

## **Lab Report: 08**

**Report Name: Wireshark Installation and Use**

**Course code: ICT-3208**

**Course title: Computer Networks Lab**

**Date of**

**Performance: 04/03/2021**

**Date of**

**Submission: 05/03/2021**

### **SUBMITTED BY**

**Name: Shakhera khanom &  
Masuk Mia**

**ID: IT-18033 & IT-18049**

**3<sup>rd</sup> year 2<sup>nd</sup> semester**

**Session: 2017-18**

**Department of ICT,**

**MBSTU.**

### **SUBMITTED TO**

**Nazrul Islam**

**Assistant Professor**

**Department of ICT,**

**MBSTU.**

## Experiment No: 08

### Experiment Name: Wireshark Installation and Use.

#### Objectives:

- Wireshark installation
- Wireshark Usage
- Protocol analysis and examples

#### Theory:

**Wireshark:** Wireshark is a free and open-source network protocol analyzer widely used around the globe. It captures every packet getting in or out of a network interface and shows them in a nicely formatted text. It is used by Network Engineers all over the world.

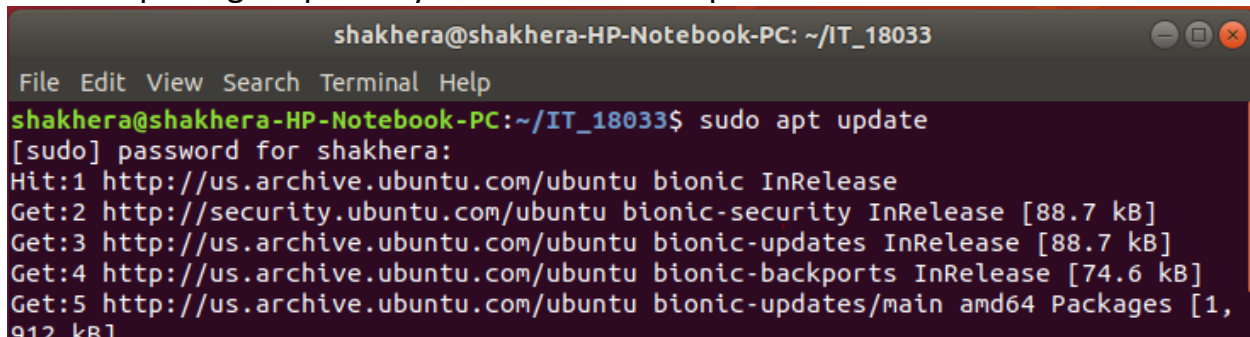
Wireshark is cross platform and it is available for Linux, Windows and Mac OS. You get the same user experience in any operating system you use

#### Installing Wireshark:

First update the APT package repository cache with the following command:

*\$ sudo apt update*

The APT package repository cache should be updated.

A screenshot of a terminal window titled 'shakhera@shakhera-HP-Notebook-PC: ~/IT\_18033'. The terminal shows the command 'sudo apt update' being executed. The output indicates that the package lists were successfully updated from several sources, including security updates and backports. The terminal text is as follows:

```
shakhera@shakhera-HP-Notebook-PC: ~/IT_18033
File Edit View Search Terminal Help
shakhera@shakhera-HP-Notebook-PC:~/IT_18033$ sudo apt update
[sudo] password for shakhera:
Hit:1 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1,912 kB]
```

Now, Run the following command to install Wireshark on your Ubuntu machine:

*\$ sudo apt get install wireshark*

```
shakhera@shakhera-HP-Notebook-PC:~/IT_18033$ sudo apt install wireshark
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  efibootmgr libegl1-mesa libfwup1 libllvm9 libwayland-egl1-mesa
  libwireshark11 libwiretap8 libwscodec2 libwsutil9 linux-headers-5.3.0-28
  linux-headers-5.3.0-28-generic linux-image-5.3.0-28-generic
  linux-modules-5.3.0-28-generic linux-modules-extra-5.3.0-28-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libminizip1 libwireshark-data libwireshark14 libwiretap11 libwsutil12 tshark
  wireshark-common wireshark-qt
Suggested packages:
  geoupdate geoup-database-extra libjs-leaflet libjs-leaflet.markercluster
  snmp-mibs-downloader wireshark-doc
The following NEW packages will be installed:
```

Wireshark should be installed.

To be able to capture packets as normal user, add your to Wireshark group using following command:

```
$ sudo usermod -aG wireshark $(whoami)
```

Also change dumpcap binary file permissions.

```
$ sudo chmod + /usr/bin/dumpcap
```

```
shakhera@shakhera-HP-Notebook-PC:~/IT_18033$ sudo chmod +x /usr/bin/dumpcap
shakhera@shakhera-HP-Notebook-PC:~/IT_18033$ |
```

Now reboot your computer with the following command:

```
$ sudo reboot
```

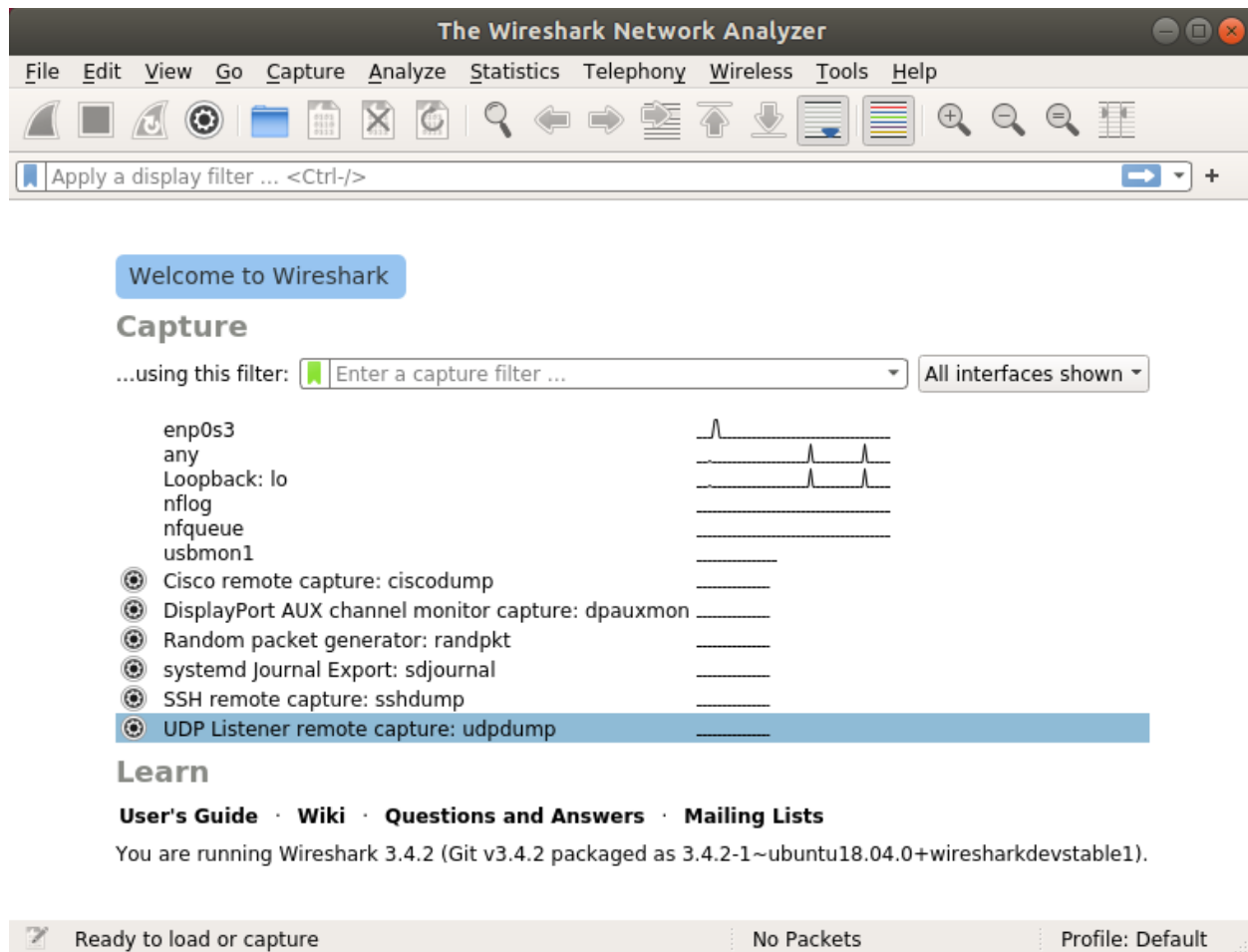
### Starting Wireshark:

Now that Wireshark is installed, You can also run the following command to start Wireshark from the Terminal:

```
$ sudo wireshark
```

