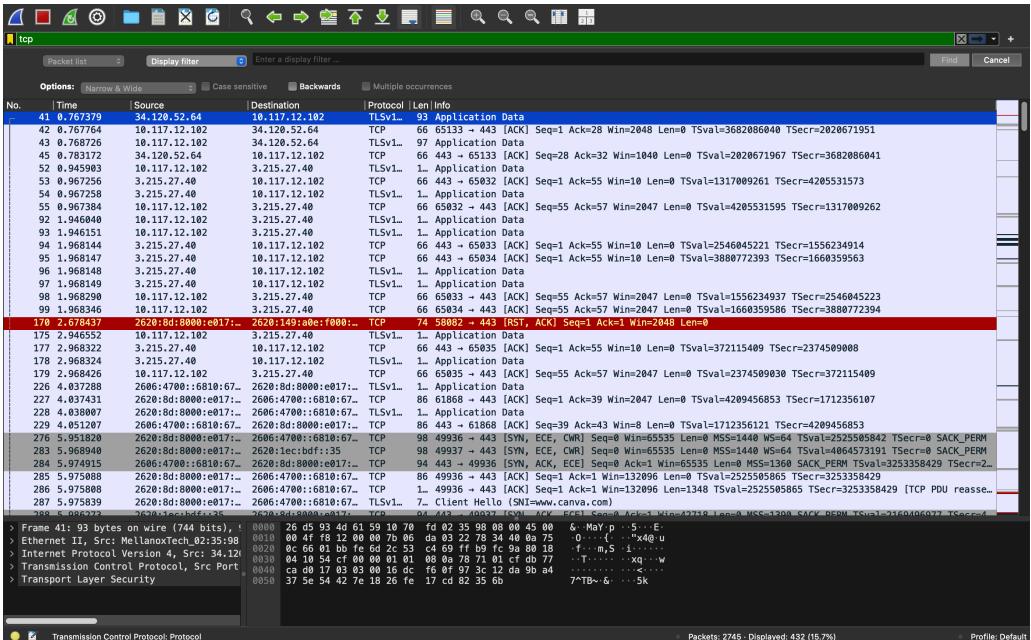
3.4 TCP Filter

Filter only TCP packets.

```
python pktsniffer.py -r clasp_03_02_2025.pcap -tcp
```

Wireshark filter:

```
tcp
```



(a) Wireshark TCP Filter

(b) pktsniffer TCP Filter

Figure 6: TCP filter demonstration.

