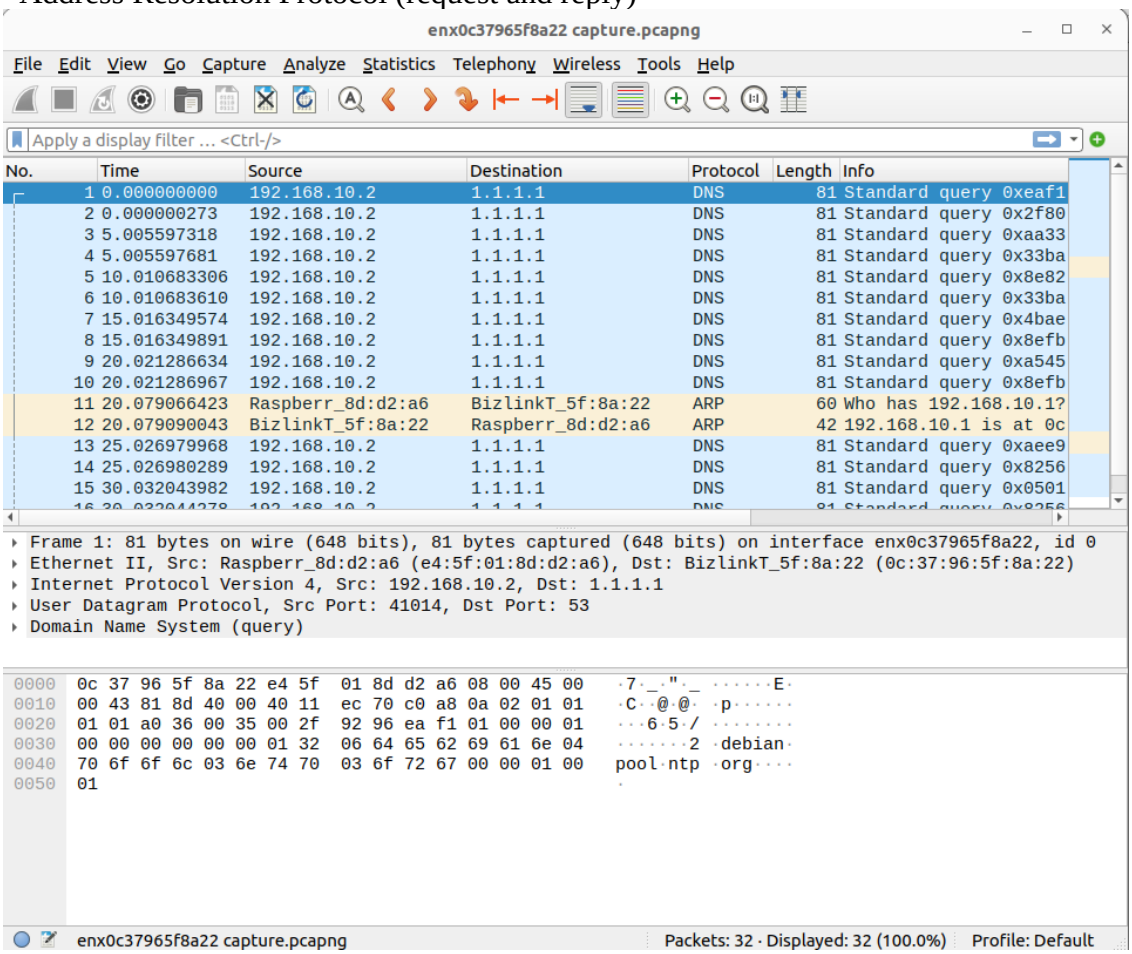
Exercise 1

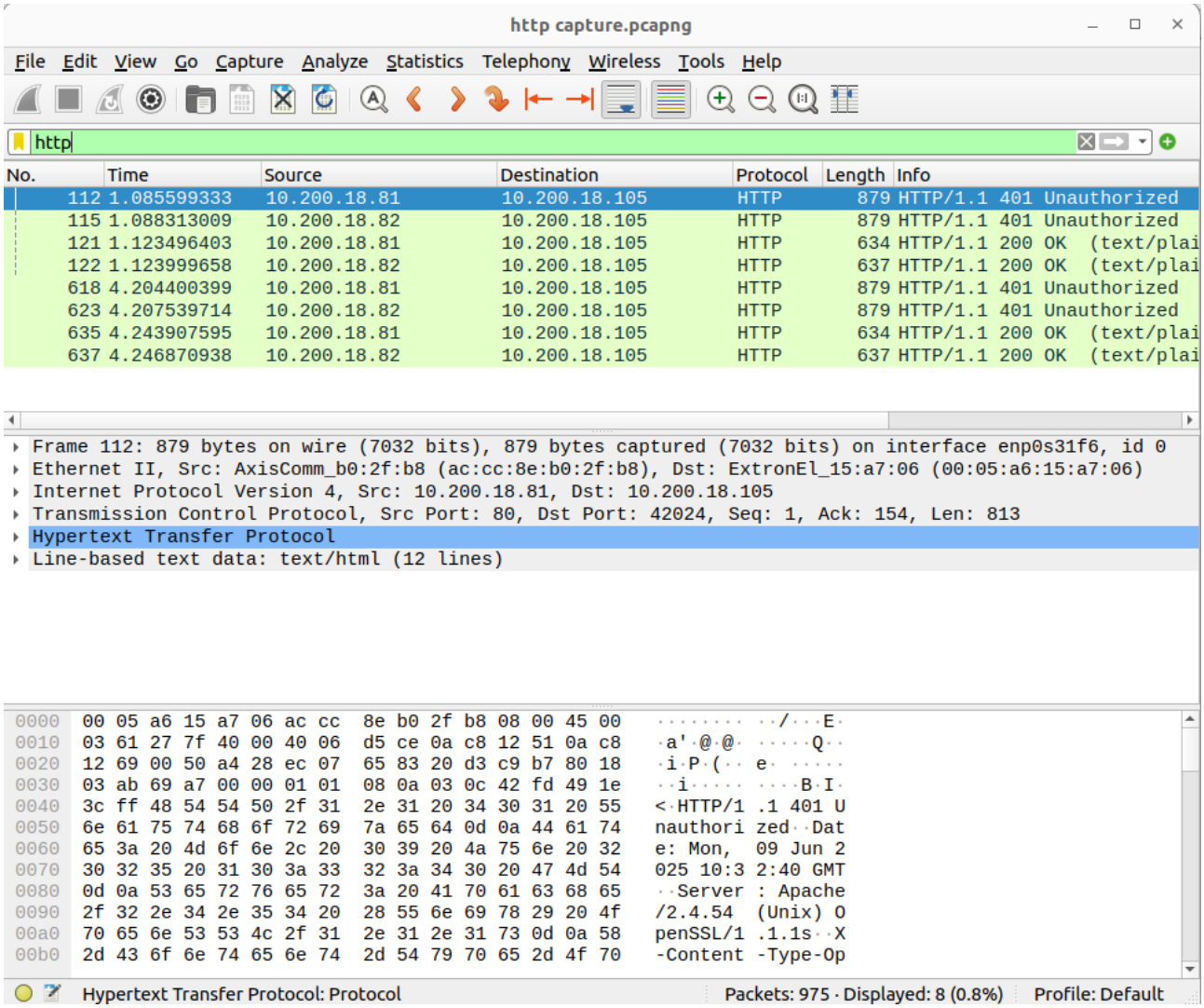
Traffic capture

Selected enx0c37965f8a22 to capture from
Captured Packets:

- Domain Name System (Query)
- Address Resolution Protocol (request and reply)



