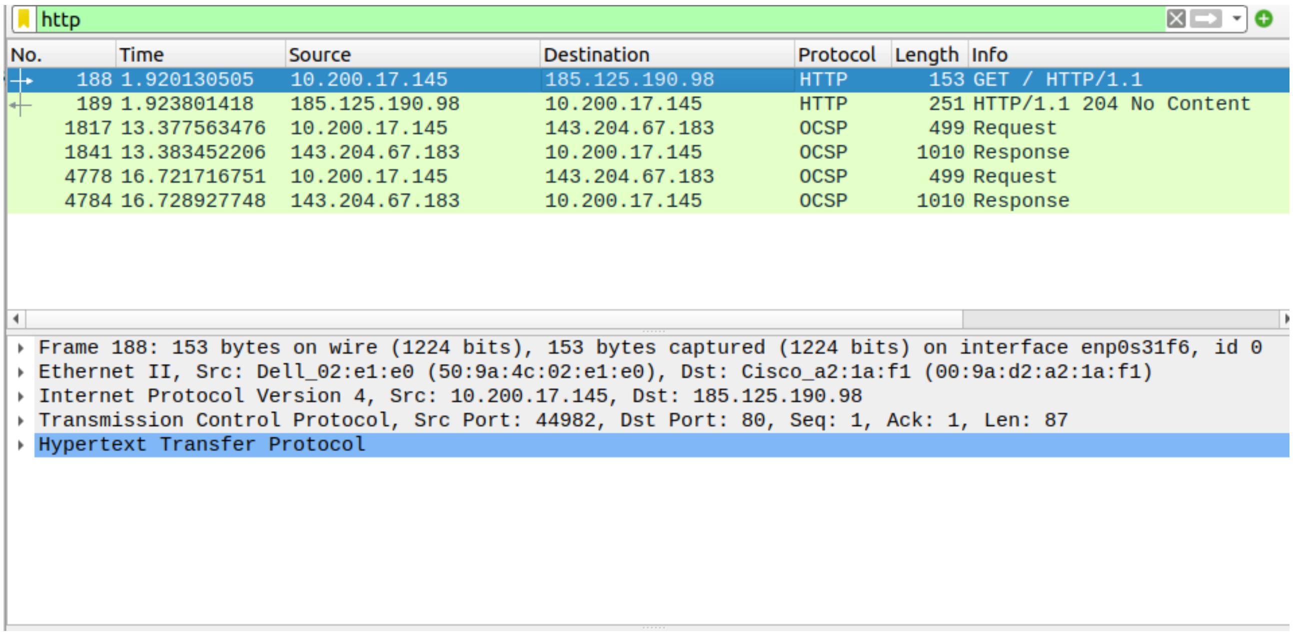
1. Packet captured includes DNS packets, ARP packets

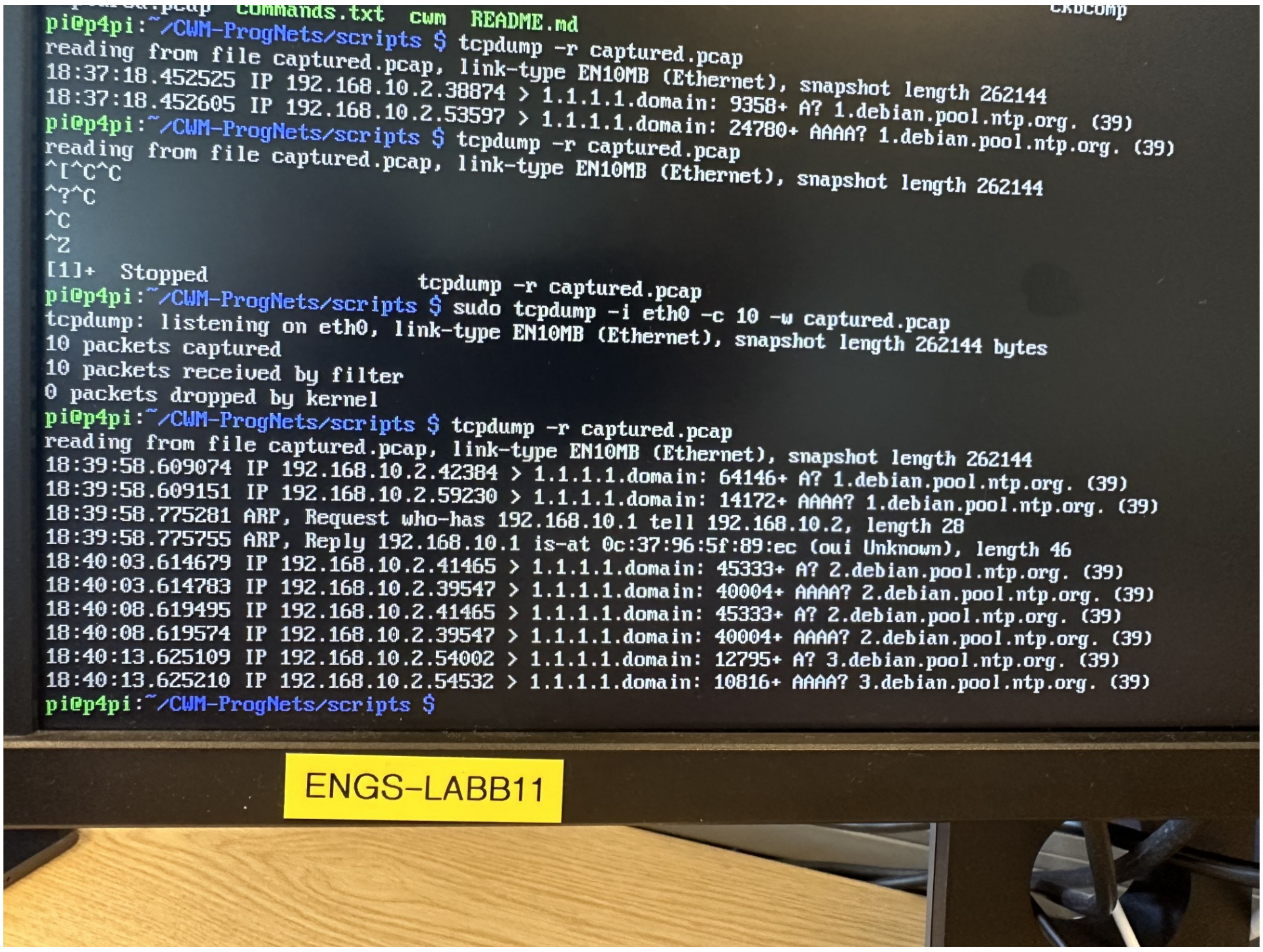
2.



```
http

No.   Time          Source           Destination      Protocol  Length  Info
  188 1.920130505   10.200.17.145    185.125.190.98   HTTP         153  GET / HTTP/1.1
  189 1.923801418   185.125.190.98   10.200.17.145    HTTP         251  HTTP/1.1 204 No Content
