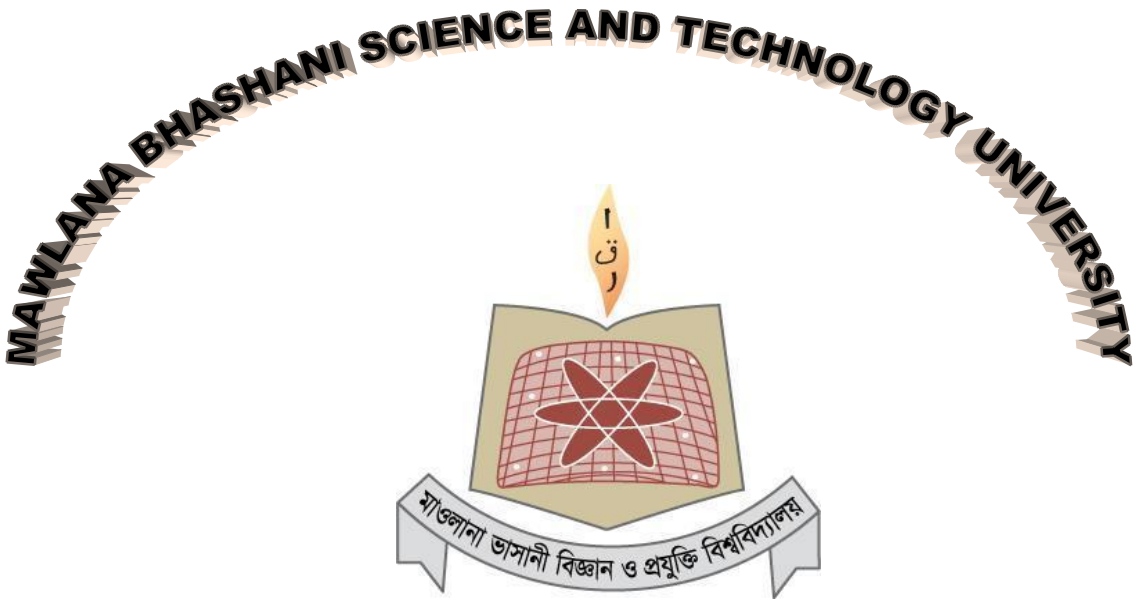


Lab Report: 08

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
3rd year 2nd 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:

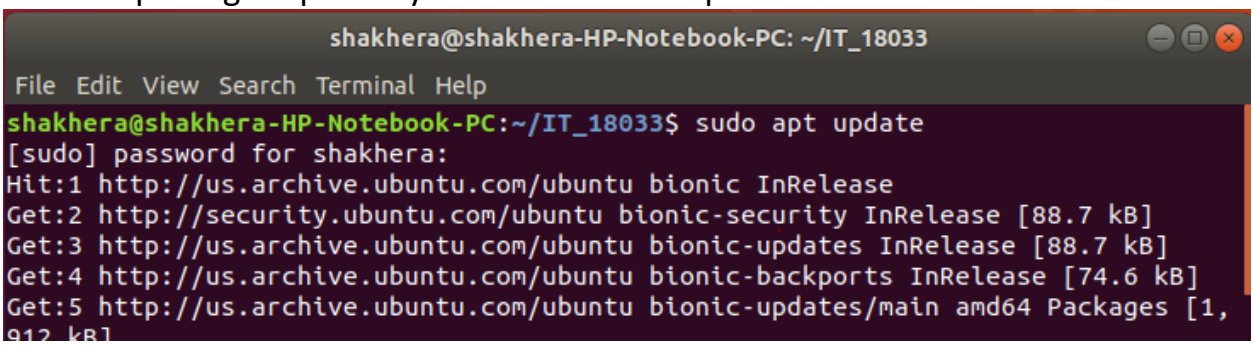
Wireshark: Wireshark is a free and open-source network protocol analyzer widely used around the globe. It captures every packet getting in or out of a network interface and shows them in a nicely formatted text. It is used by Network Engineers all over the world.
Wireshark is cross platform and it is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



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 libwscodecs2 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:
  geoipupdate geoip-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

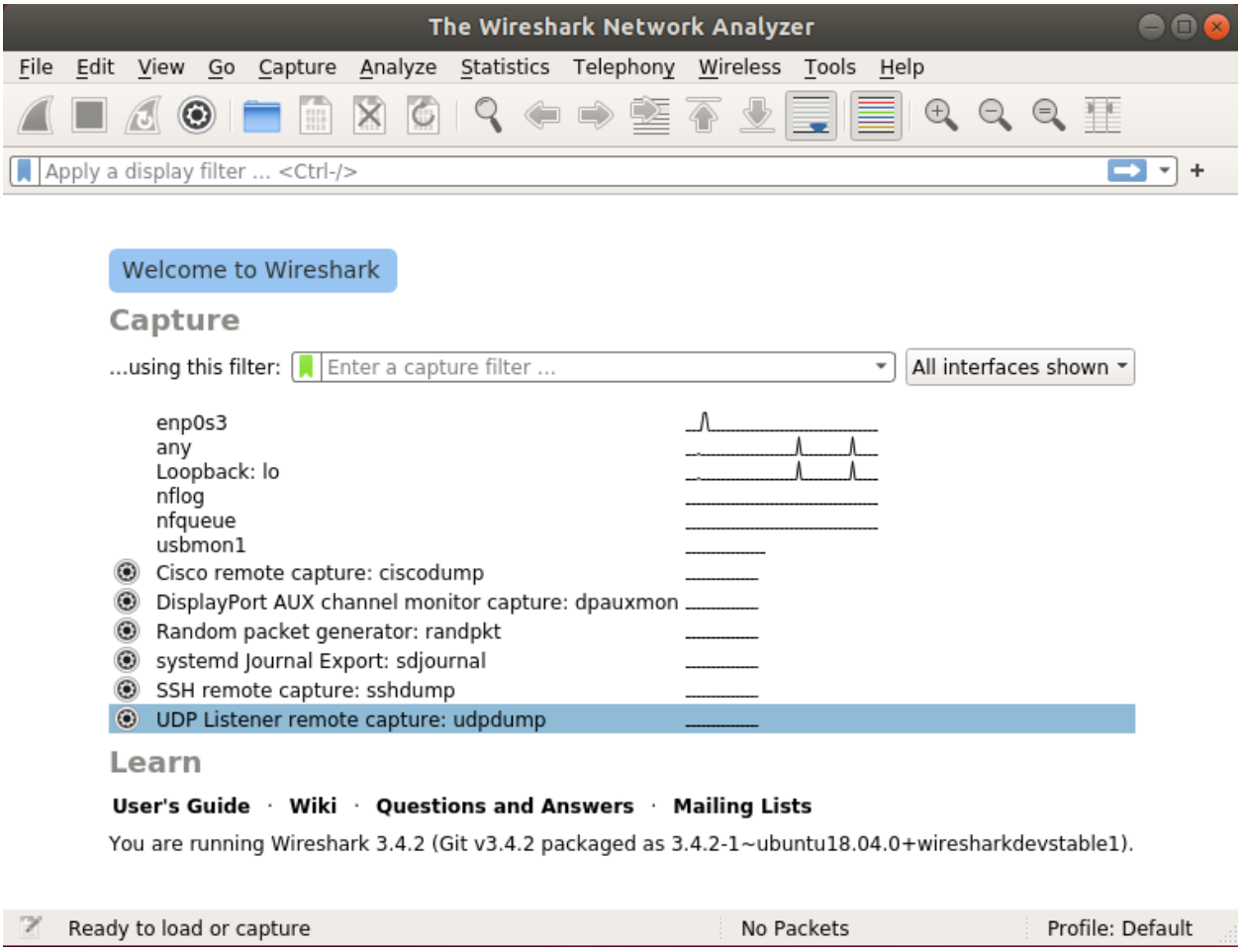
Starting Wireshark:

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

$ sudo wireshark

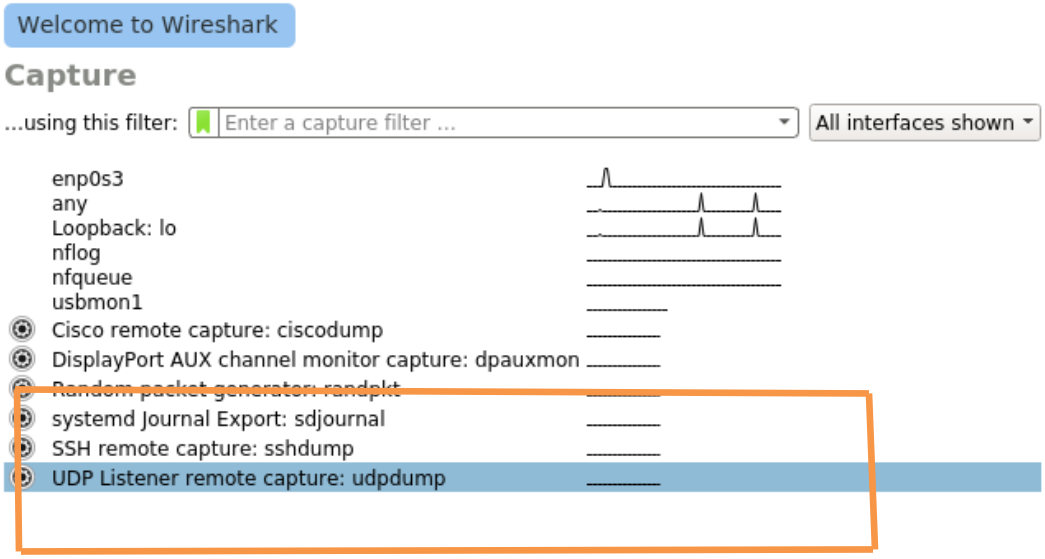
Wireshark will start in your computer.

