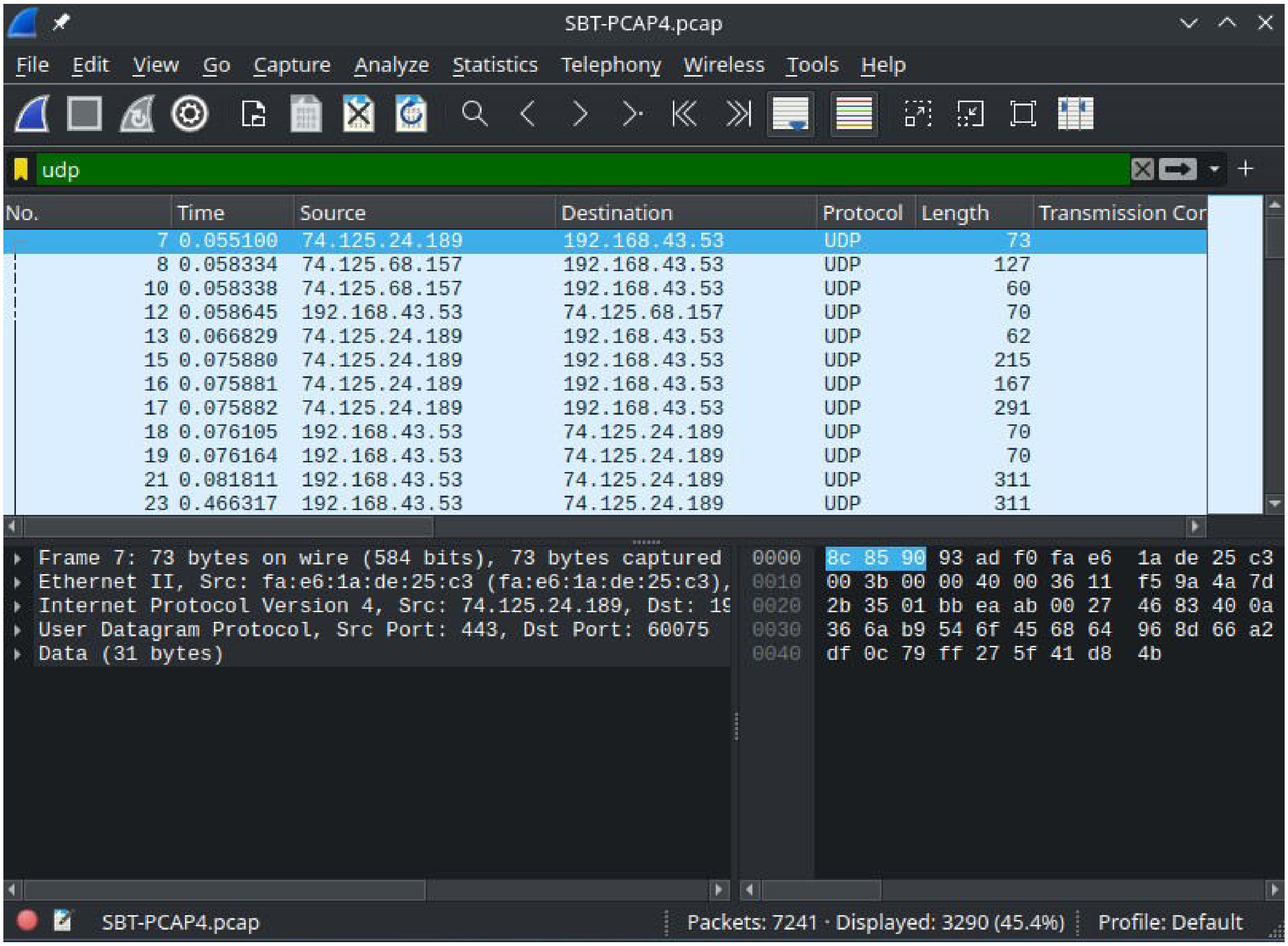
TCPDump Challenge [Activity]

Disclaimer: Sorry! I will be explaining this activity with the help of wireshark tool and not with tcpdump command. Will try to upload the explanation with tcpdump as soon as possible.

PCAP 4

1.How many UDP packets have been captured?

Search udp in the display filter and check the number of the packets displayed. In this case it will be 3290 packets.



udp

```
No.     Time       Source           Destination      Protocol  Length  Transmission Cor
      7 0.055100   74.125.24.189    192.168.43.53    UDP           73
      8 0.058334   74.125.68.157    192.168.43.53    UDP          127
     10 0.058338   74.125.68.157    192.168.43.53    UDP           60
     12 0.058645   192.168.43.53    74.125.68.157    UDP           70
     13 0.066829   74.125.24.189    192.168.43.53    UDP           62
     15 0.075880   74.125.24.189    192.168.43.53    UDP          215
     16 0.075881   74.125.24.189    192.168.43.53    UDP          167
     17 0.075882   74.125.24.189    192.168.43.53    UDP          291
     18 0.076105   192.168.43.53    74.125.24.189    UDP           70
     19 0.076164   192.168.43.53    74.125.24.189    UDP           70
     21 0.081811   192.168.43.53    74.125.24.189    UDP          311
     23 0.466317   192.168.43.53    74.125.24.189    UDP          311
```

