Mawlana Bhashani Science & Technology University

Lab report no : 08

Lab report on : Lab-wireshark display lecture.

Course Code : ICT3208

Course Title : Computer Network Lab

Submitted by	**Submitted To**
Name : Maskur Al Shal sabil ID: IT18021 Session : 2017-2018 Dept of ICT MBSTU	Name : Nazrul Islam Assistant Professor Dept of ICT MBSTU

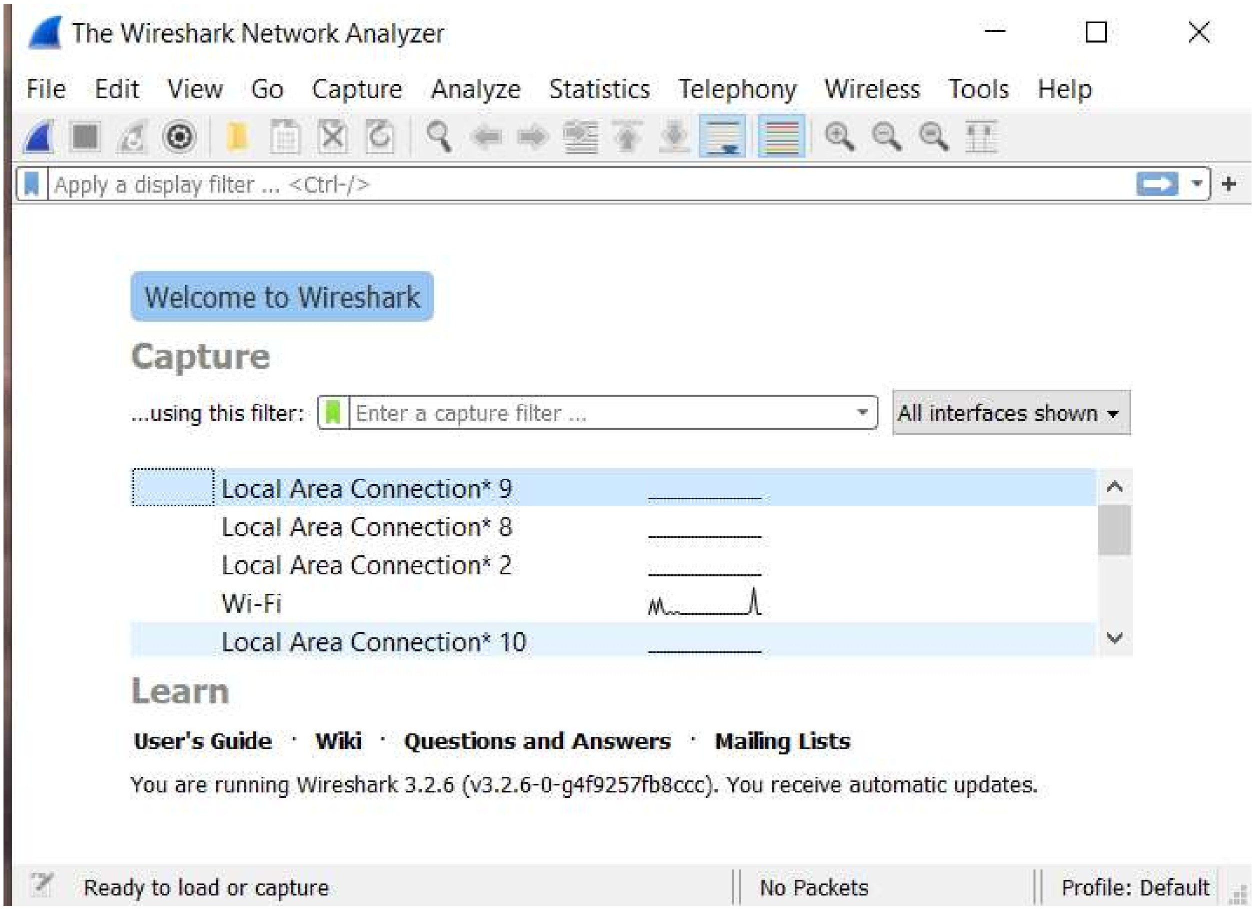
Wireshark :

This is a network protocol analyzer which use to

- ➔ Capture the network packets
- ➔ To display the details about the packet
- ➔ Troubleshooting network problems
- ➔ Learning network protocol internals

To install We can download and install it from its official website : https://www.wireshark.org

After installation : when we run the wireshark


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Local Area Connection* 9
Local Area Connection* 8
Local Area Connection* 2
Wi-Fi
Local Area Connection* 10
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
Ready to load or capture
No Packets
Profile: Default

You are running Wireshark 3.2.6 (v3.2.6-0-g4f9257fb8ccc). You receive automatic updates.

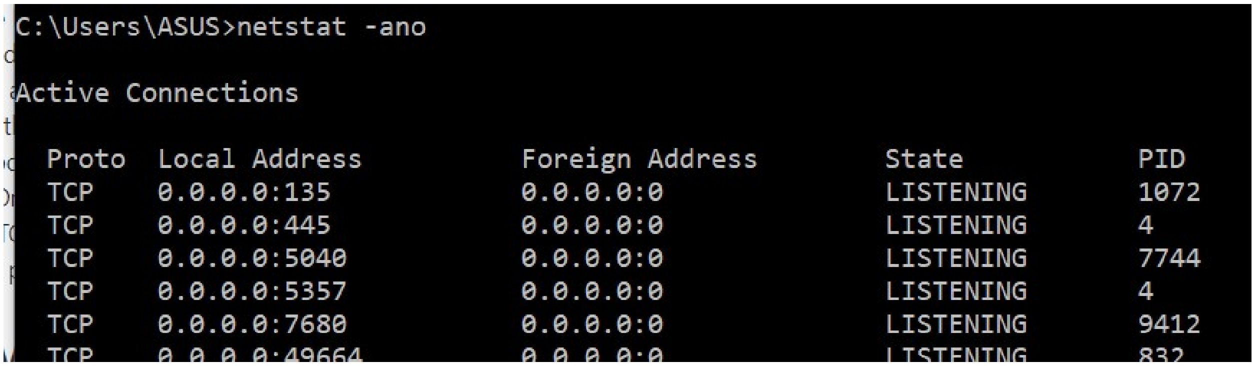
Before go to wireshark lets take a analogy on ip and port number : so In previous time we used post office to send a letter to our friend or someone else . so we need to add our address and the destination address . So in networking this is the Ip address . Now when the letter delivered to our friend the postman go to his door to serve the letter. Here his door is the port .

So to check which port are used in the our computer in windows 10 we can follow the process

Step 1 : windows +r and type cmd and then enter

The command prompt will appear

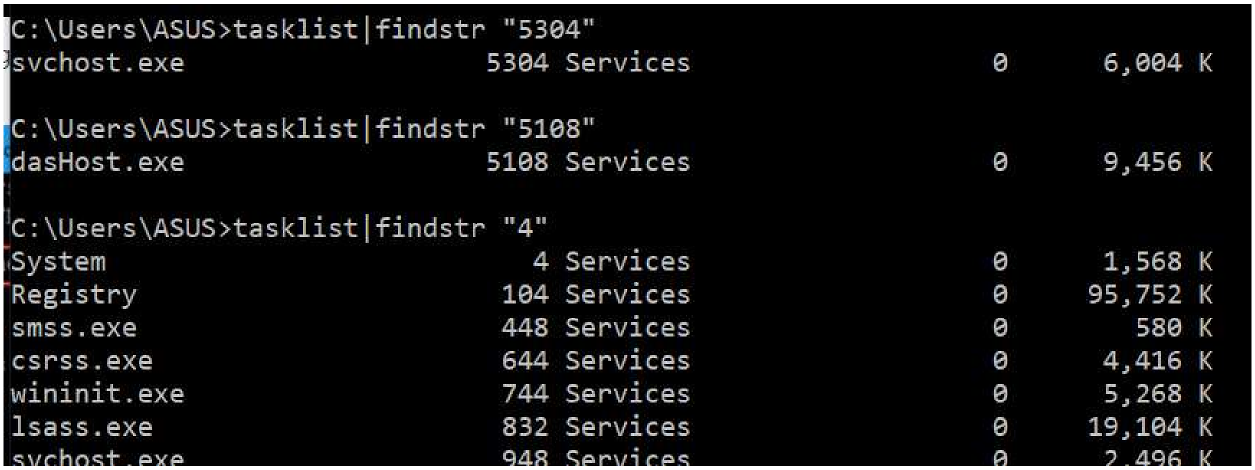
Step2 – type netstat   -ano to list all ports



```
C:\Users\ASUS>netstat -ano

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       1072
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING       7744
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING       9412
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       832
```

Step3 -> to locate the targer pid :

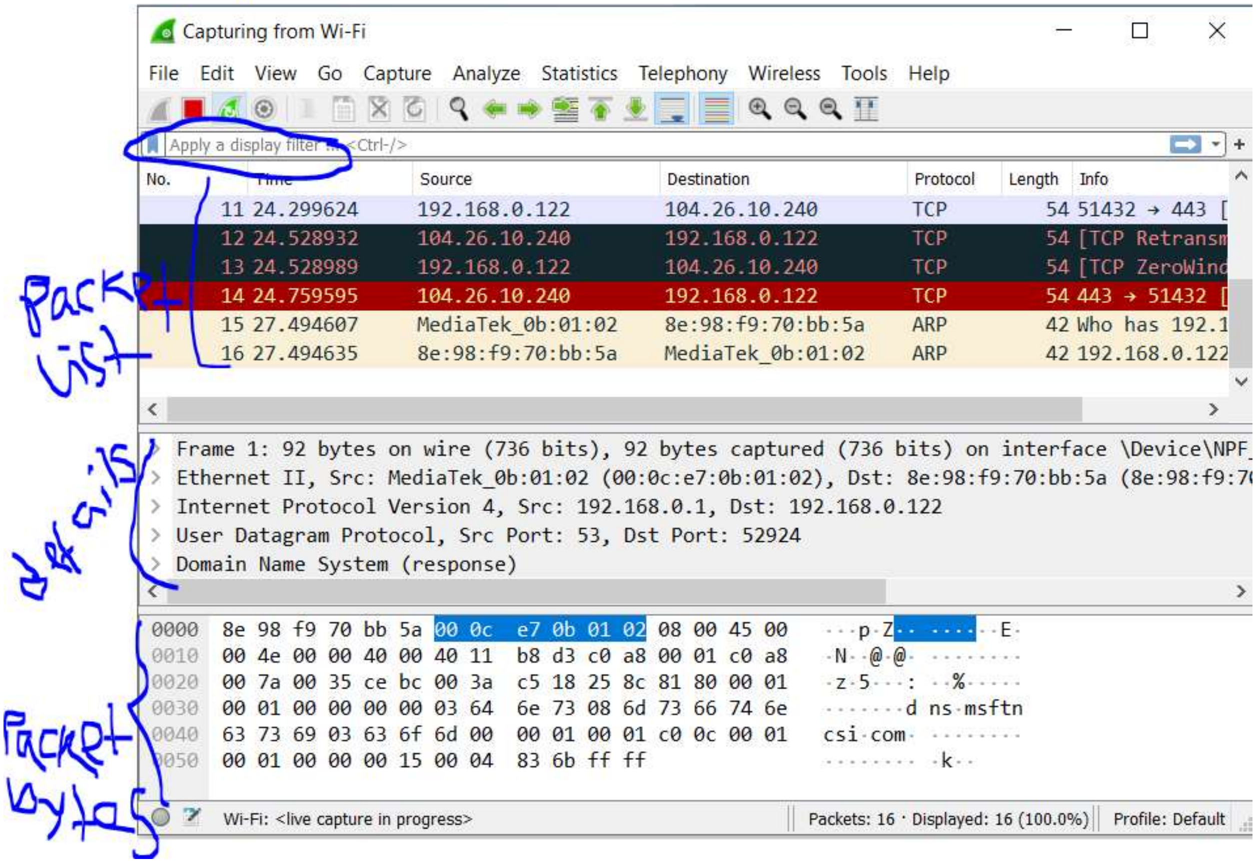
Type : tasklist | findstr “pid number” and hit enter



```
C:\Users\ASUS>tasklist|findstr "5304"
svchost.exe                   5304 Services                   0      6,004 K

C:\Users\ASUS>tasklist|findstr "5108"
dasHost.exe                   5108 Services                   0      9,456 K

C:\Users\ASUS>tasklist|findstr "4"
System                           4 Services                   0      1,568 K
Registry                       104 Services                   0     95,752 K
smss.exe                       448 Services                   0        580 K
csrss.exe                      644 Services                   0      4,416 K
wininit.exe                    744 Services                   0      5,268 K
lsass.exe                      832 Services                   0     19,104 K
svchost.exe                    948 Services                   0      2,496 K
```

To end up this service, run taskkill /f /t /im vmms.exe.

