Practical-8

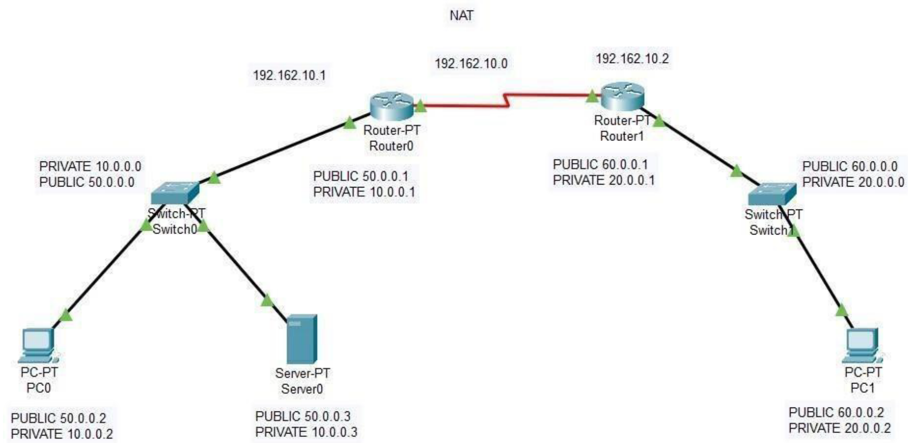
Aim: Examine Network Address Translation (NAT) in CISCO packet tracer.

Objectives :

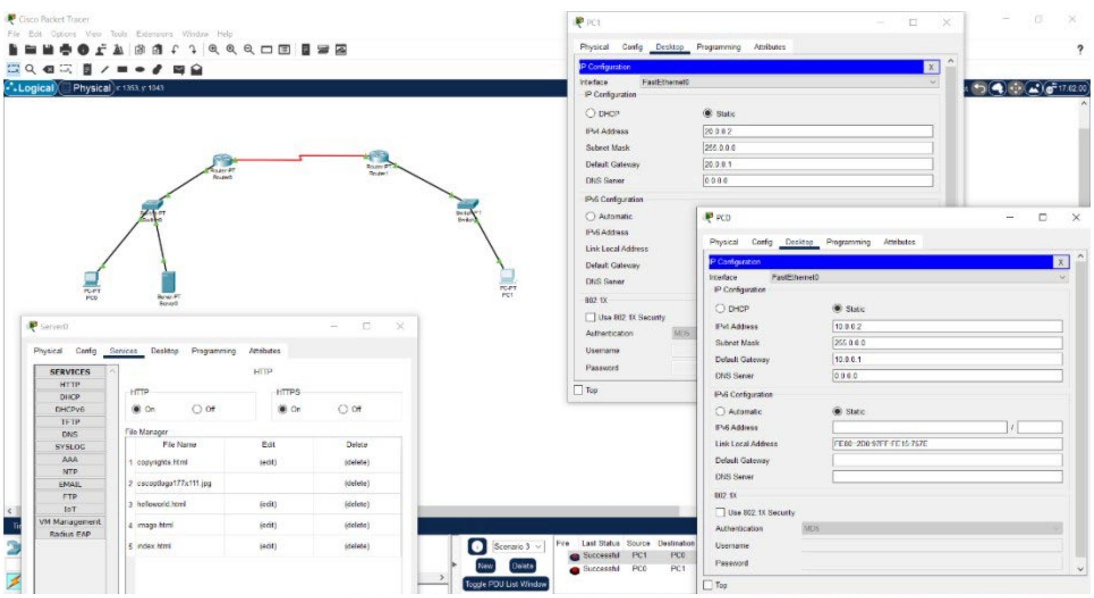
Examine NAT processes as traffic traverses a NAT border router.

Background / Preparation :

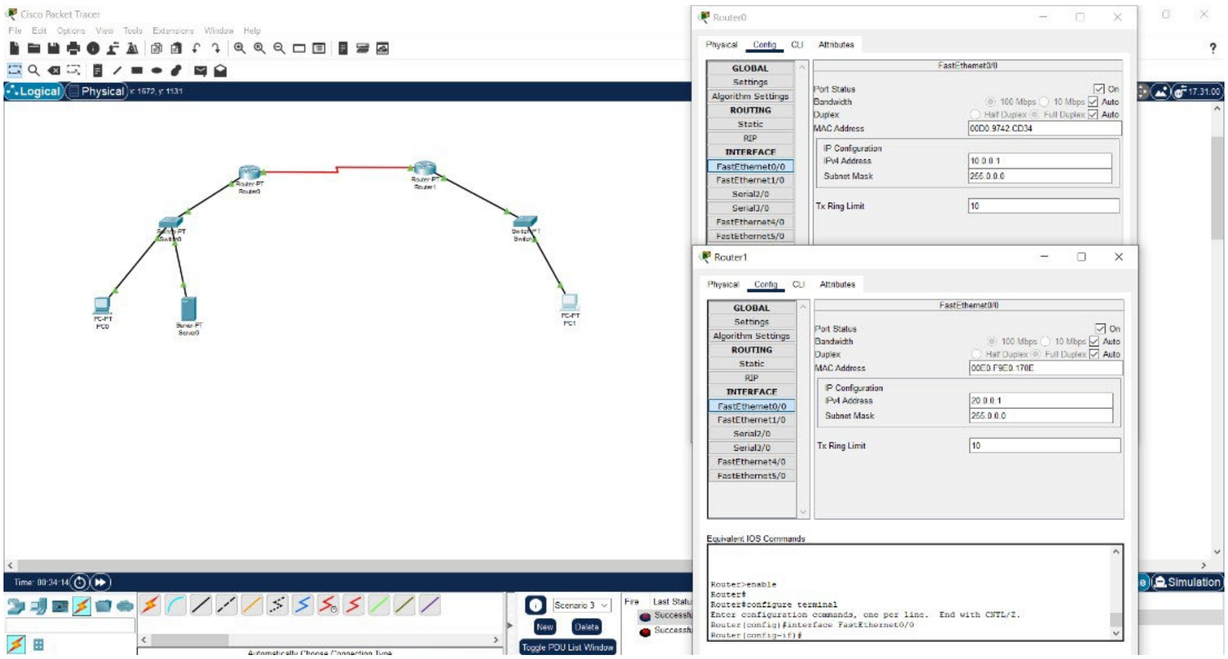
In this activity, you will use Packet Tracer Simulation mode to examine the contents of the IP header as traffic crosses the NAT border router.



