LAB # 01

NAME *:ABDULLAH ZUNORAIN*

REG_NUMBER *: 19JZELE0338*

SECTION *: A*

SUBJECT *: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO *: SYED UZAIR GILLANI*

DEPT : *ELECTRICAL COMMUNICATION*

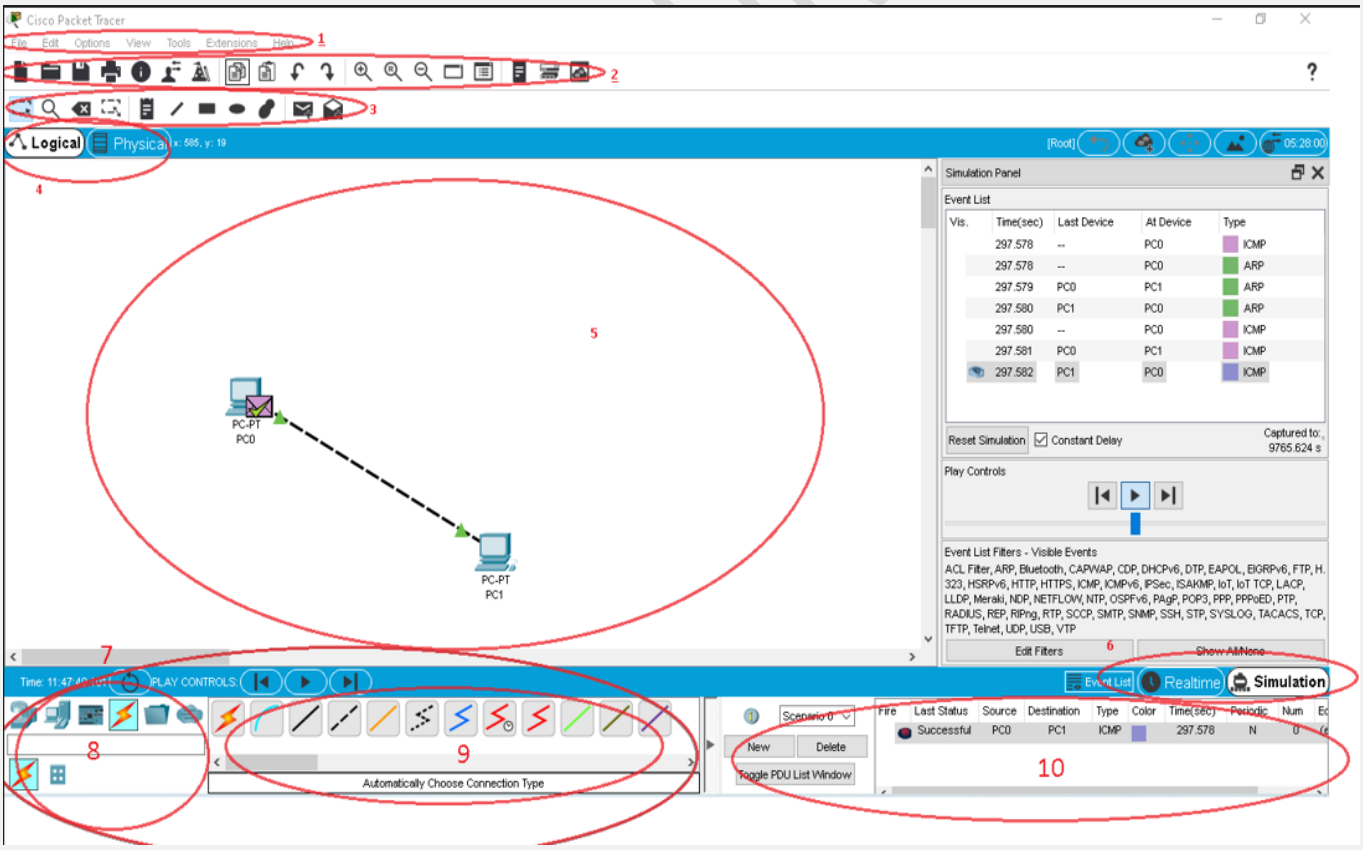
TITLE: INTRODUCTION TO PACKET TRACER

LAB OBJECTIVE:

- Introduction to "Cisco Packet Tracer Software"
- Creating Devices in this software
- Adding Modules
- Making Connections
- Creating Networks

Lab Content:

a) Cisco Packet Tracer Overview:



1. **Menu Bar:**

This bar provides the **File, Edit, Options, View, Tools, Extensions,** and **Help** menus. You will find basic commands such as **Open, Save, Print,** and **Preferences** in these menus. You will also be able to access the **Activity Wizard** from the **Extensions** menu.

2. **Main Tool Bar:**

This bar provides shortcut icons to the **File** and **Edit** menu commands. This bar also provides buttons for **Zoom**, the **drawing Palette**, and the **Device Template Manager**. On the right, you will also find the **Network Information** button, which you can use to enter a description for the current network (or any text you wish to include).

3. **Common Tools Bar:**

This bar provides access to these commonly used **workspace tools: Select, Move Layout, Place Note, Delete, Inspect, Add Simple PDU,** and **Add Complex PDU.** See "Workspace Basics" for more information.

4. **Logical/Physical Workspace and Navigation Bar:**

You can toggle between the Physical Workspace and the Logical Workspace with the tabs on this bar. In Logical Workspace, this bar also allows you to navigate through levels of a cluster, create a **new New Cluster, Move Object, Set Tiled Background,** and **Viewport**. In Physical Workspace, this bar allows you to navigate through physical locations, create a **New City**, create a **New Building**, create a **New Closet**, **Move Object**, apply **Grid** to the background, **Set Background**, and go to the **Working Closet**.

5. **Workspace:**

This area is where you will create your **network**, **watch simulations**, and view many kinds of **information and statistics** .

6. Realtime/Simulation Bar:

You can toggle between Realtime Mode and Simulation Mode with the tabs on this bar. This bar also provides buttons to **Power Cycle Devices** as well as the **Play Control** buttons and the **Event List** toggle button in Simulation Mode. Also, it contains a clock that displays the **relative Time** in Realtime Mode and Simulation mode.

7. Network Component Box:

This box is where you choose devices and connections to put into the workspace. It contains the **Device-Type Selection Box** and the **Device-Specific Selection Box**.

8. Device-Type Selection:

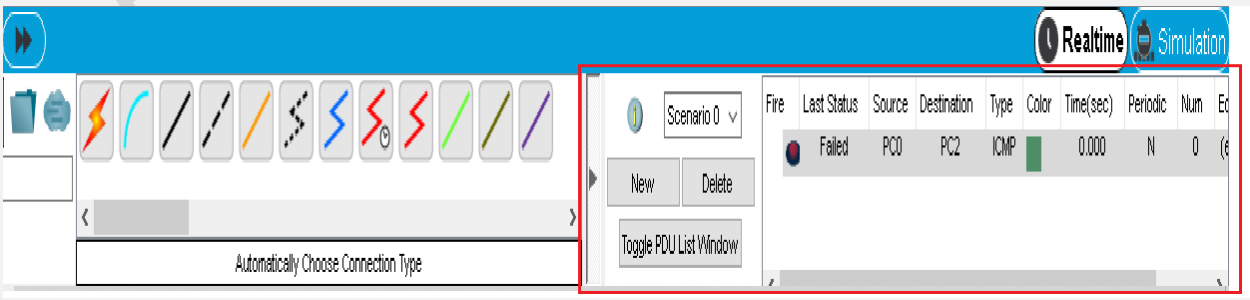
This box contains the type of devices and connections available in Packet Tracer. The **Device-Specific Selection Box** will change depending on which type of device you choose.

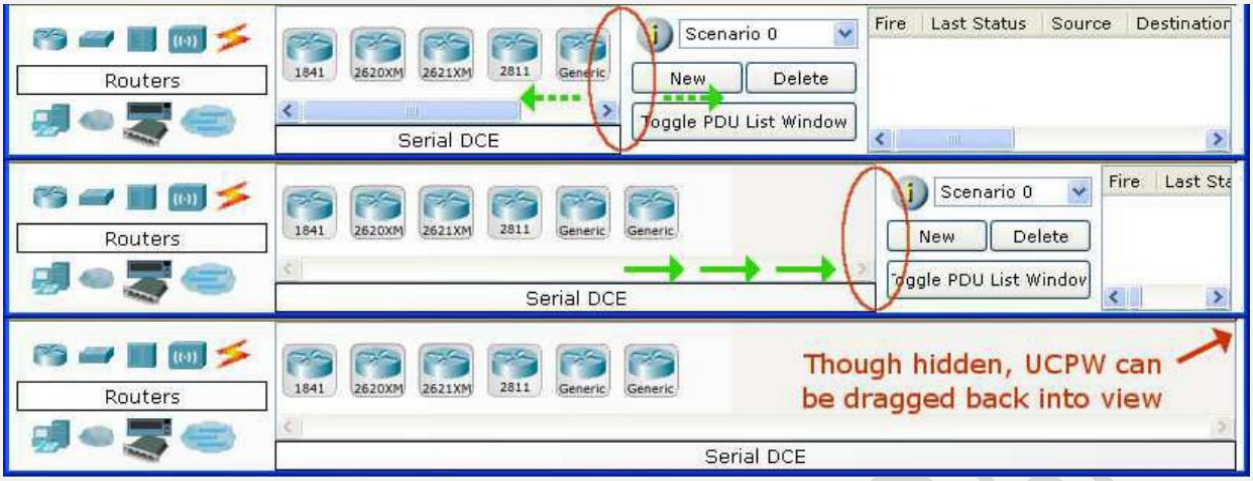
9. BoxDevice-Specific Selection Box:

This box is where you choose specifically which devices you want to put in your network and which connections to make.

10. User Created Packet Window* :

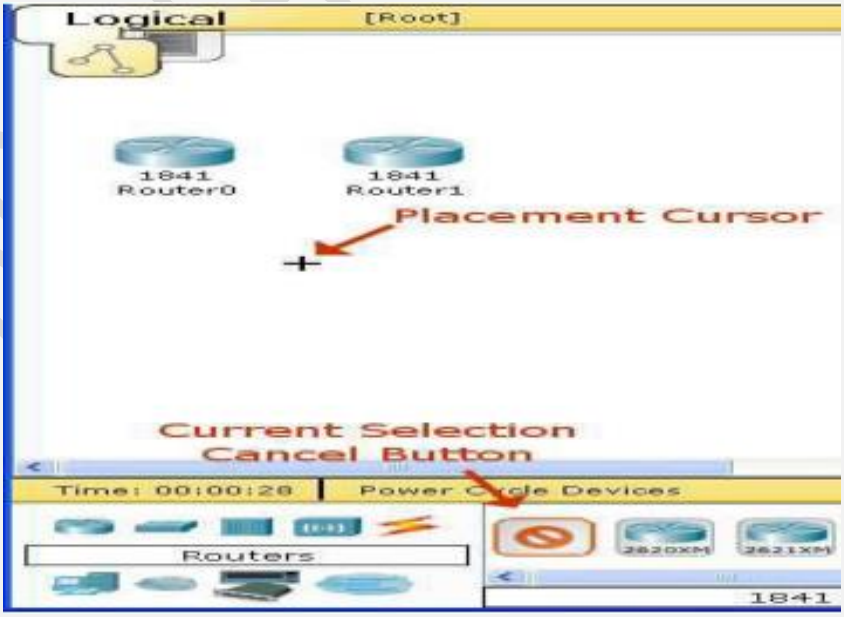
This window manages the packets you put in the network during simulation scenarios. See the "Simulation Mode" section for more details.





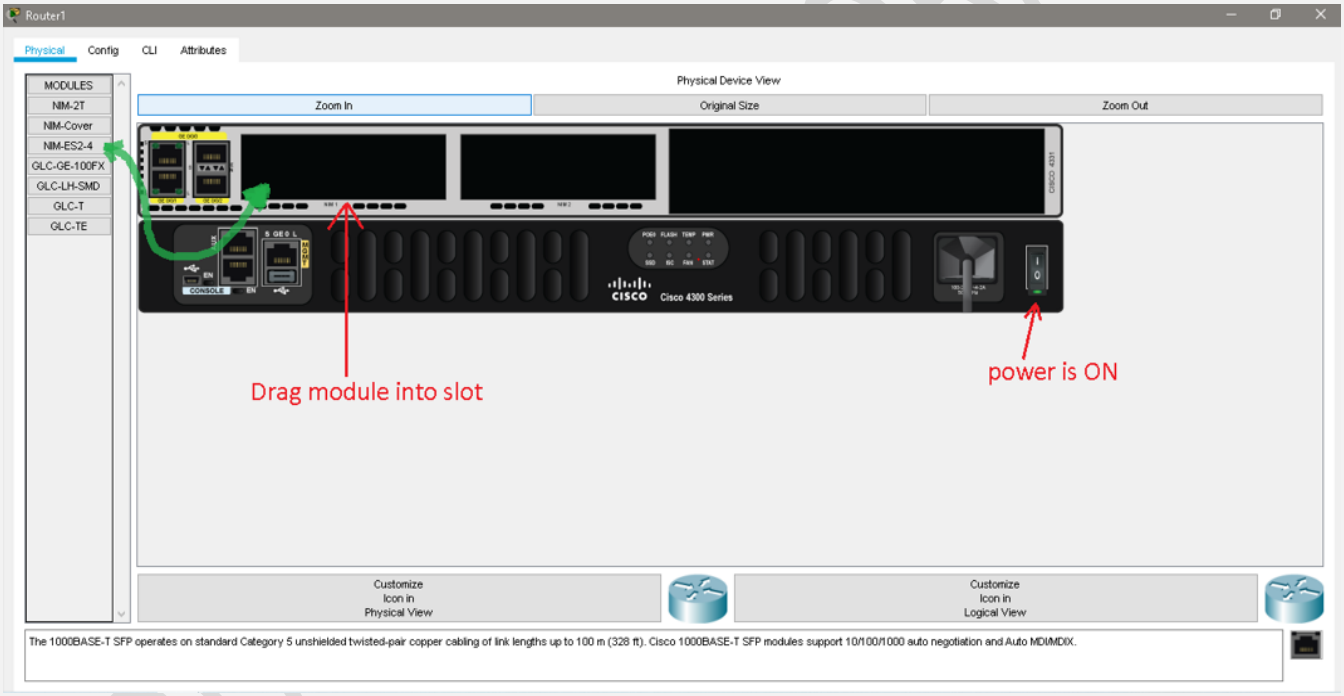
b) Creating Devices:

- Choose a device type from the Device- Type Selection box
- Click on the desired device model from the Device-Specific Selection box.
- Click on a location in the workspace to put your device in that location.
- If you want to cancel your selection, press the Cancel icon for that device.
- Alternatively, you can click and drag a device from the Device-Specific Selection box onto the workspace.
- You can also click and drag a device directly from the Device-Type Selection box and a default device model will be chosen.



c) Adding Modules:

- Click on a device to bring up its configuration window.
- By default, you will be in the Physical Device View subpanel of the device.
- You can browse (by clicking) through the list of modules and read their description in the information box at the bottom.
- When you have found the module you want to add, simply drag it from the list into a compatible bay on the device picture.
- You can remove a module by dragging it from the device back into the list.



d) Making Connections:

- To make a connection between two devices, first click the Connections icon from the Device-Type Selection box to bring up the list of available connections.
- Then click the appropriate cable type.
- The mouse pointer will change into a "connection" cursor.
- Click on the first device and choose an appropriate interface to which to connect.
- Then click on the second device and do the same.

- A connection cable will appear between the two devices, along with link lights showing the link status on each end (for interfaces that have link lights).

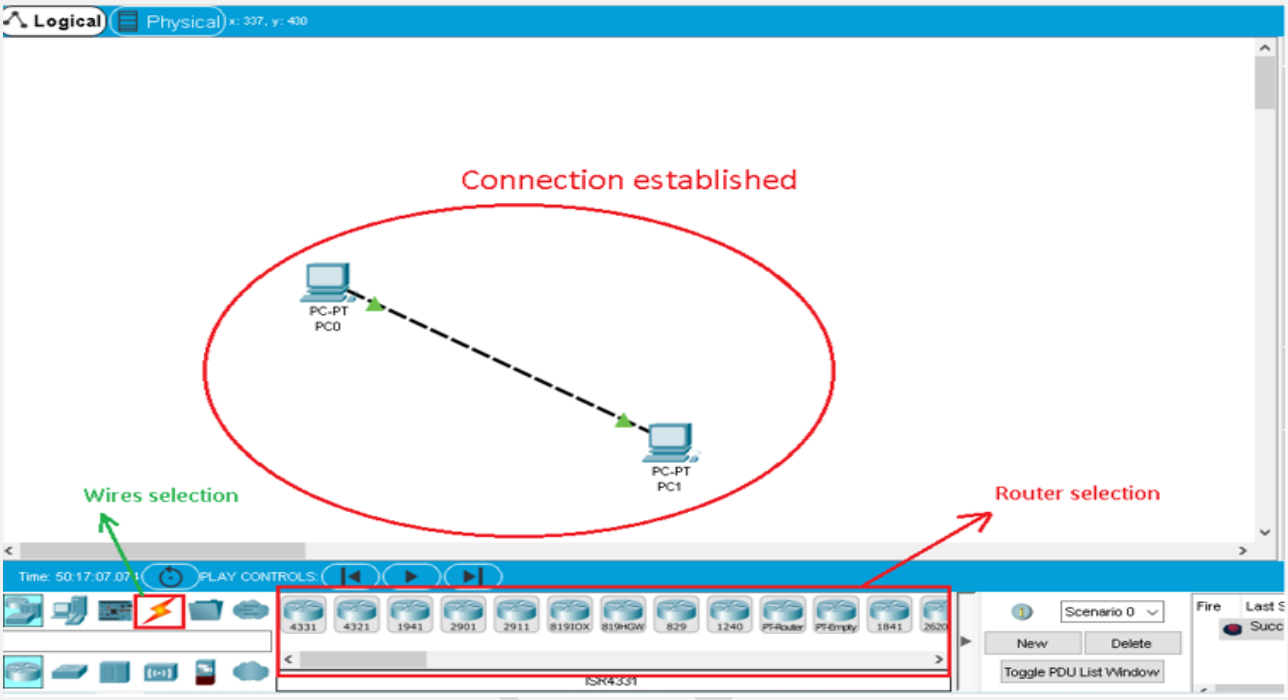


FIG-A

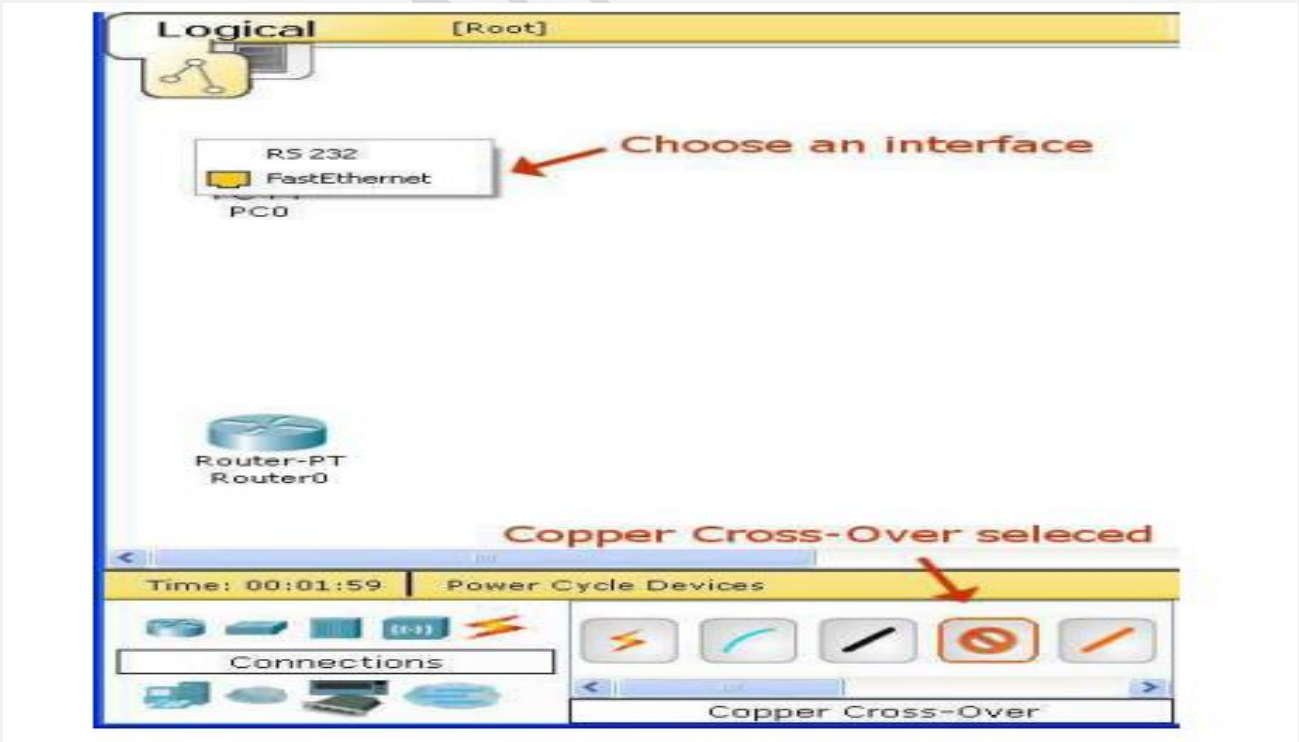


FIG-B

e) Creating Networks:

- Start creating a network by first selecting the End Devices. Add a Generic PC and a Generic Server to the workspace.

- Under Connections, select the Copper Straight-through cable (solid black line) and connect the devices with it. The red lights on the link indicate that the connection is not working. Now, use the Delete tool to remove the Copper Straight-through cable, and use a Copper Cross-over cable (dashed line) instead. The lights should turn green at this point, and if the mouse pointer is held over either the PC or the Server, the link status will be shown as "Up." The network should look similar to the picture below.

- Click on the PC. While paying attention to the link lights, turn the power on, off, and on again. Follow the same steps for the server. The link lights turn red when the device is off. This means that the link is down or is not working. The link lights turn green when the device is turned back on.

- Try all three ways to learn about the devices. First, mouse over the devices to see basic configuration information about them. Second, click on each device with the Select tool to show the device configuration window, which provides several ways to configure the device. Third, use the Inspect tool to view tables the network device will build as it learns about the network around it. In this example, only the ARP tables will appear. Since the devices have not been configured yet, the ARP tables are empty. Always remember to close windows after viewing them or they will clutter the workspace.

- Open the PC configuration window and change the settings using the Config tab. Change the display name to Client and set the DNS server to 192.168.0.105. Under Interface, click FastEthernet and set the IP address as 192.168.0.110. Packet Tracer automatically calculates other parameters. Make sure that the Port Status box is checked. For future reference, note that other Ethernet interface settings, such as bandwidth, duplex, MAC address, and subnet mask can be modified using this window.

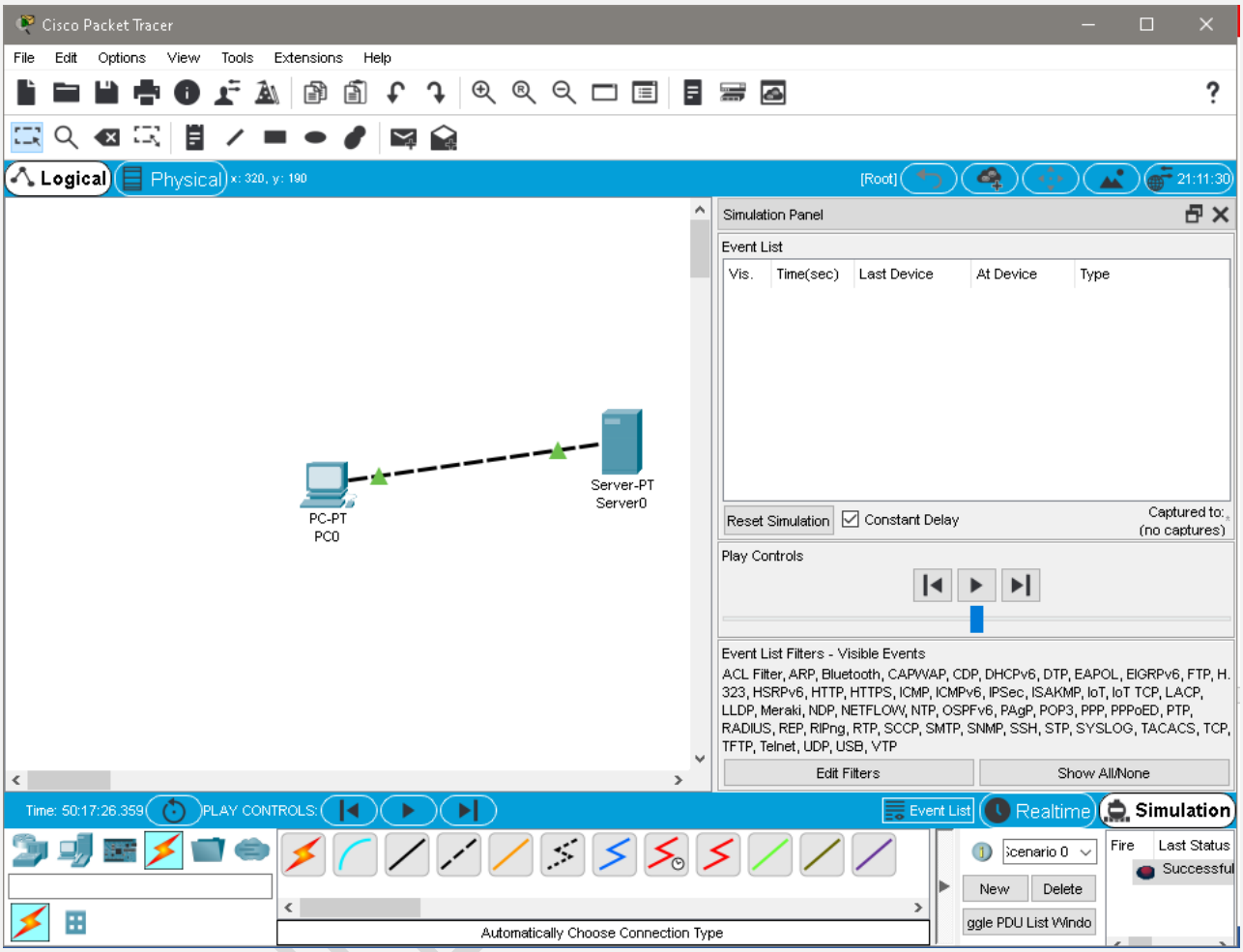
- Go to the Desktop Tab and click on IP Configuration. Notice that the IP address, subnet mask and DNS server can be changed here as well.

- Open the Server configuration window and go to the Config tab. Change the display name to Web Server. Click FastEthernet and set the IP address as 192.168.0.105. Make sure that the Port Status is also on. Click DNS and set the domain name as www.firstlab.com. Set the IP address as 192.168.0.105 and click Add. Finally, check to make sure that the service for DNS is on.

- Reposition the network devices by dragging them to a new location. Add a network description by using the “i” button on the upper right corner. Then add some text labels within the Logical Workspace by using the Place Note tool.

- Load a background grid using the Set Tiled Background button.

- Save your work using the File > Save As option and create a meaningful filename.



LAB # 02

NAME : ABDULLAH ZUNORAIN

REG NO: 19JZELE0338

SUBJECT: COMPUTER COMMUNICATION NETWORKING

SUBMITTED TO: SYED UZAIR GILLANI

SECTION: A

DEPT: ELECTRICAL COMMUNICATION

TITLE: DETAIL OVERVIEW OF "PEER TO PEER NETWORK" IN PACKET TRACER.

OBJECTIVES:

- The main objective is to know about that how peer to peer Network Connection has been established in Packet Tracer.
- Create a simple peer-to-peer network between two PCs.
- Identify the proper cable to connect the two PCs.
- Configure workstation IP address information.
- Test connectivity using the ping command.

COMPONENTS:

- TWO PC'S
- CROSS-OVER CABLE
- NETWORK INTERFACE CARD

Background / Preparation:

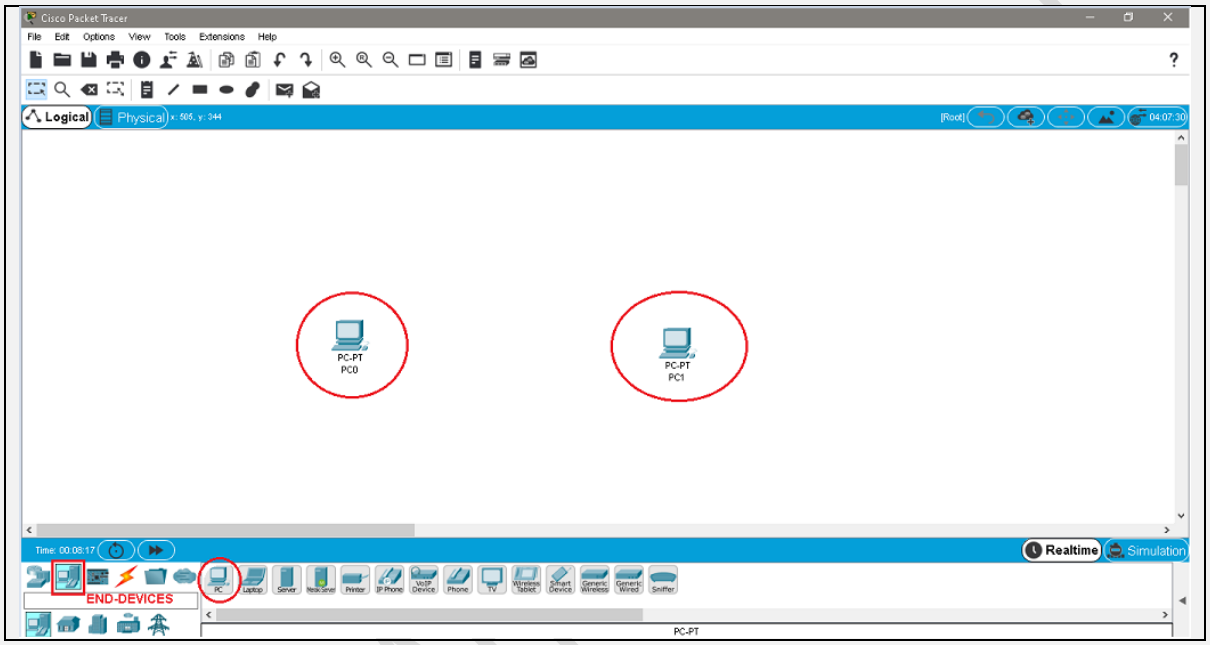
- In Peer to Peer architecture every node is connected to other node directly for exchanging information instead of connected to central server.
- Every computer node is referred as peer and they do the job of client as well as server both.
- Every peer provides services to other peers as well as uses services provided by other peers.

In this lab we focuses on the ability to connect two PCs to create a simple peer-to-peer Ethernet LAN between two workstations. The workstations will be directly connected to each other without using a hub or switch. In addition to the Layer 1 physical and Layer 2 data link connections, the computers must also be configured with the correct IP network settings, which is Layer 3, so that they can communicate. A crossover cable is the same type that would be used as backbone or vertical cabling to connect switches together. Connecting the PCs in this manner can be very useful for transferring files at high speed and for troubleshooting interconnecting devices between PCs.

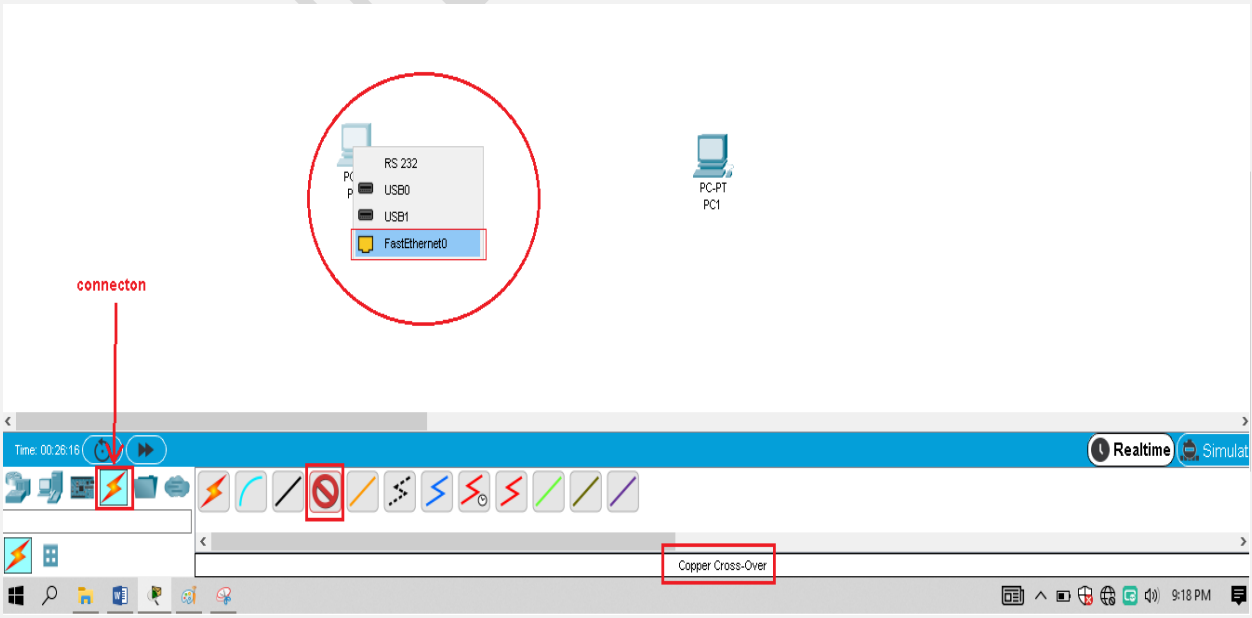
REAL-TIME(MODE) AND SIMULATION(MODE) PROCEDURE:

a) Real time:

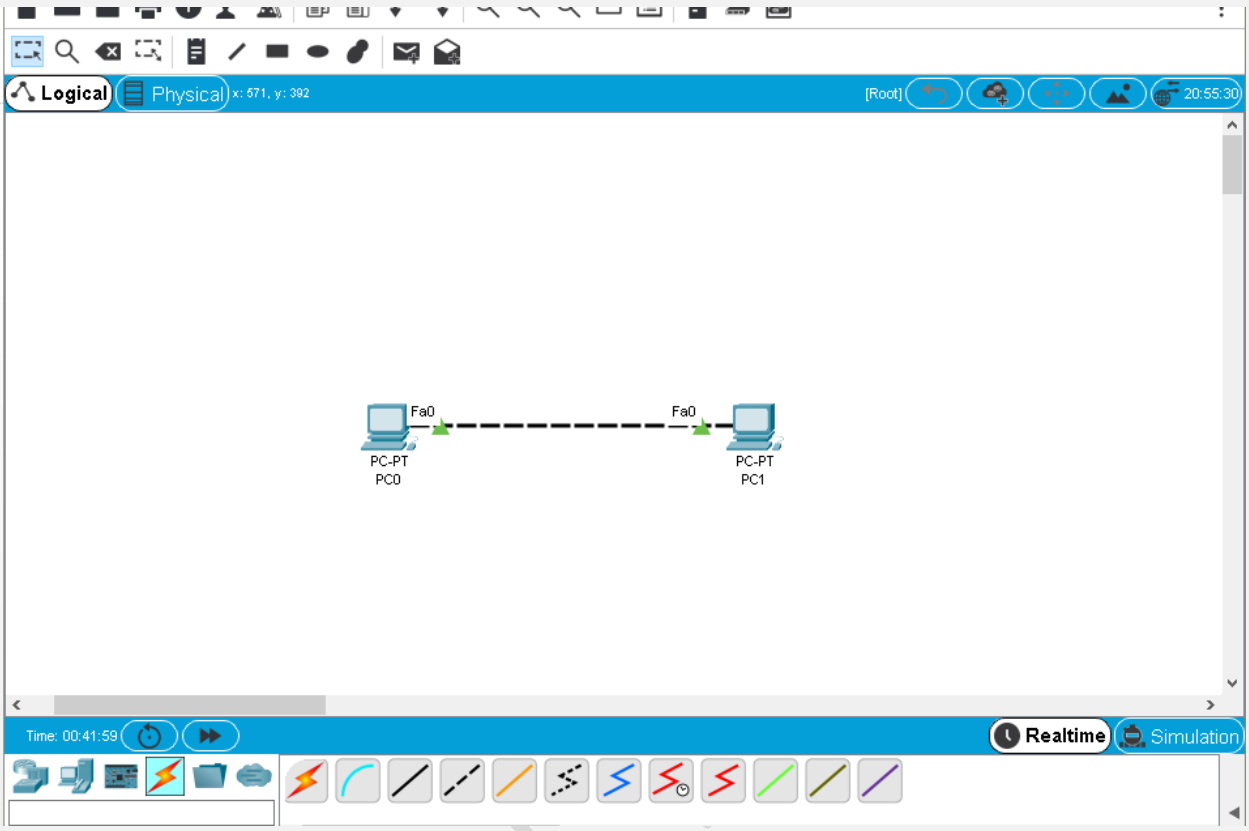
Open software ***cisco packet tracer***.Click ***End Devices*** icon (lower left corner) or press ***CTRL + ALT + V***,then drag ***PC_PT*** (Personal Computer) and drop to ***workspace*** as shown in below screenshot.