Main window :


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Packet list
details
Packet bytes
Wi-Fi: <live capture in progress>
Packets: 16 · Displayed: 16 (100.0%)
Profile: Default

```
No.  Time       Source              Destination         Protocol  Length  Info
  11 24.299624  192.168.0.122       104.26.10.240       TCP           54  51432 → 443 [
  12 24.528932  104.26.10.240       192.168.0.122       TCP           54  [TCP Retransm
  13 24.528989  192.168.0.122       104.26.10.240       TCP           54  [TCP ZeroWinc
  14 24.759595  104.26.10.240       192.168.0.122       TCP           54  443 → 51432 [
  15 27.494607  MediaTek_0b:01:02   8e:98:f9:70:bb:5a   ARP           42  Who has 192.1
  16 27.494635  8e:98:f9:70:bb:5a   MediaTek_0b:01:02   ARP           42  192.168.0.122
```

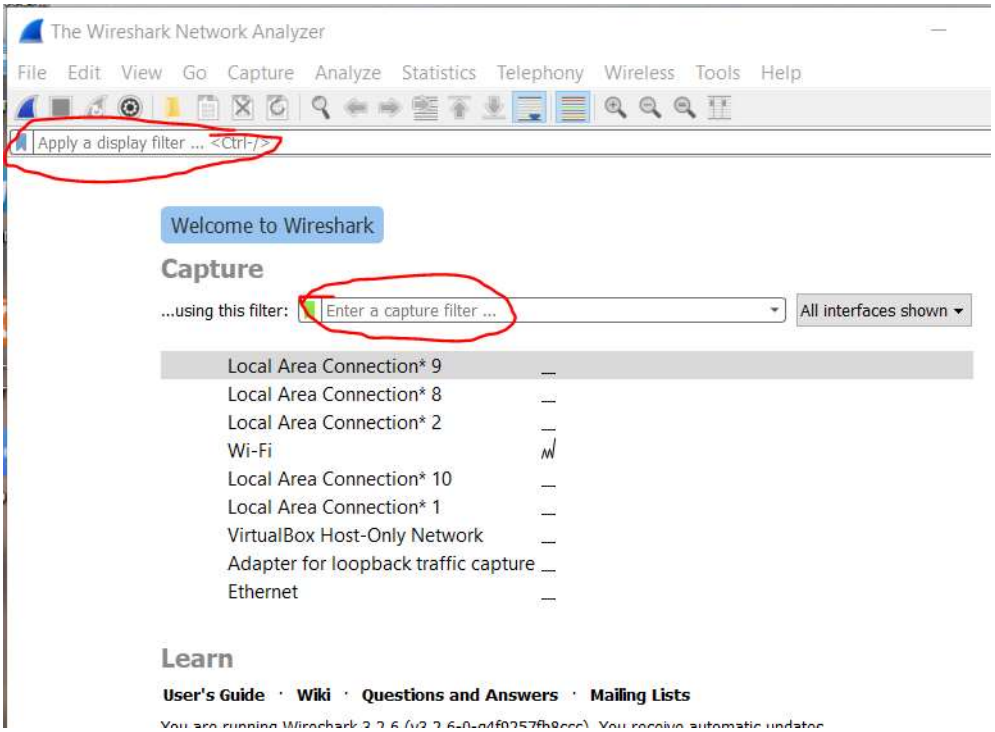
```
> Frame 1: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface \Device\NPF_
> Ethernet II, Src: MediaTek_0b:01:02 (00:0c:e7:0b:01:02), Dst: 8e:98:f9:70:bb:5a (8e:98:f9:7
> Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.122
> User Datagram Protocol, Src Port: 53, Dst Port: 52924
> Domain Name System (response)
```

```
0000  8e 98 f9 70 bb 5a 00 0c  e7 0b 01 02 08 00 45 00   ···p·Z·· ······E·
0010  00 4e 00 00 40 00 40 11  b8 d3 c0 a8 00 01 c0 a8   ·N··@·@· ········
0020  00 7a 00 35 ce bc 00 3a  c5 18 25 8c 81 80 00 01   ·z·5···: ··%·····
0030  00 01 00 00 00 00 03 64  6e 73 08 6d 73 66 74 6e   ·······d ns·msftn
0040  63 73 69 03 63 6f 6d 00  00 01 00 01 c0 0c 00 01   csi·com· ········
0050  00 01 00 00 00 15 00 04  83 6b ff ff               ········ ·k··
```

Filter : There is two filters in wireshark . The one is display filter and another one is capture filter .

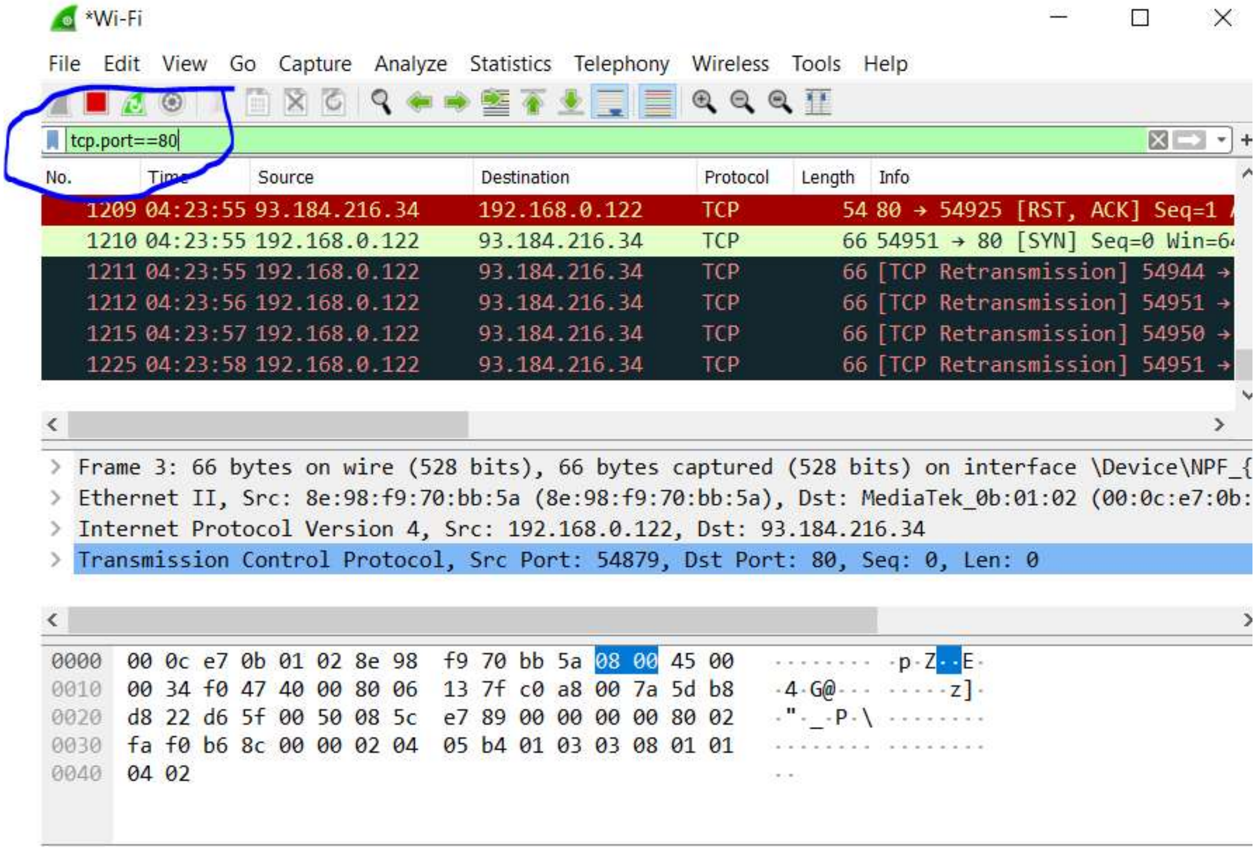
Display filter use the command like(tcp.port==80)

Capture filter use the command like(tcp port 80)


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Local Area Connection* 9
Local Area Connection* 8
Local Area Connection* 2
Wi-Fi
Local Area Connection* 10
Local Area Connection* 1
VirtualBox Host-Only Network
Adapter for loopback traffic capture
Ethernet
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
You are running Wireshark 3.2.6 (v3.2.6-0-g4f9257fb8ccc). You receive automatic updates

Display filter :

Show only tcp protocol on specific port


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port==80

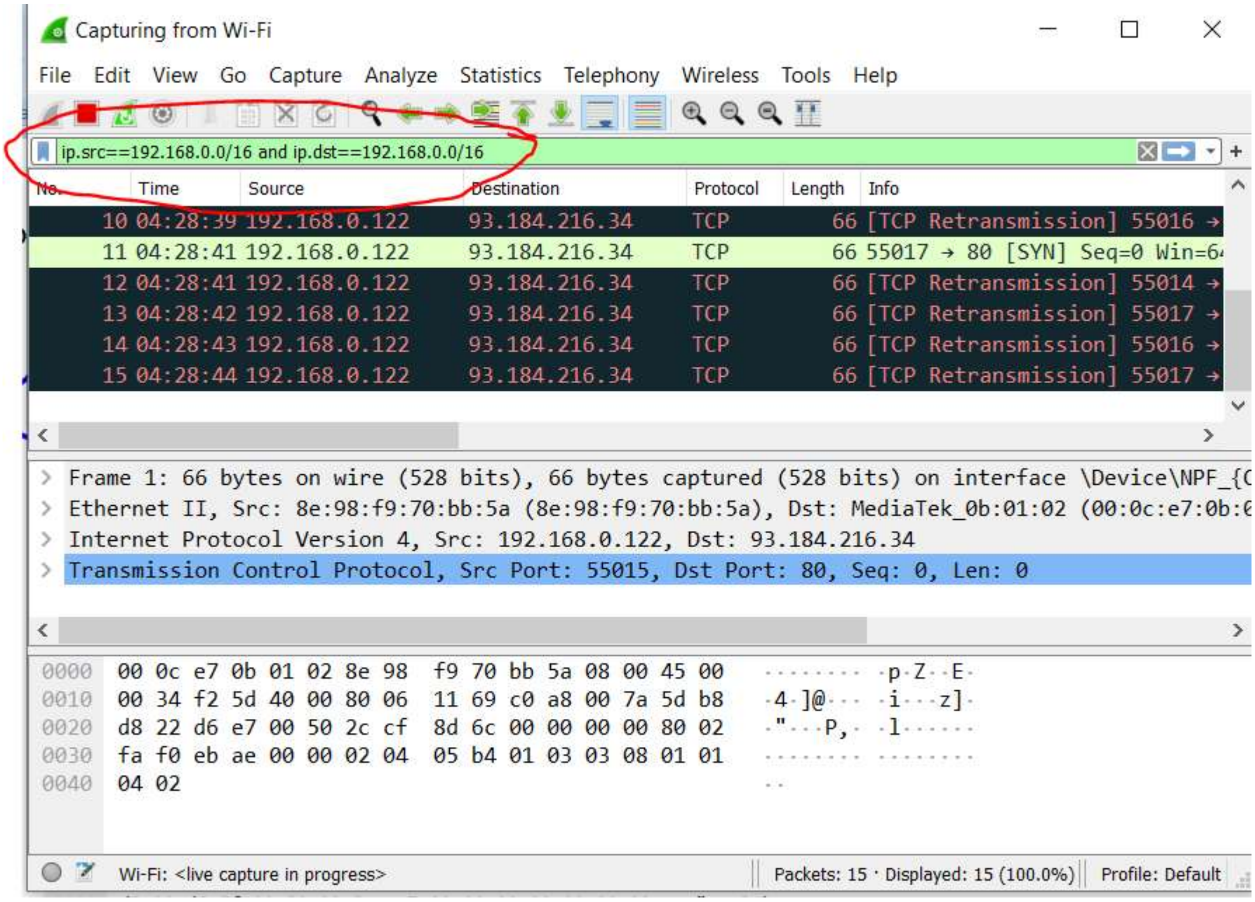
```
No.  Time     Source          Destination     Protocol Length Info
1209 04:23:55 93.184.216.34   192.168.0.122   TCP      54 80 → 54925 [RST, ACK] Seq=1 A
1210 04:23:55 192.168.0.122   93.184.216.34   TCP      66 54951 → 80 [SYN] Seq=0 Win=64
1211 04:23:55 192.168.0.122   93.184.216.34   TCP      66 [TCP Retransmission] 54944 →
1212 04:23:56 192.168.0.122   93.184.216.34   TCP      66 [TCP Retransmission] 54951 →
1215 04:23:57 192.168.0.122   93.184.216.34   TCP      66 [TCP Retransmission] 54950 →
1225 04:23:58 192.168.0.122   93.184.216.34   TCP      66 [TCP Retransmission] 54951 →
```

```
> Frame 3: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b:
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 54879, Dst Port: 80, Seq: 0, Len: 0
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 f0 47 40 00 80 06  13 7f c0 a8 00 7a 5d b8   ·4·G@··· ·····z]·
0020  d8 22 d6 5f 00 50 08 5c  e7 89 00 00 00 00 80 02   ·"·_·P·\ ········
0030  fa f0 b6 8c 00 00 02 04  05 b4 01 03 03 08 01 01   ········ ········
0040  04 02                                              ··
```

Show only traffic in the LAN (192.168.x.x), between workstations and servers -- no Internet:

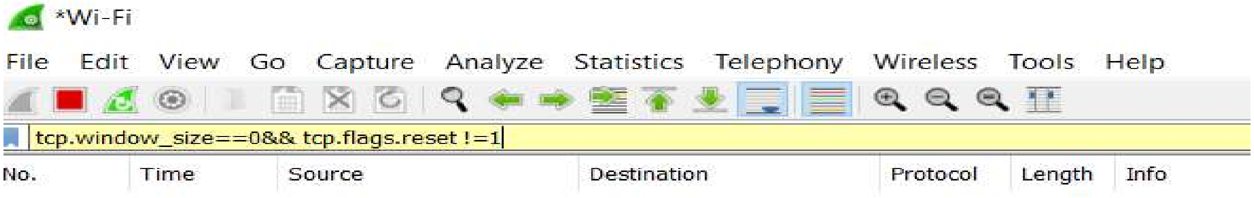
- `ip.src==192.168.0.0/16 and ip.dst==192.168.0.0/16`


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.src==192.168.0.0/16 and ip.dst==192.168.0.0/16

