Networking With Linux Lab

Subject Code: MCAL26

A Project Report Submitted in
Fulfilment of the Degree of

MASTER

In

COMPUTER APPLICATION

Year 2022-2023

By

Mr. Ravishankar Jaiswal

(Application Id-172047)

Semester-II

Under the Guidance of

Prof. Dr. Kavita Chouk

Institute of Distance and Open Learning
Vidya Nagari, Kalina, Santacruz East – 400098.

University of Mumbai

PCP Center

[Satish Pradhan Dyanasadhana College, Thane]

Institute of Distance and Open Learning

Vidya Nagari, Kalina, Santacruz East – 400098.

CERTIFICATE

This to certify that, **" Ravishankar Jaiswal"** appearing **Master's in computer application (Semester II) Application ID: 172047** satisfactory completed the prescribed practical of **MCAL26 -Networking With Linux Lab** as laid down by the University of Mumbai for the academic year 2022-23.

Teacher In Charge	External Examiner	Coordinator – M.C.A

Date:

Place: -

INDEX

Exercise	Topic	Date	Signature
1	Installation of NS3 (Network Simulator		
2	Installation of NetAnim		
3	Installation of Wireshark		
4	Program to simulate UDP server client		
5	Program to simulate FTP using TCP protocol		
6	Analyze the network traffic using WireShark		
7	Analyze the performance parameters of network using Wire Shark		

Practical No-1

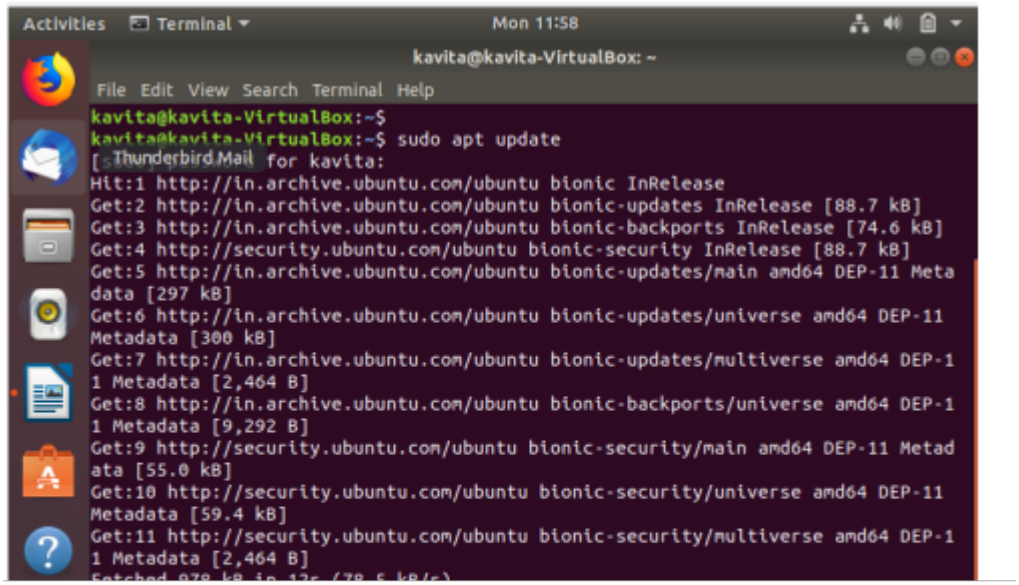
Installation of NS3 (Network Simulator):

Network Simulation is the software which tells the behaviour of computer networks. It is common useful method to calculate different network topologies exclusive of real-world implementation.

For NS3 installation on Ubuntu 18 version.

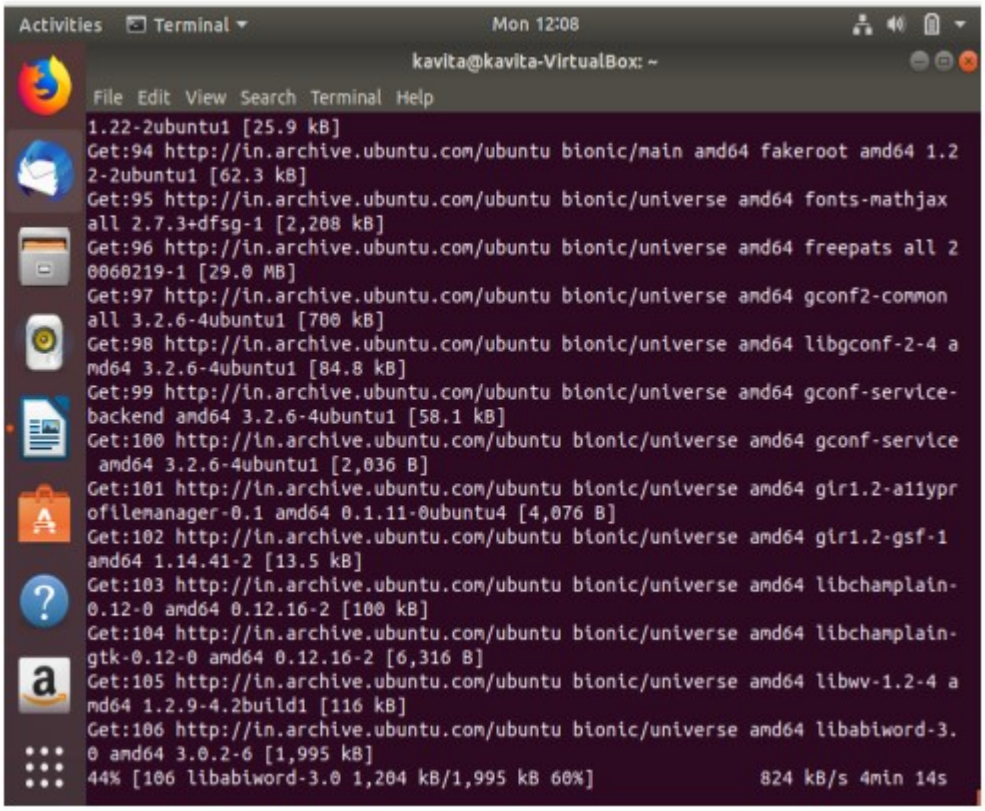
We have to execute following commands

Step 1: sudo apt update



Step 2:sudo apt install g++ python3 python3-dev pkg-config sqlite3 cmake python3-setuptools git qtbase5-dev qtchooser qt5-qmake qtbase5-dev-tools gir1.2-goocanvas-2.0 python3-gi python3-gi-cairo python3-pygraphviz gir1.2-gtk-3.0 ipython3 openmpi-bin openmpi-common openmpi-doc libopenmpi-dev autoconf cvs bzr unrar gsl-bin libgsl-dev libgslcblas0 wireshark tcpdump sqlite sqlite3 libsqlite3-dev libxml2 libxml2-dev libc6-dev libc6-dev-i386 libclang-dev llvm-dev automake python3-pip libxml2 libxml2-dev libboost-all-dev

Step 3: Download ns-allinone-3.36.1.tar.bz2 from the website nsnam.org. https://www.nsnam.org/releases/ns-allinone-3.36.1.tar.bz2



Step 4: Unzip the above file content to the home folder (in my case, its /home) - Check your home folder and do it accordingly.

To unzip use the GUI with Right click and extract and select the /home folder.