Selected enp0s31f6
Apply display filter HTTP
start capture and stop

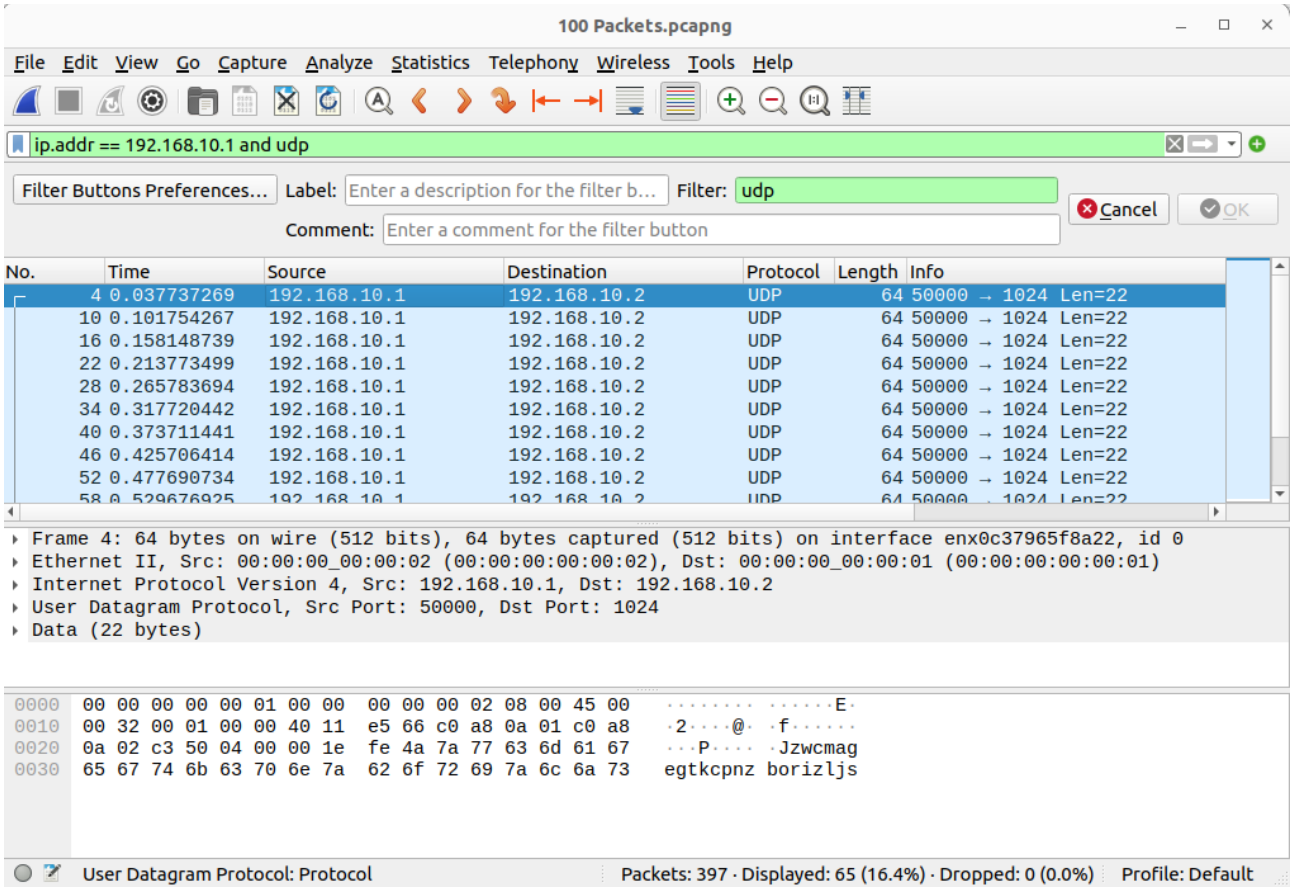


Capture Raspberry Pi using tcp dump

```
pi@p4pi:~ $
pi@p4pi:~ $ sudo tcpdump -i eth0 -c 10 -w captured.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
10 packets captured
10 packets received by filter
0 packets dropped by kernel
pi@p4pi:~ $ tcpdump -r captured.pcap
reading from file captured.pcap, link-type EN10MB (Ethernet), snapshot length 262144
18:00:09.423744 IP 192.168.10.2.ssh > 192.168.10.1.44982: Flags [P.], seq 1023742234:1023742278, ack 2952574732, win 501, options [nop,nop,TS val 3883994997 ecr 2034911232], length 44
18:00:09.423859 IP 192.168.10.2.ssh > 192.168.10.1.44982: Flags [P.], seq 44:96, ack 1, win 501, options [nop,nop,TS val 3883994998 ecr 2034911232], length 52
18:00:09.423948 IP 192.168.10.2.ssh > 192.168.10.1.44982: Flags [P.], seq 96:164, ack 1, win 501, options [nop,nop,TS val 3883994998 ecr 2034911232], length 68
18:00:09.424039 IP 192.168.10.2.ssh > 192.168.10.1.44982: Flags [P.], seq 164:232, ack 1, win 501, options [nop,nop,TS val 3883994998 ecr 2034911232], length 68
18:00:09.424303 IP 192.168.10.1.44982 > 192.168.10.2.ssh: Flags [.], ack 44, win 1632, options [nop,nop,TS val 2034911284 ecr 3883994997], length 0
18:00:09.424304 IP 192.168.10.1.44982 > 192.168.10.2.ssh: Flags [.], ack 96, win 1632, options [nop,nop,TS val 2034911284 ecr 3883994998], length 0
18:00:09.424304 IP 192.168.10.1.44982 > 192.168.10.2.ssh: Flags [.], ack 164, win 1632, options [nop,nop,TS val 2034911284 ecr 3883994998], length 0
18:00:09.424357 IP 192.168.10.1.44982 > 192.168.10.2.ssh: Flags [.], ack 232, win 1632, options [nop,nop,TS val 2034911284 ecr 3883994998], length 0