No.   Time      Source          Destination    Protocol  Length Info
10 04:28:39 192.168.0.122   93.184.216.34   TCP       66 [TCP Retransmission] 55016 →
11 04:28:41 192.168.0.122   93.184.216.34   TCP       66 55017 → 80 [SYN] Seq=0 Win=6
12 04:28:41 192.168.0.122   93.184.216.34   TCP       66 [TCP Retransmission] 55014 →
13 04:28:42 192.168.0.122   93.184.216.34   TCP       66 [TCP Retransmission] 55017 →
14 04:28:43 192.168.0.122   93.184.216.34   TCP       66 [TCP Retransmission] 55016 →
15 04:28:44 192.168.0.122   93.184.216.34   TCP       66 [TCP Retransmission] 55017 →

> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{C
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b:0
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 55015, Dst Port: 80, Seq: 0, Len: 0

0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 f2 5d 40 00 80 06  11 69 c0 a8 00 7a 5d b8   ·4·]@··· ·i···z]·
0020  d8 22 d6 e7 00 50 2c cf  8d 6c 00 00 00 00 80 02   ·"···P,· ·l······
0030  fa f0 eb ae 00 00 02 04  05 b4 01 03 03 08 01 01   ········ ········
0040  04 02                                              ··

Wi-Fi: <live capture in progress>    Packets: 15 · Displayed: 15 (100.0%)    Profile: Default
```

TCP buffer full -- *Source is instructing Destination to stop sending data*

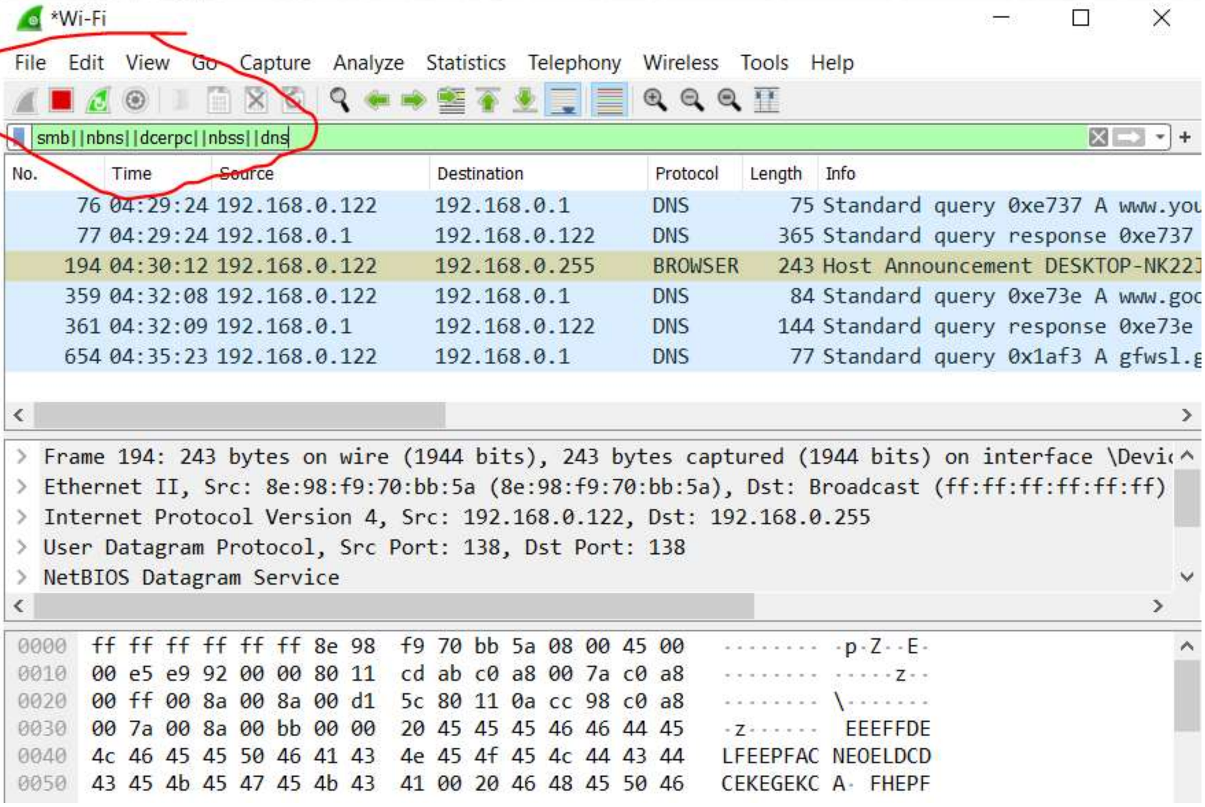
- `tcp.window_size == 0 && tcp.flags.reset != 1`


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.window_size==0&& tcp.flags.reset !=1

No.   Time   Source   Destination   Protocol   Length   Info
```

Filter on Windows -- *Filter out noise, while watching Windows Client - DC exchanges*

- `smb || nbns || dcerpc || nbss || dns`


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
smb||nbns||dcerpc||nbss||dns
```

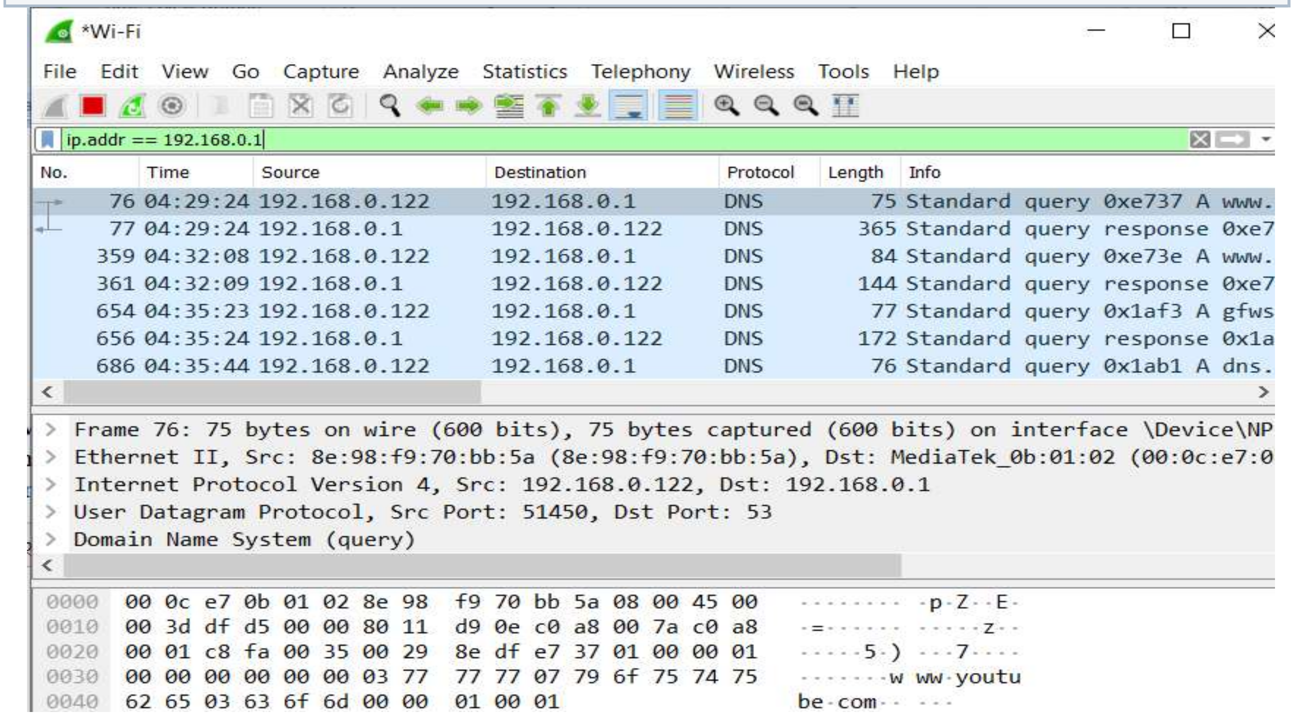
```
No.   Time     Source         Destination    Protocol Length Info
   76 04:29:24 192.168.0.122  192.168.0.1    DNS          75 Standard query 0xe737 A www.you
   77 04:29:24 192.168.0.1    192.168.0.122  DNS         365 Standard query response 0xe737
  194 04:30:12 192.168.0.122  192.168.0.255  BROWSER     243 Host Announcement DESKTOP-NK22J
  359 04:32:08 192.168.0.122  192.168.0.1    DNS          84 Standard query 0xe73e A www.goc
  361 04:32:09 192.168.0.1    192.168.0.122  DNS         144 Standard query response 0xe73e
  654 04:35:23 192.168.0.122  192.168.0.1    DNS          77 Standard query 0x1af3 A gfwsl.g
```

```
> Frame 194: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits) on interface \Devic
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 192.168.0.255
> User Datagram Protocol, Src Port: 138, Dst Port: 138
> NetBIOS Datagram Service
```

```
0000  ff ff ff ff ff ff 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 e5 e9 92 00 00 80 11  cd ab c0 a8 00 7a c0 a8   ········ ·····z··
0020  00 ff 00 8a 00 8a 00 d1  5c 80 11 0a cc 98 c0 a8   ········ \·······
0030  00 7a 00 8a 00 bb 00 00  20 45 45 45 46 46 44 45   ·z······  EEEFFDE
0040  4c 46 45 45 50 46 41 43  4e 45 4f 45 4c 44 43 44   LFEEPFAC NEOELDCD
0050  43 45 4b 45 47 45 4b 43  41 00 20 46 48 45 50 46   CEKEGEKC A· FHEPF
```

We can add multiple protocols fields . For example, “ip.addr” matches against the IP source and the destination addresses in the IP header . The same is true for “tcp.port”, “udp.port”, “eth.addr” and others. It ‘s format is noted here :

```
ip.addr == 192.168.0.1
```


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr == 192.168.0.1
```

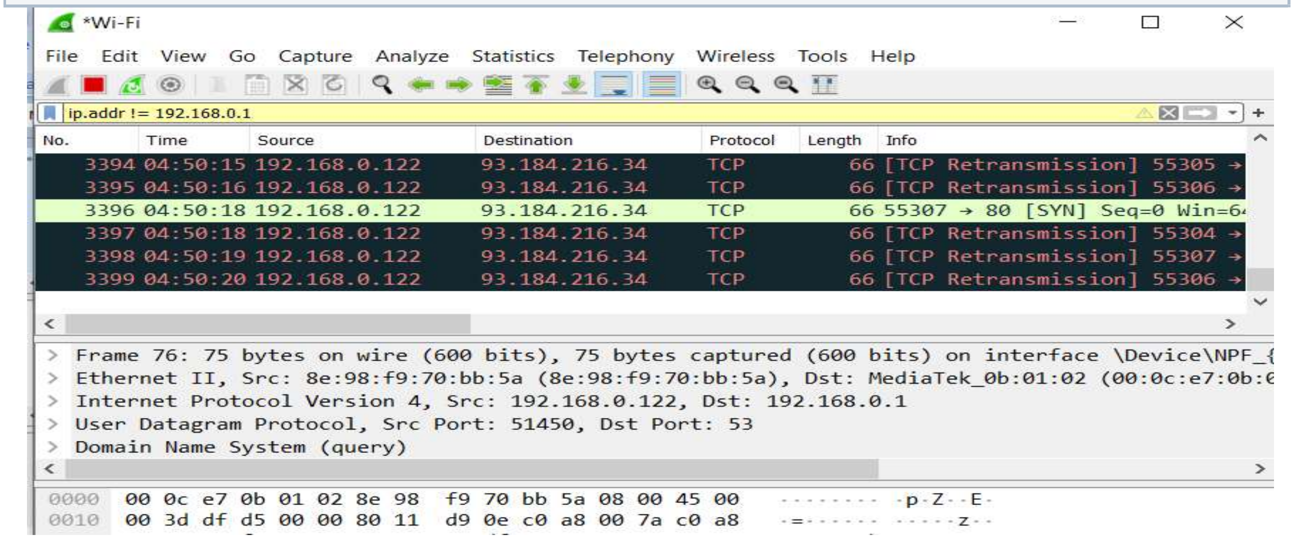
```
No.   Time     Source          Destination     Protocol Length Info
   76 04:29:24 192.168.0.122   192.168.0.1     DNS          75 Standard query 0xe737 A www.
   77 04:29:24 192.168.0.1     192.168.0.122   DNS         365 Standard query response 0xe7
  359 04:32:08 192.168.0.122   192.168.0.1     DNS          84 Standard query 0xe73e A www.
  361 04:32:09 192.168.0.1     192.168.0.122   DNS         144 Standard query response 0xe7
  654 04:35:23 192.168.0.122   192.168.0.1     DNS          77 Standard query 0x1af3 A gfws
  656 04:35:24 192.168.0.1     192.168.0.122   DNS         172 Standard query response 0x1a
  686 04:35:44 192.168.0.122   192.168.0.1     DNS          76 Standard query 0x1ab1 A dns.
```

```
> Frame 76: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface \Device\NP
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 192.168.0.1
> User Datagram Protocol, Src Port: 51450, Dst Port: 53
> Domain Name System (query)
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 3d df d5 00 00 80 11  d9 0e c0 a8 00 7a c0 a8   ·=······ ·····z··
0020  00 01 c8 fa 00 35 00 29  8e df e7 37 01 00 00 01   ·····5·) ···7····
0030  00 00 00 00 00 00 03 77  77 77 07 79 6f 75 74 75   ·······w ww·youtu
0040  62 65 03 63 6f 6d 00 00  01 00 01                  be·com·· ···
```

If we want to filter out any traffic except a specific ip then we can try the following the :

```
ip.addr != 192.168.0.1
```