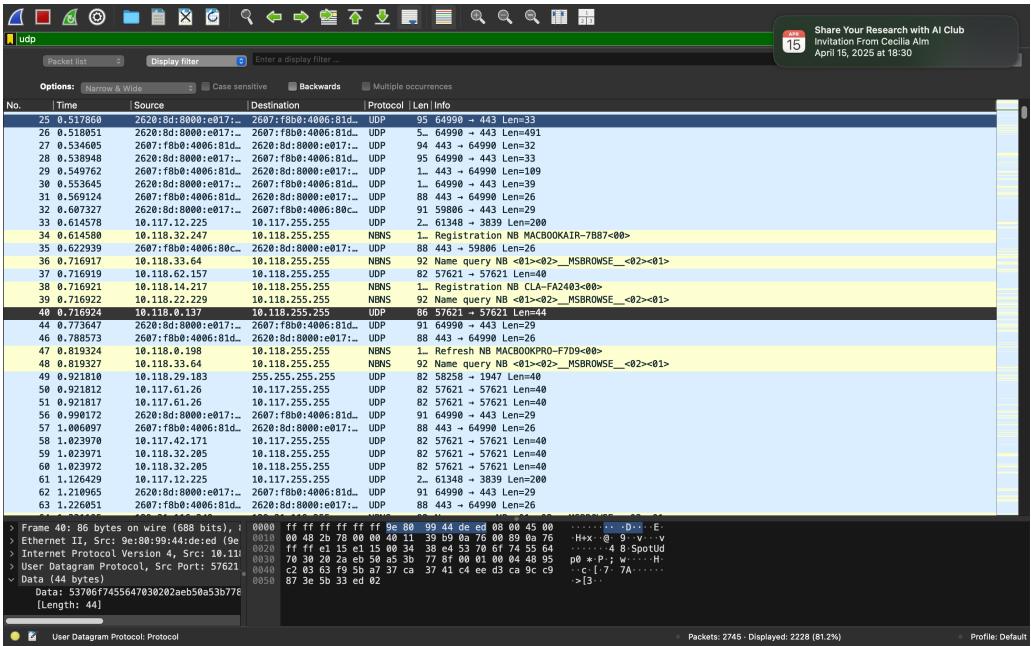
3.5 UDP Filter

Filter only UDP packets.

```
python pktsniffer.py -r clasp_03_02_2025.pcap -udp
```

Wireshark filter:

```
udp
```



(a) Wireshark UDP Filter

(b) pktsniffer UDP Filter

Figure 7: UDP filter demonstration.

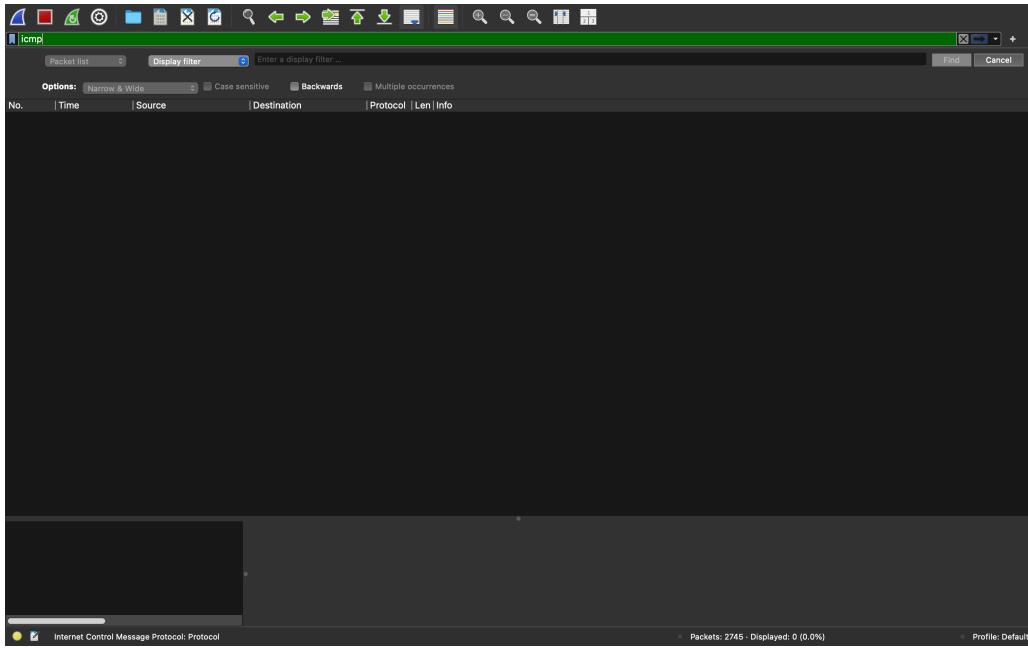
3.6 ICMP Filter

Filter only ICMP packets.