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

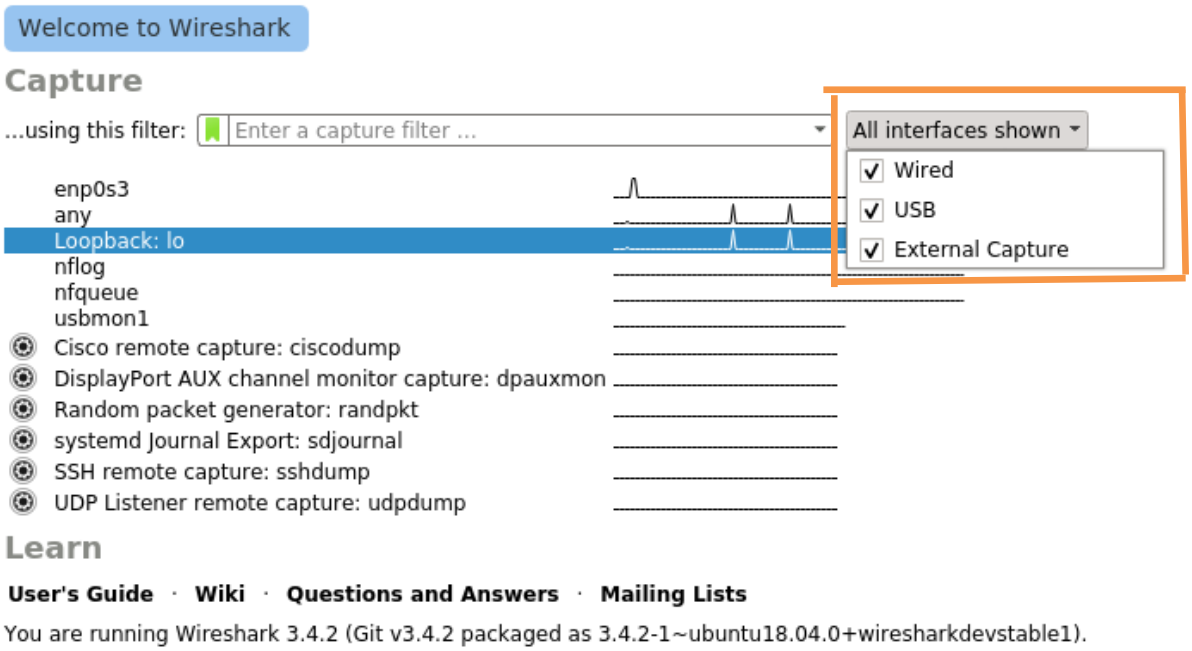


Capturing Packets Using Wireshark:

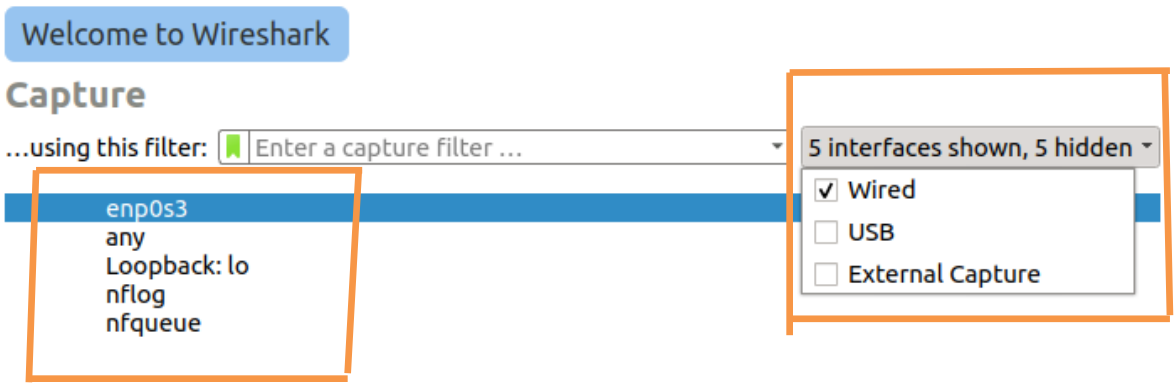
When you start Wireshark, you will see a list of interfaces that you can capture packets to and from.



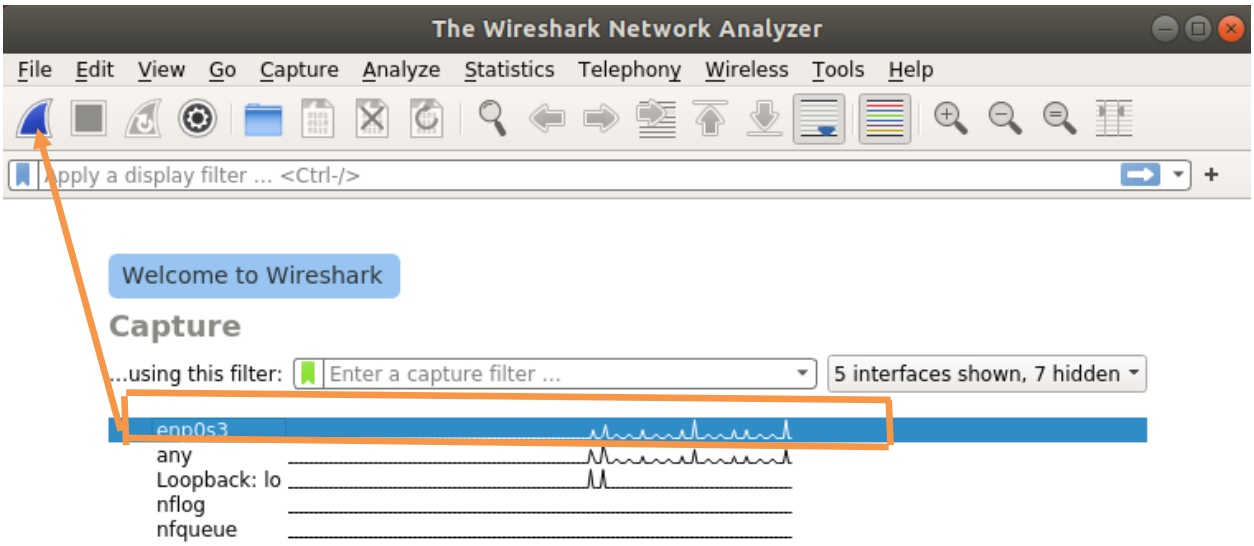
There are many types of interfaces you can monitor using Wireshark, for example, **Wired, USB**, and many external devices. You can choose to show specific types of interfaces in the welcome screen from the marked section of the screenshot below.



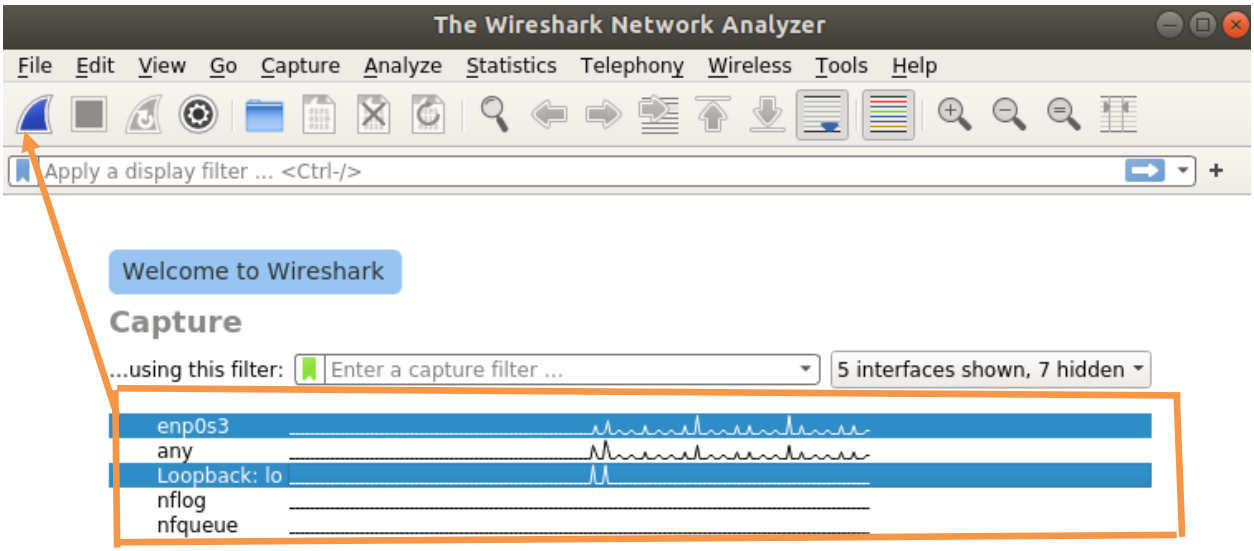
Here, I listed only the **Wired** network interfaces.