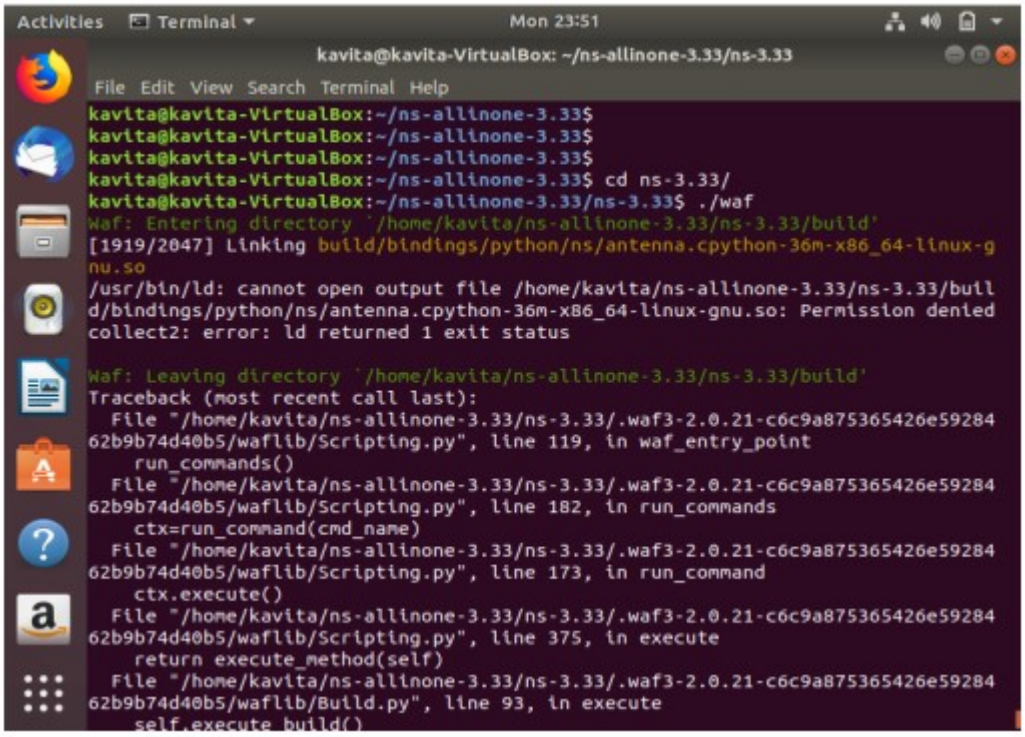
Else you can use the command

$ tar jxvf ns-allinone-3.36.1.tar.bz2

Step 5: Step 4: Go to the folder

$ cd ns-allinone-3.36.1/

$ ./build.py --enable-examples --enable-tests



Step 6: Once the installation is done. You can run the example as shown

$ cd ns-3.36.1/

$ ./ns3 run hello-simulator
Hello Simulator

Practical No-2

Installation of NetAnim

NetAnim is the network Animator that comes with NS3.During compilation of process of NS3 NetAnim may not be compiled .so we need to compile NetAnim. It is an offline network animator tool that now comes with NS3 that nsallinone3. All versions. By using NetAnim we can animate NS3 simulator, using the xml file trace the output in the simulation. NetAnim is the software which execute xml file to generate graphical output on NS3 simulator

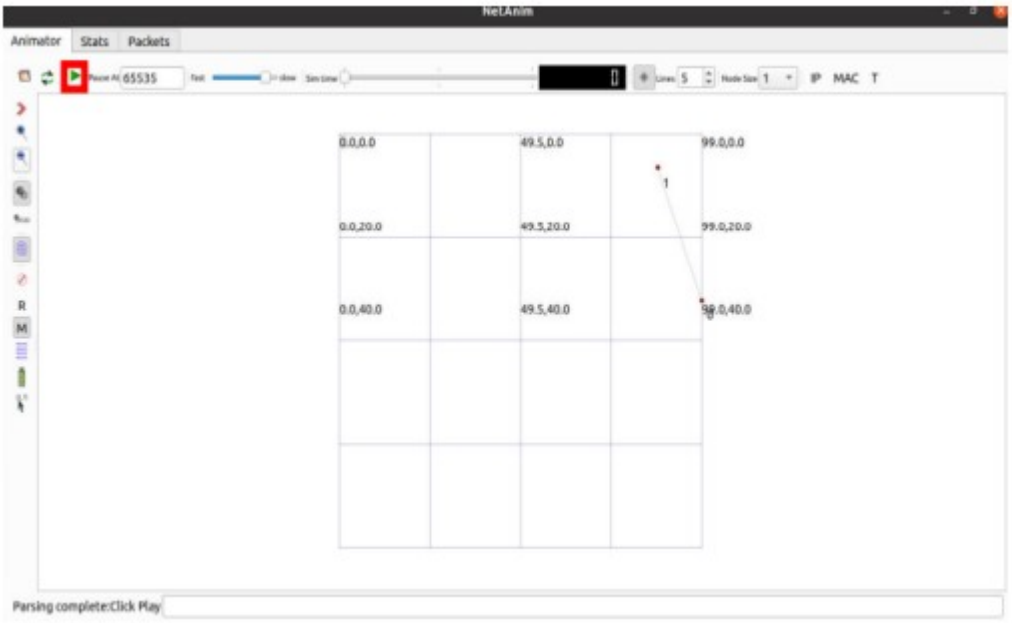
Method1:

NetAnim can be downloaded from the
URL http://code.nsnam.org/jabraham3/netanim-3.104

Method 2:
Step 1:
sudo apt-get install NetAnim
The above command is case sensitive (NetAnim)
Step 2:
To run once it's installed
NetAnim file.xml
If you already has a file.xml.