 1817 13.377563476  10.200.17.145    143.204.67.183   OCSP         499  Request
 1841 13.383452206  143.204.67.183   10.200.17.145    OCSP        1010  Response
 4778 16.721716751  10.200.17.145    143.204.67.183   OCSP         499  Request
 4784 16.728927748  143.204.67.183   10.200.17.145    OCSP        1010  Response

Frame 188: 153 bytes on wire (1224 bits), 153 bytes captured (1224 bits) on interface enp0s31f6, id 0
Ethernet II, Src: Dell_02:e1:e0 (50:9a:4c:02:e1:e0), Dst: Cisco_a2:1a:f1 (00:9a:d2:a2:1a:f1)
Internet Protocol Version 4, Src: 10.200.17.145, Dst: 185.125.190.98
Transmission Control Protocol, Src Port: 44982, Dst Port: 80, Seq: 1, Ack: 1, Len: 87
Hypertext Transfer Protocol
```

Open a webpage while the it is capturing, apply filter to find http packet

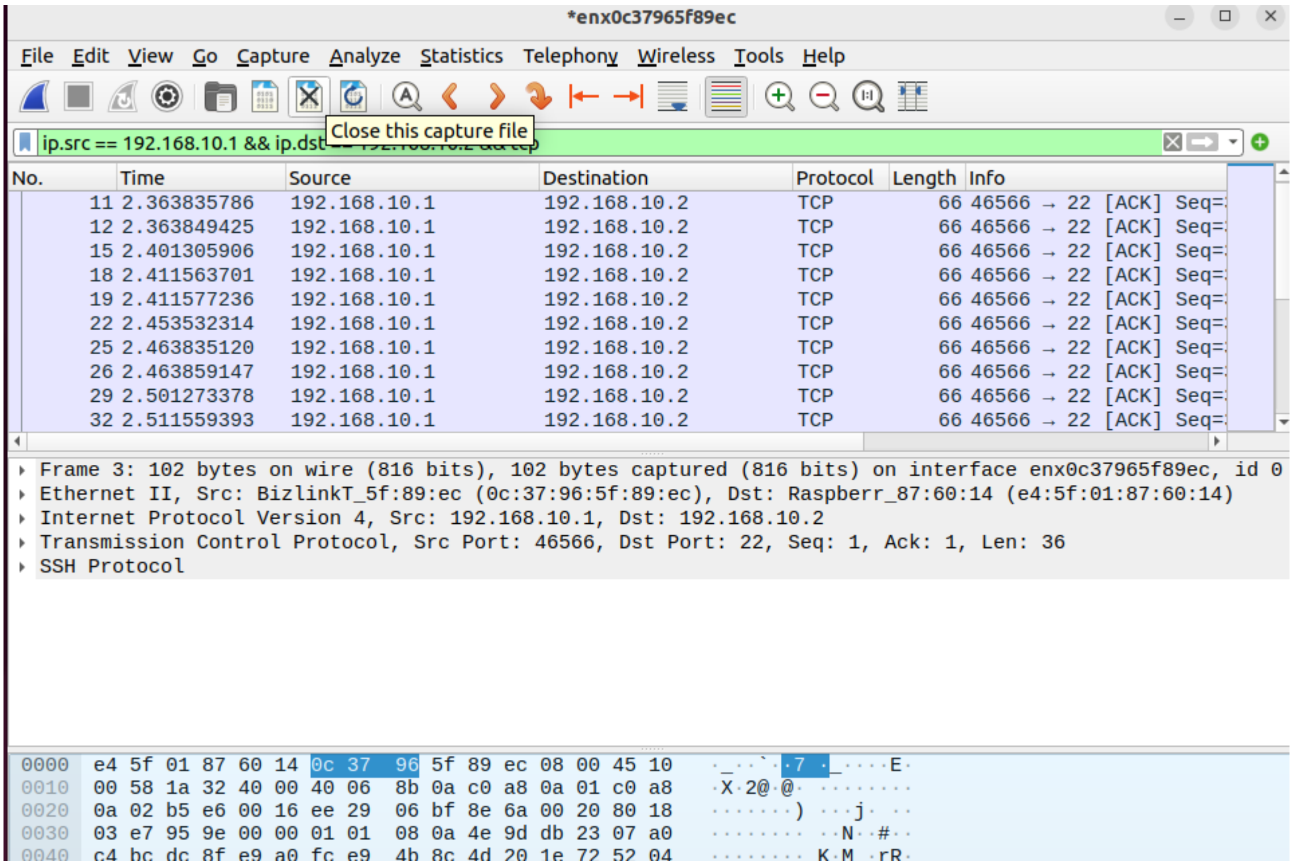
3.



