Mawlana Bhashani Science and Technology University

Lab-Report

Report No : 08

Course code : ICT-3208

Course title : Installing Wireshark in Linux

Date of Performance : 05-03-21

Date of Submission: 05-03-21

Submitted by

Name: MD.Abdullah Al Mamun

ID:IT-18040

3th year 2ndsemester

Session: 2017-2018

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

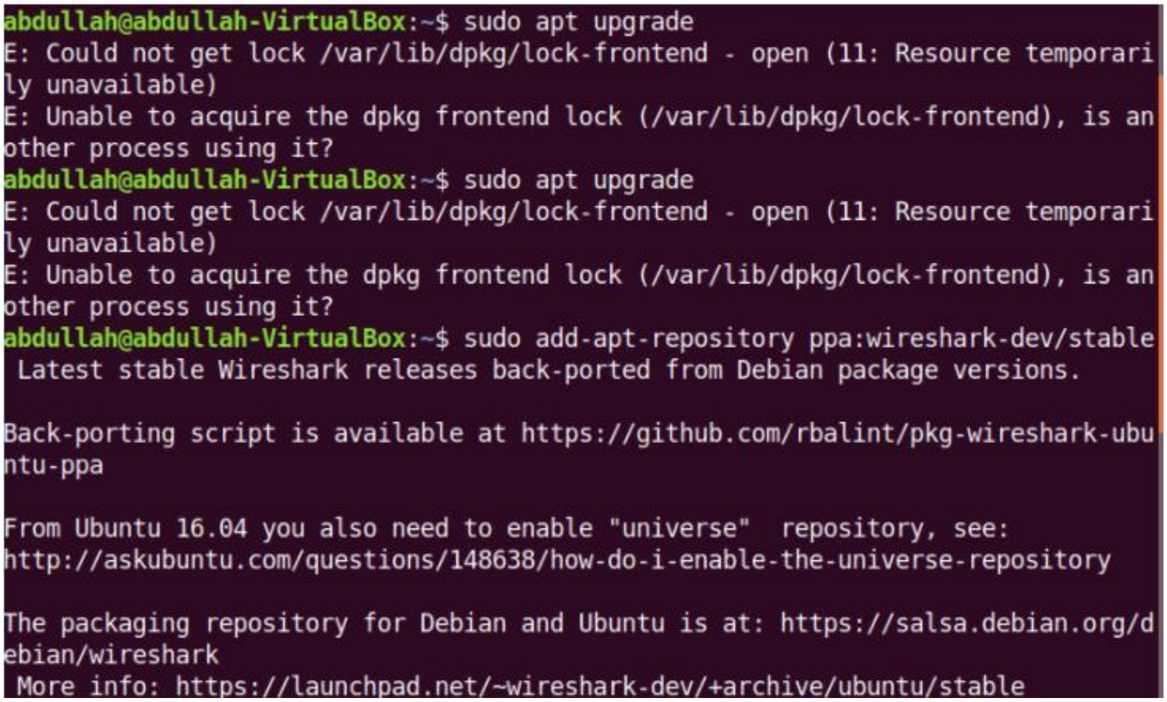
Dept. of ICT

MBSTU.

INSTALLING WIRESHARK:

Wireshark is a network packet analyzer. 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. How to install Wireshark is given below step by step: First update the APT package repository cache with the following command:

$ sudo apt update The APT package repository cache should be updated**.**



```
abdullah@abdullah-VirtualBox:~$ sudo apt upgrade
E: Could not get lock /var/lib/dpkg/lock-frontend - open (11: Resource temporari
ly unavailable)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), is an
other process using it?
abdullah@abdullah-VirtualBox:~$ sudo apt upgrade
E: Could not get lock /var/lib/dpkg/lock-frontend - open (11: Resource temporari
ly unavailable)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), is an
other process using it?
abdullah@abdullah-VirtualBox:~$ sudo add-apt-repository ppa:wireshark-dev/stable
 Latest stable Wireshark releases back-ported from Debian package versions.

Back-porting script is available at https://github.com/rbalint/pkg-wireshark-ubu
ntu-ppa

From Ubuntu 16.04 you also need to enable "universe"  repository, see:
http://askubuntu.com/questions/148638/how-do-i-enable-the-universe-repository

The packaging repository for Debian and Ubuntu is at: https://salsa.debian.org/d
ebian/wireshark
 More info: https://launchpad.net/~wireshark-dev/+archive/ubuntu/stable
```

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

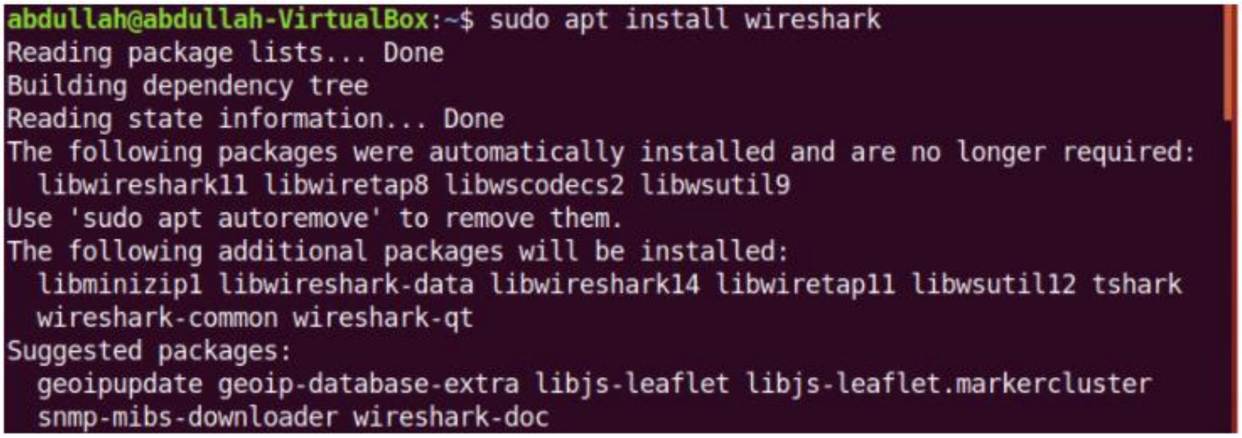
$ sudo apt get install wireshark

Wireshark should be installed.