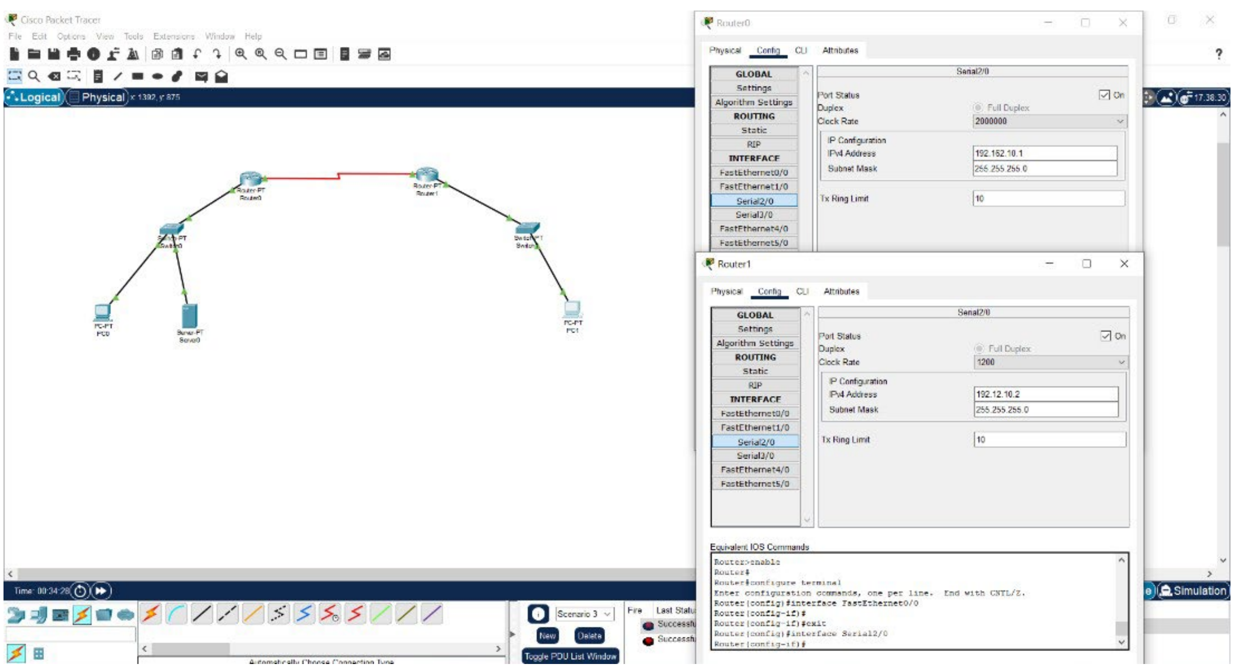
Step 1: Assign IP address to pc.



Step 2: Assign IP address in FastEthernet0/0 of both routers.

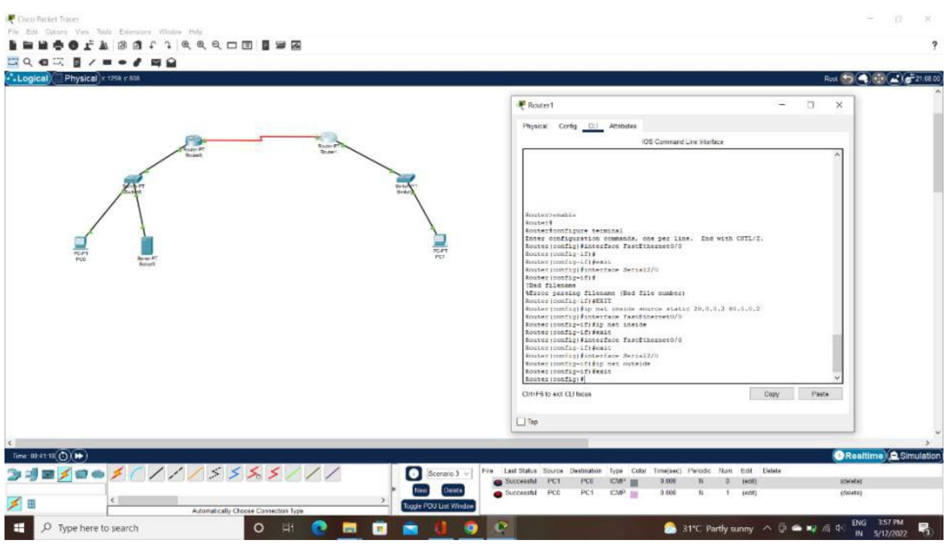


Step 3: Assign IP address in Serial2/0 of both routers.



Step 4: Later on type the NAT commands in the CLI of both the routers.

```
Router(config-if)#exit
Router(config)#ip nat inside source static 10.0.0.2 50.0.0.2
Router(config)#ip nat inside source static 10.0.0.3 50.0.0.3
Router(config)#
Router(config)#
Router(config)#interface FastEthernet0/0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#
Router(config)#
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet1/0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#
Router(config)#
Router(config)#interface FastEthernet1/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#
```



Preparation:

Step 1: Prepare the network for Simulation mode.

Verify that the network is ready to send and receive traffic. All the link lights should be green. If some link lights are still amber, you can switch between Simulation and

Realtime mode several times to force the lights to turn green faster. Switch to Simulation mode before going to the next step.

Step 2: Send an HTTP request from an inside host to an outside web server.

- Click Customer PC. Click the Desktop tab and then Web Browser. In the URL field, type the web address for the ISP server (www.ispserver.com). Make sure that you are in Simulation mode, and then click Go.
- In the event list, notice that Customer PC queues a DNS request and sends out an ARP request. You can view the contents of the ARP request by either clicking on the packet in the topology or clicking on the packet color under Info in the Event List window.
- In the PDU Information at Device: Customer PC window, which IP address is Customer PC attempting to find a MAC address for?
- In the Event List window, click Capture/Forward twice. Which device answers the ARP request from Customer PC? Which MAC address is placed inside the ARP reply?
- In the Event List window, click Capture/Forward twice. Customer PC accepts the ARP replay and then builds another packet. What is the protocol for this new packet? If you click Outbound PDU Details for this packet, you can see the details of the protocol.
- In the Event List window, click Capture/Forward twice. Click the packet at the www.customerserver.com server. Then click the Outbound PDU Details tab. Scroll down to the bottom to see the Application Layer data. What is the IP address for the ISP server?
- In the Event List window, click Capture/Forward twice. Customer PC now formulates another ARP request. Why?
- In the Event List window, click Capture/Forward 10 times until Customer PC formulates an HTTP request packet. Customer PC finally has enough information to request a web page from the ISP server.
- In the Event List window, click Capture/Forward three times. Click the packet at Customer Router to examine the contents. Customer Router is a NAT border router. What is the inside local address and the inside global address for Customer PC?
- In the Event List window, click Capture/Forward seven times until the HTTP reply reaches

Customer Router. Examine the contents of the HTTP reply and notice that the inside local and global addresses have changed again as the packet is forwarded on to Customer PC.

Step 3: Send an HTTP request from an outside host to an inside web server.

- Customer Server provides web services to the public (outside addresses) through the domain name www.customerserver.com. Follow a process similar to Step 2 to observe an HTTP request on ISP Workstation.
- •Click ISP Workstation. Click the Desktop tab, and then Web Browser. In the URL field, type the Customer Server web address (www.customerserver.com). Make sure that you are in Simulation mode, and then click Go.
- •You can either click Auto Capture/Play or Capture/Forward to step through each stage of the process. The same ARP and DNS processes occur before the ISP Workstation can formulate an HTTP request.