Step 3: Select xml file

Step4: Run the simulation by clicking

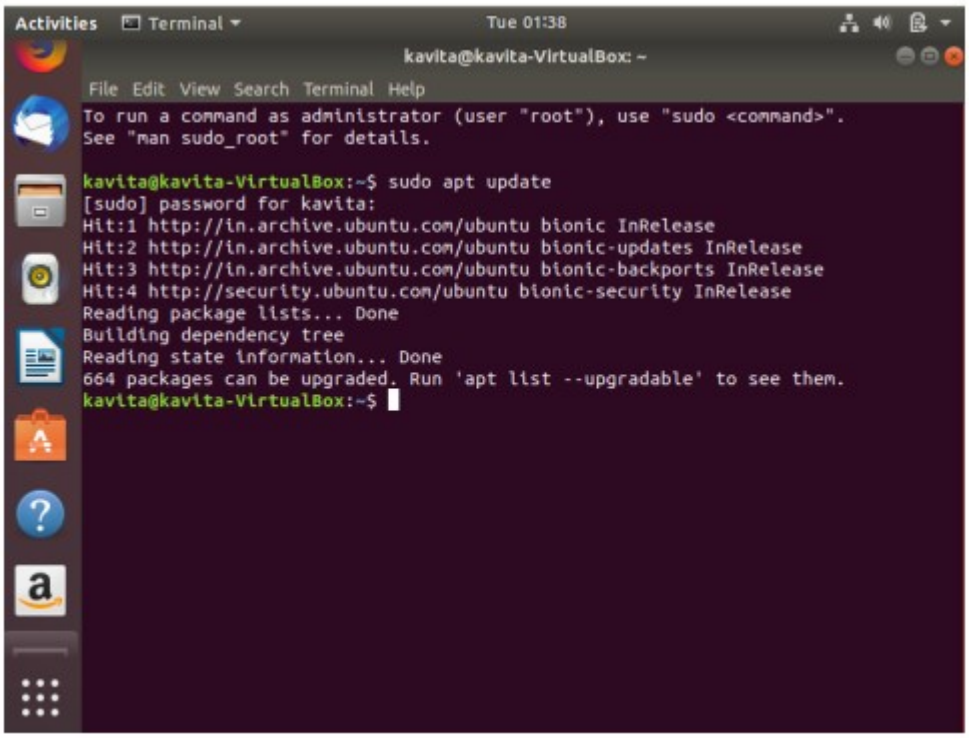


Practical No-3

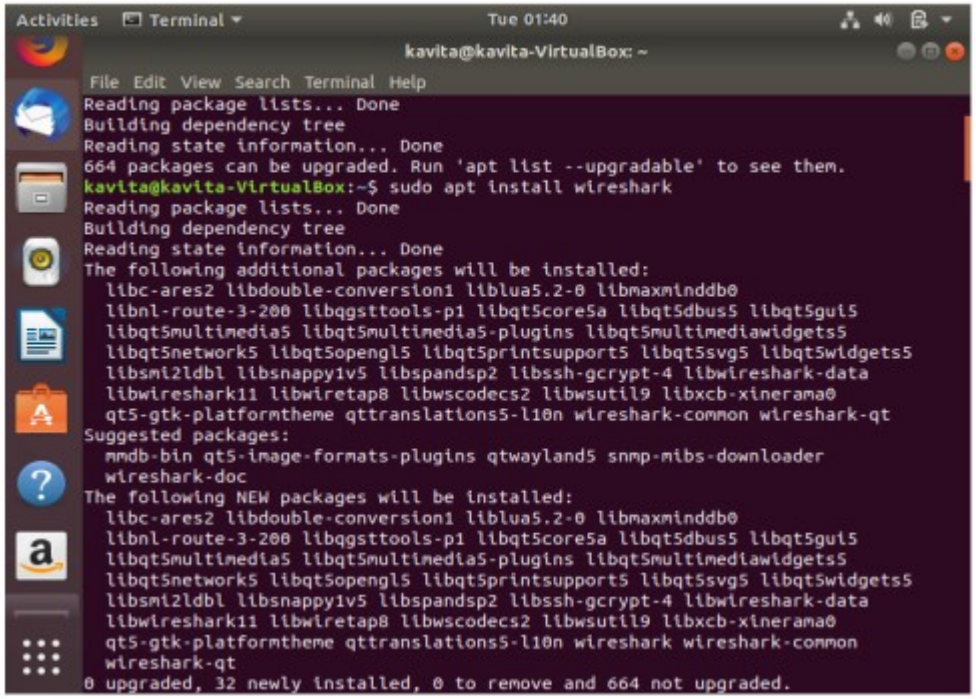
Installation of Wireshark

It is the network protocol for analayzing freely available packages.Wireshark is network packet analyzer.It is used to check incoming and outgoing packets in the network and save it offline analysis.It works on Windows,linux,macOS,FreeBSD etc.

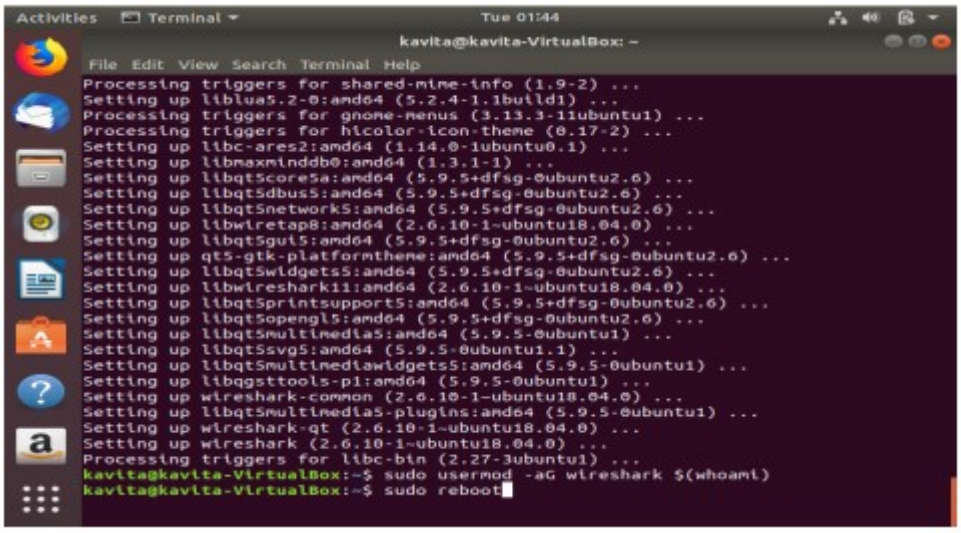
Step 1: Update the system by using sudo apt update



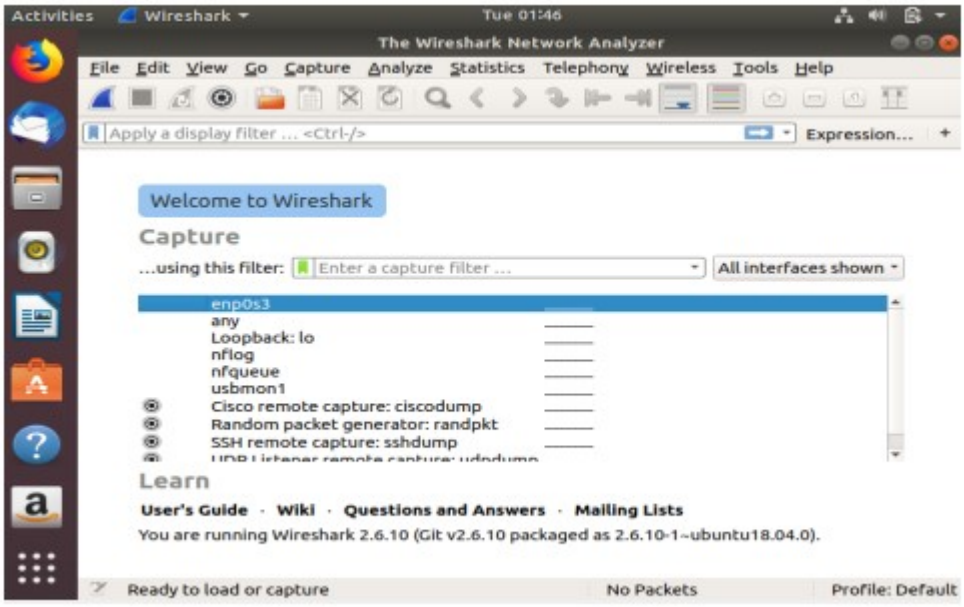
Step2: Install wireshark using sudo apt install wireshark



Step3: Add user in wireshark by using usermod -aG wireshark $(whoami)
Reboot the system-sudo reboot



Step4: To run Wireshark type the command (CI) on terminal Wireshark



Step5: To run Wireshark using GUI go to application and then Wireshark

Practical No-4

Program to simulate UDP server client

Procedure :

i. Create a new project = “UDPserverclient”

ii. Create Package = “UDP”

iii. Create two files under package as “UDPserver.java” and “UDPclient.java”

iv. Write the programs in respective file to send data from server to client.

v. For transfer of a file, create Datagram socket in both with same number. (Socket numbers can be given from 1024 to 65532.

vi. Also define read from beginning to end of file in both programs.

ii. First run “UDPserver.java” then run “UDPclient.java”.

iii. Output will be displayed in console window.

Program for UDP Server :

```
package UDP;
import java.io.*;
import java.net.*;
public class UDPserver {
public static void main(String[] args) throws IOException {
DatagramSocket server = new DatagramSocket(3500);
byte[] buf = new byte[256];
DatagramPacket packet = new
```

DatagramPacket(buf,buf.length);

server.receive(packet);

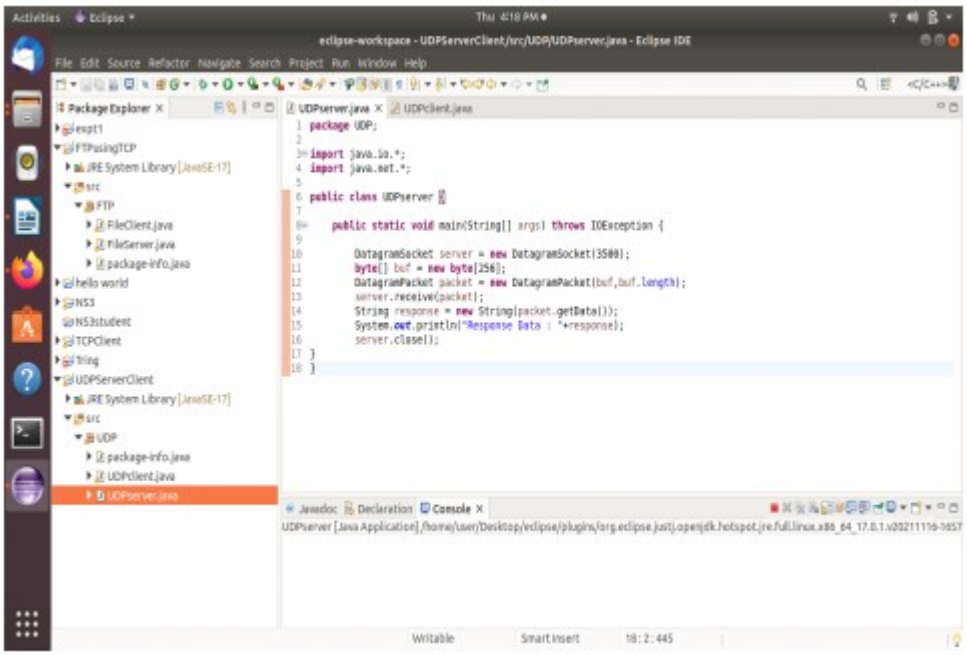
String response = new String(packet.getData());