```
commands.txt  cwm  README.md                                                    ckbcomp
pi@p4pi:~/CWM-ProgNets/scripts $ tcpdump -r captured.pcap
reading from file captured.pcap, link-type EN10MB (Ethernet), snapshot length 262144
18:37:18.452525 IP 192.168.10.2.38874 > 1.1.1.1.domain: 9358+ A? 1.debian.pool.ntp.org. (39)
18:37:18.452605 IP 192.168.10.2.53597 > 1.1.1.1.domain: 24780+ AAAA? 1.debian.pool.ntp.org. (39)
pi@p4pi:~/CWM-ProgNets/scripts $ tcpdump -r captured.pcap
reading from file captured.pcap, link-type EN10MB (Ethernet), snapshot length 262144
^[^C^C
^?^C
^C
^Z
[1]+  Stopped                 tcpdump -r captured.pcap
pi@p4pi:~/CWM-ProgNets/scripts $ 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:~/CWM-ProgNets/scripts $ tcpdump -r captured.pcap
reading from file captured.pcap, link-type EN10MB (Ethernet), snapshot length 262144
18:39:58.609074 IP 192.168.10.2.42384 > 1.1.1.1.domain: 64146+ A? 1.debian.pool.ntp.org. (39)
18:39:58.609151 IP 192.168.10.2.59230 > 1.1.1.1.domain: 14172+ AAAA? 1.debian.pool.ntp.org. (39)
18:39:58.775281 ARP, Request who-has 192.168.10.1 tell 192.168.10.2, length 28
18:39:58.775755 ARP, Reply 192.168.10.1 is-at 0c:37:96:5f:89:ec (oui Unknown), length 46
18:40:03.614679 IP 192.168.10.2.41465 > 1.1.1.1.domain: 45333+ A? 2.debian.pool.ntp.org. (39)
18:40:03.614783 IP 192.168.10.2.39547 > 1.1.1.1.domain: 40004+ AAAA? 2.debian.pool.ntp.org. (39)
18:40:08.619495 IP 192.168.10.2.41465 > 1.1.1.1.domain: 45333+ A? 2.debian.pool.ntp.org. (39)
18:40:08.619574 IP 192.168.10.2.39547 > 1.1.1.1.domain: 40004+ AAAA? 2.debian.pool.ntp.org. (39)
18:40:13.625109 IP 192.168.10.2.54002 > 1.1.1.1.domain: 12795+ A? 3.debian.pool.ntp.org. (39)
18:40:13.625210 IP 192.168.10.2.54532 > 1.1.1.1.domain: 10816+ AAAA? 3.debian.pool.ntp.org. (39)
pi@p4pi:~/CWM-ProgNets/scripts $
```

.

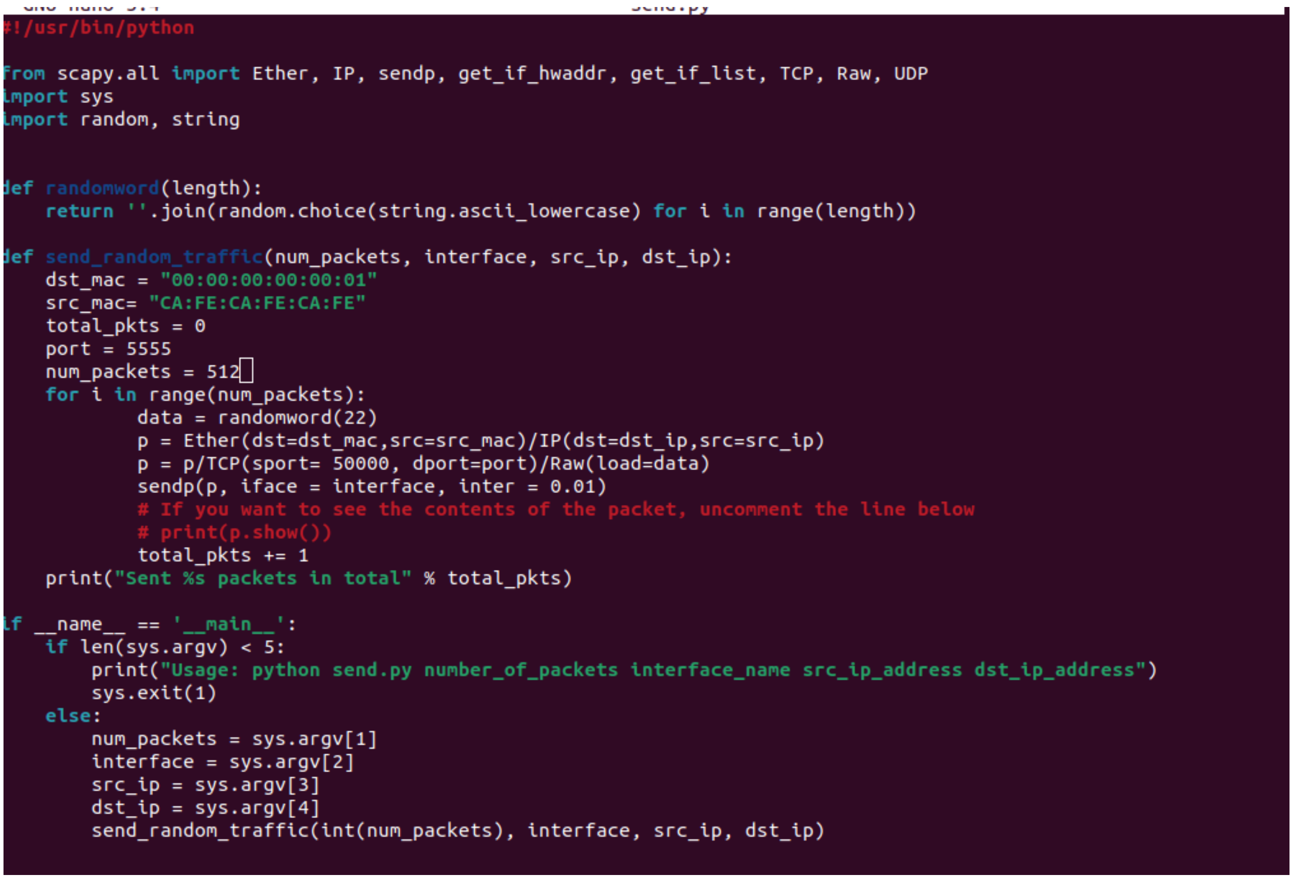
4.


*enx0c37965f89ec
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Close this capture file

```