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr != 192.168.0.1
```

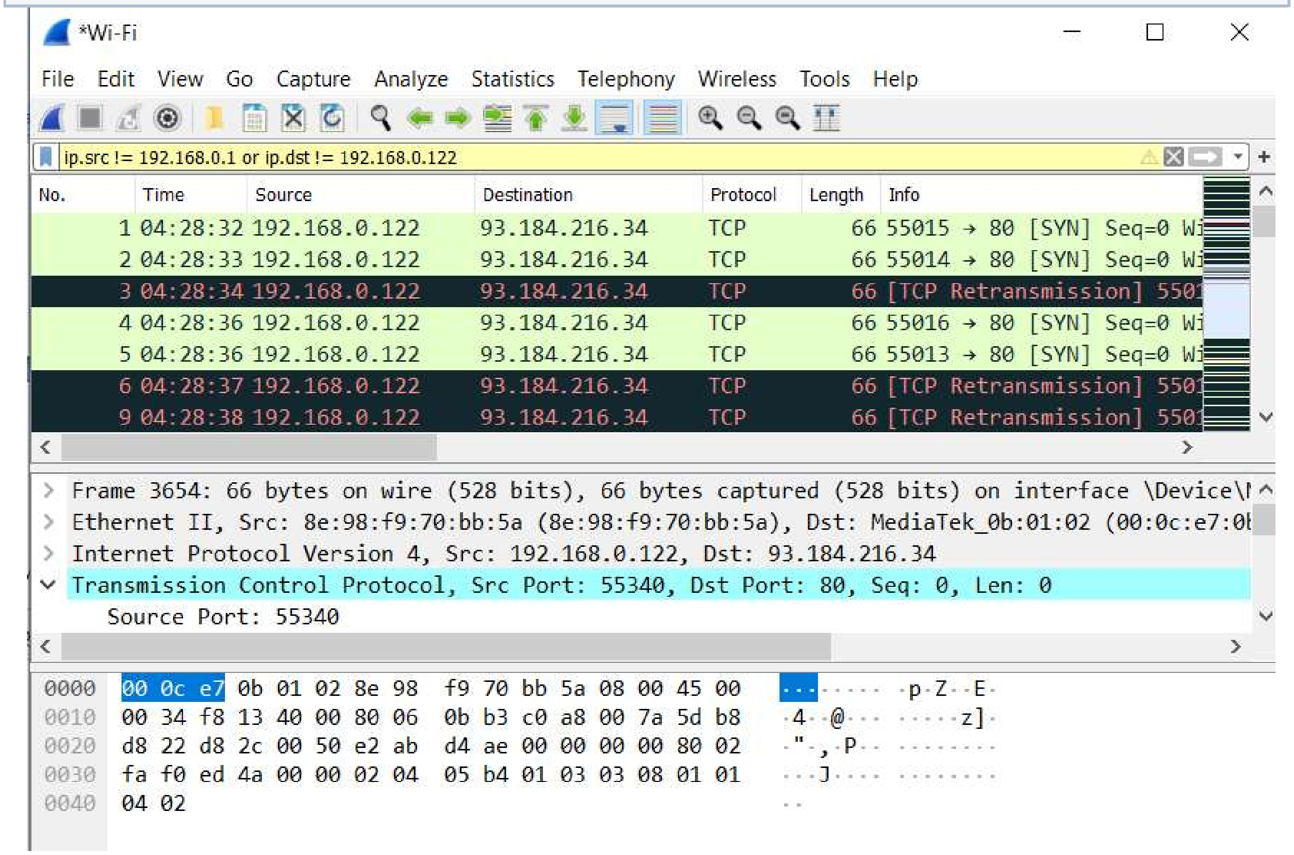
```
No.   Time     Source          Destination     Protocol Length Info
 3394 04:50:15 192.168.0.122   93.184.216.34   TCP          66 [TCP Retransmission] 55305 →
 3395 04:50:16 192.168.0.122   93.184.216.34   TCP          66 [TCP Retransmission] 55306 →
 3396 04:50:18 192.168.0.122   93.184.216.34   TCP          66 55307 → 80 [SYN] Seq=0 Win=64
 3397 04:50:18 192.168.0.122   93.184.216.34   TCP          66 [TCP Retransmission] 55304 →
 3398 04:50:19 192.168.0.122   93.184.216.34   TCP          66 [TCP Retransmission] 55307 →
 3399 04:50:20 192.168.0.122   93.184.216.34   TCP          66 [TCP Retransmission] 55306 →
```

```
> Frame 76: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface \Device\NPF_{
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b:0
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 192.168.0.1
> User Datagram Protocol, Src Port: 51450, Dst Port: 53
> Domain Name System (query)
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 3d df d5 00 00 80 11  d9 0e c0 a8 00 7a c0 a8   ·=······ ·····z··
```

We can use or ,and operator in the filter field :

```
ip.src != 192.168.0.1 or ip.dst != 192.168.0.122
```


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.src != 192.168.0.1 or ip.dst != 192.168.0.122

```
No.    Time     Source          Destination    Protocol  Length Info
     1 04:28:32 192.168.0.122   93.184.216.34  TCP           66 55015 → 80 [SYN] Seq=0 Wi
     2 04:28:33 192.168.0.122   93.184.216.34  TCP           66 55014 → 80 [SYN] Seq=0 Wi
     3 04:28:34 192.168.0.122   93.184.216.34  TCP           66 [TCP Retransmission] 5501
     4 04:28:36 192.168.0.122   93.184.216.34  TCP           66 55016 → 80 [SYN] Seq=0 Wi
     5 04:28:36 192.168.0.122   93.184.216.34  TCP           66 55013 → 80 [SYN] Seq=0 Wi
     6 04:28:37 192.168.0.122   93.184.216.34  TCP           66 [TCP Retransmission] 5501
     9 04:28:38 192.168.0.122   93.184.216.34  TCP           66 [TCP Retransmission] 5501
```

```
> Frame 3654: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\N
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 93.184.216.34
v Transmission Control Protocol, Src Port: 55340, Dst Port: 80, Seq: 0, Len: 0
     Source Port: 55340
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 f8 13 40 00 80 06  0b b3 c0 a8 00 7a 5d b8   ·4··@··· ·····z]·
0020  d8 22 d8 2c 00 50 e2 ab  d4 ae 00 00 00 00 80 02   ·"·,·P·· ········
0030  fa f0 ed 4a 00 00 02 04  05 b4 01 03 03 08 01 01   ···J···· ········
0040  04 02                                              ··
```

Capture Filters :

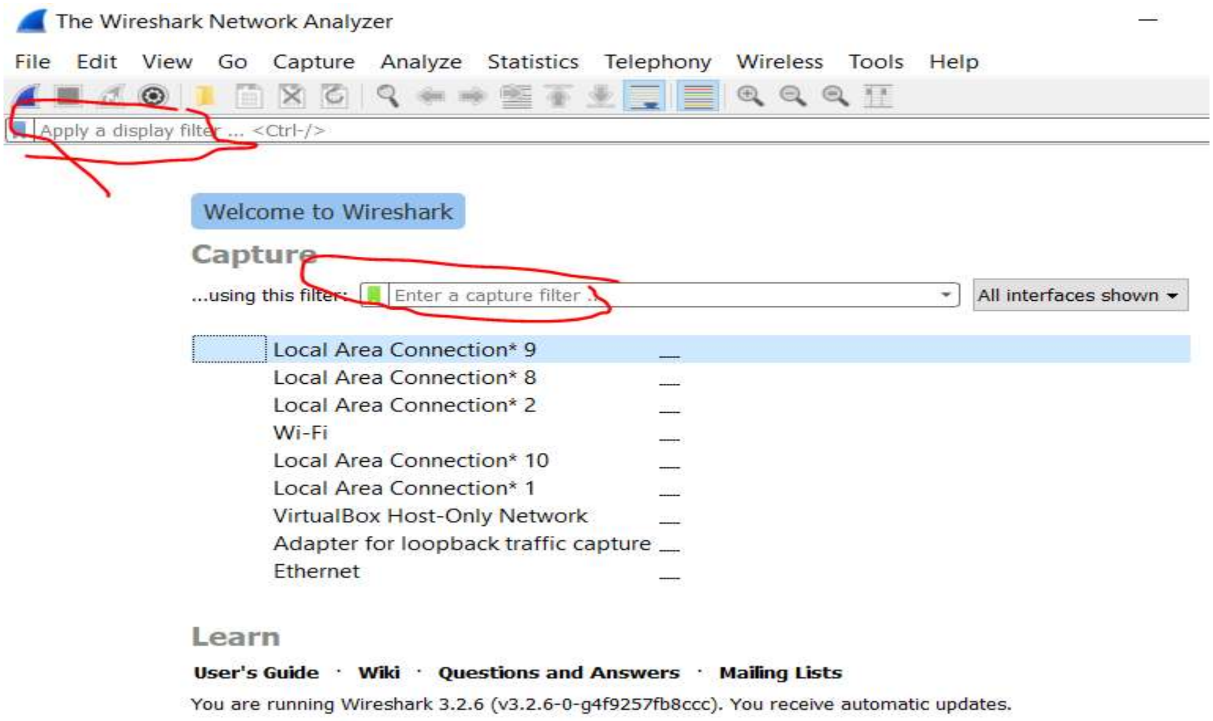
Now take a look what’s the difference between capture filter and display filter from the “wiki.wireshark.org”

Capture filter is not a display filter

Capture filters (like tcp port 80) are not to be confused with display filters (like tcp.port == 80). The former are much more limited and are used to reduce the size of a raw packet capture. The latter are used to hide some packets from the packet list.

Capture filters are set before starting a packet capture and cannot be modified during the capture. Display filters on the other hand do not have this limitation and you can change them on the fly.

In the main window, one can find the capture filter just above the interfaces list and in the interfaces dialog. The display filter can be changed above the packet list as can be seen in this picture:


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

Welcome to Wireshark

Capture

...using this filter: Enter a capture filter ... All interfaces shown

Local Area Connection* 9
Local Area Connection* 8
Local Area Connection* 2
Wi-Fi
Local Area Connection* 10
Local Area Connection* 1
VirtualBox Host-Only Network
Adapter for loopback traffic capture
Ethernet

Learn

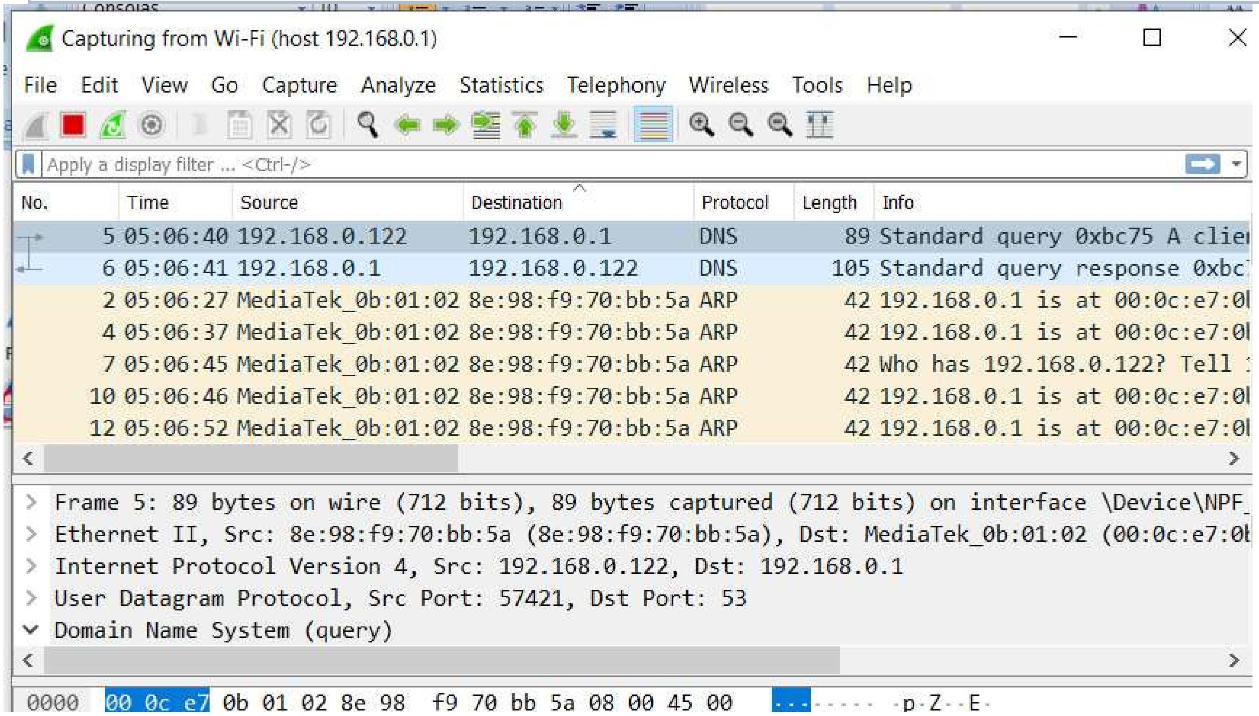
User's Guide · Wiki · Questions and Answers · Mailing Lists

You are running Wireshark 3.2.6 (v3.2.6-0-g4f9257fb8ccc). You receive automatic updates.

Examples:

Capture only traffic to or from IP address 172.18.5.4:

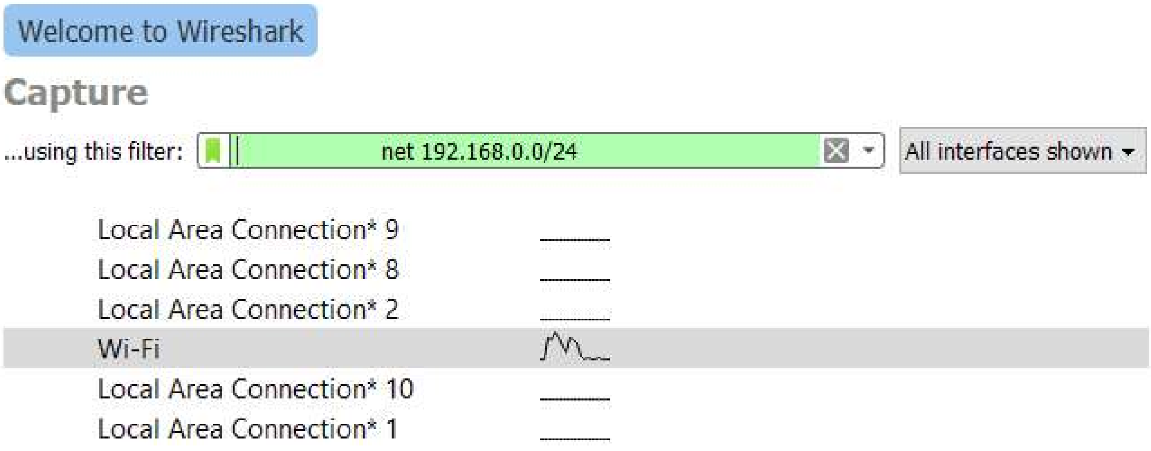
```
• host 172.18.5.4
```


Capturing from Wi-Fi (host 192.168.0.1)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time     Source           Destination       Protocol Length Info
    5 05:06:40 192.168.0.122    192.168.0.1       DNS          89 Standard query 0xbc75 A clie
    6 05:06:41 192.168.0.1      192.168.0.122     DNS         105 Standard query response 0xbc
    2 05:06:27 MediaTek_0b:01:02 8e:98:f9:70:bb:5a ARP          42 192.168.0.1 is at 00:0c:e7:0
    4 05:06:37 MediaTek_0b:01:02 8e:98:f9:70:bb:5a ARP          42 192.168.0.1 is at 00:0c:e7:0
    7 05:06:45 MediaTek_0b:01:02 8e:98:f9:70:bb:5a ARP          42 Who has 192.168.0.122? Tell 1
   10 05:06:46 MediaTek_0b:01:02 8e:98:f9:70:bb:5a ARP          42 192.168.0.1 is at 00:0c:e7:0
   12 05:06:52 MediaTek_0b:01:02 8e:98:f9:70:bb:5a ARP          42 192.168.0.1 is at 00:0c:e7:0

> Frame 5: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device\NPF_
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 192.168.0.1
> User Datagram Protocol, Src Port: 57421, Dst Port: 53
v Domain Name System (query)

0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ........ .p.Z..E.
```