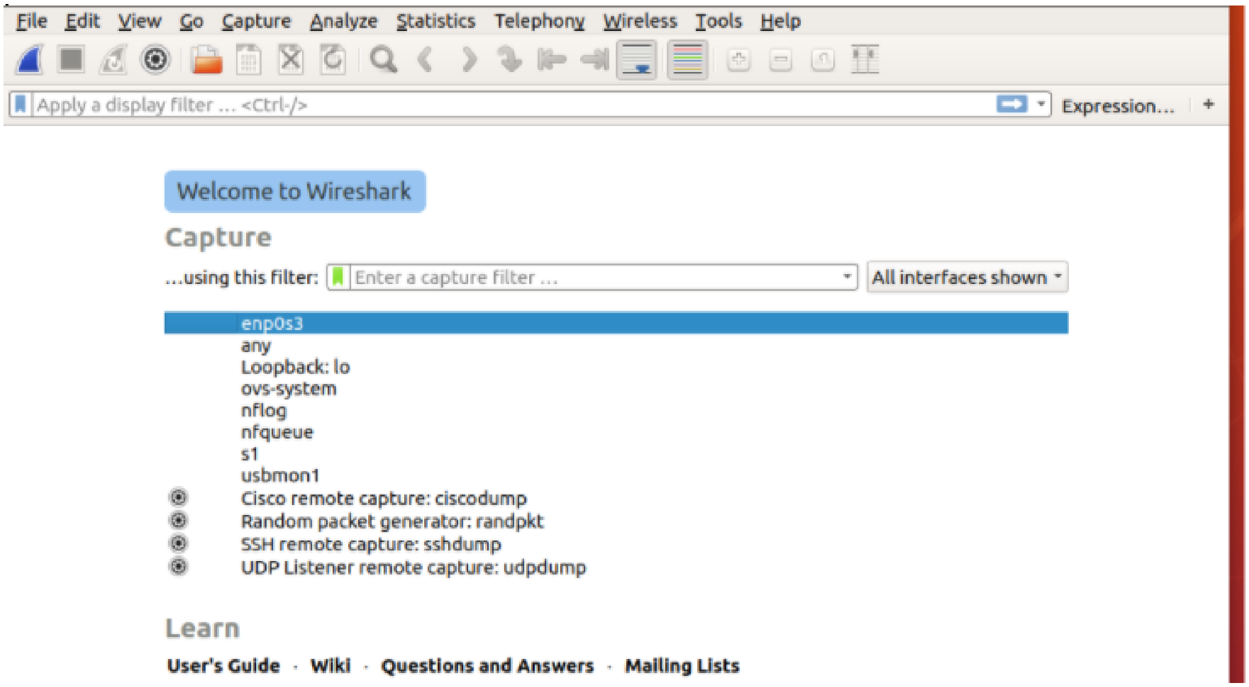
Run the following command to add your user to the Wireshark group:

$ sudo usermod -aG wireshark $(whoami) Now reboot your computer with the following command: $ sudo reboot Now run Wireshark using the following command:

$ sudo wireshark

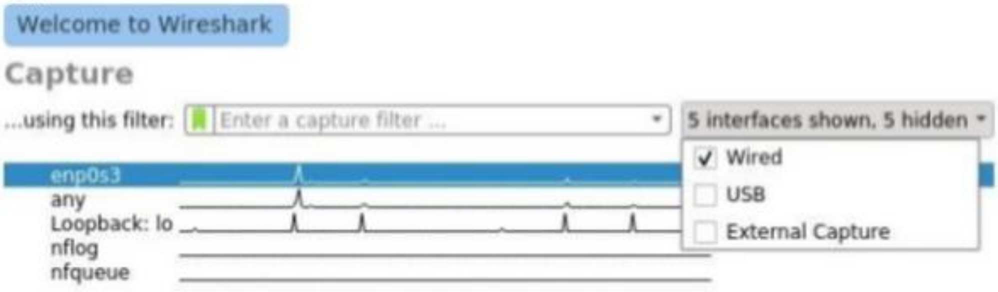


```
abdullah@abdullah-VirtualBox:~$ sudo apt install wireshark
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libwireshark11 libwiretap8 libwscodecs2 libwsutil9
Use 'sudo apt autoremove' 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
```


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Expression…
+
Welcome to Wireshark
Capture
…using this filter:
Enter a capture filter …
All interfaces shown
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

```
enp0s3
any
Loopback: lo
ovs-system
nflog
nfqueue
s1
usbmon1
Cisco remote capture: ciscodump
Random packet generator: randpkt
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
```

There are many types of interfaces you can monitor using Wireshark, for example, Wired, Wireless, 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.



Welcome to Wireshark

Capture

...using this filter: Enter a capture filter ...