Click ***Connections*** icon or press **CTRL + ALT + 0** , then select ***Copper Cross-Over*** cable or ***Authomatically Choose Connection Type*** cable to connect both PC's using ***FastEthernet0*** as shown below;

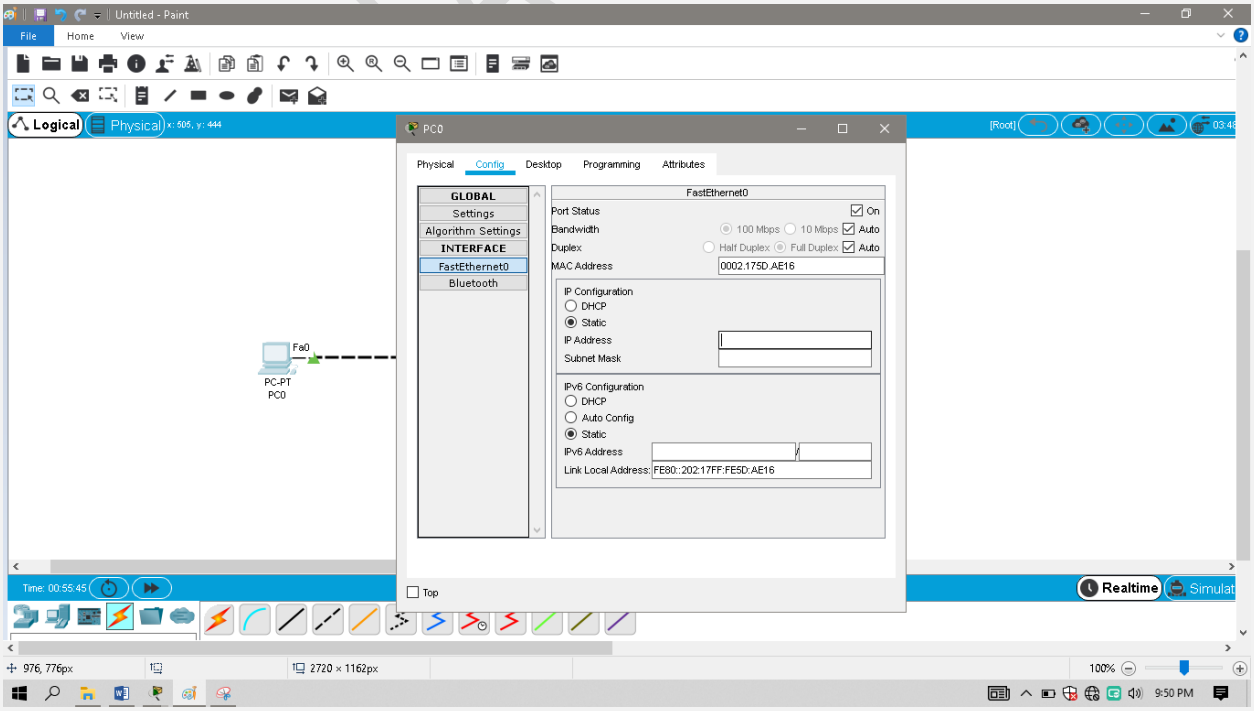


Now the connection is established as the **Green Dots** shown in the pic below;

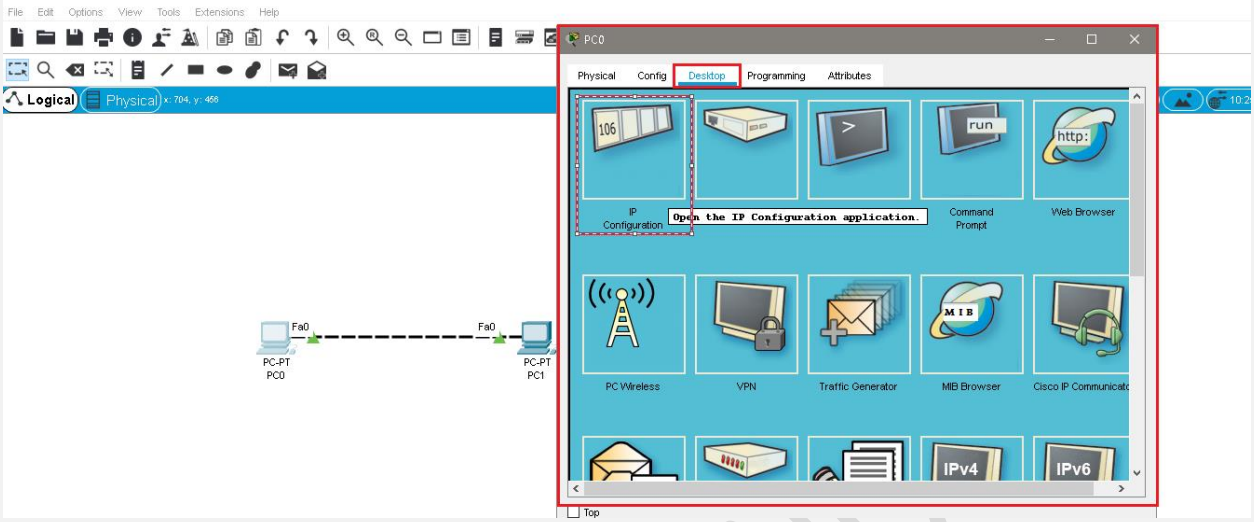


Now if we want to *ping* **PC-1** from **PC-0** we have to first assign the IP-addresses to both ***PC's.***

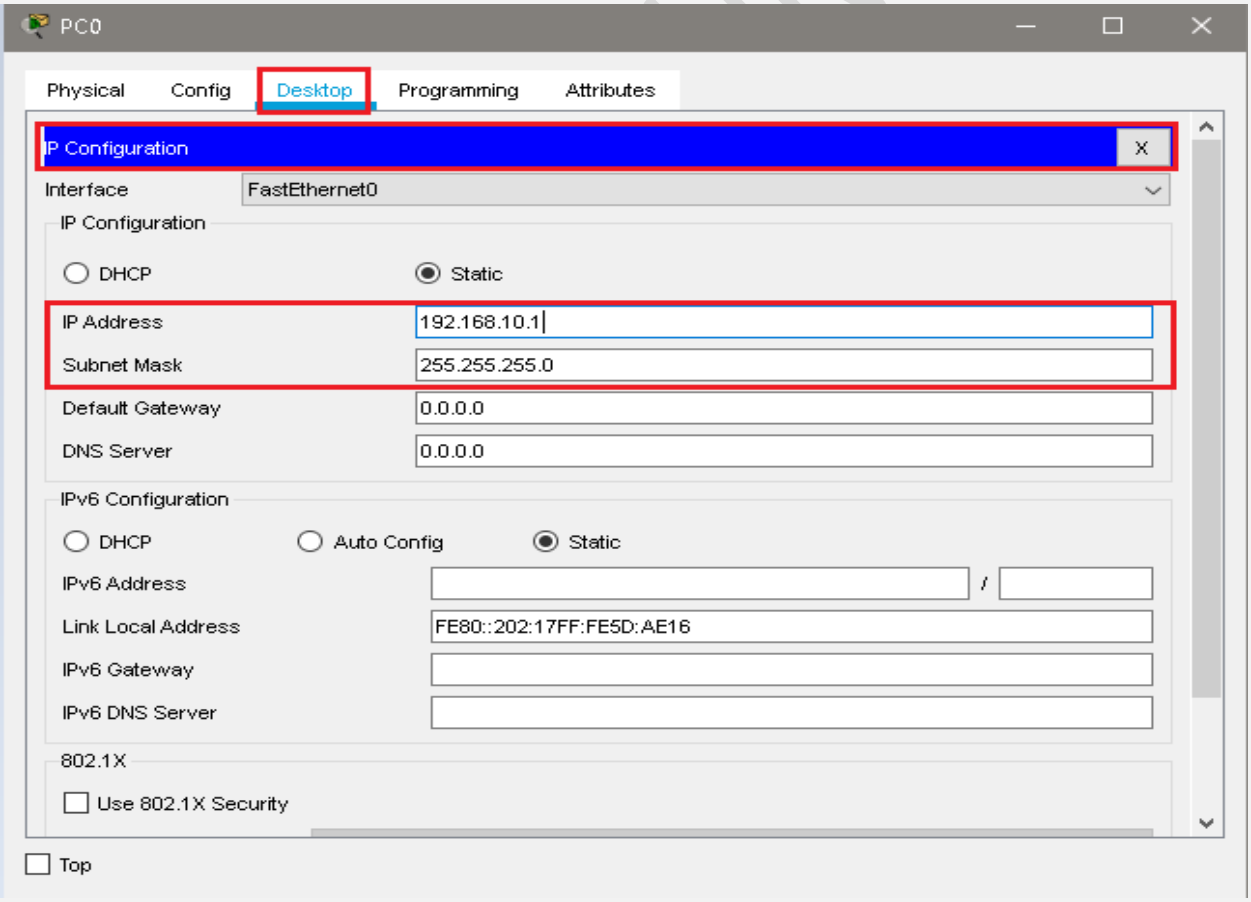
For this we have to double-click on the PC ,a new window will be open as shown below;



Then in new window we have to select the **Desktop,** at desktop we have to select **IP-configuration Box** as shown below;

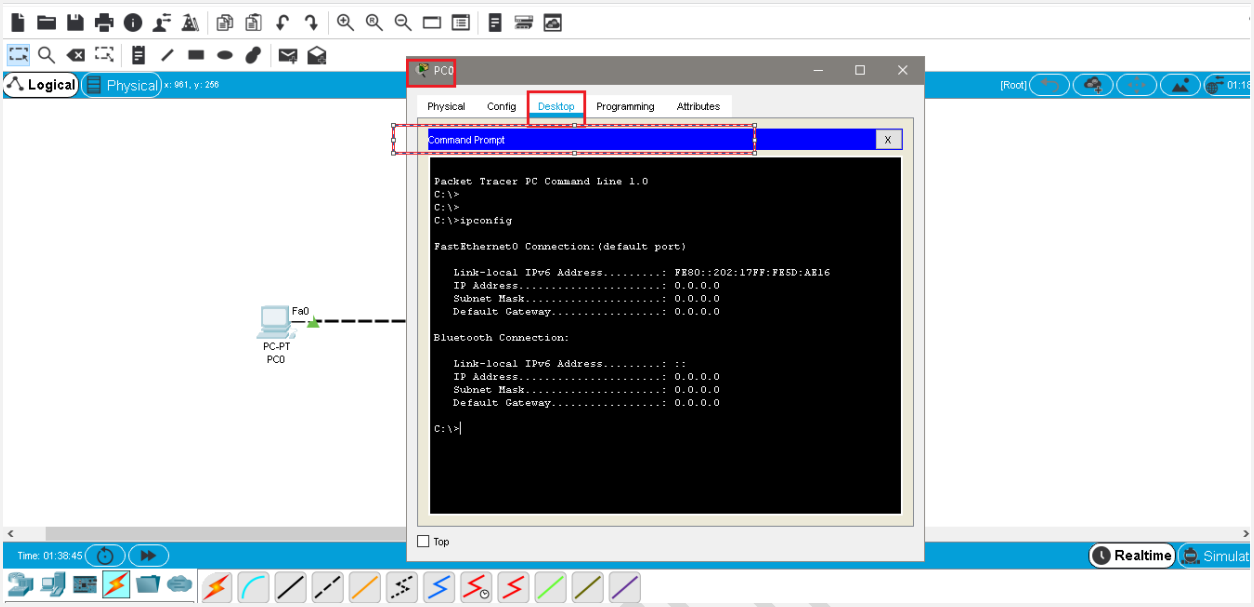


Then select the **IP-configuration-Box** and assign an **IP-adress** as shown below;



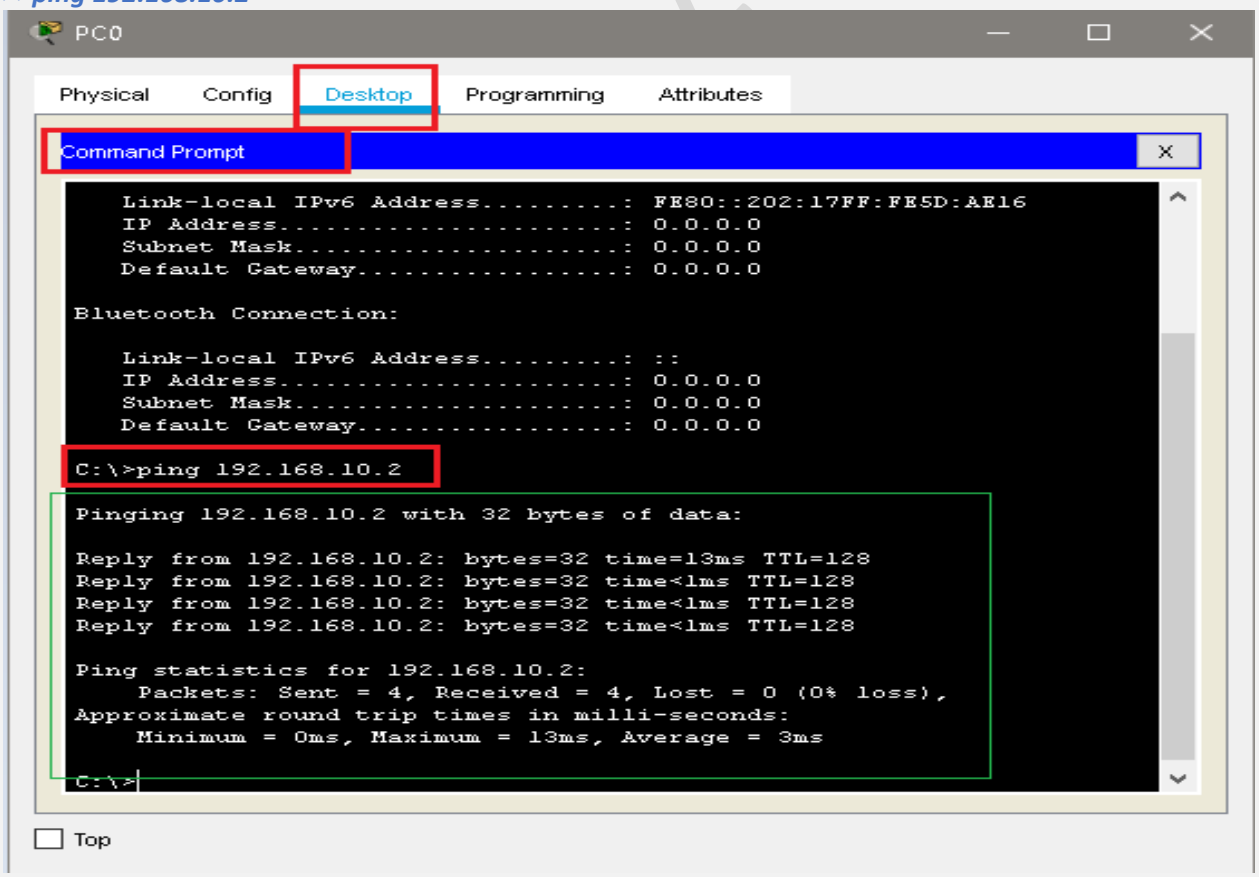
- *IP-Address PC0 = 192.168.10.1 , Subnet Mask = 255.255.255.0*
- *IP-Address PC1 = 192.168.10.2, Subnet Mask = 255.255.255.0*

Now to ping **PC1** from **PC0** we have to first *double-click* on the PC , then go to **Desktop** then select the **Command-Prompt** as shown in the below screenshot;



Now we have to use the **Ping command** to ping the **PC1** from **PC0** as shown below;

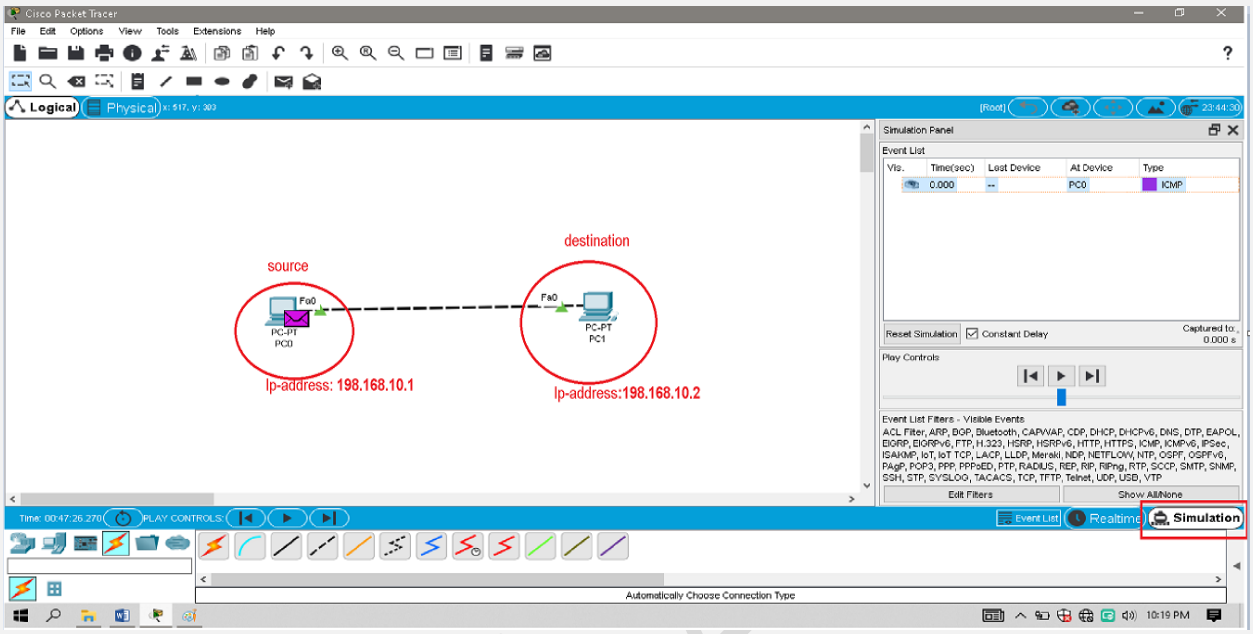
>>ping 192.168.10.2



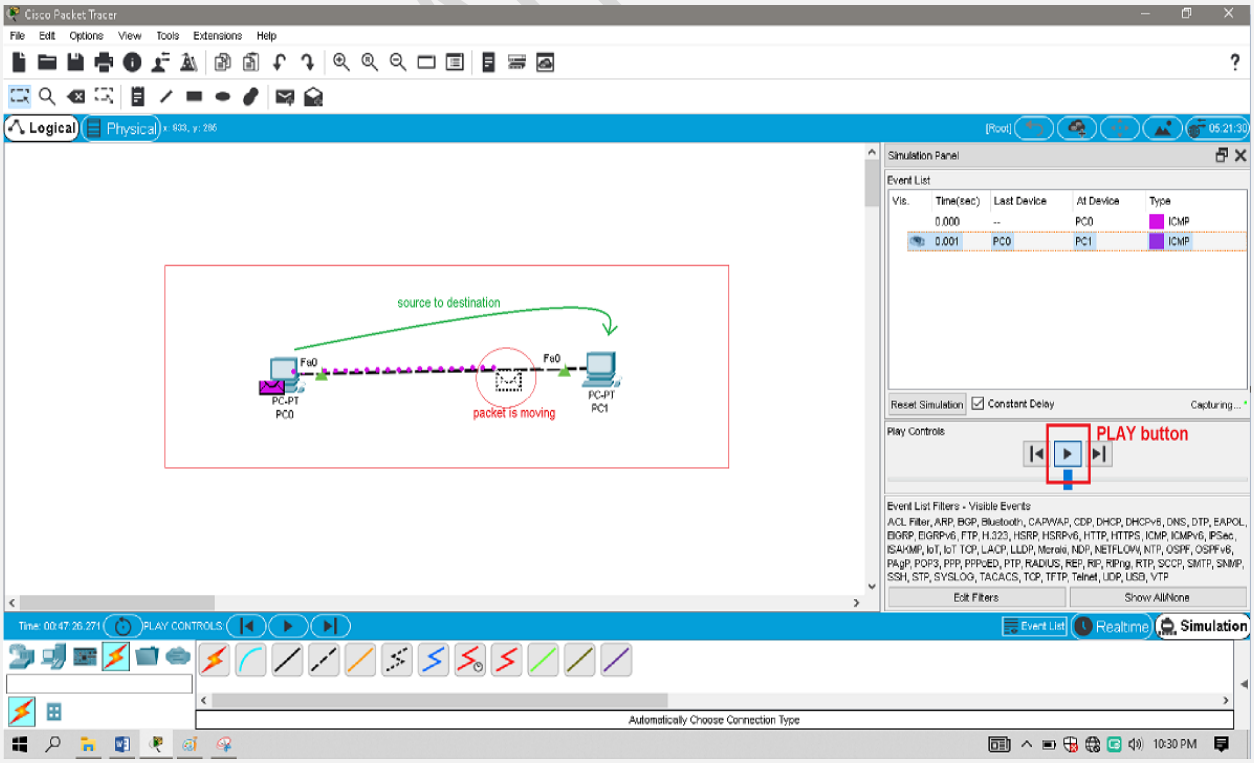
b) Simulation mode:

First of all we have to drag two PC as we had done above then select the **simulation-mode** at the right corner as I mentioned it by red square box.

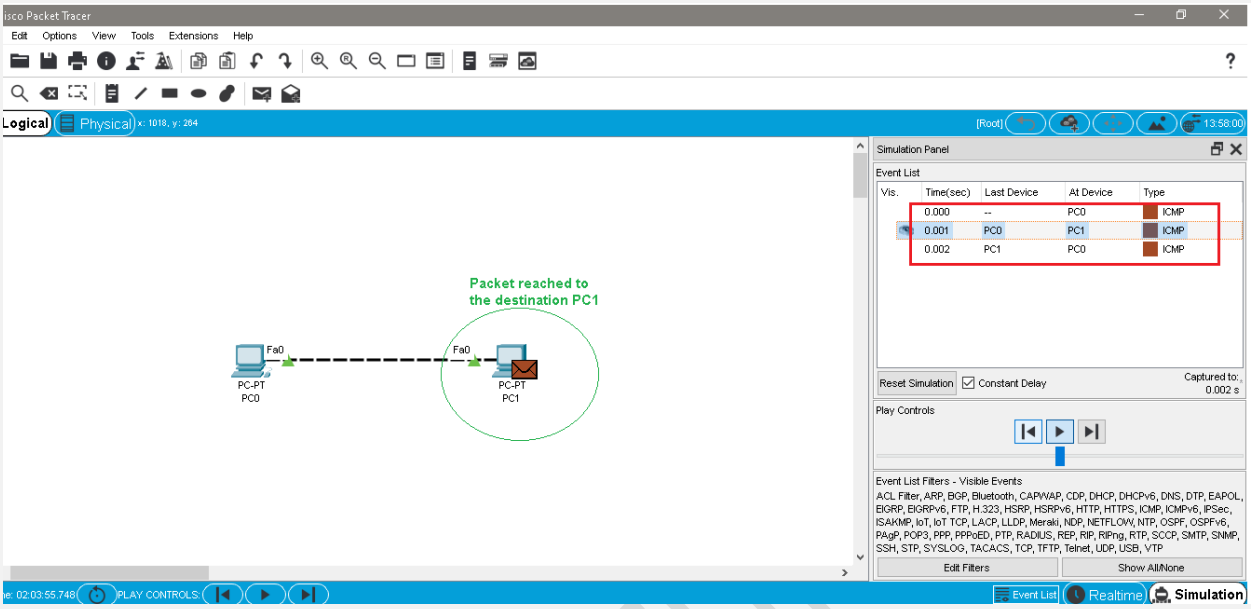
Then drag **simple PDU** and 1st paste at **source PC** and then paste on *destination PC* as shown in the screenshot below;



Now we will **play** the simulation, so the simple PDU packet will start going from source to destination as shown below in the screenshot.



In the below pic, we clearly saw the *PDU packet*, that it is reached to the destination.



In the above screenshot we also saw that when the packet is moving from the PC1 to PC0, it uses **ICMP(Internet Control Message Protocol) protocol.**

ICMP PROTOCOL:

- *It Is a Network-Layer Protocol used by network devices to communicate.*
- A ping command sends an **ICMP** echo request to the target host, and the target host responds with an echo reply.

LAB # 03

NAME *: ABDULLAH ZUNORAIN*

REG_NUMBER *: 19JZELE0338*

SECTION *: A*

DEPT: *ELECTRICAL COMMUNICATION*

SUBJECT *: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO *: SYED UZAIR GILLANI*

CAMPUS : *JALOZAI*

TITLE: A DETAIL OVERVIEW OF A HUB AND A SWITCH IN PACKET TRACER

LAB OBJECTIVE:

- To get familiar with hubs and switches.
- To know how to connect hubs and switches with pc's or other devices.
- To know about the Data sharing through hub and swithes.

LAB SUMMARY:

This lab is based on how to connect switches and hubs to the number of PC's, Laptop or other devices and will observe the transmission of the Data from one PC to another through Hub and switches.

THEORY:

Cisco Packet Tracer:

It is an innovative and powerful networking simulation tool used for practice, discovery and troubleshooting.

We used Cisco Packet Tracer, which is online publically available software, for simulation of the project.

Devices Used:

- Hub-PT
- 2960 network switches
- Three PC-PT
- Copper Straight-Through cables

A. PROCEDURE FOR CONNECTING HUB:

a) Realtime Mode:

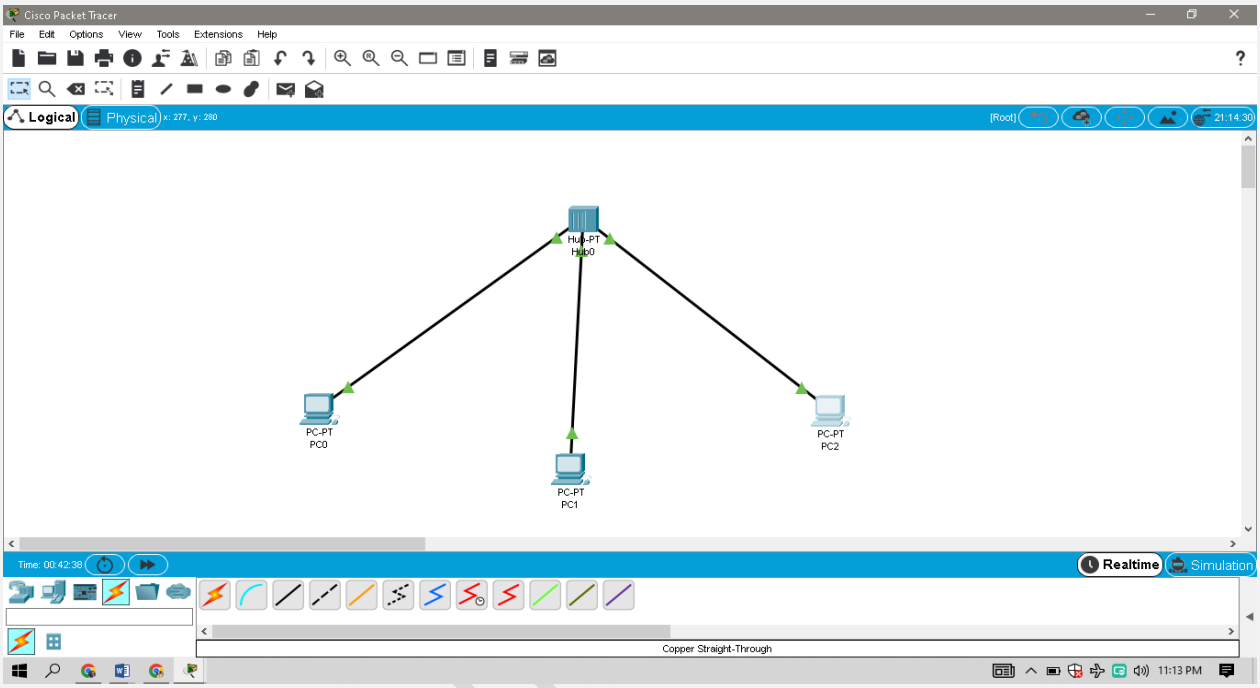
In Cisco Packet Tracer, multiple computers can communicate through a hub.

Firstly we have to open Cisco packet tracer,then we have to select the Hub from the device Type selection.

Then we have to select the **Hub0** from **BoxDevice-Specific Selection-Box** and drag it to **workspace.**

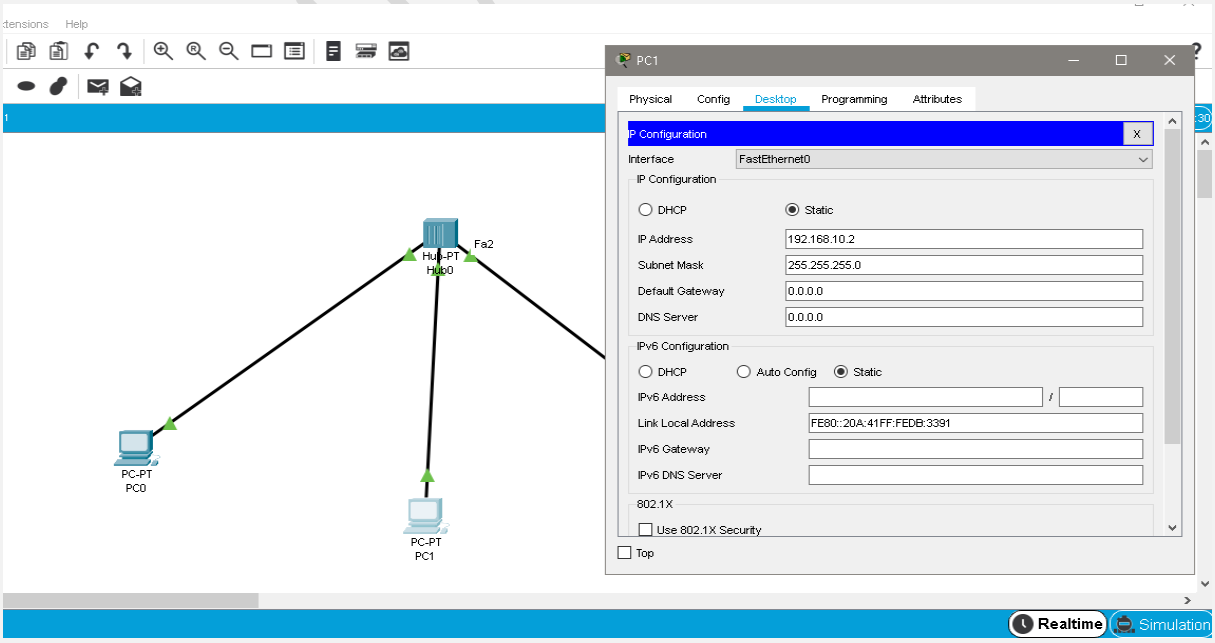
Now from **End-devices** we have to select **PC's**(PC0,PC1,PC2,etc) and then drag it to the **workspace**.

Then we have to connect these **PC's** to the **hub** through **Copper Straight-through** cable using **FastEthernet** ports as shown in the Fig below.



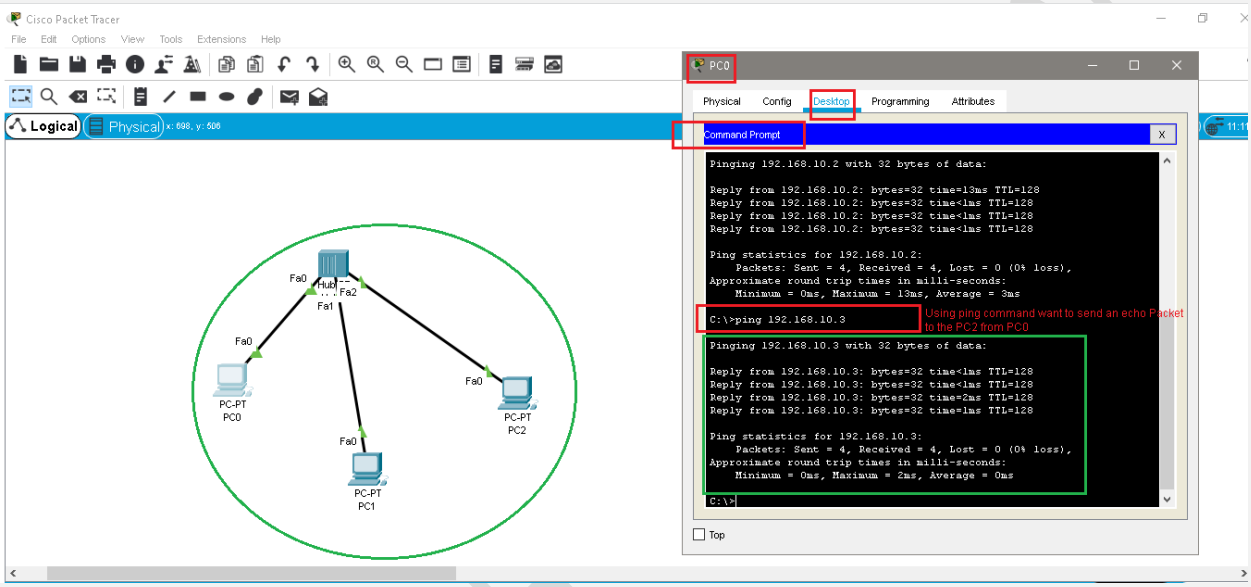
- **Assigning IP-addresses:**

Now we have to assign **IP addresses** to these PC's one by one as shown below;



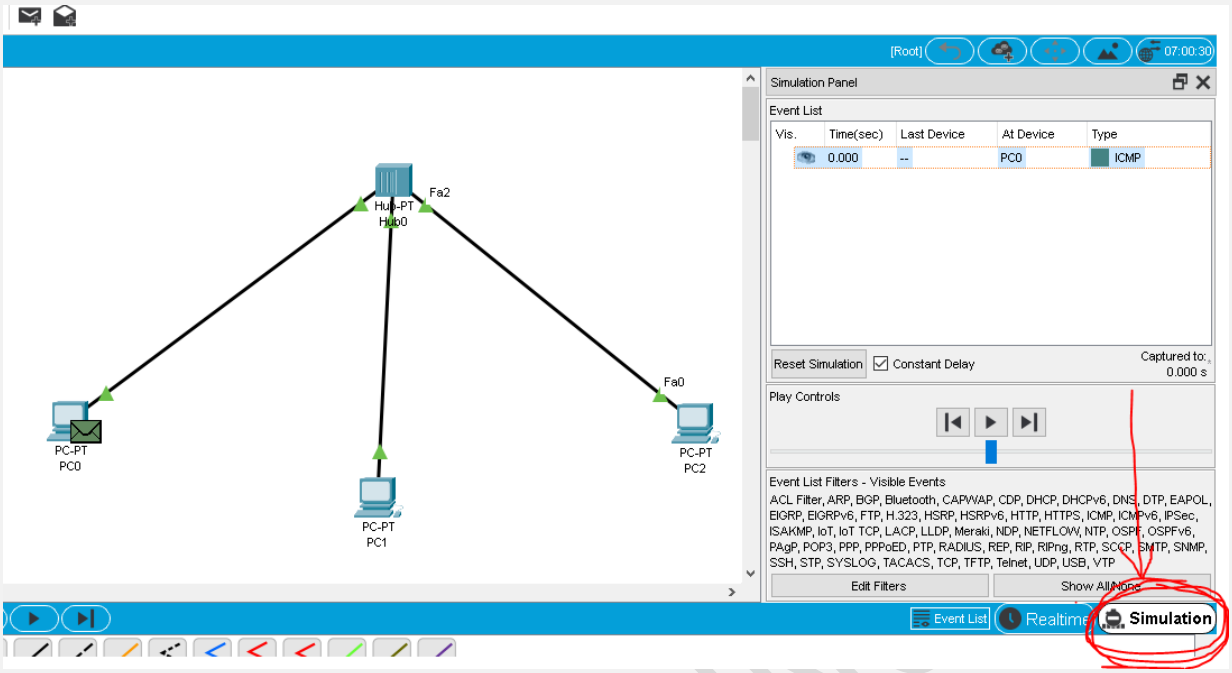
As in the above fig's it is clearly shown that there are **two Green dots** on both ends of the cable so it means that the connection is successfully established.

- Now to **ping** PC2 from PC0, we have to first *double-click* on the PC then a new window will be open as shown below , then we will click on **Desktop** and then select the **Command prompt,** at command prompt we will use ping command to ping other PC's that are conneted to that same **Hub** as shown below;

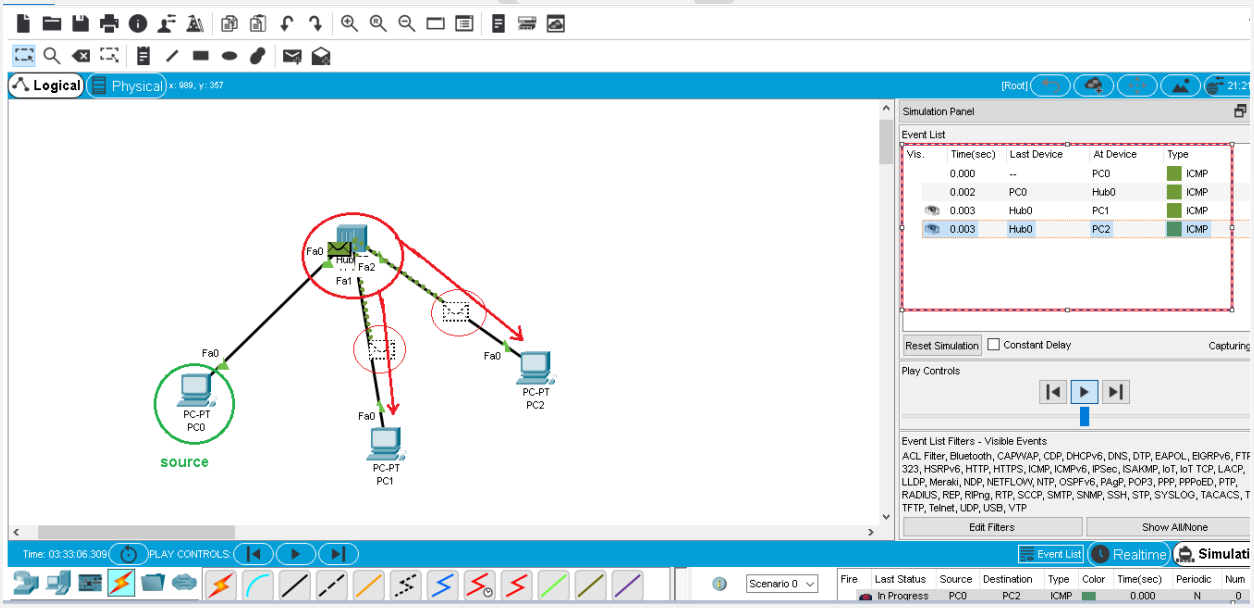


b) Simulation Mode:

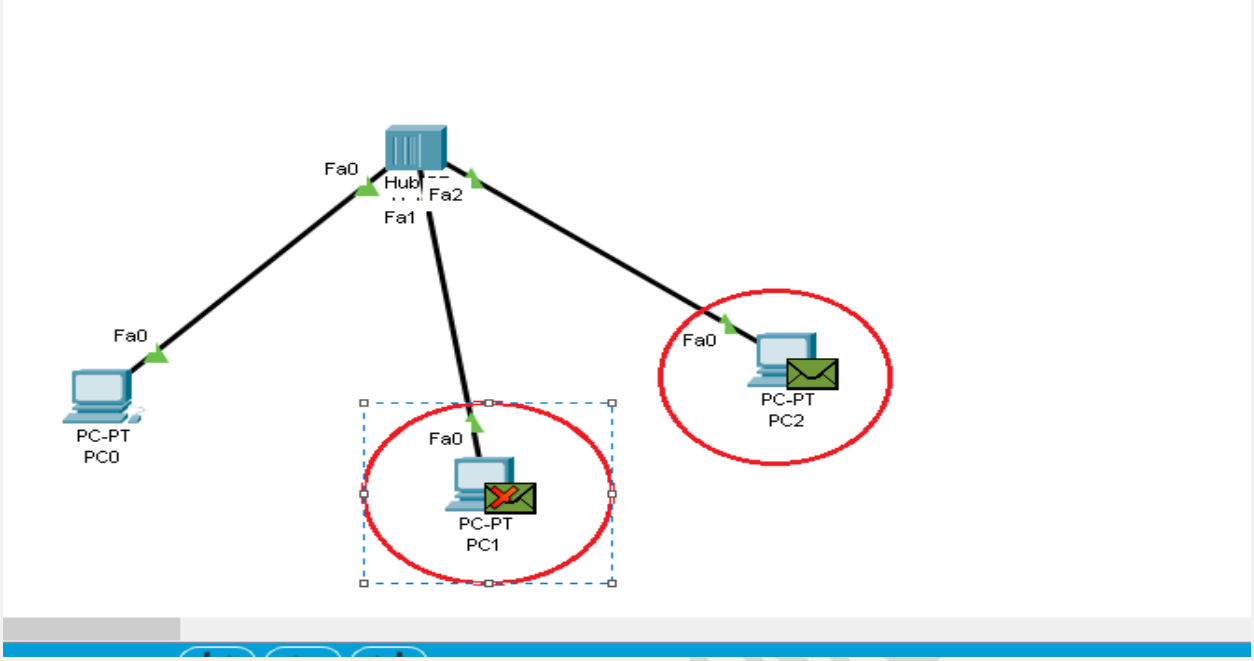
Now if we want to see the Data transmission process in **simulation mode** so we have to select the simulation bar at the right corner as shown in the fig below;



Now select the **simple PDU** and drag it to the **Workspace** and then paste on the PC as **Source** and then paste on other PC as **Destination** and then play in simulation mode, now the packet will move from the Source to all the devices that are connected to the HUB because hub have the property of sharing copy of data to all devices that are connected with it.



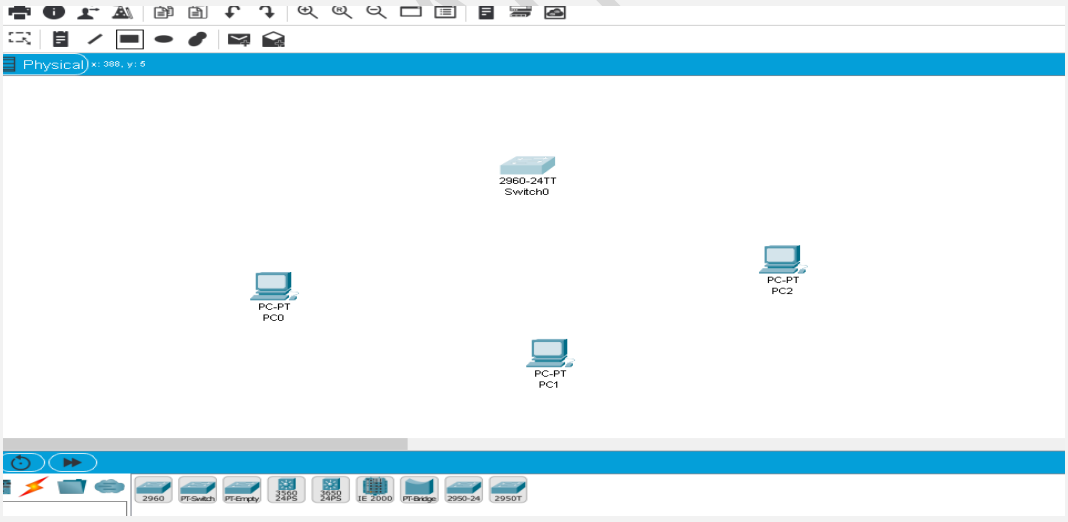
As in the above fig we are clearly saw that the **packets** the were sended from the **Source** is going to all the PC's that are connected to the **Hub**, it is because in case of Hub configuration, the packet will first go to the Hub from the Source and in the next step the Hub will send copies of the packet to all the PC's that were connected with Hub to ask them which one has the destination adrress to match with that packet. So the destination device will keep that packet while the rest of PC's will reject .



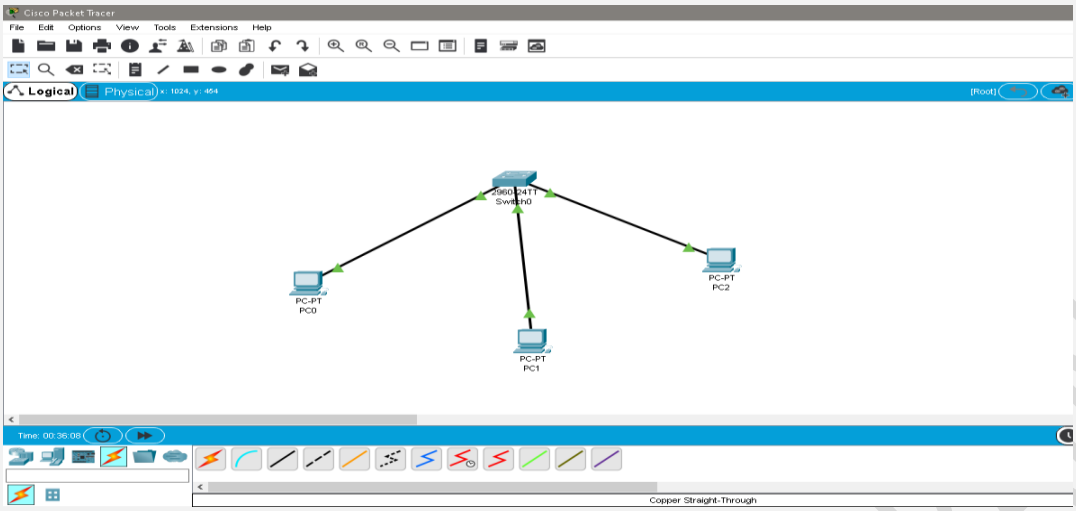
B. PROCEDURE FOR CONNECTING SWITCHES:

a) Realtime Mode:

First we have to open the **Packet Tracer** and then from the **End devices** we have to select 3 **PC's** and From **Network Devices** we have to select the **switch** and drag them into the **Workspace** as shown below;



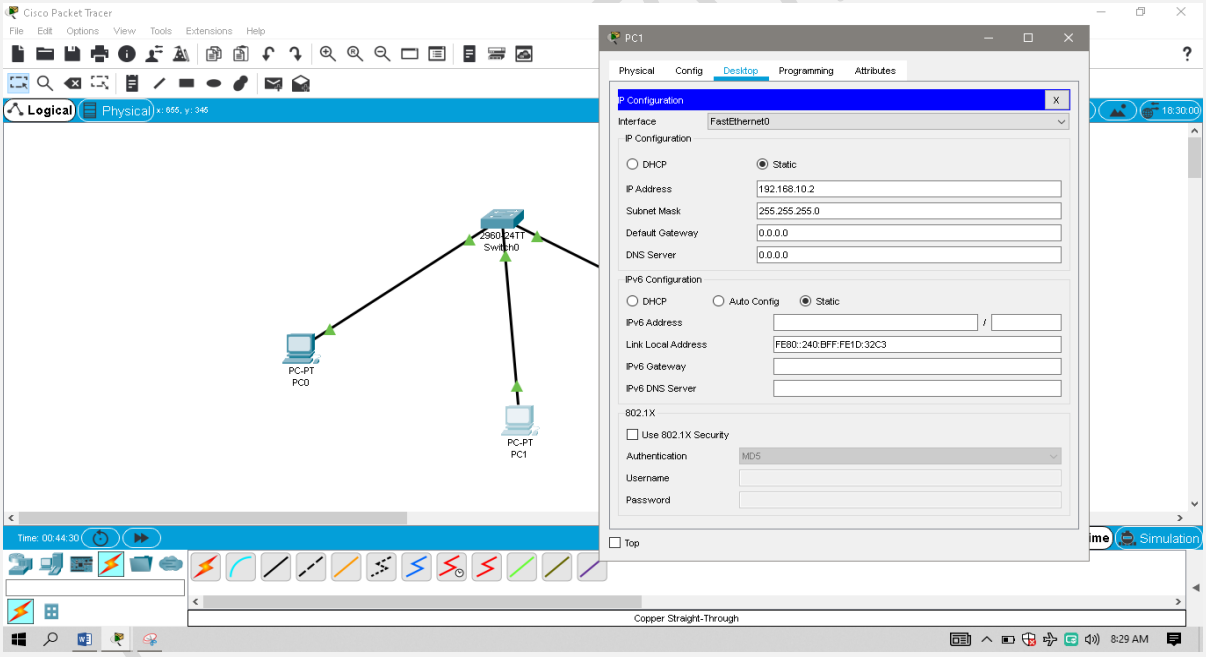
Now, to connect **Switch** with **PC's** we will use **Straight-through** cables using **FastEthernet** Ports as shown below;



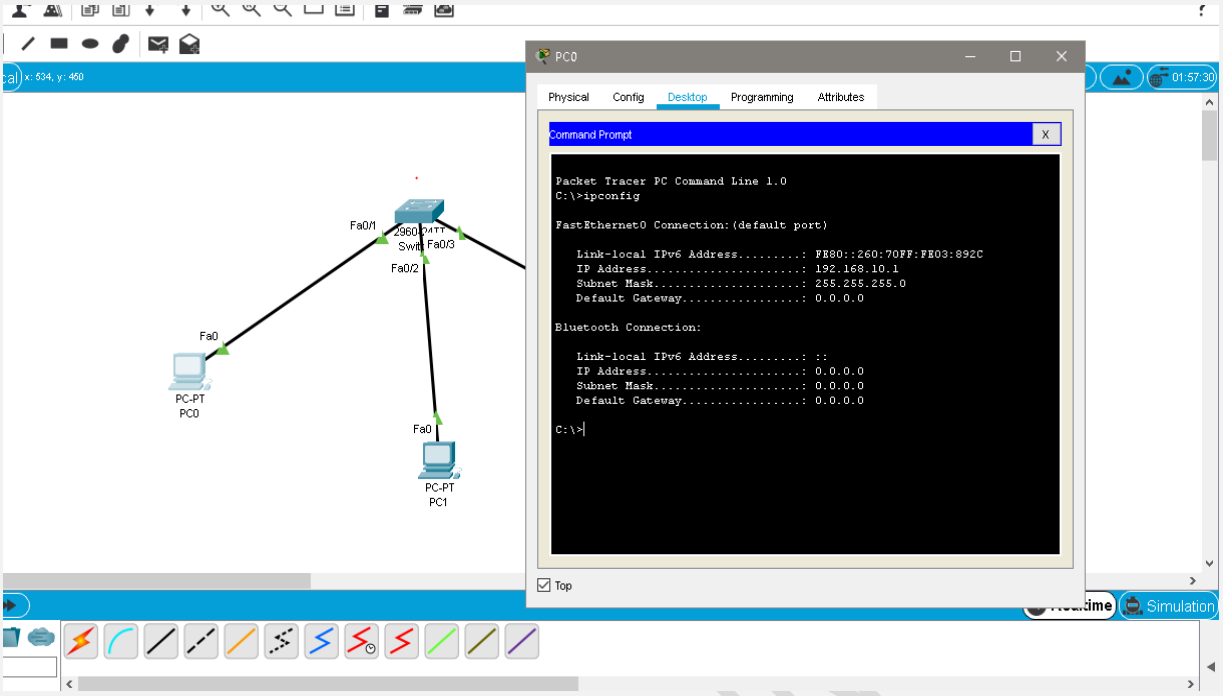
As in the above pic **Green Dots** appeared at both ends of the cables so it means our configuration is done successfully.