Capture traffic to or from a range of IP addresses:

```
•   net 192.168.0.0/24
```

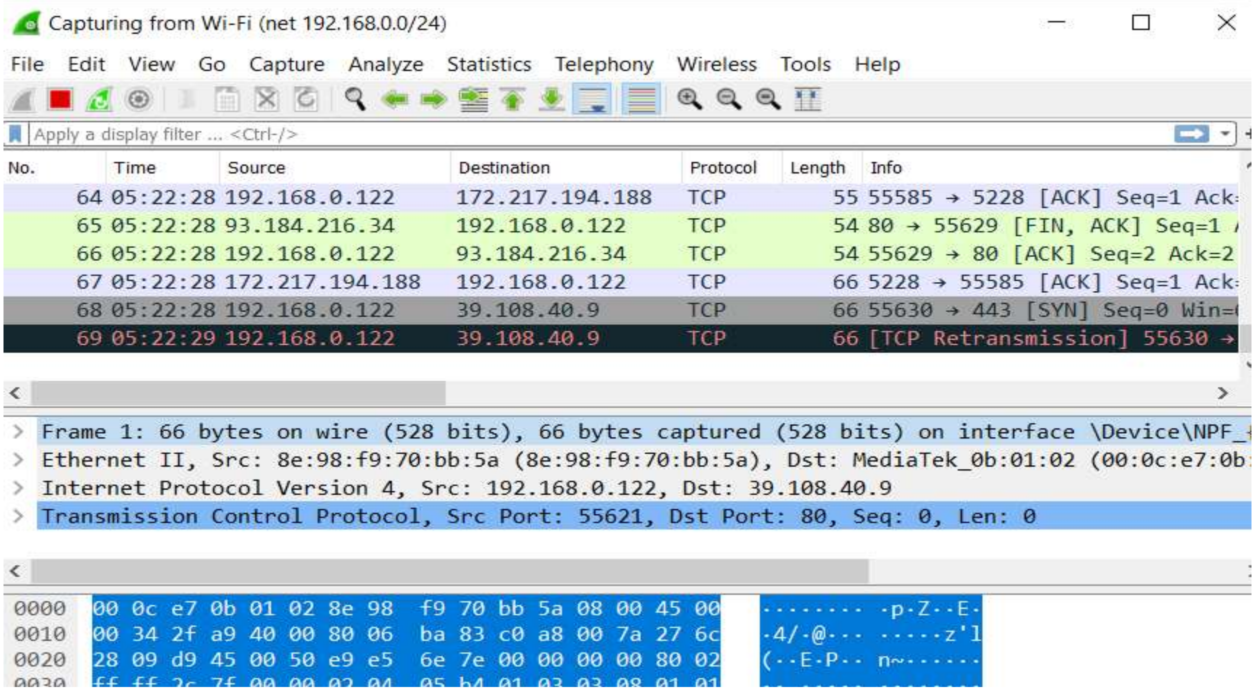


Welcome to Wireshark

Capture

...using this filter: net 192.168.0.0/24 | All interfaces shown

Local Area Connection* 9
Local Area Connection* 8
Local Area Connection* 2
Wi-Fi
Local Area Connection* 10
Local Area Connection* 1



Capturing from Wi-Fi (net 192.168.0.0/24)

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time     Source         Destination     Protocol Length Info
  64 05:22:28 192.168.0.122  172.217.194.188 TCP        55 55585 → 5228 [ACK] Seq=1 Ack
  65 05:22:28 93.184.216.34  192.168.0.122   TCP        54 80 → 55629 [FIN, ACK] Seq=1 A
  66 05:22:28 192.168.0.122  93.184.216.34   TCP        54 55629 → 80 [ACK] Seq=2 Ack=2
  67 05:22:28 172.217.194.188 192.168.0.122  TCP        66 5228 → 55585 [ACK] Seq=1 Ack
  68 05:22:28 192.168.0.122  39.108.40.9     TCP        66 55630 → 443 [SYN] Seq=0 Win=
  69 05:22:29 192.168.0.122  39.108.40.9     TCP        66 [TCP Retransmission] 55630 →
```

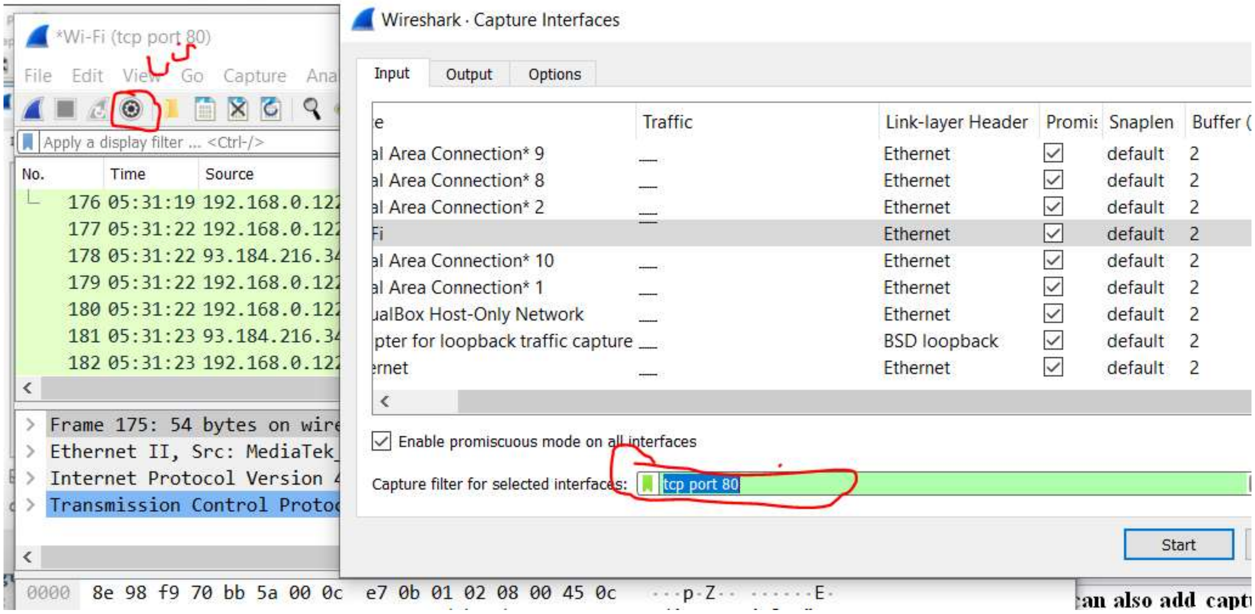
```
> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 39.108.40.9
> Transmission Control Protocol, Src Port: 55621, Dst Port: 80, Seq: 0, Len: 0
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 2f a9 40 00 80 06  ba 83 c0 a8 00 7a 27 6c   ·4/·@··· ·····z'l
0020  28 09 d9 45 00 50 e9 e5  6e 7e 00 00 00 00 80 02   (··E·P·· n~······
0030  ff ff 2c 7f 00 00 02 04  05 b4 01 03 03 08 01 01   ········ ········
```

We can also use this as alternative:

```
src net 192.168.0.0 mask 255.255.255.0
```

So here we can also add capture filter from the capture option :



Apply a display filter ... <Ctrl-/>

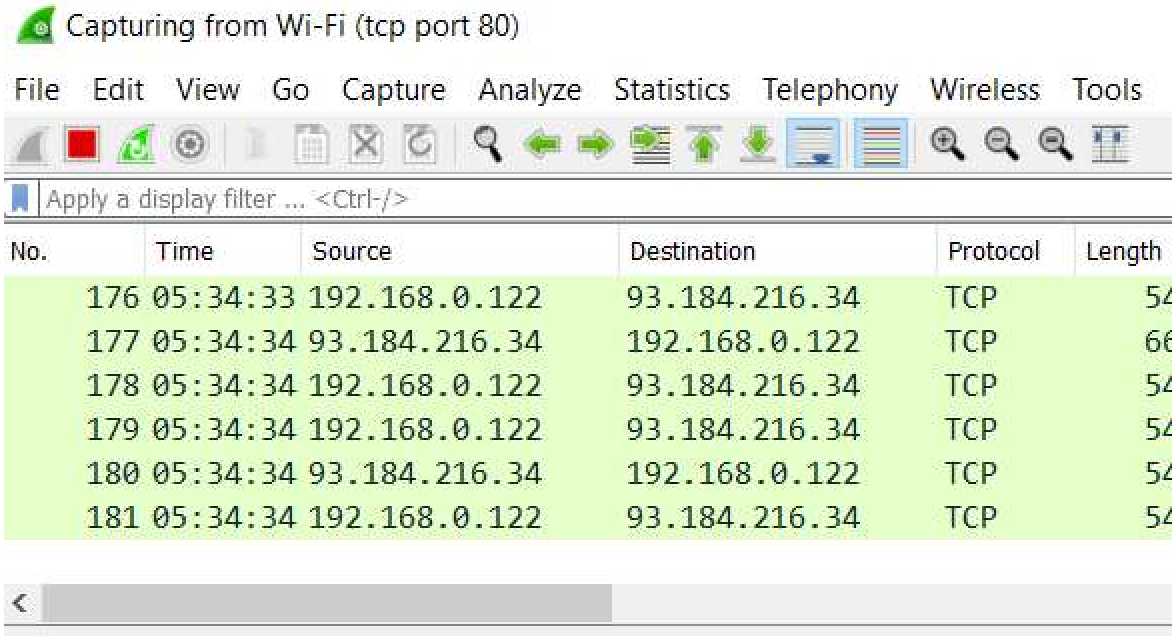
```
No.   Time     Source
  176 05:31:19 192.168.0.12
  177 05:31:22 192.168.0.12
  178 05:31:22 93.184.216.34
  179 05:31:22 192.168.0.12
  180 05:31:22 192.168.0.12
  181 05:31:23 93.184.216.34
  182 05:31:23 192.168.0.12

> Frame 175: 54 bytes on wire
> Ethernet II, Src: MediaTek_
> Internet Protocol Version 4
> Transmission Control Protoc