System.out.println("Response Data : "+response);

server.close();

}

}



Program for UDP Client :

package UDP;

import java.io.*;

import java.net.*;

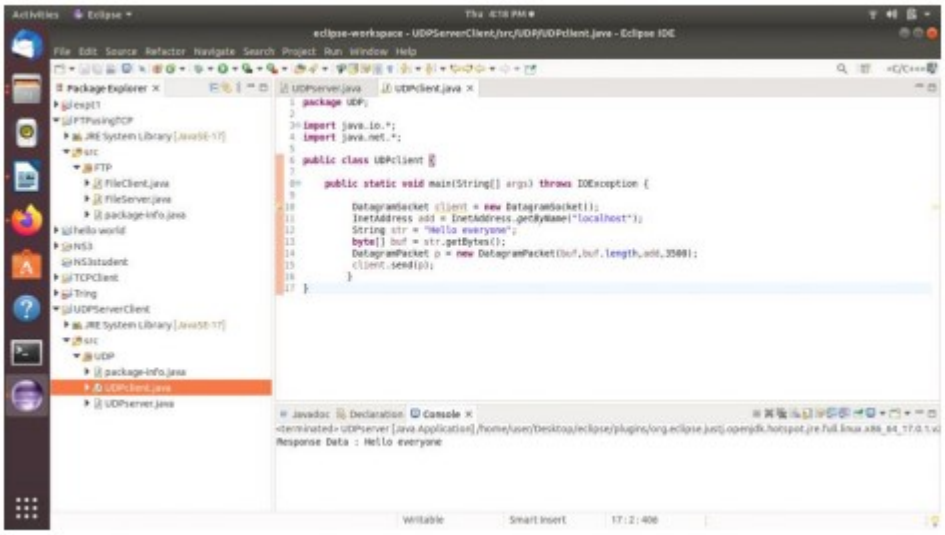
public class UDPclient {

public static void main(String[] args) throws IOException {

DatagramSocket client = new DatagramSocket();

InetAddress add = InetAddress.getByName("localhost");

```
String str = "Hello everyone";
byte[] buf = str.getBytes();
DatagramPacket p = new
DatagramPacket(buf,buf.length,add,3500);
client.send(p);
}
}
```



Practical No-5

Program to simulate FTP using TCP protocol

Procedure :

i. Create a new project = “FTPusingTCP”

ii. Create Package = “FTP”

iii. Create two files under package as “FileServer.java” and “FileClient.java”

iv. Write the programs in respective file to transfer a file from server to client.

v. Give the path which file content need to transfer with new file name as destination.

Example = “Test.txt” to “cubic.txt”

vi. For transfer of a file, create TCP socket in both with same number. (Socket numbers can be given from 1024 to 65532.

vii. Also define path, size of the file, read from beginning to end of file in both programs.

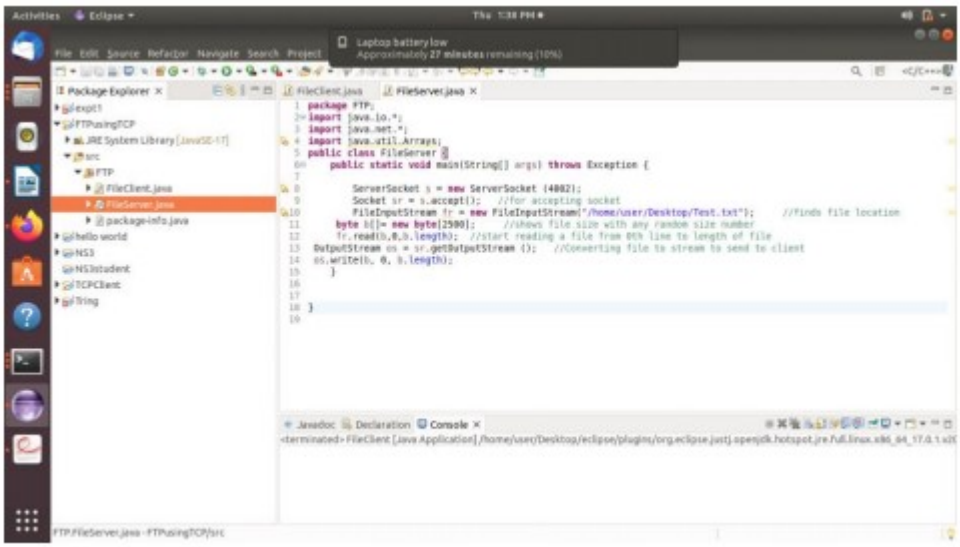
Vii. First run “FileServer.java” then run “FileClient.java”.

Viii. Hereafter you can verify the file is transfered with new name succesfully by FTP using TCP protocol.

Program for TCP Server :

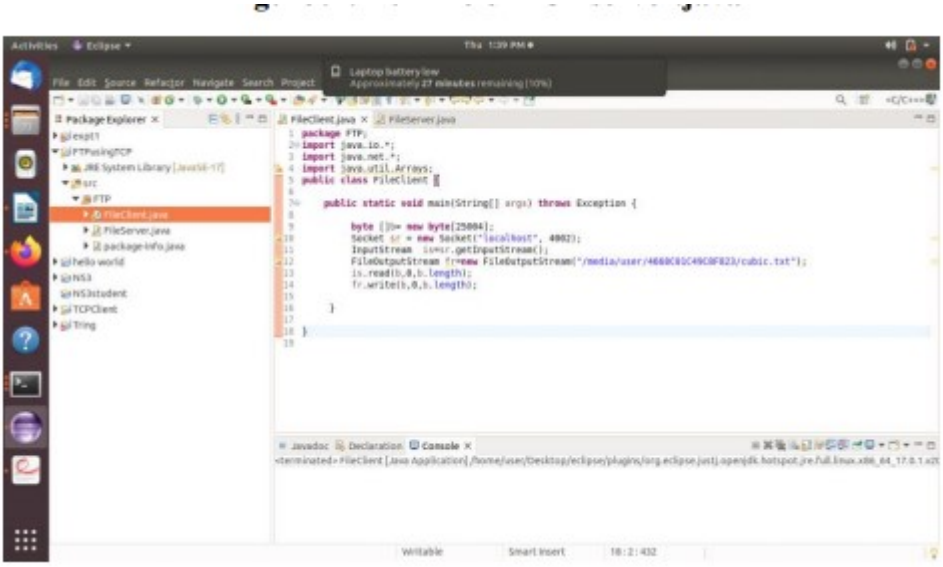
```
package FTP;
import java.io.*;
import java.net.*;
```

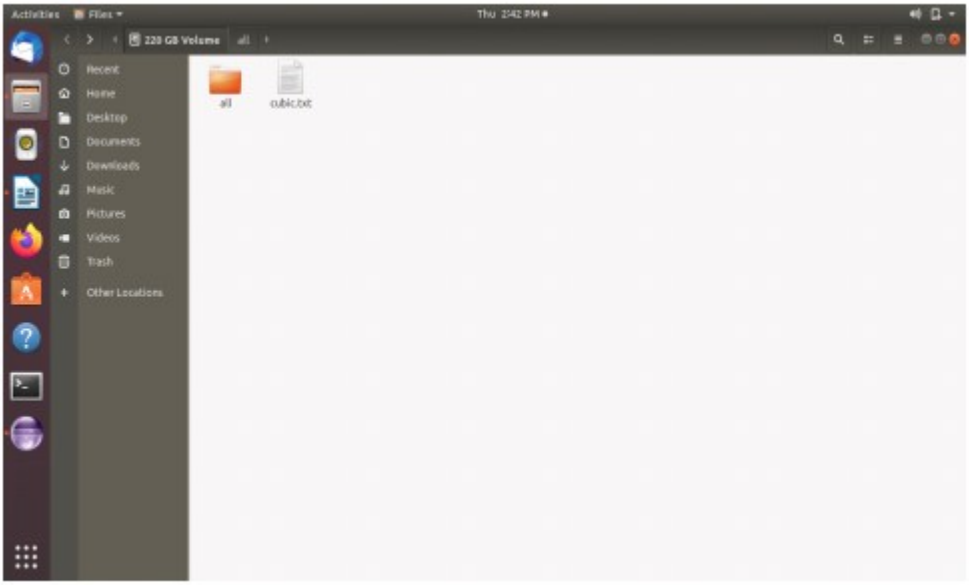
```
import java.util.Arrays;
public class FileServer {
public static void main(String[] args) throws Exception {
ServerSocket s = new ServerSocket (4002);
Socket sr = s.accept(); //for accepting socket
FileInputStream fr = new
FileInputStream("/home/user/Desktop/Test.txt"); //finds file location
byte b[]= new byte[2500]; //shows file size with any random size
number
fr.read(b,0,b.length); //start reading a file from 0th line to length of
file
OutputStream os = sr.getOutputStream (); //Converting file to stream to
send to client
os.write(b, 0, b.length);
}
}
```