○ Assigning IP-addresses:

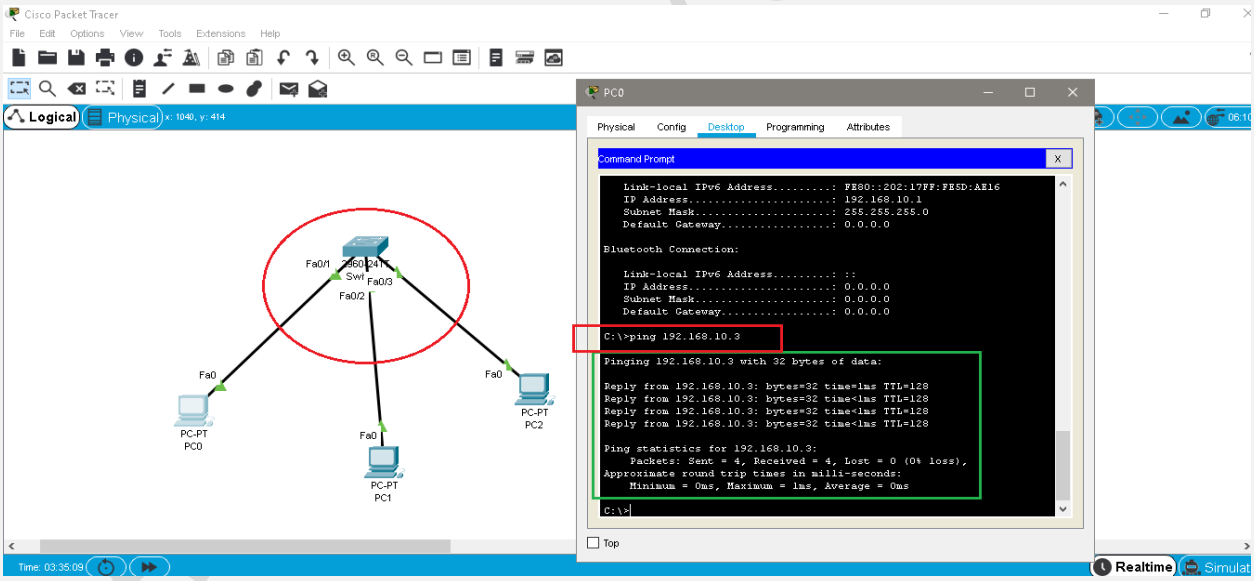
Now again we have to assign the **IP-addresses** to each **PC** as shown in the pic below;



Now if we want to check the ip-addresses of the pc's we have to double click on the pc and then go the **Desktop** and in the Desktop we have to select the **Command Prompt** where we have to enter the command **>> ipconfig** as shown below;



Now if we want to **ping** the pc2 from pc0, we will simply enter the command **,** *ping ip-address of pc2* (**>>ping 192.168.10.3**) as shown below;

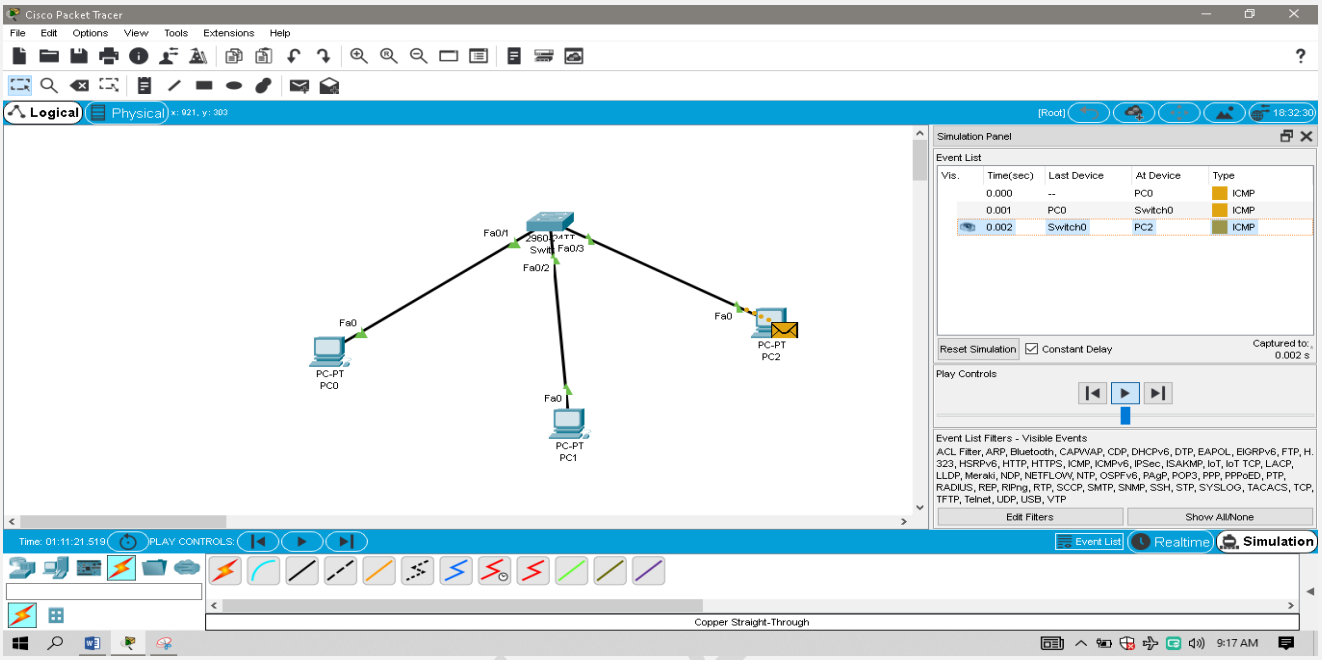


b) Simulation Mode:

Now if we want to see the whole process in **detail** , we have to go to the **simulation mode**,Then we have to select a single packet (**simple PDU**) to see the trajectory that a packet follows,then we have to select the **Source** and **destination** as well, as to mention that from where to where the packet will move.

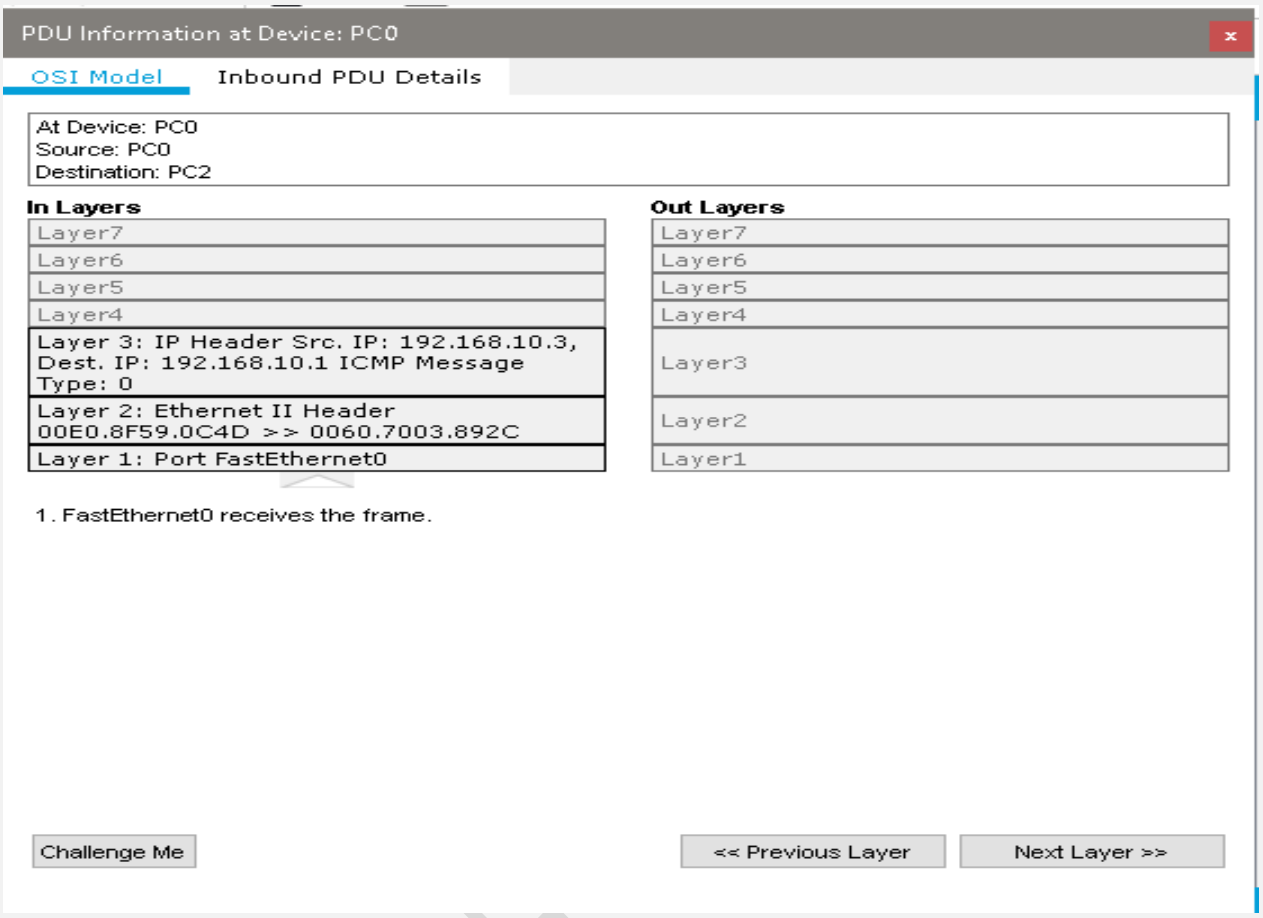
Now by playing the simulation we will see the trajectory of the packet as shown below ,the packet will move from the **Source** to the **destination** only through the switch.

In case of the **switch**, the packet will directly goes to the Destination only and will not sending the packet copies to all other PC's as shown below;



- **PDU Information:**

Now to know about the **ICMP information** we will simply double click on **ICMP** as mentioned in the **Event list** ,so as another **interface** will be opened as shown below;


PDU Information at Device: PC0
OSI Model
Inbound PDU Details
At Device: PC0
Source: PC0
Destination: PC2
In Layers
Out Layers
Layer7
Layer6
Layer5
Layer4
Layer 3: IP Header Src. IP: 192.168.10.3, Dest. IP: 192.168.10.1 ICMP Message Type: 0
Layer3
Layer 2: Ethernet II Header 00E0.8F59.0C4D >> 0060.7003.892C
Layer2
Layer 1: Port FastEthernet0
Layer1
1. FastEthernet0 receives the frame.
Challenge Me
<< Previous Layer
Next Layer >>

LAB # 04

NAME : ABDULLAH ZUNORAIN

REG_NUMBER : 19JZELE0338

SECTION : A

DEPT: ELECTRICAL COMMUNICATION

SUBJECT : COMPUTER COMMUNICATION NETWORKING

SUBMITTED TO : SYED UZAIR GILLANI

TITLE: A DETAIL OVERVIEW OF A ROUTER IN PACKET TRACER

LAB OBJECTIVE:

To know that how to connects two are more than two Local Area Network(LAN) using Router.

LAB SUMMARY:

In lab we will first make two LAN's and then by using router we will connect the LAN's with each other.

Also we will use ping command to ping the PC from LAN 1 to LAN 2.

Software Used:

Cisco Packet Tracer:

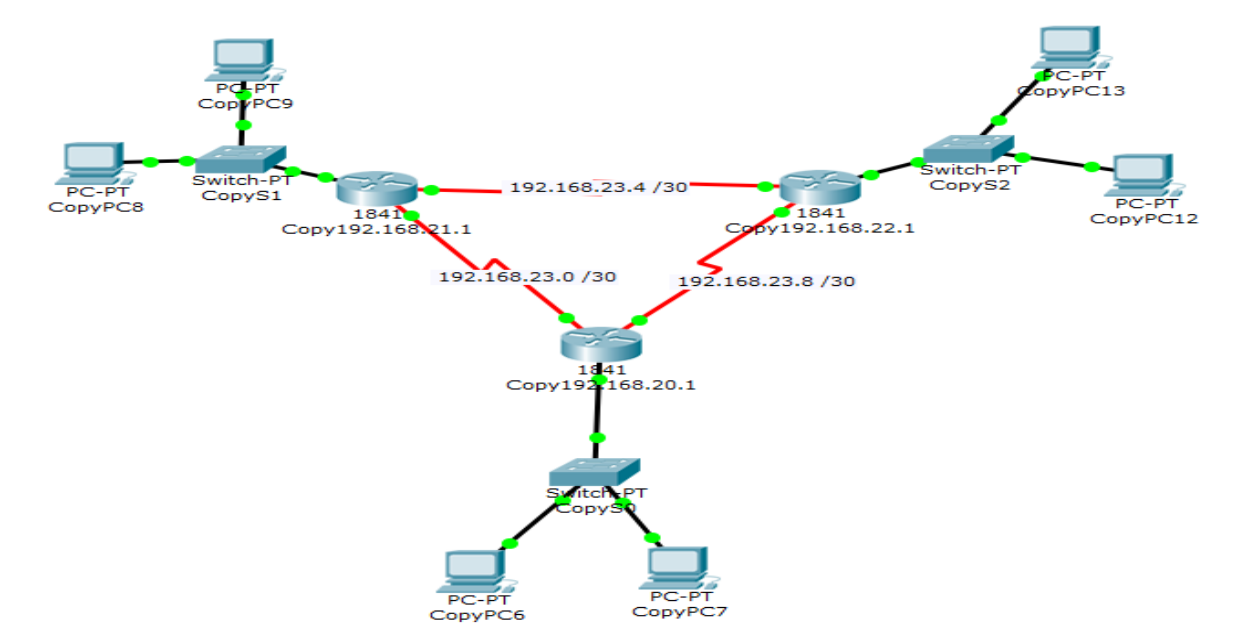
It is an innovative and powerful networking simulation tool used for practice, discovery and troubleshooting.

We used Cisco Packet Tracer, which is online publically available software, for simulation of the project.

Devices Used:

- 2960 network switches.
- Three PC's
- Copper Straight-Through cable
- 2901 router

Working of Router:

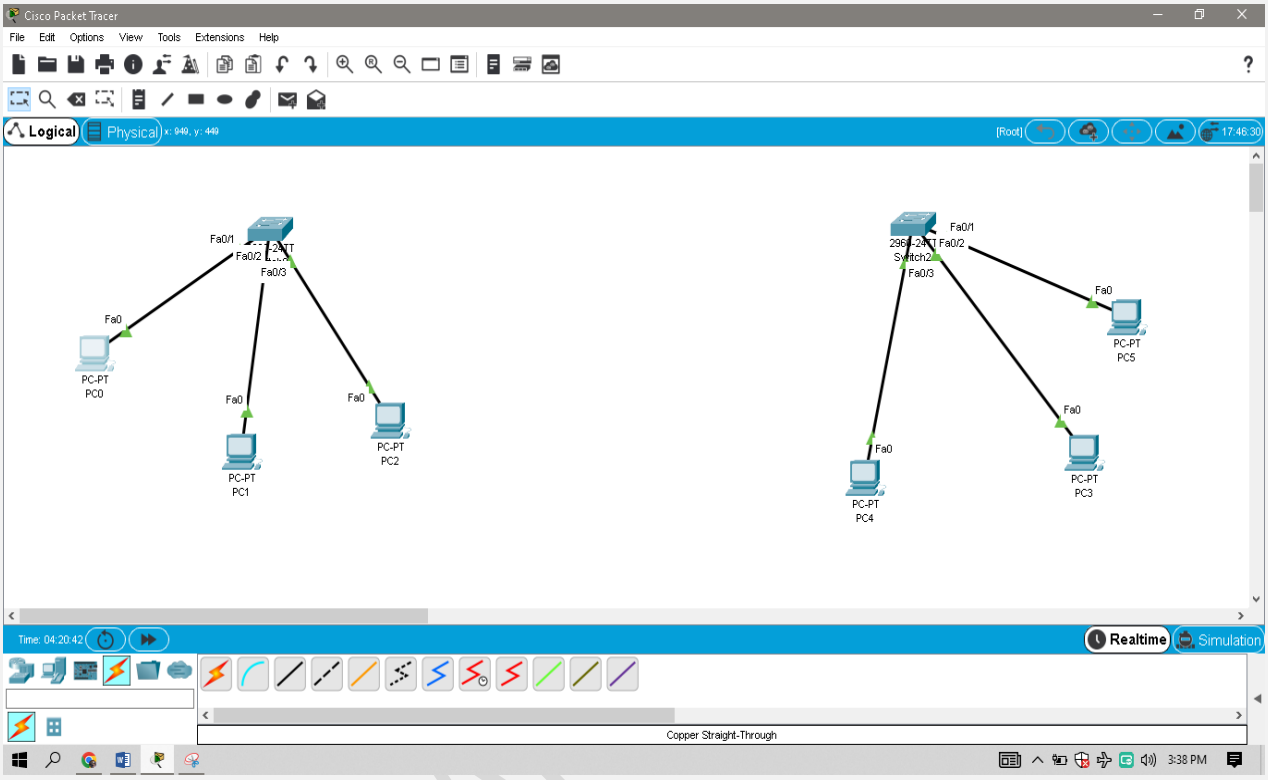


A router is a networking device that forwards data packets between **computer networks(for example LAN's)** . Routers perform the **traffic directing functions** on the *Internet*. Data sent through the internet, such as a **web page or email**, is in the form of *data packets*. A packet is typically forwarded from one router to another router through the networks that constitute an internetwork (e.g. the Internet) until it reaches its *destination node*.

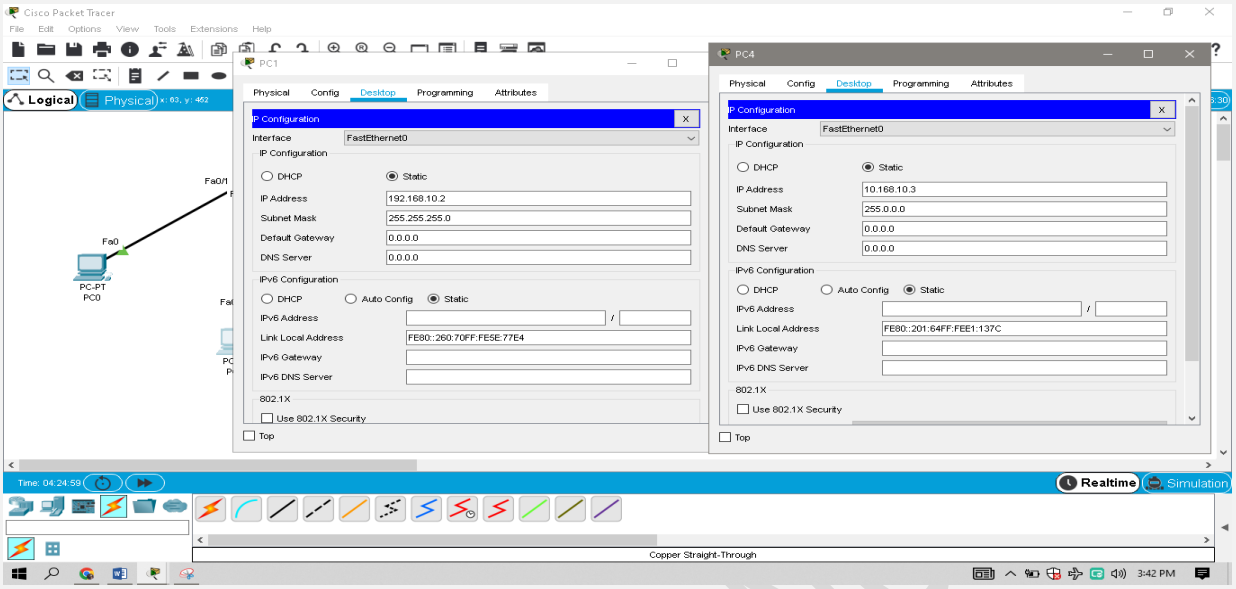
A router is connected to **two or more data lines** from different **IP-networks**. When a data packet comes in on one of the lines, the router reads the *network address information* in the ***packet header*** to determine the ultimate destination. Then, using information in its routing table or routing policy, it directs the packet to the next network on its journey.

PROCEDURE OF CONFIGURING A ROUTER IN CISCO PACKET TRACER:

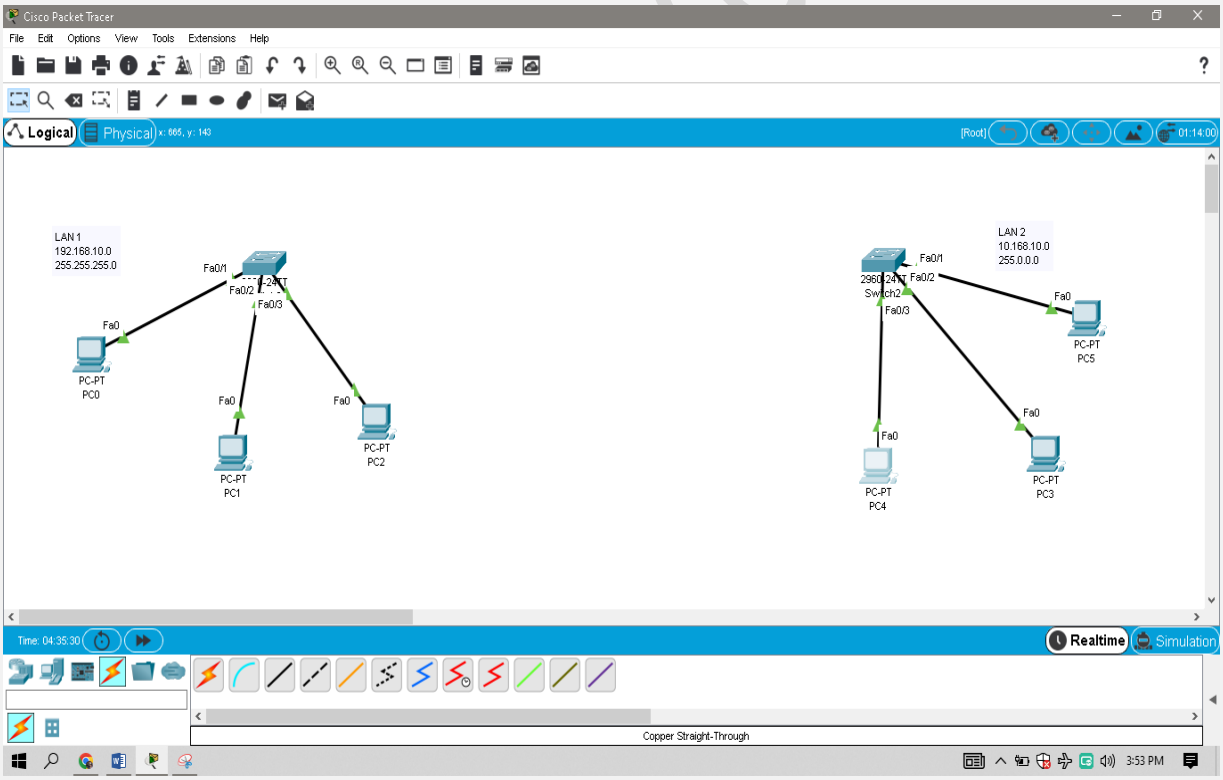
1) First create two **local area networks(LAN)** using **switches**.



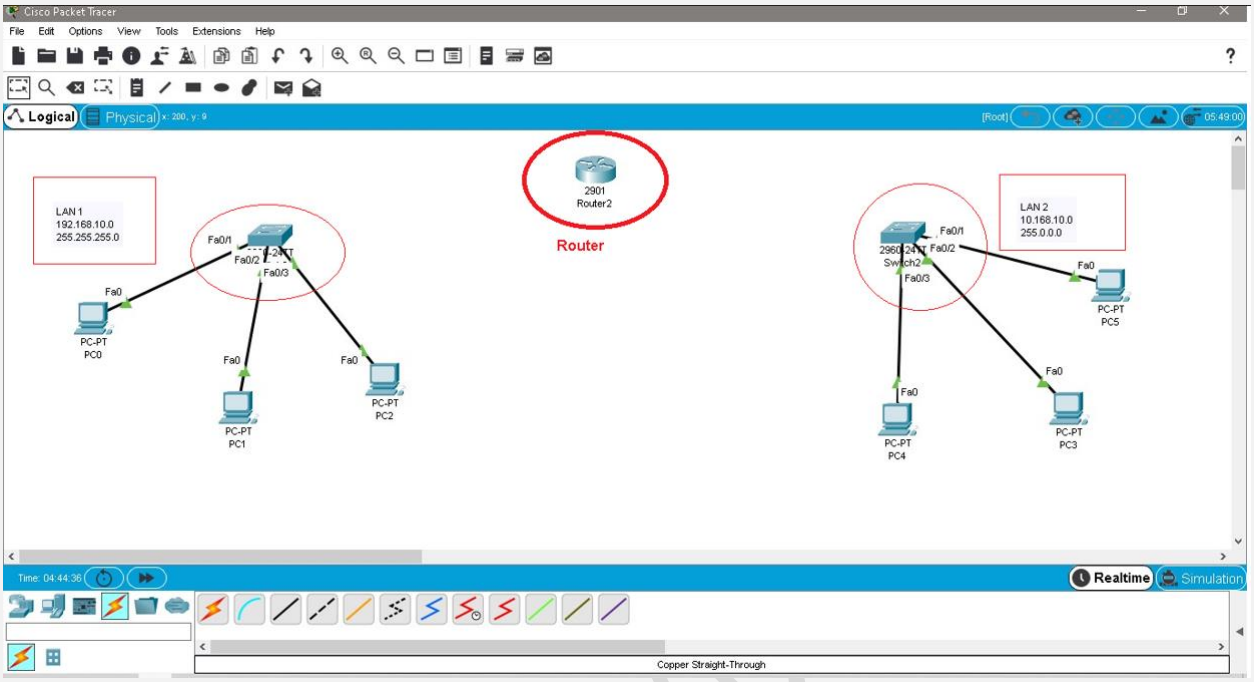
2) Set **IP-addresses** for all the *PCs* as shown below;



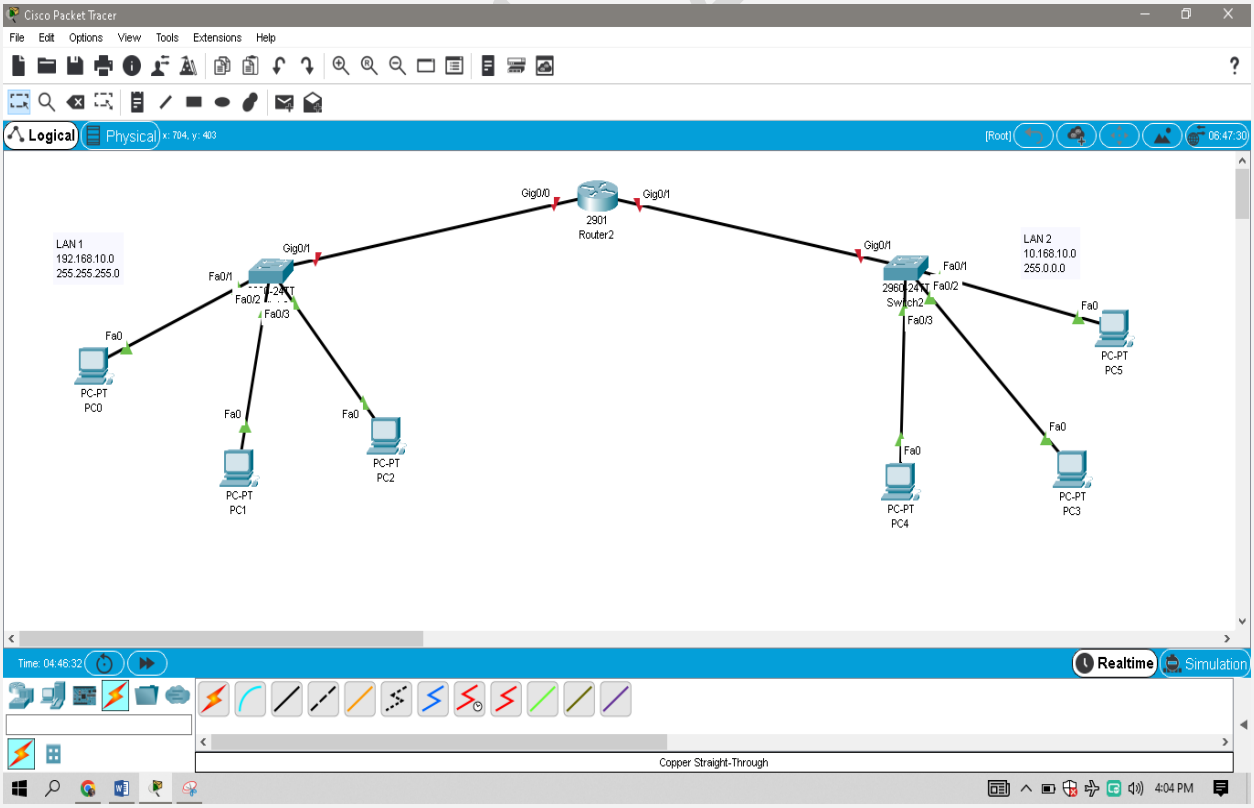
3) Now we haveto created **two LANs** as shown below;



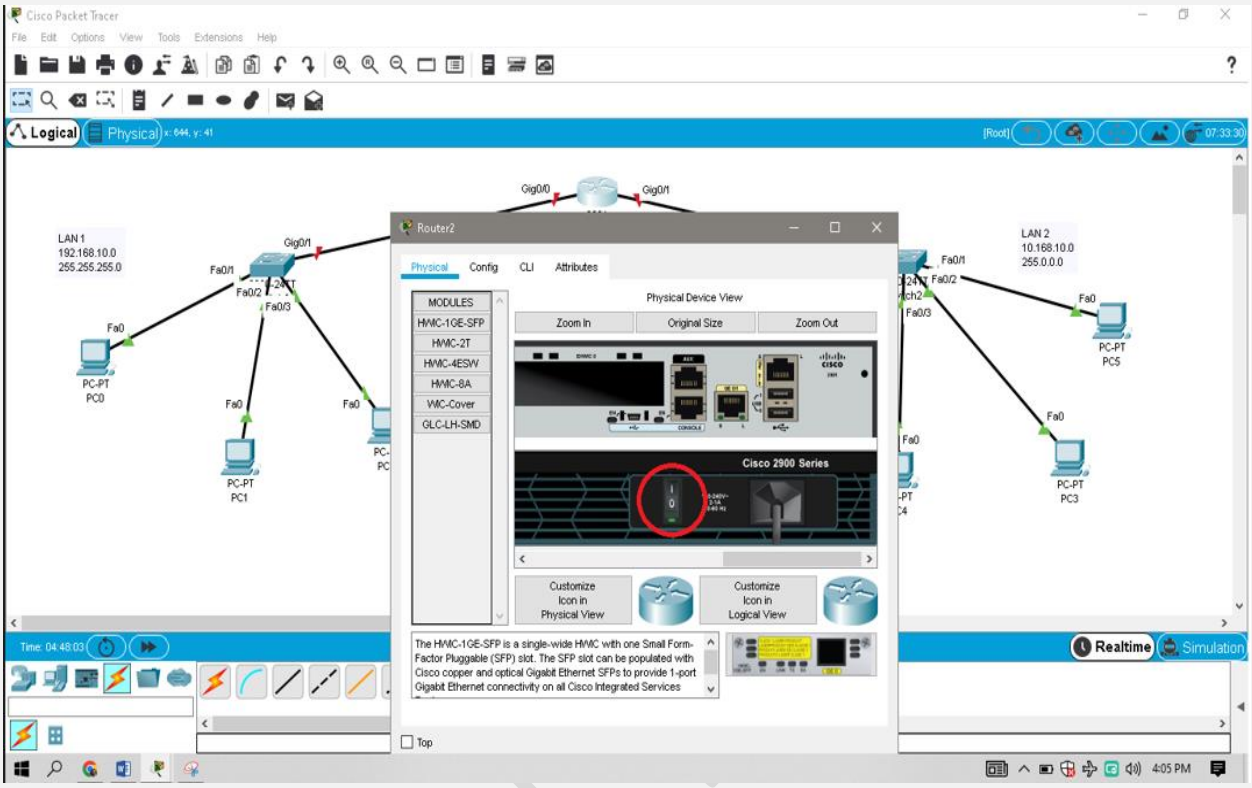
4) Next add **router** to your *topology* that we can **ping** a PC in *LAN 2* from *LAN 1*,so for this we have to drag router ,as shown below;



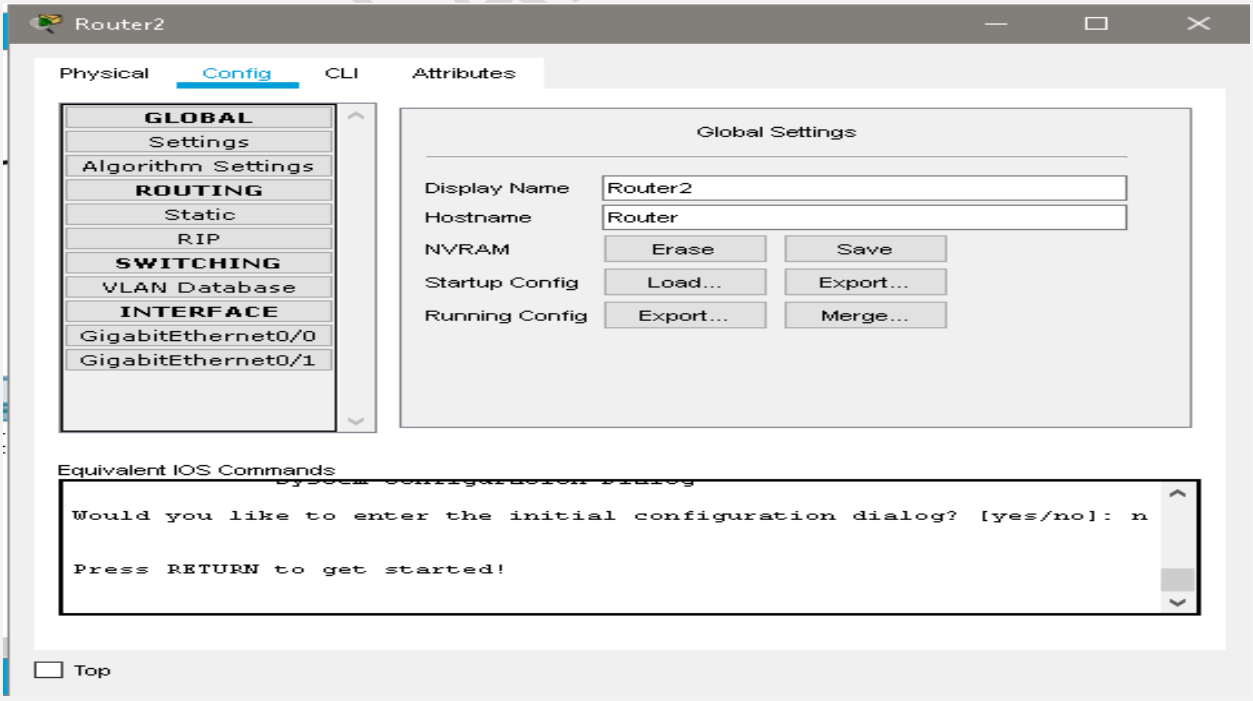
5) Now connect both networks**(LAN)** using **router** as shown below;



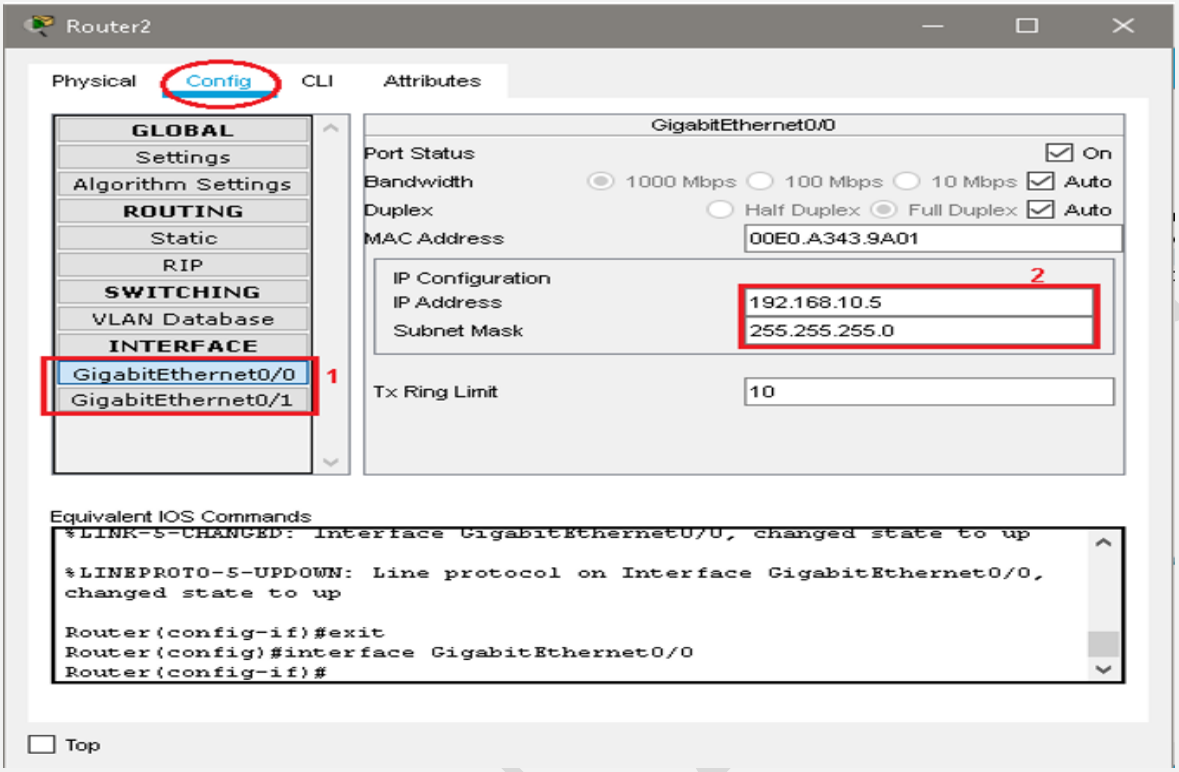
6) Now **double-click** on the Router to open a new *window* then open the **Physical** interface of the router and turn **ON** the router as shown in the below screen shot.



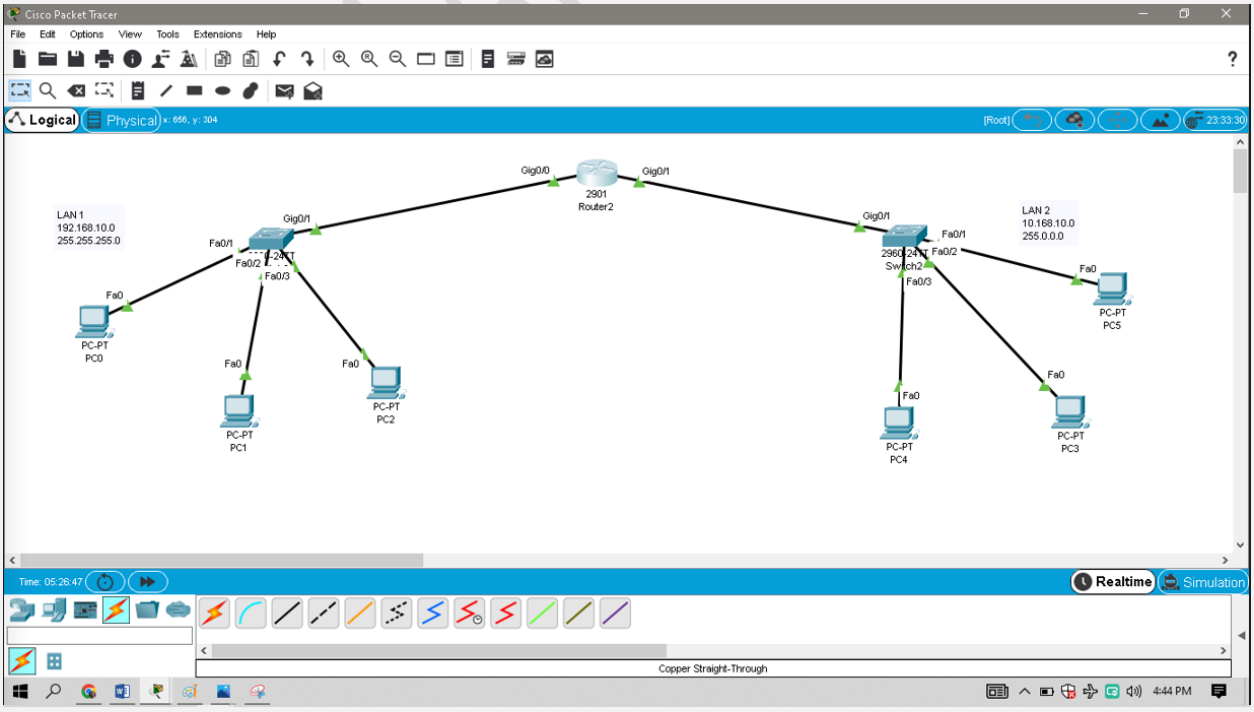
7) Next click on the **config tab**;



8) And give **IP-addresses** to both the **Interfaces(GigabitEthernet0/0 & GigabitEthernet0/1)** of the *router* as shown below;



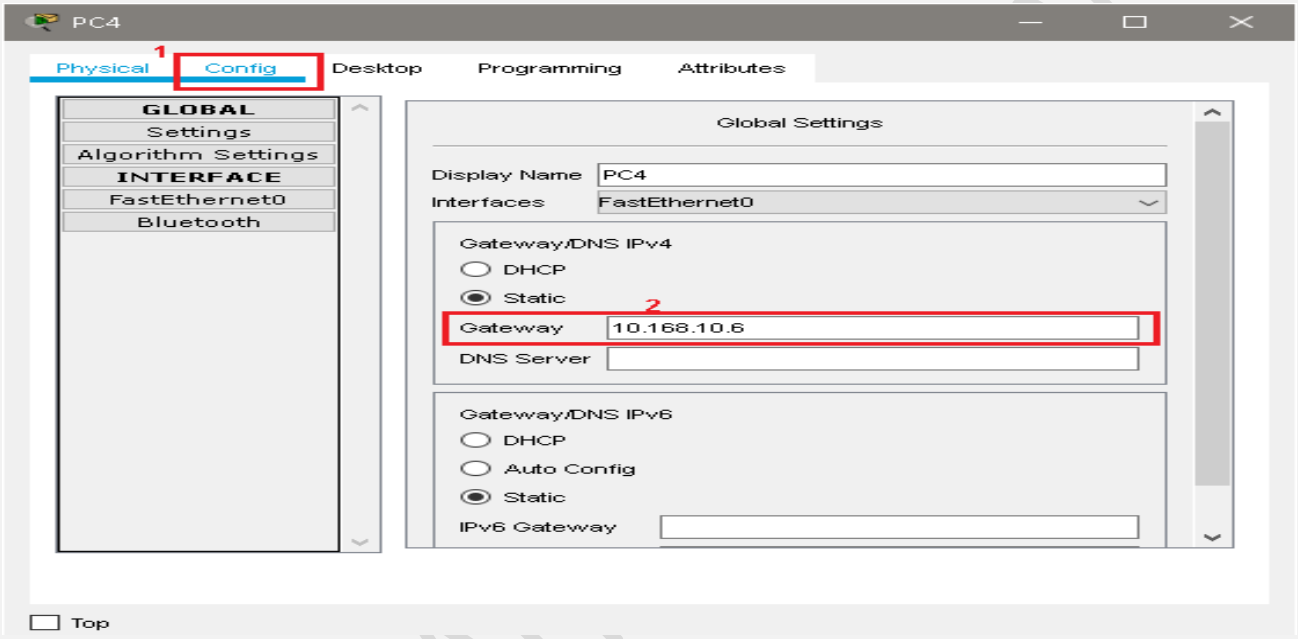
9) Now in the below pic we can clearly see that the **configuration** is *successfully* done.