ip.src == 192.168.10.1 && ip.dst == 192.168.10.2 && tcp

No.  Time           Source         Destination    Protocol  Length Info
  11 2.363835786    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  12 2.363849425    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  15 2.401305906    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  18 2.411563701    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  19 2.411577236    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  22 2.453532314    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  25 2.463835120    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  26 2.463859147    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  29 2.501273378    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=
  32 2.511559393    192.168.10.1   192.168.10.2   TCP       66 46566 → 22 [ACK] Seq=

▸ Frame 3: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface enx0c37965f89ec, id 0
▸ Ethernet II, Src: BizlinkT_5f:89:ec (0c:37:96:5f:89:ec), Dst: Raspberr_87:60:14 (e4:5f:01:87:60:14)
▸ Internet Protocol Version 4, Src: 192.168.10.1, Dst: 192.168.10.2
▸ Transmission Control Protocol, Src Port: 46566, Dst Port: 22, Seq: 1, Ack: 1, Len: 36
▸ SSH Protocol

0000  e4 5f 01 87 60 14 0c 37  96 5f 89 ec 08 00 45 10   ·_··`··7 ·_····E·
0010  00 58 1a 32 40 00 40 06  8b 0a c0 a8 0a 01 c0 a8   ·X·2@·@· ········
0020  0a 02 b5 e6 00 16 ee 29  06 bf 8e 6a 00 20 80 18   ·······) ···j· ··
0030  03 e7 95 9e 00 00 01 01  08 0a 4e 9d db 23 07 a0   ········ ··N··#··
0040  c4 bc dc 8f e9 a0 fc e9  4b 8c 4d 20 1e 72 52 04   ········ K·M ·rR·
```

filter as shown
size = 66 bytes
protocol = TCP

Modified script



```
#!/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 = 5555
    num_packets = 512
    for i in range(num_packets):
            data = randomword(22)
            p = Ether(dst=dst_mac,src=src_mac)/IP(dst=dst_ip,src=src_ip)
            p = p/TCP(sport= 50000, 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)
```