Practical-9

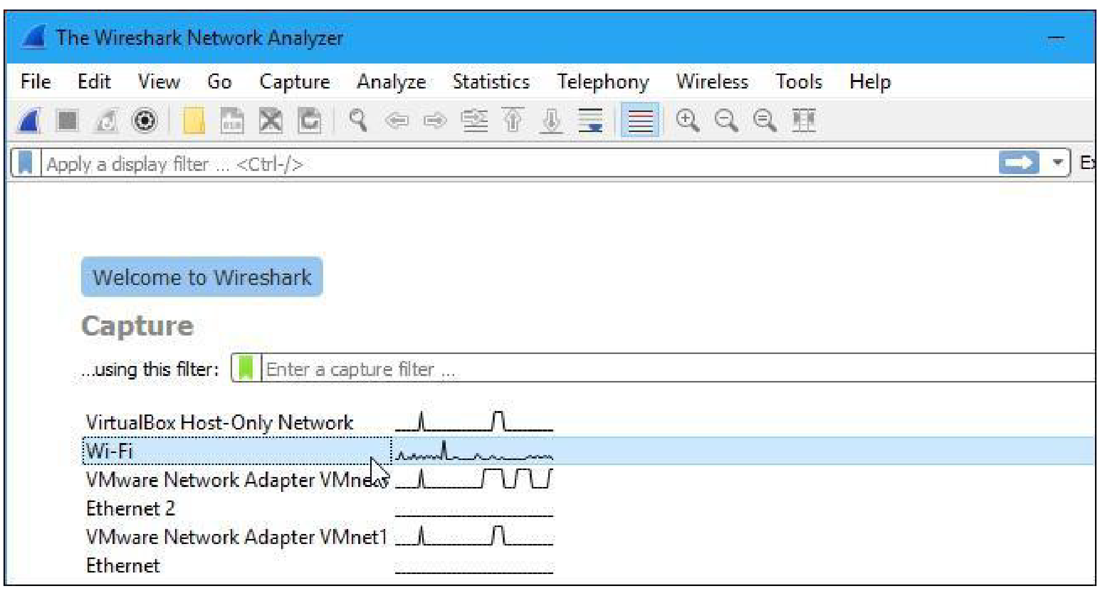
Aim: Introduction to packet capturing using Wireshark.

Wireshark, a network analysis tool formerly known as Ethereal, captures packets in real time and display them in human-readable format. Wireshark includes filters, color coding, and other features that let you dig deep into network traffic and inspect individual packets.

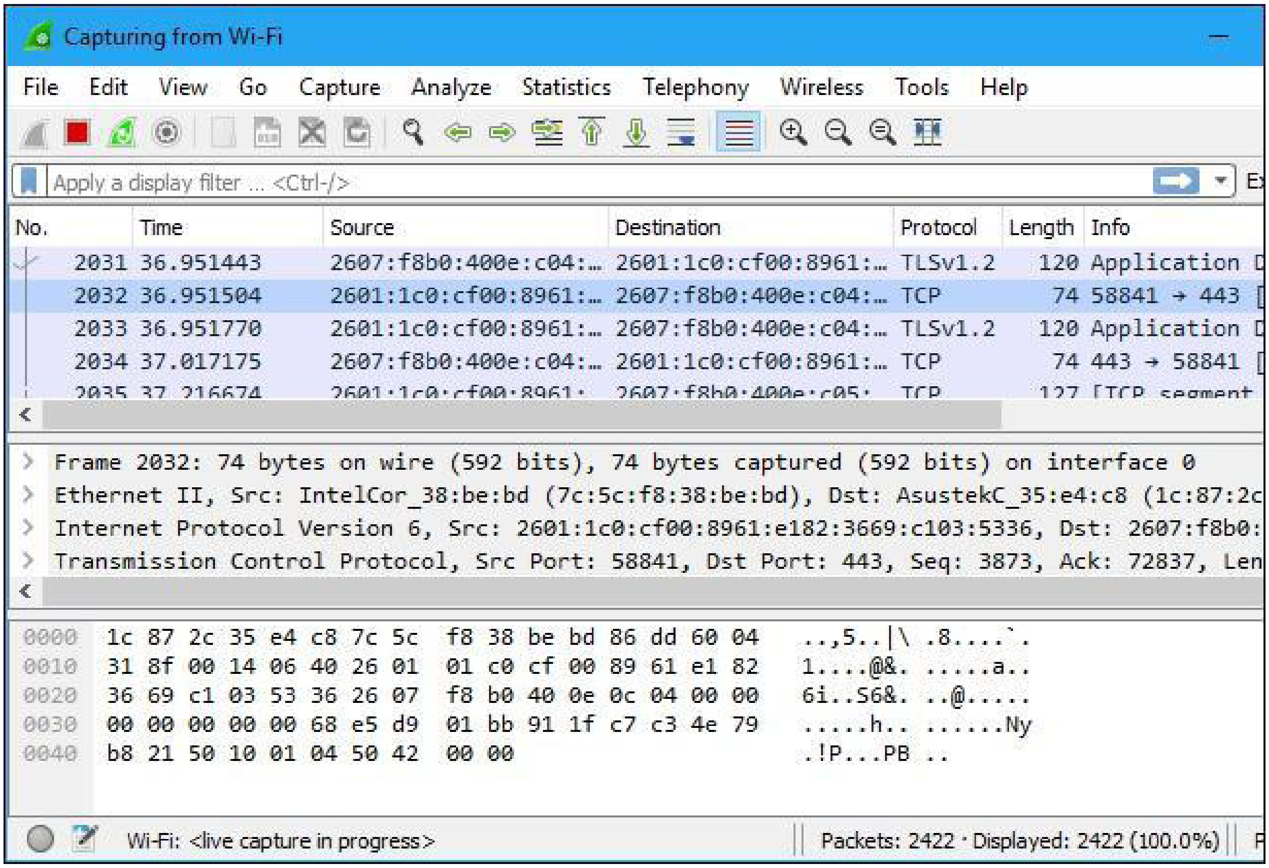
This tutorial will get you up to speed with the basics of capturing packets, filtering them, and inspecting them. You can use Wireshark to inspect a suspicious program's network traffic, analyze the traffic flow on your network, or troubleshoot network problems.