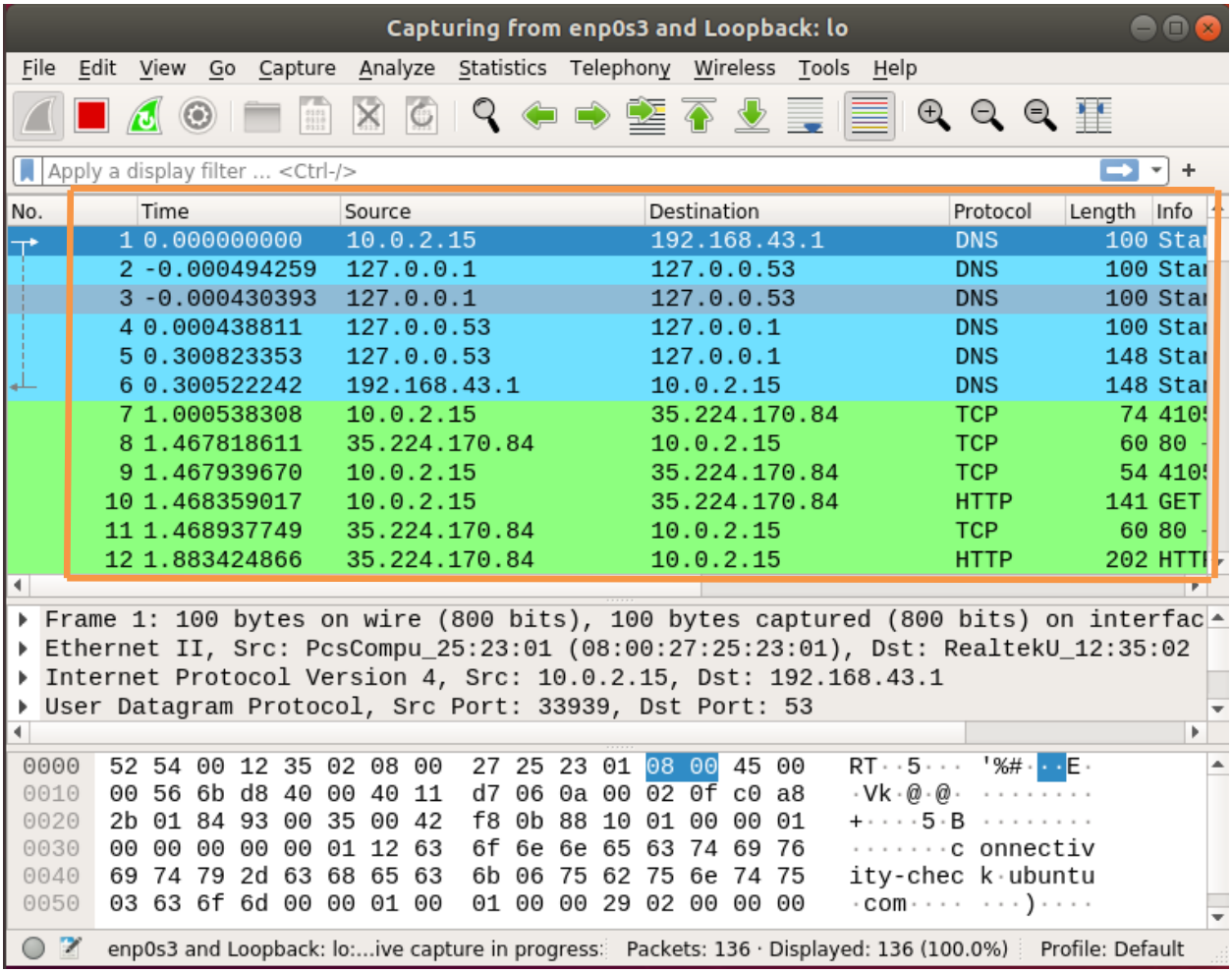
Next, to start capturing packets, you have to select the interface (which in my case is enp0s3) and click on the Start capturing packets icon as marked in the image below



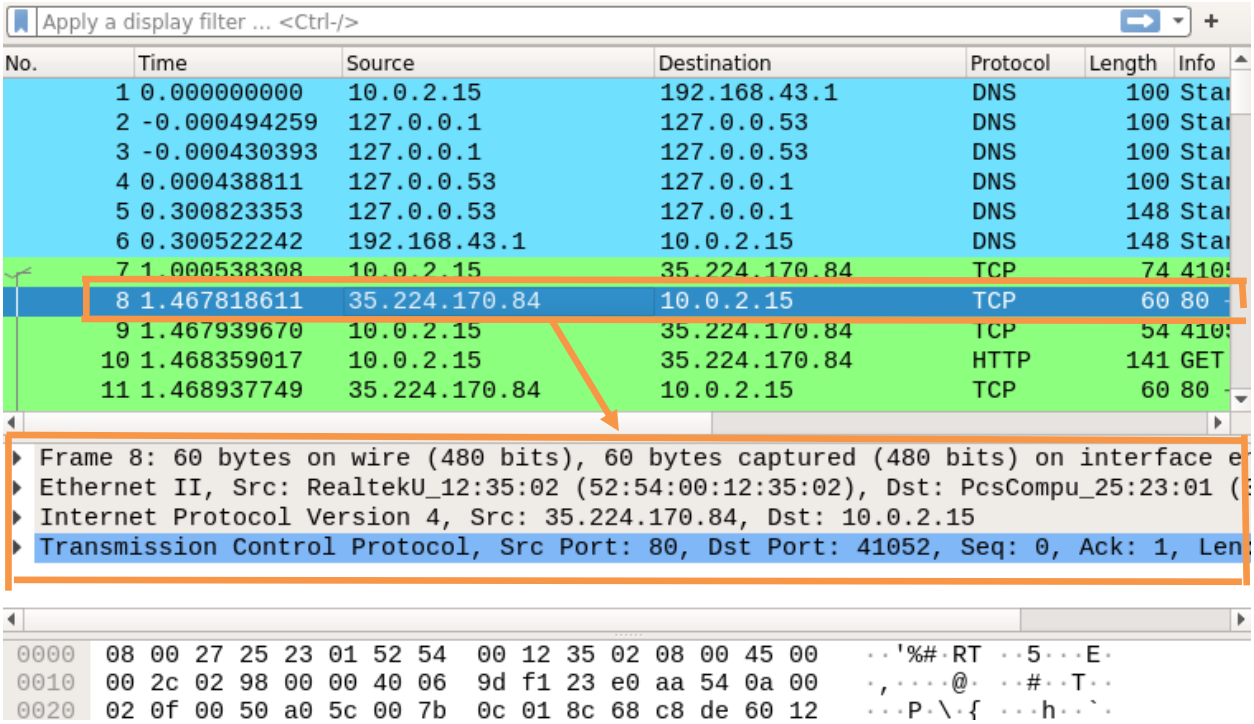
You can also capture packets to and from multiple interfaces at the same time. Just press and hold the CTRL button while clicking on the interfaces that you want to capture to and from and then hit the Start capturing packets icon as marked in the image below.



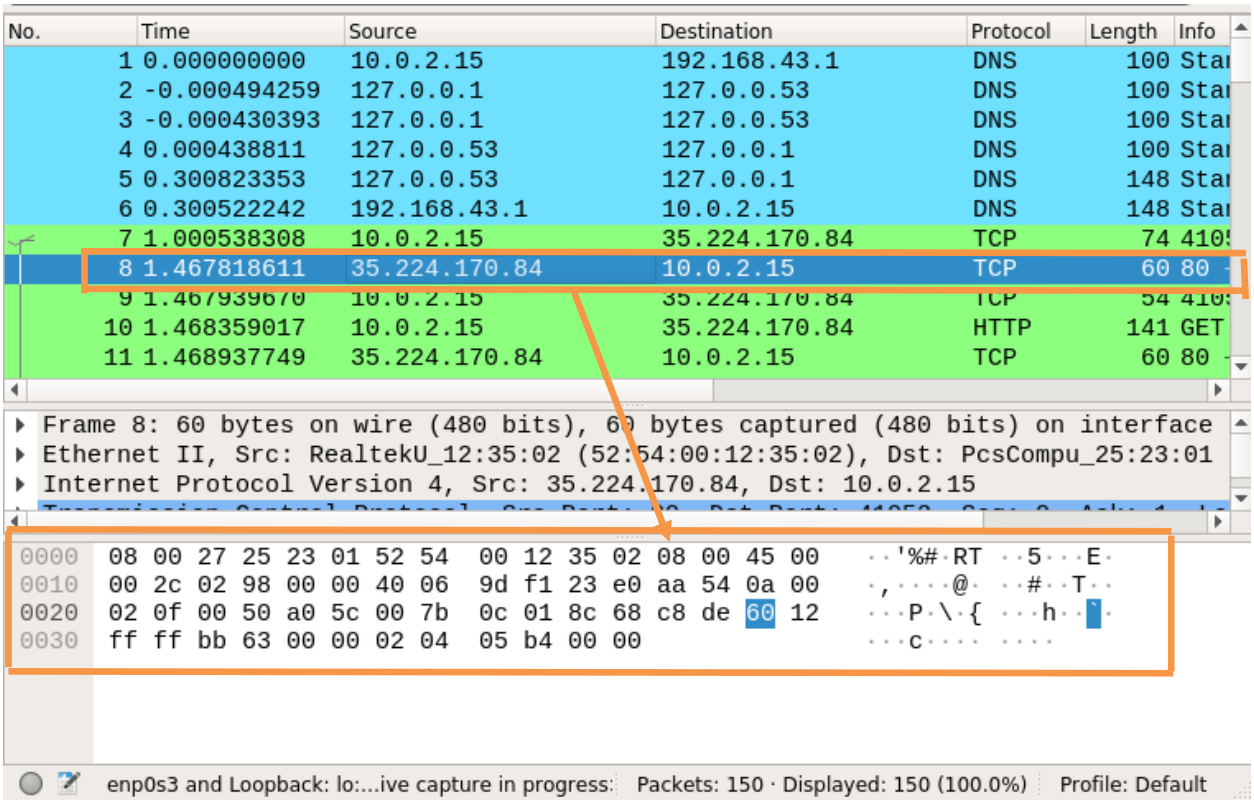
Next, I tried using *ping google.com* command in the terminal and as you can see, many packets were captured