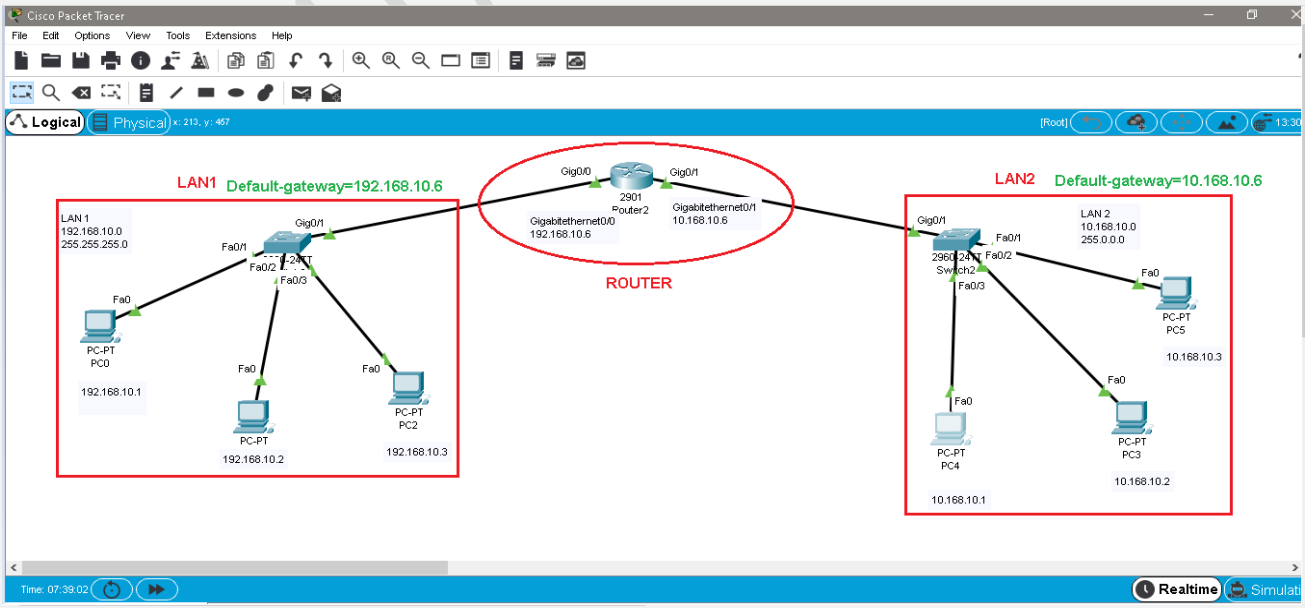


PINGING A PC IN LAN-2 FROM LAN-1:

Now if we want to ping a PC in **LAN 2** from **LAN1** we can do it through **router**, so for this we have to first select the **Default Gateway= 192.168.10.6** in all the PC's in the **LAN1**.

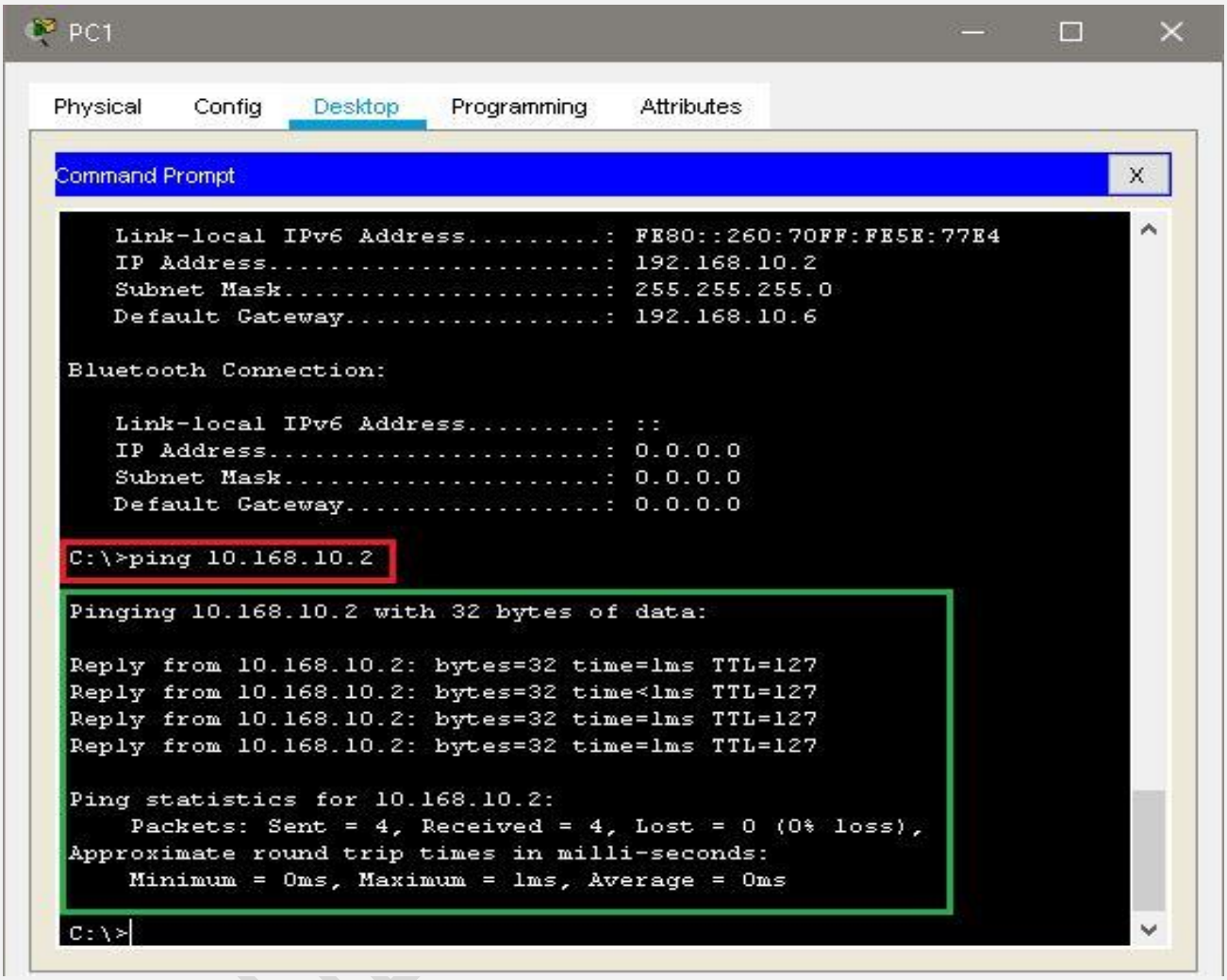
Similarly for **LAN2**,we have to select the **Default Gateway= 10.168.10.6** for all PC's in LAN2 as shown below;





Now we can **ping** the PC's in *LAN2* from *LAN1*.

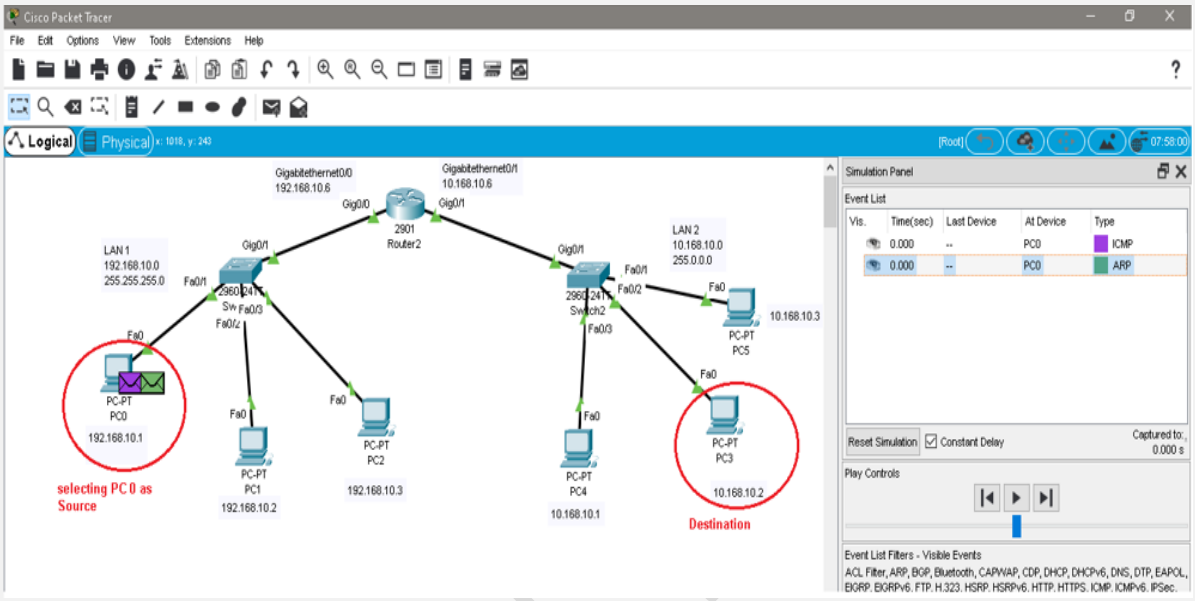
As shown below I pinged the **PC-3(10.168.10.2)** in *LAN2* from **PC-1(192.168.10.2)** as shown below;



Simulation Mode:

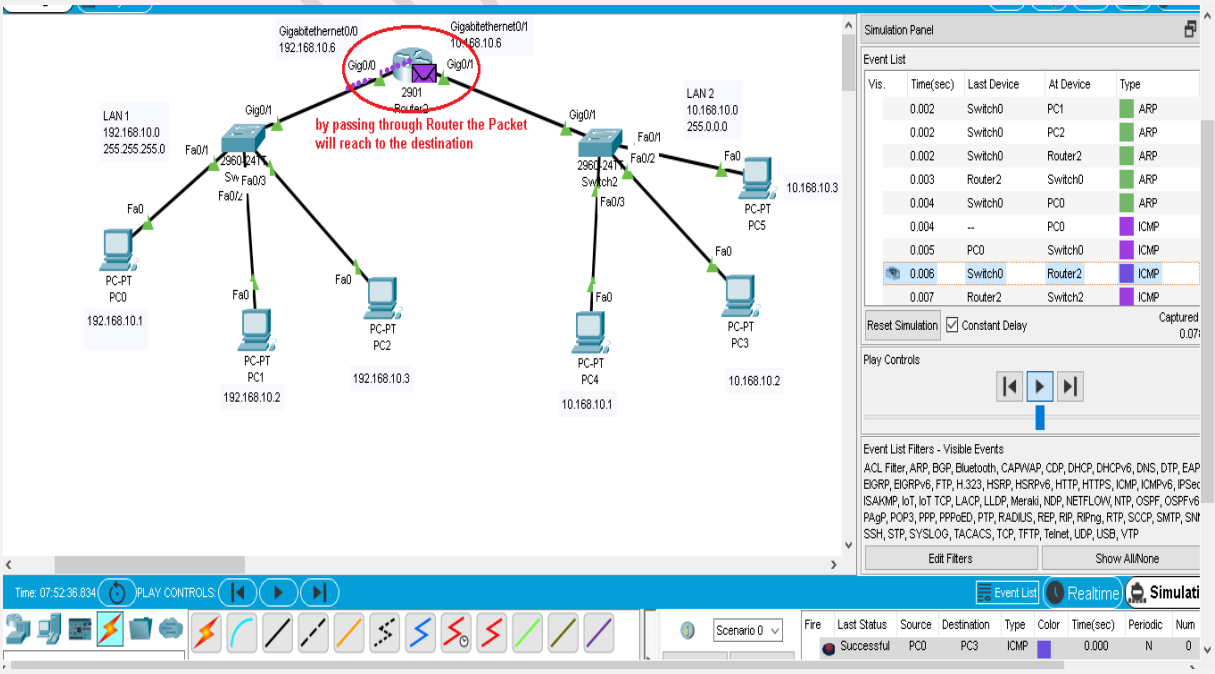
- In **simulation mode** we will see the trajectory followed by the packet in detail.

For this we have to select the **Simple PDU** and drag to ***workspace*** and paste it on the **PC0** for selecting the ***Source in LAN 1*** and then paste on the **PC3** in the ***LAN 2*** for choosing **as *detination*** as shown below;

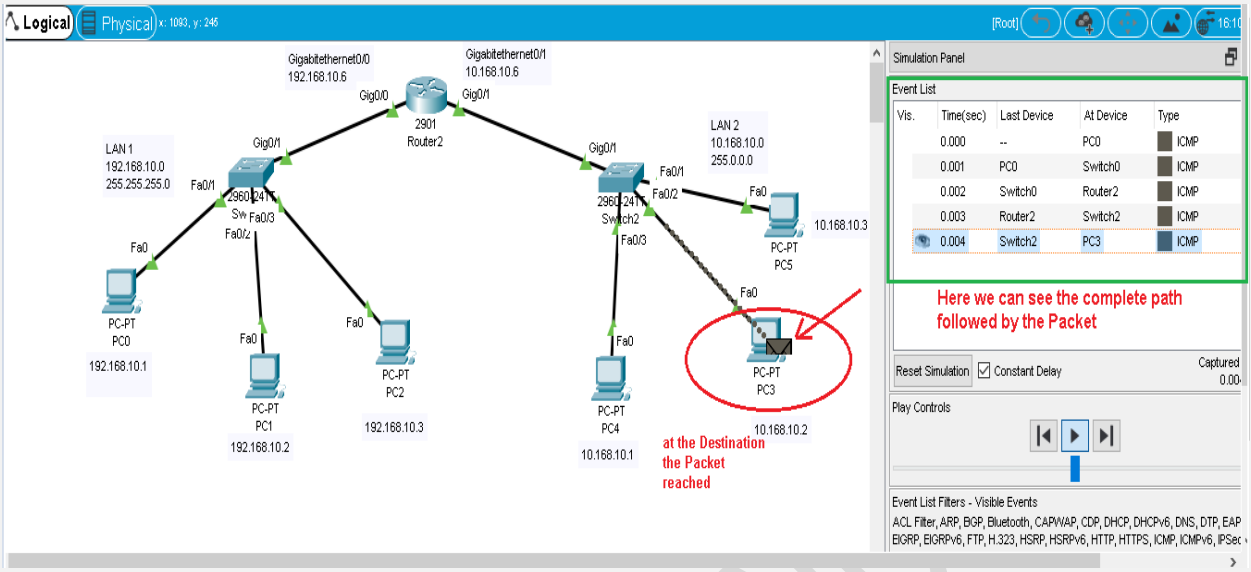


Now for playing the *simulation* we have to click on the **Play button**.

Now when we click the Play Button the packet will move from ***source*** to the ***destination*** by passing through ***Router*** as shown below;



Now we see that the **Packet** reached to the **Destination** *successfully* in the below pic;



ICMP PROTOCOL:

- *It Is a Network-Layer Protocol used by network devices to communicate.*
- A ping command sends an **ICMP** echo request to the target host, and the target host responds with an echo reply.

LAB # 05

NAME : ABDULLAH ZUNORAIN

***REG NO*:** 19JZELE0338

SUBJECT: COMPUTER COMMUNICATION NETWORKING

SUBMITTED TO: SYED UZAIR GILLANI

SECTION: A

DEPT: ELECTRICAL COMMUNICATION

TITLE : PC NETWORK TCP/IP CONFIGURATION

PART_(a);

OBJECTIVES

- Identify tools used for discovering a computer's network configuration with various operating systems.
- Gather information, including the connection, host name, MAC(Layer2) address, and TCP/IP Network(Layer 3)
- Compare the network information to that of other PC's on the network.

BACKGROUND

This lab assumes that you are using Windows NT/2000/XP/Vista/7/8/10. This is nondestructive lab that you can perform on any host without changing the system configuration.

Ideally, you perform this lab in a LAN environment that connects to the Internet. You can use a single remote connection via a dial up modem , DSL or any other. You will need the IP address information which the instructor should provide.

Step 1: Connect to the Internet.

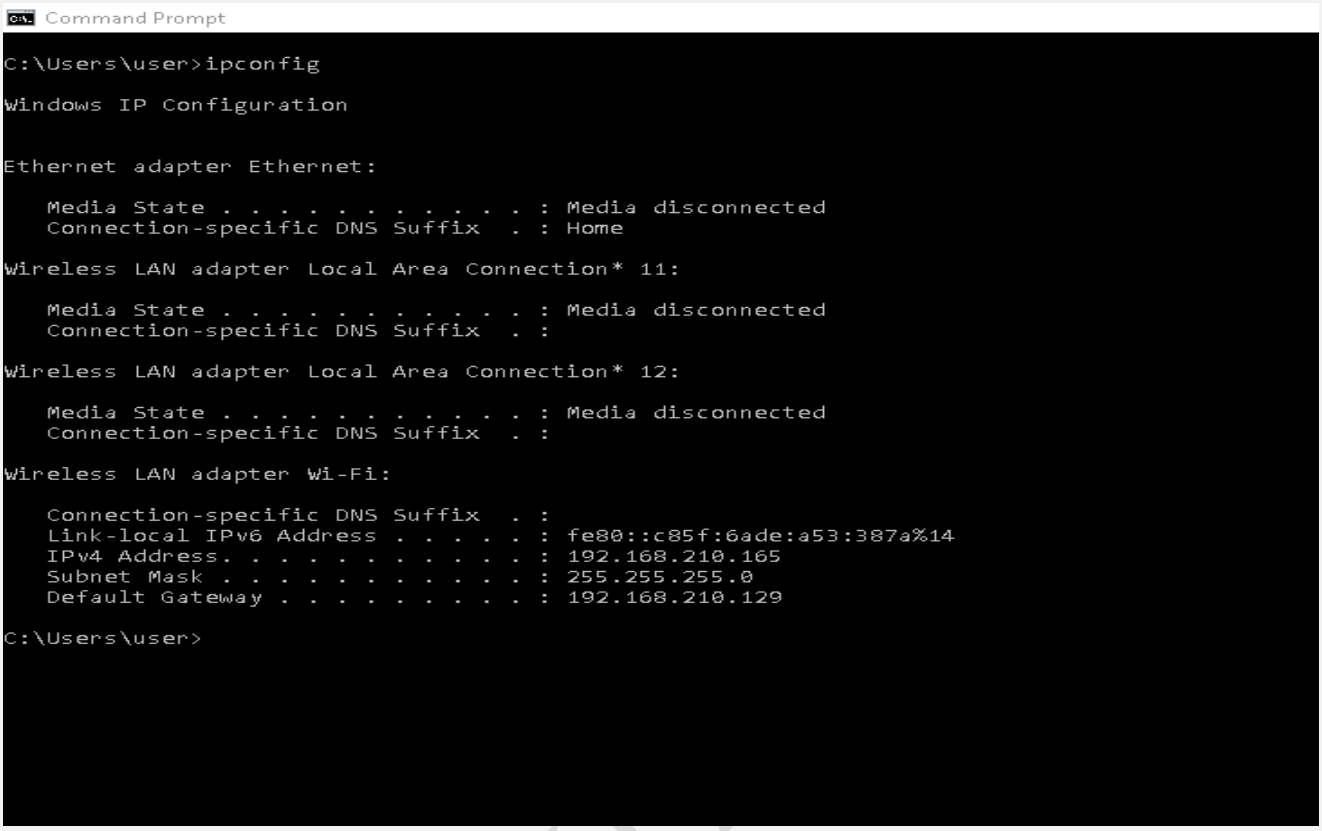
Establish and verify connectivity to the Internet. This step ensures the computer has an IP address.

Step 2: Gather TCP/IP configuration information.

a. Use the Start menu to open the command-prompt

(Start>Programs>Accessories>Command Prompt or Start>Programs>Command Prompt).

b. Type ipconfig and press Enter key. The spelling of the ipconfig is critical, but the case is not.



c. The screen shows the IP address, subnet mask and the default gateway. The IP address and the default gateway should be in the same network or subnet; otherwise this host wouldn't be able to communicate outside the network.

Step 3. Record the following TCP/IP information for this computer.

a. IP address: 192.168.210.165

b. Subnet mask: 255.255.255.0

c. Default gateway: 192.168.210.129

Step 4. Compare this computer's TCP/IP configuration to that of others on the LAN.

If this computer is on a LAN, compare the information of several machines (Hosts).

a. Are there any similarities?

YES, Subnet mask(255.255.255.0) and Default gateway(192.168.223.229) are similar.

b. What is similar about the IP addresses?

In IP-addresses , Network address(192.168.223) are same as other computers which are also connected to the same network.

c. What is similar about the default gateway?

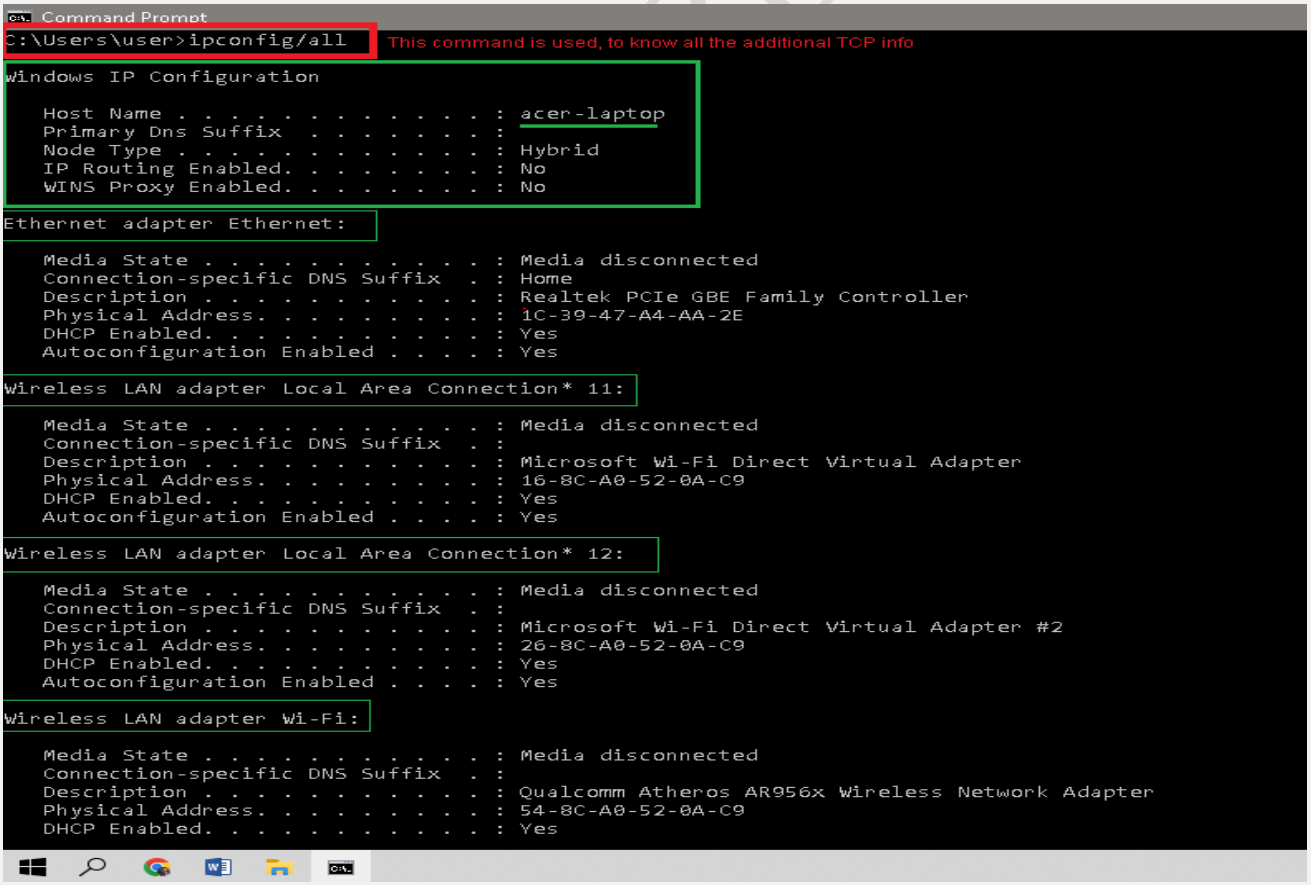
Default gateway is similar to that of other computer's connected to the same network(router).

d. Record a couple of the IP addresses(of your nearby hosts).

1. 192.168.223.165
2. 192.168.223.41
3. 192.168.223.151

Step 5. Check additional TCP/IP configuration information.

a. To see more information, type ipconfig/all and press Enter key. The figure shows the detailed IP configuration of this computer on the screen.



You should see the following information: the *host name(computer name)*, the *Physical address of this machine*, *IP address*, *subnet Mask*, *Default Gateway* and *DNS Servers*.

b. In the LAN, compare your result with a few nearby computers. What similarities do you see in the physical(MAC) address?

Physical address of each system is different.

c. Write down the computer's host name:

Acer-Laptop.

d. Write down the host names of a couple of other computer:

i DESKTOP-4674D02

ii DELL_LAPTOP

iii HP-DESKTOP

Step 6. Close the screen when finished.

DONE.

PART_(b);

TITLE : Using ping and tracert from a workstation

Objective

- Learn to use the TCP/IP ping command from a workstation.
- Learnt to use the tracert command from a workstation.
- Observe name-resolution occurrences using WINS and DNS servers.

Background/Preparation

This lab assumes that you are using any version of Windows. This is a nondestructive lab that you can perform on any machine without changing the system configuration. Ideally, you perform this lab in a LAN environment that connects to the Internet. You can use a single remote connection via a dialup mode or DSL connection. You need the IP addresses that were recorded in the previous lab.

ACCESS:

Ping is used to determine whether the remote host is active or inactive. If a certain site is not pinged, but the other sites can, then it's a pretty good sign that your Internet network is fine and that site is down. On the other hand, if you can't ping any site, then likely your entire network connection is down that needs rebooting.

TIME AND DISTANCE:

Ping command is also used to determine how long it takes to bounce a packet off of another site. In networking language it means giving Internet distance. For example, a web site hosted on one's computer with a different Internet service provider (ISP) might go through more routers and be farther away in network distance than a site on the other side of the ocean with a direct connection to the Internet backbone. If a site seems slow, then ping distance of that site can be compared with that of other Internet sites to find out whether it is the site, the network, system that is slow.

DOMAIN IP ADDRESS:

Ping command is used to probe either a domain name or an IP address; if a domain name is pinged, then it displays the corresponding IP address in its response.

WORKING:

Ping command uses a series of Internet Control Message Protocol (ICMP) Echo messages to perform its operation. It first sends an echo request packet to anaddress, and then waits for a reply.

The ping is successful only if:

- The echo request gets to the destination.
- The destination is able to get an echo reply back to the source within apredetermined time called a timeout. The default value of this timeout is twoseconds on Cisco routers. Ping command can be used along with other different codes to get following information:

ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] target_name.

where target_name can be either IP Address or host name;Option Description

-t	Ping the specified host until stopped. To see stats & Continue, type Control-Break; to stop, type Control-C
-a (Resolve Addresses)	Resolve address to host names
-n count (Echo Count)	Sends echo packets specified by count; Default is 4
-l size (Send buffer Size)	Use to increase or decrease the size of the ICMP packets sent in the ping request; Min=0, Max=65500
-f (Set Packet don't Fragment Flag)	Select to send a DO NOT FRAGMENT flag in the packet. The packet will not be fragmented by Gateways on the route. Use in conjunction with Send Buffer Size option.
-i TTL (Time to Live)	Sets the number of hops (routers) that the ping request can traverse before it is discarded. The TTL field in the packet is decremented by one each time it passes through a router. When the number reaches zero, the router discards the packet and sends a TTL Expired ICMP message back.
-v TOS (Type of Service)	Sets the TOS value in the ICMP packet for routers that are set up to treat packets with certain types of service differently than others. ToS is not used very often, and most routers ignore it.
-r count (Record Route for Hops)	Records the route of the outgoing packet and the returning packet in the record field; Min=1, Max=9
-s count (Timestamp for Hops)	Specified the timestamp for the number of hops Min=1, Max=4
-w timeout	Timeout in milliseconds that ping waits for each reply

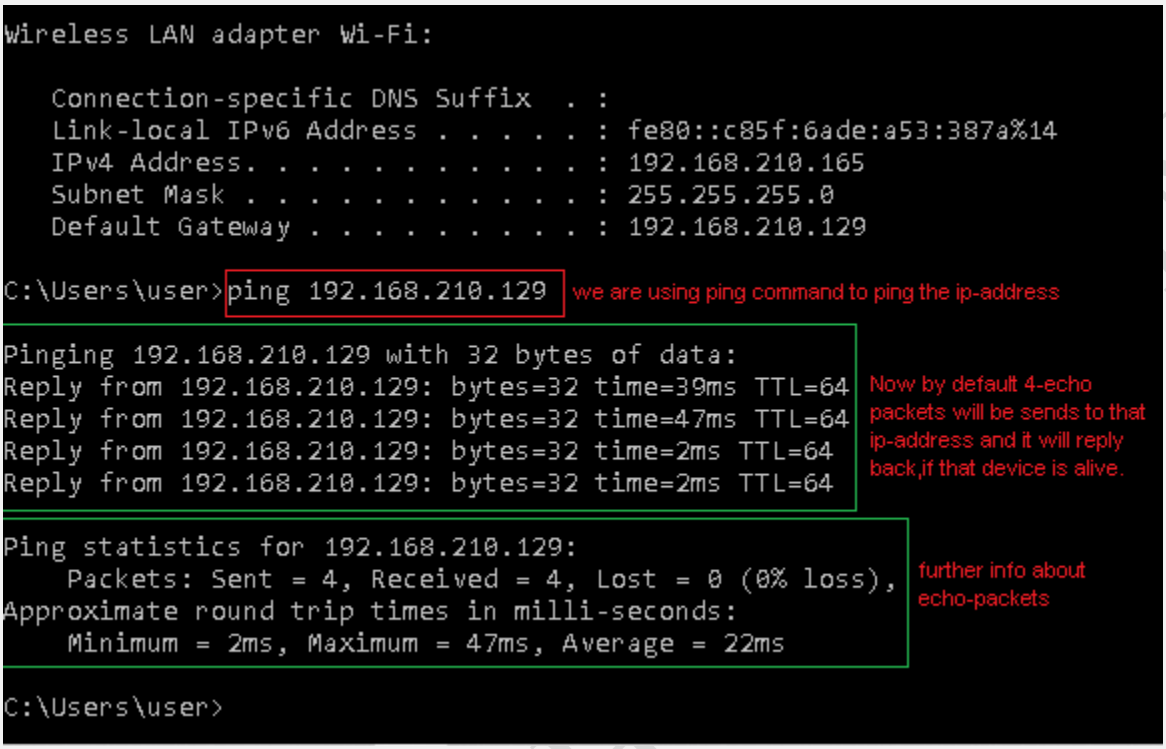
Sept 1. Establish and verify connectivity to the Internet.

This step ensures that the computer has an IP address.

Step 2. Open the Command prompt (MS-DOS). Ping the IP address of another computer.

a. In the window, type ping, a space, and the IP address of a computer recorded in the previous lab.

Pinging the IP-address of Default gateway;



Sample pinging

The following is a sample output of pinging `en.wikipedia.org` under Linux with the iputils version of ping:

```
[user@box] ping en.wikipedia.org
PING rr.pmtpa.wikimedia.org (66.230.200.100) 56(84) bytes of data. 64 bytes
from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=1 ttl=52 time=87.7 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=2 ttl=52
time=95.6 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=3 ttl=52
time=85.4 ms
```

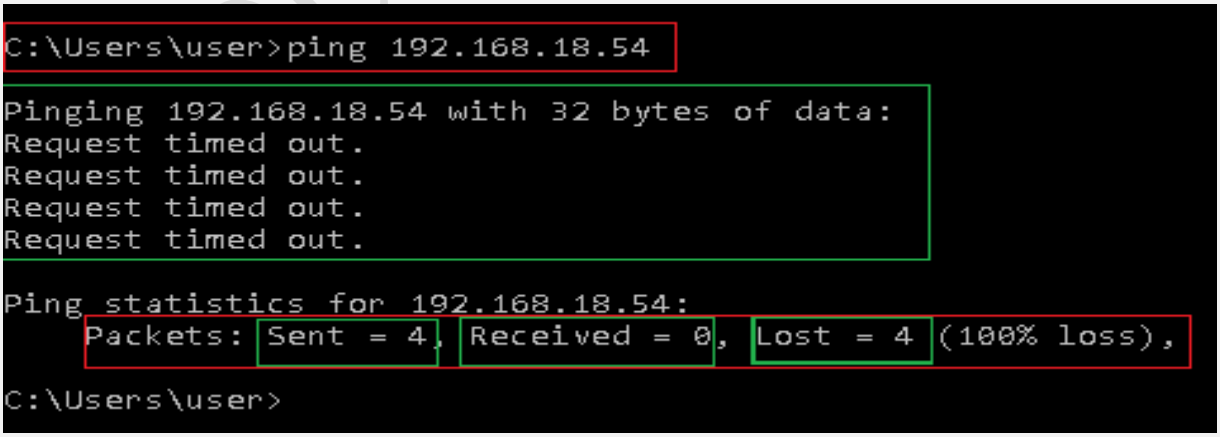
```
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=4 ttl=52
time=95.8 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=5 ttl=52
time=87.0 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=6 ttl=52
time=97.6 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=7 ttl=52
time=87.3 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=8 ttl=52
time=97.5 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=9 ttl=52
time=78.1 ms
64 bytes from rr.pmtpa.wikimedia.org (66.230.200.100): icmp_seq=10
ttl=52time=79.5 ms


--- rr.pmtpa.wikimedia.babunlaut ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 8998ms rtt
min/avg/max/mdev = 78.162/89.213/97.695/6.836 ms
```

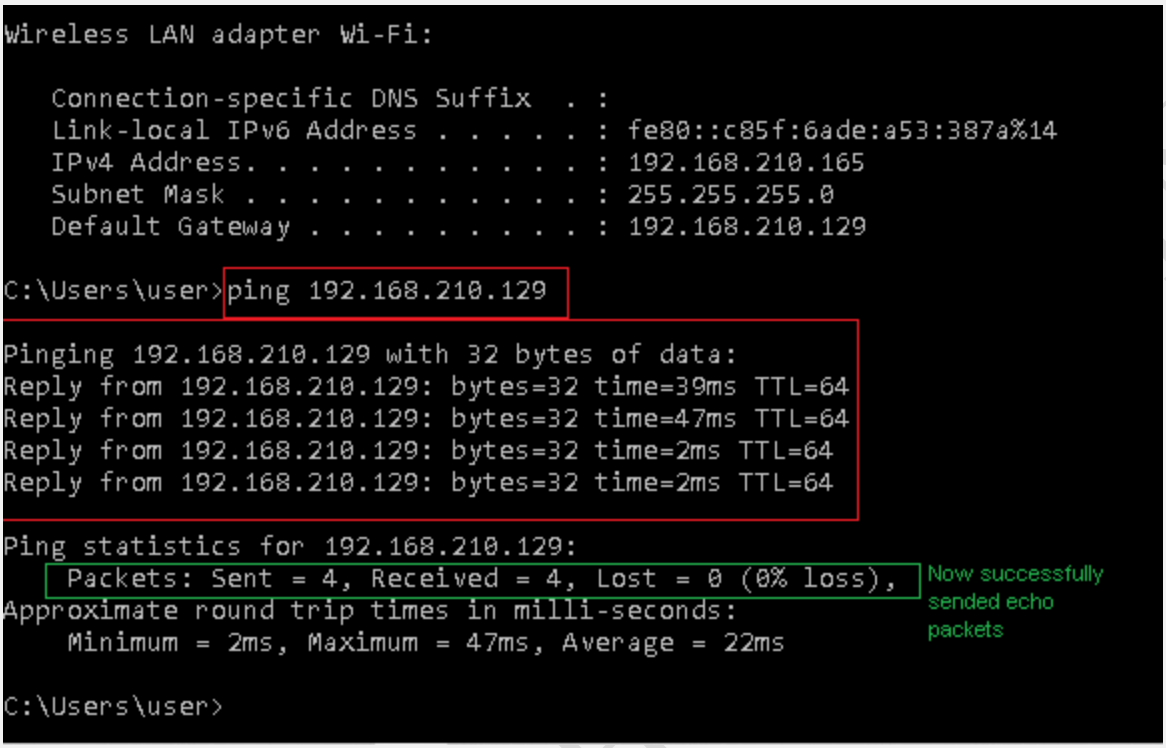
Ping uses the Internet Control Message Protocol (ICMP) echo-request and echo-reply feature to test physical connectivity. Because ping reports on four attempts, it gives an indication the reliability of the connection. Look over the result and verify that the ping was successful. Was the ping successful? If not, report to the instructor.

b. Ask the IP address of the nearby computers and ping. Note the result.

As the computer connected to other router than my router so the ping of that computer results as request timed out as shown below;

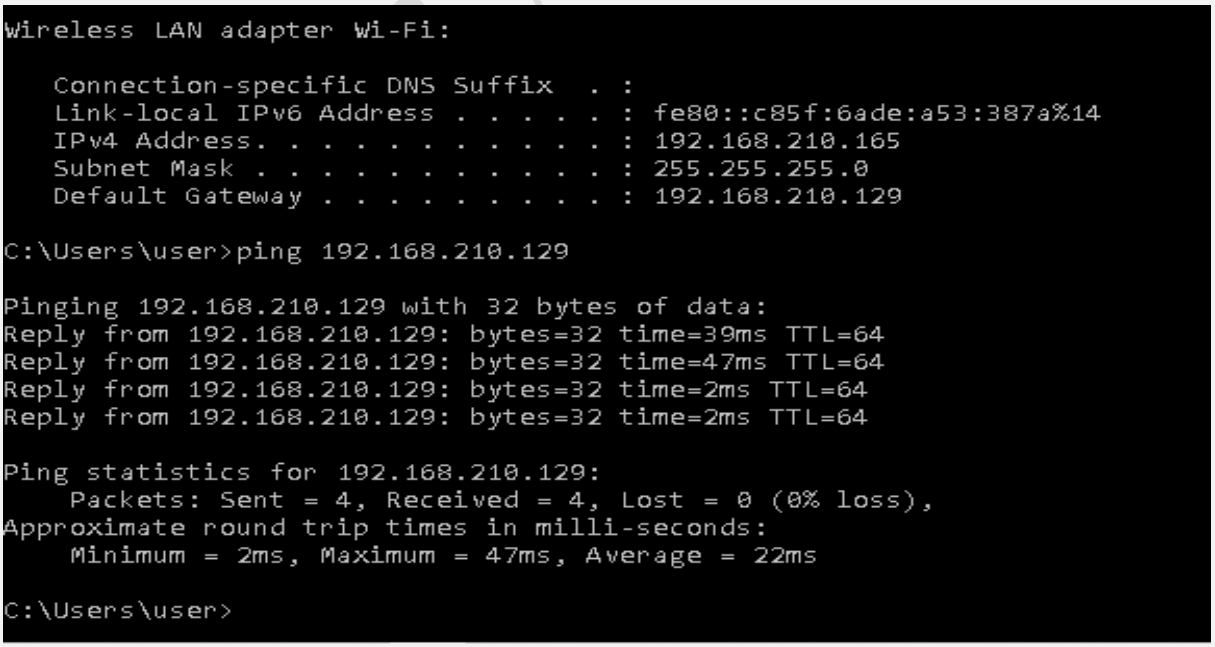


While if we have nearby computer connected to the same router then the pinging of IP-address of that computer will be successful. As here in below pic we are ping the default gate-way.