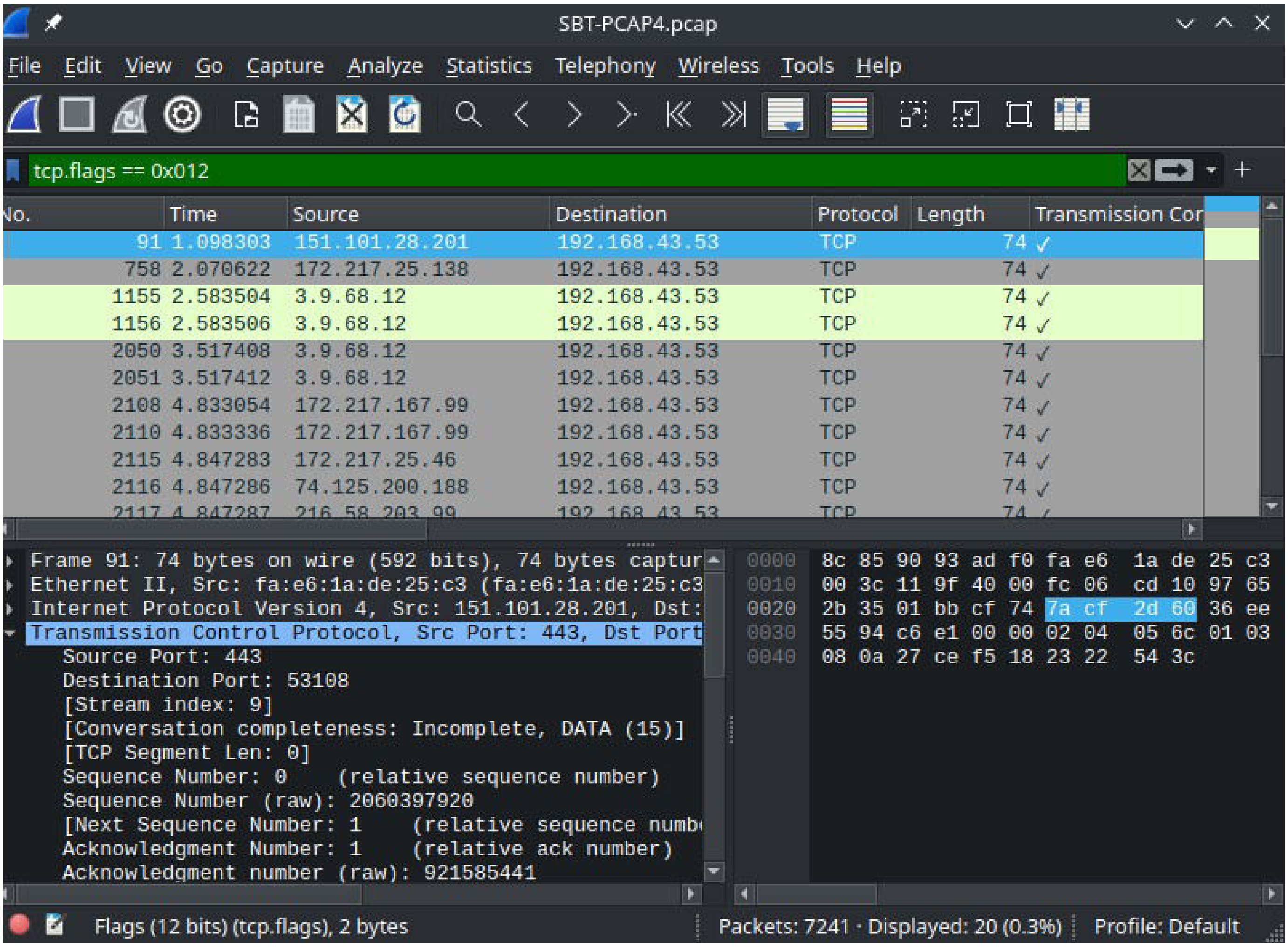
```
▸ Frame 7: 73 bytes on wire (584 bits), 73 bytes captured
▸ Ethernet II, Src: fa:e6:1a:de:25:c3 (fa:e6:1a:de:25:c3),
▸ Internet Protocol Version 4, Src: 74.125.24.189, Dst: 19
▸ User Datagram Protocol, Src Port: 443, Dst Port: 60075
▸ Data (31 bytes)
```

```
0000  8c 85 90 93 ad f0 fa e6  1a de 25 c3
0010  00 3b 00 00 40 00 36 11  f5 9a 4a 7d
0020  2b 35 01 bb ea ab 00 27  46 83 40 0a
0030  36 6a b9 54 6f 45 68 64  96 8d 66 a2
0040  df 0c 79 ff 27 5f 41 d8  4b
```

SBT-PCAP4.pcap | Packets: 7241 · Displayed: 3290 (45.4%) | Profile: Default

2.How many TCP packets have both the SYN and ACK flags set?

Apply the filter ‘tcp.flags == 0x012’ to get the tcp flag that have both the SYN and ACK flags set. (If you didn’t know how to use the display filter for these kind of questions, I will give you a head start for this. In this case search tcp and search for the for packet with both ACK and SYN flags set. In the packet header window under flags header, select the property you want to be displayed and right click on it and hover over select as filter and option. You will see the syntax of the filter to be applied for the coressponding property.) By following the any of the method you will get 20 packets finally.



tcp.flags == 0x012

No.	Time	Source	Destination	Protocol	Length	Transmission Cor
91	1.098303	151.101.28.201	192.168.43.53	TCP	74	✓
758	2.070622	172.217.25.138	192.168.43.53	TCP	74	✓
1155	2.583504	3.9.68.12	192.168.43.53	TCP	74	✓
1156	2.583506	3.9.68.12	192.168.43.53	TCP	74	✓
2050	3.517408	3.9.68.12	192.168.43.53	TCP	74	✓
2051	3.517412	3.9.68.12	192.168.43.53	TCP	74	✓
2108	4.833054	172.217.167.99	192.168.43.53	TCP	74	✓
2110	4.833336	172.217.167.99	192.168.43.53	TCP	74	✓
2115	4.847283	172.217.25.46	192.168.43.53	TCP	74	✓
2116	4.847286	74.125.200.188	192.168.43.53	TCP	74	✓
2117	4.847287	216.58.203.99	192.168.43.53	TCP	74	✓

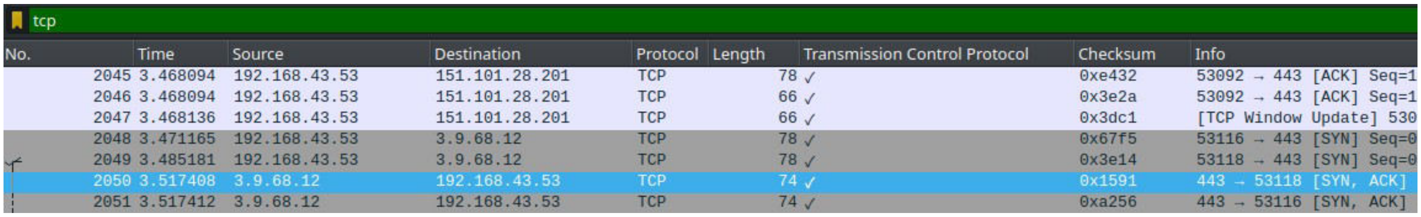
```
Frame 91: 74 bytes on wire (592 bits), 74 bytes captur
Ethernet II, Src: fa:e6:1a:de:25:c3 (fa:e6:1a:de:25:c3
Internet Protocol Version 4, Src: 151.101.28.201, Dst:
Transmission Control Protocol, Src Port: 443, Dst Port
   Source Port: 443
   Destination Port: 53108
   [Stream index: 9]
   [Conversation completeness: Incomplete, DATA (15)]
   [TCP Segment Len: 0]
   Sequence Number: 0    (relative sequence number)
   Sequence Number (raw): 2060397920
   [Next Sequence Number: 1    (relative sequence numb
   Acknowledgment Number: 1    (relative ack number)
   Acknowledgment number (raw): 921585441
```

```
0000  8c 85 90 93 ad f0 fa e6  1a de 25 c3
0010  00 3c 11 9f 40 00 fc 06  cd 10 97 65
0020  2b 35 01 bb cf 74 7a cf  2d 60 36 ee
0030  55 94 c6 e1 00 00 02 04  05 6c 01 03
0040  08 0a 27 ce f5 18 23 22  54 3c
```

Flags (12 bits) (tcp.flags), 2 bytes · Packets: 7241 · Displayed: 20 (0.3%) · Profile: Default