Program for Client :

```
package FTP;
import java.io.*;
import java.net.*;
import java.util.Arrays;
public class FileClient {
public static void main(String[] args) throws Exception {
byte []b= new byte[25004];
Socket sr = new Socket("localhost", 4002);
InputStream is=sr.getInputStream();
FileOutputStream fr=new
fileOutputStream("/media/user/4668C01C49C8F823/cubic.txt");
is.read(b,0,b.length);
fr.write(b,0,b.length);
}
}
```




Activities
Files
Thu 2:42 PM
220 GB Volume
all
Recent
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
Trash
Other Locations
all
cubic.txt

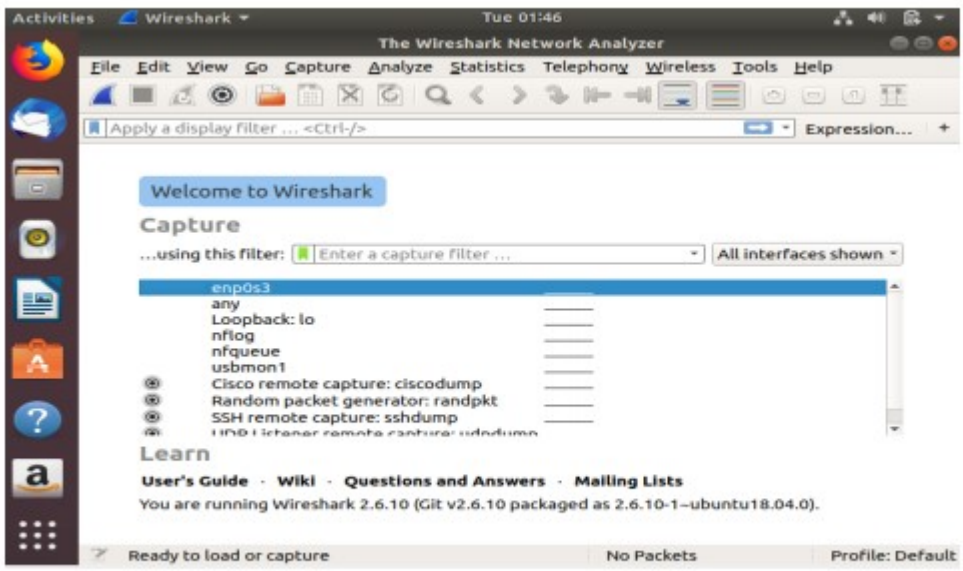
Practical No-6

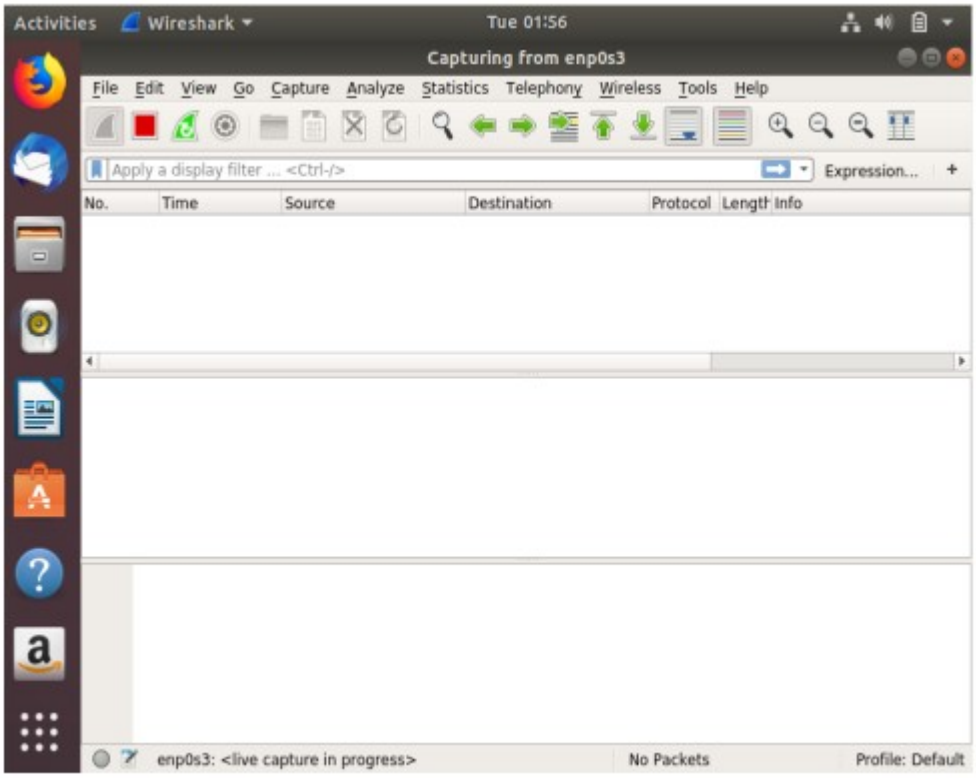
Analyze the network traffic using WireShark

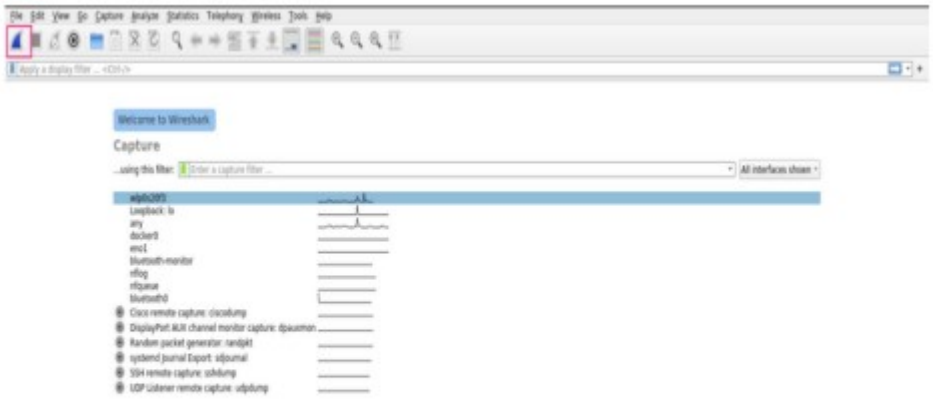
Wireshark is the world's foremost and widely-used network protocol analyzer. It lets us to know what's happening on our network at a microscopic level and is the de facto standard across many commercial and non-profit enterprises, government agencies, and educational institutions.