Capturing Packets :

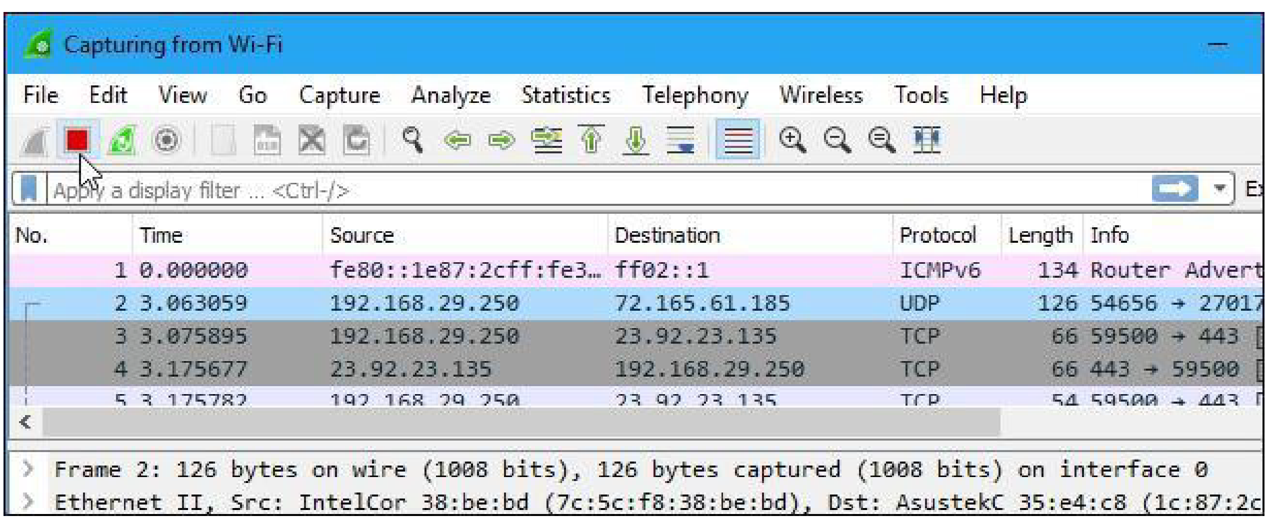
After downloading and installing Wireshark, you can launch it and double-click the name of a network interface under Capture to start capturing packets on that interface. For example, if you want to capture traffic on your wireless network, click your wireless interface. You can configure advanced features by clicking Capture > Options, but this isn't necessary for now.



As soon as you click the interface's name, you'll see the packets start to appear in real time. Wireshark captures each packet sent to or from your system. If you have promiscuous mode enabled—it's enabled by default—you'll also see all the other packets on the network instead of only packets addressed to your network adapter. To check if promiscuous mode is enabled, click Capture > Options and verify the "Enable promiscuous mode on all interfaces" checkbox is activated at the bottom of this window.



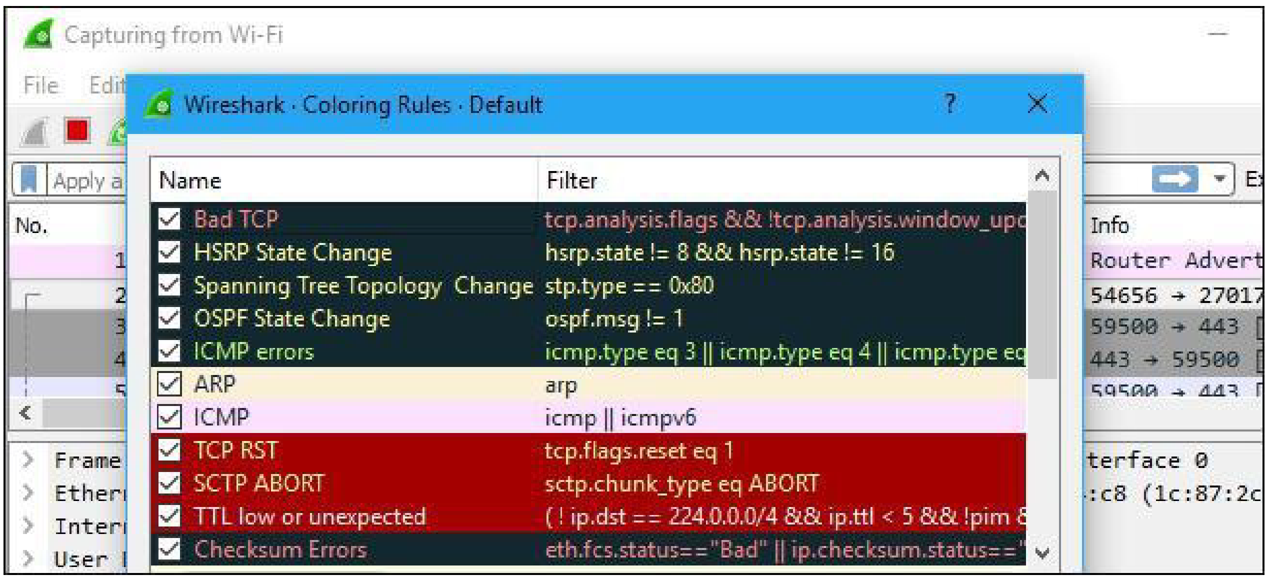
Click the red "Stop" button near the top left corner of the window when you want to stop capturing traffic.



Color Coding :

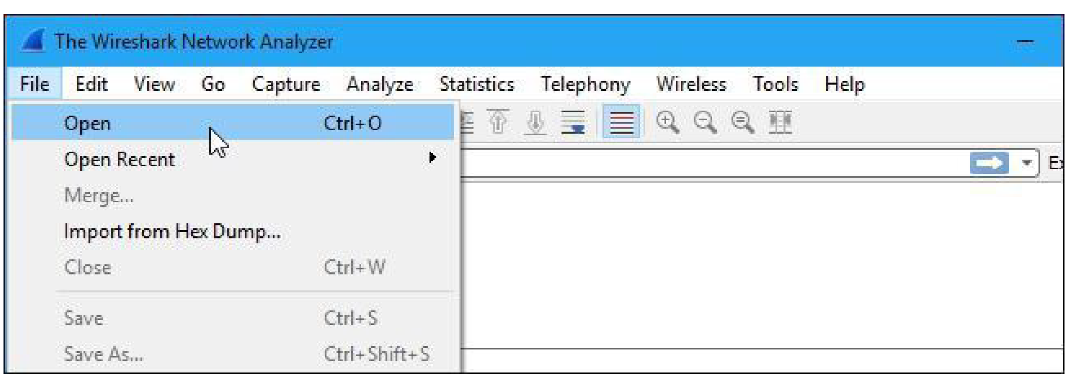
You'll probably see packets highlighted in a variety of different colors. Wireshark uses colors to help you identify the types of traffic at a glance. By default, light purple is TCP traffic, light blue is UDP traffic, and black identifies packets with errors—for example, they could have been delivered out of order.

To view exactly what the color codes mean, click View > Coloring Rules. You can also customize and modify the coloring rules from here, if you like.