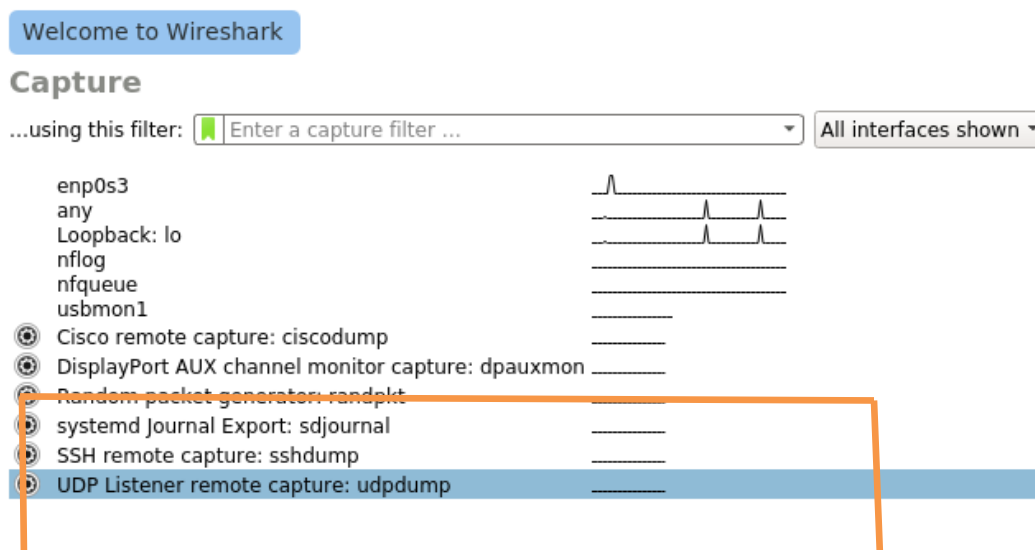
Wireshark will start in your computer.

```
shakhera@shakhera-HP-Notebook-PC:~/IT_18033$ wireshark
12:04:06.730      Warn Could not compile "of" in colorfilters file "/home/shakhera/.wireshark/colorfilters".
"of" is neither a field nor a protocol name.
12:04:06.730      Warn Could not compile "Checksum Errors" in colorfilters file "/home/shakhera/.wireshark/colorfilters".
Neither "cdp.checksum_bad" nor "1" are field or protocol names.
```

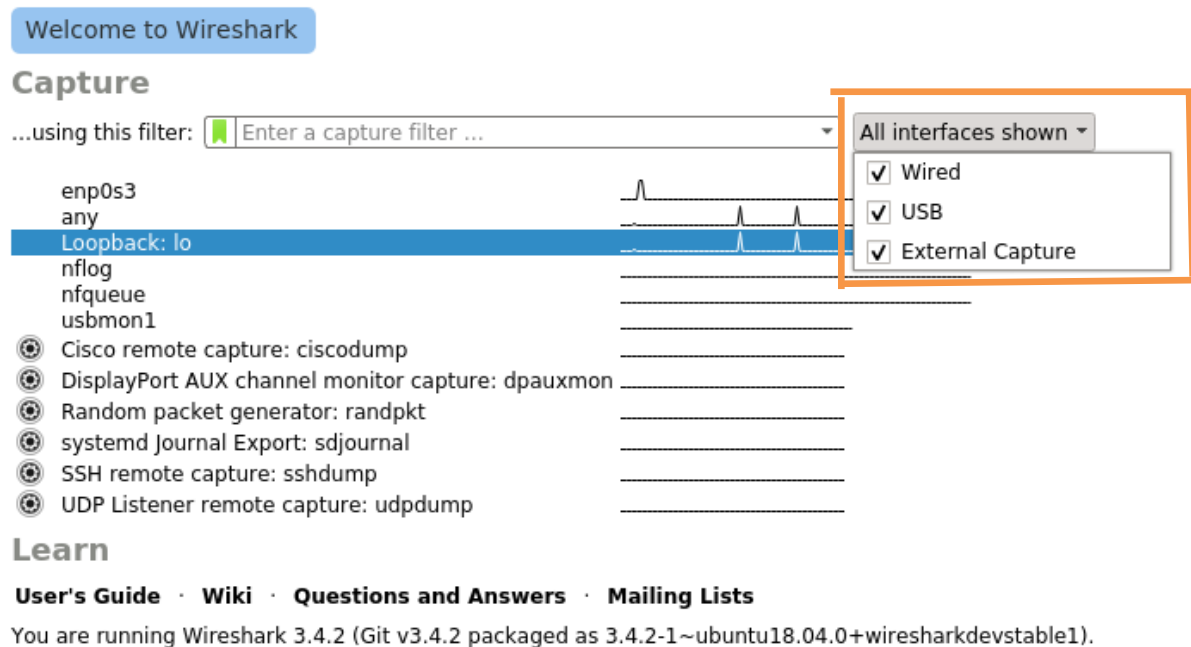


## Capturing Packets Using Wireshark:

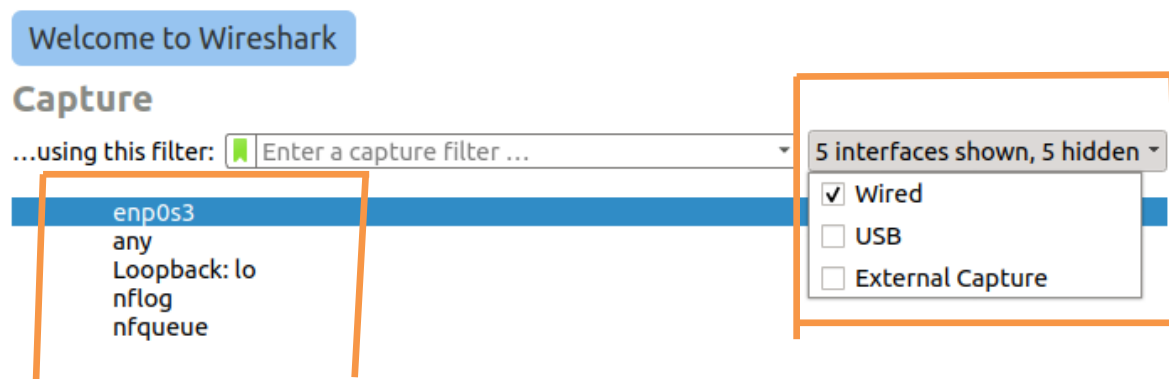
When you start Wireshark, you will see a list of interfaces that you can capture packets to and from.



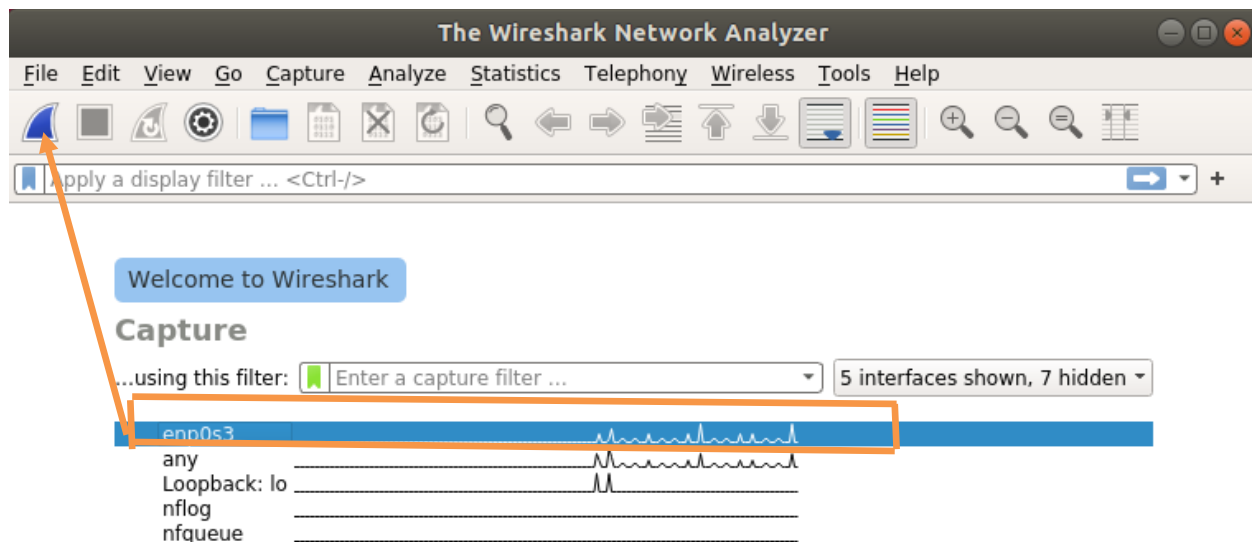
There are many types of interfaces you can monitor using Wireshark, for example, **Wired**, **USB**, and many external devices. You can choose to show specific types of interfaces in the welcome screen from the marked section of the screenshot below.