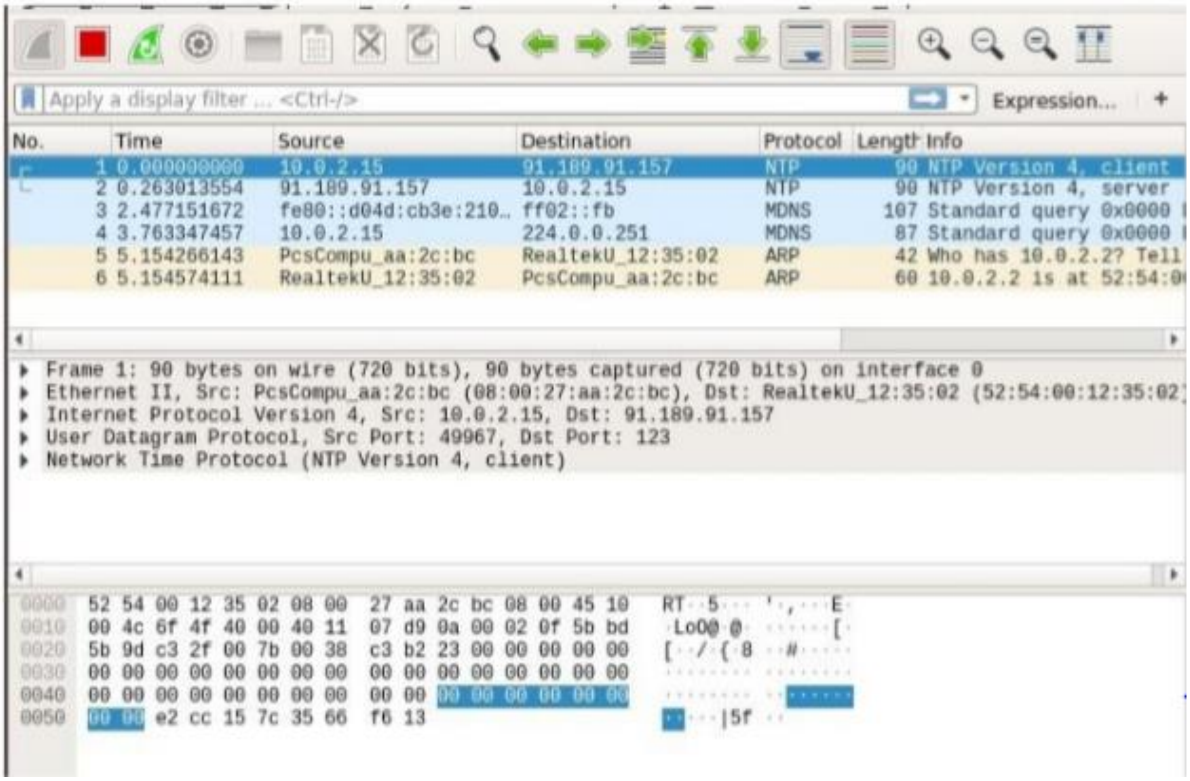
5 interfaces shown, 5 hidden

☑ Wired
□ USB
□ External Capture

Now to start capturing packets, just select the interface (in my case interface ens33) and click on the Start capturing packets icon as marked in the screenshot below.

You can also capture packets to and from multiple interfaces at the same time. Just press and hold <Ctrl> and click on the interfaces that you want to capture packets to and from and then click on the Start capturing packets icon as marked in the screenshot below.

I pinged google.com from the terminal and many packets were captured

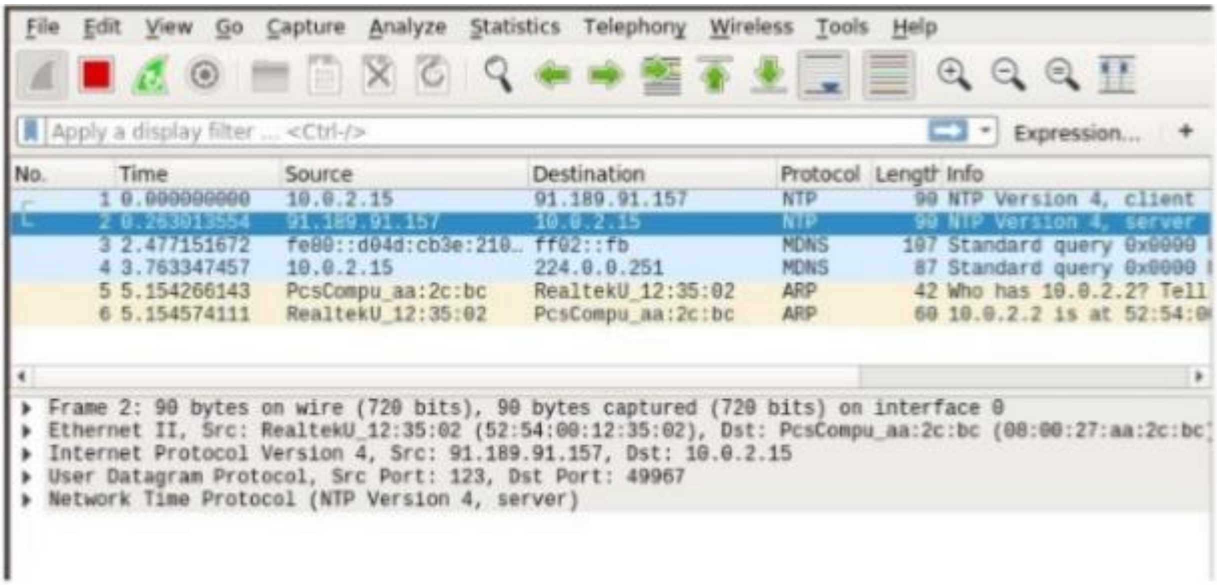


```
No.  Time         Source                   Destination          Protocol Length Info
  1 0.000000000  10.0.2.15                91.189.91.157        NTP        90 NTP Version 4, client
  2 0.263013554  91.189.91.157            10.0.2.15            NTP        90 NTP Version 4, server
  3 2.477151672  fe80::d04d:cb3e:210…     ff02::fb             MDNS      107 Standard query 0x0000
  4 3.763347457  10.0.2.15                224.0.0.251          MDNS       87 Standard query 0x0000
  5 5.154266143  PcsCompu_aa:2c:bc        RealtekU_12:35:02    ARP        42 Who has 10.0.2.2? Tell
  6 5.154574111  RealtekU_12:35:02        PcsCompu_aa:2c:bc    ARP        60 10.0.2.2 is at 52:54:0
```

```
▸ Frame 1: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▸ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 91.189.91.157
▸ User Datagram Protocol, Src Port: 49967, Dst Port: 123
▸ Network Time Protocol (NTP Version 4, client)
```

```
0000  52 54 00 12 35 02 08 00  27 aa 2c bc 08 00 45 10   RT··5··· '·,···E·
0010  00 4c 6f 4f 40 00 40 11  07 d9 0a 00 02 0f 5b bd   ·LoO@·@· ······[·
0020  5b 9d c3 2f 00 7b 00 38  c3 b2 23 00 00 00 00 00   [··/·{·8 ··#·····
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 e2 cc 15 7c 35 66  f6 13                     ·····|5f ··
```