0000  8e 98 f9 70 bb 5a 00 0c  e7 0b 01 02 08 00 45 0c   ···p·Z·· ······E·
```

e	Traffic	Link-layer Header	Promis	Snaplen	Buffer (
al Area Connection* 9	__	Ethernet	☑	default	2
al Area Connection* 8	__	Ethernet	☑	default	2
al Area Connection* 2	__	Ethernet	☑	default	2
Fi		Ethernet	☑	default	2
al Area Connection* 10	__	Ethernet	☑	default	2
al Area Connection* 1	__	Ethernet	☑	default	2
ualBox Host-Only Network	__	Ethernet	☑	default	2
pter for loopback traffic capture	__	BSD loopback	☑	default	2
rnet	__	Ethernet	☑	default	2

☑ Enable promiscuous mode on all interfaces

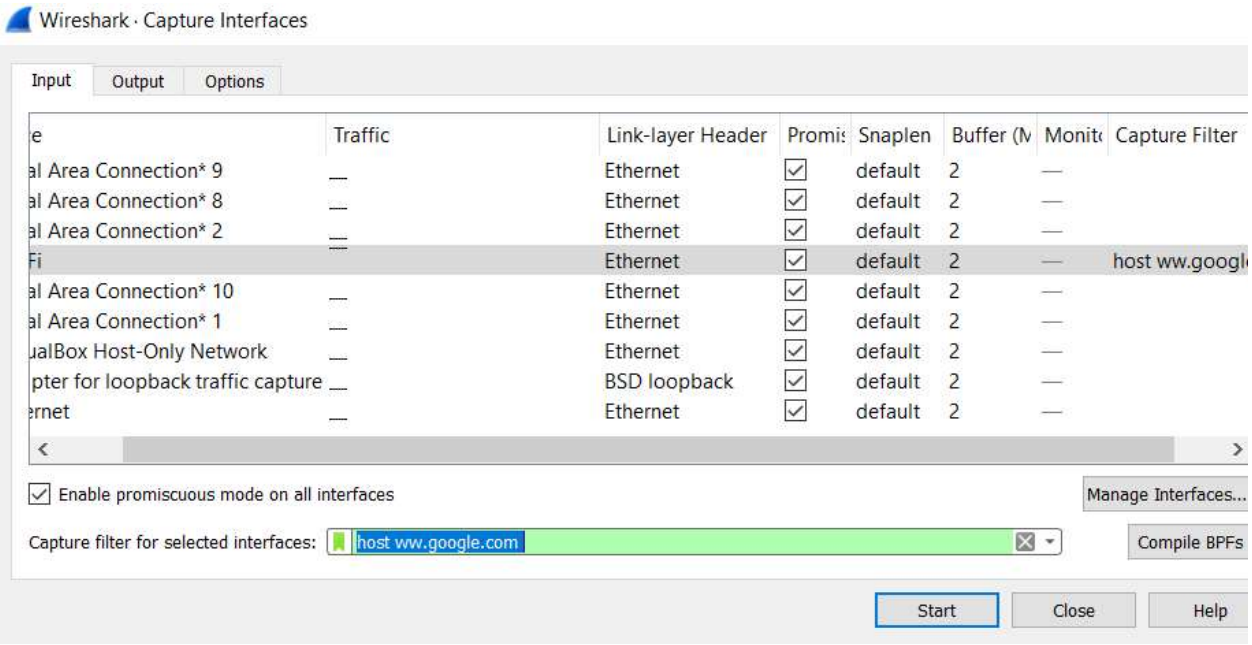
Capture filter for selected interfaces: tcp port 80

Here first we need to stop the current capture packet and then click on the capture option and then add the capture filter in the capture interface .


Capturing from Wi-Fi (tcp port 80)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools

Apply a display filter ... <Ctrl-/>

```
No.   Time     Source         Destination    Protocol  Length
  176 05:34:33 192.168.0.122  93.184.216.34  TCP       54
  177 05:34:34 93.184.216.34  192.168.0.122  TCP       66
  178 05:34:34 192.168.0.122  93.184.216.34  TCP       54
  179 05:34:34 192.168.0.122  93.184.216.34  TCP       54
  180 05:34:34 93.184.216.34  192.168.0.122  TCP       54
  181 05:34:34 192.168.0.122  93.184.216.34  TCP       54
```



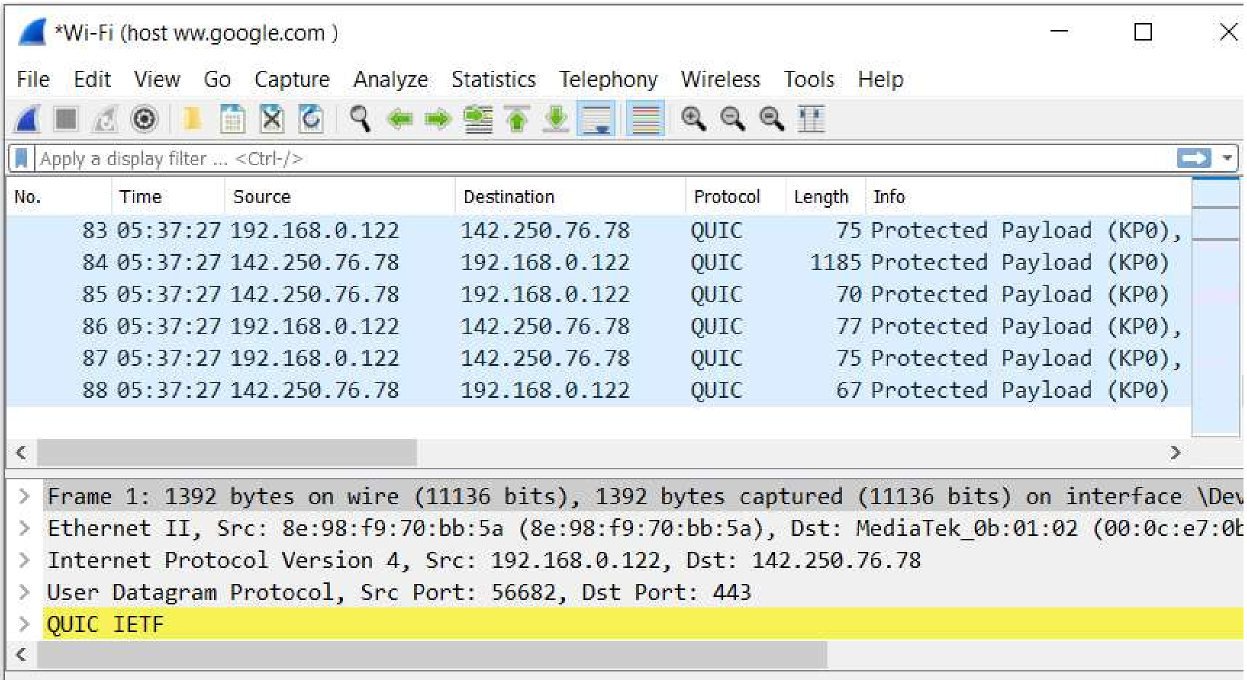
Wireshark · Capture Interfaces

e	Traffic	Link-layer Header	Promi:	Snaplen	Buffer (M	Monit	Capture Filter
al Area Connection* 9	__	Ethernet	☑	default	2	—	
al Area Connection* 8	__	Ethernet	☑	default	2	—	
al Area Connection* 2	__	Ethernet	☑	default	2	—	
Fi		Ethernet	☑	default	2	—	host ww.googl
al Area Connection* 10	__	Ethernet	☑	default	2	—	
al Area Connection* 1	__	Ethernet	☑	default	2	—	
ualBox Host-Only Network	__	Ethernet	☑	default	2	—	
pter for loopback traffic capture	__	BSD loopback	☑	default	2	—	
ernet	__	Ethernet	☑	default	2	—	

☑ Enable promiscuous mode on all interfaces

Capture filter for selected interfaces: host ww.google.com

Manage Interfaces...
Compile BPFs
Start
Close
Help



*Wi-Fi (host ww.google.com)

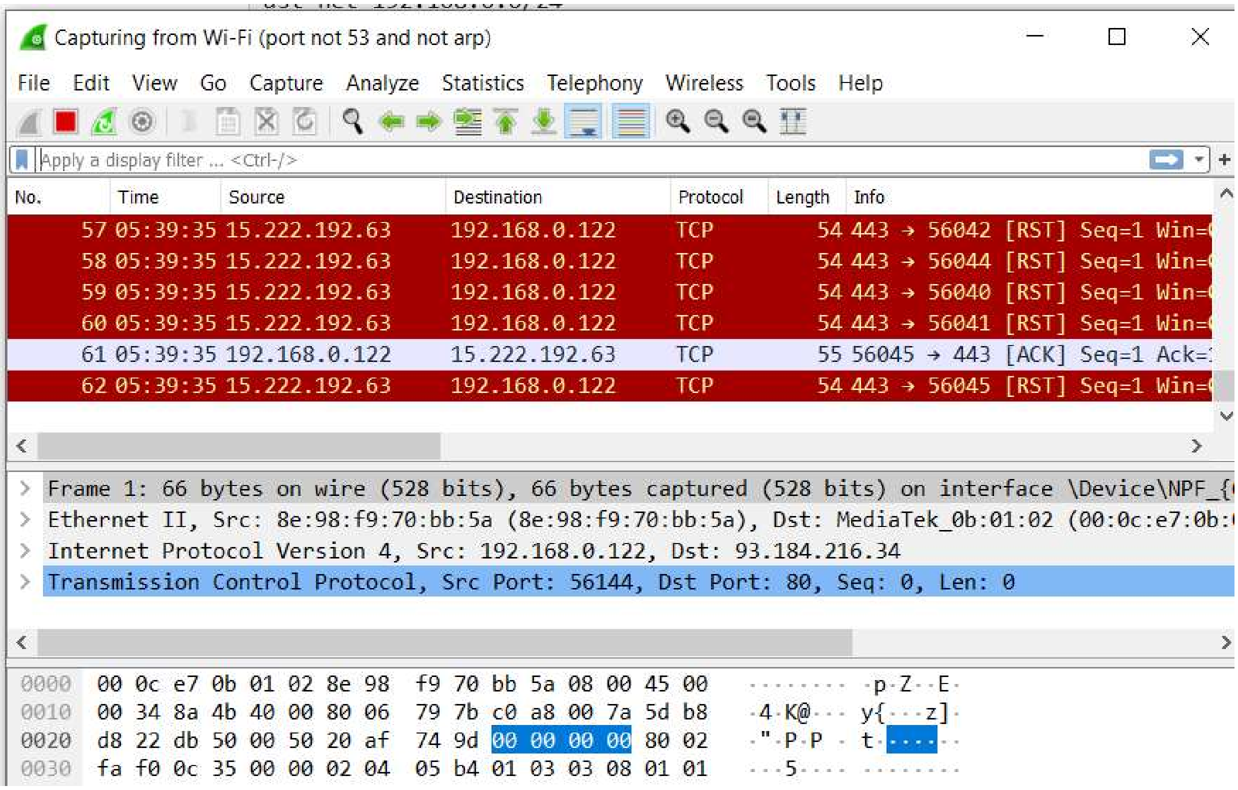
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.     Time      Source          Destination     Protocol  Length Info
     83 05:37:27 192.168.0.122   142.250.76.78   QUIC         75 Protected Payload (KP0),
     84 05:37:27 142.250.76.78   192.168.0.122   QUIC       1185 Protected Payload (KP0)
     85 05:37:27 142.250.76.78   192.168.0.122   QUIC         70 Protected Payload (KP0)
     86 05:37:27 192.168.0.122   142.250.76.78   QUIC         77 Protected Payload (KP0),
     87 05:37:27 192.168.0.122   142.250.76.78   QUIC         75 Protected Payload (KP0),
     88 05:37:27 142.250.76.78   192.168.0.122   QUIC         67 Protected Payload (KP0)
```

```
> Frame 1: 1392 bytes on wire (11136 bits), 1392 bytes captured (11136 bits) on interface \Dev
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 142.250.76.78
> User Datagram Protocol, Src Port: 56682, Dst Port: 443
> QUIC IETF
```

Capture except all ARP and DNS traffic:

- `port not 53 and not arp`


Capturing from Wi-Fi (port not 53 and not arp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

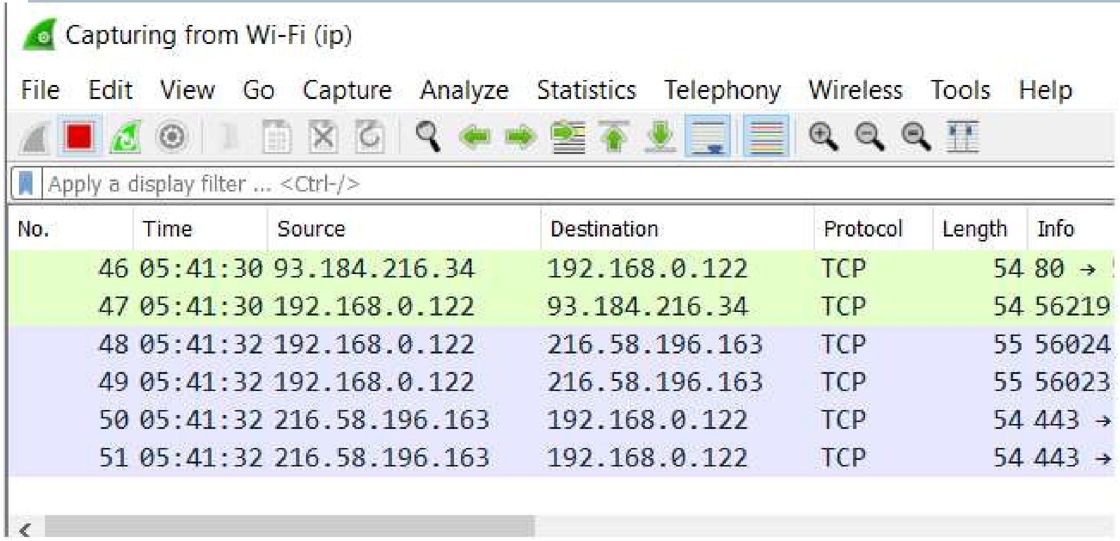
```
No.   Time     Source          Destination    Protocol  Length  Info
   57 05:39:35 15.222.192.63   192.168.0.122  TCP       54 443 → 56042 [RST] Seq=1 Win=(
   58 05:39:35 15.222.192.63   192.168.0.122  TCP       54 443 → 56044 [RST] Seq=1 Win=(
   59 05:39:35 15.222.192.63   192.168.0.122  TCP       54 443 → 56040 [RST] Seq=1 Win=(
   60 05:39:35 15.222.192.63   192.168.0.122  TCP       54 443 → 56041 [RST] Seq=1 Win=(
   61 05:39:35 192.168.0.122   15.222.192.63  TCP       55 56045 → 443 [ACK] Seq=1 Ack=1
   62 05:39:35 15.222.192.63   192.168.0.122  TCP       54 443 → 56045 [RST] Seq=1 Win=(
```

```
> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{(
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e7:0b:(
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 56144, Dst Port: 80, Seq: 0, Len: 0
```

```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 8a 4b 40 00 80 06  79 7b c0 a8 00 7a 5d b8   ·4·K@··· y{···z]·
0020  d8 22 db 50 00 50 20 af  74 9d 00 00 00 00 80 02   ·"·P·P · t·······
0030  fa f0 0c 35 00 00 02 04  05 b4 01 03 03 08 01 01   ···5···· ········
```