18:00:11.051235 IP 192.168.10.2.57856 > 1.1.1.1.domain: 3372+ A? 0.debian.pool.ntp.org. (39)
18:00:11.051363 IP 192.168.10.2.36585 > 1.1.1.1.domain: 28013+ AAAA? 0.debian.pool.ntp.org. (39)
pi@p4pi:~ $
```

Sending Traffic

1. Can you define a filter to capture only the packets you sent?
Using ip.addr == 192.168.10.1 and udp



2. What is the packet size?
512bits

3. What is the protocol used (TCP or UDP)?
UDP

Modified Script send_TCP.py

```
#!/usr/bin/python

from scapy.all import Ether, IP, sendp, get_if_hwaddr, get_if_list, TCP, Raw, UDP
import sys
import random, string


def randomword(length):
    return ''.join(random.choice(string.ascii_lowercase) for i in range(length))

def send_random_traffic(num_packets, interface, src_ip, dst_ip):
    dst_mac = "00:00:00:00:00:01"
    src_mac= "CA:FE:CA:FE:CA:FE"
    total_pkts = 0
    port = 1024
    for i in range(num_packets):
            data = randomword(458)
            p = Ether(dst=dst_mac,src=src_mac)/IP(dst=dst_ip,src=src_ip)
            p = p/TCP(sport= 5555, dport=port)/Raw(load=data)
            sendp(p, iface = interface, inter = 0.01)
            # If you want to see the contents of the packet, uncomment the line below
            # print(p.show())
            total_pkts += 1
    print("Sent %s packets in total" % total_pkts)

if __name__ == '__main__':
    if len(sys.argv) < 5:
        print("Usage: python send.py number_of_packets interface_name src_ip_address dst_ip_address")
        sys.exit(1)
    else:
        num_packets = sys.argv[1]
        interface = sys.argv[2]
        src_ip = sys.argv[3]
        dst_ip = sys.argv[4]
        send_random_traffic(int(num_packets), interface, src_ip, dst_ip)
```