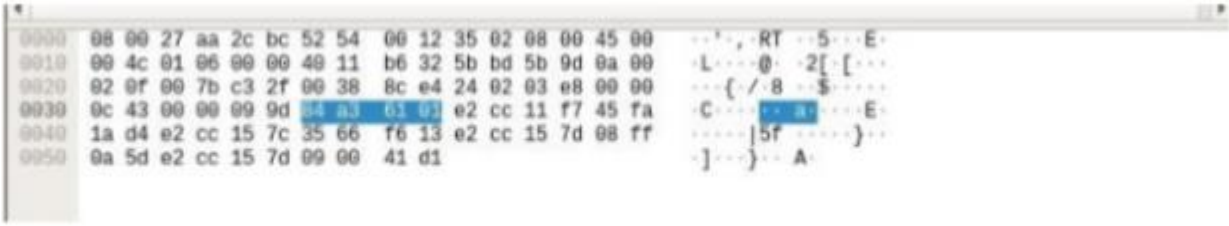
Now you can click on a packet to select it. Selecting a packet would show many information about that packet. As you can see, information about different layers of TCP/IP Protocol is listed.



```
No.   Time          Source                   Destination         Protocol Length Info
    1 0.000000000   10.0.2.15                91.189.91.157       NTP        90 NTP Version 4, client
    2 0.263013554   91.189.91.157            10.0.2.15           NTP        90 NTP Version 4, server
    3 2.477151672   fe80::d04d:cb3e:210…     ff02::fb            MDNS      107 Standard query 0x0000
    4 3.763347457   10.0.2.15                224.0.0.251         MDNS       87 Standard query 0x0000
    5 5.154266143   PcsCompu_aa:2c:bc        RealtekU_12:35:02   ARP        42 Who has 10.0.2.2? Tell
    6 5.154574111   RealtekU_12:35:02        PcsCompu_aa:2c:bc   ARP        60 10.0.2.2 is at 52:54:0
```

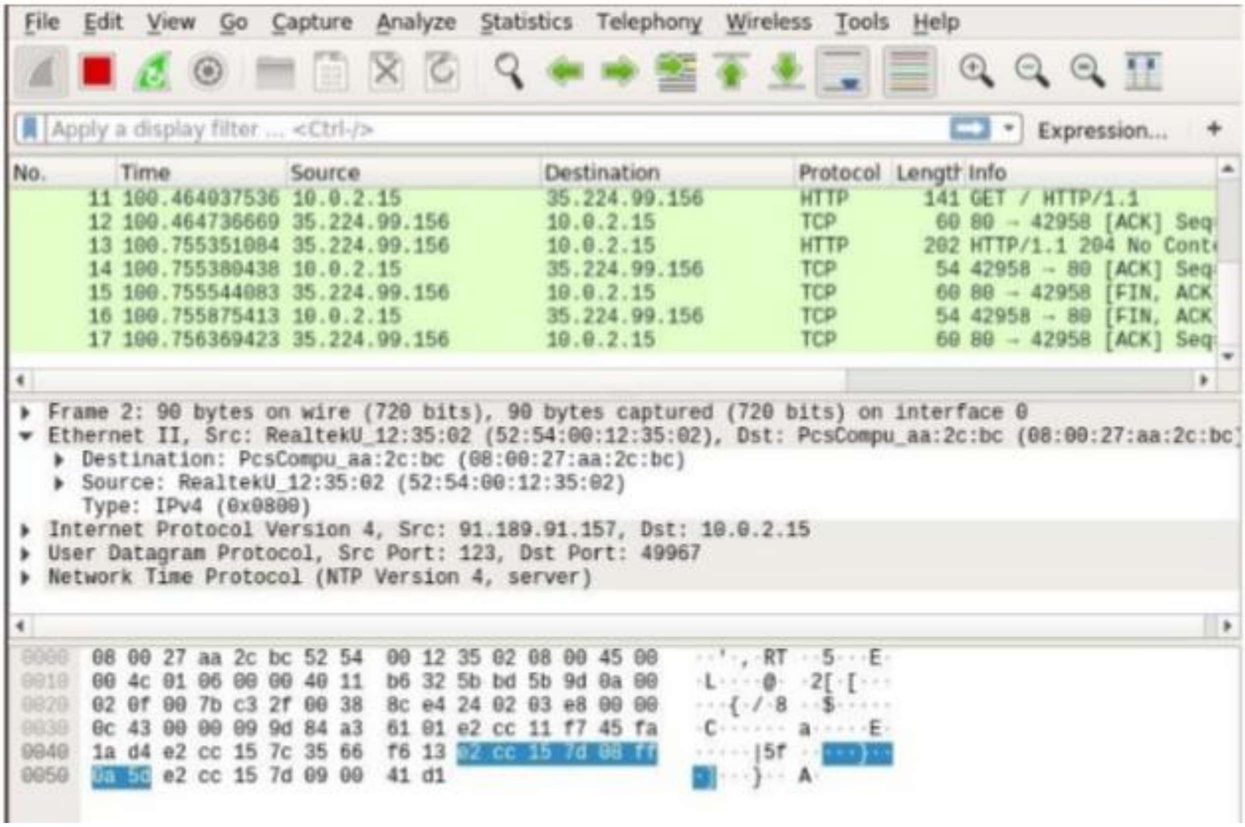
```
▸ Frame 2: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▸ Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc)
▸ Internet Protocol Version 4, Src: 91.189.91.157, Dst: 10.0.2.15
▸ User Datagram Protocol, Src Port: 123, Dst Port: 49967
▸ Network Time Protocol (NTP Version 4, server)
```

You can also see the RAW data of that particular packet



```
0000  08 00 27 aa 2c bc 52 54  00 12 35 02 08 00 45 00   ··'·,·RT ··5···E·
0010  00 4c 01 06 00 00 40 11  b6 32 5b bd 5b 9d 0a 00   ·L····@· ·2[·[···
0020  02 0f 00 7b c3 2f 00 38  8c e4 24 02 03 e8 00 00   ···{·/·8 ··$·····
0030  0c 43 00 00 09 9d 84 a3  61 01 e2 cc 11 f7 45 fa   ·C······ a·····E·
0040  1a d4 e2 cc 15 7c 35 66  f6 13 e2 cc 15 7d 08 ff   ·····|5f ·····}··
0050  0a 5d e2 cc 15 7d 09 00  41 d1                     ·]···}·· A·
```

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



```
No.  Time           Source         Destination    Protocol Length Info
  11 100.464037536  10.0.2.15      35.224.99.156  HTTP       141 GET / HTTP/1.1
  12 100.464736669  35.224.99.156  10.0.2.15      TCP         60 80 → 42958 [ACK] Seq
  13 100.755351084  35.224.99.156  10.0.2.15      HTTP       202 HTTP/1.1 204 No Cont
  14 100.755380438  10.0.2.15      35.224.99.156  TCP         54 42958 → 80 [ACK] Seq
  15 100.755544083  35.224.99.156  10.0.2.15      TCP         60 80 → 42958 [FIN, ACK
  16 100.755875413  10.0.2.15      35.224.99.156  TCP         54 42958 → 80 [FIN, ACK
  17 100.756369423  35.224.99.156  10.0.2.15      TCP         60 80 → 42958 [ACK] Seq
```