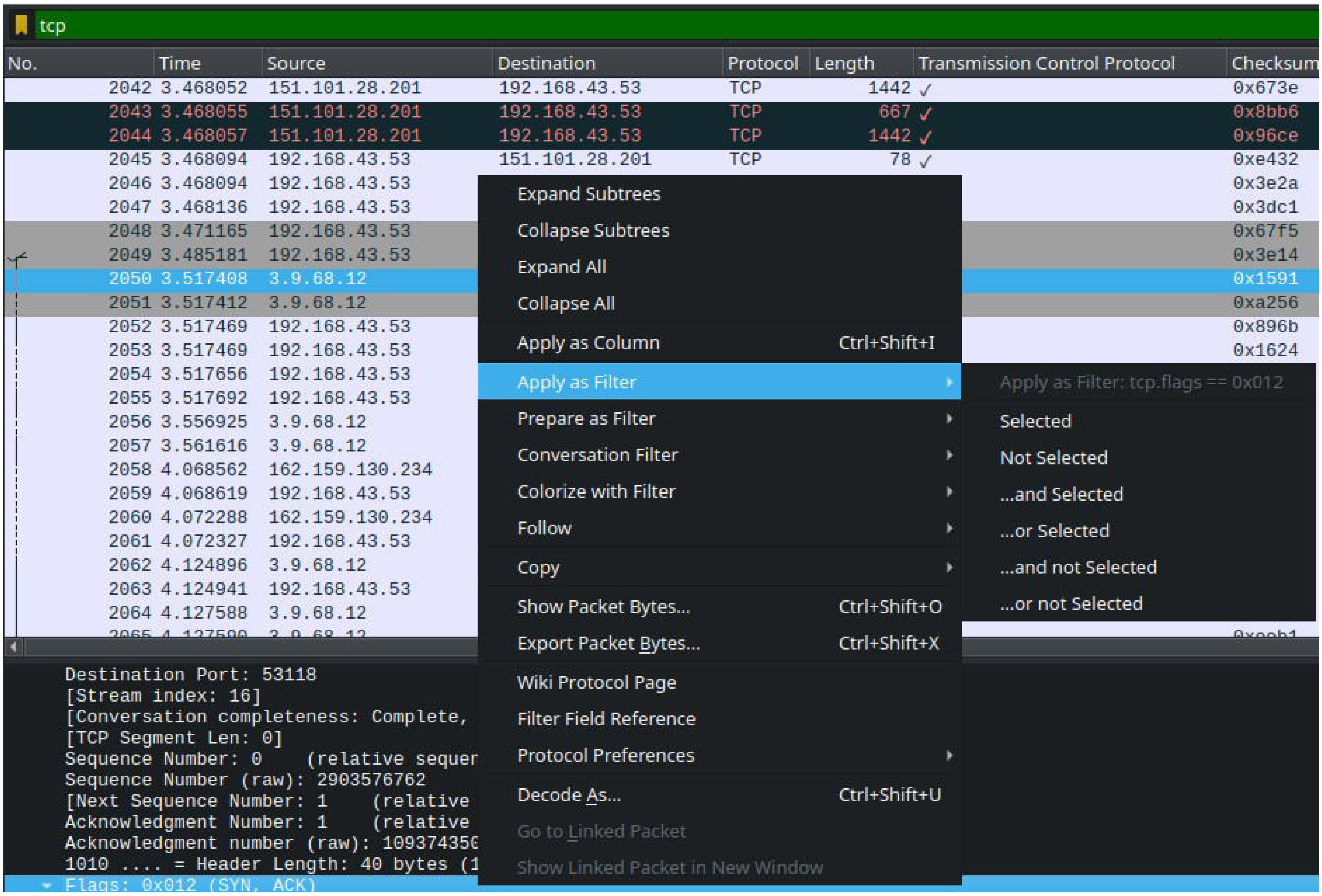
How to find the syntax of a property to search in a display filter?

Search a filter you know with respect to the question. To get the exact answer, find a packet with all the required properties and select on the property in packet header. Hover over the apply as filter option, you will the required syntax to search .



tcp

No.	Time	Source	Destination	Protocol	Length	Transmission Control Protocol	Checksum	Info
2045	3.468094	192.168.43.53	151.101.28.201	TCP	78	✓	0xe432	53092 → 443 [ACK] Seq=1
2046	3.468094	192.168.43.53	151.101.28.201	TCP	66	✓	0x3e2a	53092 → 443 [ACK] Seq=1
2047	3.468136	192.168.43.53	151.101.28.201	TCP	66	✓	0x3dc1	[TCP Window Update] 530
2048	3.471165	192.168.43.53	3.9.68.12	TCP	78	✓	0x67f5	53116 → 443 [SYN] Seq=0
2049	3.485181	192.168.43.53	3.9.68.12	TCP	78	✓	0x3e14	53118 → 443 [SYN] Seq=0
2050	3.517408	3.9.68.12	192.168.43.53	TCP	74	✓	0x1591	443 → 53118 [SYN, ACK]
2051	3.517412	3.9.68.12	192.168.43.53	TCP	74	✓	0xa256	443 → 53116 [SYN, ACK]



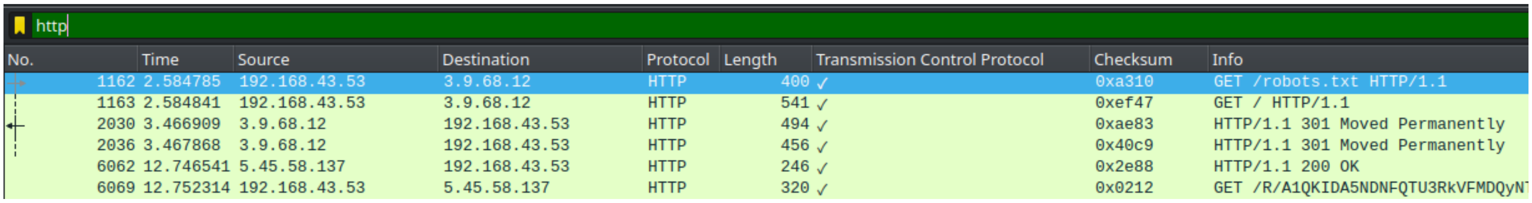
tcp

No.	Time	Source	Destination	Protocol	Length	Transmission Control Protocol	Checksum
2042	3.468052	151.101.28.201	192.168.43.53	TCP	1442	✓	0x673e
2043	3.468055	151.101.28.201	192.168.43.53	TCP	667	✓	0x8bb6
2044	3.468057	151.101.28.201	192.168.43.53	TCP	1442	✓	0x96ce
2045	3.468094	192.168.43.53	151.101.28.201	TCP	78	✓	0xe432
2046	3.468094	192.168.43.53					0x3e2a
2047	3.468136	192.168.43.53					0x3dc1
2048	3.471165	192.168.43.53					0x67f5
2049	3.485181	192.168.43.53					0x3e14
2050	3.517408	3.9.68.12					0x1591
2051	3.517412	3.9.68.12					0xa256
2052	3.517469	192.168.43.53					0x896b
2053	3.517469	192.168.43.53					0x1624
2054	3.517656	192.168.43.53					
2055	3.517692	192.168.43.53					
2056	3.556925	3.9.68.12					
2057	3.561616	3.9.68.12					
2058	4.068562	162.159.130.234					
2059	4.068619	192.168.43.53					
2060	4.072288	162.159.130.234					
2061	4.072327	192.168.43.53					
2062	4.124896	3.9.68.12					
2063	4.124941	192.168.43.53					
2064	4.127588	3.9.68.12					

```
Destination Port: 53118
[Stream index: 16]
[Conversation completeness: Complete,
[TCP Segment Len: 0]
Sequence Number: 0    (relative sequer
Sequence Number (raw): 2903576762
[Next Sequence Number: 1    (relative
Acknowledgment Number: 1    (relative
Acknowledgment number (raw): 109374356
1010 .... = Header Length: 40 bytes (1
Flags: 0x012 (SYN, ACK)
```