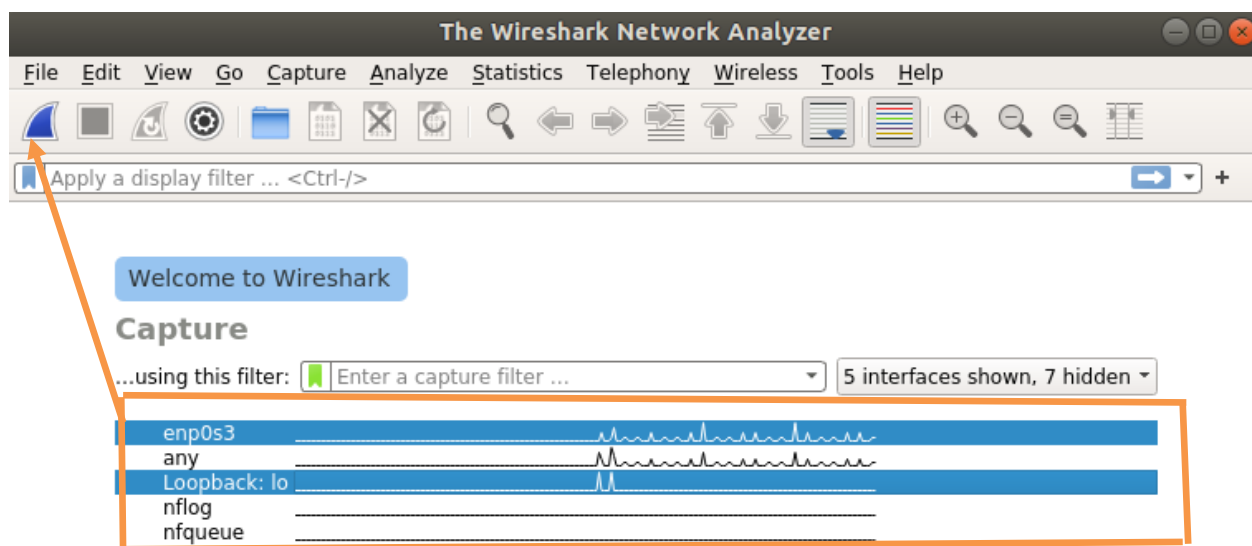
Here, I listed only the **Wired** network interfaces.



Next, to start capturing packets, you have to select the interface (which in my case is enp0s3) and click on the Start capturing packets icon as marked in the image below



You can also capture packets to and from multiple interfaces at the same time. Just press and hold the CTRL button while clicking on the interfaces that you want to capture to and from and then hit the Start capturing packets icon as marked in the image below.



Next, I tried using *ping google.com* command in the terminal and as you can see, many packets were captured

Capturing from enp0s3 and Loopback: lo						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	192.168.43.1	DNS	100	Standard query query
2	-0.000494259	127.0.0.1	127.0.0.53	DNS	100	Standard query response
3	-0.000430393	127.0.0.1	127.0.0.53	DNS	100	Standard query response
4	0.000438811	127.0.0.53	127.0.0.1	DNS	100	Standard query response
5	0.300823353	127.0.0.53	127.0.0.1	DNS	148	Standard query response
6	0.300522242	192.168.43.1	10.0.2.15	DNS	148	Standard query response
7	1.000538308	10.0.2.15	35.224.170.84	TCP	74	4101
8	1.467818611	35.224.170.84	10.0.2.15	TCP	60	80
9	1.467939670	10.0.2.15	35.224.170.84	TCP	54	4101
10	1.468359017	10.0.2.15	35.224.170.84	HTTP	141	GET
11	1.468937749	35.224.170.84	10.0.2.15	TCP	60	80
12	1.883424866	35.224.170.84	10.0.2.15	HTTP	202	HTTP

▶	Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface
▶	Ethernet II, Src: PcsCompu_25:23:01 (08:00:27:25:23:01), Dst: RealtekU_12:35:02
▶	Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.43.1
▶	User Datagram Protocol, Src Port: 33939, Dst Port: 53

0000	52 54 00 12 35 02 08 00 27 25 23 01 08 00 45 00	RT..5... '%#..E.
0010	00 56 6b d8 40 00 40 11 d7 06 0a 00 02 0f c0 a8	.Vk.@.@. ....
0020	2b 01 84 93 00 35 00 42 f8 0b 88 10 01 00 00 01	+...5.B ....
0030	00 00 00 00 00 01 12 63 6f 6e 6e 65 63 74 69 76	.....c onnectiv
0040	69 74 79 2d 63 68 65 63 6b 06 75 62 75 6e 74 75	ity-chec k.ubuntu
0050	03 63 6f 6d 00 00 01 00 01 00 00 29 02 00 00 00	.com.... ..)

Now you can select on any packet to check that particular packet. After clicking on a particular packet you can see the information about different layers of TCP/IP Protocol associated with it.



Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	192.168.43.1	DNS	100	Sta
2	-0.000494259	127.0.0.1	127.0.0.53	DNS	100	Sta
3	-0.000430393	127.0.0.1	127.0.0.53	DNS	100	Sta
4	0.000438811	127.0.0.53	127.0.0.1	DNS	100	Sta
5	0.300823353	127.0.0.53	127.0.0.1	DNS	148	Sta
6	0.300522242	192.168.43.1	10.0.2.15	DNS	148	Sta
7	1.000538308	10.0.2.15	35.224.170.84	TCP	74	410
8	1.467818611	35.224.170.84	10.0.2.15	TCP	60	80
9	1.467939670	10.0.2.15	35.224.170.84	TCP	54	410
10	1.468359017	10.0.2.15	35.224.170.84	HTTP	141	GET
11	1.468937749	35.224.170.84	10.0.2.15	TCP	60	80

▶ Frame 8: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface e  
 ▶ Ethernet II, Src: RealtekU\_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu\_25:23:01 (3  
 ▶ Internet Protocol Version 4, Src: 35.224.170.84, Dst: 10.0.2.15  
 ▶ Transmission Control Protocol, Src Port: 80, Dst Port: 41052, Seq: 0, Ack: 1, Len

0000	08 00 27 25 23 01 52 54	00 12 35 02 08 00 45 00	..'%#·RT··5···E·
0010	00 2c 02 98 00 00 40 06	9d f1 23 e0 aa 54 0a 00	,····@· ··#··T··
0020	02 0f 00 50 a0 5c 00 7b	0c 01 8c 68 c8 de 60 12	···P·\·{···h···

You can also see the RAW data of that particular packet.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	192.168.43.1	DNS	100	Sta
2	-0.000494259	127.0.0.1	127.0.0.53	DNS	100	Sta
3	-0.000430393	127.0.0.1	127.0.0.53	DNS	100	Sta
4	0.000438811	127.0.0.53	127.0.0.1	DNS	100	Sta
5	0.300823353	127.0.0.53	127.0.0.1	DNS	148	Sta
6	0.300522242	192.168.43.1	10.0.2.15	DNS	148	Sta
7	1.000538308	10.0.2.15	35.224.170.84	TCP	74	410
8	1.467818611	35.224.170.84	10.0.2.15	TCP	60	80
9	1.467939670	10.0.2.15	35.224.170.84	TCP	54	410
10	1.468359017	10.0.2.15	35.224.170.84	HTTP	141	GET
11	1.468937749	35.224.170.84	10.0.2.15	TCP	60	80

▶ Frame 8: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface  
 ▶ Ethernet II, Src: RealtekU\_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu\_25:23:01  
 ▶ Internet Protocol Version 4, Src: 35.224.170.84, Dst: 10.0.2.15  
 ▶ Transmission Control Protocol, Src Port: 80, Dst Port: 41052, Seq: 0, Ack: 1, Len

0000	08 00 27 25 23 01 52 54	00 12 35 02 08 00 45 00	..'%#·RT··5···E·
0010	00 2c 02 98 00 00 40 06	9d f1 23 e0 aa 54 0a 00	,····@· ··#··T··
0020	02 0f 00 50 a0 5c 00 7b	0c 01 8c 68 c8 de 60 12	···P·\·{···h···
0030	ff ff bb 63 00 00 02 04	05 b4 00 00	···c····

enp0s3 and Loopback: lo:...ive capture in progress: Packets: 150 · Displayed: 150 (100.0%) Profile: Default



You can also click on the arrows to expand packet data for a particular TCP/IP Protocol Layer

**Capturing from enp0s3 and Loopback: lo**

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

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

No.	Time	Source	Destination	Protocol	Length	In
4	0.000438811	127.0.0.53	127.0.0.1	DNS	100	S
5	0.300823353	127.0.0.53	127.0.0.1	DNS	148	S
6	0.300522242	192.168.43.1	10.0.2.15	DNS	148	S
7	1.000538308	10.0.2.15	35.224.170.84	TCP	74	4
8	1.467818611	35.224.170.84	10.0.2.15	TCP	60	8
9	1.467939670	10.0.2.15	35.224.170.84	TCP	54	4

**Frame 8: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface Ethernet II, Src: RealtekU\_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu\_25:23:01 (08:00:27:25:23:01)**