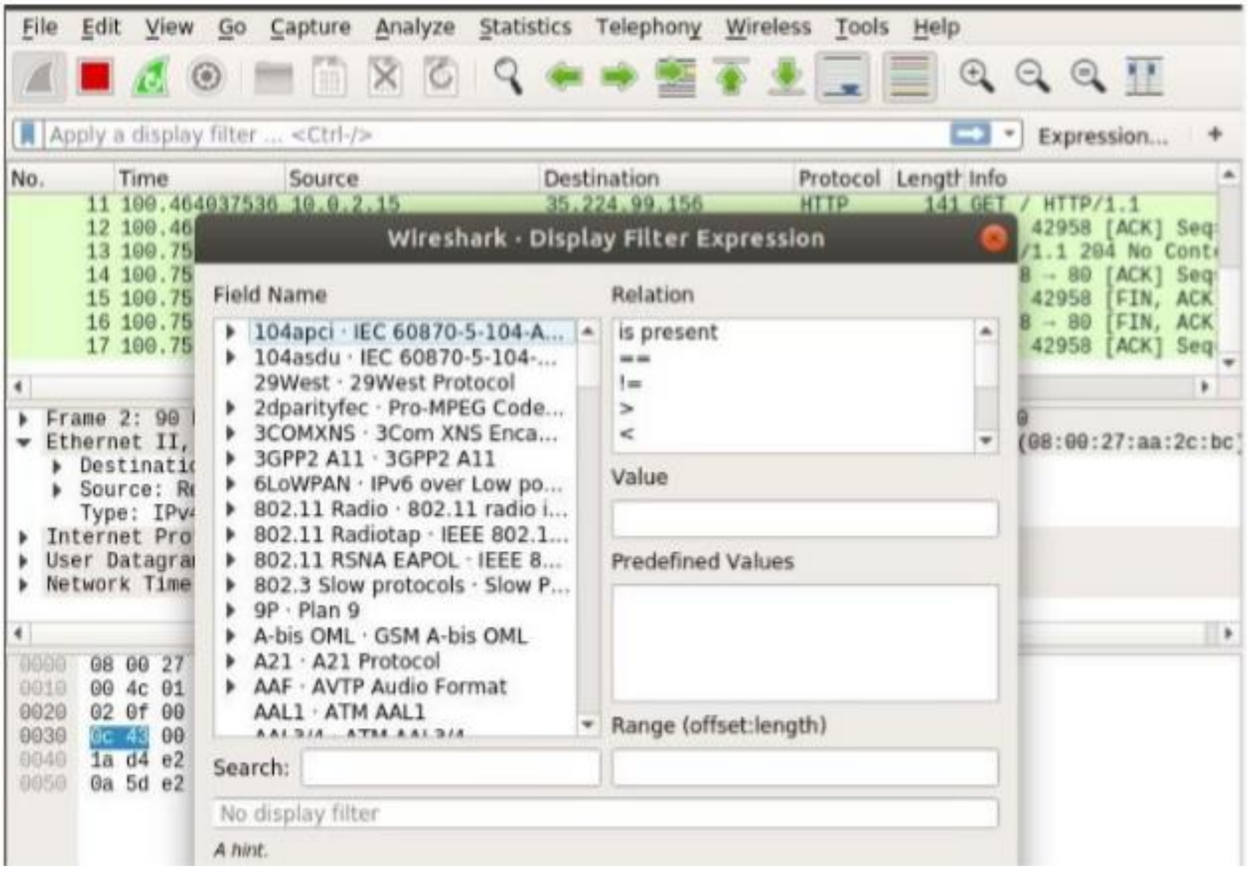
```
▸ Frame 2: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0
▾ Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc)
  ▸ Destination: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc)
  ▸ Source: RealtekU_12:35:02 (52:54:00:12:35:02)
    Type: IPv4 (0x0800)
▸ Internet Protocol Version 4, Src: 91.189.91.157, Dst: 10.0.2.15
▸ User Datagram Protocol, Src Port: 123, Dst Port: 49967
▸ Network Time Protocol (NTP Version 4, server)
```

```
0000  08 00 27 aa 2c bc 52 54  00 12 35 02 08 00 45 00   ··',·RT ··5···E·
0010  00 4c 01 06 00 00 40 11  b6 32 5b bd 5b 9d 0a 00   ·L····@· ·2[·[···
0020  02 0f 00 7b c3 2f 00 38  8c e4 24 02 03 e8 00 00   ···{·/·8 ··$·····
0030  0c 43 00 00 09 9d 84 a3  61 01 e2 cc 11 f7 45 fa   ·C······ a·····E·
0040  1a d4 e2 cc 15 7c 35 66  f6 13 e2 cc 15 7d 08 ff   ·····|5f ·····}··
0050  0a 5d e2 cc 15 7d 09 00  41 d1                     ·]···}·· A·
```