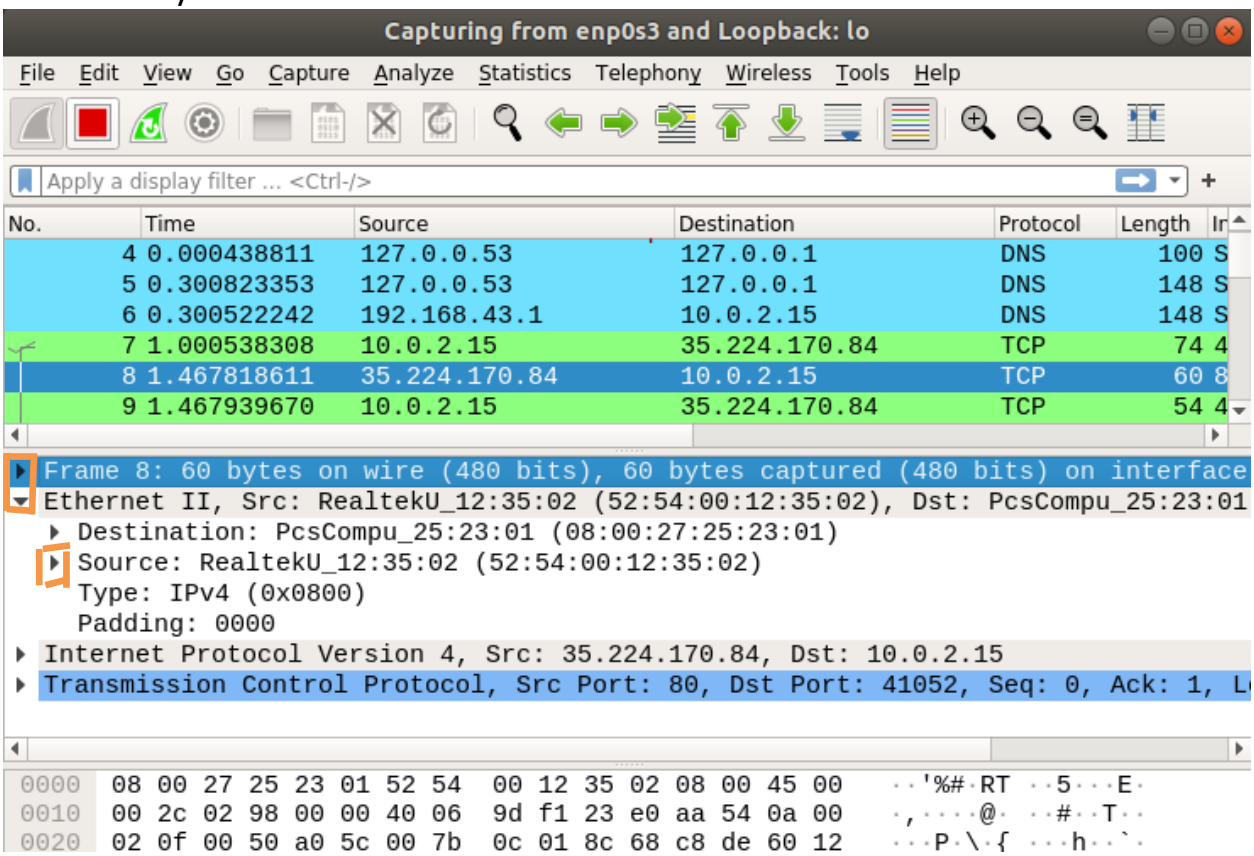
Now you can select on any packet to check that particular packet. After clicking on a particular packet you can see the information about different layers of TCP/IP Protocol associated with it.



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



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

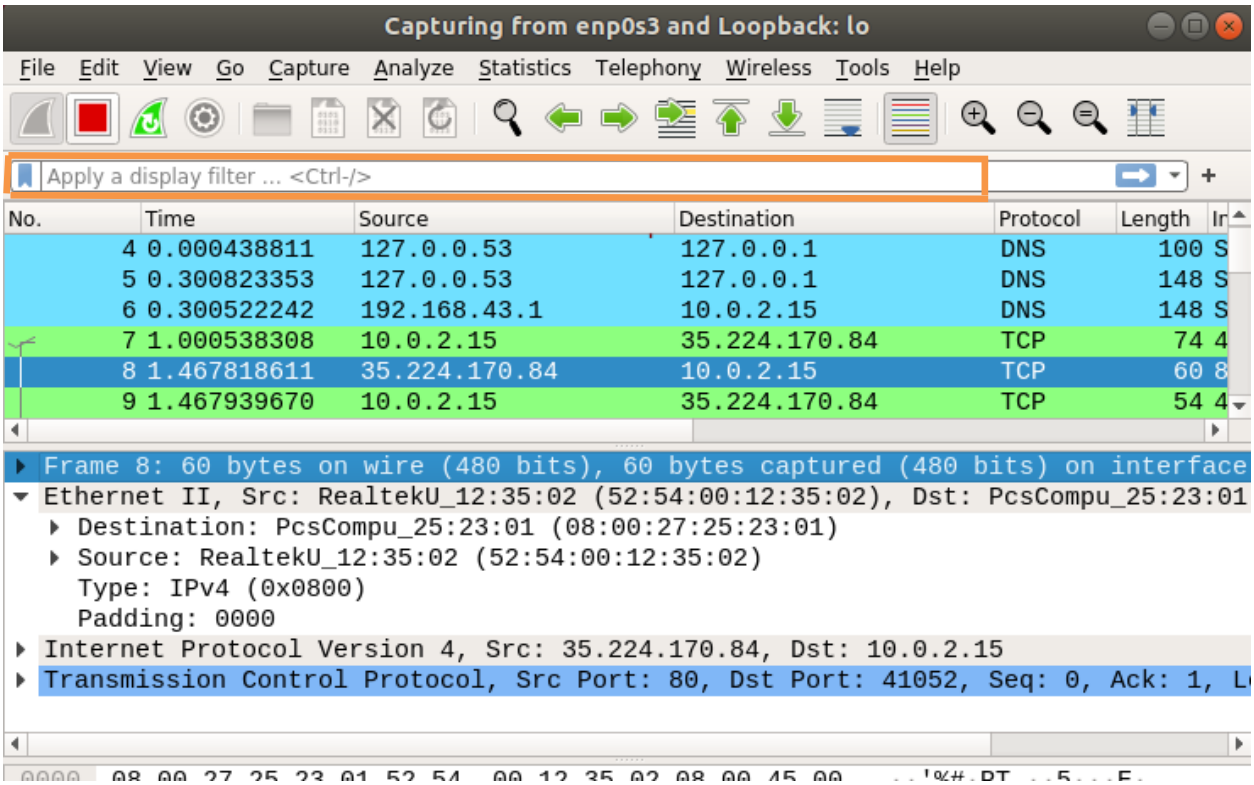


Filtering Packets Using Wireshark:

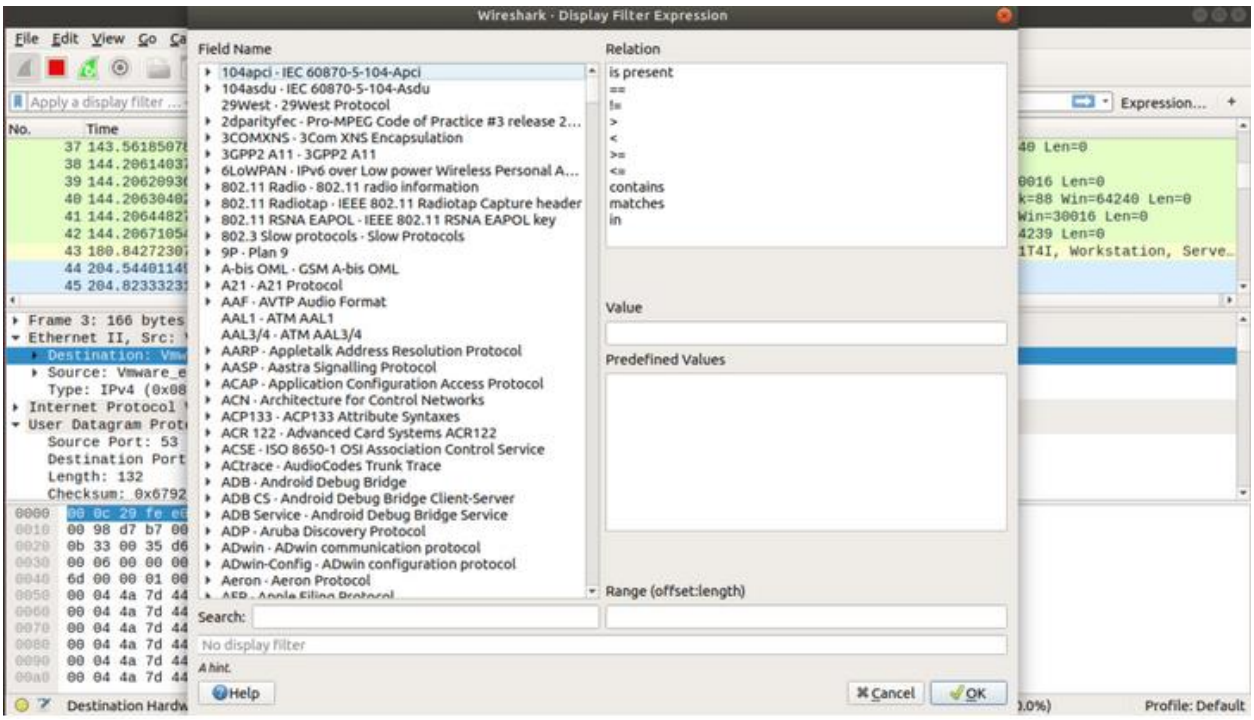
On a busy network thousands or millions of packets will be captured each second. So the list will be so long that it will be nearly impossible to scroll through the list and search for certain type of packet.