Capture only IPv4 traffic - the shortest filter, but sometimes very useful to get rid of lower layer protocols like ARP and STP:

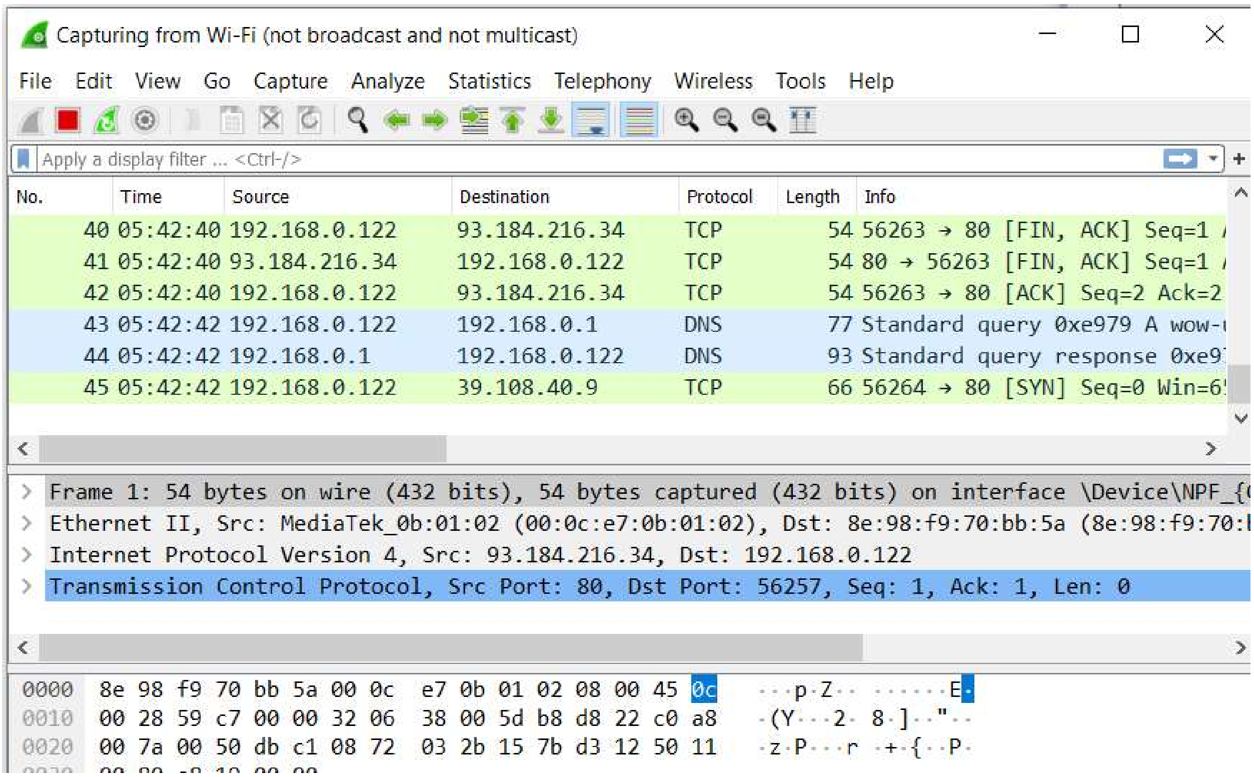
```
•  ip
```


Capturing from Wi-Fi (ip)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time     Source         Destination     Protocol  Length  Info
   46 05:41:30 93.184.216.34  192.168.0.122   TCP       54 80 →
   47 05:41:30 192.168.0.122  93.184.216.34   TCP       54 56219
   48 05:41:32 192.168.0.122  216.58.196.163  TCP       55 56024
   49 05:41:32 192.168.0.122  216.58.196.163  TCP       55 56023
   50 05:41:32 216.58.196.163 192.168.0.122   TCP       54 443 →
   51 05:41:32 216.58.196.163 192.168.0.122   TCP       54 443 →
```

Capture only unicast traffic - useful to get rid of noise on the network if you only want to see traffic to and from your machine, not, for example, broadcast and multicast announcements:

```
•  not broadcast and not multicast
```


Capturing from Wi-Fi (not broadcast and not multicast)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time     Source         Destination    Protocol  Length Info
   40 05:42:40 192.168.0.122  93.184.216.34  TCP           54 56263 → 80 [FIN, ACK] Seq=1 /
   41 05:42:40 93.184.216.34  192.168.0.122  TCP           54 80 → 56263 [FIN, ACK] Seq=1 /
   42 05:42:40 192.168.0.122  93.184.216.34  TCP           54 56263 → 80 [ACK] Seq=2 Ack=2
   43 05:42:42 192.168.0.122  192.168.0.1    DNS           77 Standard query 0xe979 A wow-u
   44 05:42:42 192.168.0.1    192.168.0.122  DNS           93 Standard query response 0xe9
   45 05:42:42 192.168.0.122  39.108.40.9    TCP           66 56264 → 80 [SYN] Seq=0 Win=6!
```

```
> Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{
> Ethernet II, Src: MediaTek_0b:01:02 (00:0c:e7:0b:01:02), Dst: 8e:98:f9:70:bb:5a (8e:98:f9:70:
> Internet Protocol Version 4, Src: 93.184.216.34, Dst: 192.168.0.122
> Transmission Control Protocol, Src Port: 80, Dst Port: 56257, Seq: 1, Ack: 1, Len: 0
```

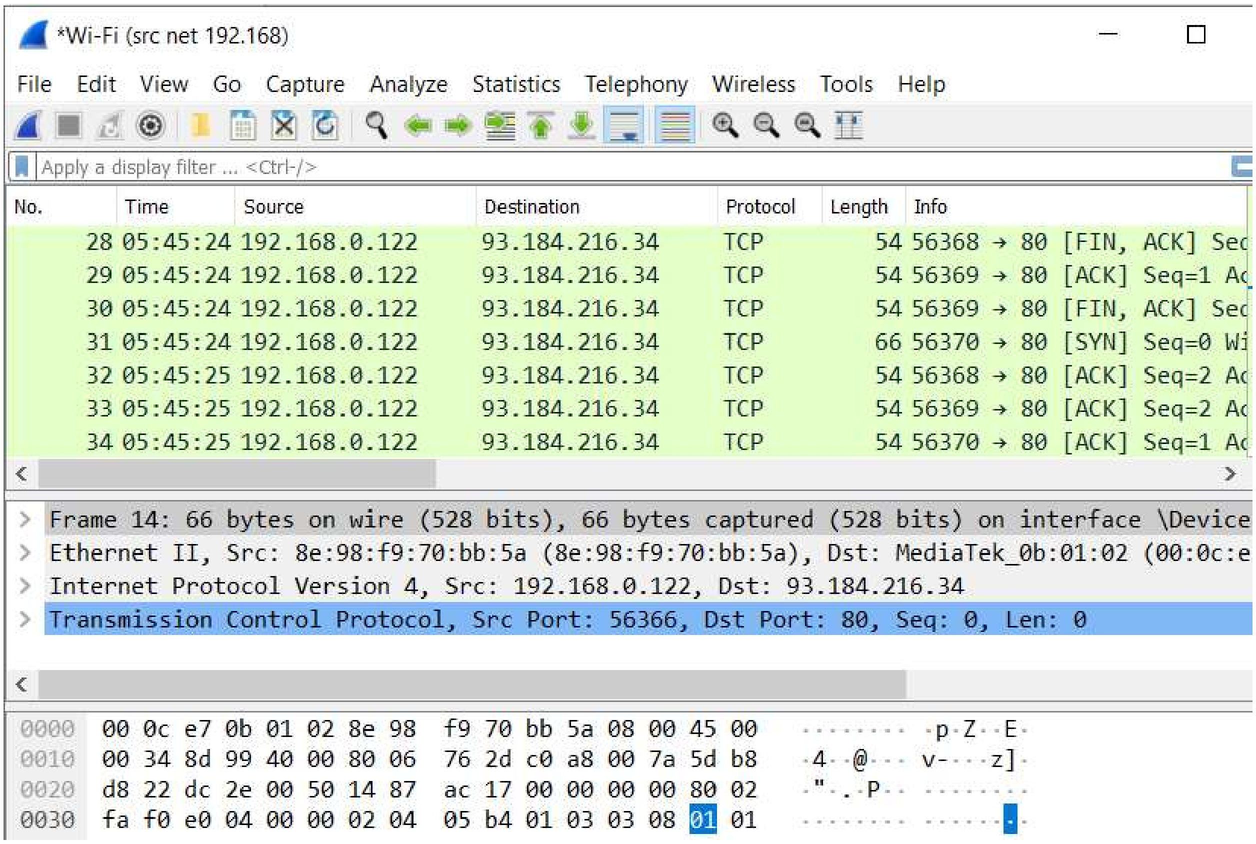
```
0000  8e 98 f9 70 bb 5a 00 0c  e7 0b 01 02 08 00 45 0c   ···p·Z·· ······E·
0010  00 28 59 c7 00 00 32 06  38 00 5d b8 d8 22 c0 a8   ·(Y···2· 8·]··"··
0020  00 7a 00 50 db c1 08 72  03 2b 15 7b d3 12 50 11   ·z·P···r ·+·{··P·
```

Capture VLAN traffic:

```
•  vlan
```

Capture all traffic originating (source) in the IP range 192.168.XXX.XXX:

```
•  src net 192.168
```


*Wi-Fi (src net 192.168)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.     Time      Source          Destination     Protocol  Length  Info
     28 05:45:24 192.168.0.122   93.184.216.34   TCP          54 56368 → 80 [FIN, ACK] Sec
     29 05:45:24 192.168.0.122   93.184.216.34   TCP          54 56369 → 80 [ACK] Seq=1 Ac
     30 05:45:24 192.168.0.122   93.184.216.34   TCP          54 56369 → 80 [FIN, ACK] Sec
     31 05:45:24 192.168.0.122   93.184.216.34   TCP          66 56370 → 80 [SYN] Seq=0 Wi
     32 05:45:25 192.168.0.122   93.184.216.34   TCP          54 56368 → 80 [ACK] Seq=2 Ac
     33 05:45:25 192.168.0.122   93.184.216.34   TCP          54 56369 → 80 [ACK] Seq=2 Ac
     34 05:45:25 192.168.0.122   93.184.216.34   TCP          54 56370 → 80 [ACK] Seq=1 Ac
```