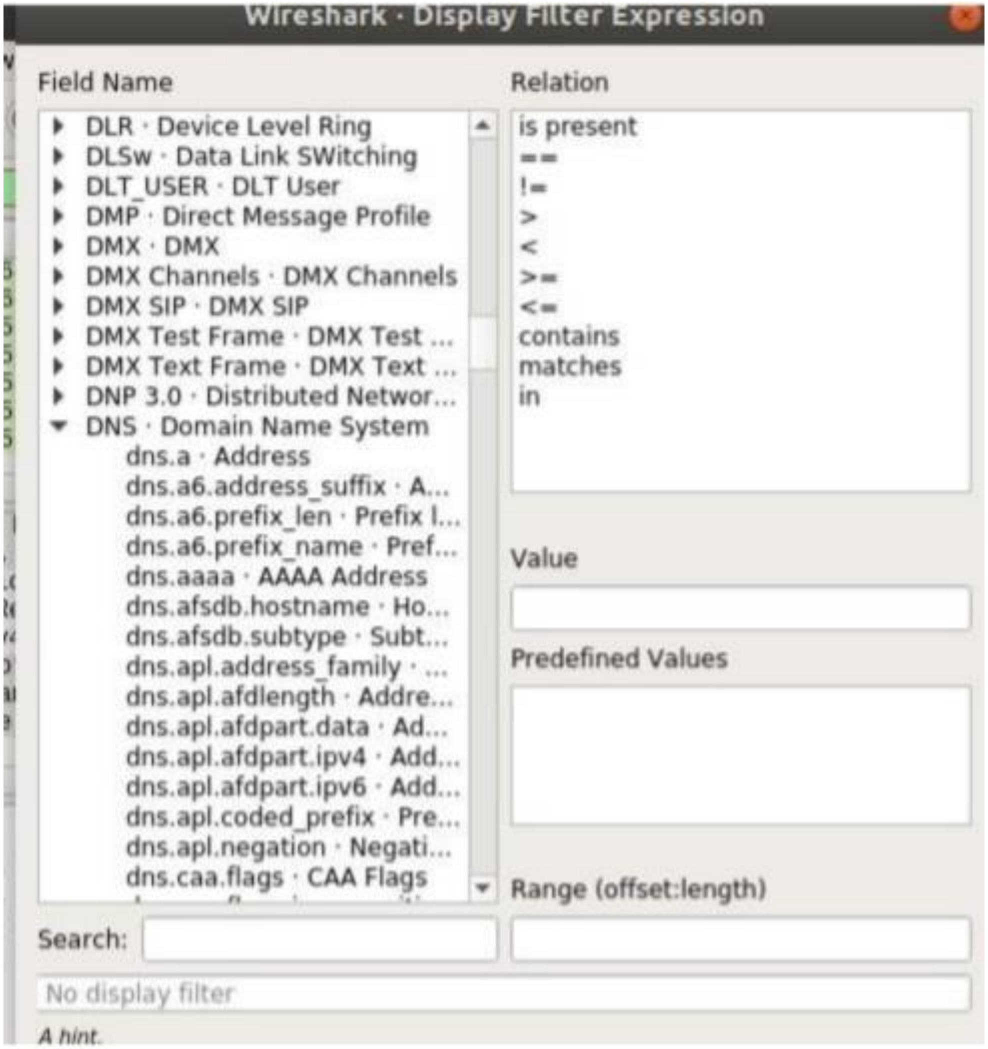
To filter packets, you can directly type in the filter expression in the textbox as marked in the screenshot below.

A new window should open as shown in the screenshot below. From here you can create filter expression to search packets very specifically.

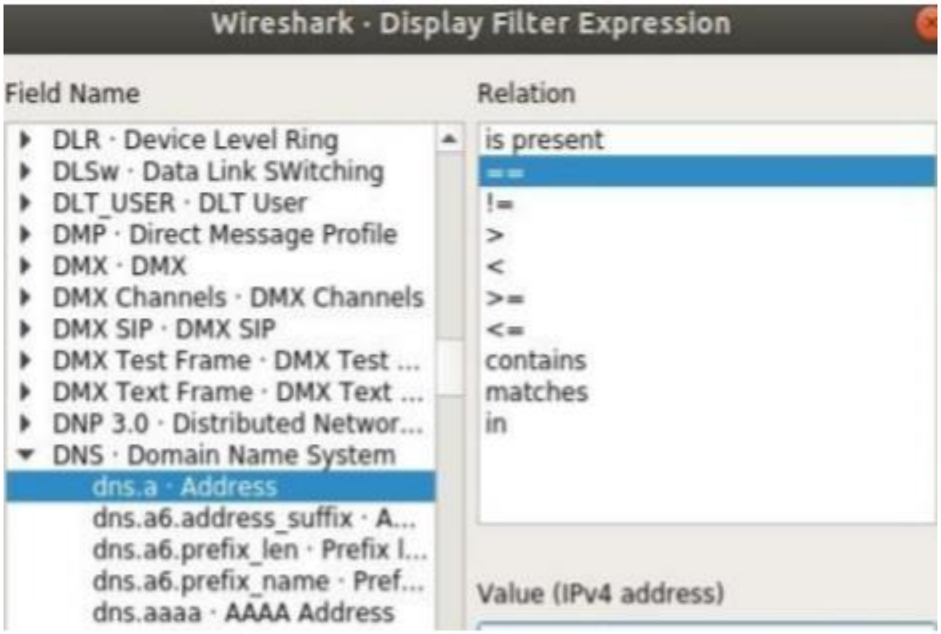
In the Field Name section almost all the networking protocols are listed. The list is huge. You can type in what protocol you're looking for in the Search textbox and the Field Name section would show the ones that matched.


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
No. Time Source Destination Protocol Length Info
11 100.464037536 10.0.2.15 35.224.99.156 HTTP 141 GET / HTTP/1.1
12 100.46 42958 [ACK] Seq
13 100.75 /1.1 204 No Cont
14 100.75 8 → 80 [ACK] Seq
15 100.75 42958 [FIN, ACK
16 100.75 8 → 80 [FIN, ACK
17 100.75 42958 [ACK] Seq
Frame 2: 90
Ethernet II,
Destinati
Source: R
Type: IPv
Internet Pro
User Datagra
Network Time
(08:00:27:aa:2c:bc
0000 08 00 27
0010 00 4c 01
0020 02 0f 00
0030 0c 43 00
0040 1a d4 e2
0050 0a 5d e2
Wireshark · Display Filter Expression
Field Name
104apci · IEC 60870-5-104-A...
104asdu · IEC 60870-5-104-...
29West · 29West Protocol
2dparityfec · Pro-MPEG Code...
3COMXNS · 3Com XNS Enca...
3GPP2 A11 · 3GPP2 A11
6LoWPAN · IPv6 over Low po...
802.11 Radio · 802.11 radio i...
802.11 Radiotap · IEEE 802.1...
802.11 RSNA EAPOL · IEEE 8...
802.3 Slow protocols · Slow P...
9P · Plan 9
A-bis OML · GSM A-bis OML
A21 · A21 Protocol
AAF · AVTP Audio Format
AAL1 · ATM AAL1
Relation
is present
==
!=
>
<
Value
Predefined Values
Range (offset:length)
Search:
No display filter
A hint.