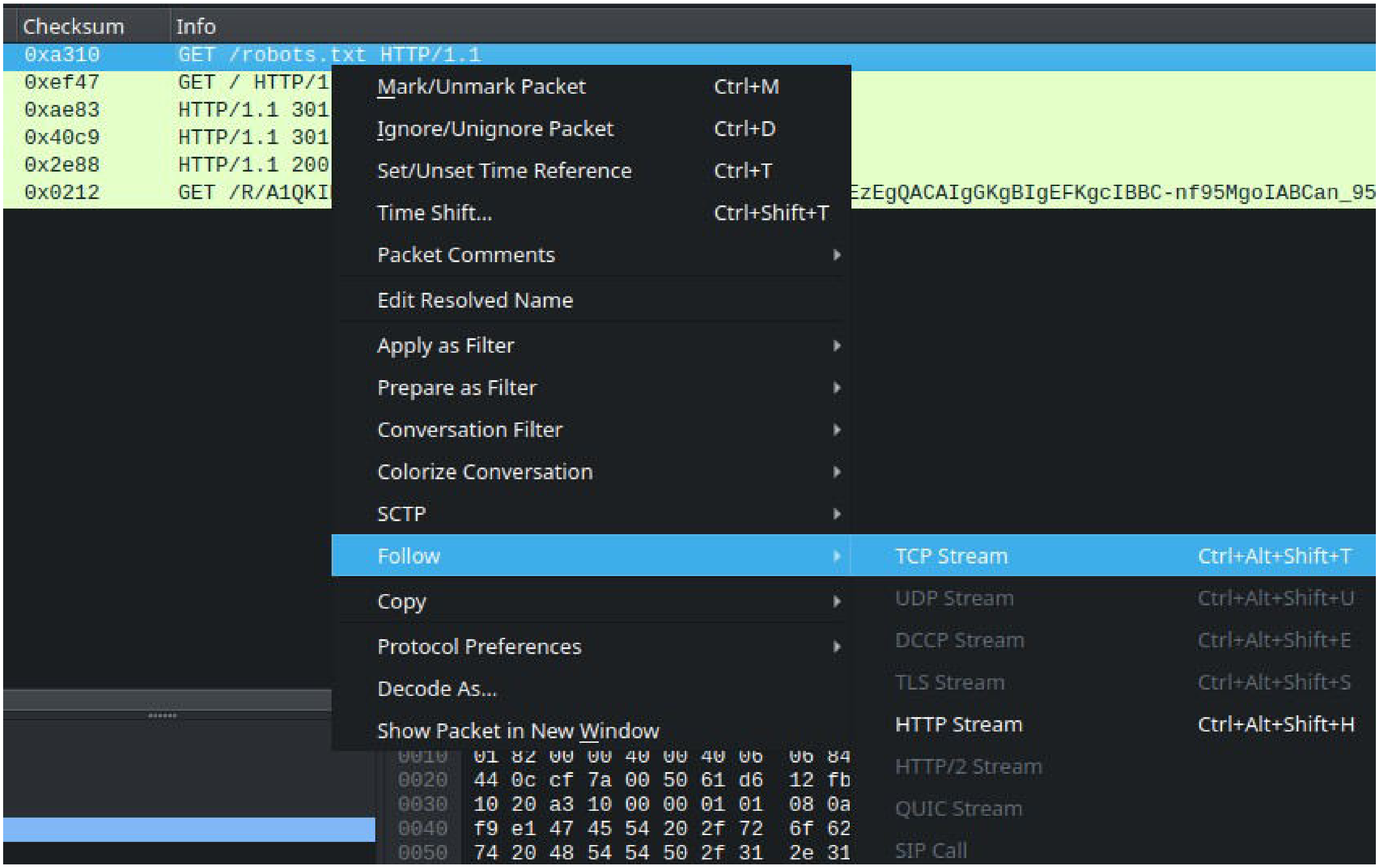
3.Which version of Chrome was used to connect to securityblue.team?

Search for ‘http’ in the display filter. Right click on the the any of the packets with get method in the info column. Select follow and tcp stream options after right clicking the packet. Now the pop-up window contains the answer to the question. The version of the chrome is 80.0.3987.87.



http

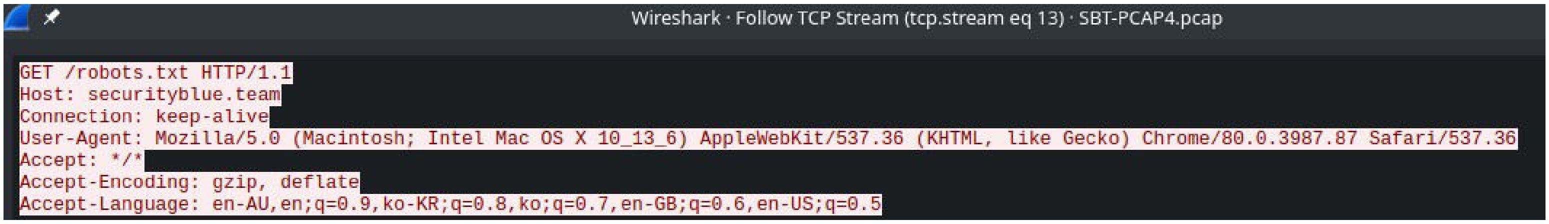
No.	Time	Source	Destination	Protocol	Length	Transmission Control Protocol	Checksum	Info
1162	2.584785	192.168.43.53	3.9.68.12	HTTP	400	✓	0xa310	GET /robots.txt HTTP/1.1
1163	2.584841	192.168.43.53	3.9.68.12	HTTP	541	✓	0xef47	GET / HTTP/1.1
2030	3.466909	3.9.68.12	192.168.43.53	HTTP	494	✓	0xae83	HTTP/1.1 301 Moved Permanently
2036	3.467868	3.9.68.12	192.168.43.53	HTTP	456	✓	0x40c9	HTTP/1.1 301 Moved Permanently
6062	12.746541	5.45.58.137	192.168.43.53	HTTP	246	✓	0x2e88	HTTP/1.1 200 OK
6069	12.752314	192.168.43.53	5.45.58.137	HTTP	320	✓	0x0212	GET /R/A1QKIDA5NDNFQTU3RkVFMDQyN



Checksum	Info
0xa310	GET /robots.txt HTTP/1.1
0xef47	GET / HTTP/1
0xae83	HTTP/1.1 301
0x40c9	HTTP/1.1 301
0x2e88	HTTP/1.1 200
0x0212	GET /R/A1QKI EzEgQACAIgGKgBIgEFKgcIBBC-nf95MgoIABCan_95

```
0010  01 82 00 00 40 00 40 06   06 84
0020  44 0c cf 7a 00 50 61 d6   12 fb
0030  10 20 a3 10 00 00 01 01   08 0a
0040  f9 e1 47 45 54 20 2f 72   6f 62
0050  74 20 48 54 54 50 2f 31   2e 31
```

Mark/Unmark Packet Ctrl+M
Ignore/Unignore Packet Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comments
Edit Resolved Name
Apply as Filter
Prepare as Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As...
Show Packet in New Window
TCP Stream Ctrl+Alt+Shift+T
UDP Stream Ctrl+Alt+Shift+U
DCCP Stream Ctrl+Alt+Shift+E
TLS Stream Ctrl+Alt+Shift+S
HTTP Stream Ctrl+Alt+Shift+H
HTTP/2 Stream
QUIC Stream
SIP Call