- Destination: PcsCompu\_25:23:01 (08:00:27:25:23:01)
- Source: RealtekU\_12:35:02 (52:54:00:12:35:02)
- Type: IPv4 (0x0800)
- Padding: 0000
- Internet Protocol Version 4, Src: 35.224.170.84, Dst: 10.0.2.15
- Transmission Control Protocol, Src Port: 80, Dst Port: 41052, Seq: 0, Ack: 1, L

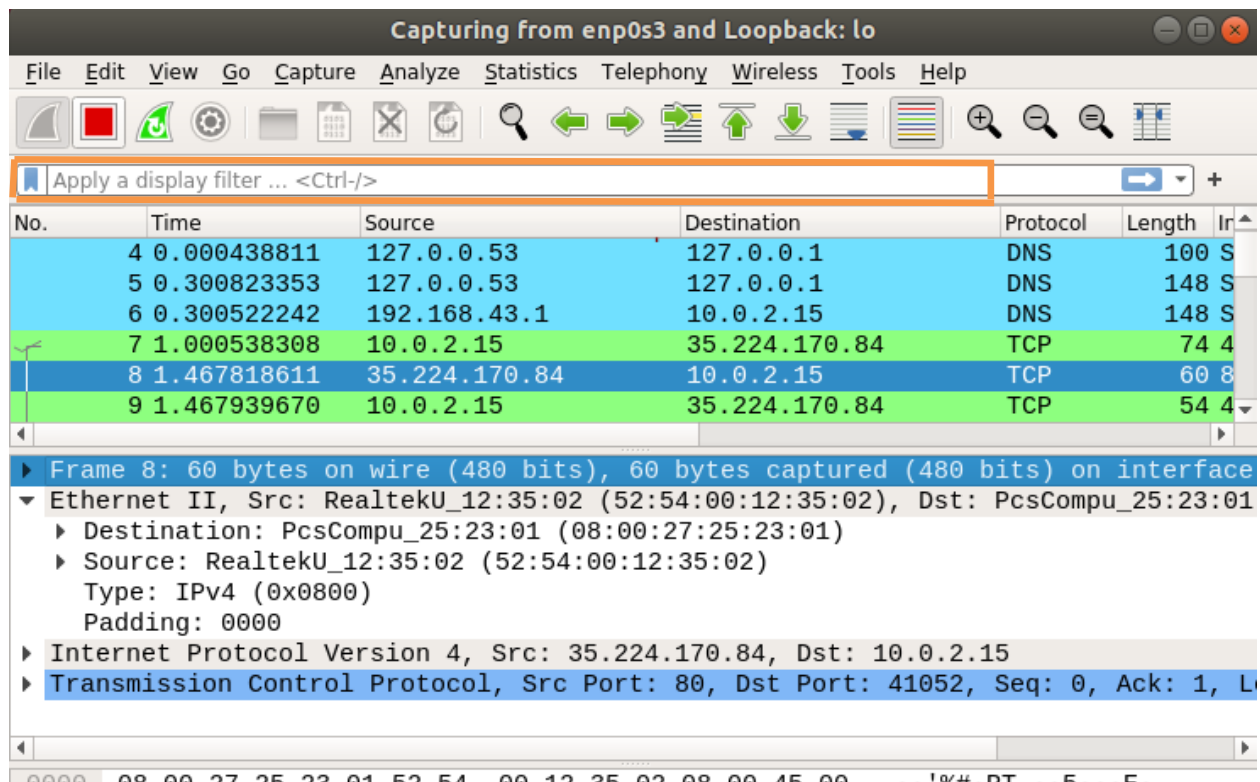
0000 08 00 27 25 23 01 52 54 00 12 35 02 08 00 45 00 . . '%# . RT . . 5 . . E .  
0010 00 2c 02 98 00 00 40 06 9d f1 23 e0 aa 54 0a 00 . , . . . @ . . # . . T . .  
0020 02 0f 00 50 a0 5c 00 7b 0c 01 8c 68 c8 de 60 12 . . . P . \ . { . . h . .` .

## Filtering Packets Using Wireshark:

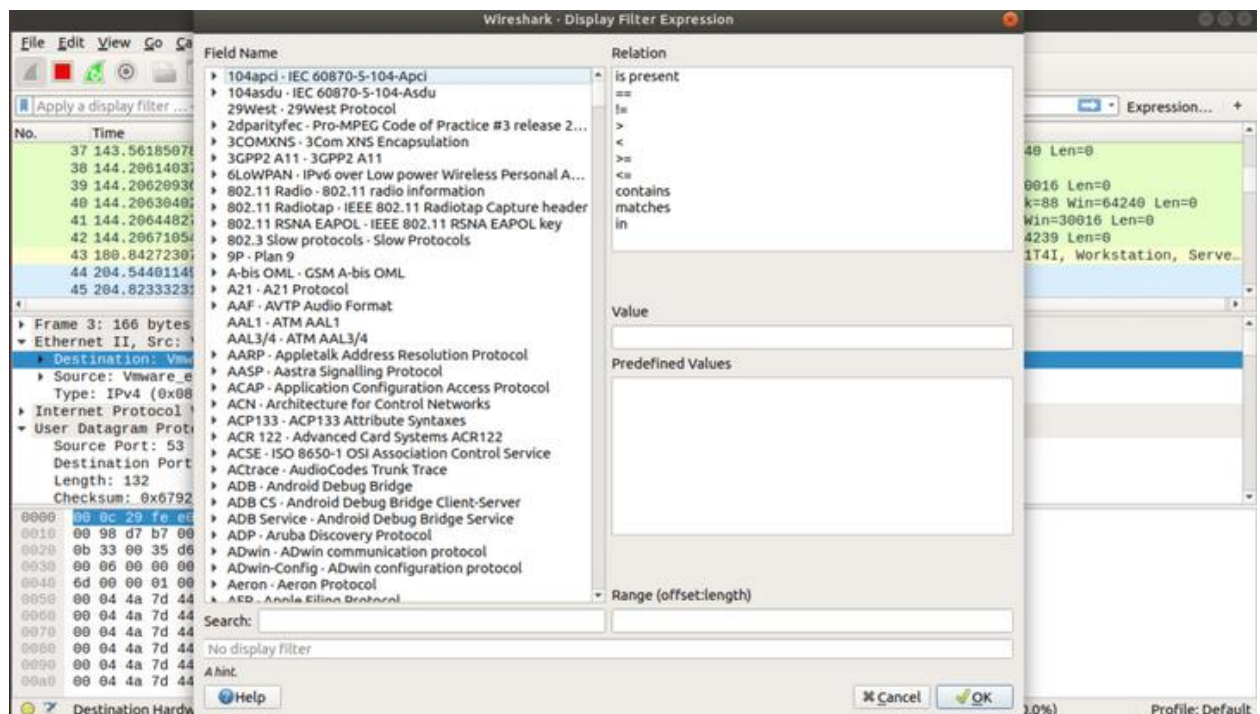
On a busy network thousands or millions of packets will be captured each second. So the list will be so long that it will be nearly impossible to scroll through the list and search for certain type of packet.

The good thing is, in Wireshark, you can filter the packets and see only the packets that you need.