Sample Captures :

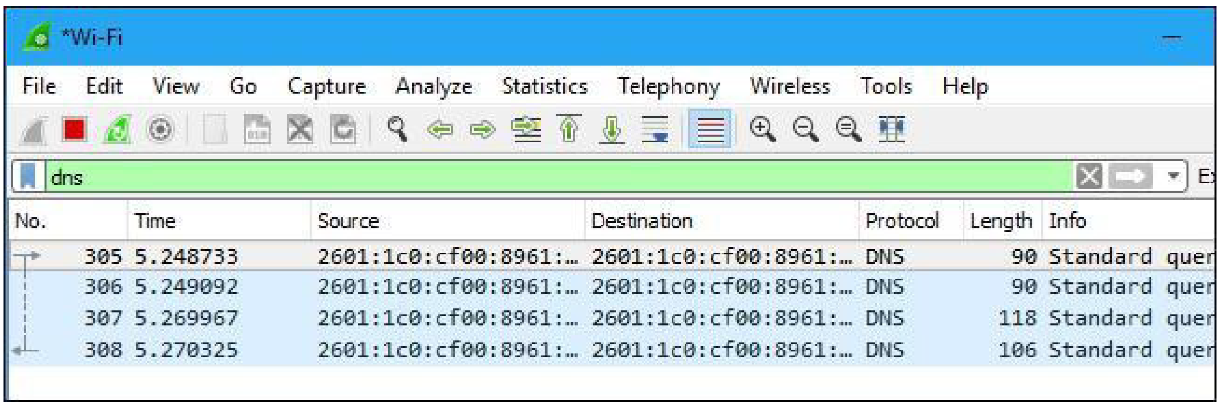
If there's nothing interesting on your own network to inspect, Wireshark's wiki has you covered. The wiki contains a page of sample capture files that you can load and inspect. Click File > Open in Wireshark and browse for your downloaded file to open one.You can also save your own captures in Wireshark and open them later. Click File > Save to save your captured packets.



Filtering Packets :

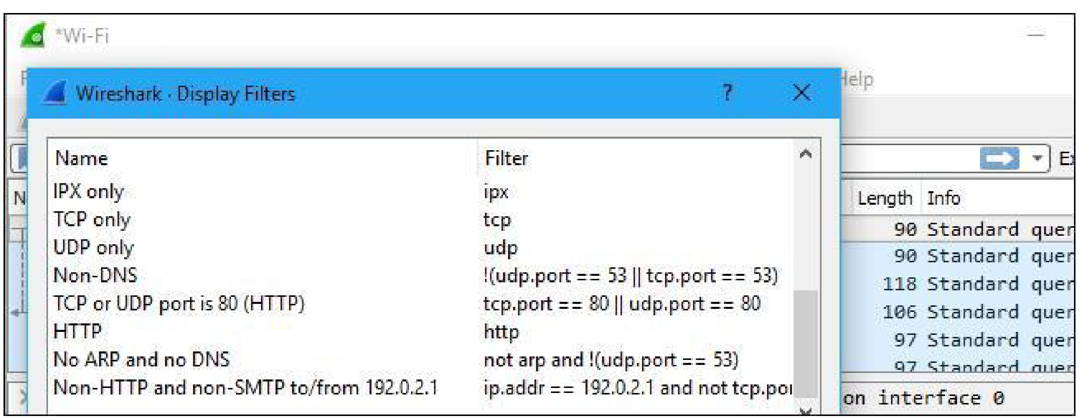
If you're trying to inspect something specific, such as the traffic a program sends when phoning home, it helps to close down all other applications using the network so you can narrow down the traffic. Still, you'll likely have a large amount of packets to sift through. That's where Wireshark's filters come in.

The most basic way to apply a filter is by typing it into the filter box at the top of the window and clicking Apply (or pressing Enter). For example, type "dns" and you'll see only DNS packets. When you start typing, Wireshark will help you autocomplete your filter.

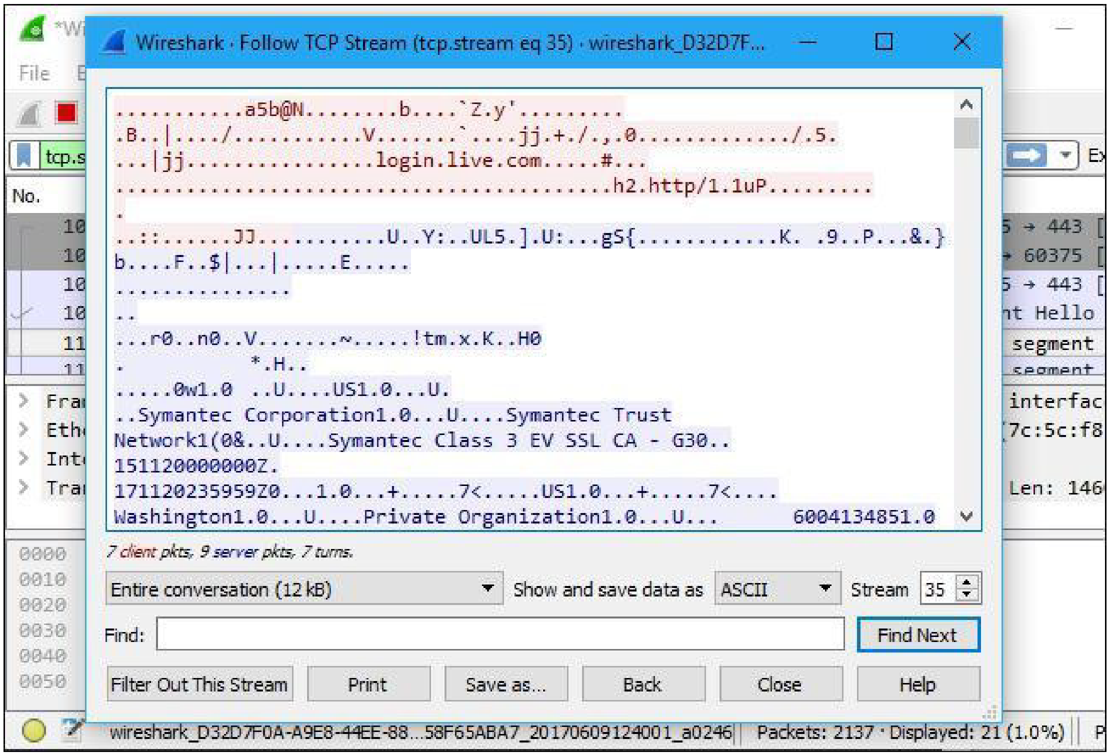


You can also click Analyze > Display Filters to choose a filter from among the default filters included in Wireshark. From here, you can add your own custom filters and save them to easily access them in the future.