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

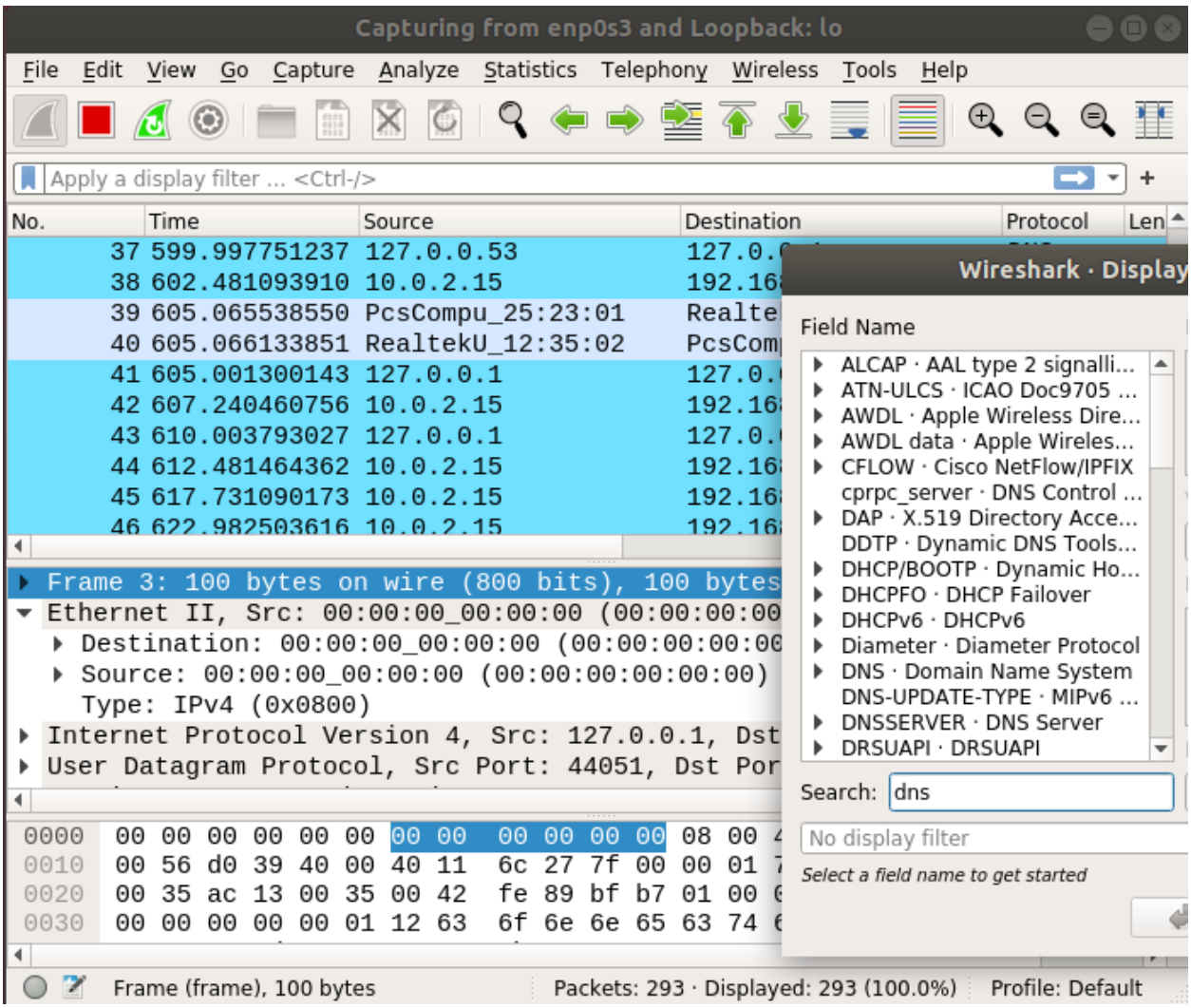
To filter packets, you can directly type in the filter expression in the textbox as marked in the screenshot below.



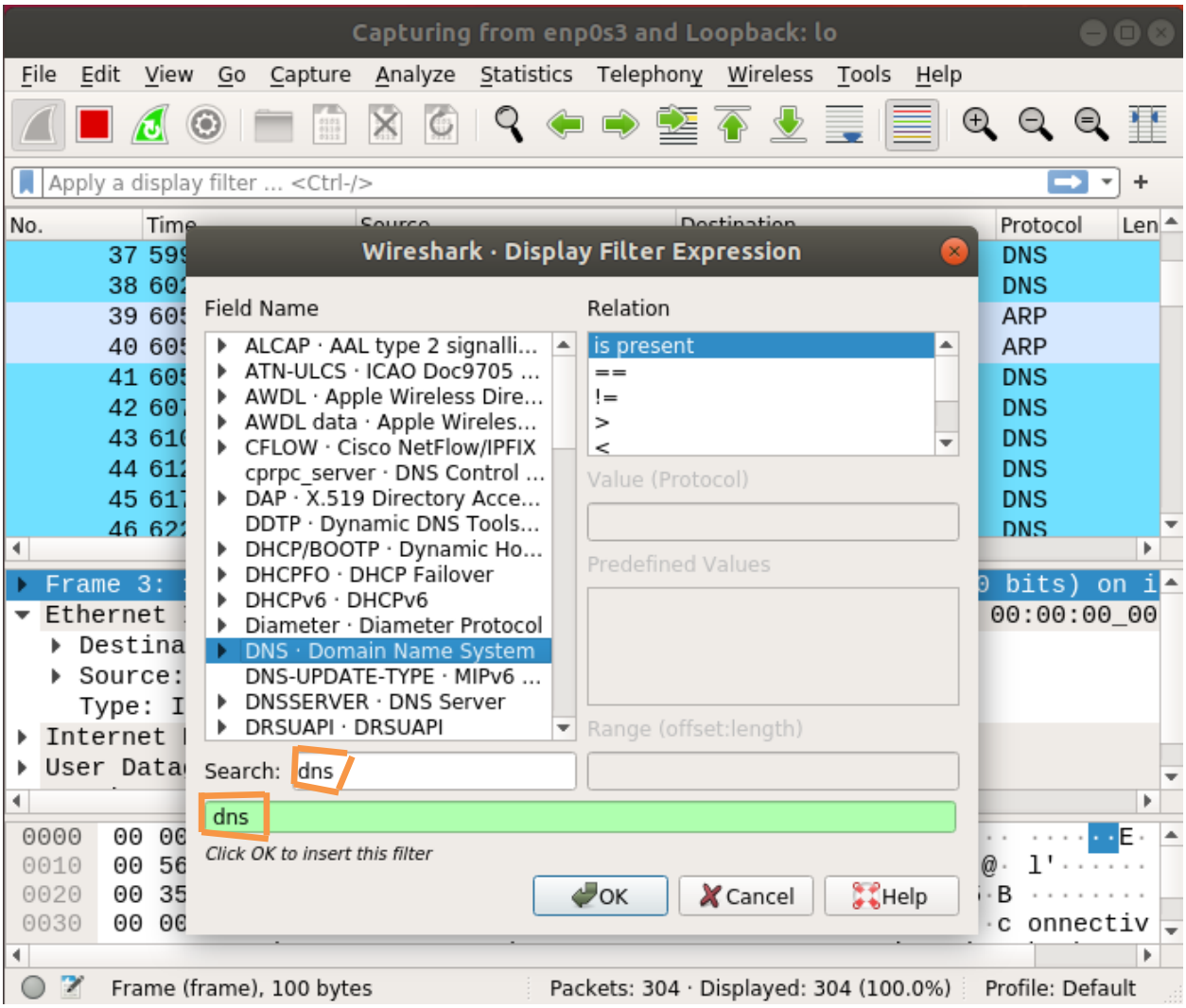
You can also filter packets captured by Wireshark graphically. To do that, click on the Expression button. From here you can create filter expression to search packets very specifically.



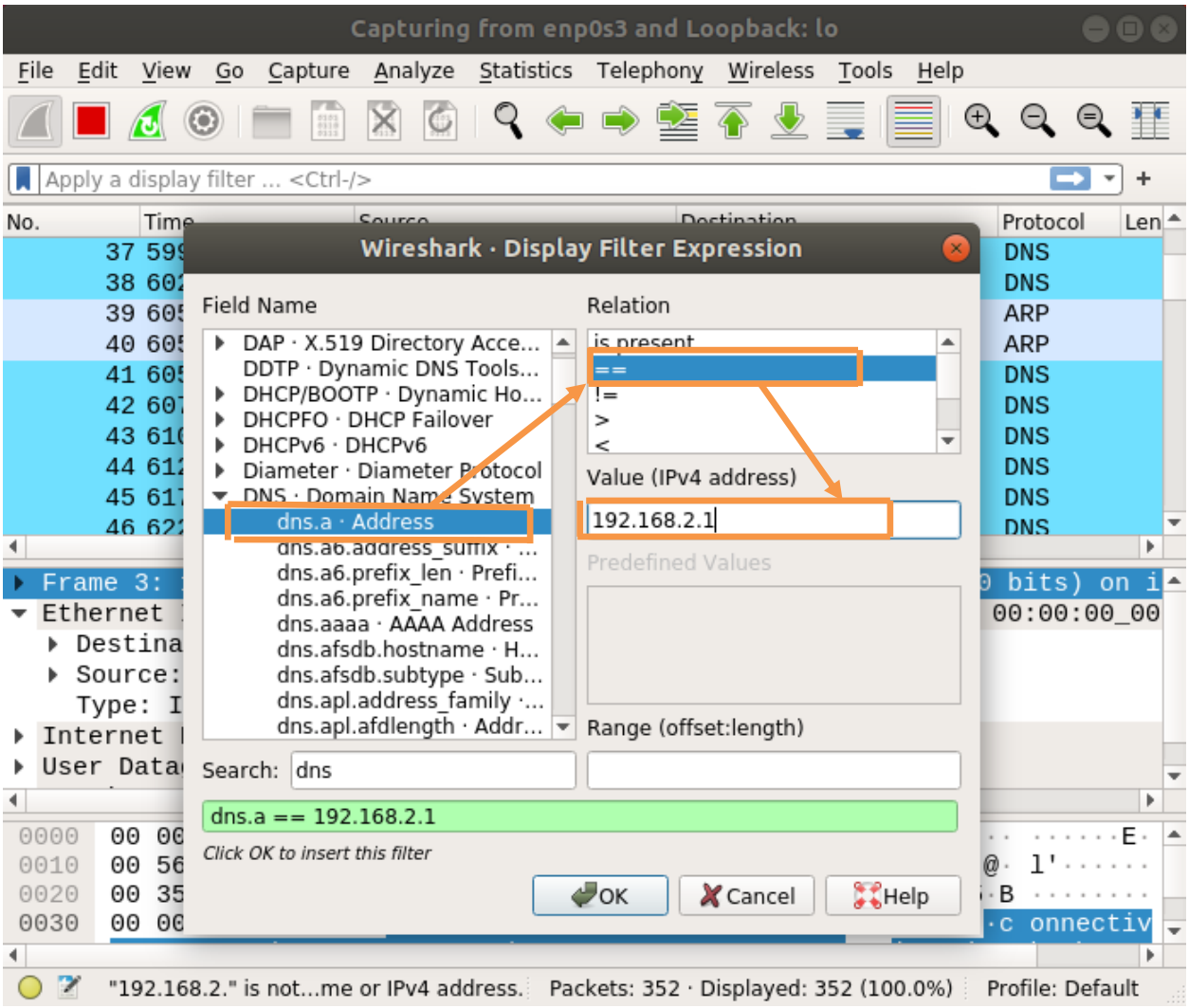
In the **Field Name** section almost all the networking protocols are listed. The list is huge. You can type in what protocol you're looking for in the **Search** textbox and the **Field Name** section would show the ones that matched.