C. Ping the IP address of Default gateway and DNS servers. Was the result successful?

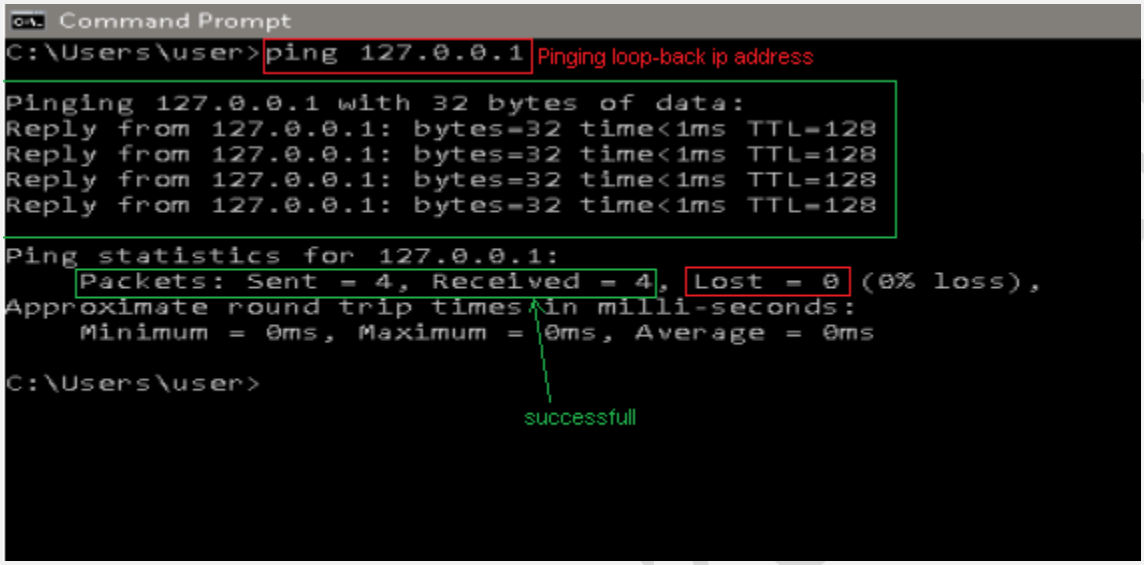
Yes, the result successful;



d. Ping the computer's loopback IP address. Type the following command:

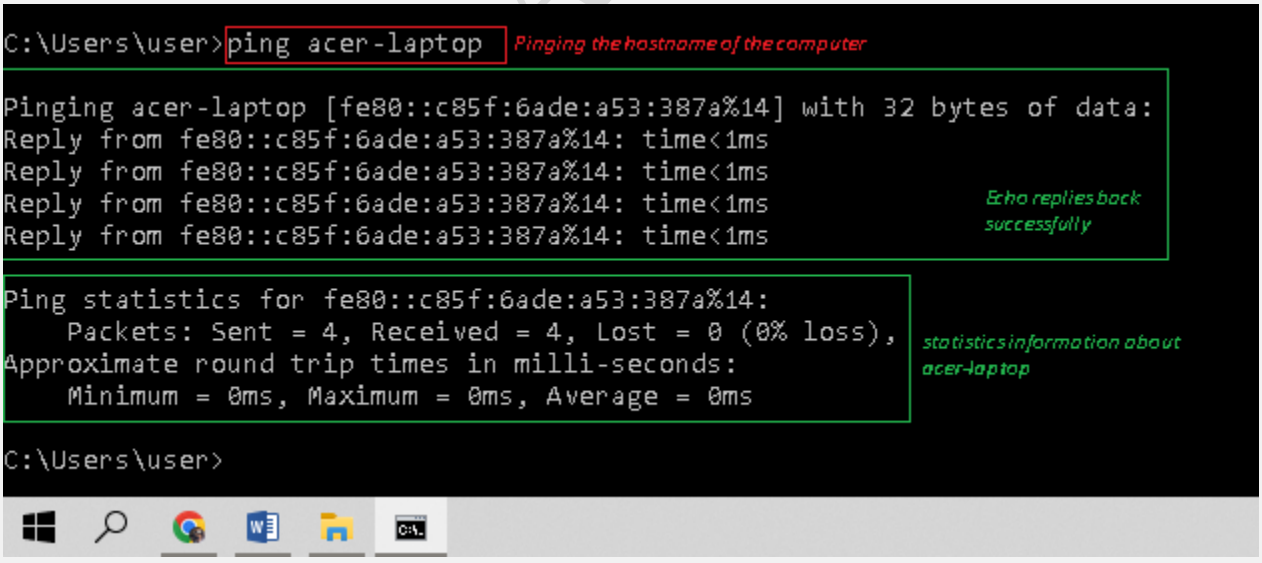
ping 127.0.0.1 ;

Yes, pinging loop-back is successful and it's screen shot is given below;

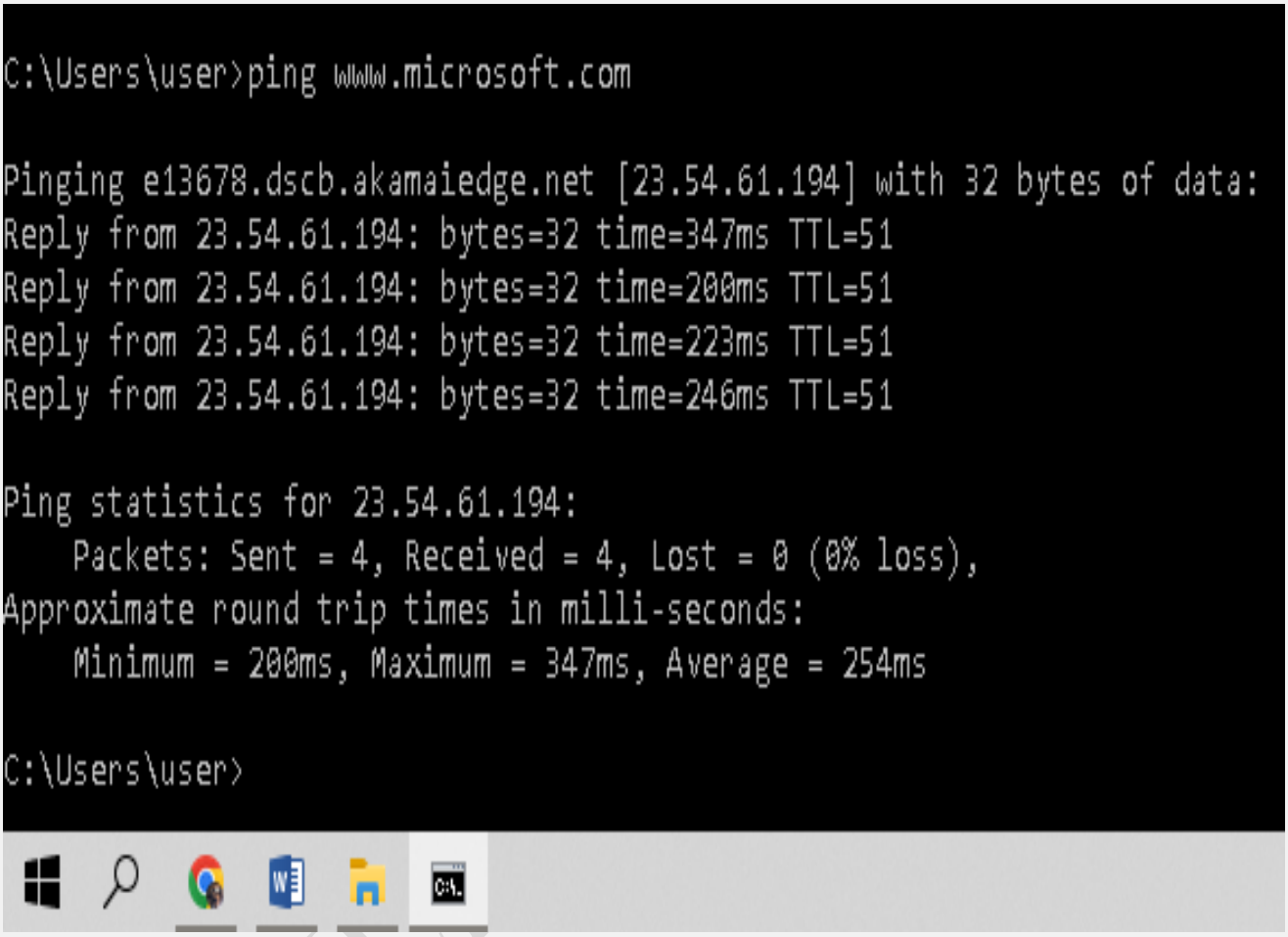


The address 127.0.0.1 is reserved for loopback testing. If the ping is successful, then TCP/IP is properly installed and functioning on this computer.

e. **Ping the hostname of the computer that you recorded in lab 1.1.**



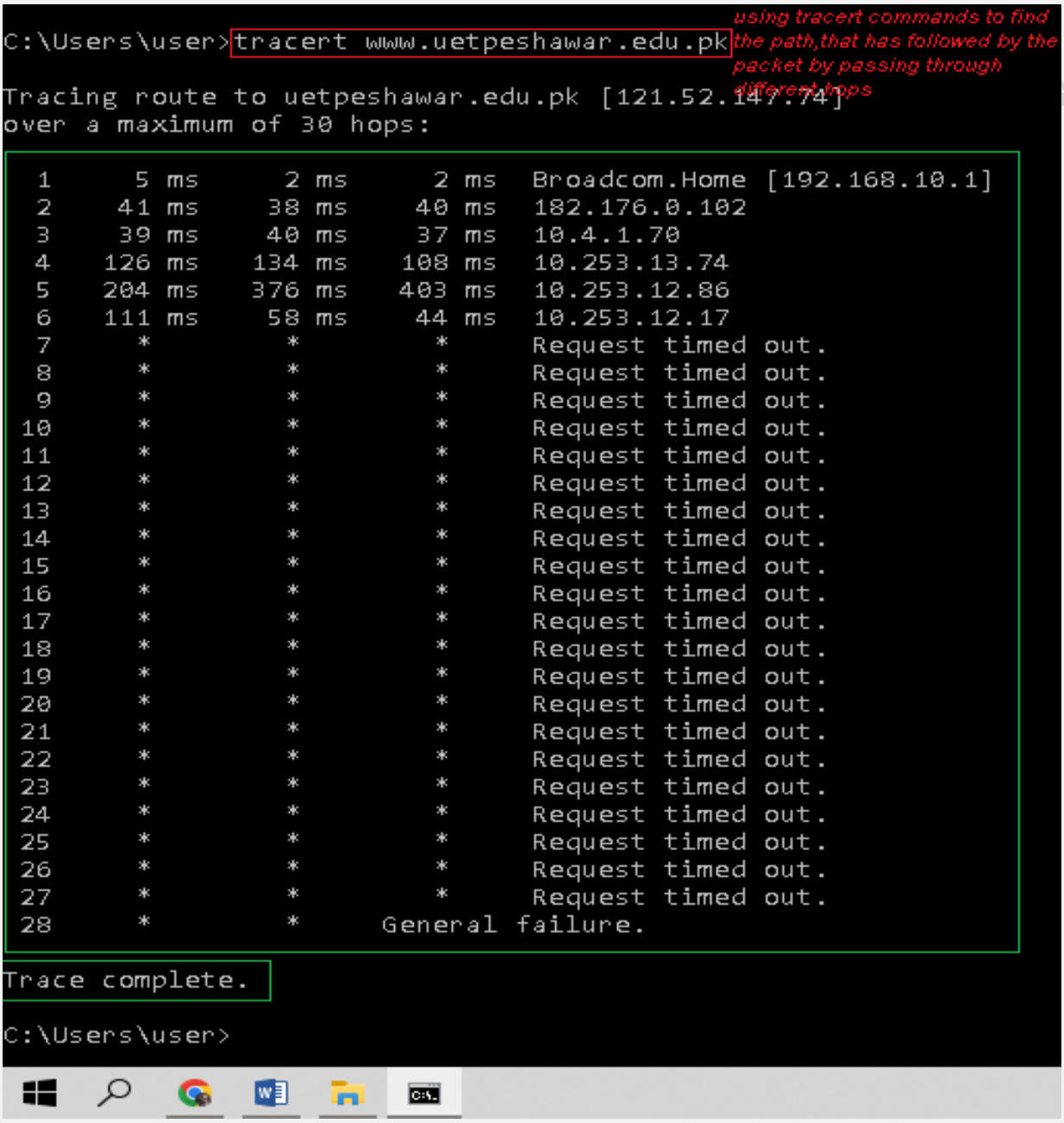
f. **Ping the Microsoft website(www.microsoft.com)**



Step 3.

- **Trace the route to the UET website: type tracert www.uetpeshawar.edu.pk and press Enter key.**

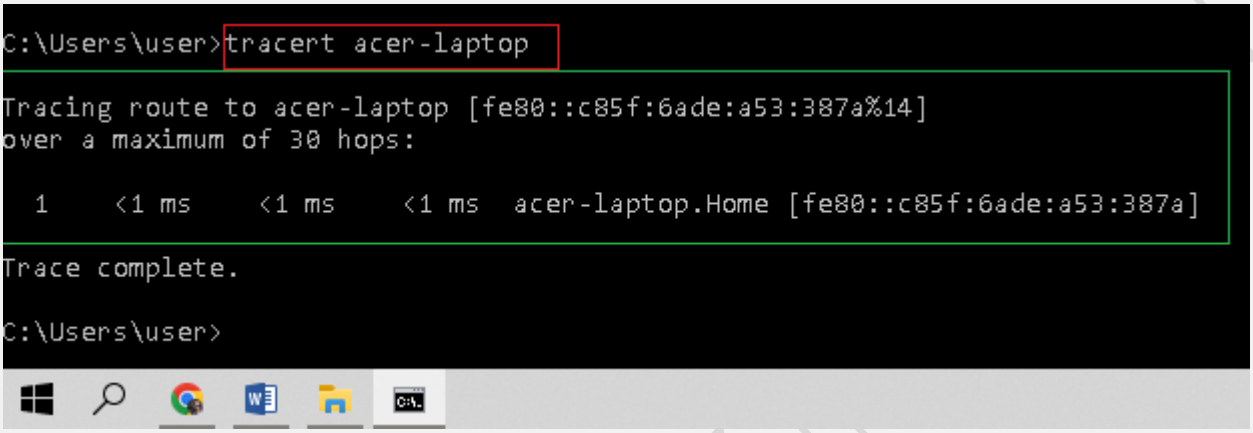
The result shows the complete route to the site. The number of hopes in path.



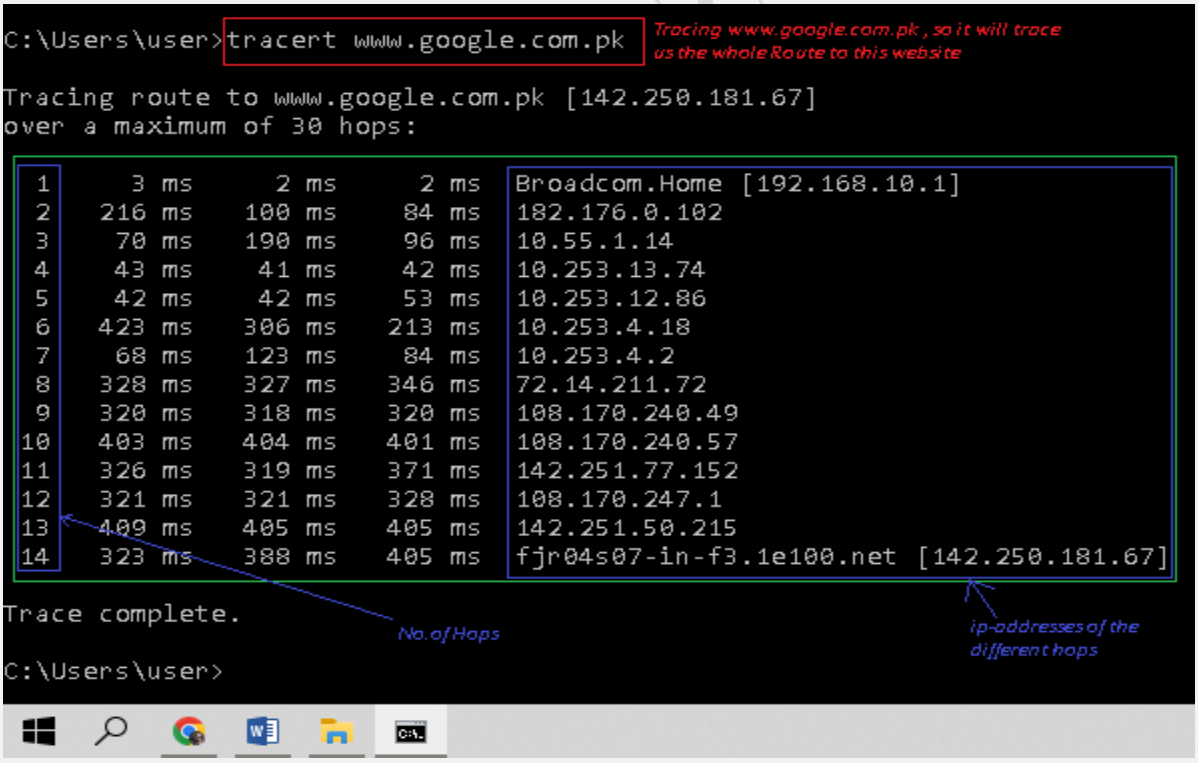
ii **Trace a local host name or IP address in your local area network (LAN).**

Record the output and interpret.

As I traced my own laptop so the result is given below;



For another site(Trace the route to the GOOGLE PAKISTAN website);



Step 4. Close the window.

DONE.

LAB NO.06

NAME : *ABDULLAH ZUNORAIN*

REG NO : *19JZELE0338*

SUBJECT : *COMPUTER COMMUNICATION NETWORK*

SUBMITTED TO : *DR. UZAIR GILLANI*

SECTION : *A*

DEPT : *ELECTRICAL COMM*

TITLE: AN INTRODUCTION TO WIRESHARK

Objectives

- Introduction to Packet Analyzer.
- Introduction to Wireshark.

Background

In this lab you will be introduced with Wireshark. Wireshark or Ethereal is a packet analyzer. A network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible. You could think of a network packet analyzer as a measuring device used to examine what's going on inside a network cable, just like a voltmeter is used by an electrician to examine what's going on inside an electric cable (but at a higher level, of course). Ethereal is perhaps one of the best open source packet analyzers available today. Here are some examples people uses Ethereal for:

- Network administrators use it to troubleshoot network problems
- Network security engineers use it to examine security problems
- Developers use it to debug protocol implementations
- People use it to learn network protocol internals

How does Wireshark Work

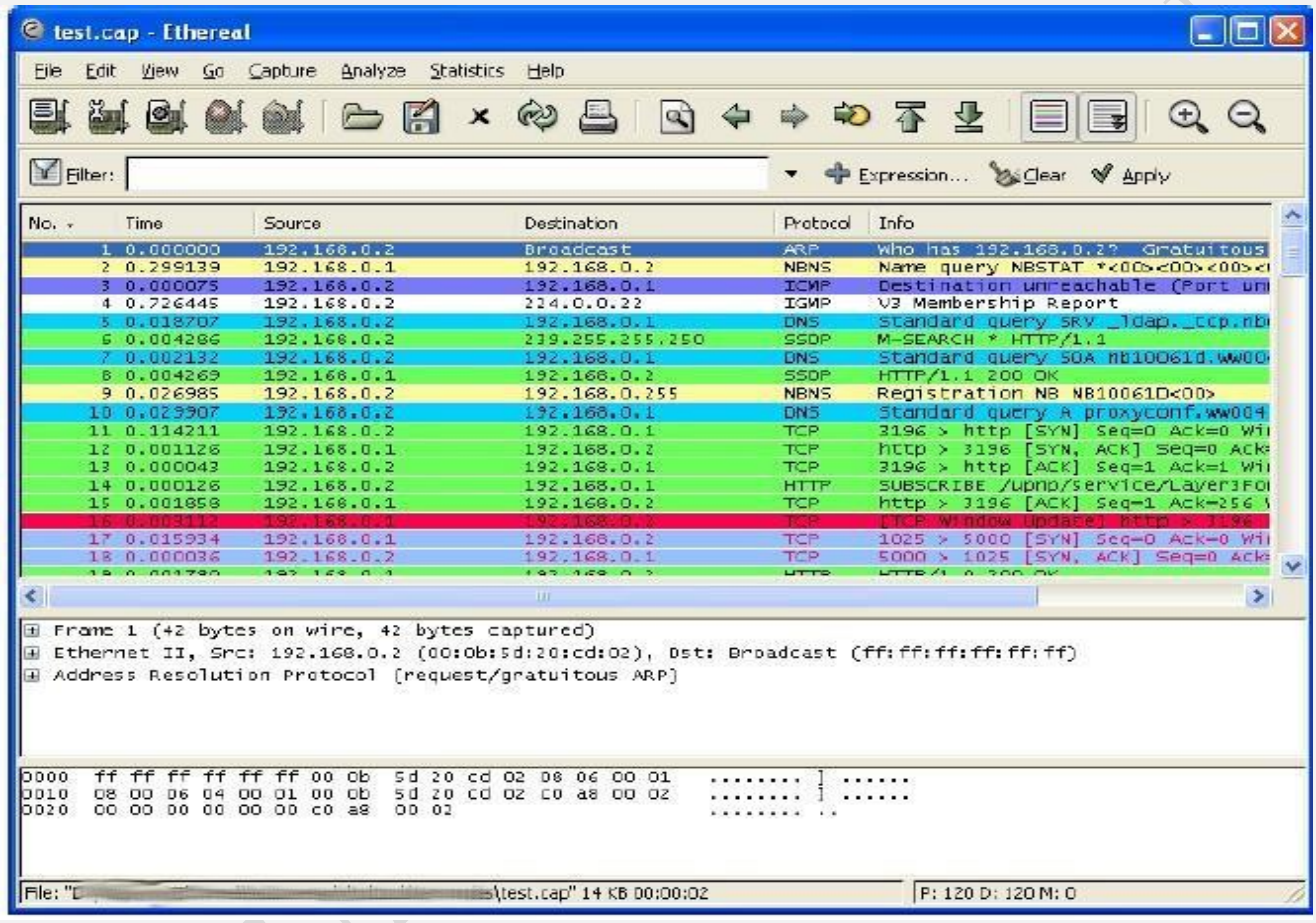
Packets entering or leaving a computer go through operating system that is responsible for handling them and passing them to the correct application. Licbpcap is a library having different functions to capture these packets from the operating system without any change at all. Wireshark uses this library to capture packets, store them temporarily in kernel memory an if desired save them onto stack.

Wireshark Interface

Ethereal's main window consists of parts that are commonly known from many other GUI programs.

1. The *menu* is used to start actions.
2. The *main toolbar* provides quick access to frequently used items from the menu.
3. The *filter toolbar* provides a way to directly manipulate the currently used display filter.
4. The *packet list pane* displays a summary of each packet captured. By clicking on packets in this pane you control what is displayed in the other two panes.
5. The *packet details pane* displays the packet selected in the packet list pane in more detail.

6. The *packet bytes pane* displays the data from the packet selected in the packet list pane, and highlights the field selected in the packet details pane.
7. The *status bar* shows some detailed information about the current program state and the captured data.



The Menu

It contains the following items:

- **File** This menu contains items to open and merge capture files, save / print / export capture files in whole or in part, and to quit from Ethereal.
- **Edit** This menu contains items to find a packet, time reference or mark one or more packets, set your preferences, (cut, copy, and paste are not presently implemented).
- **View** This menu controls the display of the captured data, including the colorization of packets, zooming the font, show a packet in a separate window, expand and collapse trees in packet details.

- **Go** This menu contains items to go to a specific packet.
- **Capture** This menu allows you to start and stop captures and to edit capture filters.
- **Analyze** This menu contains items to manipulate display filters, enable or disable the dissection of protocols, configure user specified decodes and follow a TCP stream.
- **Statistics** This menu contains menu-items to display various statistic windows, including a summary of the packets that have been captured, display protocol hierarchy statistics and much more.
- **Help** This menu contains items to help the user, like access to some basic help, a list of the supported protocols, manual pages, online access to some of the WebPages, and the usual about dialog.

The "Main" toolbar

The main toolbar provides quick access to frequently used items from the menu.

Toolbar Icon	Toolbar Item	Corresponding Menu Item	Description
	Interfaces...	Capture/Interfaces...	This item brings up the Capture Interfaces List dialog box
	Options...	Capture/Options...	This item brings up the Capture Options dialog box
	Start	Capture/Start	This item starts capturing packets with the options form the last time.
	Stop	Capture/Stop	This item stops the currently running live capture process
	Restart	Capture/Restart	This item stops the currently running live capture process and re-starts it again, for convenience.

	Open...	File/Open...	This item brings up the file open dialog box that allows you to load a capture file for viewing.
	Save As...	File/Save As...	This item allows you to save the current capture file to whatever file you would like. It pops up the Save Capture File As dialog box

The "Filter" toolbar

The filter toolbar lets you quickly edit and apply display filters

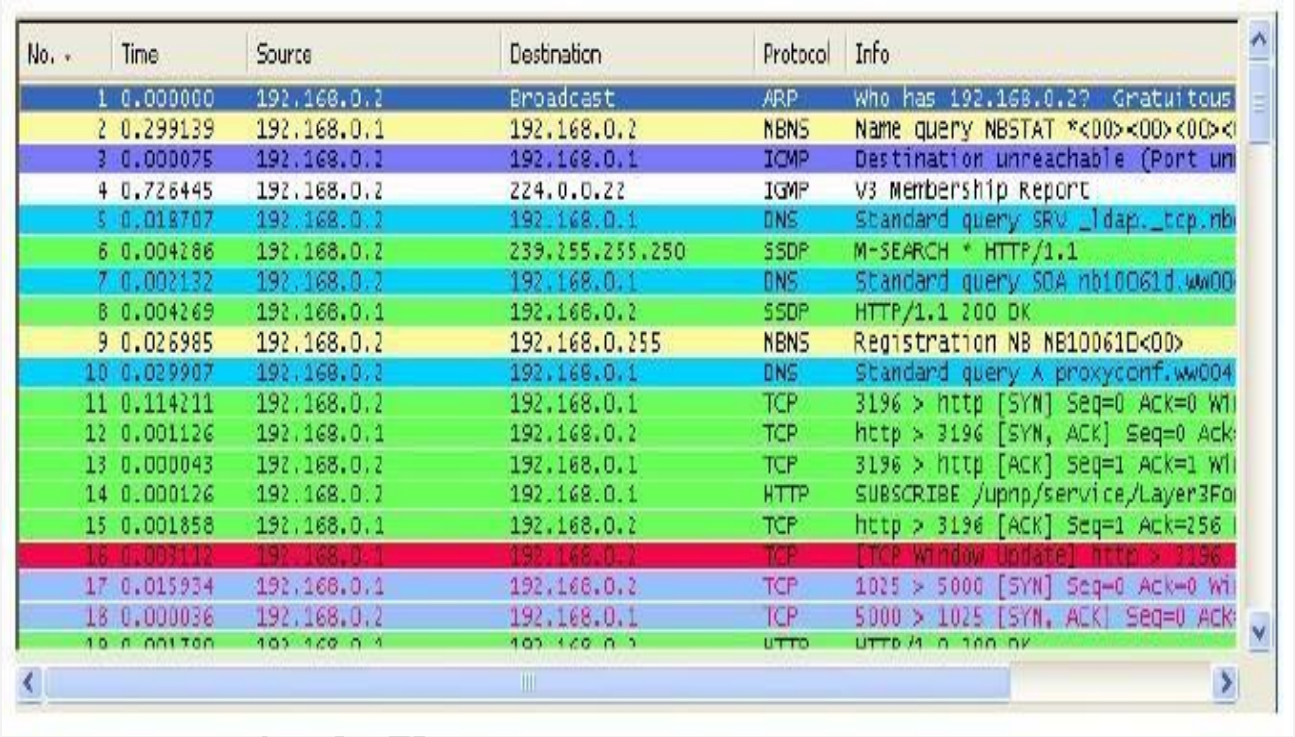
• The leftmost button labeled "Filter:" can be clicked to bring up the filter construction dialog,

• The left middle text box provides an area to enter or edit display filter stringsA syntax check of your filter string is done while you are typing. The background will turn red if you enter an incomplete or

invalid string, and will become green when you enter a valid string. You can click on the pull down arrow to select a previously-entered filter string from a list. The entries in the pull down list will remain available even after a program restart.

The "Packet List" pane

The packet list pane displays all the packets in the current capture file. Each line in the packet list corresponds to one packet in the capture file. If you select a line in this pane, more details will be displayed in the "Packet Details" and "Packet Bytes" panes.



The "Packet Details" pane

The packet details pane shows the current packet (selected in the "Packet List" pane) in a more detailed form.

This pane shows the protocols and protocol fields of the packet selected in the "Packet List" pane. The protocols and fields of the packet are displayed using a tree, which can be expanded and collapsed.

The "Packet Bytes" pane

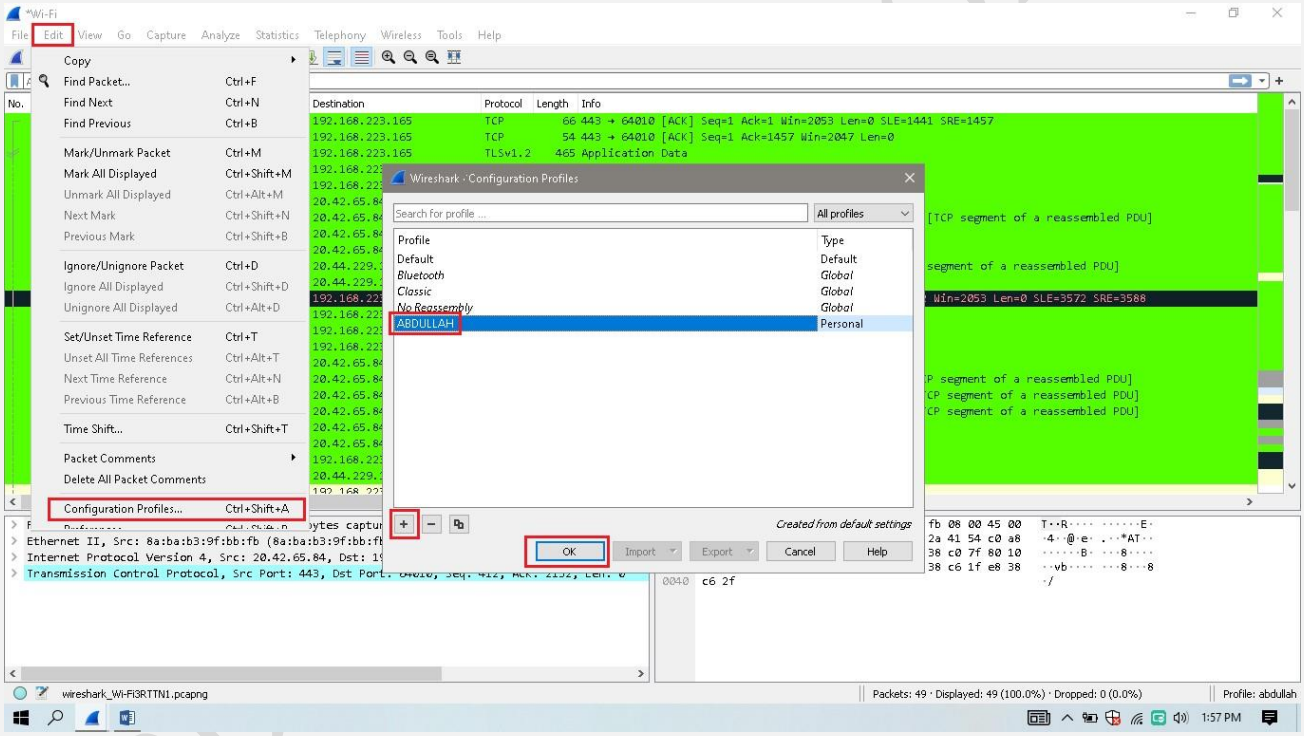
The packet bytes pane shows the data of the current packet (selected in the "Packet List" pane) in a hexdump style. As usual for a hexdump, the left side shows the offset in the packet data, in the middle the packet data is shown in a hexadecimal representation and on the right the corresponding ASCII characters (or . if not appropriate) are displayed.

LAB TASKS

1. **Get use to the short cuts on the toolbar.**

   a) ***CREATING PROFILE*** :

   1st of all we have to go to the **Menu_edit** then from Menu edit we have to select the **Configuration-profile** and then select the **Plus-icon** for creating new profile .



Now any change or setting that we are going to made, will only remain confined to that particular profile that we created.

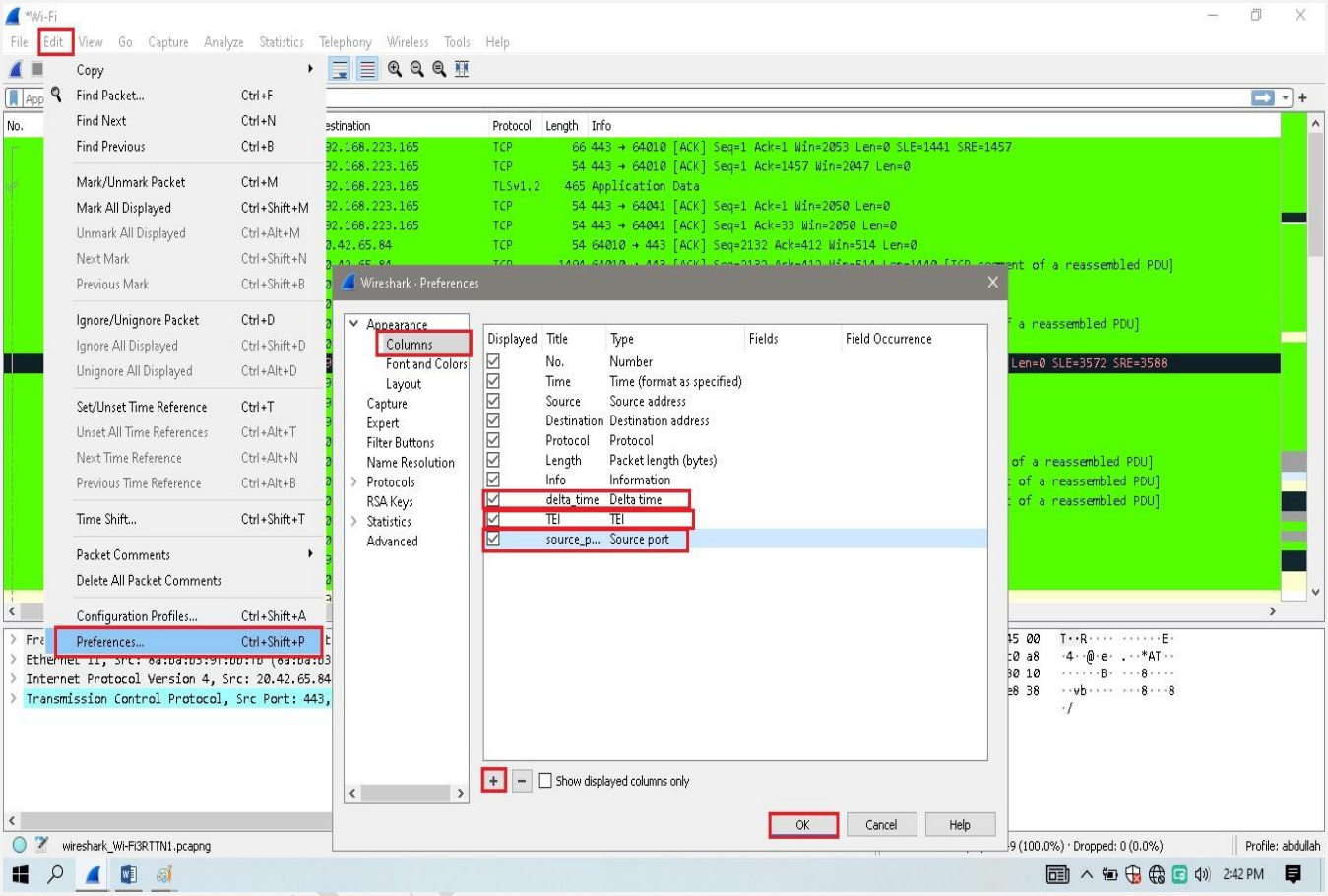
b) ***ADDING OR REMOVING COLOUMNS :***

We can also adding or removing coloumns for our convenient, if we are trying to troubleshoot a problem and to get more information about packets.

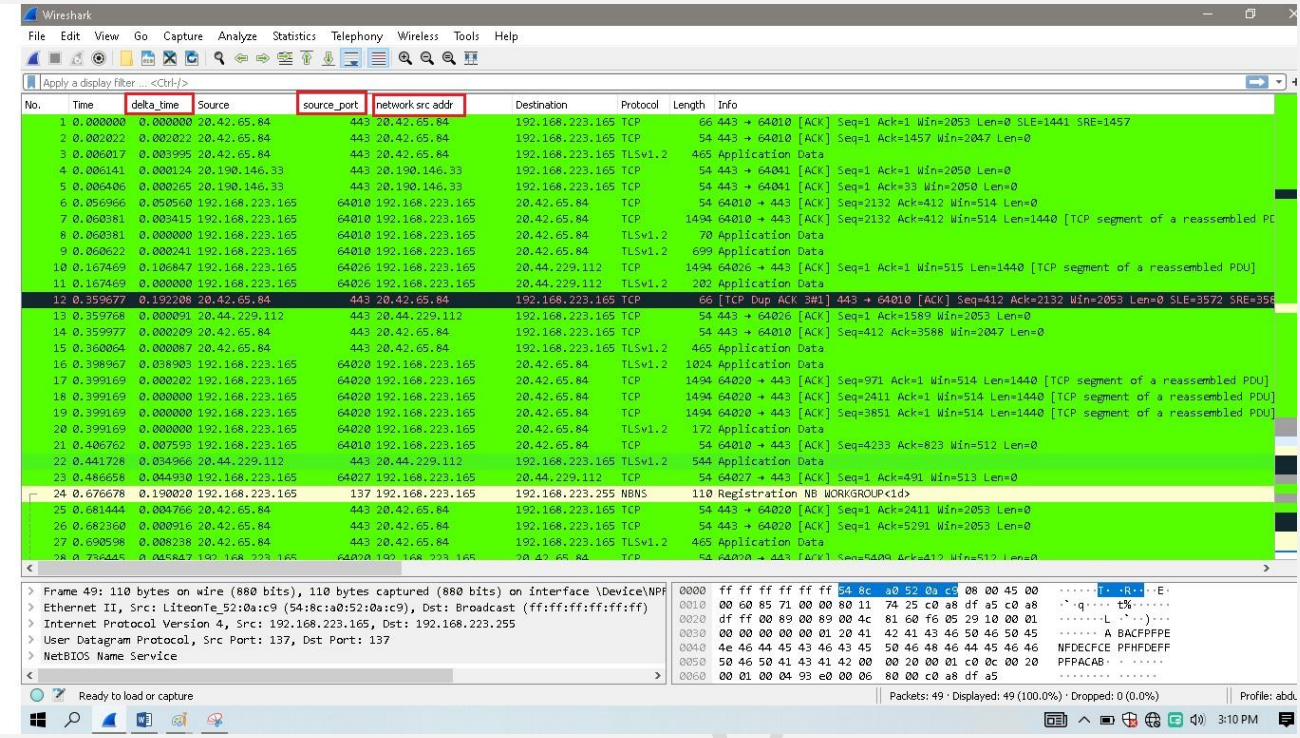
Now if we want to add some coloumns, we have to go to the **Menu_edit** then from edit we have to select the **Preferences ,** then in the preferences screen we have to go to the **Appearance** , and then

select the **Coloums** , and here we have to add more coloumns that were needed in troubleshooting or for other information.

*As shown in the below **Screenshots** ;*

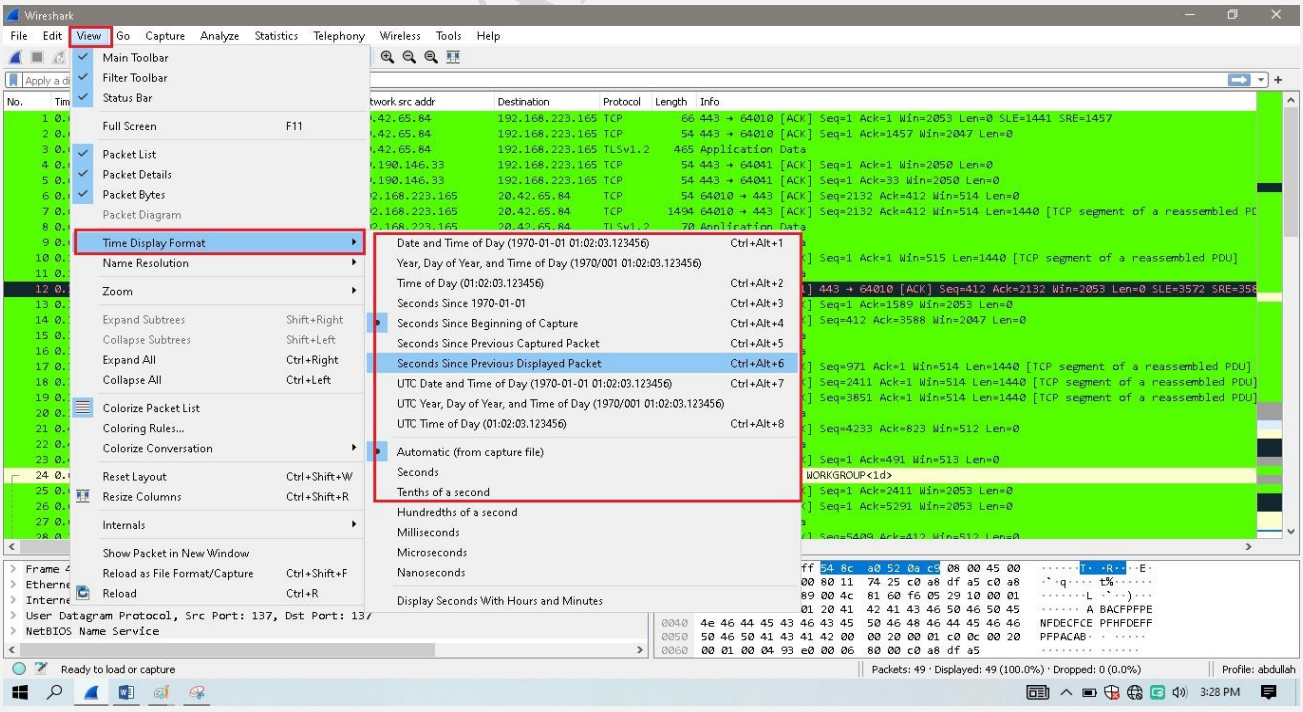


Now in the next Screenshot we had shown the added coloumns (delta_time , source port , network_src addr) below;



c) *TIME FORMAT :*

We can also changes the Time_Format from the **Menu_view ,** then from the view select the **Display_Time_Format ,** and then we have to choose the desired format of the time as shown in the below screenshot.



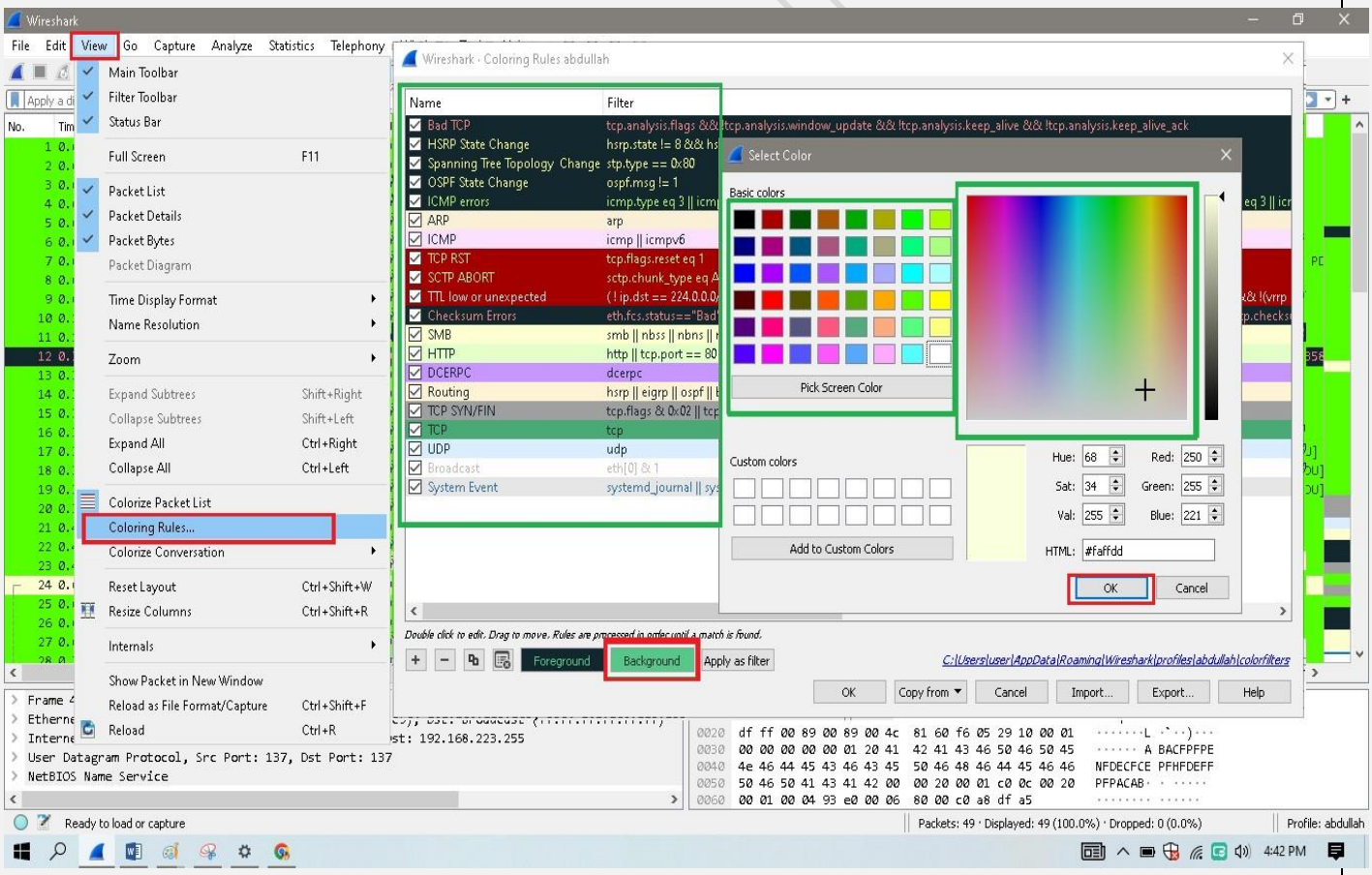
d) PACKET COLORIZATION:

A very useful mechanism available in Wireshark is packet colorization. You can set up Wireshark so that it will colorize packets according to a display filter. This allows you to emphasize the packets you might be interested in.