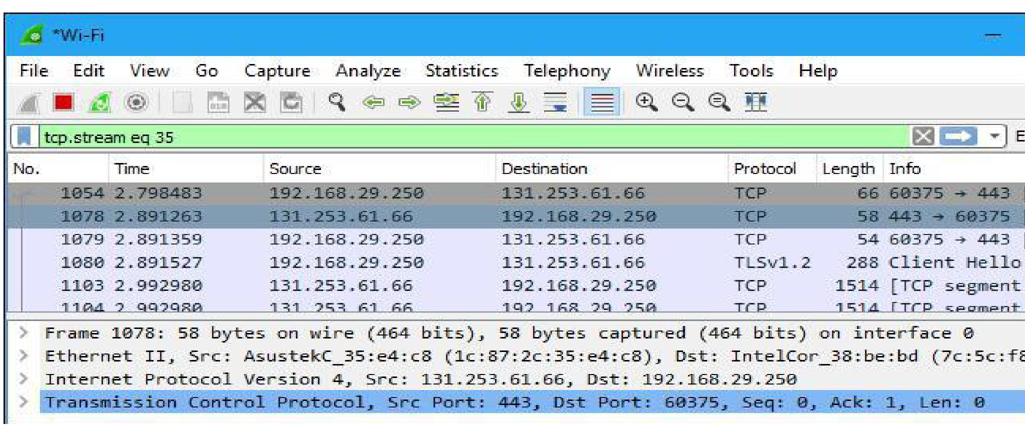
For more information on Wireshark's display filtering language, read the Building display filter expressions page in the official Wireshark documentation.



Another interesting thing you can do is right-click a packet and select Follow > TCP Stream. You'll see the full TCP conversation between the client and the server. You can also click other protocols in the Follow menu to see the full conversations for other protocols, if applicable.

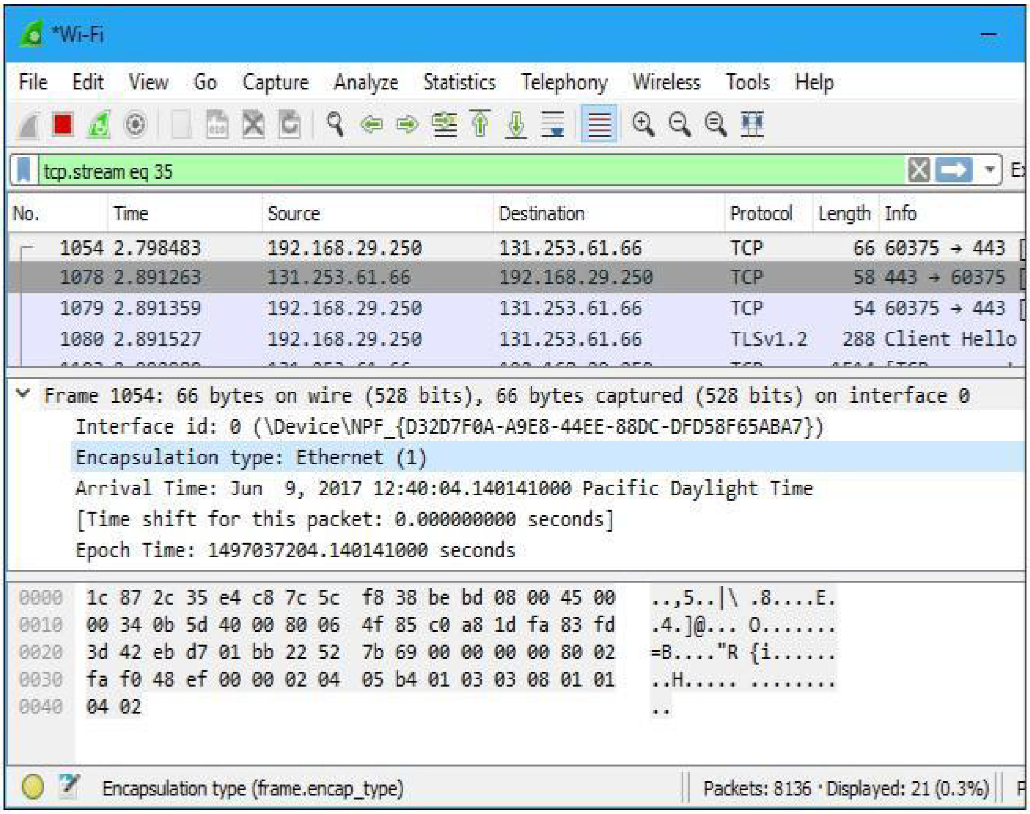


Close the window and you'll find a filter has been applied automatically. Wireshark is showing you the packets that make up the conversation.

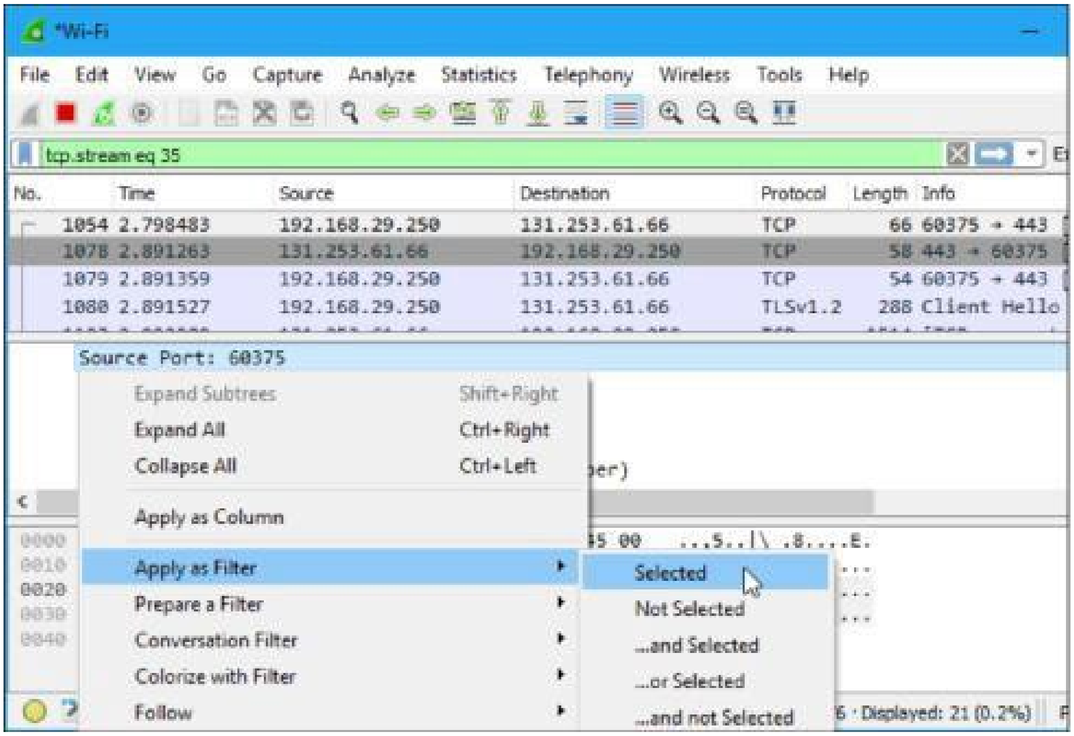


Inspecting Packets :

Click a packet to select it and you can dig down to view its details.



You can also create filters from here — just right-click one of the details and use the Apply as Filter submenu to create a filter based on it.



Wireshark is an extremely powerful tool, and this tutorial is just scratching the surface of what you can do with it. Professionals use it to debug network protocol implementations, examine security problems and inspect network protocol internals.