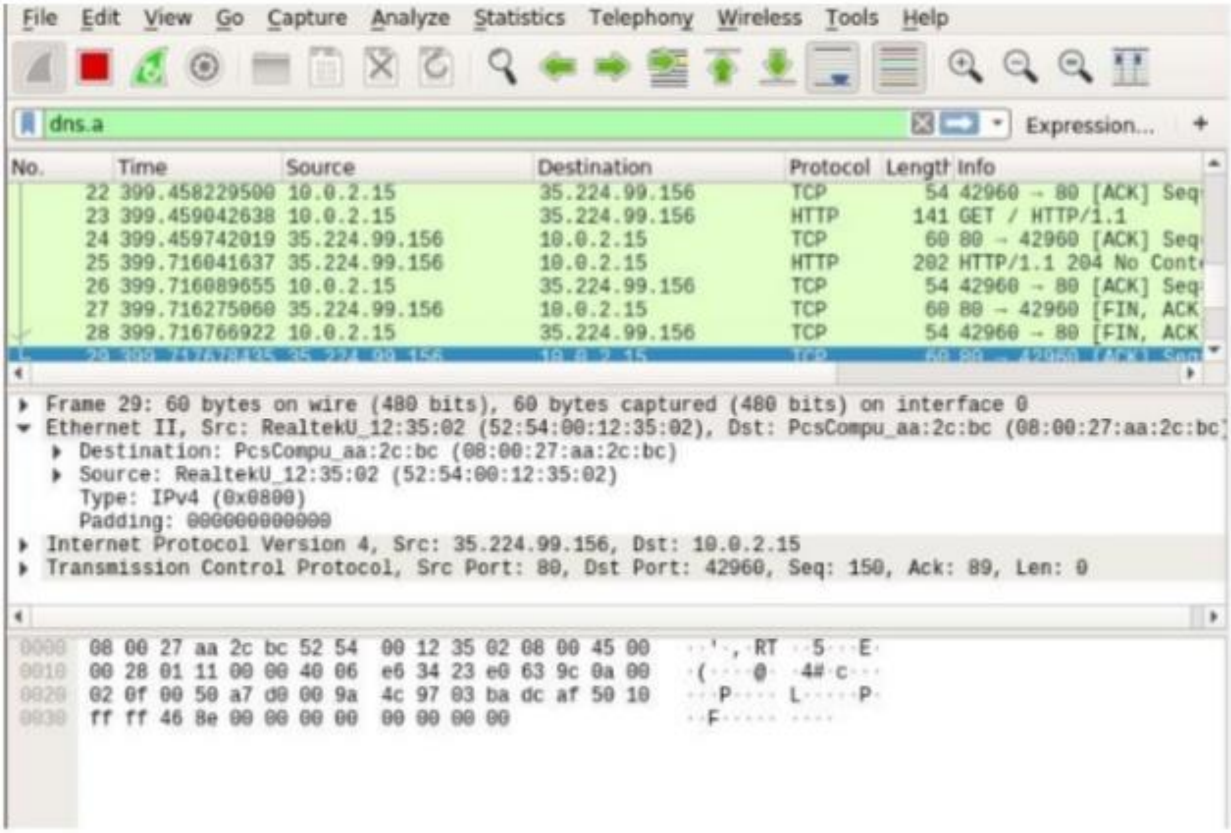
I am going to filter out all the DNS packets. So I selected DNS Domain Name System from the Field Name list. You can also click on the arrow on any protocol.


Wireshark · Display Filter Expression
Field Name
DLR · Device Level Ring
DLSw · Data Link SWitching
DLT_USER · DLT User
DMP · Direct Message Profile
DMX · DMX
DMX Channels · DMX Channels
DMX SIP · DMX SIP
DMX Test Frame · DMX Test ...
DMX Text Frame · DMX Text ...
DNP 3.0 · Distributed Networ...
DNS · Domain Name System
dns.a · Address
dns.a6.address_suffix · A...
dns.a6.prefix_len · Prefix l...
dns.a6.prefix_name · Pref...
dns.aaaa · AAAA Address
dns.afsdb.hostname · Ho...
dns.afsdb.subtype · Subt...
dns.apl.address_family · ...
dns.apl.afdlength · Addre...
dns.apl.afdpart.data · Ad...
dns.apl.afdpart.ipv4 · Add...
dns.apl.afdpart.ipv6 · Add...
dns.apl.coded_prefix · Pre...
dns.apl.negation · Negati...
dns.caa.flags · CAA Flags
Relation
is present
==
!=
>
<
>=
<=
contains
matches
in
Value
Predefined Values
Range (offset:length)
Search:
No display filter
A hint.

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.


Wireshark · Display Filter Expression
Field Name
DLR · Device Level Ring
DLSw · Data Link SWitching
DLT_USER · DLT User
DMP · Direct Message Profile
DMX · DMX
DMX Channels · DMX Channels
DMX SIP · DMX SIP
DMX Test Frame · DMX Test ...
DMX Text Frame · DMX Text ...
DNP 3.0 · Distributed Networ...
DNS · Domain Name System
dns.a · Address
dns.a6.address_suffix · A...
dns.a6.prefix_len · Prefix l...
dns.a6.prefix_name · Pref...
dns.aaaa · AAAA Address
Relation
is present
==
!=
>
<
>=
<=
contains
matches
in
Value (IPv4 address)

As you can see, only the DNS protocol packets are shown



```
No.   Time            Source          Destination     Protocol Length Info
   22 399.458229500 10.0.2.15       35.224.99.156   TCP        54 42960 → 80 [ACK] Seq
   23 399.459042638 10.0.2.15       35.224.99.156   HTTP      141 GET / HTTP/1.1
   24 399.459742019 35.224.99.156   10.0.2.15       TCP        60 80 → 42960 [ACK] Seq
   25 399.716041637 35.224.99.156   10.0.2.15       HTTP      202 HTTP/1.1 204 No Conte
   26 399.716089655 10.0.2.15       35.224.99.156   TCP        54 42960 → 80 [ACK] Seq
   27 399.716275060 35.224.99.156   10.0.2.15       TCP        60 80 → 42960 [FIN, ACK
   28 399.716766922 10.0.2.15       35.224.99.156   TCP        54 42960 → 80 [FIN, ACK
   29 399.717678435 35.224.99.156   10.0.2.15       TCP        60 80 → 42960 [ACK] Seq
```