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.**

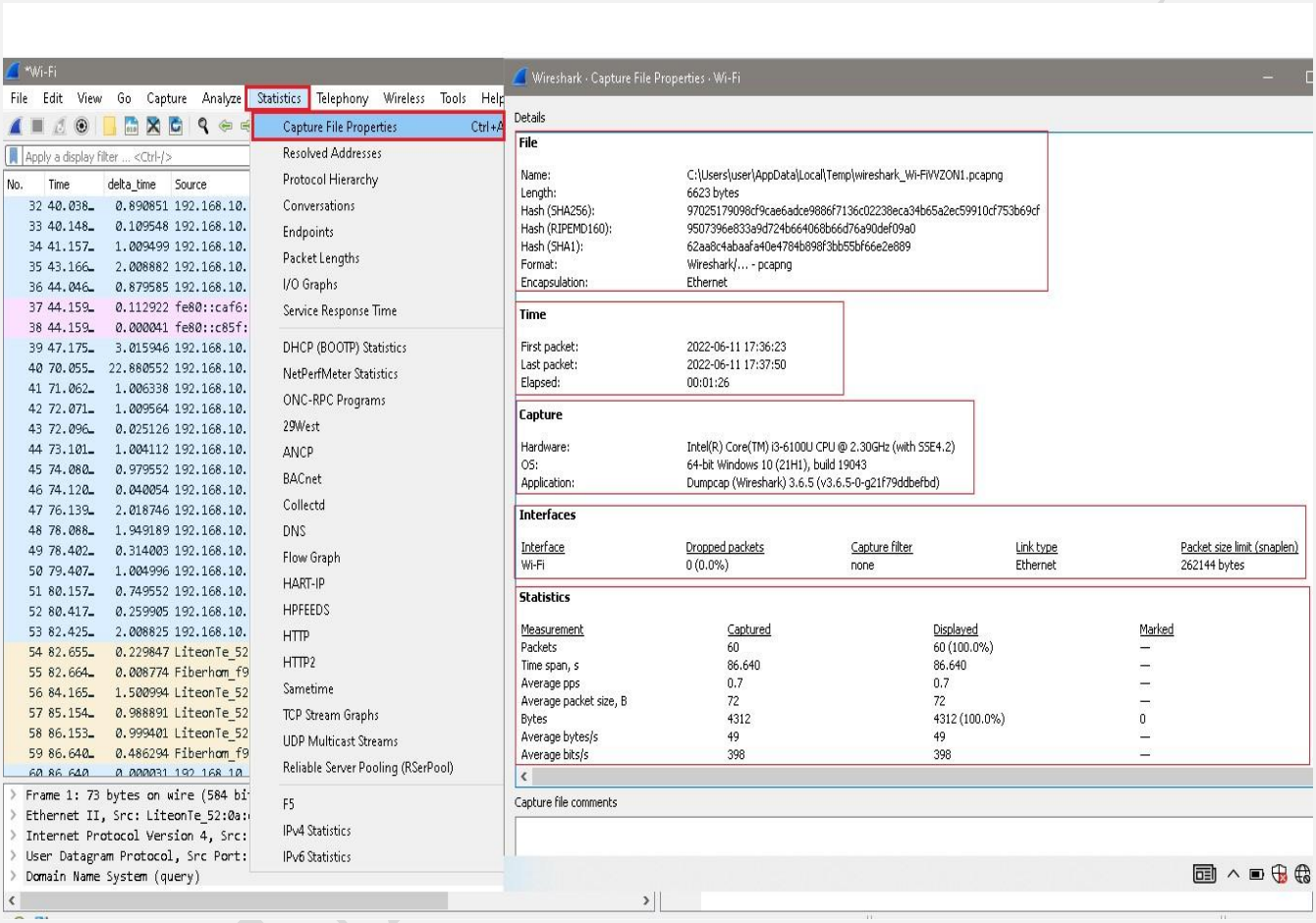
To permanently colorize packets, select **View → Coloring Rules**.... Wireshark will display the **"Coloring Rules" dialog box** ;

As shown in Figure below;



e) CAPTURE FILE PROPERTIES :

1ST we have to go to the Statistics_Menu and then select the Capture_File_Properties as shown in the below fig;



The above dialog shows the following information:

❖ Details

Notable information about the capture file.

❖ File

General information about the capture file, including its full path, size, cryptographic hashes, file format, and encapsulation.

❖ Time

The timestamps of the first and the last packet in the file along with their difference.

❖ Capture

Information about the capture environment. This will only be shown for live captures or if this information is present in a saved capture file. The pcapng format supports this, while pcap doesn't.

❖ Interfaces

Information about the capture interface or interfaces.

❖ Statistics

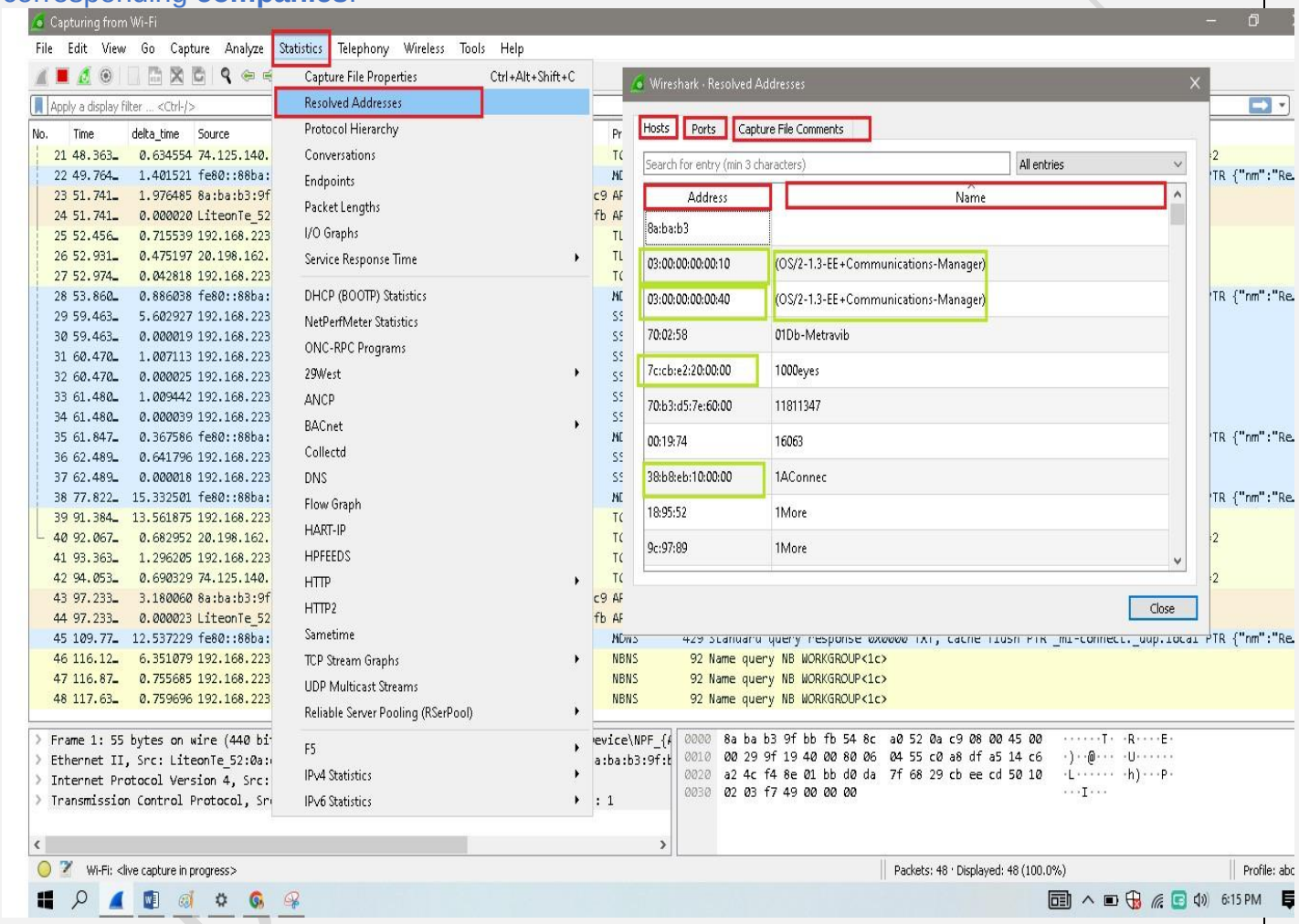
A statistical summary of the capture file. If a display filter is set, you will see values in the *Captured* column, and if any packets are marked, you will see values in the *Marked* column. The values in the *Captured* column will remain the same as before, while the values in the *Displayed* column will reflect the values corresponding to the packets shown in the display. The values in the *Marked* column will reflect the values corresponding to the marked packages.

❖ Capture file comments

Some capture file formats (notably pcapng) allow a text comment for the entire file. You can view and edit this comment here.

f) RESOLVED ADDRESSES:

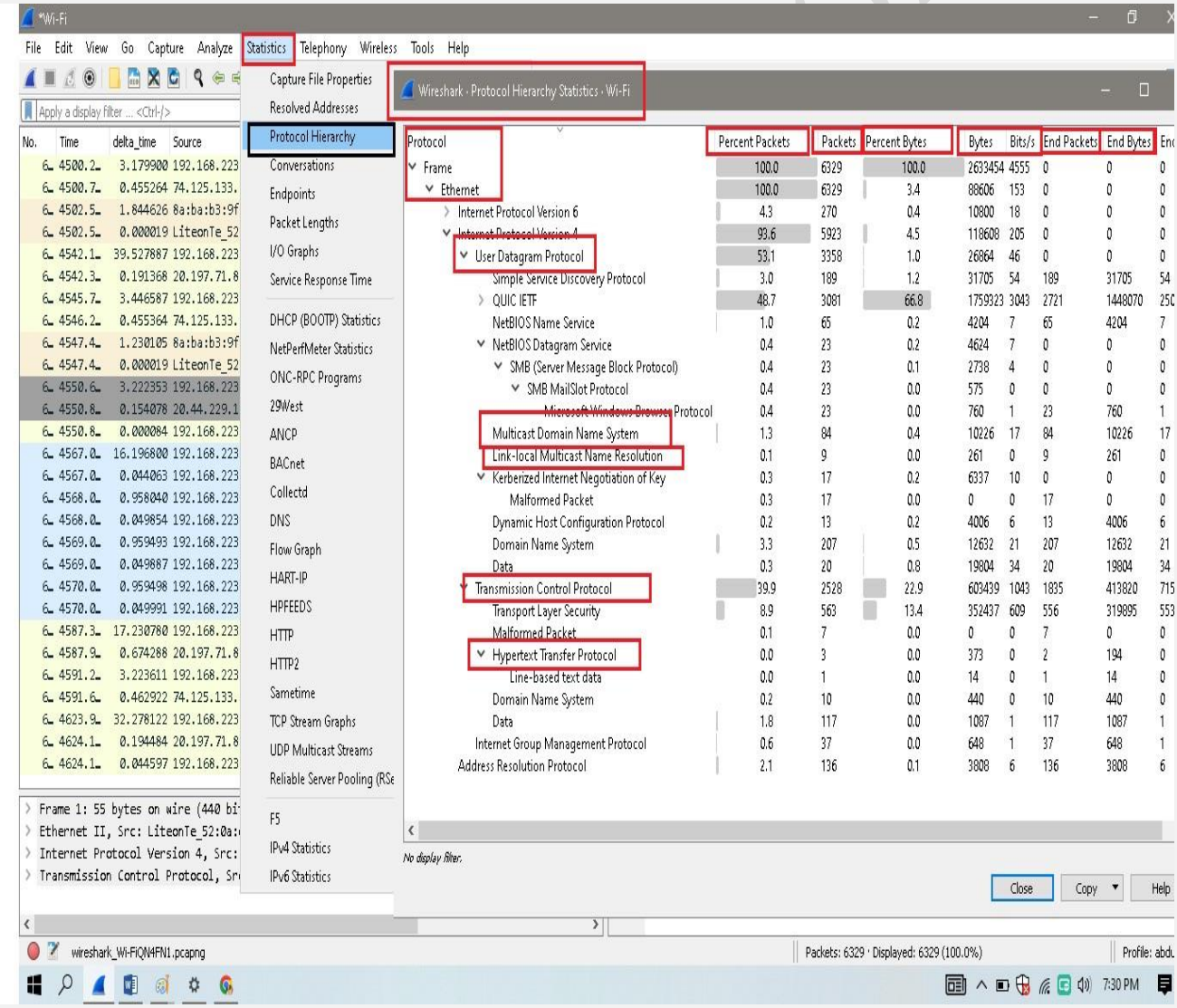
In order to access the statistics in Wireshark, click on **Statistics** and go to **Resolved Addresses:**
The Resolved Addresses window will give you a list at the top of all of the **IP addresses** and **DN$ names** that were resolved in your **packet capture.** It also gives us the **mac_adresses** and their corresponding **companies**.



g) ***PROTOCOL HIERARCHY:***

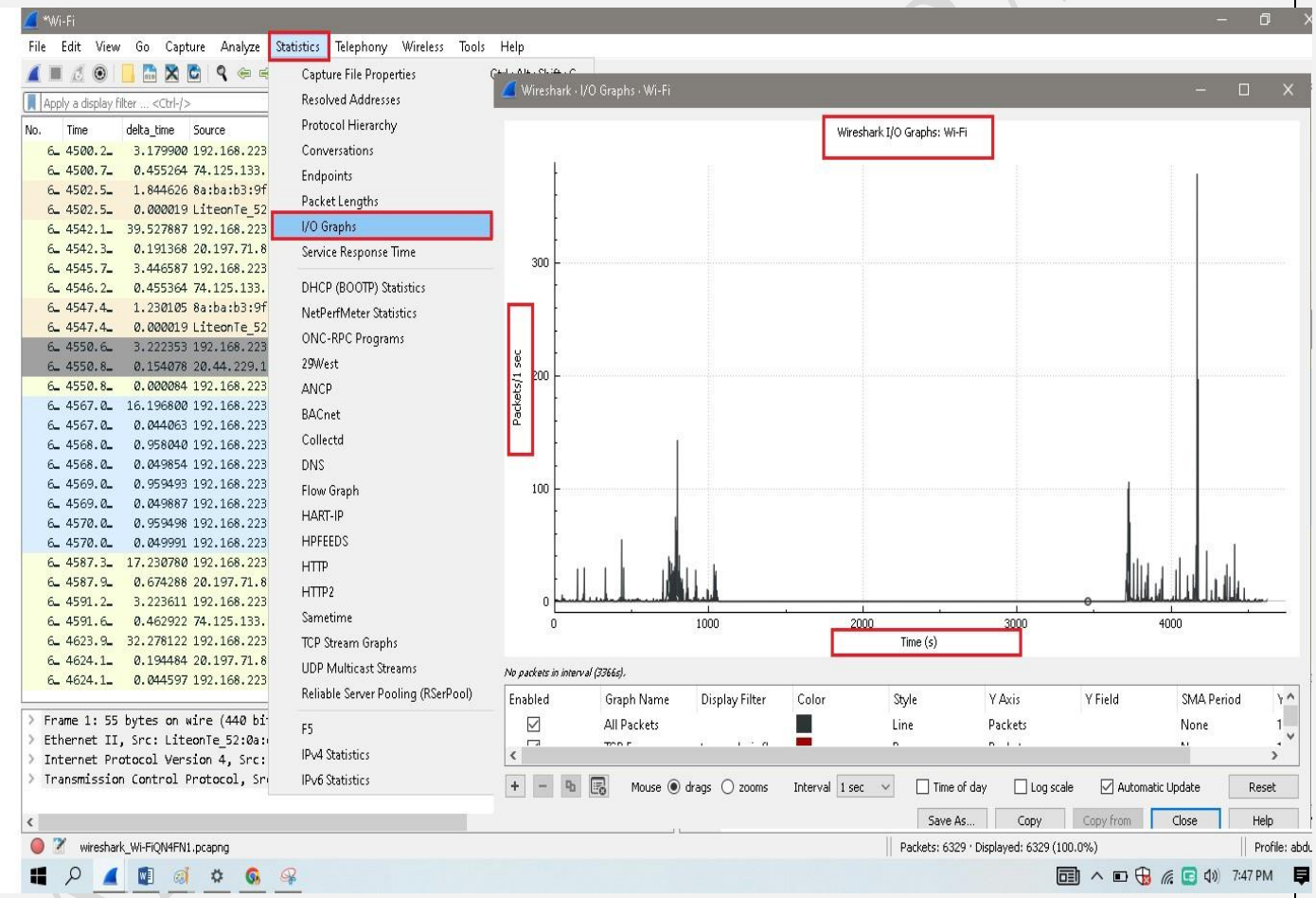
The protocol hierarchy of the captured packets. This is **a tree of all the protocols in the capture**. Each row contains the statistical values of one protocol. Two of the columns (Percent Packets and Percent Bytes) serve double duty as bar graphs.

You can generate a protocol hierarchy chart in Wireshark by **selecting the Protocol Hierarchy option from the Statistics drop-down menu**.



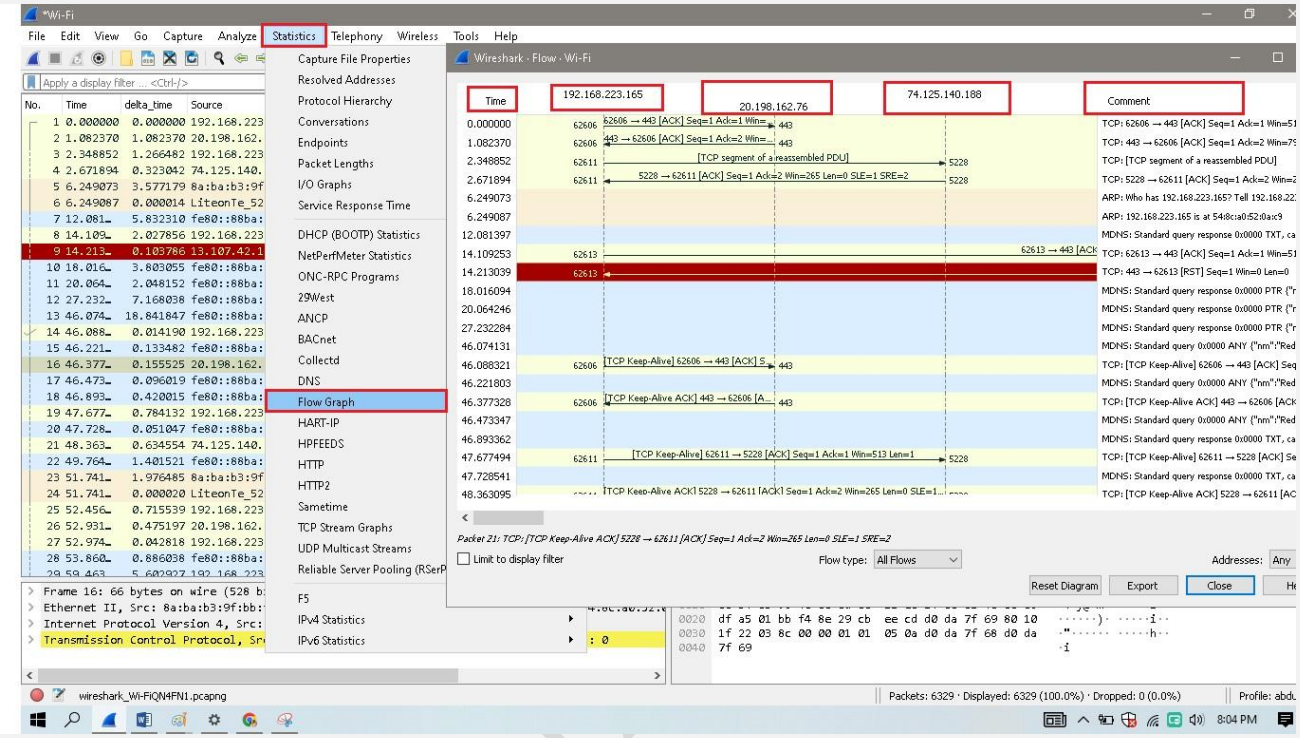
h) I/O GRAPH :

Wireshark IO Graphs will **show you the overall traffic seen in a capture file which is usually measured in rate per second in bytes or packets** (which you can always change if you prefer bits/bytes per second). In default the x-axis is the tick interval per second, and y-axis is the packets per tick (per second).



i) FLOW GRAPH :

The **Flow Graph** window shows connections between hosts. It displays the packet time, direction, ports and comments for each captured connection.



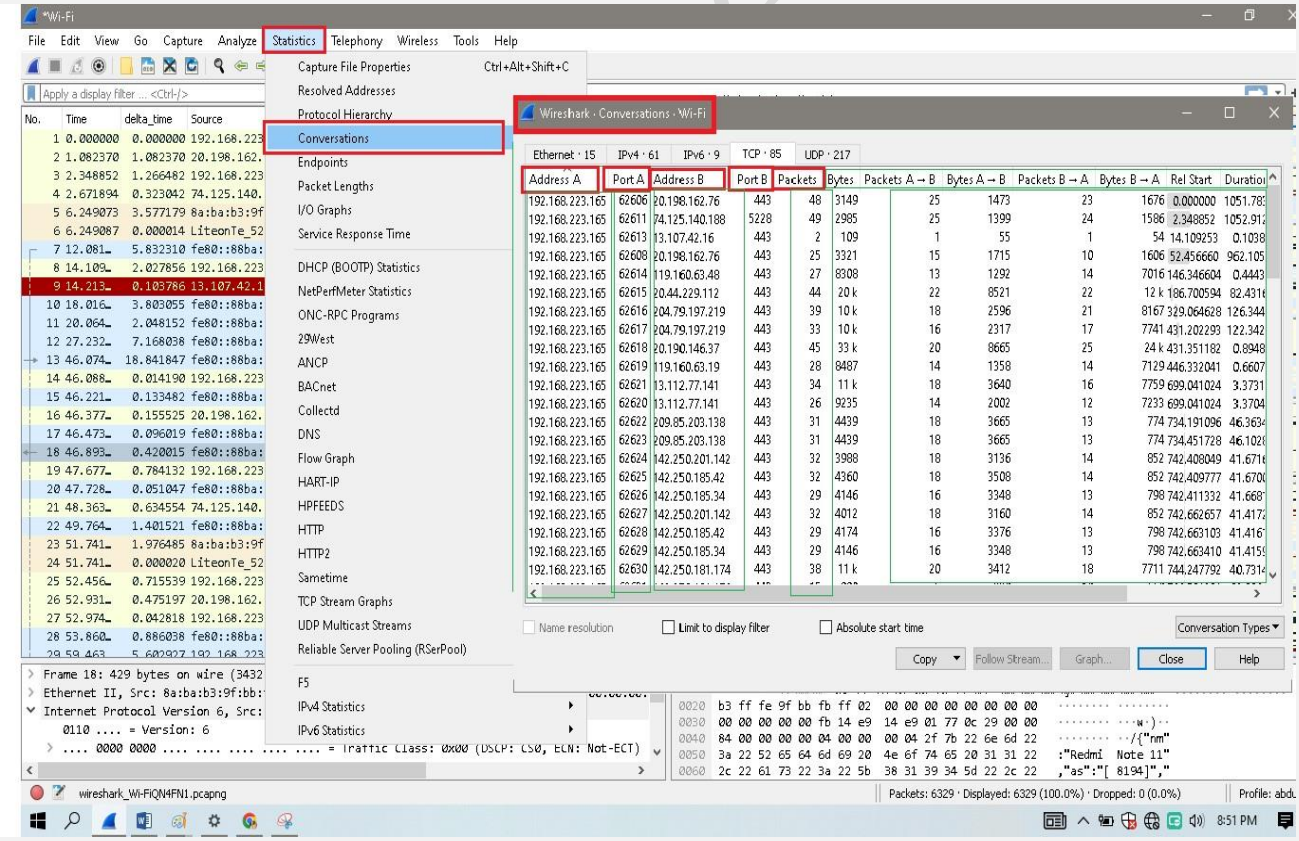
- The numbers in each row at the very left of the window represent the time packet.
- The numbers at the both ends of each arrow between hosts represent the port numbers.
- Left-click a row to select a corresponding packet in the packet list.
- Right-click on the graph for additional options, such as selecting the previous, current, or next packet in the packet list. This menu also contains shortcuts for moving the diagram.

j) CONVERSATION:

A network conversation is the traffic between two specific endpoints. For-example, an IP conversation is all the traffic between two IP addresses.

The conversations window is similar to the endpoint Window. Along with addresses, packet counters, and byte counters the conversation window adds four columns: the start time of the conversation (“Rel Start”) or (“Abs Start”), the duration of the conversation in seconds, and the average bits (not bytes) per second in each direction. A timeline graph is also drawn across the “Rel Start” / “Abs Start” and “Duration” columns. Each row in the list shows the statistical values for exactly one conversation.

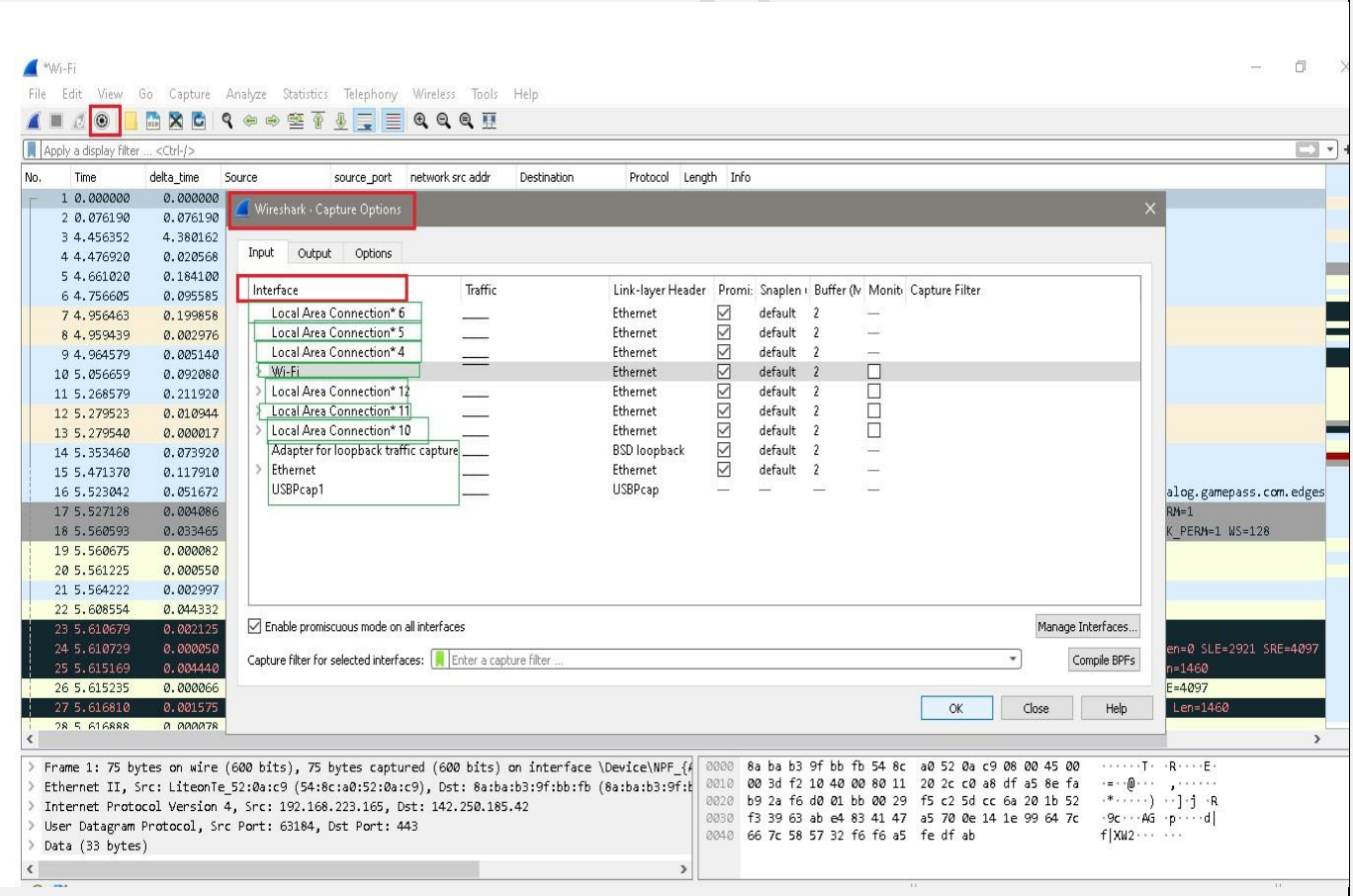
Name resolution will be done if selected in the window and if it is active for the specific protocol layer (MAC layer for the selected Ethernet endpoints page). *Limit to display filter* will only show conversations matching the current display filter. *Absolute start time* switches the start time column between relative (“Rel Start”) and absolute (“Abs Start”) times. Relative start times match the “Seconds Since First Captured Packet” time display format in the packet list and absolute start times match the “Time of Day” display format.



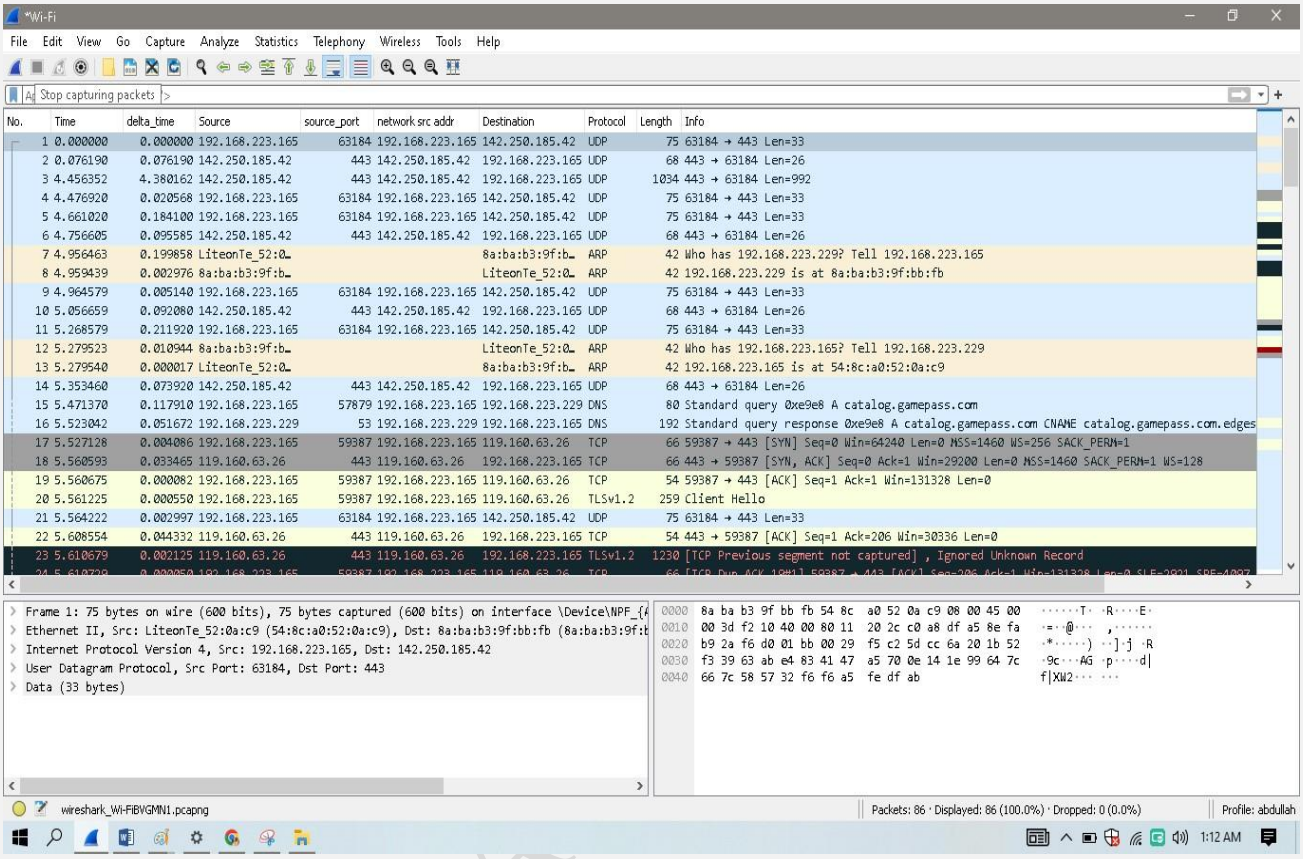
2. *Capture packets on a particular interface, find out what protocols have been use and save them in a file :*

A. Capture packets on a particular interface :

To capture packets from the wire, you can select Capture > Interfaces from the main drop-down menu. This will show all of the interfaces on the system as shown in figure. Here you can choose to capture packets from a sensor interface or another interface. To begin capturing packets from a particular interface, click on that interface.



As shown below in the fig the packets are capturized;



When you've finished collecting packets, click the Stop button under the Capture dropdown menu. At this point, you should be presented with data to be analyzed.

Looking at the image above, you will notice that Wireshark is divided into three panes.

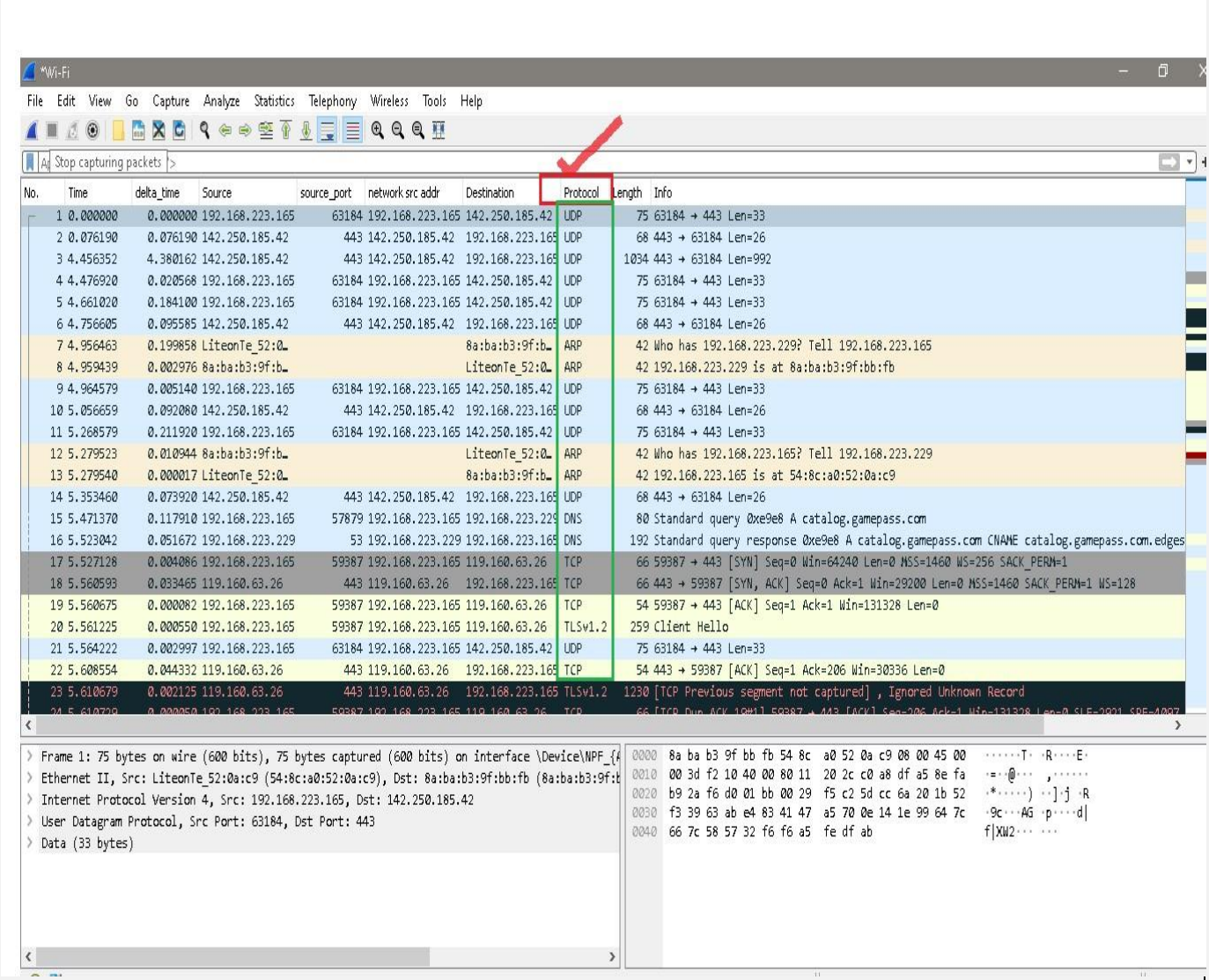
- The **uppermost** is the packet list pane, which shows each packet summarized into a single line, with individual **fields separated as columns**. The **default columns** include a **packet number**, a **timestamp** (defaulting to the time since the beginning of the capture), **source** and **destination address**, **protocol**, **packet length**, and an **info** column that contains protocol-specific information.
- The **middle pane** is the **packet details pane**, and shows detailed information about the data fields contained within the packet that is selected in the packet list pane.
- The **bottom pane** is the **packet bytes pane**, and details the individual bytes that comprise a packet, shown in hex and ASCII format.

The important thing to note when interacting with these three panes is that the data that each one displays is linked to actions taken in the other panes. When you click on a packet in the **packet list pane**, it shows data related to that packet in the **packet details** and

packet bytes panes. Furthermore, when you click on a field in the packet details pane, it will highlight the bytes associated with that field in the packet bytes pane. This is ideal for visually bouncing around to different packets and determining their properties quickly.

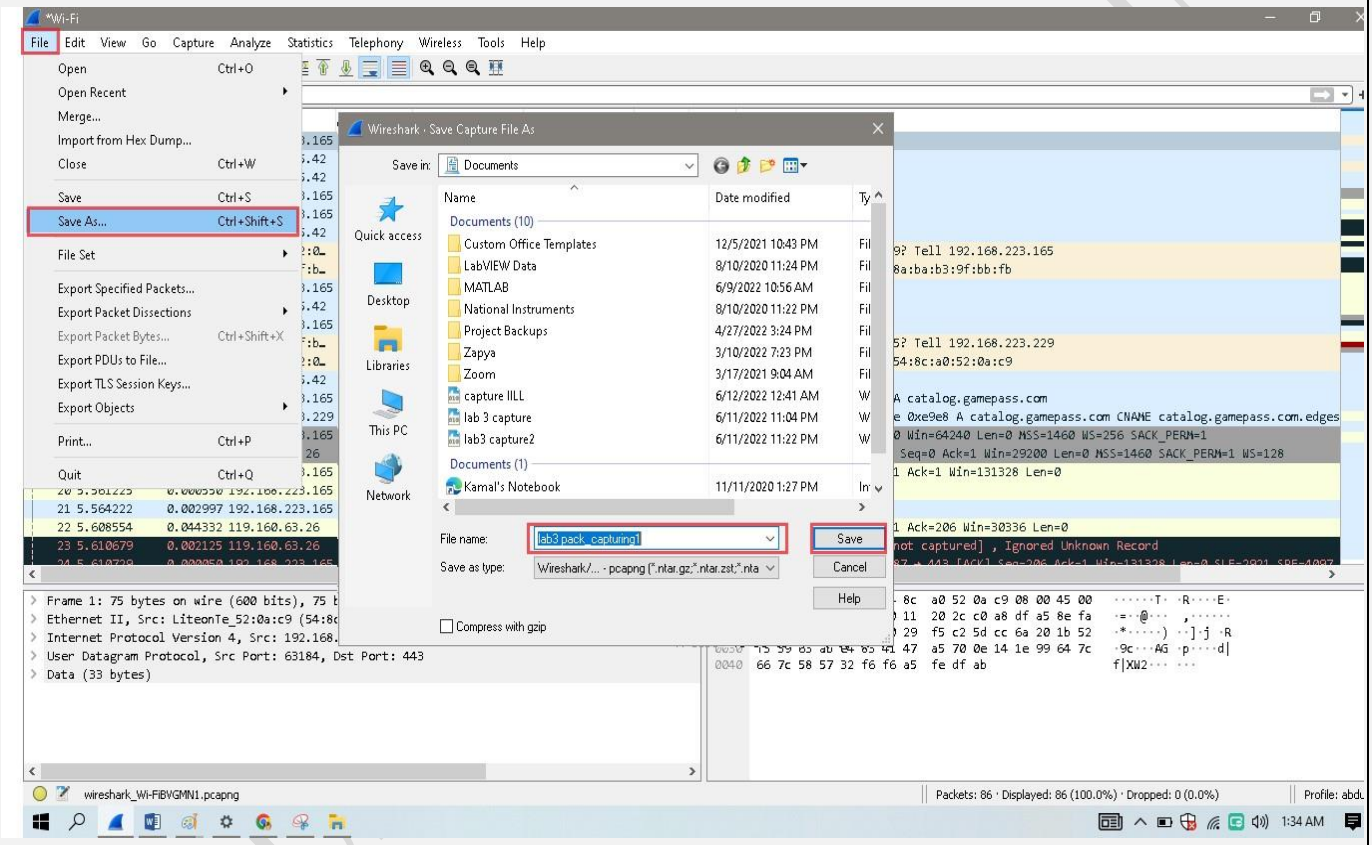
B. PROTOCOL THAT HAS BEEN USED :

There are several protocols are used in capturing packets as shown in fig below;



C. SAVE THESE CAPTURING PACKETS INTO FILE:

Now save these capturing packets into file by the procedure shown in the figure;



Ist of all we have to go to the **Menu_File** then select the **Save_As** then go to the **File_Name** to select a name for that file.