Practical-10

Aim: Implementing socket programming with UDP and TCP.

A) 1st Method:-

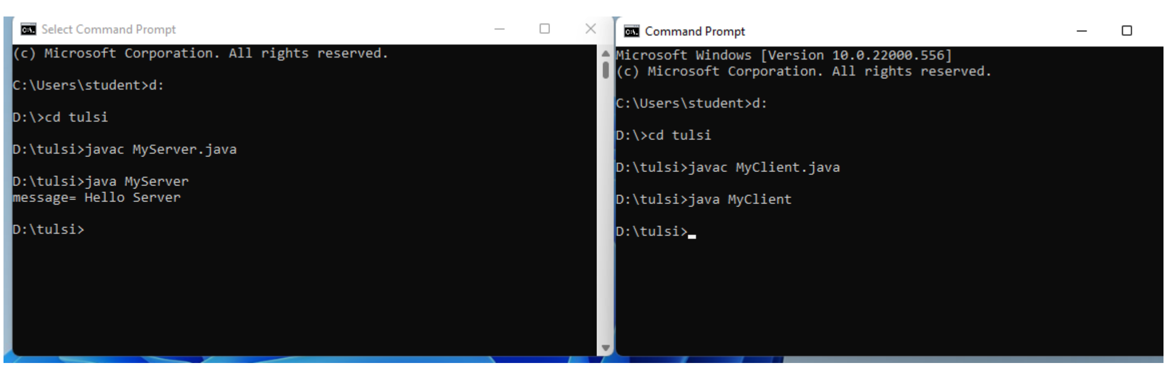
i. Code For Server Side :

```
import java.io.*; import java.net.*;
public class MyServer {
public static void main(String[] args){ try{
ServerSocket ss=new ServerSocket(6666); Socket s=ss.accept();//establishes connection
DataInputStream dis=new DataInputStream(s.getInputStream()); String
str=(String)dis.readUTF();
System.out.println("message= "+str); ss.close();
}catch(Exception e){System.out.println(e);}
}
}
```

ii. Code For Client Side :

```
import java.io.*; import java.net.*; public class MyClient {
public static void main(String[] args) { try{
Socket s=new Socket("localhost",6666);
DataOutputStream dout=new DataOutputStream(s.getOutputStream());
dout.writeUTF("Hello Server");
dout.flush();
dout.close();
s.close();
}catch(Exception e){System.out.println(e);}
}
}
```

Output:



B) Read And Write Method

i. Code For Server Side :

```
import java.net.*; import java.io.*; class MyServer{
public static void main(String args[])throws Exception{ ServerSocket ss=new
ServerSocket(3333);
Socket s=ss.accept();
DataInputStream din=new DataInputStream(s.getInputStream()); DataOutputStream
dout=new DataOutputStream(s.getOutputStream()); BufferedReader br=new
BufferedReader(new InputStreamReader(System.in));

String str="",str2=""; while(!str.equals("stop")){ str=din.readUTF();
System.out.println("client says: "+str); str2=br.readLine(); dout.writeUTF(str2);
dout.flush();
}
din.close();
s.close();
ss.close();
}}
```

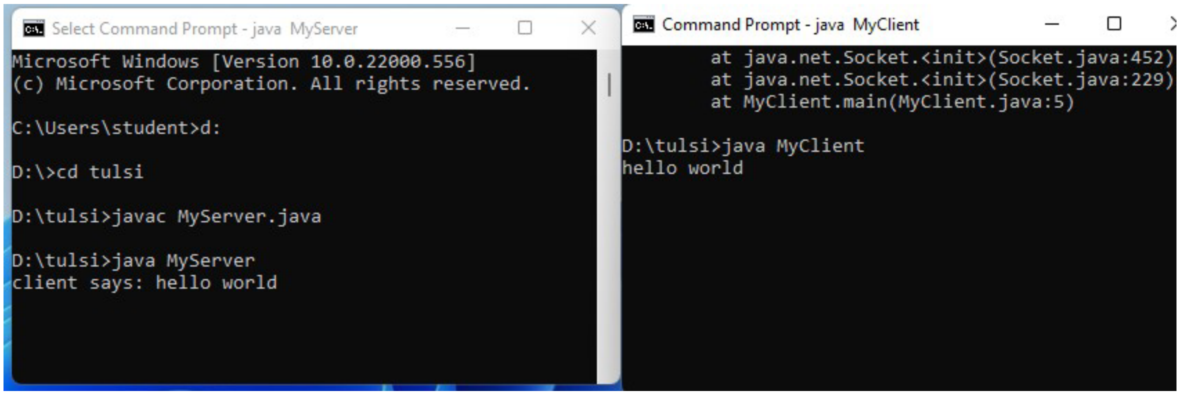
ii. Code For Client Side :

```
import java.net.*; import java.io.*; class MyClient{
public static void main(String args[])throws Exception{
Socket s=new Socket("localhost",3333);
DataInputStream din=new DataInputStream(s.getInputStream());
DataOutputStream dout=new DataOutputStream(s.getOutputStream()); BufferedReader
br=new BufferedReader(new InputStreamReader(System.in));

String str="",str2=""; while(!str.equals("stop")){ str=br.readLine(); dout.writeUTF(str);
dout.flush(); str2=din.readUTF();
System.out.println("Server says: "+str2);
}

dout.close();
s.close();
}}
```

Output:


Select Command Prompt - java MyServer
Microsoft Windows [Version 10.0.22000.556]
(c) Microsoft Corporation. All rights reserved.
C:\Users\student>d:
D:\>cd tulsi
D:\tulsi>javac MyServer.java
D:\tulsi>java MyServer
client says: hello world
Command Prompt - java MyClient
at java.net.Socket.<init>(Socket.java:452)
at java.net.Socket.<init>(Socket.java:229)
at MyClient.main(MyClient.java:5)
D:\tulsi>java MyClient
hello world

Practical-11

Aim: Case Study: Understanding of network design & components available at your institute.

Case Study: MBIT Networking

Introduction:

The MBIT College has 4 blocks namely A, B, C and D. A is the admin block where all servers and cyber security is present. B and D blocks have labs where the rack based servers are present for network connection to the PC. C block has no lab for network connections

Topology:

Topology is defined as the arrangement of devices in a network. In MBIT A block has the needed servers where the request for website is entertained. There are 2 access points, one near the A block and other between MOS lab and D block. B block has a rack based server which contains wired router and 2 switch panels and 2 patch panels. The same devices are present in the other B block and D block.

A-Block:

A block has the server room where all the connections are accessed of B-block and D-block. It has 4 servers which performs different functionalities. They are produced by AVIEW Company. They are of the type system*3400 IBM. They are standalone servers. The first server is the “db server”. Its function is to store filer. It is also called the domain control server. The second server is the Anti-Virus Server. It contains the software to protect the system such as Quick Heal Security. It protects the PC from viruses. The third is the WEB server. Any request of a URL from the clients is entertained here. It helps to access the different website for example Google etc. the fourth server is the IIT BOMBAY server. Any online discussions and seminars at IIT can be accessed through it.