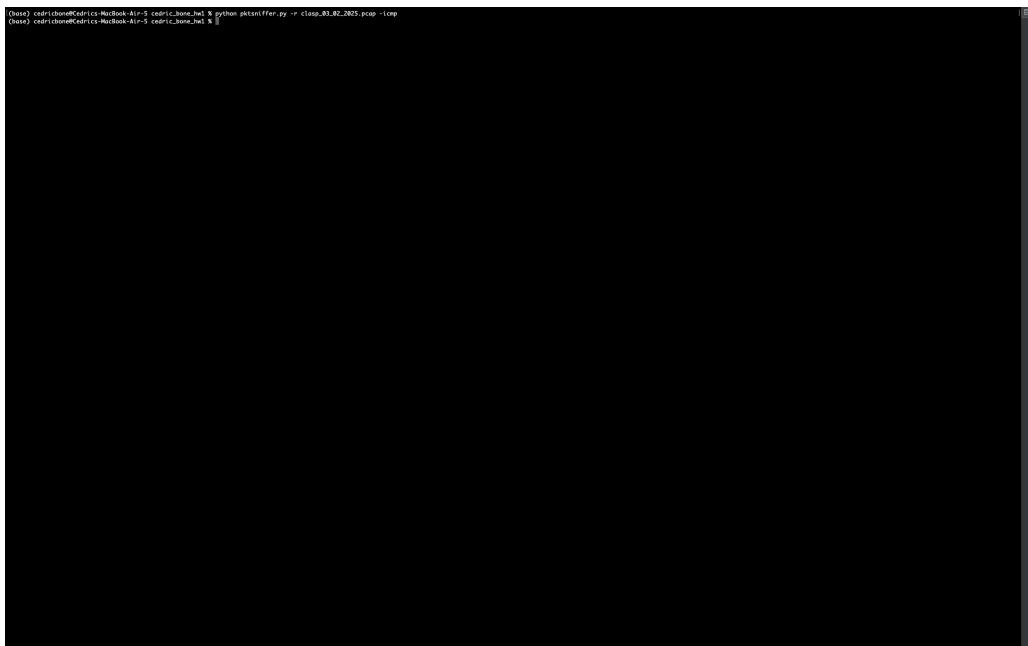
```
python pktsniffer.py -r clasp_03_02_2025.pcap -icmp
```

Wireshark filter:

```
icmp
```



(a) Wireshark ICMP Filter



(b) pktsniffer ICMP Filter

Figure 8: ICMP filter demonstration.

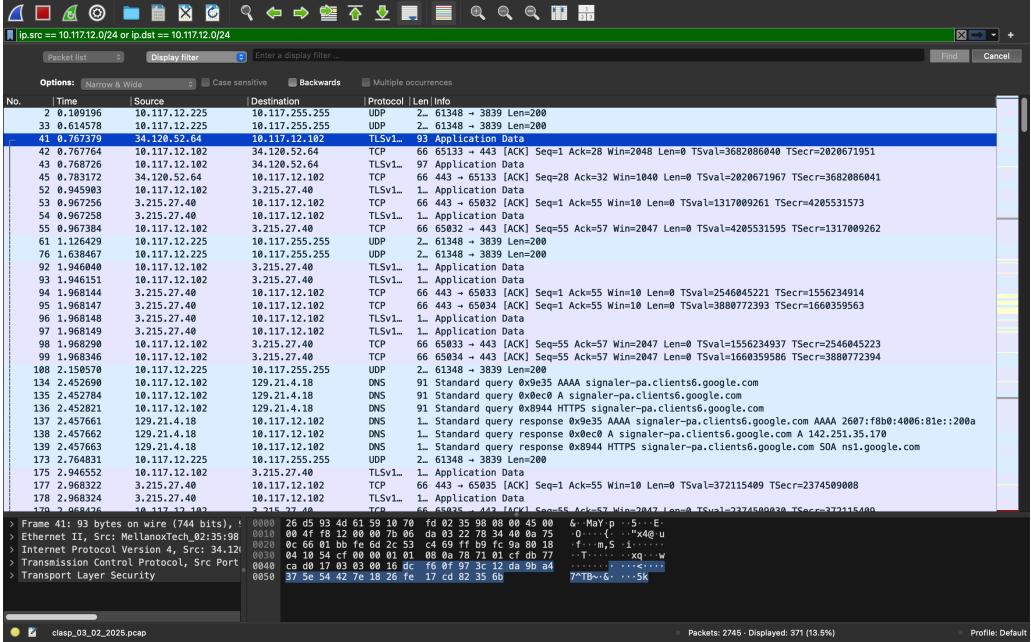
3.7 Network Filter

Packets where the source or destination IP contains 10.117.12.

```
python pktsniffer.py -r clasp_03_02_2025.pcap -net 10.117.12
```

Wireshark filter:

```
ip.src == 10.117.12.0/24 or ip.dst == 10.117.12.0/24
```



(a) Wireshark Net Filter

(b) pktsniffer Net Filter

Figure 9: Network filter demonstration.

4 Conclusion

pktsniffer.py reads from a .pcap file and provides a summary of each packet's Ethernet, IP, and TCP/UDP/ICMP headers. The filters allow subsets of the packets. Comparisons with Wireshark confirm that the program displays accurate header information.