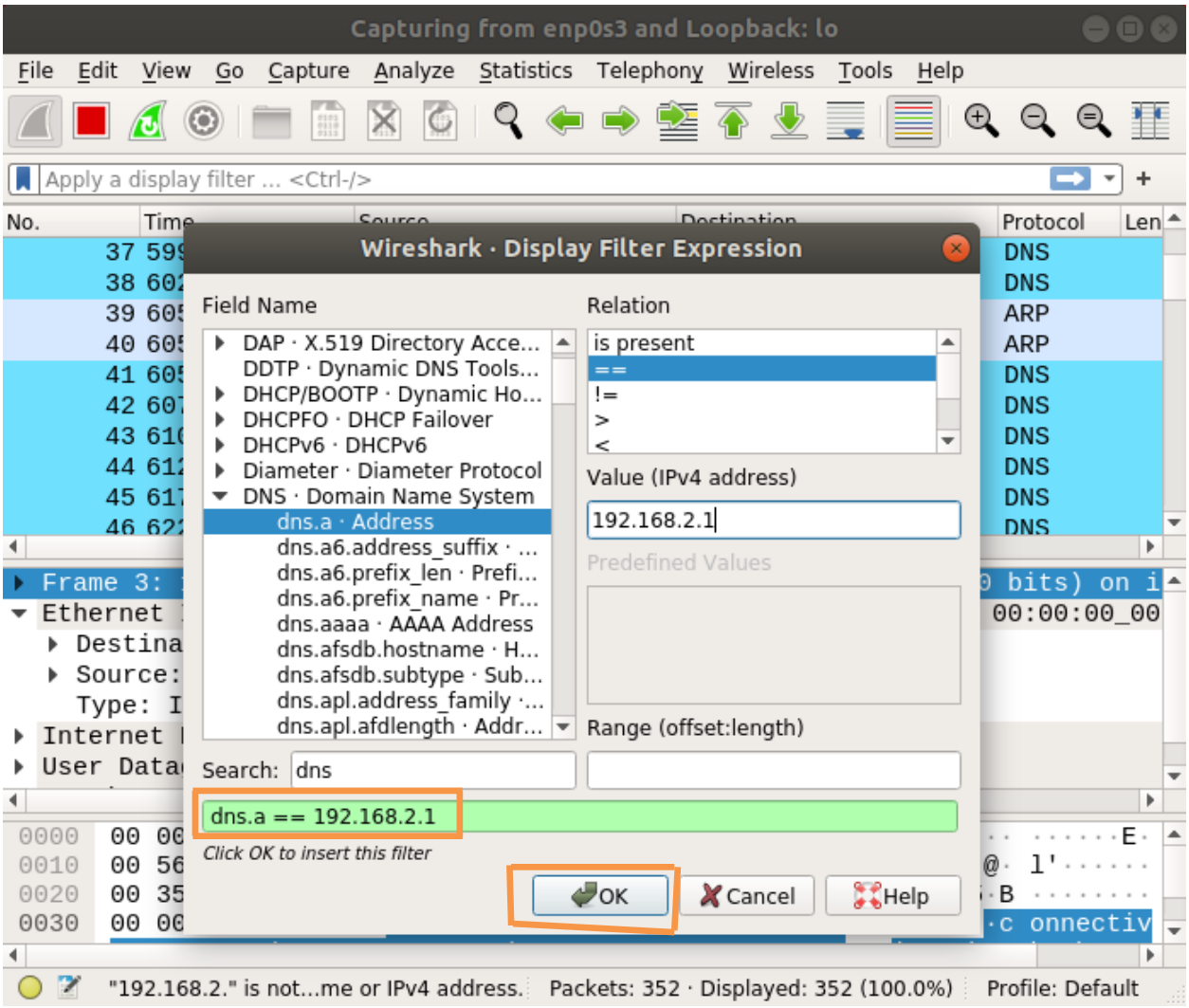
In this article, I am going to filter out all the DNS packets. So I selected **DNS Domain Name System** from the **Field Name** list. You can also click on the **arrow** on any protocol.



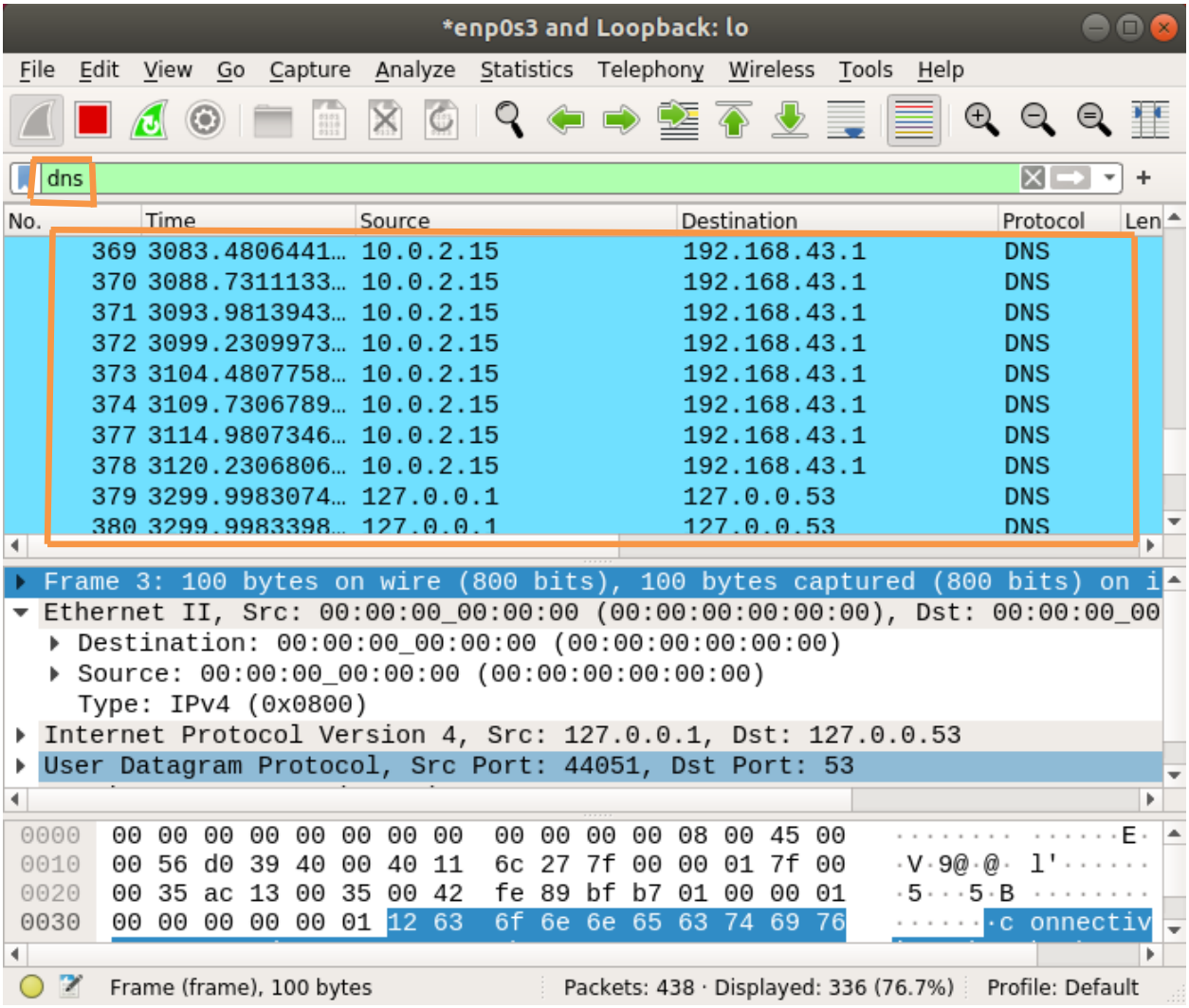
You can also use relational operators to test whether some field is equal to, not equal to, great than or less than some value. I searched for all the **DNS IPv4** address which is equal to **192.168.2.1** as you can see in the screenshot below.