Along with these four servers it also contains two identical switch racks. Each switch rack has a patch panel with 48 ports and a switch panel with 96switches. They are manufactured by the CISKO Company. The four servers are stand-alone servers. The first access point of wireless router is near the bridge.

B-Block:

In B-BLOCK we have a rack based server which is in B-006 lab. It has a horizontal server named XP Pro-Liant DL 180G6 which is produced by CISCO Company. It has two (2) patch panel and two (2) with panels of 48 ports and 96 switch respectively. It also has a wired router which is produced by CISCO OF 2800 series.

C-Block:

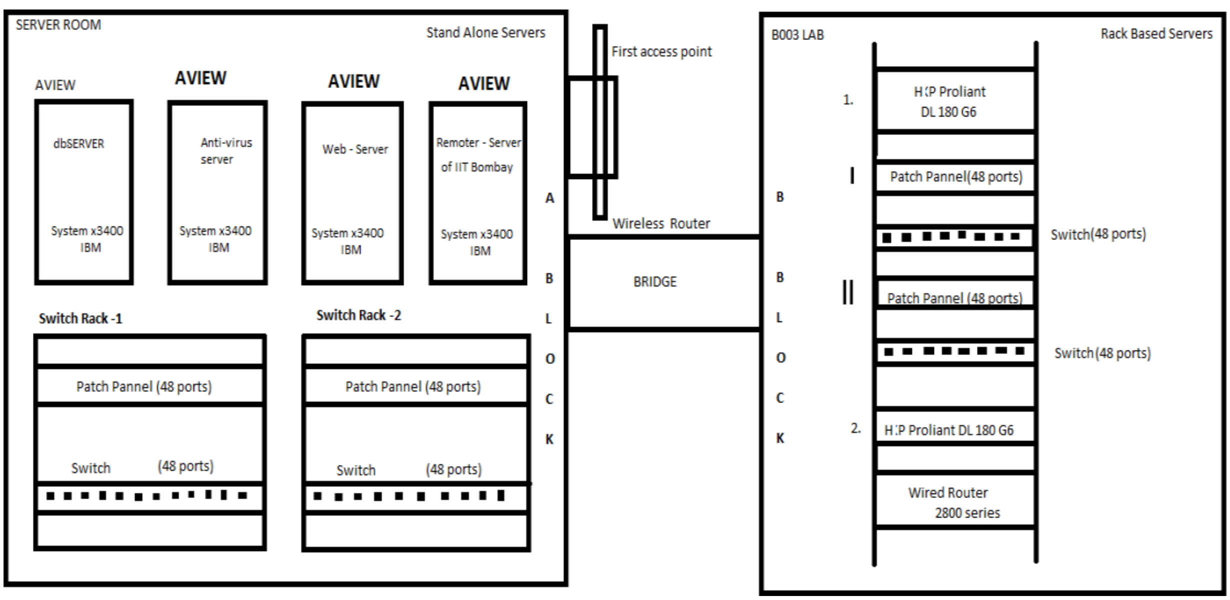
The C-BLOCK does not have any network connections.

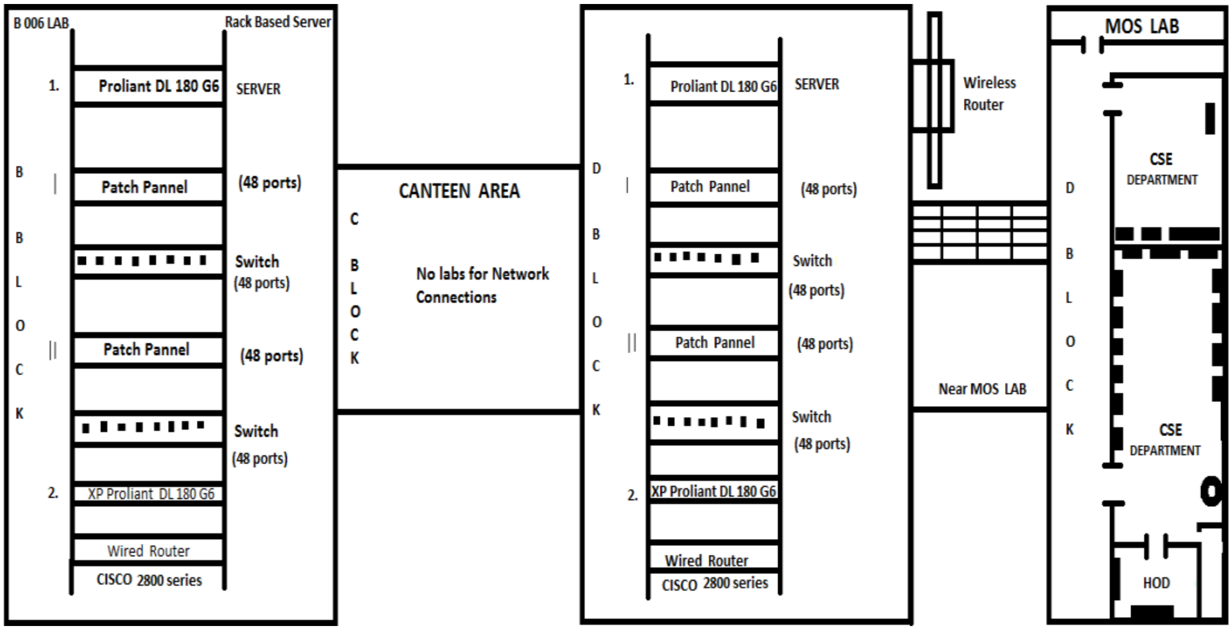
D-Block:

A wireless router i.e. second access point is present between the MOS lab and D-BLOCK.

Common Structure:

The number of ports in the patch panel defines the possible number of PC's which can be connected. There is a wired connected between the patch and the switch panel. And the switch panel connects it to the server room via a common switch. All these switches from the different labs go to the server room. The blinking light panel is called the switch panel. The cabling in the computer rooms of all the blocks is structured cabling.




B 006 LAB
Rack Based Server
1.
Proliant DL 180 G6
SERVER
Patch Pannel
(48 ports)
Switch
(48 ports)
Patch Pannel
(48 ports)
Switch
(48 ports)
2.
XP Proliant DL 180 G6
Wired Router
CISCO 2800 series
B BLOCK
CANTEEN AREA
C BLOCK
No labs for Network Connections
D BLOCK
1.
Proliant DL 180 G6
SERVER
Patch Pannel
(48 ports)
Switch
(48 ports)
Patch Pannel
(48 ports)
Switch
(48 ports)
2.
XP Proliant DL 180 G6
Wired Router
CISCO 2800 series
Wireless Router
Near MOS LAB
MOS LAB
D BLOCK
CSE DEPARTMENT
CSE DEPARTMENT
HOD