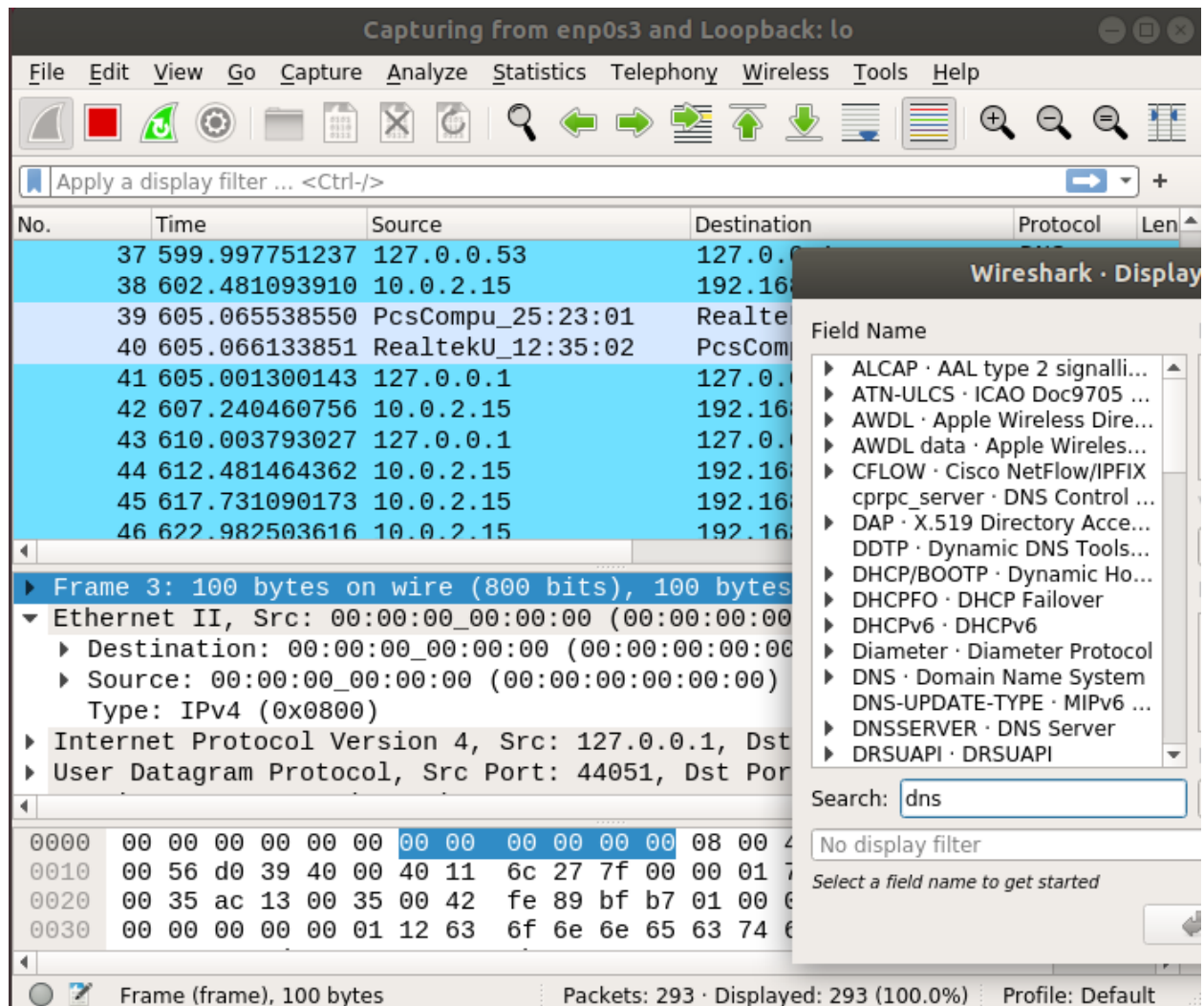
To filter packets, you can directly type in the filter expression in the textbox as marked in the screenshot below.



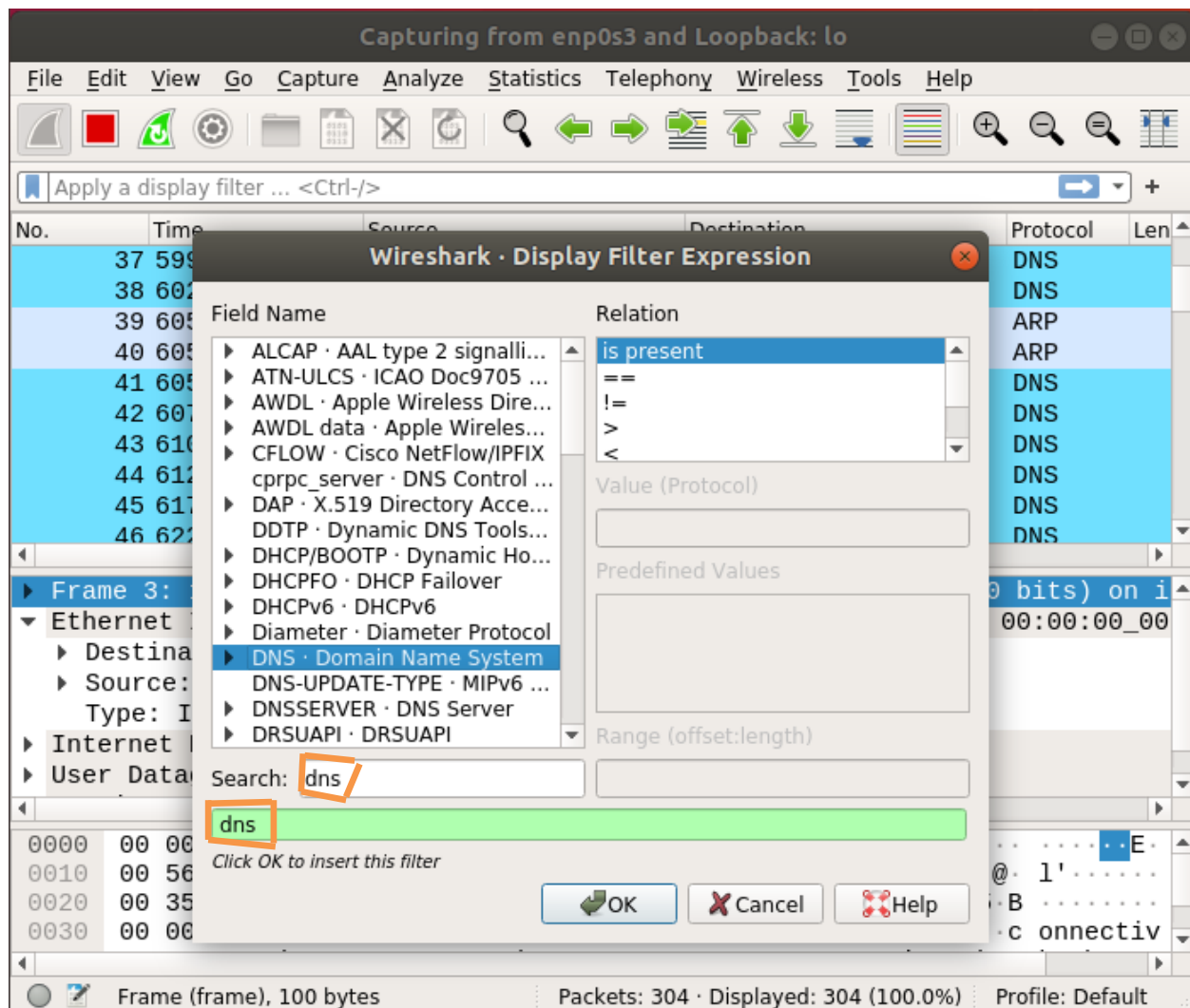
You can also filter packets captured by Wireshark graphically. To do that, click on the Expression button. From here you can create filter expression to search packets very specifically.



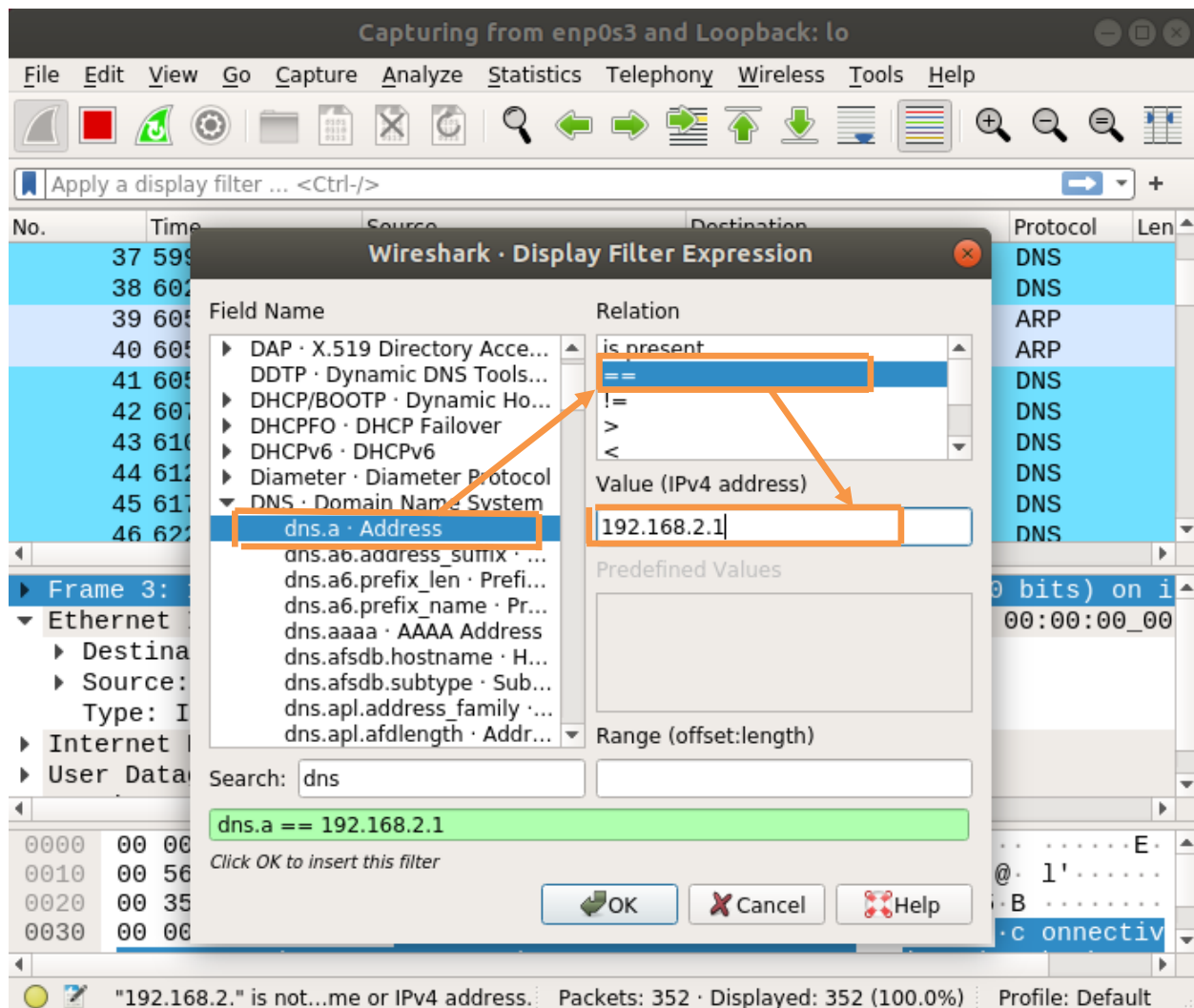
In the **Field Name** section almost all the networking protocols are listed. The list is huge. You can type in what protocol you're looking for in the **Search** textbox and the **Field Name** section would show the ones that matched.