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**

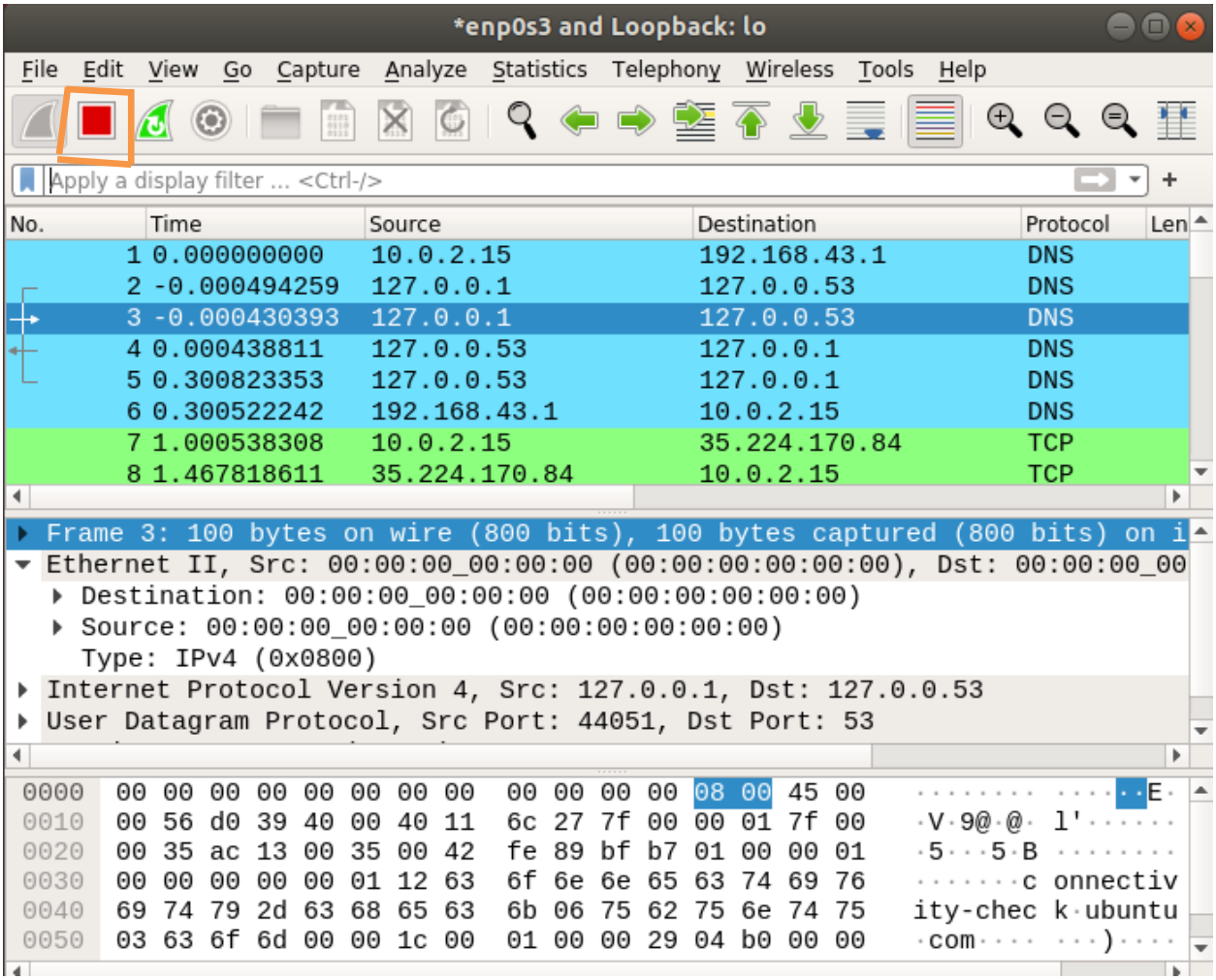


Now click on the marked icon to Apply the filter. As you can see, only the DNS protocol packets are shown.



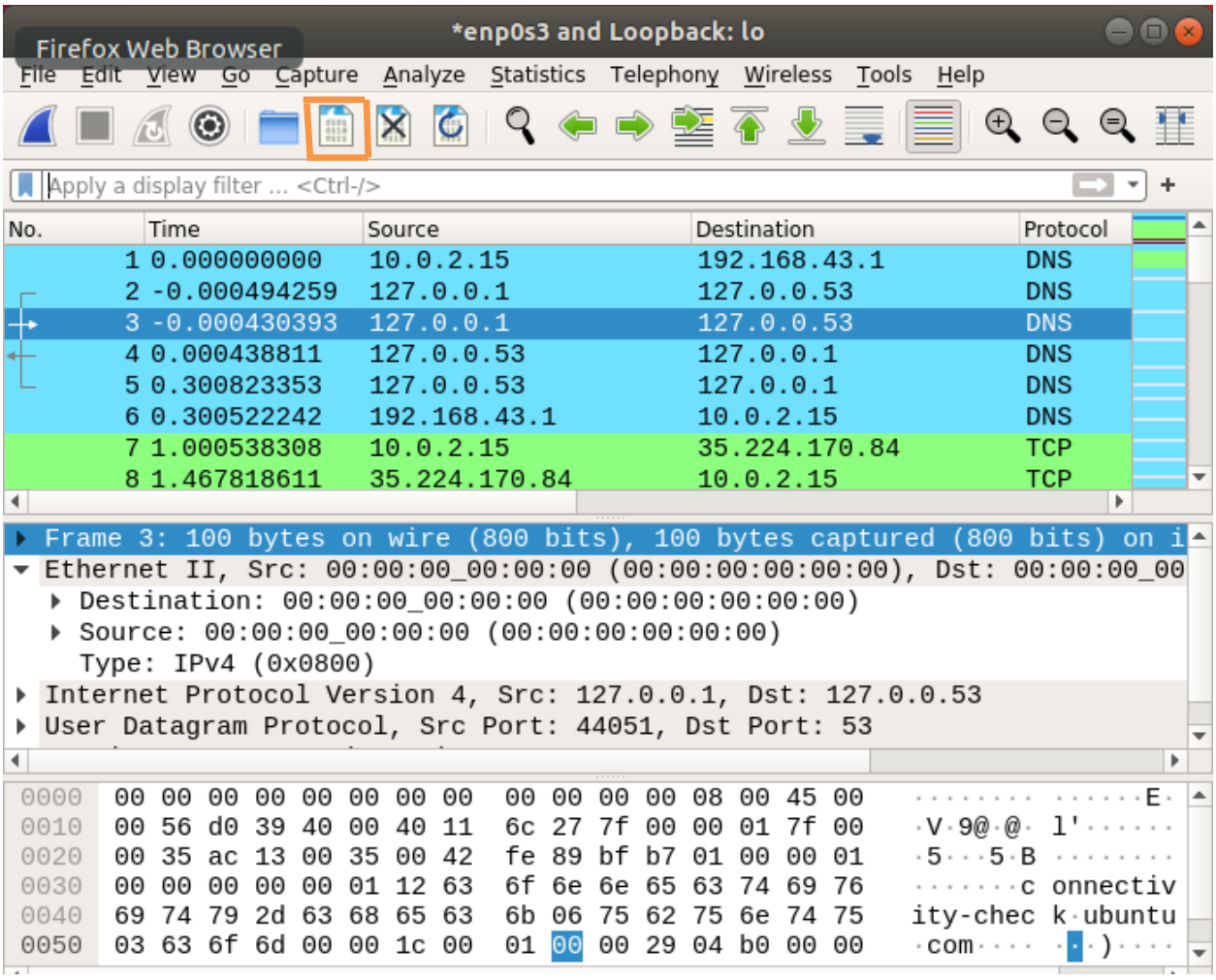
Stopping Packet Capture in Wireshark:

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

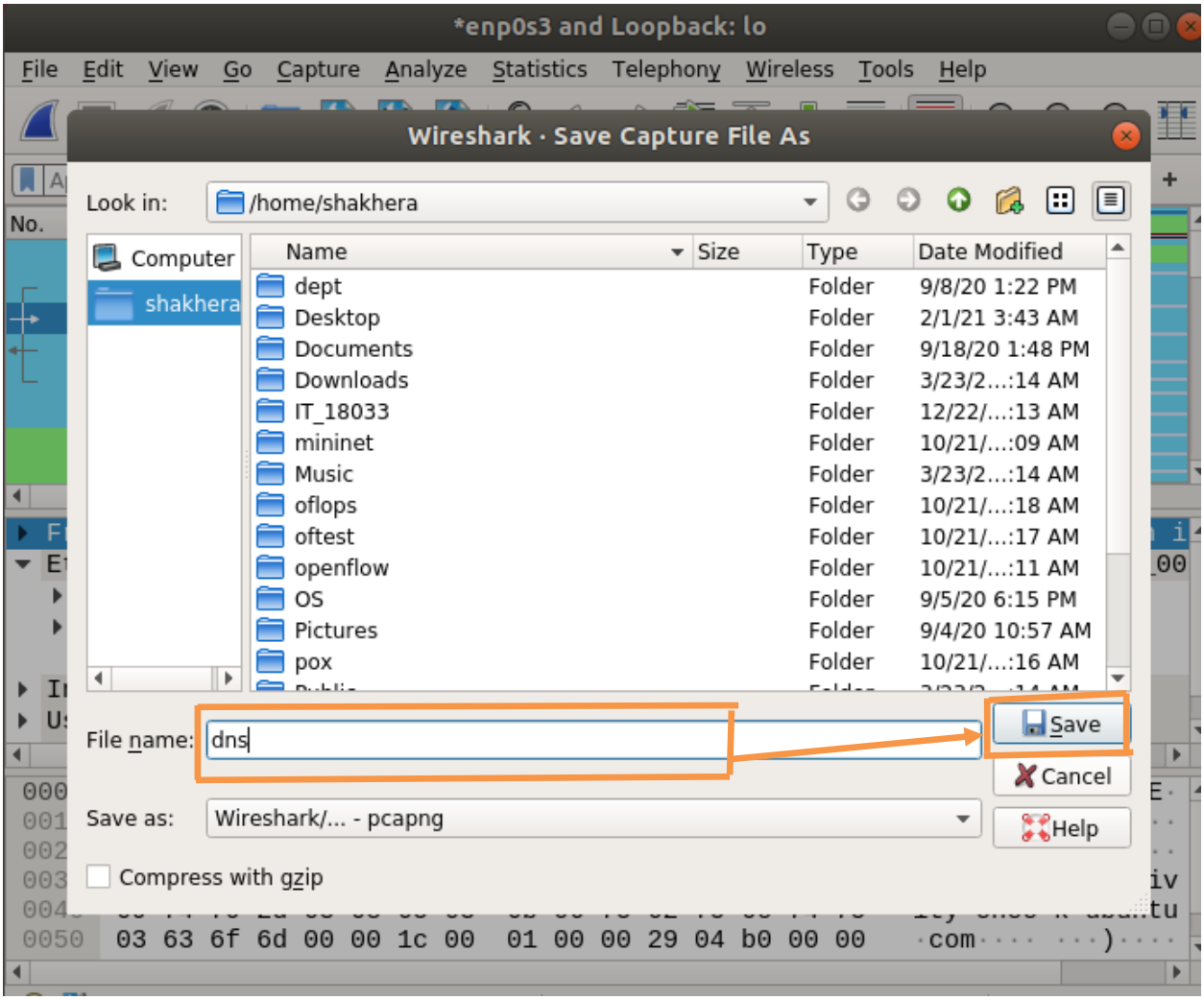


Saving Captured Packets to a File:

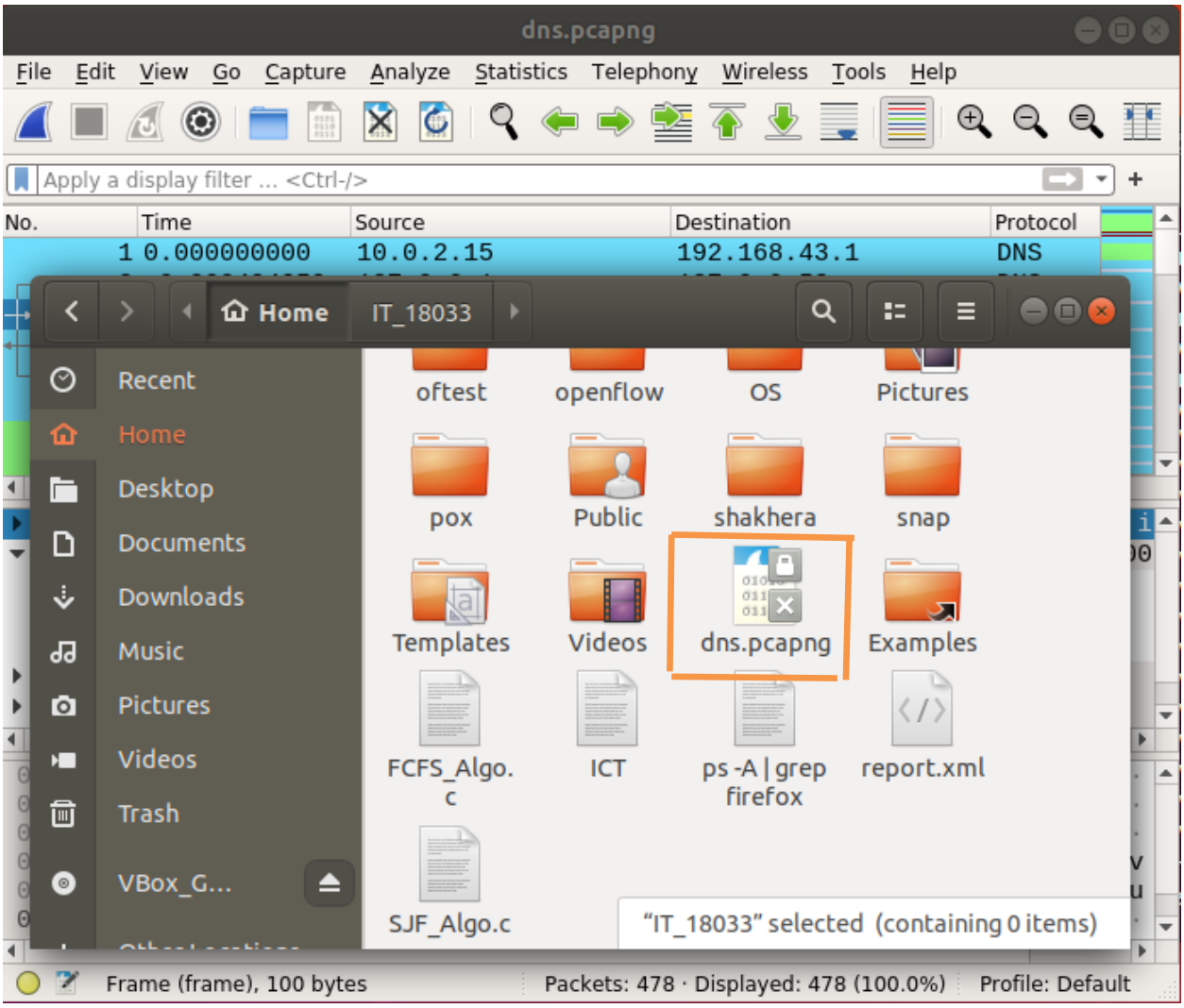
You can click on the marked icon to save captured packets to a file for future use.



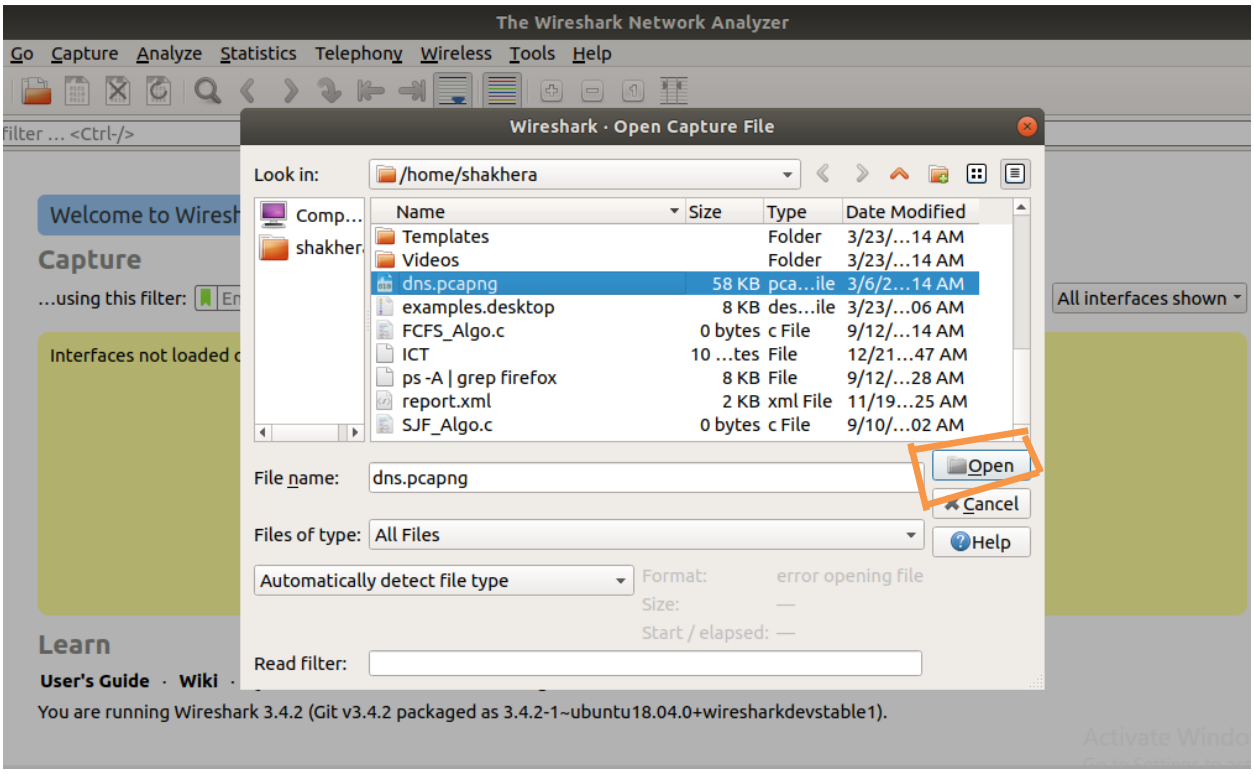
Now select a destination folder, type in the file name and click on **Save**



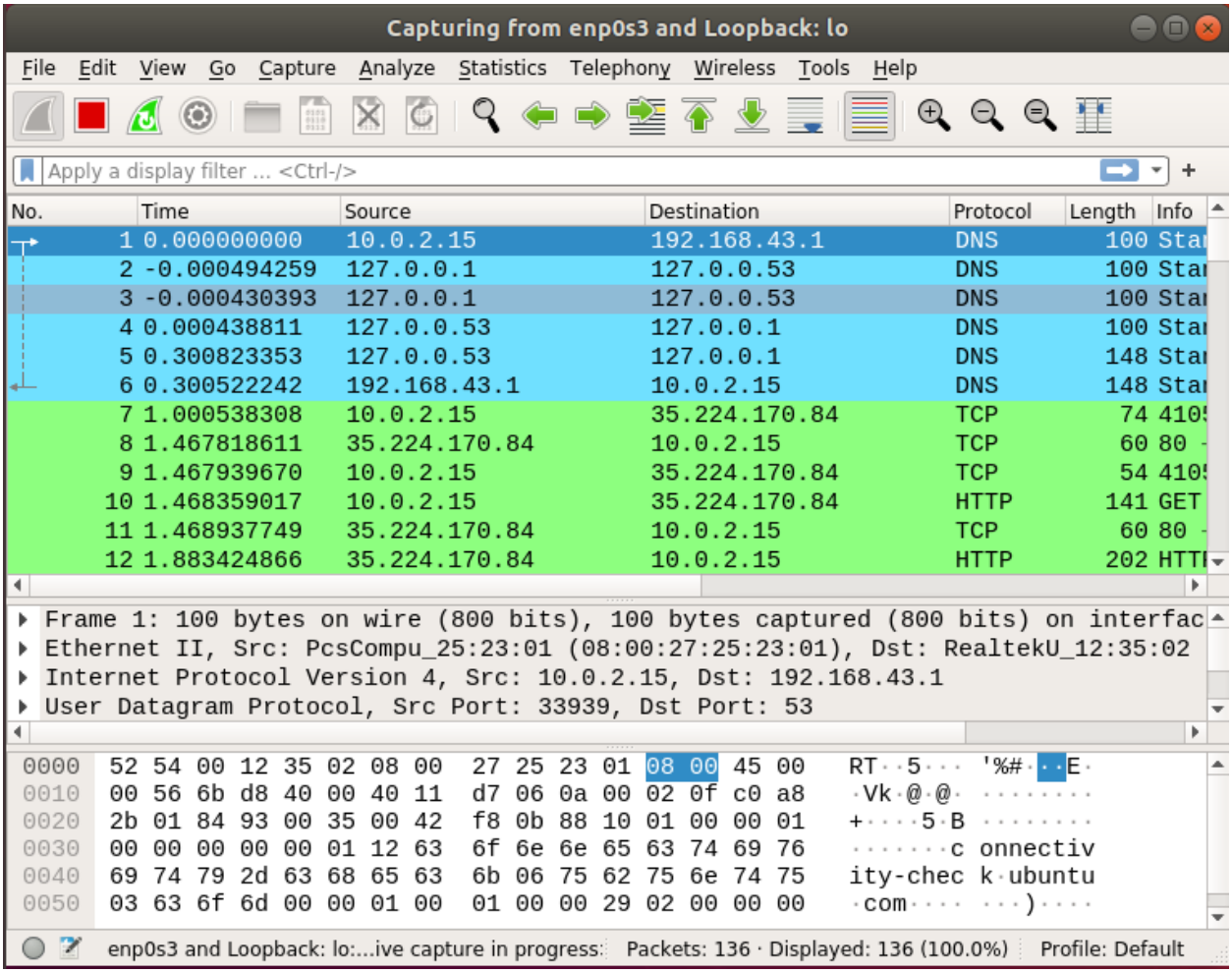
The file should be saved.



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



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.