LAB NO.07

NAME : ABDULLAH ZUNORAIN

REG NO : 19JZELE0338

SUBJECT : COMPUTER COMMUNICATION NETWORK

SUBMITTED TO : DR. UZAIR GILLANI

SECTION : A

TITLE: CAPTURING DATA THROUGH WIRESHARK

Objective

- Set Wireshark ready to capture packets.
- Arrange results using filters
- Analyze captured packets

Background

Capturing live network data is one of the major features of Ethereal. The Ethereal capture engine provides the following features:

- Capture from different kinds of network hardware (Ethernet, Token Ring, ATM,).
- Stop the capture on different triggers like: amount of captured data, captured time, captured number of packets.
- Simultaneously show decoded packets while keep on capturing.
- Filter packets, reducing the amount of data to be captured.
- Capturing into multiple files while doing a long term capture, and in addition the option to form a ring buffer of these files, keeping only the last x files, useful for a "very long term" capture.

The "Capture Interfaces" dialog box

When you select "Interfaces..." from the Capture menu, Ethereal pops up the "Capture Interfaces" dialog Box. This dialog box will only show the local interfaces Ethereal knows of. As Ethereal might not be able to detect all local interfaces, and it cannot detect the remote interfaces available, there could be more capture interfaces available than listed.

Description: The interface description provided by the operating system.

IP: The first IP address Ethereal could resolve from this interface. If no address could be resolved (e.g. no DHCP server available), "unknown" will be displayed. If more than one IP address could be resolved, only the first is shown (unpredictable which one in that case).

Packets: The number of packets captured from this interface, since this dialog was opened will be grayed out, if no packet was captured in the last second.

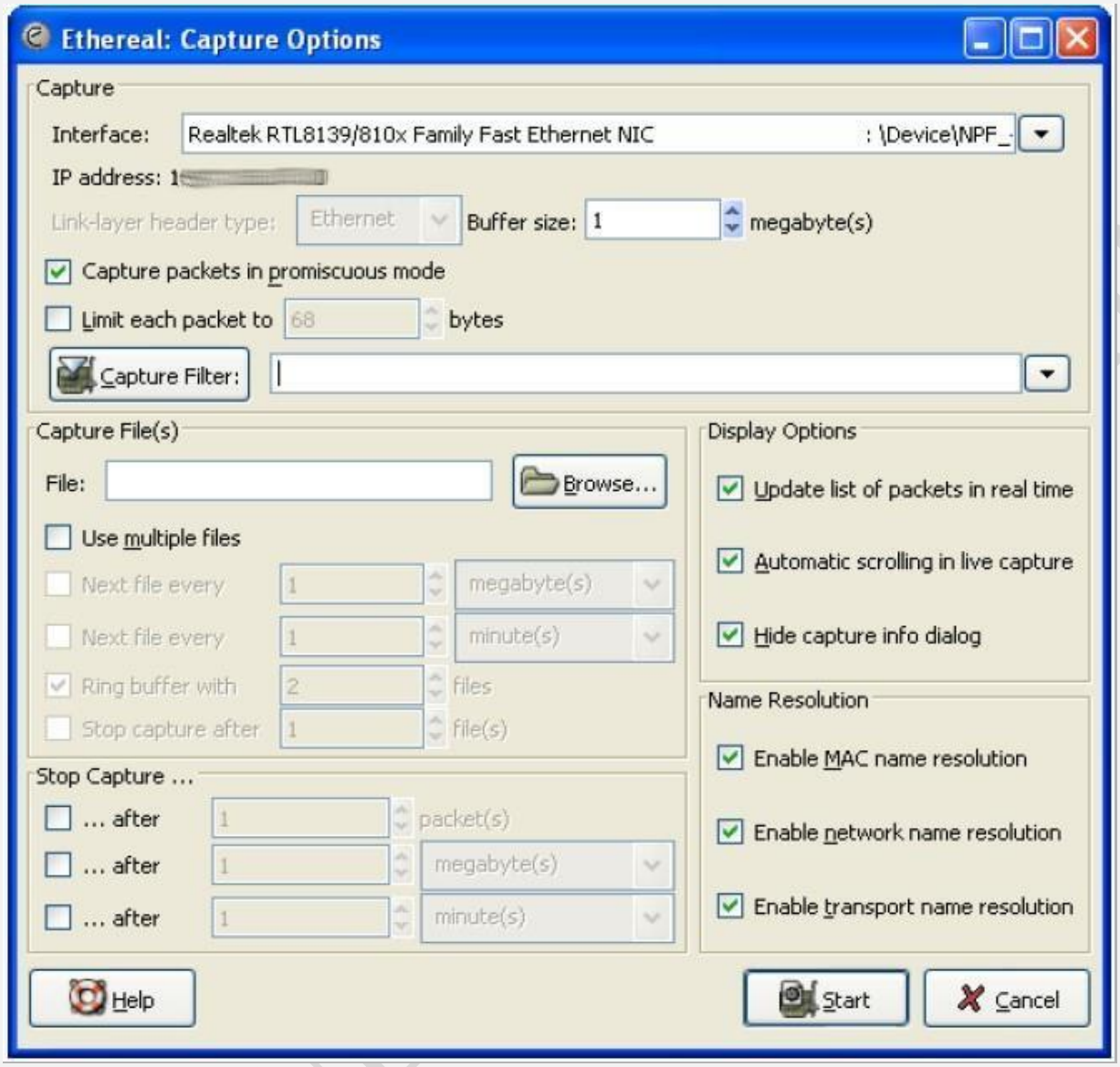
Packets/s: Number of packets captured in the last second will be grayed out, if no packet was captured in the last second.

Stop: Stop a currently running capture.

Capture: Start a capture on this interface immediately, using the settings from the last capture.

Prepare: Open the Capture Options dialog with this interface selected,.

Close: Close this dialog box.



Interface: This field specifies the interface you want to capture on. You can only capture on one interface, and you can only capture on interfaces that Ethereal has found on the system. It is a dropdown list, so simply click on the button on the right hand side and select the interface you want. **IP address:** The IP address of the selected interface. If no address could be resolved from the system, "unknown" will be shown.

Link-layer header type: Unless you are in the rare situation that you need this, just keep the default.

Buffer size: n megabyte(s): Enter the buffer size to be used while capturing. This is the size of kernel buffer which will keep the captured packets, until they are written to disk. If you encounter packet drops, try increasing this value.

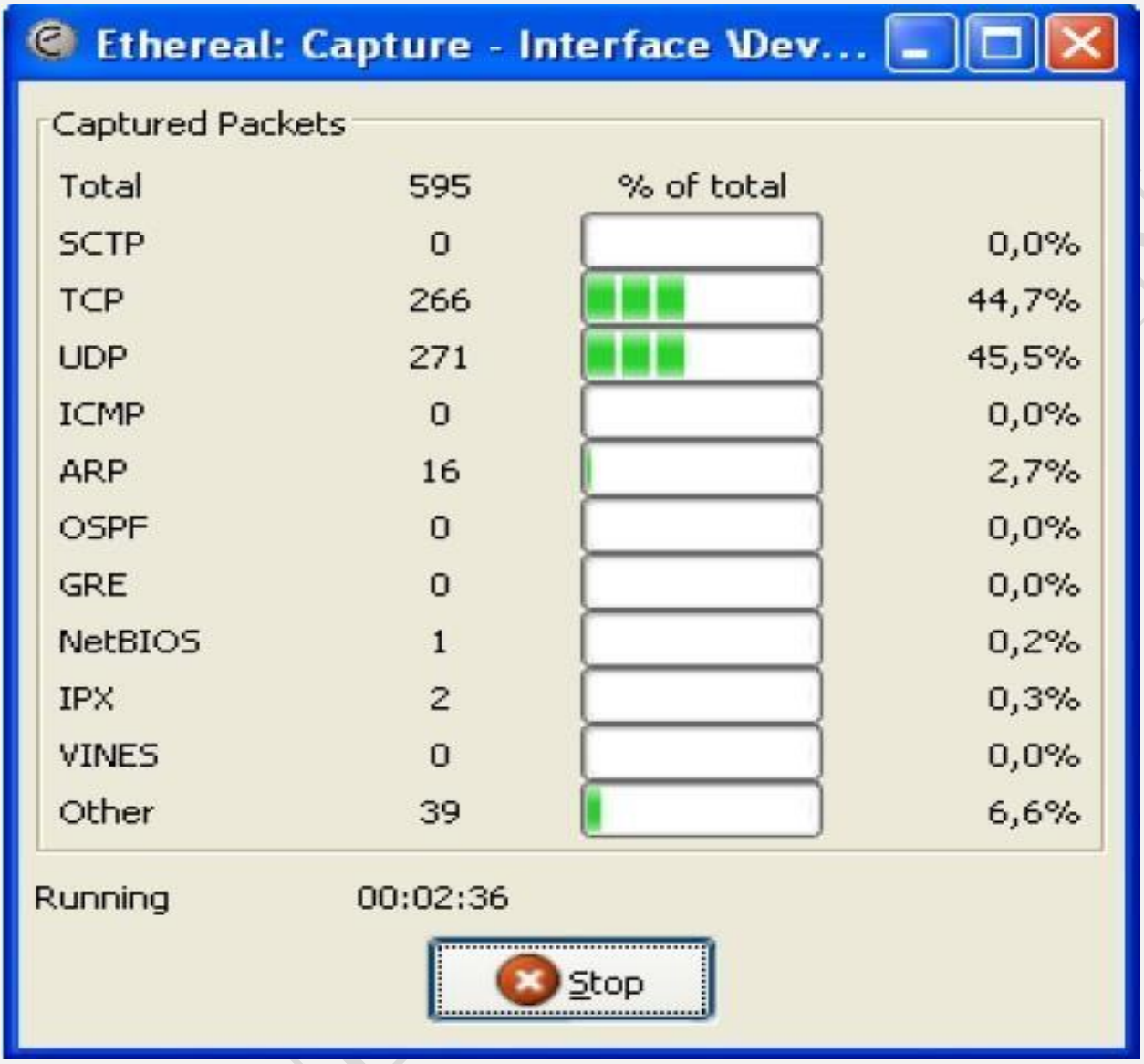
Capture packets in promiscuous mode: This checkbox allows you to specify that Ethereal should put the interface in promiscuous mode when capturing. If you do not specify this, Ethereal will only capture the packets going to or from your computer (not all packets on your LAN segment). **Limit each packet to n bytes** This field allows you to specify the maximum amount of data that will be captured for each packet, and is sometimes referred to as the **snaplen**. If disabled, the default is 65535, which will be sufficient for most protocols. Some rules of thumb: • If you are unsure, just keep the default value.
Capture Filter This field allows you to specify a capture filter. It defaults to empty, or no filter.

Filtering While Capturing

Packet capturing can be controlled by using different types of filters. Display filters are also very commonly used. Display filters allow you to concentrate on the packets you are interested in while hiding the currently uninteresting ones. They allow you to select packets by:

- Protocol
- The presence of a field
- The values of fields
- A comparison between fields

While a Capture is running



This dialog box will inform you about the number of captured packets and the time since the capture was started. The selection which protocols are counted cannot be changed. Simply stop by clicking the stop icon or specify the conditions and the capturing will stop automatically when

conditions are met. Conditions can be

After n packets: Stop capturing after the given number of packets have been captured.

After n megabytes: Stop capturing after the given number of bytes of have been captured.

After n minutes: Stop capturing after the given time has elapsed.

Working with Captured Packets

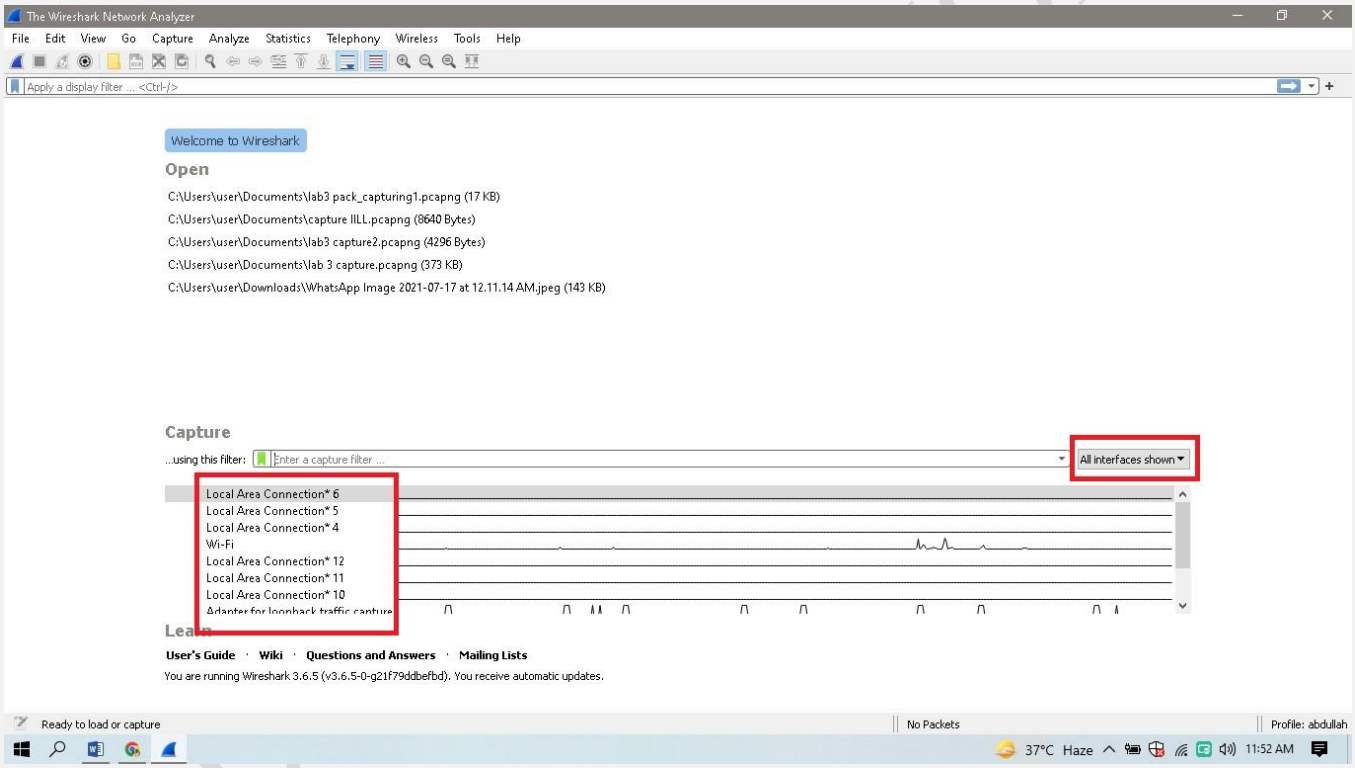
To analyze captured packets sound knowledge of TCP/IP suit is necessary. Packets are organized according to sequence number, protocol, time stamp etc.

LAB TASK

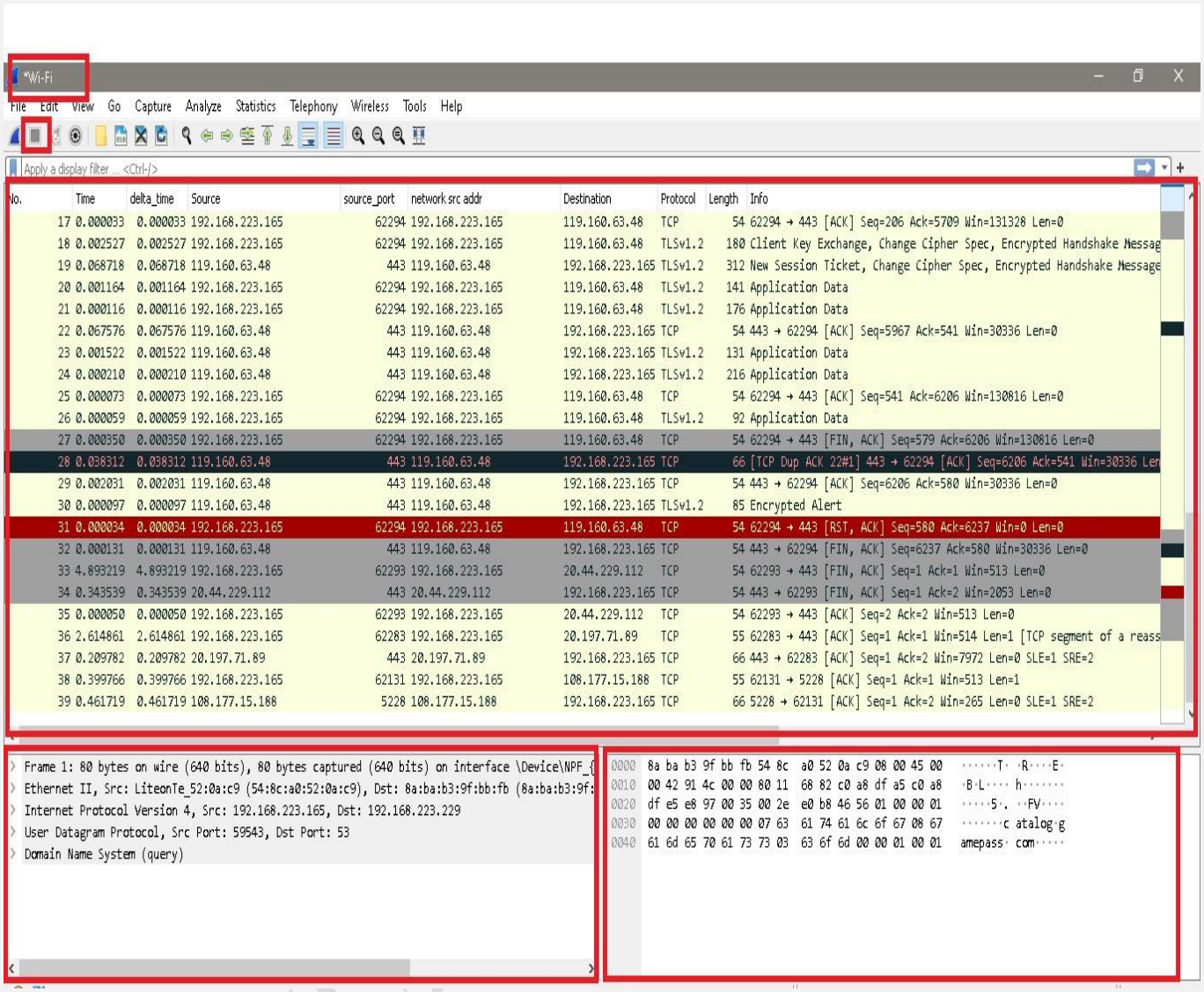
A. *Capture packets and analyze them using different kind of filters;*

i. *CAPTURING PACKETS:*

To capture packets from the wire and analyze them using different kind of filters, you can 1st open the **Wire Shark** software then select a specific **Interfaces** from the main **window of Wire Shark**. This will show all of the interfaces on the system. Here you can choose to capture packets from a sensor interface or another interface. To begin capturing packets from a particular interface, click on that **particular interface**.

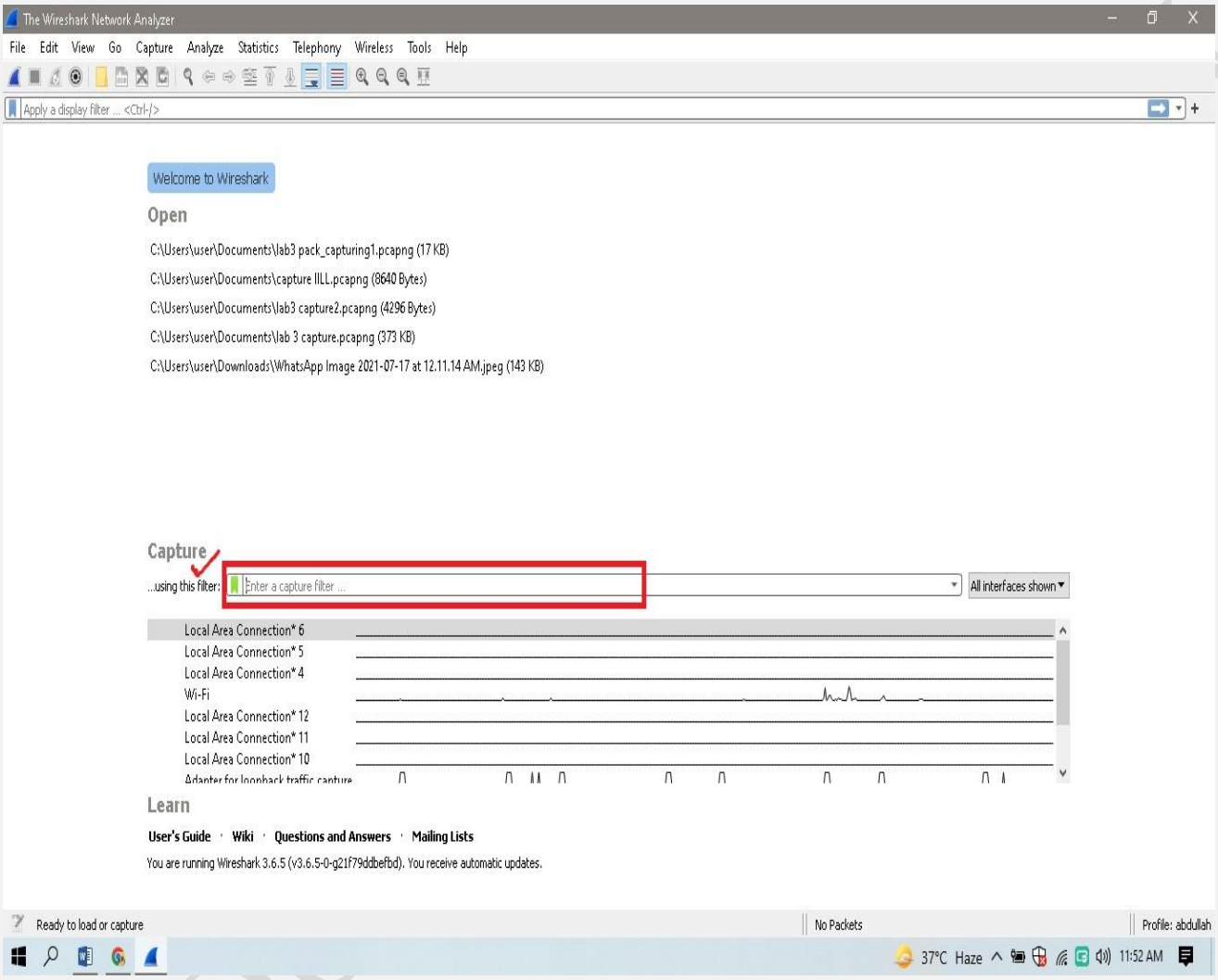


In the below fig, I had selected the wifi interface as shown below;

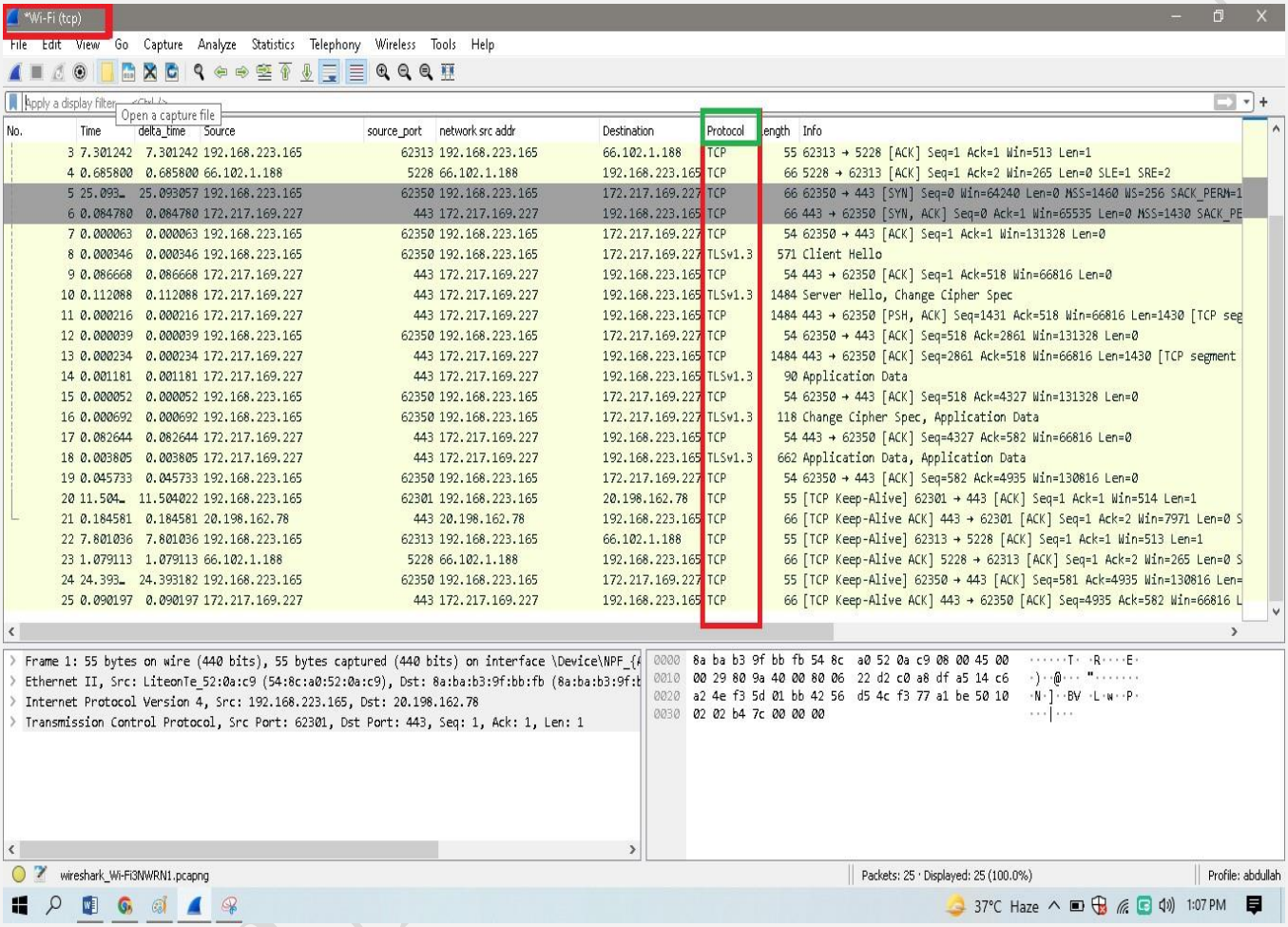


ii. CAPTURE FILTER:

In the main window, one can find the capture filter just above the interfaces list and in the interfaces dialog. The display filter can be changed above the packet list as can be seen in this picture.

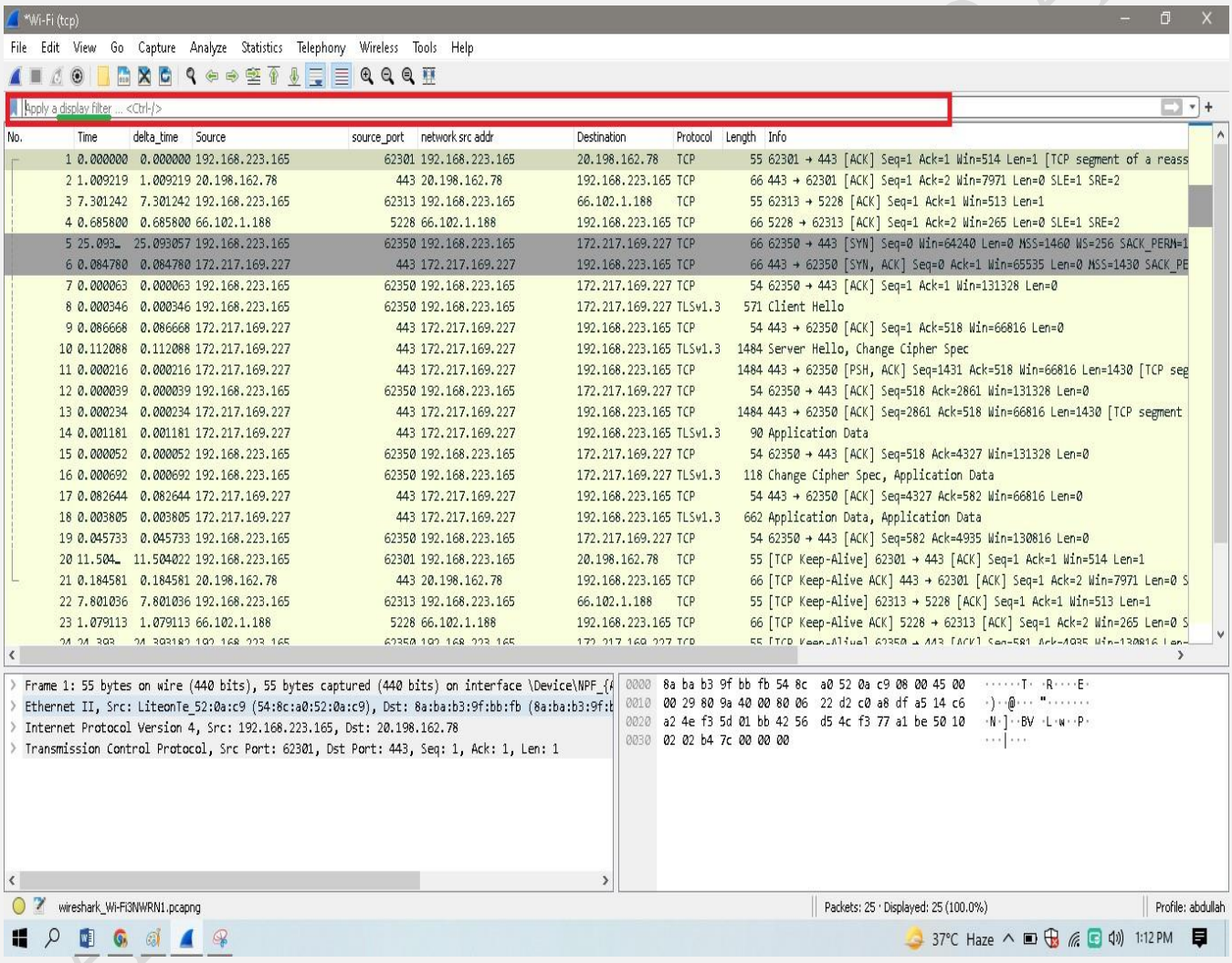


As in the below Fig I selected the capture filter as **TCP** , so the wireshark capturing only the TCP type protocol for capturizing as shown in fig;



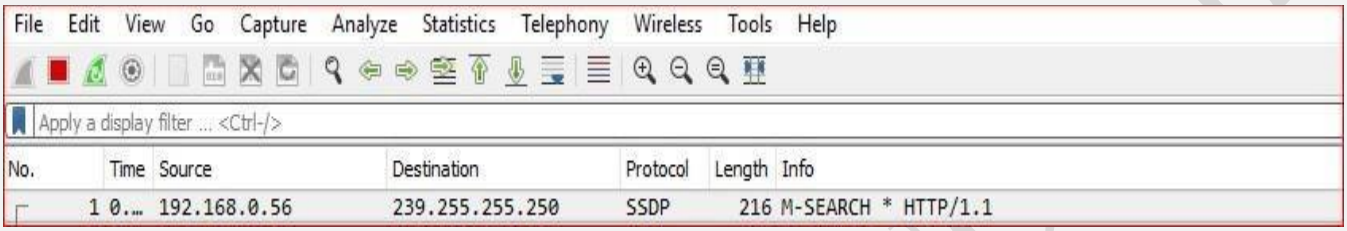
iii. DISPLAY FILTER :

Display filters are **used for filtering which packets are displayed** and are discussed below. Display filters allow you to concentrate on the packets you are interested in while hiding the currently uninteresting ones. They allow you to only display packets based on: Protocol. The presence of a field. **enables you to precisely control which packets are displayed**.



iv. DIFFERENCE BETWEEN CAPTURE FILTER AND DISPLAY FILTER:

Display Filters: This type of filter is used to reduce the packets which are showing in Wireshark. This type of filter can be changed while capturing traffic. It is generally used for hiding traffic to analyze the specific type of traffic.



Capture filters: This type of filter set before start capturing traffic in Wireshark. This type of filter can't change while capturing traffic. It is generally used for capturing a specific type of traffic.

LAB NO. 08

NAME *: ABDULLAH ZUNORAIN*

REG NO: *19JZELE0338*

SUBJECT*: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO: *SYED UZAIR GILLANI*

SECTION: *A*

TITLE: ANALYSIS OF HTTP USING WIRESHARK

Objective

- Explore the basic GET/ Response interaction,
- HTTP message format
- Retrieving HTML files with embedded objects
- HTTP authentication and security

Background

Hypertext Transfer Protocol (HTTP) is a communications protocol used to transfer or convey information on intranets and the World Wide Web. Its original purpose was to provide a way to publish and retrieve hypertext pages. HTTP is a request/response protocol between a client and a server. The client making an HTTP request - such as a web browser, or other end-user tool - is referred to as the *user agent*. The responding server - which stores or creates *resources* such as HTML files and images - is called the *origin server*.

6.1 The Basic HTTP GET/response interaction

Let's begin our exploration of HTTP by downloading a very simple HTML file - one that is very short, and contains no embedded objects. Do the following:

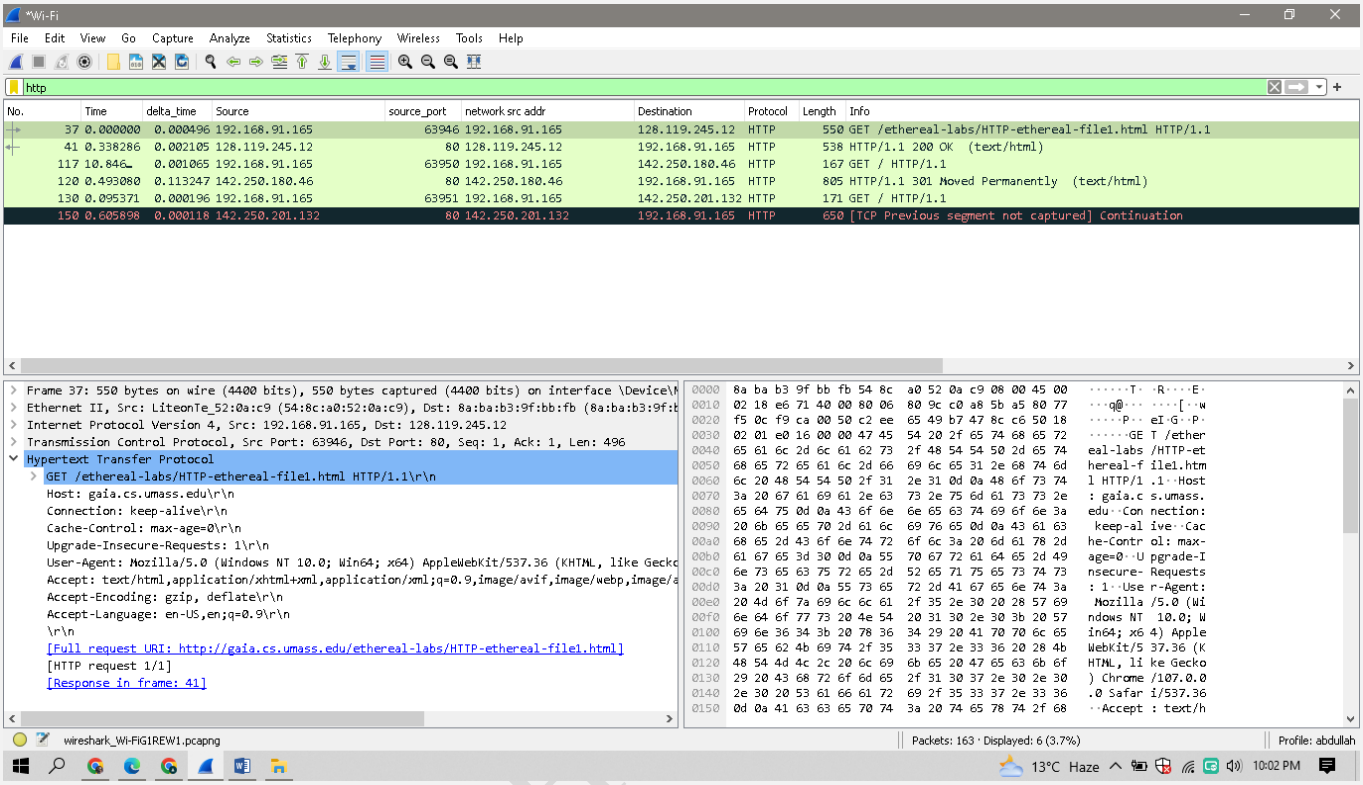
1. Start up your web browser.
2. Start up the ethereal packet sniffer, as described in the introductory lab (but don't yet begin packet capture). Enter "http" (just the letters, not the quotation marks) in the display-filterspecification window, so that only captured HTTP messages will be displayed later in the packet-listing window. (We're only interested in the HTTP protocol here, and don't want to see the clutter of all captured packets).
3. Wait a bit more than one minute (we'll see why shortly), and then begin Ethereal packet capture.
4. Enter the following to your browser

http://gaia.cs.umass.edu/ethereal-labs/HTTP-ethereal-file1.html

Your browser should display the very simple, one-line HTML file.

5. Stop Ethereal packet capture.

Your Ethereal window should look similar to the window shown in Figure 1.



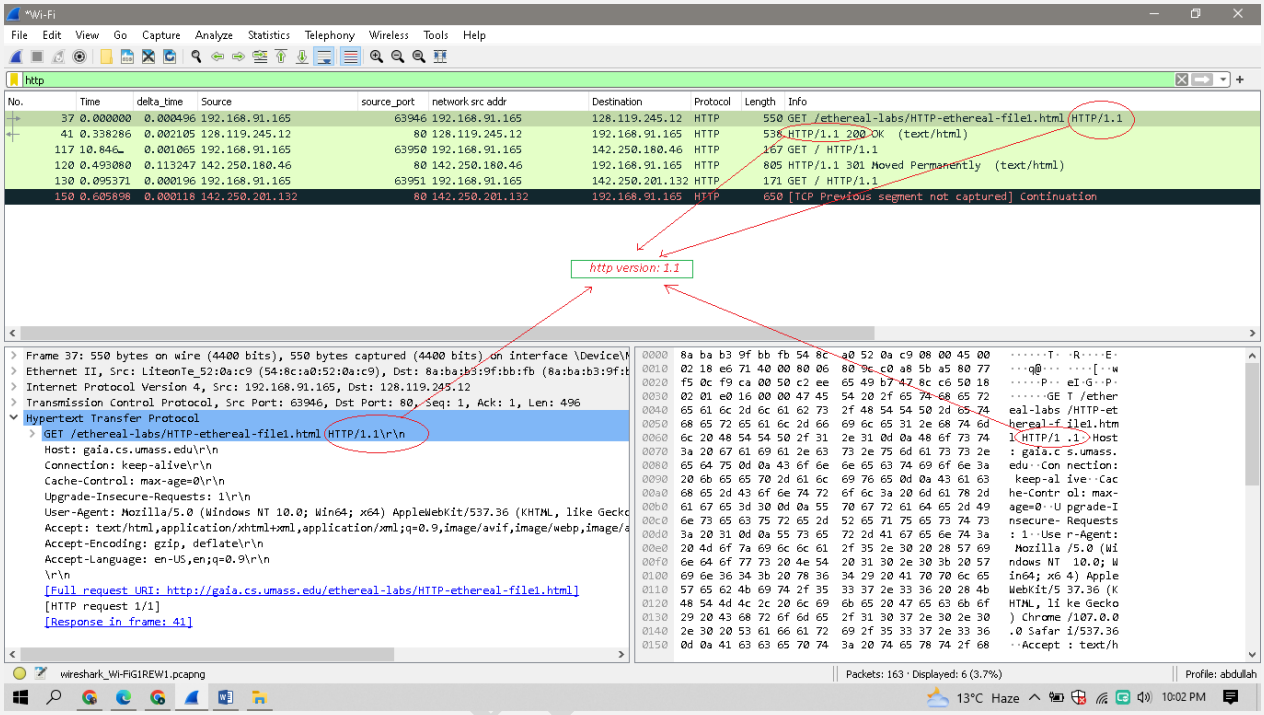
The example in Figure 1 shows in the packet-listing window that six HTTP messages were captured: the GET messages (from your browser to the gaia.cs.umass.edu web server) and the response messages from the server to your browser. The packet-contents window shows details of the selected message (in this case the HTTP GET message, which is highlighted in the packet-listing window). By looking at the information in the **HTTP GET and response messages**, answer the following questions. When answering the following questions, you should print out the

GET and response messages (see the introductory Ethereal lab for an explanation of how to do this) and indicate where in the message you've found the information that answers the following questions.