In this article, I am going to filter out all the DNS packets. So I selected **DNS Domain Name System** from the **Field Name** list. You can also click on the **arrow** on any protocol.

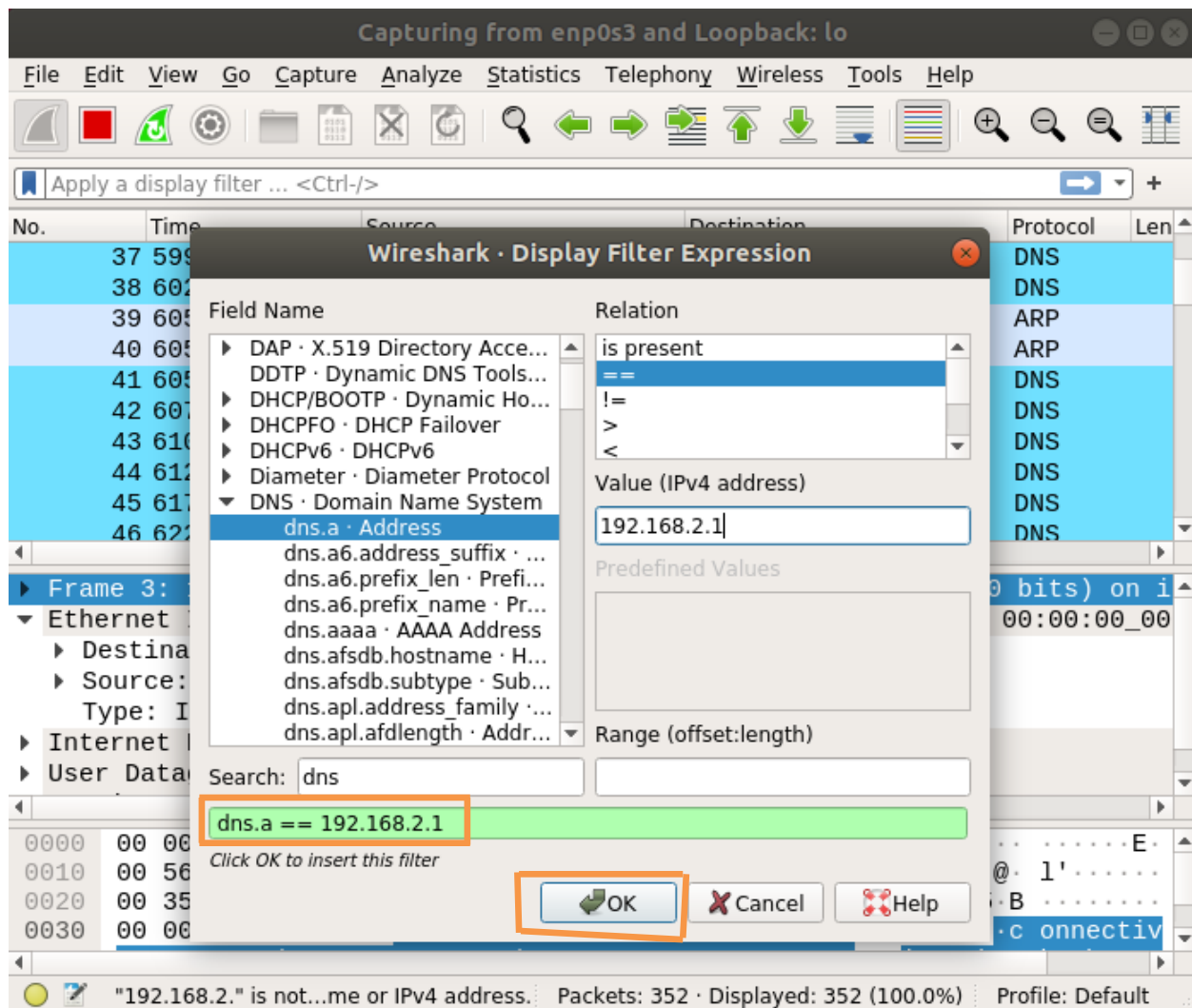


You can also use relational operators to test whether some field is equal to, not equal to, great than or less than some value. I searched for all the **DNS IPv4** address which is equal to **192.168.2.1** as you can see in the screenshot below.

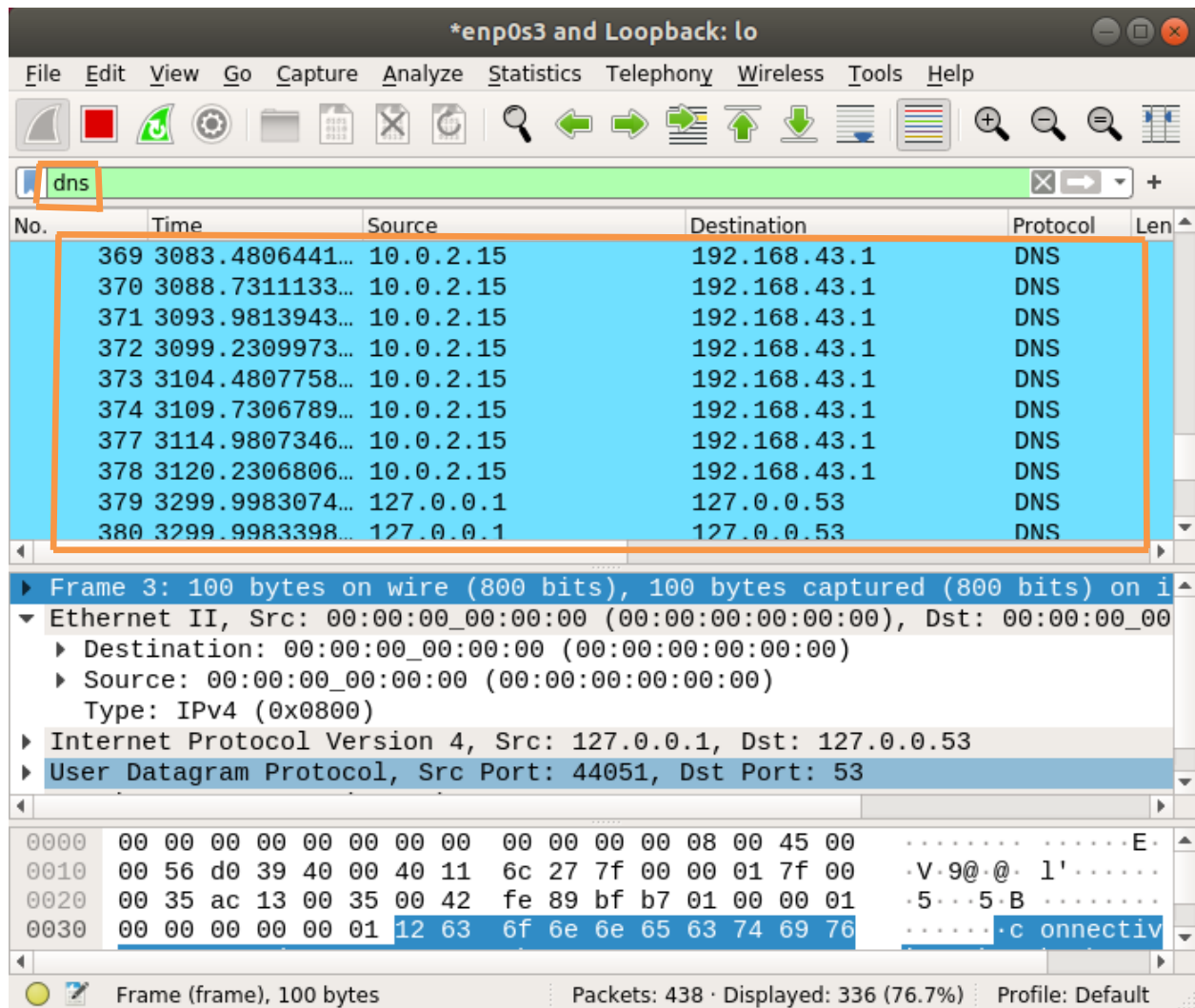


The filter expression is also shown in the marked section of the screenshot below. This is a great way to learn how to write filter expression in Wireshark. Once you're done, just click on **OK**





Now click on the marked icon to Apply the filter. As you can see, only the DNS protocol packets are shown.