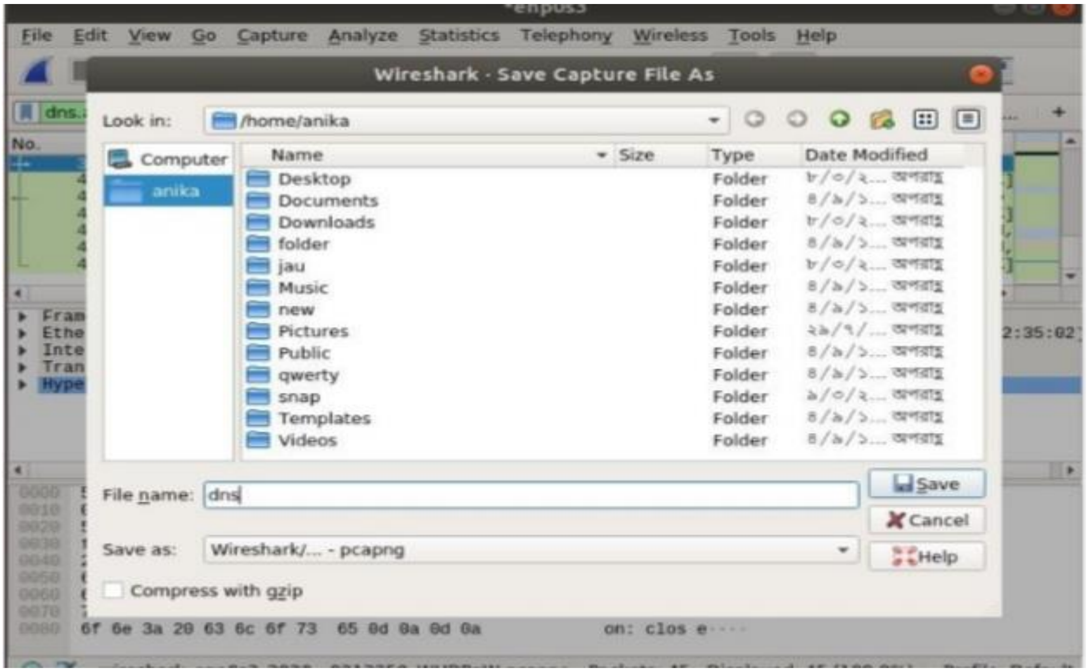
```
▸ Frame 29: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
▾ Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc)
  ▸ Destination: PcsCompu_aa:2c:bc (08:00:27:aa:2c:bc)
  ▸ Source: RealtekU_12:35:02 (52:54:00:12:35:02)
    Type: IPv4 (0x0800)
    Padding: 000000000000
▸ Internet Protocol Version 4, Src: 35.224.99.156, Dst: 10.0.2.15
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 42960, Seq: 150, Ack: 89, Len: 0
```

```
0000  08 00 27 aa 2c bc 52 54  00 12 35 02 08 00 45 00   ··'·,·RT ··5···E·
0010  00 28 01 11 00 00 40 06  e6 34 23 e0 63 9c 0a 00   ·(····@· ·4#·c···
0020  02 0f 00 50 a7 d0 00 9a  4c 97 03 ba dc af 50 10   ···P···· L·····P·
0030  ff ff 46 8e 00 00 00 00  00 00 00 00               ··F····· ····
```

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

You can click on the saved 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



Look in: /home/anika

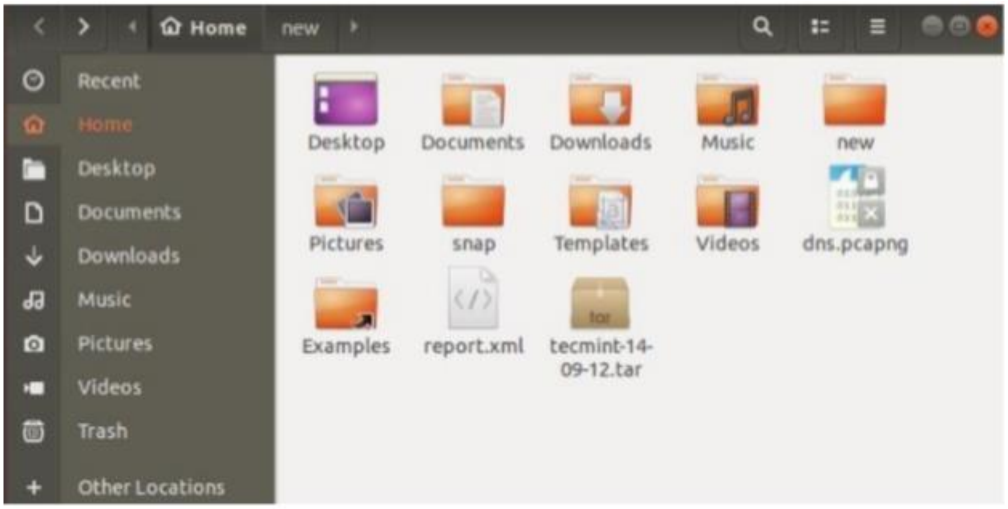
Name	Size	Type	Date Modified
Desktop		Folder	৮/৩/২... অপরাহ্ন
Documents		Folder	৪/৯/১... অপরাহ্ন
Downloads		Folder	৮/৩/২... অপরাহ্ন
folder		Folder	৪/৯/১... অপরাহ্ন
jau		Folder	৮/৩/২... অপরাহ্ন
Music		Folder	৪/৯/১... অপরাহ্ন
new		Folder	৪/৯/১... অপরাহ্ন
Pictures		Folder	২৯/৭/... অপরাহ্ন
Public		Folder	৪/৯/১... অপরাহ্ন
qwerty		Folder	৪/৯/১... অপরাহ্ন
snap		Folder	৯/৩/২... অপরাহ্ন
Templates		Folder	৪/৯/১... অপরাহ্ন
Videos		Folder	৪/৯/১... অপরাহ্ন

File name: dns

Save as: Wireshark/... - pcapng

☐ Compress with gzip

The file should be saved.



That’s how you install and use Wireshark in Linux

Conclusion: WireShark - a network protocol analyzer For Windows and *nix systems. And although I never used it on anything but my Windows PC, it's a really interesting tool to have installed - both for developers and curious minds. So what are

some uses that might potentially benefit the people who decided to install it? Personally I have three main reasons, and i am describing those below.

Capture interesting stuff

For example Windows Phone applications that come as XAP packages. Some time to set up an environment and you will be ready to intercept incoming content. Also, I found out some interesting stuff about an undocumented Zune API - also through inspecting existing transfer logs. It's really cool to see how a lot of content that is used on various web sites and application is in fact transmitted through open channels without any authentication necessary (even if that is present in the application itself). The fun fact is that you can use those channels for your own benefit (e.g. build third party clients for specific services)