Network traffic analysis is the routine task of various job roles, such as network administrator, network defenders, incident responders and others. Capture filters with protocol header values Wireshark comes with several capture and display filters. But a user can create display filters using protocol header values as well. Use this technique to analyze traffic efficiently. proto[offset:size(optional)]=value Following the above syntax, it is easy to create a dynamic capture filter, where:

- proto = desired protocol
- offset = header value
- size = data length
- value = data you want to find




Activities
Wireshark
Tue 01:56
Capturing from enp0s3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
No. Time Source Destination Protocol Length Info
enp0s3: <live capture in progress>
No Packets
Profile: Default


Welcome to Wireshark
Capture

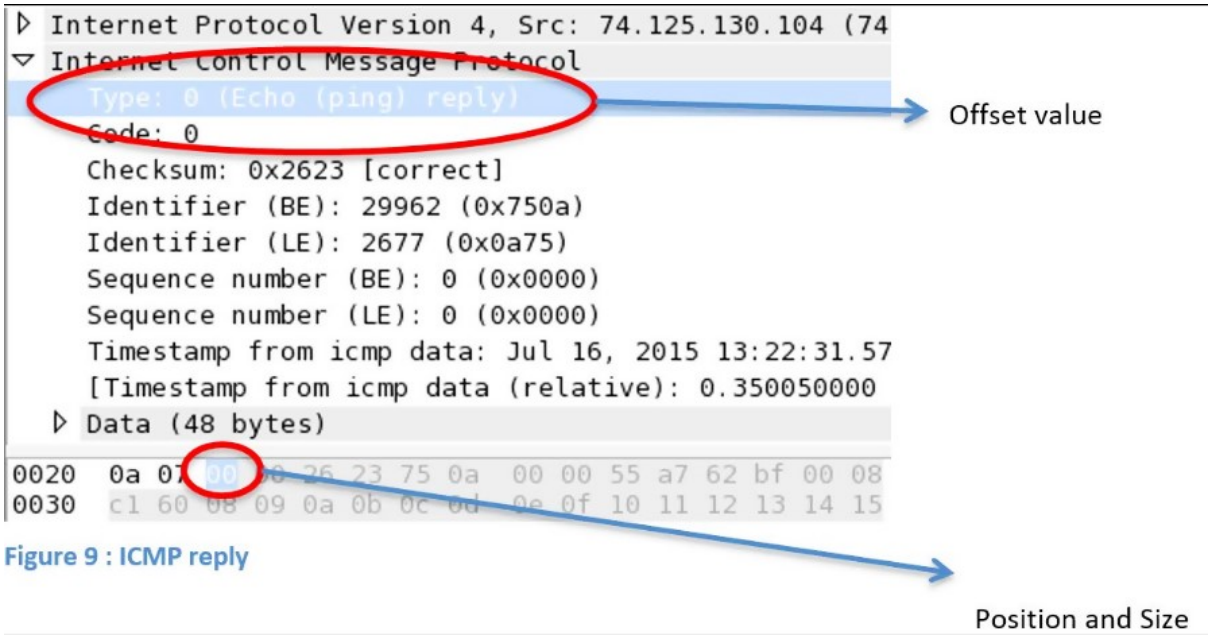


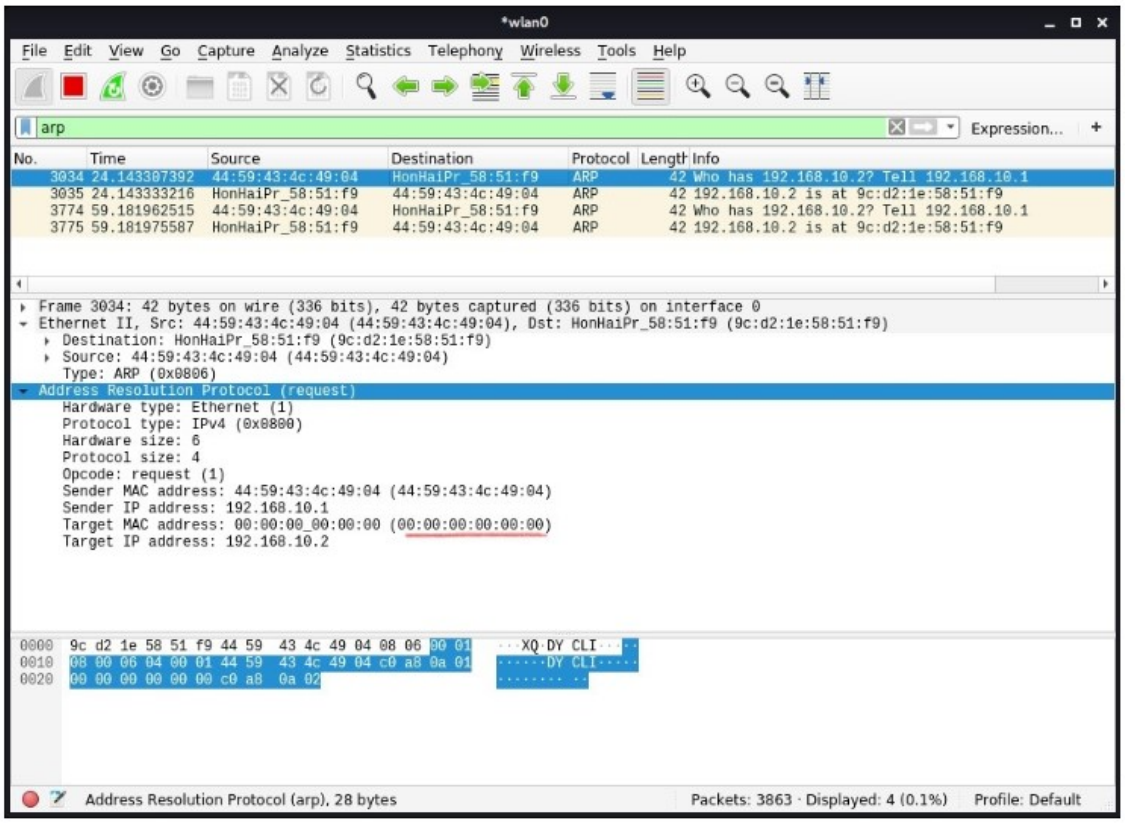
Figure 9 : ICMP reply

Analyze an ARP Request

Using the *'arp'* filter, analyze the captured traffic in Wireshark.

Observe the packet request details from Ethernet and ARP; observe the source and destination IP and sender MAC and IP address.

Monitor the victim's MAC address. Since the destination MAC address is unavailable at the request packet stage, the victim's MAC address is zero, and the destination IP is the local system IP address.