## Stopping Packet Capture in Wireshark:

You can click on the red icon as marked in the screenshot below to stop capturing Wireshark packets.



**\*enp0s3 and Loopback: lo**

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

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

No.	Time	Source	Destination	Protocol	Len
1	0.000000000	10.0.2.15	192.168.43.1	DNS	
2	-0.000494259	127.0.0.1	127.0.0.53	DNS	
3	-0.000430393	127.0.0.1	127.0.0.53	DNS	
4	0.000438811	127.0.0.53	127.0.0.1	DNS	
5	0.300823353	127.0.0.53	127.0.0.1	DNS	
6	0.300522242	192.168.43.1	10.0.2.15	DNS	
7	1.000538308	10.0.2.15	35.224.170.84	TCP	
8	1.467818611	35.224.170.84	10.0.2.15	TCP	

▶ Frame 3: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface

▼ Ethernet II, Src: 00:00:00\_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00\_00:00:00 (00:00:00:00:00:00)

- ▶ Destination: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- ▶ Source: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- Type: IPv4 (0x0800)
- ▶ Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.53
- ▶ User Datagram Protocol, Src Port: 44051, Dst Port: 53

Offset	Hex	ASCII
0000	00 00 00 00 00 00 00 00 00 00 00 00 08 00 45 00	.....E.
0010	00 56 d0 39 40 00 40 11 6c 27 7f 00 00 01 7f 00	.V.9@.@.1'.....
0020	00 35 ac 13 00 35 00 42 fe 89 bf b7 01 00 00 01	.5...5.B.....
0030	00 00 00 00 00 01 12 63 6f 6e 6e 65 63 74 69 76	.....c onnectiv
0040	69 74 79 2d 63 68 65 63 6b 06 75 62 75 6e 74 75	ity-check k.ubuntu
0050	03 63 6f 6d 00 00 1c 00 01 00 00 29 04 b0 00 00	.com.....)

## Saving Captured Packets to a File:

You can click on the marked icon to save captured packets to a file for future use.

Firefox Web Browser \*enp0s3 and Loopback: lo

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

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

No.	Time	Source	Destination	Protocol
1	0.000000000	10.0.2.15	192.168.43.1	DNS
2	-0.000494259	127.0.0.1	127.0.0.53	DNS
3	-0.000430393	127.0.0.1	127.0.0.53	DNS
4	0.000438811	127.0.0.53	127.0.0.1	DNS
5	0.300823353	127.0.0.53	127.0.0.1	DNS
6	0.300522242	192.168.43.1	10.0.2.15	DNS
7	1.000538308	10.0.2.15	35.224.170.84	TCP
8	1.467818611	35.224.170.84	10.0.2.15	TCP

Frame 3: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface

Ethernet II, Src: 00:00:00\_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00\_00:00:00

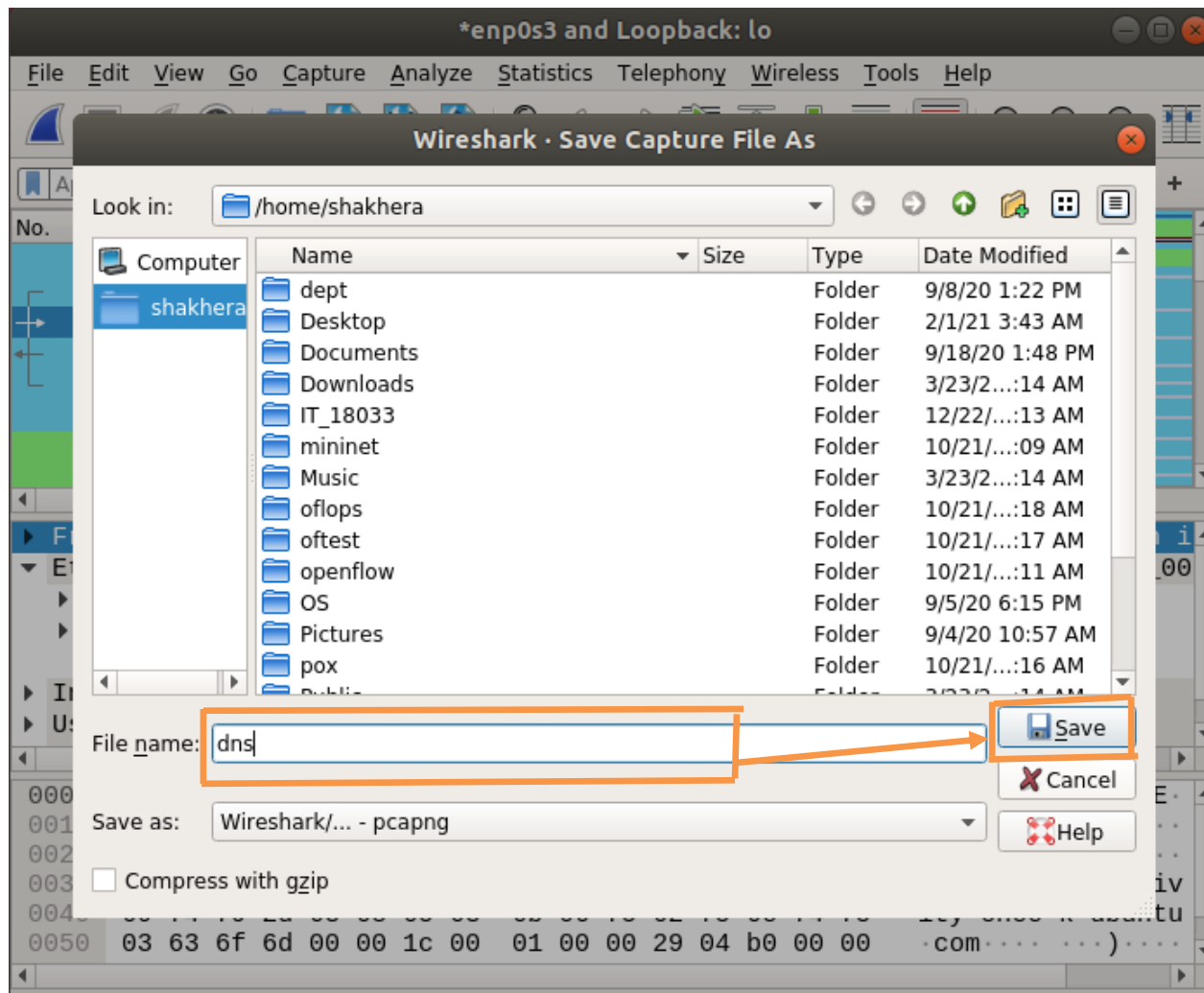
- Destination: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- Source: 00:00:00\_00:00:00 (00:00:00:00:00:00)
- Type: IPv4 (0x0800)