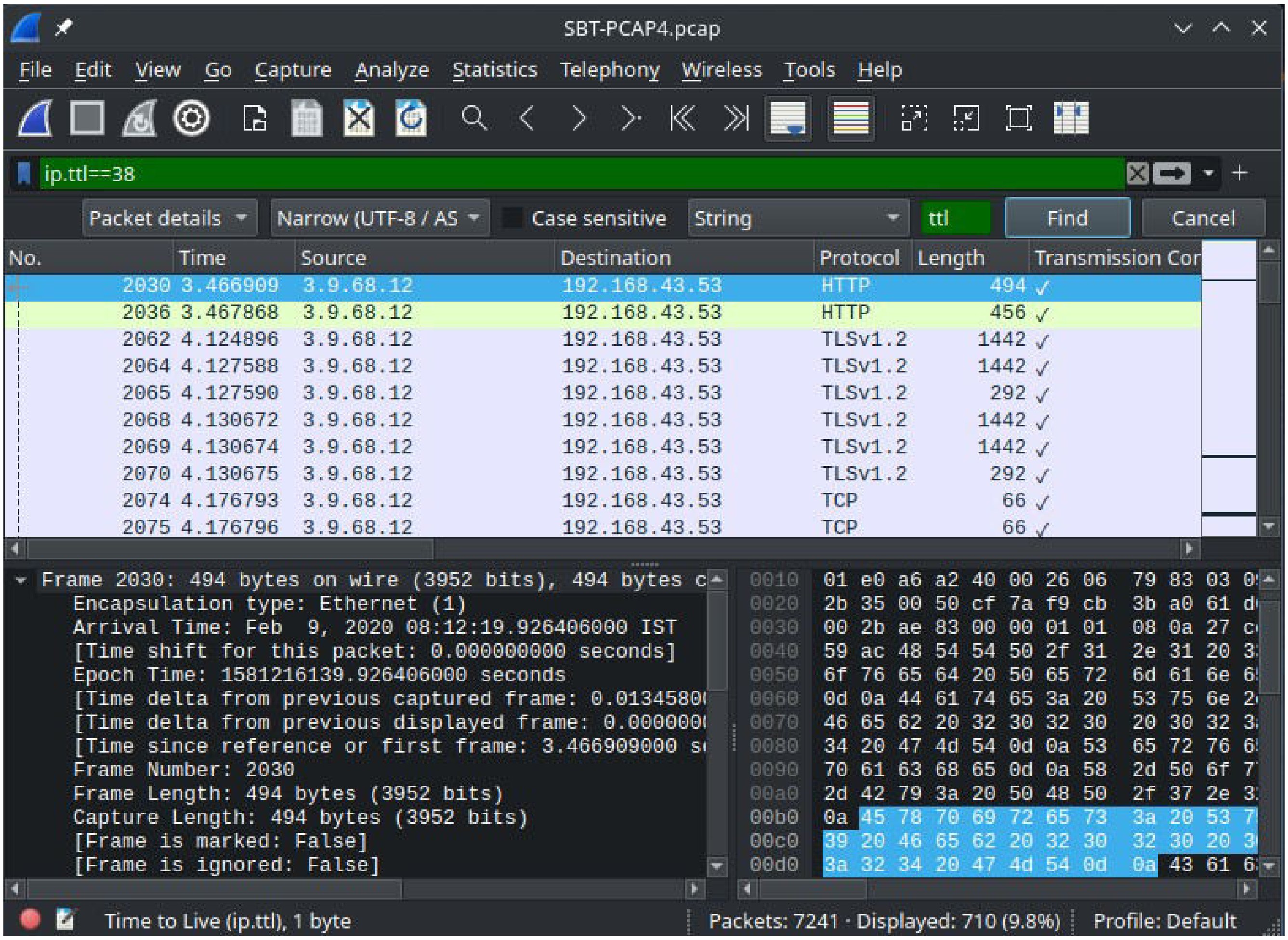


Wireshark · Follow TCP Stream (tcp.stream eq 13) · SBT-PCAP4.pcap

```
GET /robots.txt HTTP/1.1
Host: securityblue.team
Connection: keep-alive
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.87 Safari/537.36
Accept: */*
Accept-Encoding: gzip, deflate
Accept-Language: en-AU,en;q=0.9,ko-KR;q=0.8,ko;q=0.7,en-GB;q=0.6,en-US;q=0.5
```

4.How many packets have a TTL value of 38?

Apply the filter ‘ip.ttl==38’ and check out the number of packets displayed. In this case it will be 710.



ip.ttl==38

ttl

```
No.     Time      Source      Destination    Protocol Length Transmission Cor
   2030 3.466909  3.9.68.12   192.168.43.53  HTTP        494 ✓
   2036 3.467868  3.9.68.12   192.168.43.53  HTTP        456 ✓
   2062 4.124896  3.9.68.12   192.168.43.53  TLSv1.2    1442 ✓
   2064 4.127588  3.9.68.12   192.168.43.53  TLSv1.2    1442 ✓
   2065 4.127590  3.9.68.12   192.168.43.53  TLSv1.2     292 ✓
   2068 4.130672  3.9.68.12   192.168.43.53  TLSv1.2    1442 ✓
   2069 4.130674  3.9.68.12   192.168.43.53  TLSv1.2    1442 ✓
   2070 4.130675  3.9.68.12   192.168.43.53  TLSv1.2     292 ✓
   2074 4.176793  3.9.68.12   192.168.43.53  TCP          66 ✓
   2075 4.176796  3.9.68.12   192.168.43.53  TCP          66 ✓
```

```
Frame 2030: 494 bytes on wire (3952 bits), 494 bytes c
    Encapsulation type: Ethernet (1)
    Arrival Time: Feb  9, 2020 08:12:19.926406000 IST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1581216139.926406000 seconds
    [Time delta from previous captured frame: 0.01345800
    [Time delta from previous displayed frame: 0.0000000
    [Time since reference or first frame: 3.466909000 se
    Frame Number: 2030
    Frame Length: 494 bytes (3952 bits)
    Capture Length: 494 bytes (3952 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
```

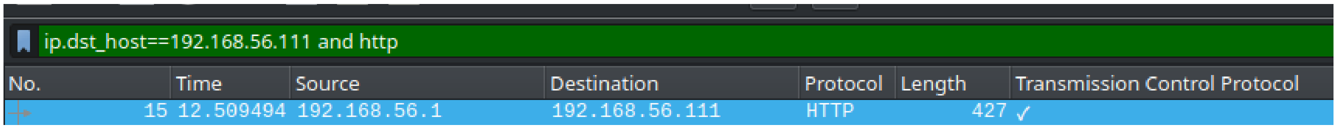
```
0010  01 e0 a6 a2 40 00 26 06  79 83 03 0
0020  2b 35 00 50 cf 7a f9 cb  3b a0 61 d
0030  00 2b ae 83 00 00 01 01  08 0a 27 c
0040  59 ac 48 54 54 50 2f 31  2e 31 20 3
0050  6f 76 65 64 20 50 65 72  6d 61 6e 6
0060  0d 0a 44 61 74 65 3a 20  53 75 6e 2
0070  46 65 62 20 32 30 32 30  20 30 32 3
0080  34 20 47 4d 54 0d 0a 53  65 72 76 6
0090  70 61 63 68 65 0d 0a 58  2d 50 6f 7
00a0  2d 42 79 3a 20 50 48 50  2f 37 2e 3
00b0  0a 45 78 70 69 72 65 73  3a 20 53 7
00c0  39 20 46 65 62 20 32 30  32 30 20 3
00d0  3a 32 34 20 47 4d 54 0d  0a 43 61 6
```

Time to Live (ip.ttl), 1 byte | Packets: 7241 · Displayed: 710 (9.8%) | Profile: Default