```
> Frame 14: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device
> Ethernet II, Src: 8e:98:f9:70:bb:5a (8e:98:f9:70:bb:5a), Dst: MediaTek_0b:01:02 (00:0c:e
> Internet Protocol Version 4, Src: 192.168.0.122, Dst: 93.184.216.34
> Transmission Control Protocol, Src Port: 56366, Dst Port: 80, Seq: 0, Len: 0
```

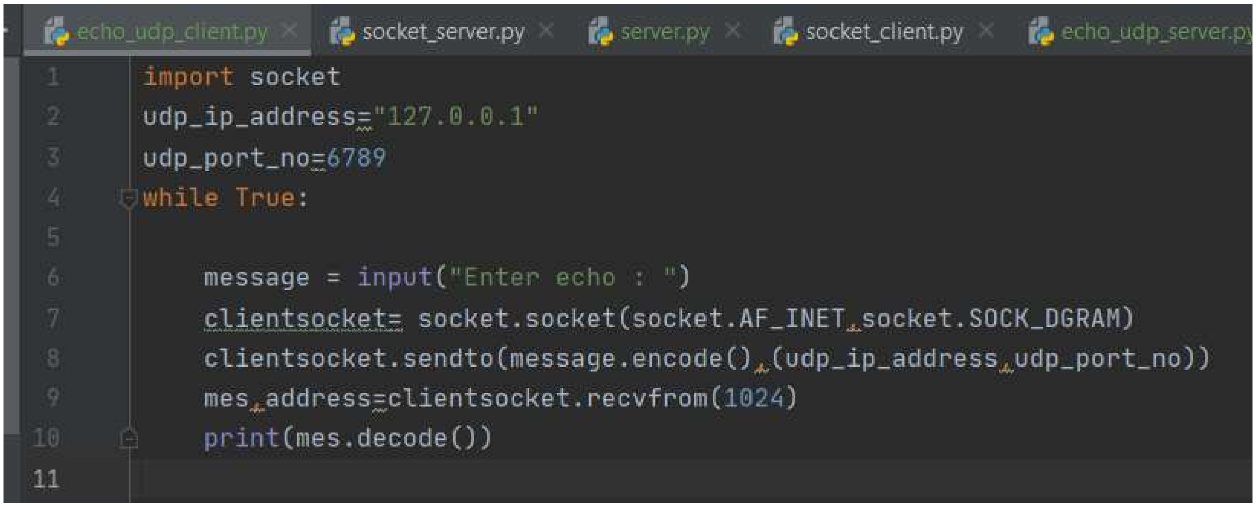
```
0000  00 0c e7 0b 01 02 8e 98  f9 70 bb 5a 08 00 45 00   ········ ·p·Z··E·
0010  00 34 8d 99 40 00 80 06  76 2d c0 a8 00 7a 5d b8   ·4··@··· v-···z]·
0020  d8 22 dc 2e 00 50 14 87  ac 17 00 00 00 00 80 02   ·"·.·P·· ········
0030  fa f0 e0 04 00 00 02 04  05 b4 01 03 03 08 01 01   ········ ········
```

Before go to the conclusion we try to capture the traffic that has been written in the previous lab report including :

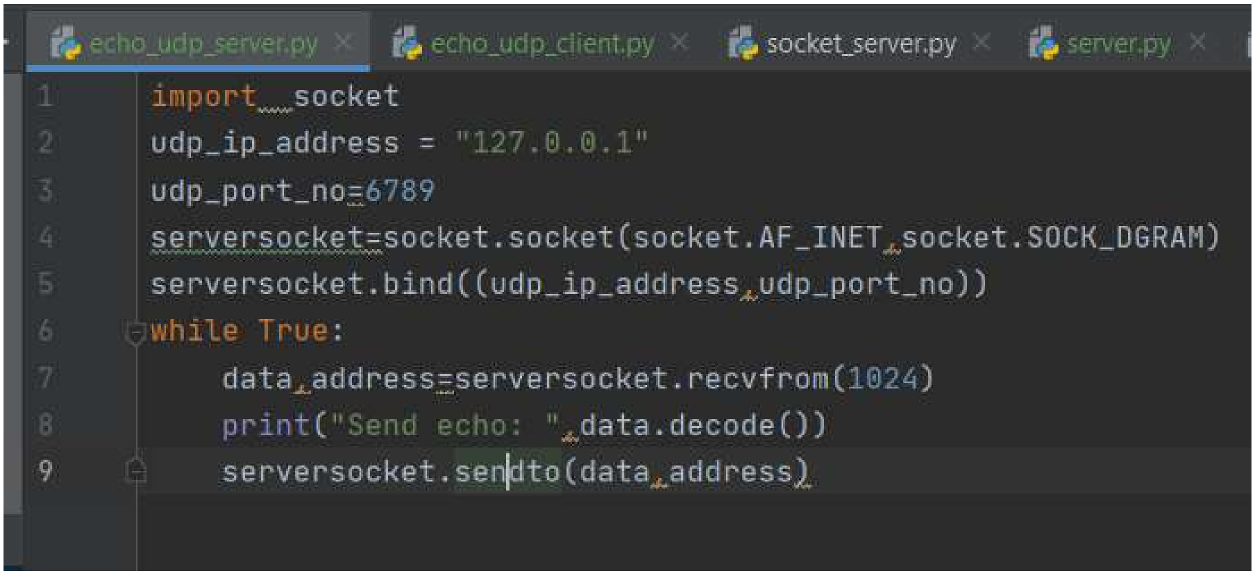
1. echo server client using udp

2.echo server client using tcp

So first here we add the python code and run this file form pycharm:



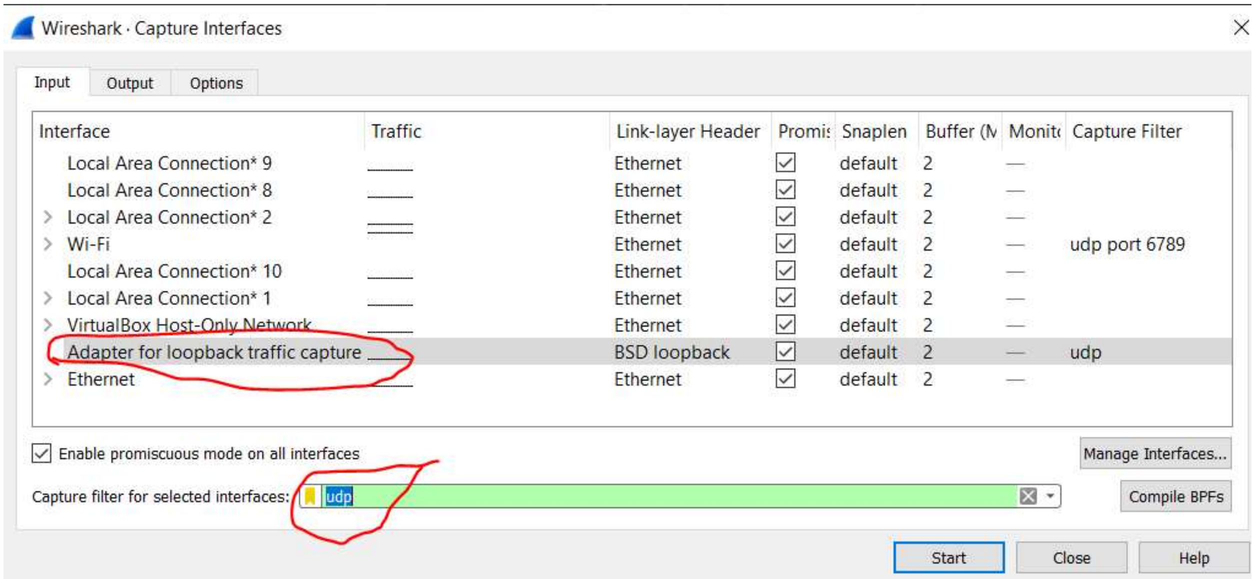
```python
import socket
udp_ip_address="127.0.0.1"
udp_port_no=6789
while True:

    message = input("Enter echo : ")
    clientsocket= socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
    clientsocket.sendto(message.encode(),(udp_ip_address,udp_port_no))
    mes,address=clientsocket.recvfrom(1024)
    print(mes.decode())

```



```
import socket
udp_ip_address = "127.0.0.1"
udp_port_no=6789
serversocket=socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
serversocket.bind((udp_ip_address,udp_port_no))
while True:
    data,address=serversocket.recvfrom(1024)
    print("Send echo: ",data.decode())
    serversocket.sendto(data,address)
```

Now we run this code and go to the wireshark to capture the traffic :

Now in wireshark click in the capture option and then select the "adapter for loopback traffic capture" and the filter field put the "udp".



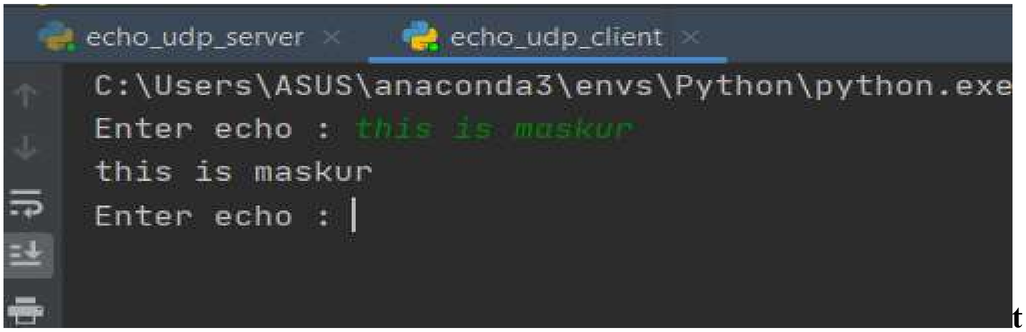
Interface	Traffic	Link-layer Header	Promis	Snaplen	Buffer (N	Monit	Capture Filter
Local Area Connection* 9		Ethernet	☑	default	2	—	
Local Area Connection* 8		Ethernet	☑	default	2	—	
Local Area Connection* 2		Ethernet	☑	default	2	—	
Wi-Fi		Ethernet	☑	default	2	—	udp port 6789
Local Area Connection* 10		Ethernet	☑	default	2	—	
Local Area Connection* 1		Ethernet	☑	default	2	—	
VirtualBox Host-Only Network		Ethernet	☑	default	2	—	
Adapter for loopback traffic capture		BSD loopback	☑	default	2	—	udp
Ethernet		Ethernet	☑	default	2	—	

☑ Enable promiscuous mode on all interfaces

Capture filter for selected interfaces: udp

Now start

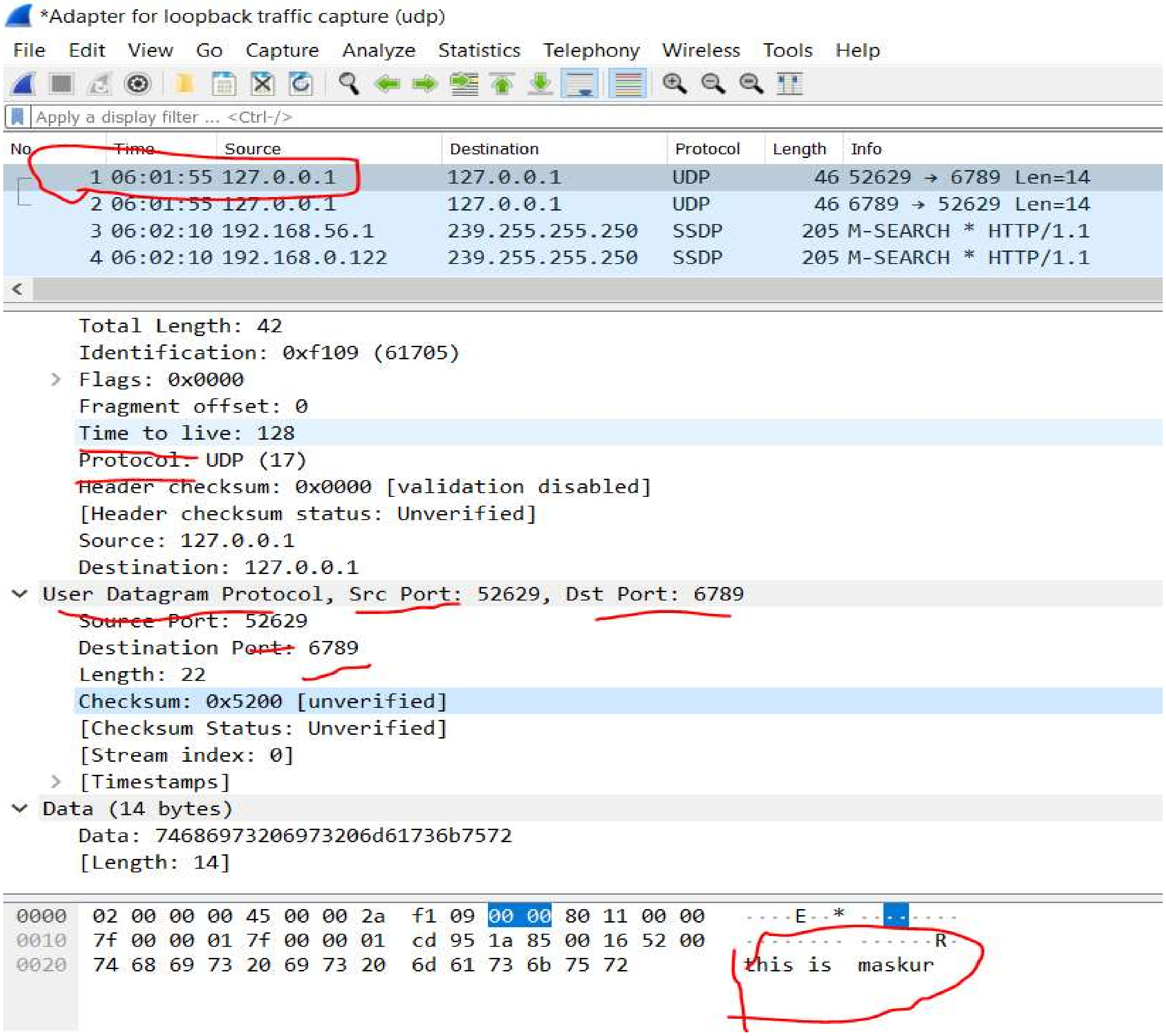
So when we send data from the client



```
C:\Users\ASUS\anaconda3\envs\Python\python.exe
Enter echo : this is maskur
this is maskur
Enter echo : |
```

t :

Now go to wireshark to see what is happening


*Adapter for loopback traffic capture (udp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No. Time       Source          Destination      Protocol Length Info
  1 06:01:55   127.0.0.1       127.0.0.1        UDP          46 52629 → 6789 Len=14
  2 06:01:55   127.0.0.1       127.0.0.1        UDP          46 6789 → 52629 Len=14
  3 06:02:10   192.168.56.1    239.255.255.250  SSDP        205 M-SEARCH * HTTP/1.1
  4 06:02:10   192.168.0.122   239.255.255.250  SSDP        205 M-SEARCH * HTTP/1.1
```

```
    Total Length: 42
    Identification: 0xf109 (61705)
  > Flags: 0x0000
    Fragment offset: 0
    Time to live: 128
    Protocol: UDP (17)
    Header checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source: 127.0.0.1
    Destination: 127.0.0.1
v User Datagram Protocol, Src Port: 52629, Dst Port: 6789
    Source Port: 52629
    Destination Port: 6789
    Length: 22
    Checksum: 0x5200 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
  > [Timestamps]
v Data (14 bytes)
    Data: 7468697320697320 6d61736b7572
    [Length: 14]
```

```
0000  02 00 00 00 45 00 00 2a  f1 09 00 00 80 11 00 00   ....E..*  ........
0010  7f 00 00 01 7f 00 00 01  cd 95 1a 85 00 16 52 00   ........  ......R.
0020  74 68 69 73 20 69 73 20  6d 61 73 6b 75 72         this is   maskur
```

In the above it shows everything about the udp traffic on this loopback interface .

It show time to live : 128

Source port : 52629

Destination prot : 6789

Data length : 14 bytes

Conclusion : This is one of the most enjoyable lab program where I learn how to use the wireshark . The basic information about the filter and also the display filter and many other filter command .And finally the last one loopback interface where I check the udp program on my local computer the echo program which is also very interesting . To do this lab report I have taken help from the slide given by my class teacher . And also from the official wireshark website and few youtube toutorials.