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.53

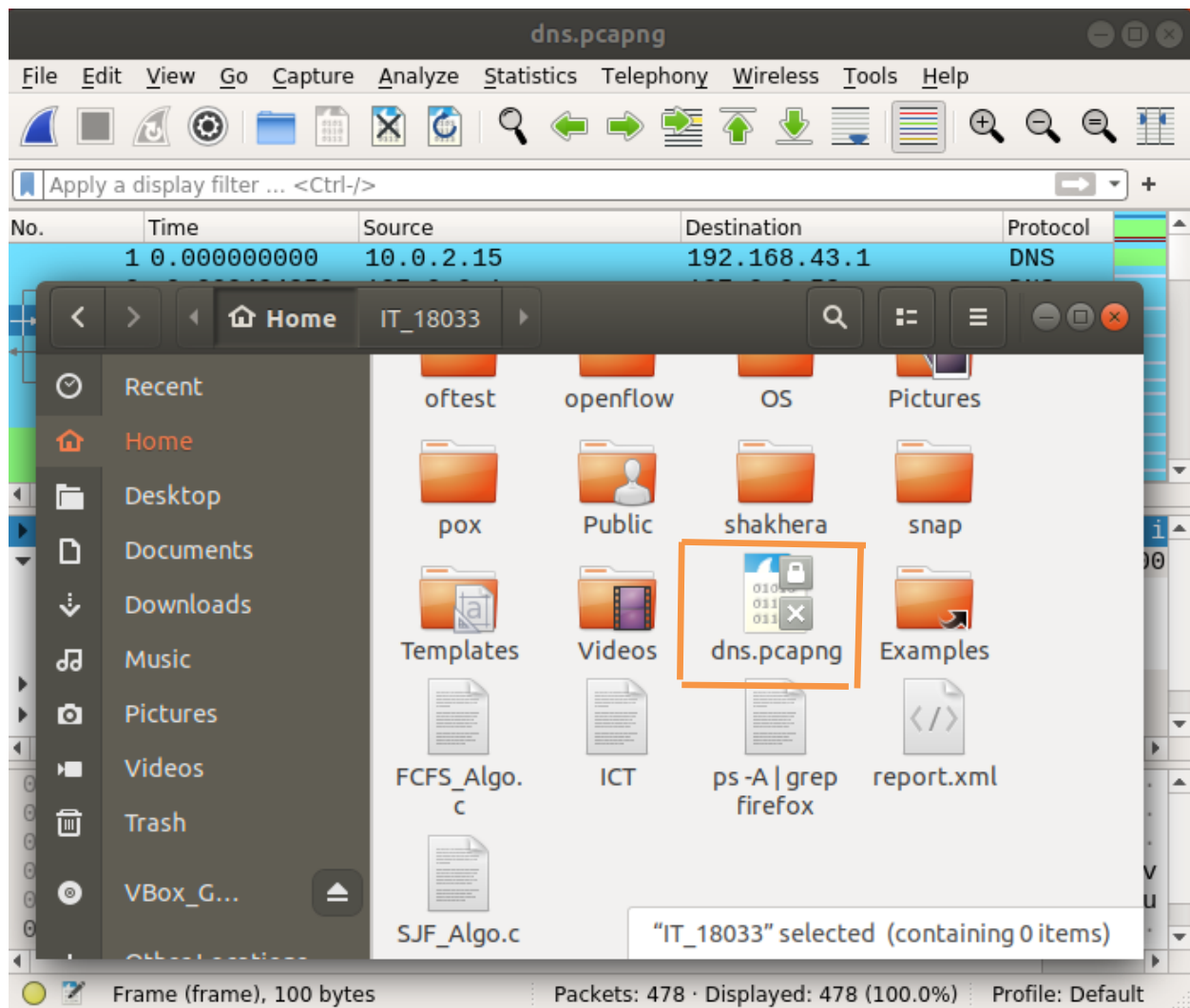
User Datagram Protocol, Src Port: 44051, Dst Port: 53

Offset	Hex	ASCII
0000	00 00 00 00 00 00 00 00 00 00 00 00 08 00 45 00	.....E.
0010	00 56 d0 39 40 00 40 11 6c 27 7f 00 00 01 7f 00	.V.9@. 1'.....
0020	00 35 ac 13 00 35 00 42 fe 89 bf b7 01 00 00 01	.5...5.B .....
0030	00 00 00 00 00 01 12 63 6f 6e 6e 65 63 74 69 76	.....c onnectiv
0040	69 74 79 2d 63 68 65 63 6b 06 75 62 75 6e 74 75	ity-check k.ubuntu
0050	03 63 6f 6d 00 00 1c 00 01 00 29 04 b0 00 00	.com....(. )....

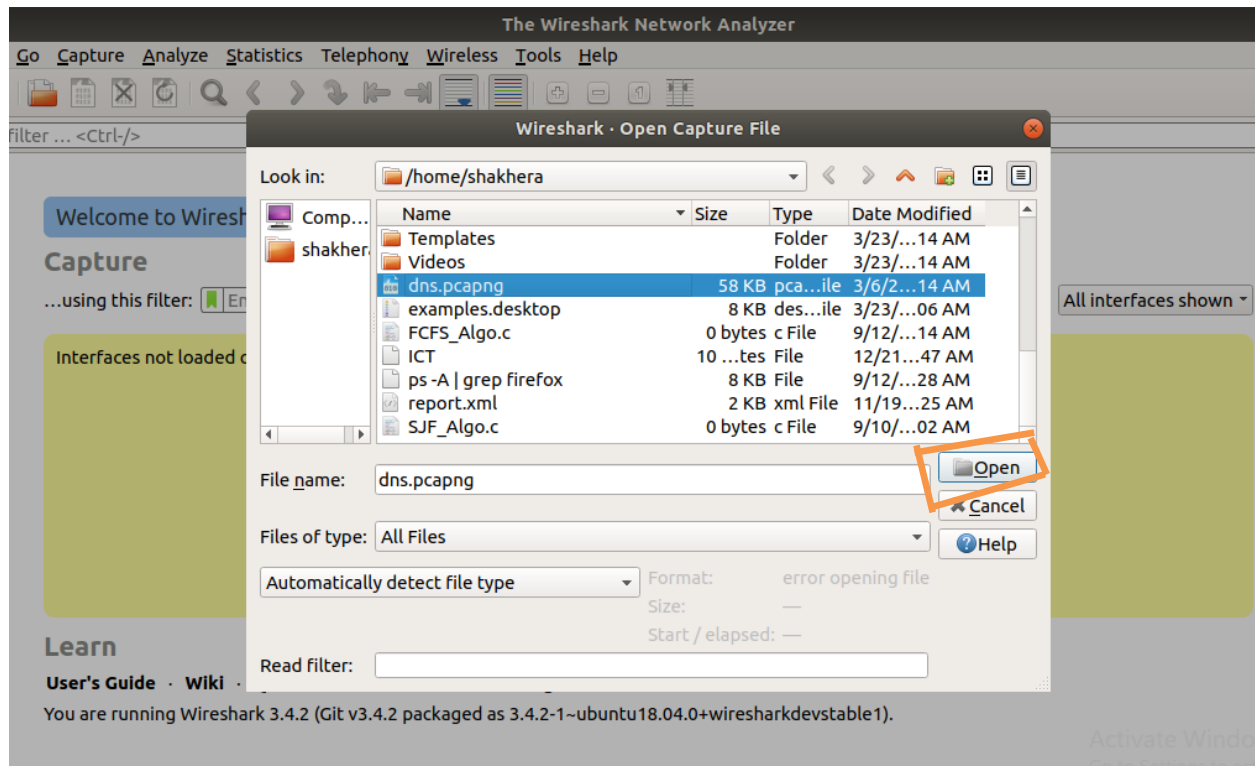
Now select a destination folder, type in the file name and click on **Save**



The file should be saved.



Now you can open and analyze the saved packets anytime. To open the file, go to File > Open from Wireshark or press <Ctrl> + o  
Then select the file and click on Open.



The captured packets should be loaded from the file

Capturing from enp0s3 and Loopback: lo						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Apply a display filter ... <Ctrl-/>						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	192.168.43.1	DNS	100	Standard query query
2	-0.000494259	127.0.0.1	127.0.0.53	DNS	100	Standard query response
3	-0.000430393	127.0.0.1	127.0.0.53	DNS	100	Standard query response
4	0.000438811	127.0.0.53	127.0.0.1	DNS	100	Standard query response
5	0.300823353	127.0.0.53	127.0.0.1	DNS	148	Standard query response
6	0.300522242	192.168.43.1	10.0.2.15	DNS	148	Standard query response
7	1.000538308	10.0.2.15	35.224.170.84	TCP	74	4101
8	1.467818611	35.224.170.84	10.0.2.15	TCP	60	80
9	1.467939670	10.0.2.15	35.224.170.84	TCP	54	4101
10	1.468359017	10.0.2.15	35.224.170.84	HTTP	141	GET
11	1.468937749	35.224.170.84	10.0.2.15	TCP	60	80
12	1.883424866	35.224.170.84	10.0.2.15	HTTP	202	HTTP
▶ Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface ▶ Ethernet II, Src: PcsCompu_25:23:01 (08:00:27:25:23:01), Dst: RealtekU_12:35:02 ▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.43.1 ▶ User Datagram Protocol, Src Port: 33939, Dst Port: 53						
0000	52 54 00 12 35 02 08 00	27 25 23 01 08 00	45 00	RT..5...	'%#..E.	
0010	00 56 b6 d8 40 00 40 11	d7 06 0a 00 02 0f c0 a8		.Vk.@.@.	.....	
0020	2b 01 84 93 00 35 00 42	f8 0b 88 10 01 00 00 01		+...5.B	.....	
0030	00 00 00 00 00 01 12 63	6f 6e 6e 65 63 74 69 76		.....c	onnectiv	
0040	69 74 79 2d 63 68 65 63	6b 06 75 62 75 6e 74 75		ity-chec	k.ubuntu	
0050	03 63 6f 6d 00 00 01 00	01 00 00 29 02 00 00 00		.com....	....)	
enp0s3 and Loopback: lo:...live capture in progress: Packets: 136 · Displayed: 136 (100.0%) Profile: Default						

**Discussion:** Using this lab, I have to know about installing Wireshark and uses. Wireshark is the best tool for network analysis and packet investigation, and is an open-source and freely available network analyzing tool. Wireshark supports many different communication protocols. There are many options and features that provides you the power to capture and analyze the network packets in a unique way.

Using Wireshark we can capture live packet data from network interface. Opening files containing packet data capture with tcpdump, Wireshark, and many other packet capture program. And also display packets with very detailed protocol information. We can also Save packet data capture. We can filter packets on many criteria, search for packets on many criteria.