PCAP 5

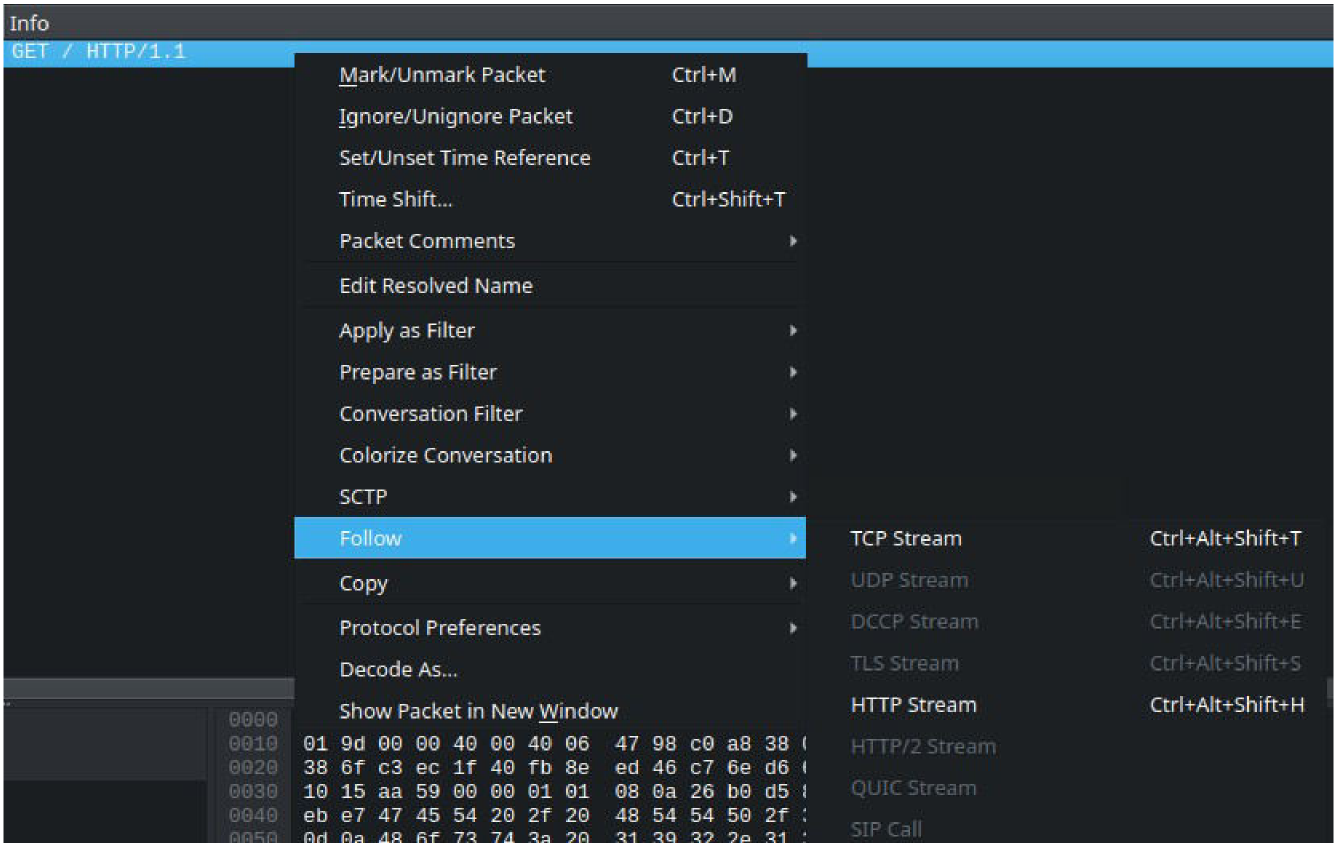
1.What is the name of the PNG file on the webserver at 192.168.56.111?

Search for ‘ip.dst_host==192.168.56.111’ in the display filter and it should have http protocol. Select the follow and tcp stream option to get human readable format of the that conversation between client and server. The name of PNG file is proprietary.png.



ip.dst_host==192.168.56.111 and http

No.	Time	Source	Destination	Protocol	Length	Transmission Control Protocol
15	12.509494	192.168.56.1	192.168.56.111	HTTP	427	✓

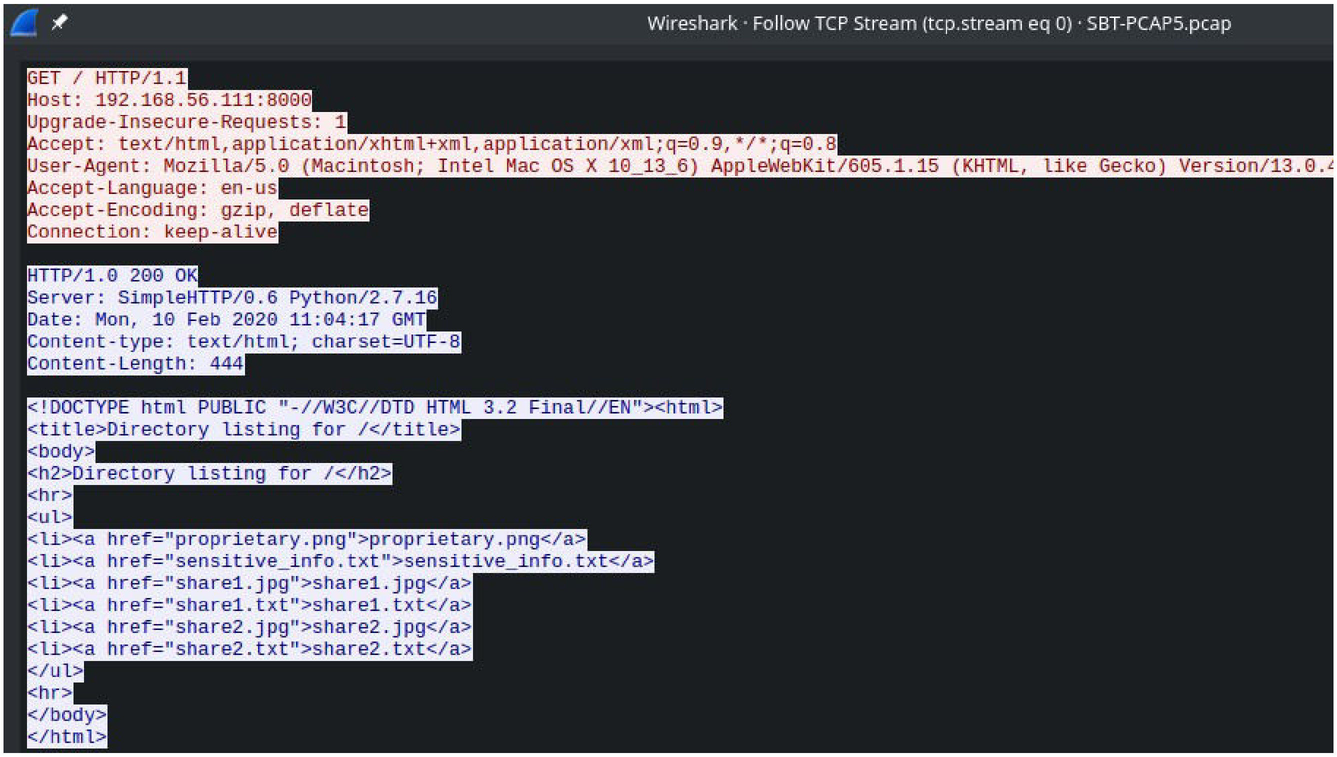


Info

GET / HTTP/1.1

Mark/Unmark Packet Ctrl+M
Ignore/Unignore Packet Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comments
Edit Resolved Name
Apply as Filter
Prepare as Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As...
Show Packet in New Window
TCP Stream Ctrl+Alt+Shift+T
UDP Stream Ctrl+Alt+Shift+U
DCCP Stream Ctrl+Alt+Shift+E
TLS Stream Ctrl+Alt+Shift+S
HTTP Stream Ctrl+Alt+Shift+H
HTTP/2 Stream
QUIC Stream
SIP Call

```
0000
0010  01 9d 00 00 40 00 40 06  47 98 c0 a8 38
0020  38 6f c3 ec 1f 40 fb 8e  ed 46 c7 6e d6
0030  10 15 aa 59 00 00 01 01  08 0a 26 b0 d5
0040  eb e7 47 45 54 20 2f 20  48 54 54 50 2f
0050  0d 0a 48 6f 73 74 3a 20  31 39 32 2e 31
```