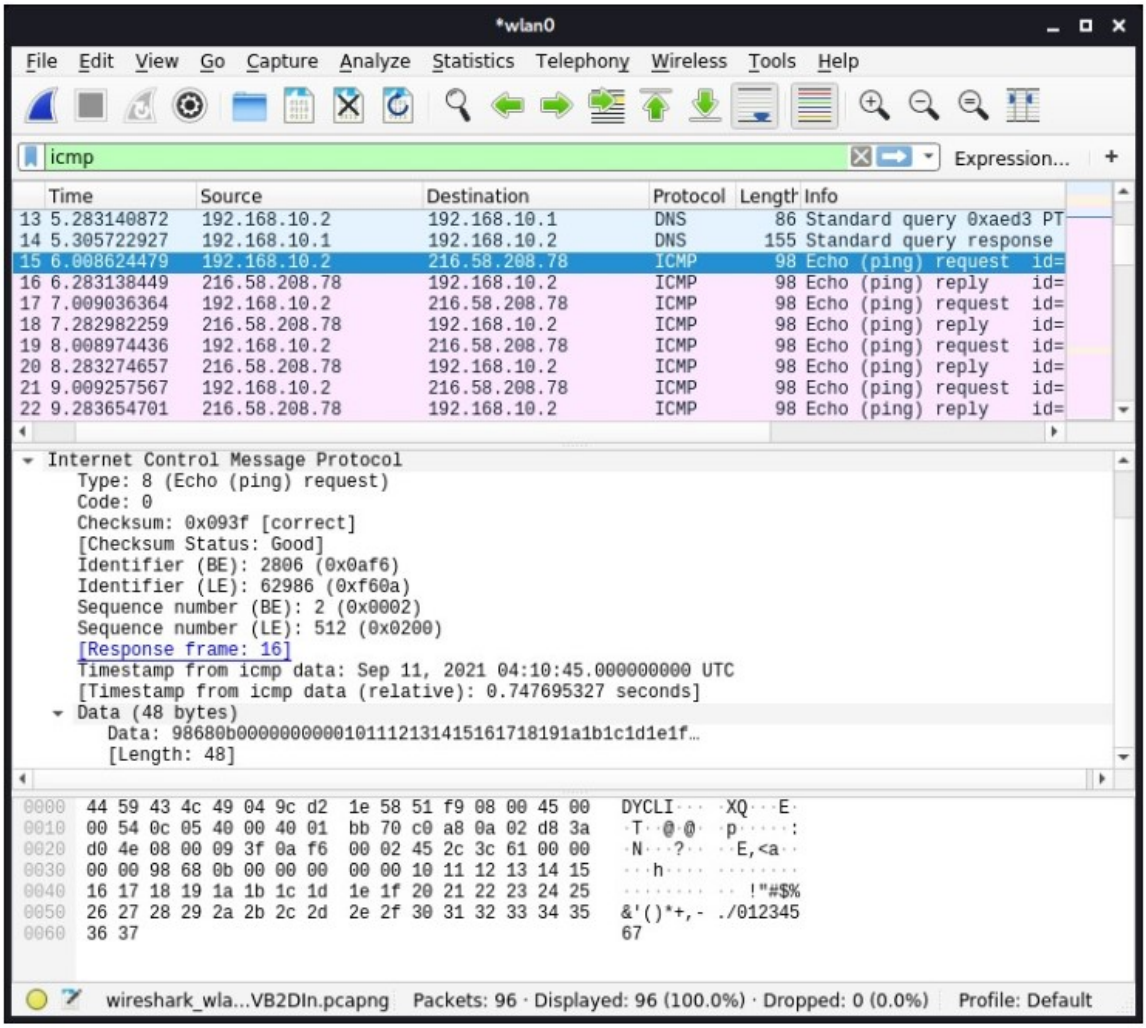


CMP traffic analysis

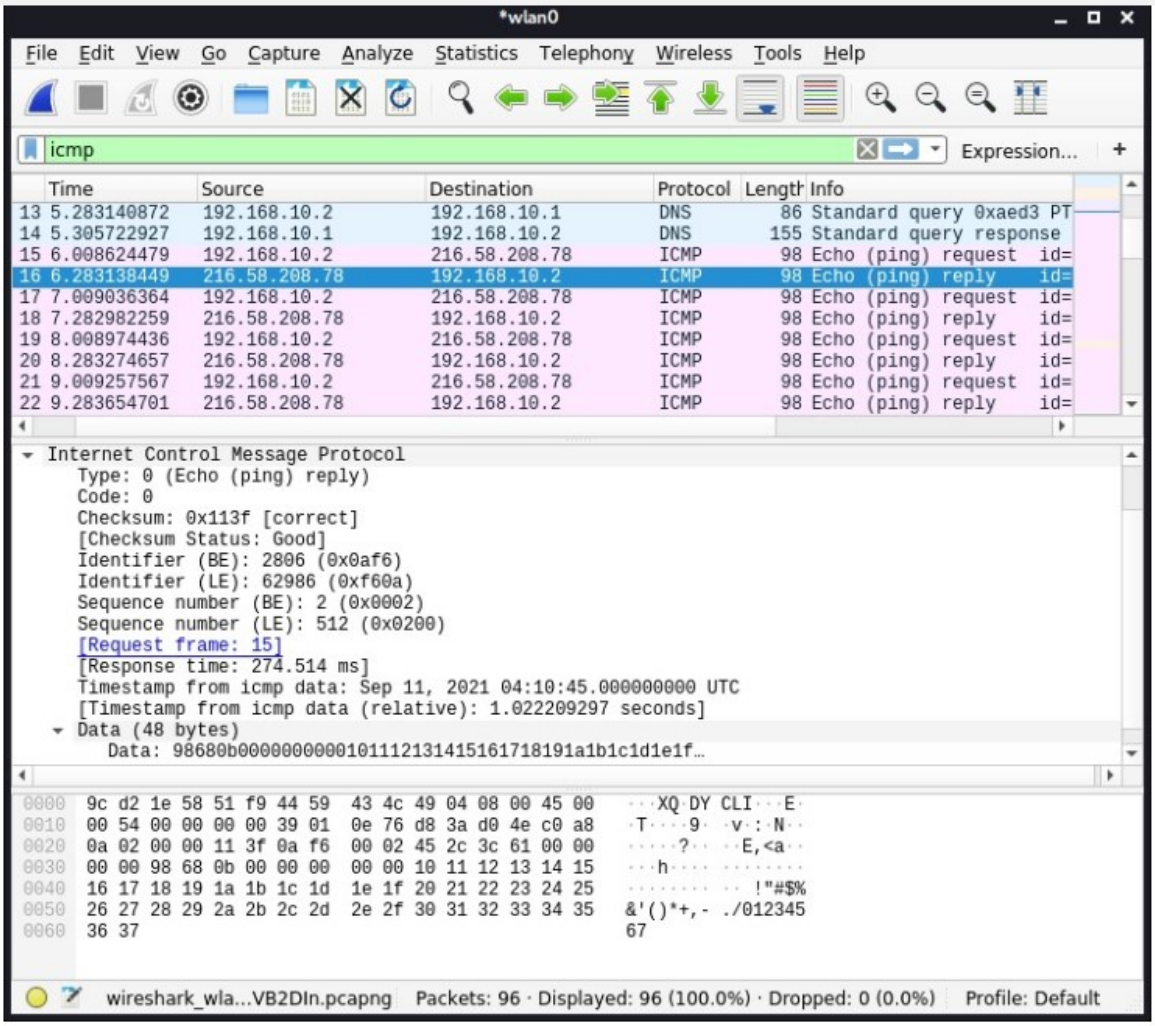
ICMP is used for error alerting and monitoring to verify whether data arrives in a timely basis at its desired destination.

To capture ICMP traffic, ping Google.com. Use the *'ICMP'* filter to see ICMP traffic. Click the ICMP echo-request packet from the Wireshark capture window and start observing the information.

In the ***request packet,*** the source IP is your (requestor) IP address. Whereas the destination IP is that of Google. You can also analyze the ICMP details like Checksum, Identifier Number, Sequence Number, etc.



In the ***response packet***, observe the swapping of IPs between source and destination. You can also compare both request and response details, as they are similar.

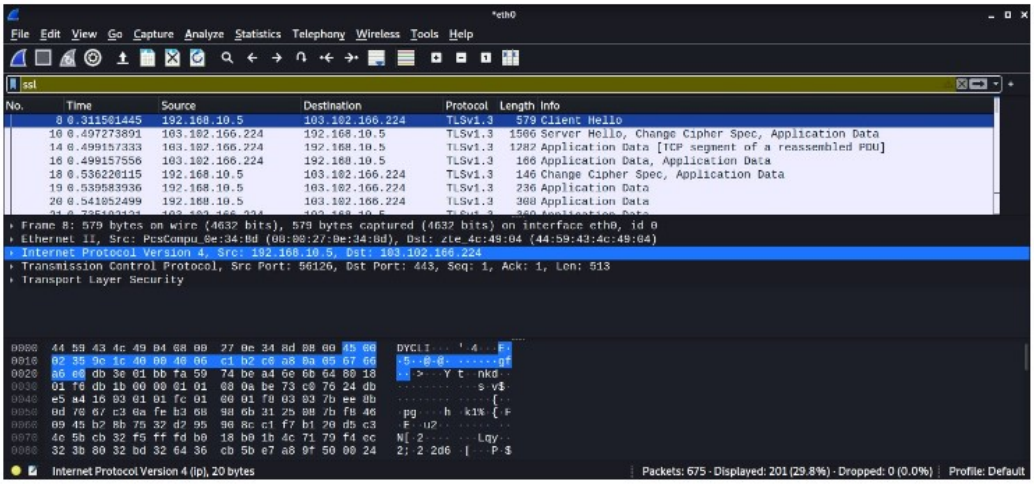


HTTPS traffic analysis

The Hypertext Transfer Application Layer Protocol (HTTP) utilizes the internet to establish protocols whenever the HTTP client/server transmits/receives HTTP requests.

Start a Wireshark capture -> Open a web browser -> Navigate to any HTTPS-based website -> Stop the Wireshark capture.

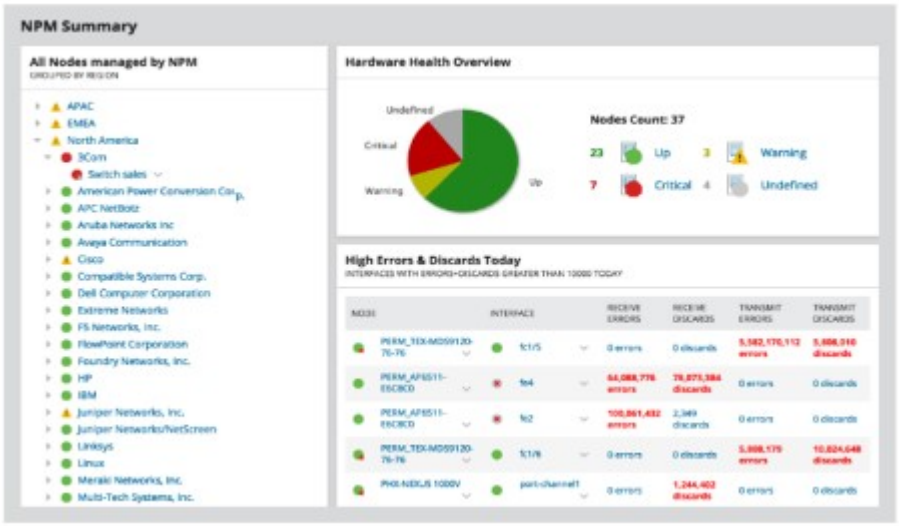
Input ' ssl' in the filter box to monitor only HTTPS traffic -> Observe the first TLS packet -> The destination IP would be the target IP (server).


*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ssl
No. Time Source Destination Protocol Length Info
8 0.311501445 192.168.10.5 103.102.166.224 TLSv1.3 579 Client Hello
10 0.497273891 103.102.166.224 192.168.10.5 TLSv1.3 1506 Server Hello, Change Cipher Spec, Application Data
14 0.499157333 103.102.166.224 192.168.10.5 TLSv1.3 1282 Application Data [TCP segment of a reassembled PDU]
16 0.499157556 103.102.166.224 192.168.10.5 TLSv1.3 166 Application Data, Application Data
18 0.536220115 192.168.10.5 103.102.166.224 TLSv1.3 146 Change Cipher Spec, Application Data
19 0.539583936 192.168.10.5 103.102.166.224 TLSv1.3 236 Application Data
20 0.541052499 192.168.10.5 103.102.166.224 TLSv1.3 308 Application Data
Frame 8: 579 bytes on wire (4632 bits), 579 bytes captured (4632 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_0e:34:0d (08:00:27:0e:34:0d), Dst: zte_4c:49:04 (44:59:43:4c:49:04)
Internet Protocol Version 4, Src: 192.168.10.5, Dst: 103.102.166.224
Transmission Control Protocol, Src Port: 56126, Dst Port: 443, Seq: 1, Ack: 1, Len: 513
Transport Layer Security
Internet Protocol Version 4 (ip), 20 bytes
Packets: 675 - Displayed: 201 (29.8%) - Dropped: 0 (0.0%)
Profile: Default

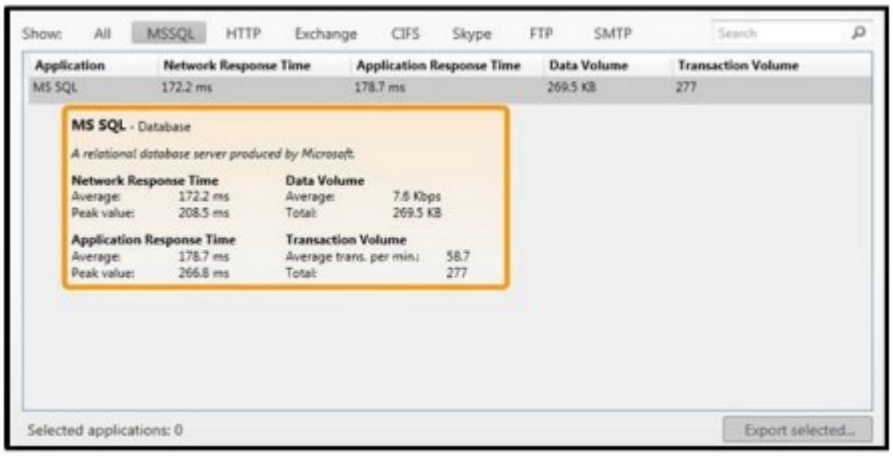
Practical No-7

Analyze the performance parameters of network using Wire Shark

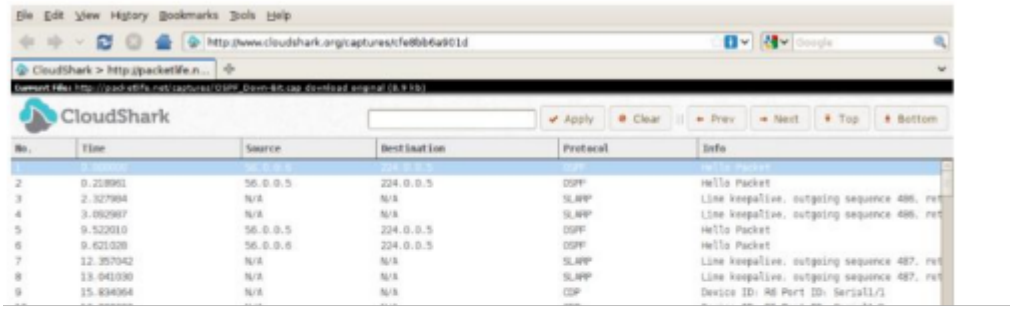
SolarWinds Network Performance Monitor



SolarWinds Response Time Viewer for Wireshark



Cloudshark is a platform designed to display network capture files directly in the browser without the need for desktop applications or tools. Simply upload, email, or link the captures files and get the results. It helps to resolve network issues faster and flawlessly. Besides, it includes features such as drag-and-drop capture files, drop-box-like activity, allows sharing of links with co-workers, and offers advanced analysis.

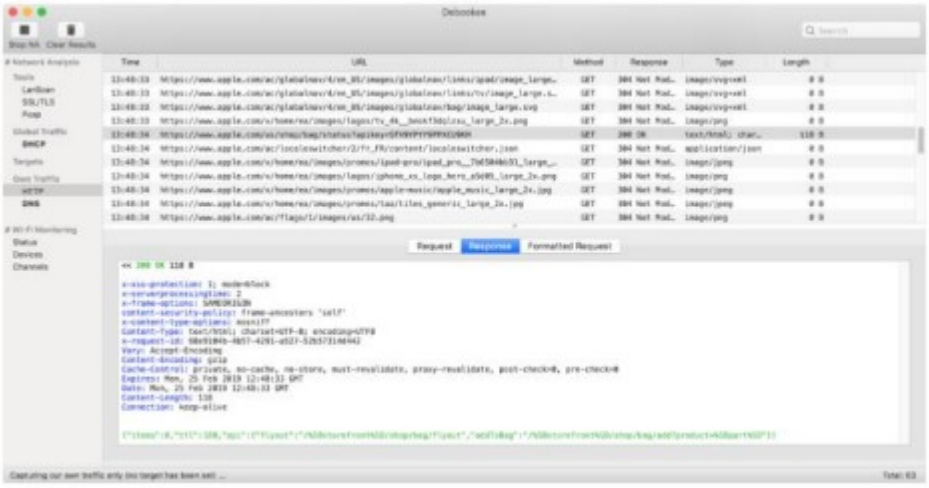


Sysdig Sysdig is a powerful, cross-platform, and open-source flexible network monitoring system designed specifically for Linux. It works with others but with limited functionalities. Sysdig uses various monitoring and troubleshooting tools for Linux system performance, such as:

- Strace (for discovering system calls)
- htop (for real-time process monitoring)
- iftop (for real-time bandwidth monitoring)
- netstat (for network connection monitoring)
- tcpdump (for raw network traffic monitoring)
- Isof (to know the process used to view the opened files)

Debooke Debooke is one of the simplest and most powerful network monitoring and traffic analyzer tools for macOS. It allows users to intercept and monitor the network traffic of any device in the same subnet. It helps to capture data network packet data from mobile, printer, Mac, TV. Features of Debooke:

- Keep track of Wi-Fi bandwidth use and consumption
- Help with network monitoring and in-depth analysis
- Display the Wi-Fi clients and the APIs to which they’re associated
- Scan IP, LAN to keep track of all the active and connected devices



Advanced Statistical Tools The captured traffic is then analyzed using Wireshark basic statistical tools and advanced tools for various performance parameters.

Graphs

Wireshark offers graphing capabilities which can present captured packets in an interesting format

that makes the analysis process much more effective and easy to adapt [14] [15]. There are multiple

types of graphs available