Wireshark · Follow TCP Stream (tcp.stream eq 0) · SBT-PCAP5.pcap

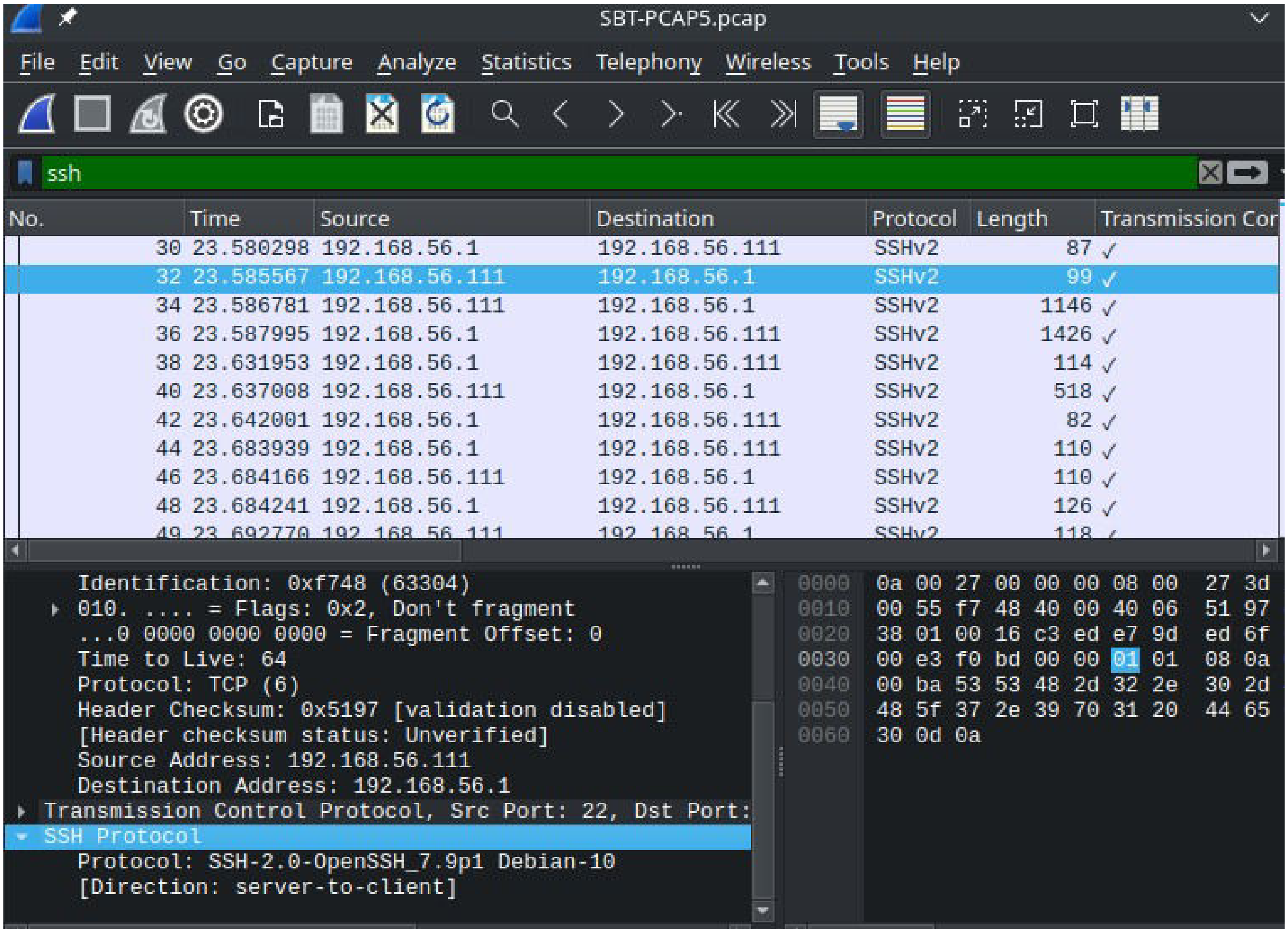
```
GET / HTTP/1.1
Host: 192.168.56.111:8000
Upgrade-Insecure-Requests: 1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/13.0.4
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Connection: keep-alive

HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/2.7.16
Date: Mon, 10 Feb 2020 11:04:17 GMT
Content-type: text/html; charset=UTF-8
Content-Length: 444

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 3.2 Final//EN"><html>
<title>Directory listing for /</title>
<body>
<h2>Directory listing for /</h2>
<hr>
<ul>
<li><a href="proprietary.png">proprietary.png</a>
<li><a href="sensitive_info.txt">sensitive_info.txt</a>
<li><a href="share1.jpg">share1.jpg</a>
<li><a href="share1.txt">share1.txt</a>
<li><a href="share2.jpg">share2.jpg</a>
<li><a href="share2.txt">share2.txt</a>
</ul>
<hr>
</body>
</html>
```

2. Which version of OpenSSH is running on the server?

Search ssh in the display filter. Check for the server packet in the display filter. Select the server packet and move to header packet window of the corresponding packet. Under the ssh protocol header, find the version of openssh running on the server. 7.9p1is the version of the ssh server.


SBT-PCAP5.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ssh

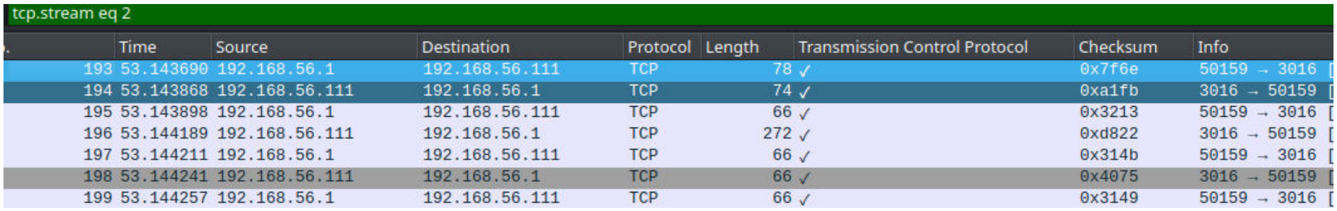
No.	Time	Source	Destination	Protocol	Length	Transmission Cor
30	23.580298	192.168.56.1	192.168.56.111	SSHv2	87	✓
32	23.585567	192.168.56.111	192.168.56.1	SSHv2	99	✓
34	23.586781	192.168.56.111	192.168.56.1	SSHv2	1146	✓
36	23.587995	192.168.56.1	192.168.56.111	SSHv2	1426	✓
38	23.631953	192.168.56.1	192.168.56.111	SSHv2	114	✓
40	23.637008	192.168.56.111	192.168.56.1	SSHv2	518	✓
42	23.642001	192.168.56.1	192.168.56.111	SSHv2	82	✓
44	23.683939	192.168.56.1	192.168.56.111	SSHv2	110	✓
46	23.684166	192.168.56.111	192.168.56.1	SSHv2	110	✓
48	23.684241	192.168.56.1	192.168.56.111	SSHv2	126	✓
49	23.692770	192.168.56.111	192.168.56.1	SSHv2	118	✓

```
    Identification: 0xf748 (63304)
  ▸ 010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: TCP (6)
    Header Checksum: 0x5197 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.56.111
    Destination Address: 192.168.56.1
▸ Transmission Control Protocol, Src Port: 22, Dst Port:
▾ SSH Protocol
    Protocol: SSH-2.0-OpenSSH_7.9p1 Debian-10
    [Direction: server-to-client]
```

```
0000  0a 00 27 00 00 00 08 00  27 3d
0010  00 55 f7 48 40 00 40 06  51 97
0020  38 01 00 16 c3 ed e7 9d  ed 6f
0030  00 e3 f0 bd 00 00 01 01  08 0a
0040  00 ba 53 53 48 2d 32 2e  30 2d
0050  48 5f 37 2e 39 70 31 20  44 65
0060  30 0d 0a
```

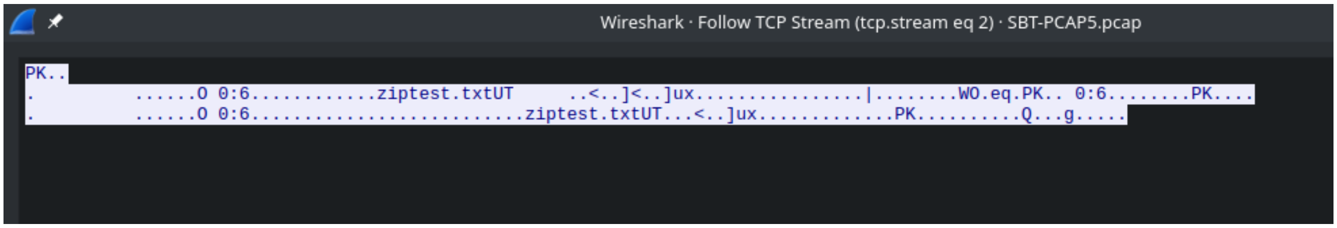
3.On which port is the .zip file being served?

Go to the conversations property of the statistics tab in the main menu. Select the tcp tab in the pop up window. Reject the port 22 directly because it is the standard port of ssh. Now left with two ports, check out of these ports by apply these ports as filter and read the conversation between them using follow and tcp stream option. After this process we will see the port 3016 serves the .zip file.



tcp.stream eq 2

.	Time	Source	Destination	Protocol	Length	Transmission Control Protocol	Checksum	Info
193	53.143690	192.168.56.1	192.168.56.111	TCP	78	✓	0x7f6e	50159 → 3016 [
194	53.143868	192.168.56.111	192.168.56.1	TCP	74	✓	0xa1fb	3016 → 50159 [
195	53.143898	192.168.56.1	192.168.56.111	TCP	66	✓	0x3213	50159 → 3016 [
196	53.144189	192.168.56.111	192.168.56.1	TCP	272	✓	0xd822	3016 → 50159 [
197	53.144211	192.168.56.1	192.168.56.111	TCP	66	✓	0x314b	50159 → 3016 [
198	53.144241	192.168.56.111	192.168.56.1	TCP	66	✓	0x4075	3016 → 50159 [
199	53.144257	192.168.56.1	192.168.56.111	TCP	66	✓	0x3149	50159 → 3016 [



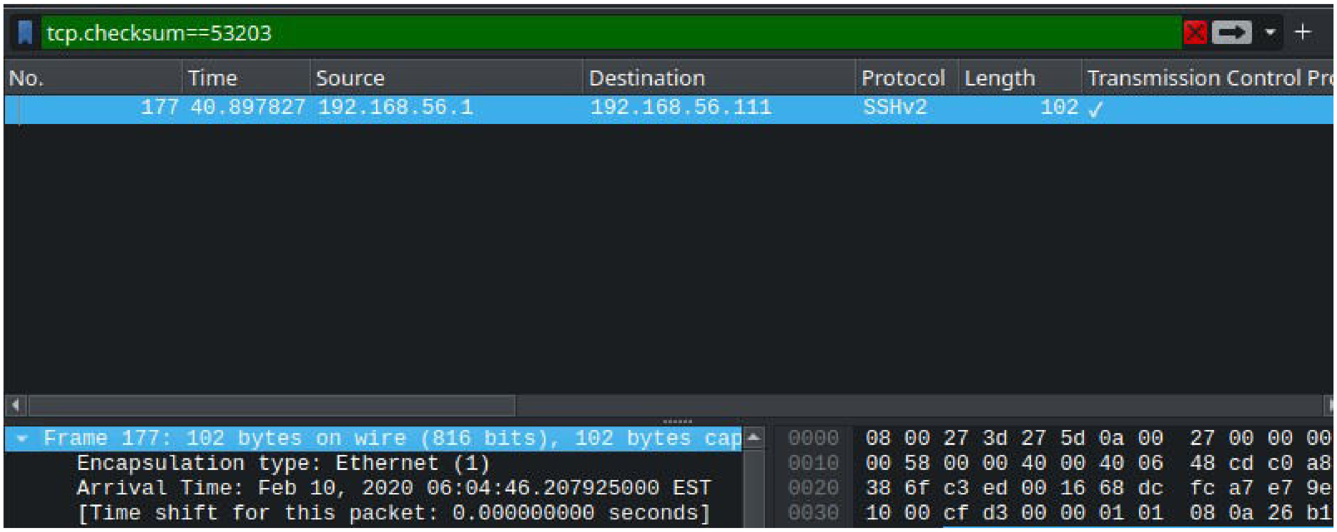
Wireshark · Follow TCP Stream (tcp.stream eq 2) · SBT-PCAP5.pcap

```
PK..
.          ......0 0:6...........ziptest.txtUT     ..<..]<..]ux...............|........WO.eq.PK.. 0:6........PK....
.          ......0 0:6.........................ziptest.txtUT...<..]ux.............PK..........Q...g.....
```

4.When was a packet with a TCP checksum value of 53203 captured? (Format: xx:xx:xx.xxxxxx)

Apply the filter tcp.checksum==53203. In the packet headaer window under frames header check the arrival time of the time of the packet. The arrival of the packet is 06:04:46.207925.

Note: If you are using the wireshark application mostly you will get the time in IST but in case if you are using kali linux's wireshark then you will have the arrival time is EST which is time standard followed. For those whom it is IST convert it to EST.(You can change you time zone in your pc/laptop



tcp.checksum==53203

```
No.     Time       Source          Destination      Protocol  Length  Transmission Control Pr
    177 40.897827  192.168.56.1    192.168.56.111   SSHv2        102  ✓
```

```
▾ Frame 177: 102 bytes on wire (816 bits), 102 bytes cap
    Encapsulation type: Ethernet (1)
    Arrival Time: Feb 10, 2020 06:04:46.207925000 EST
    [Time shift for this packet: 0.000000000 seconds]
```

```
0000  08 00 27 3d 27 5d 0a 00  27 00 00 00
0010  00 58 00 00 40 00 40 06  48 cd c0 a8
0020  38 6f c3 ed 00 16 68 dc  fc a7 e7 9e
0030  10 00 cf d3 00 00 01 01  08 0a 26 b1
```