Questions

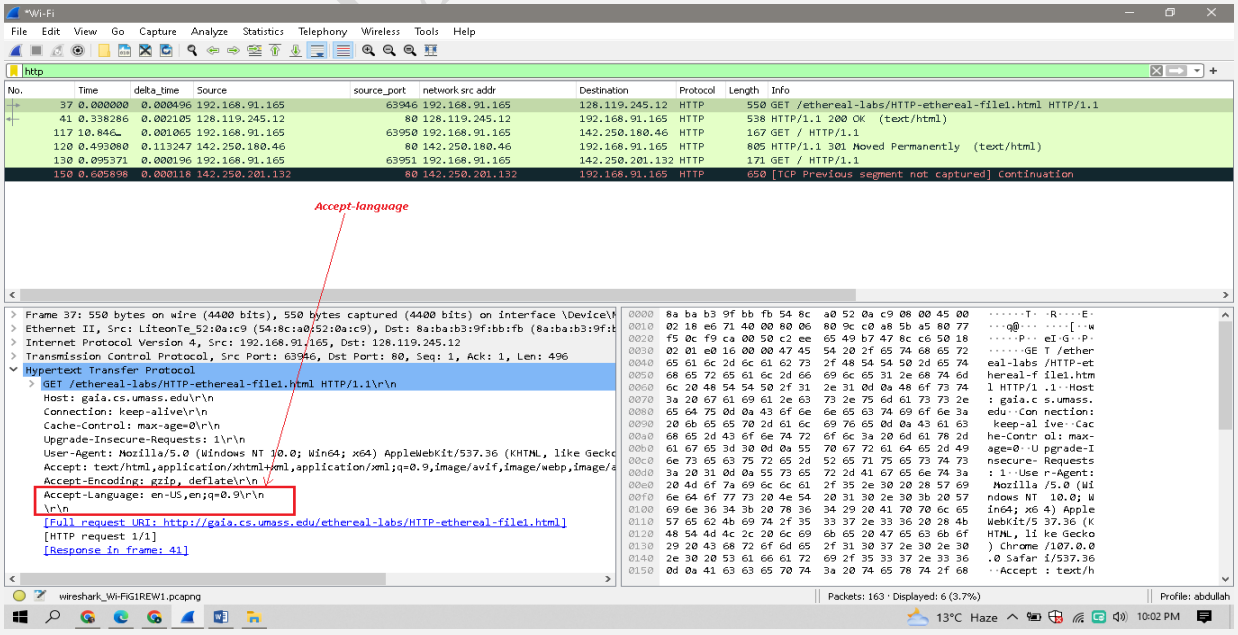
1. **Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?**

Ans: Browser running the HTTP VERSION 1.1 & HTTP SERVER VERSION 1.1



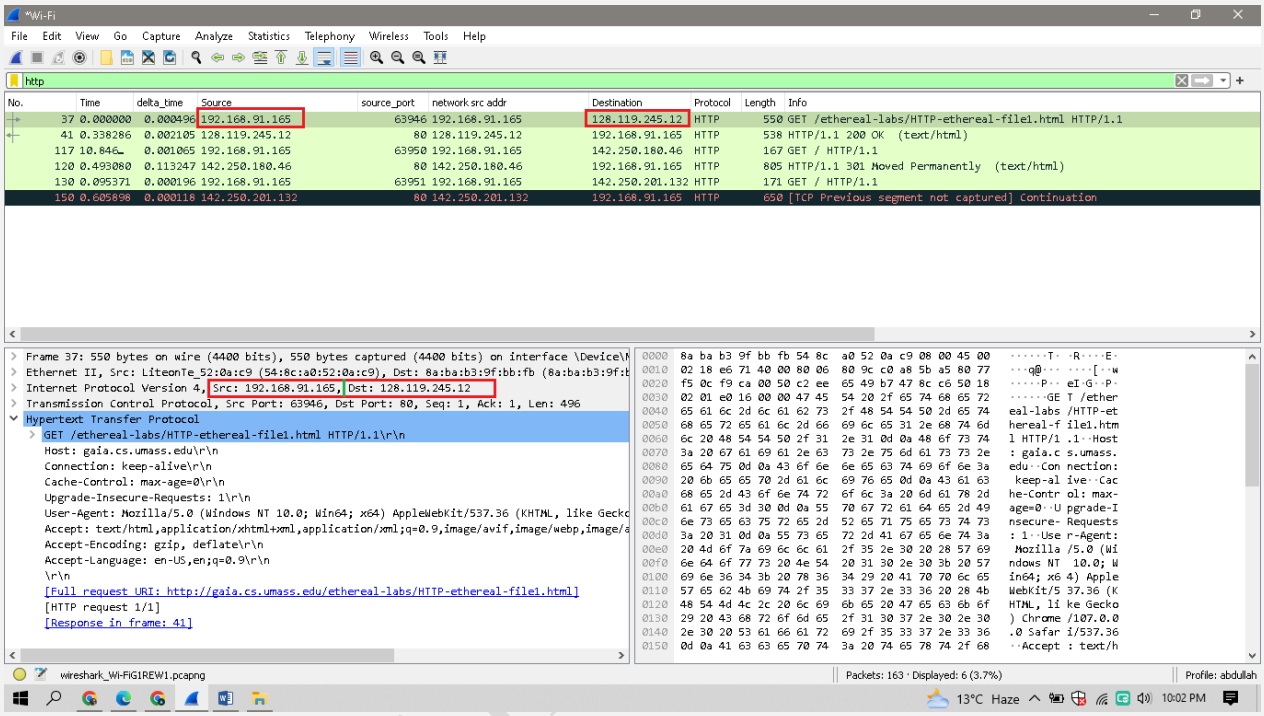
2. **What languages (if any) does your browser indicate that it can accept to the server?**

Ans: Accepted-language = en-US



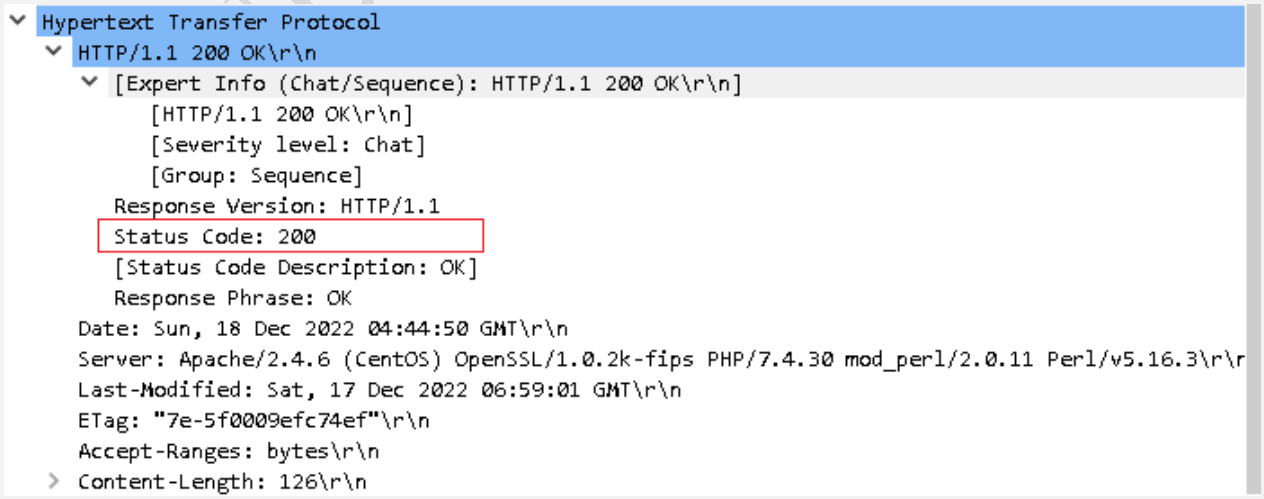
3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

Ans: My computer ip-address is **192.168.91.165** (this is a virtual IP address assigned by my virtual machine software) and the destination(gaia.cs.umass.edu server) ip-address is **128.119.245.12**,see the below screenshot;



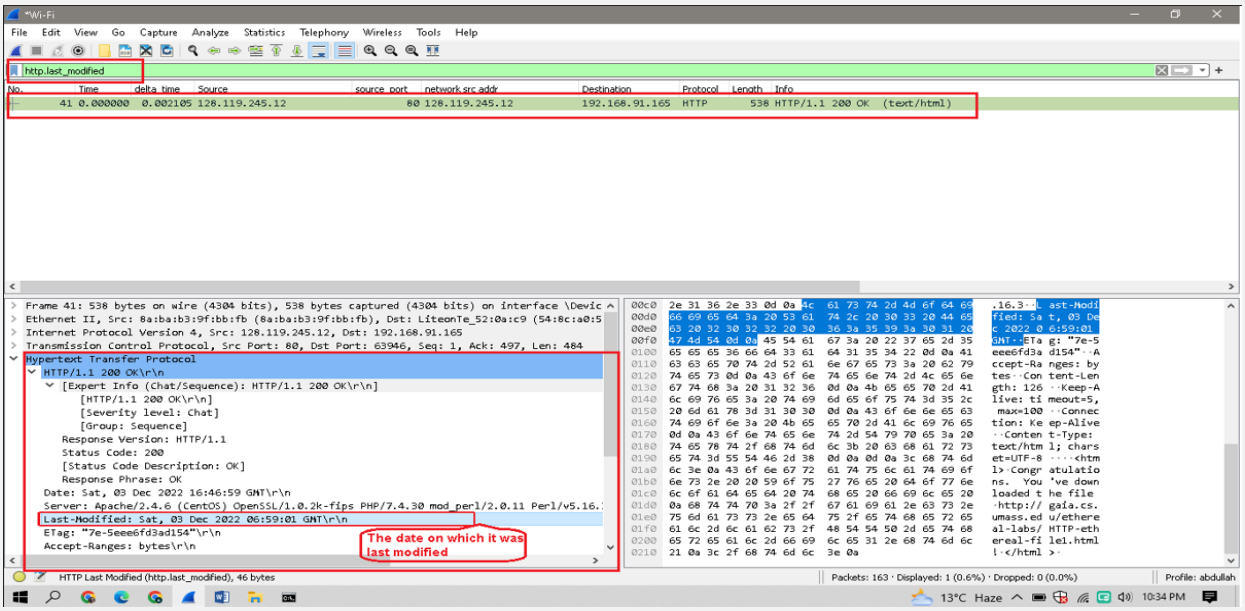
4. What is the status code returned from the server to your browser?

Ans: Status code: 200



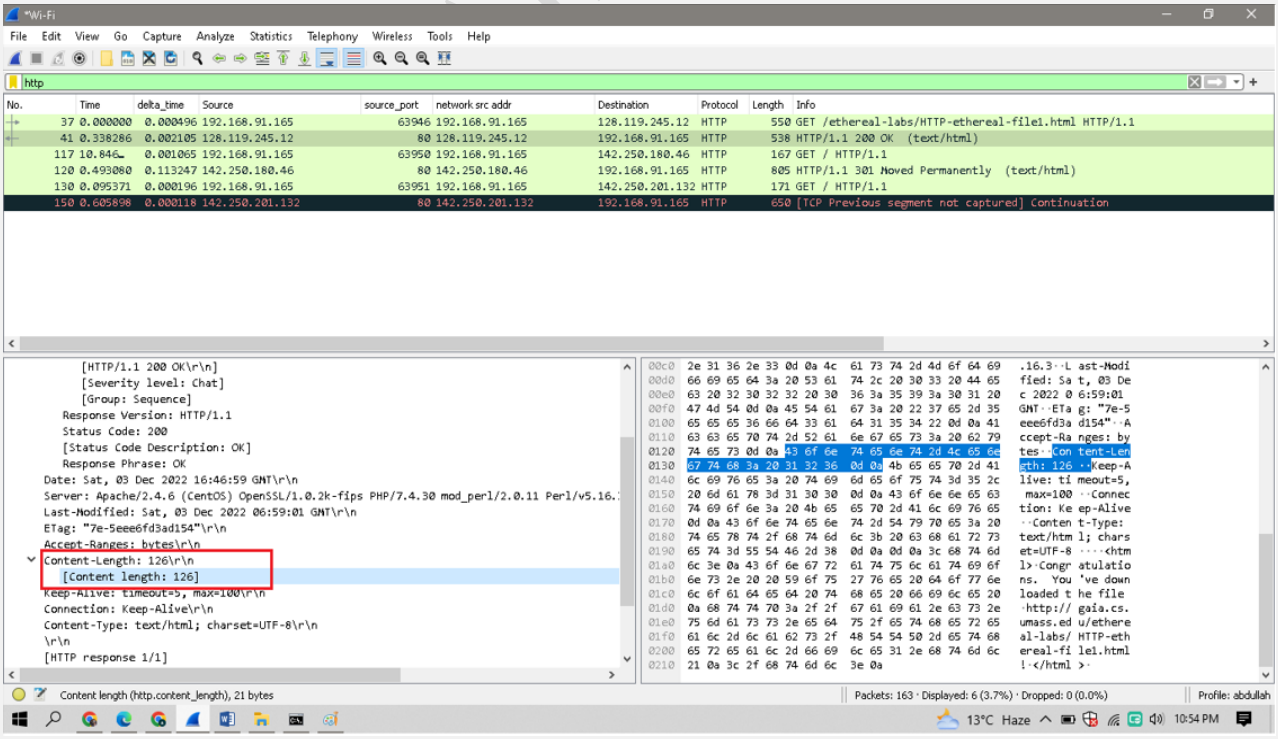
5. When the HTML file that you are retrieving was last modified at the server?

Ans: We can filter messages by http.last_modified and we see that the HTTP response I received for the html 1 file doesn't show this field. We do have a http.last_modified field in the favicon response however, as shown in the screenshot below. This says the favicon was last modified on **3 dec 2022.**



6. How many bytes of content are being returned to your browser?

Ans: Content-length: **126 bytes** , see in the screen shot below;



7. **By inspecting the raw data in the packet content window, do you see any headers within the data that are not displayed in the packet-listing window? If so, name one.**

Ans: No. The raw data appears to match up exactly with what is shown in the packet-listing window, so all of the headers can be found in the raw data.

6.2. The HTTP CONDITIONAL GET/response interaction

Before performing the steps below, make sure your browser's cache is empty. Now do the following:

• Start up your web browser, and make sure your browser's cache is cleared, as discussed above.

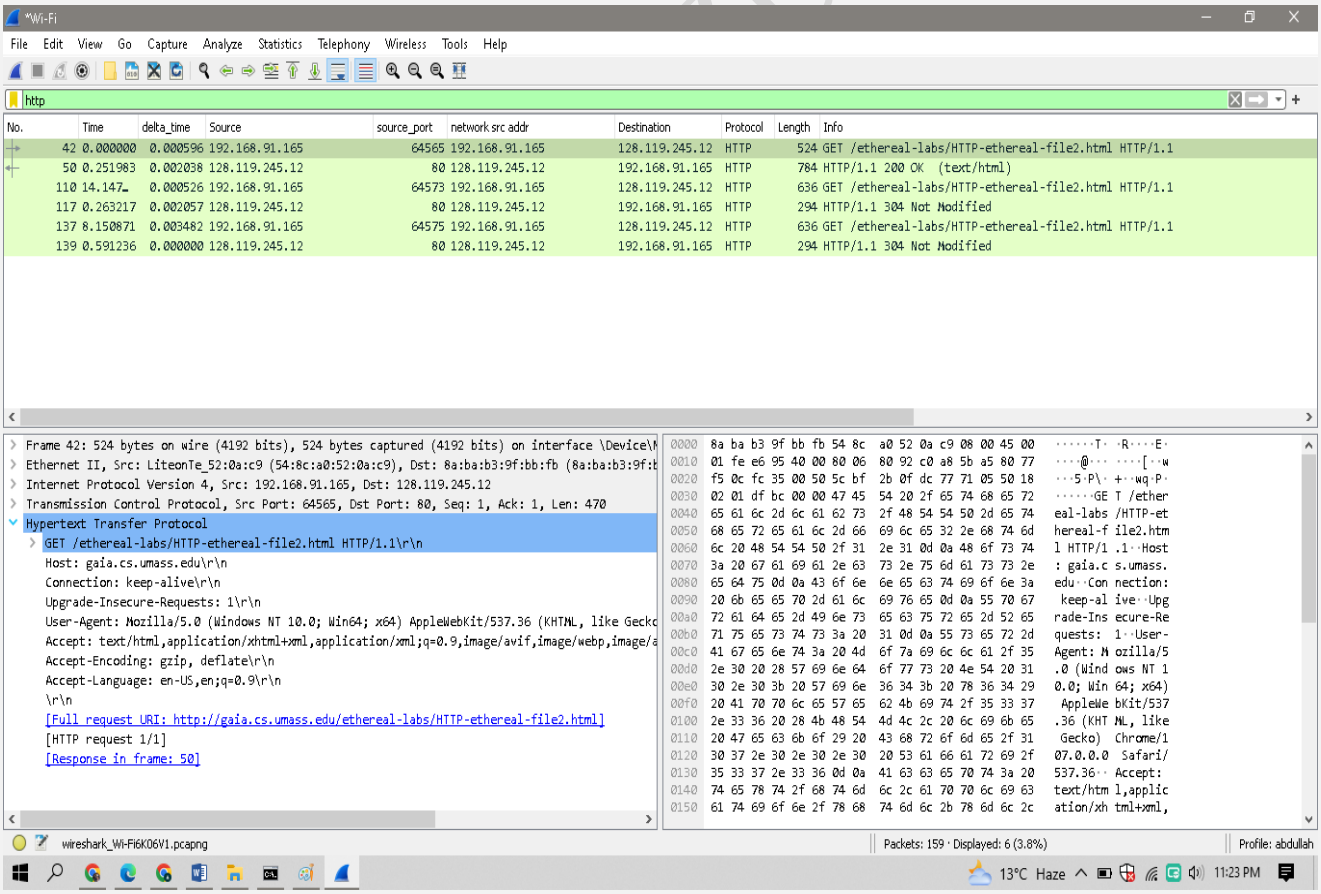
• Start up the ethereal packet snifter

• Enter the following URL into your browser

http://gaia.cs.umass.edu/ethereal-labs/HTTP-ethereal-file2.html Your browser should display a very simple five-line HTML file.

• Quickly enter the same URL into your browser again (or simply select the refresh button on your browser)

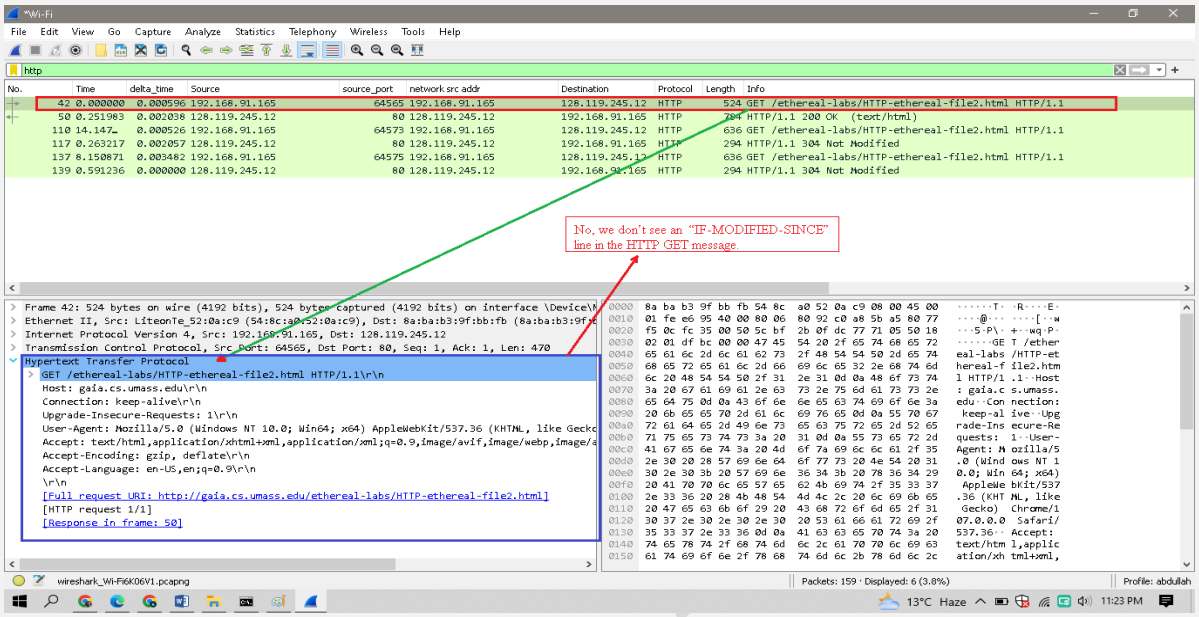
• Stop ethereal packet capture, and enter "http" in the display-filter-specification window, so that only captured HTTP messages will be displayed later in the packet-listing window.



Answer the following questions:

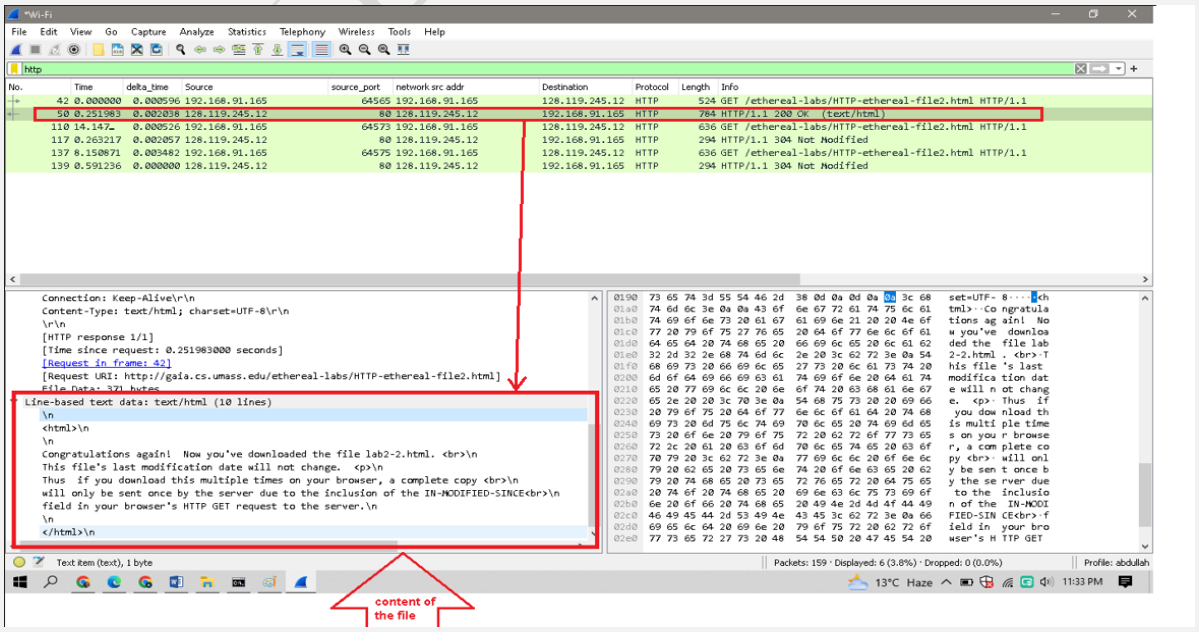
8. **Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE" line in the HTTP GET?**

Ans: No, we don't see an "IF-MODIFIED-SINCE" line in the HTTP GET message.



9. **Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?**

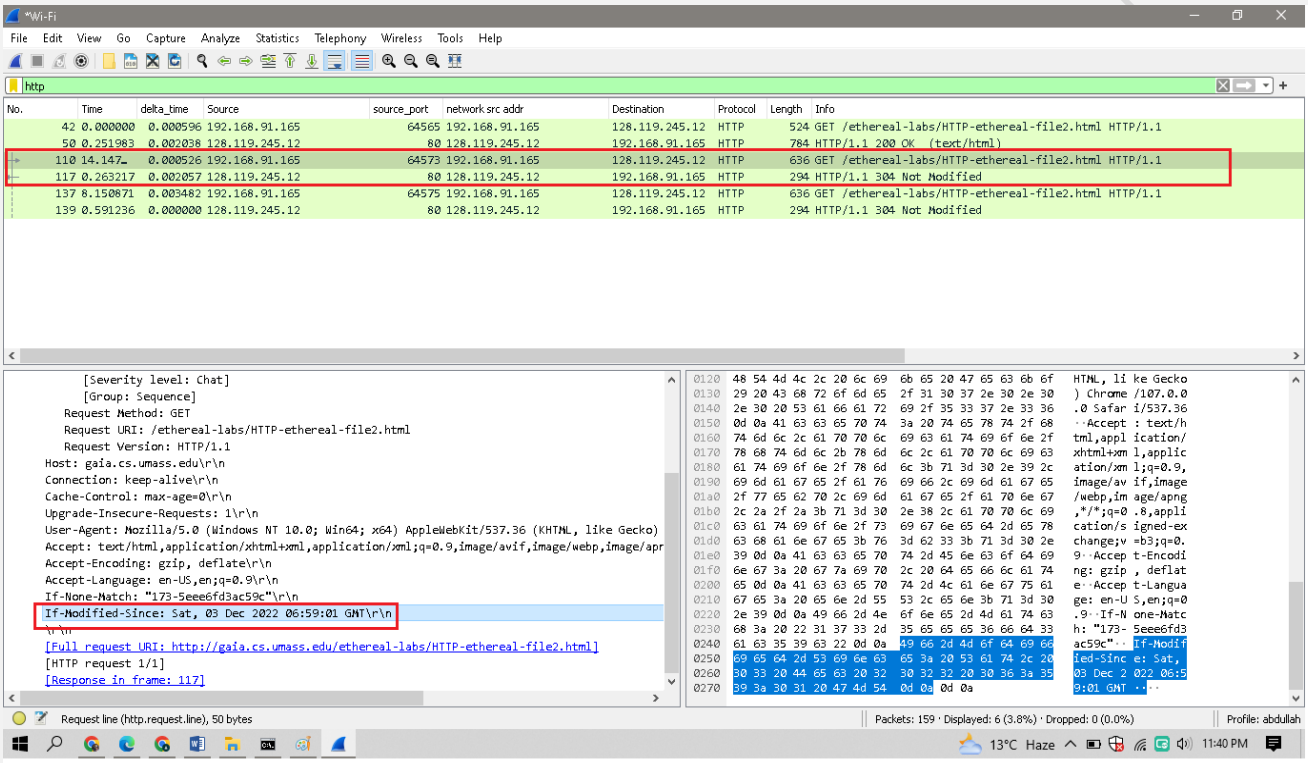
Ans: See the below screenshot of the server response with ***line-based text Data:text/html(10 lines)***



10. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE:" line in the HTTP GET? If so, what information follows the "IF-MODIFIED-SINCE:" header?

Ans: Yes,

If-Modified-Since: Sat, 03 Dec 2022 06:59:01 GMT\r\n ,As screenshot is below;



11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET?

Did the server explicitly return the contents of the file? Explain.

Ans: Status code: 304 & Response phrase: Not Modified,

The server will only send the content of the file once and will not send again the content of the file before the TTL expiry. And at that time the browser cache will responds.

LAB TASK

Do the following:

• Start up your web browser, and make sure your browser's cache is cleared, as discussed above.

• Start up the ethereal packet snifter

• Enter the following URL into your browser

http://gaia.cs.umass.edu/ethereal-labs/HTTP-ethereal-file4.html **Answer the following questions:**

12. **How many HTTP GET request messages were sent by your browser? To which Internet addresses were these GET requests sent?**

Ans: Six HTTP GET request were sended by our browser.

These GET req's were sended from my browser(source having IP-address: 192.168.91.165) To 128.119.245.12 , 52.51.131.59 , 142.250.201.142 , 142.250.201.132 , 142.250.180.46

13. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.

Ans: Based on the timestamps, it appears the images were downloaded serially. Also, the source port is incrementing each time from 49509, 49511, 49512, 49514 which means that the images were received serially over separate TCP connections.

LAB NO.09

NAME *: ABDULLAH ZUNORAIN*

REG NO: *19JZELE0338*

SUBJECT*: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO: *SYED UZAIR GILLANI*

SECTION: *A*

TITLE: ANALYSIS OF DNS PROTOCOL USING WIRESHARK

OBJECTIVES:

o Set Wireshark for capture packets for DNS.
o To Trace DNS with wireshark.
o To use Nslookup command.

THEORY:

DNS PROTOCOL:

DNS is the **system used to resolve store information about domain names** including IP addresses, mail servers, and other information.

Domain Name System (DNS) resolves hostnames to IP addresses. In this lab, we'll take a closer look at the client side of DNS. The client's role in the DNS is relatively simple, a client sends a query to its local DNS server, and receives a response back. The hierarchical DNS servers communicate with each other to either recursively or iteratively resolve the client's DNS query.

PART.1 NS LOOKUP COMMAND

To run **nslookup** command in Windows, open the Command Prompt and run **nslookup** on the command line. **nslookup** tool allows the host to query any specified DNS server for a DNS record. The queried DNS server can be a root DNS server, a top-level-domain DNS server, an authoritative DNS server, or an intermediate DNS server. To accomplish this task, **nslookup** sends a DNS query to the specified DNS server, receives a DNS reply from that same DNS server, and displays the result.

>>nslookup uetpeshawar.edu.pk

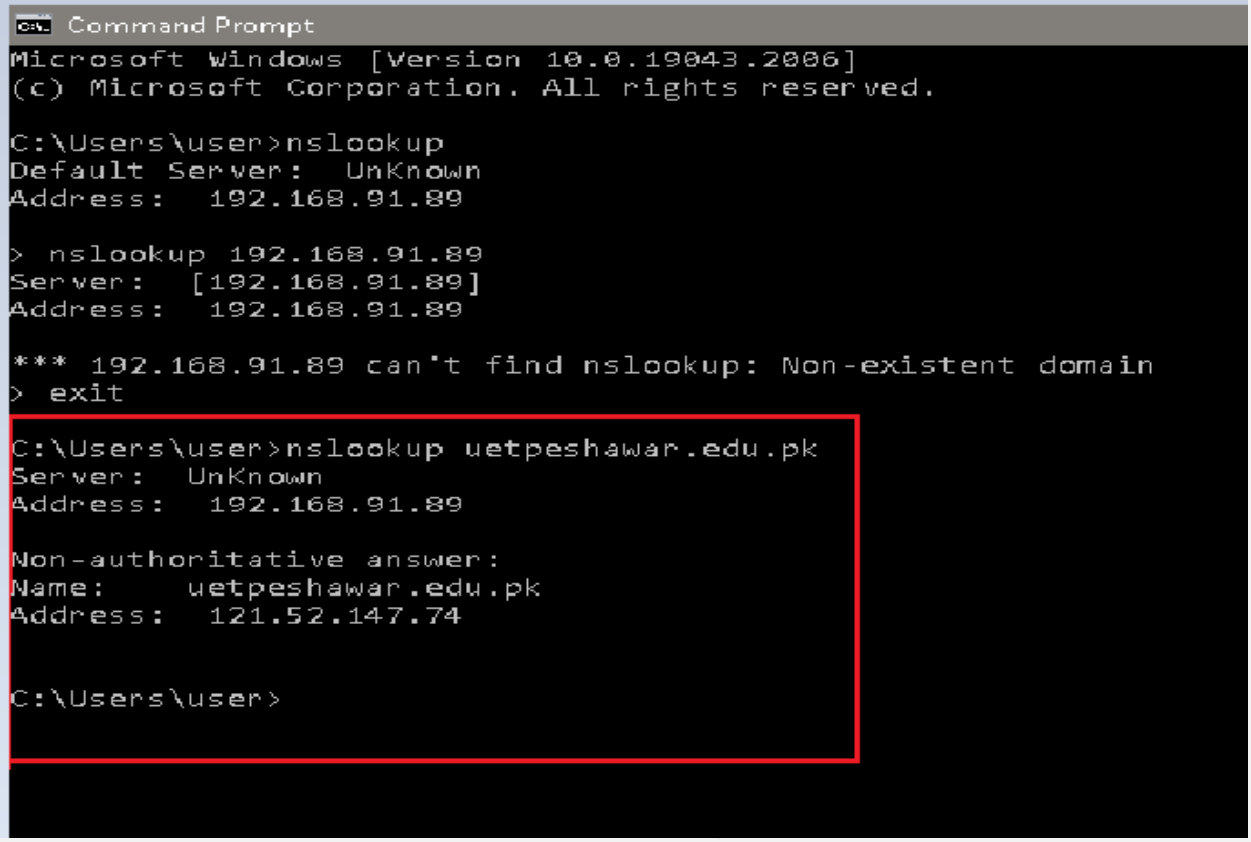


FIG 1- NSLOOKUP FOR WWW.UETPESHAWAR.EDU.PK

In fig-1, the command nslookup **www.uetpeshawar.edu.pk** is used to get IP address for the **www.uetpeshawar.edu.pk.**

There are two replies to this command, **1st** is the name and IP address of **unknown(router or anything to which the 1st DNS req is reached)**, and **2nd** is the host name and IP address of **www.uetpeshawar.edu.pk.**

>>nslookup -type=NS uetpeshawar.edu.pk

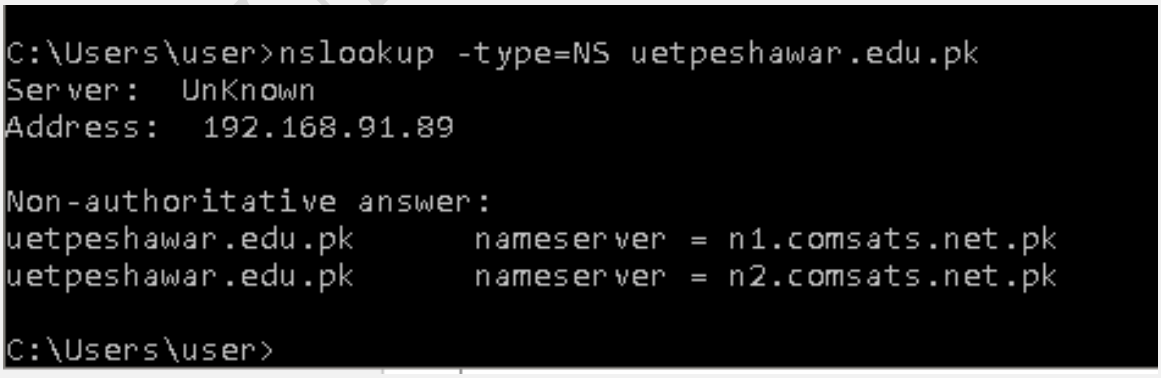


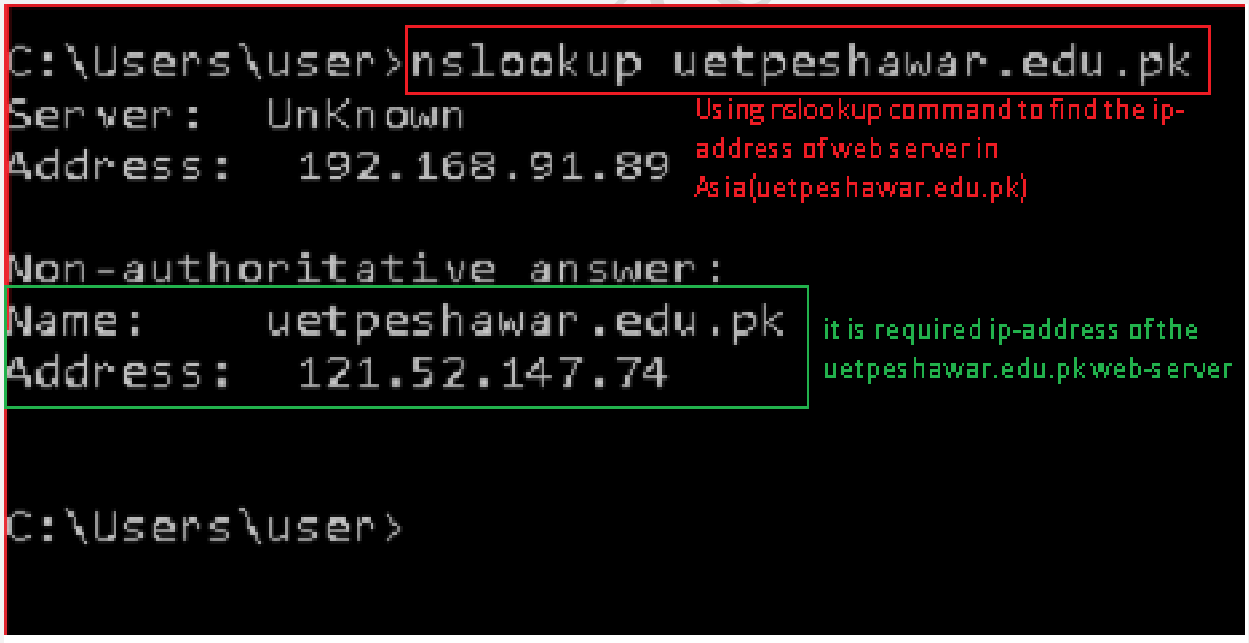
FIG-2 -NSLOOKUP FOR NAME SERVERS OF UETPESHAWAR.EDU.PK

In fig-2, the command **nslookup** uetpeshawar.edu.pk is used to get the hostnames of **authoritative DNS server** for uetpeshawar.edu.pk.

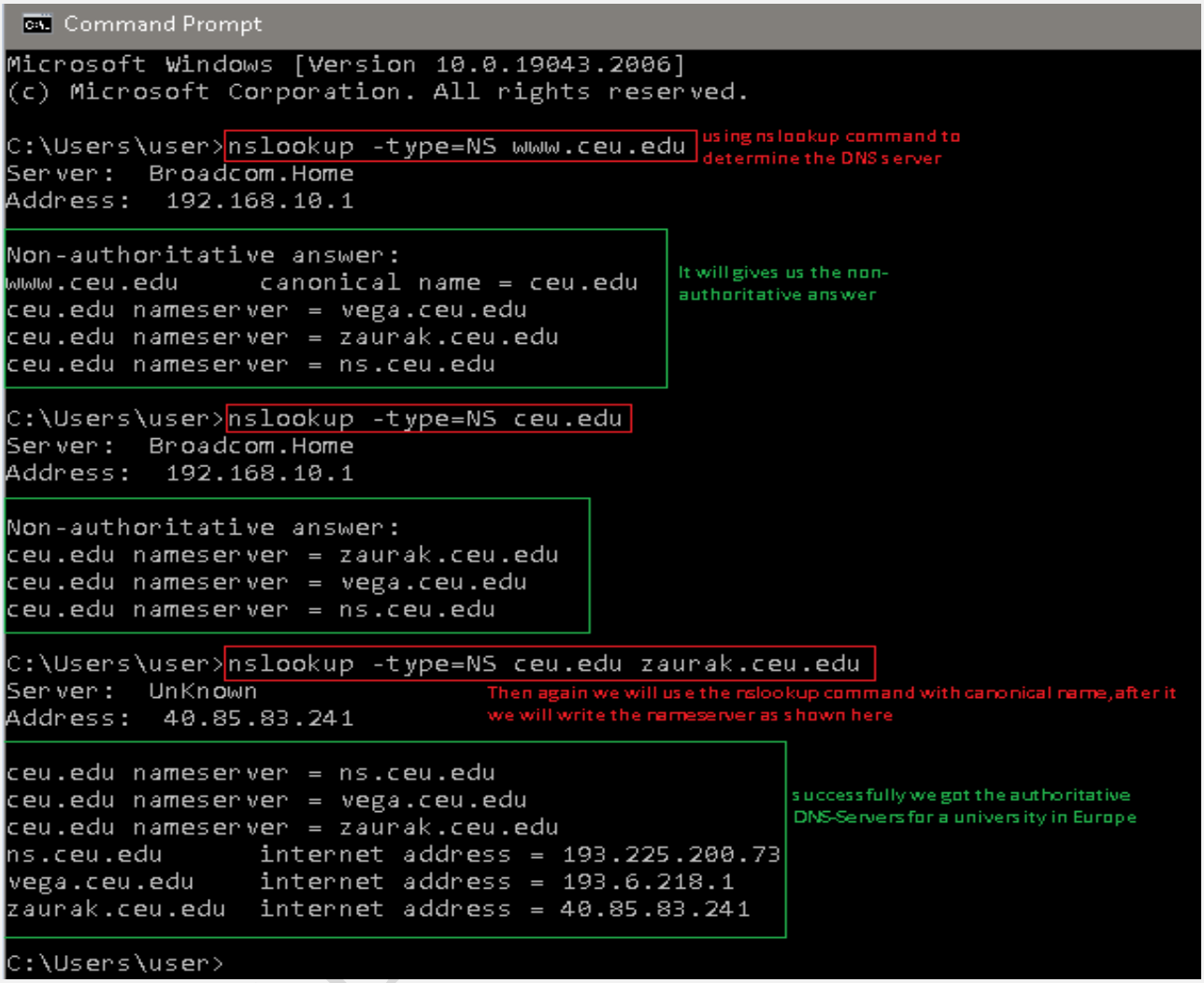
So, it gives non-authoritative answer with two name servers which means that this reply was came from cache of some servers.

- **Now that we have provided an overview of nslookup, it is time for you to test drive it yourself. Do the following (and write down the results):**

1. Run nslookup to obtain the IP address of a Web server in Asia. What is the IP address of that server?
   ANS: Name: uetpeshawar.edu.pk
   Address:121.52.147.74



2. Run nslookup to determine the authoritative DNS servers for a university in Europe.



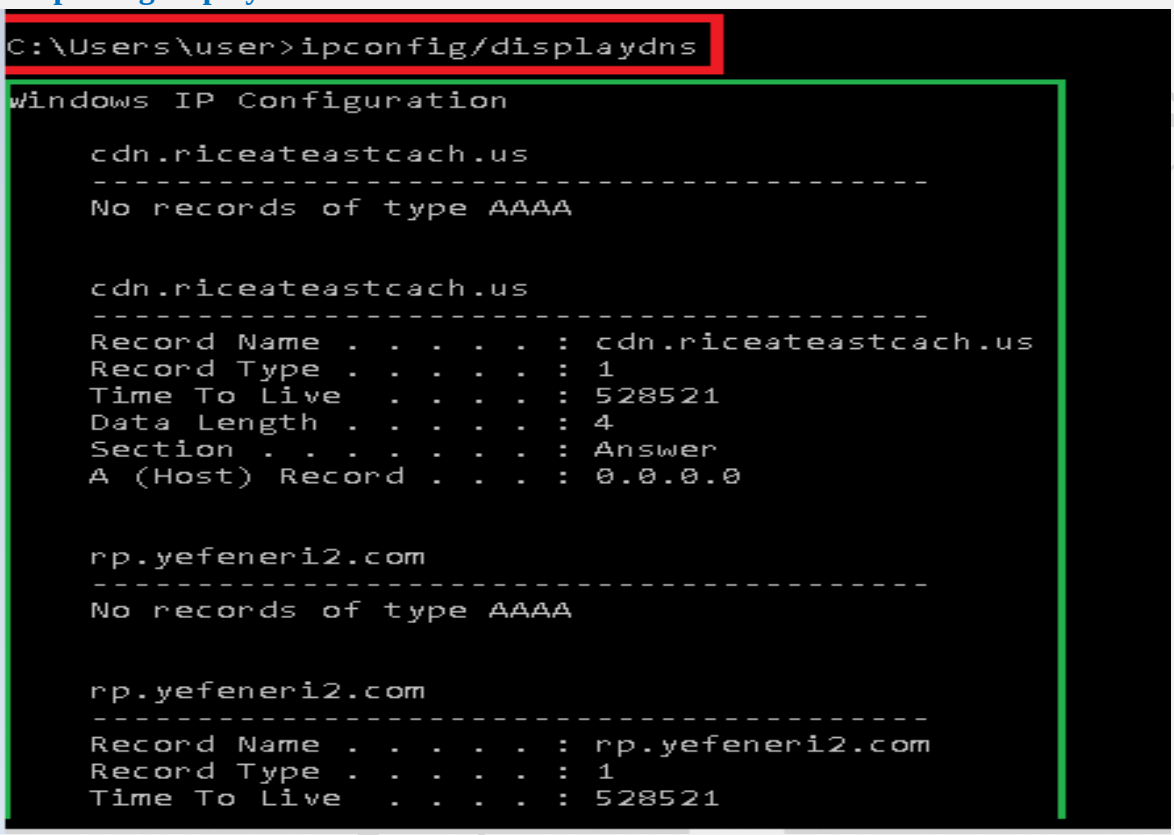
3. Run nslookup so that one of the DNS servers obtained in Question 2 is queried for the mail servers for Yahoo! mail. What is its IP address?

ANS: I was unable to get any of the DNS servers listed above to answer a query for a Yahoo mail server (even cn.mail.yahoo.com was refused).

PART.2 IPCONFIG COMMAND

Ipconfig command can also be used to show information about DNS which is stored in a host. As we know that hosts can cache the DNS records we are going to see this using ipconfig/displaydns command.

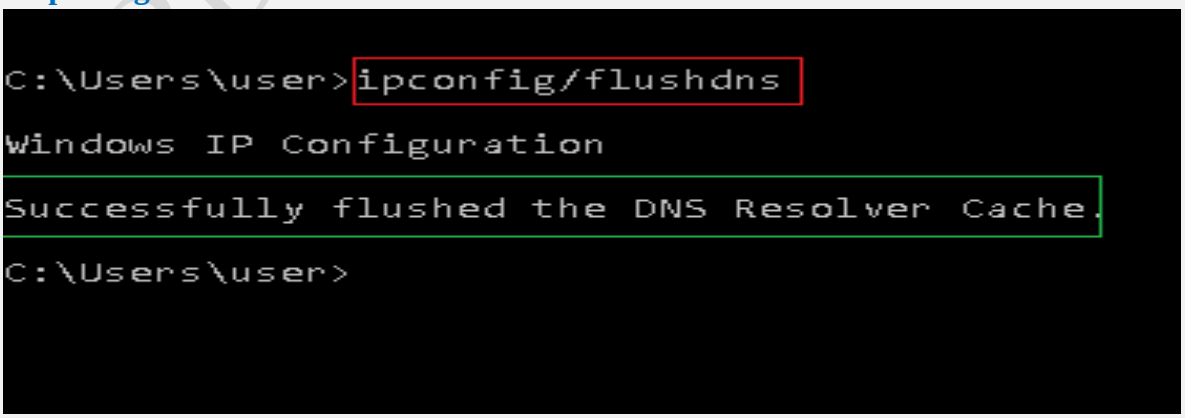
>> ipconfig/displaydns



The above screenshot shows the cached DNS servers, providing DNS record name, type, time to live, data length and canonical name for that cached DNS record.

These cached DNS records can be cleared using below command;

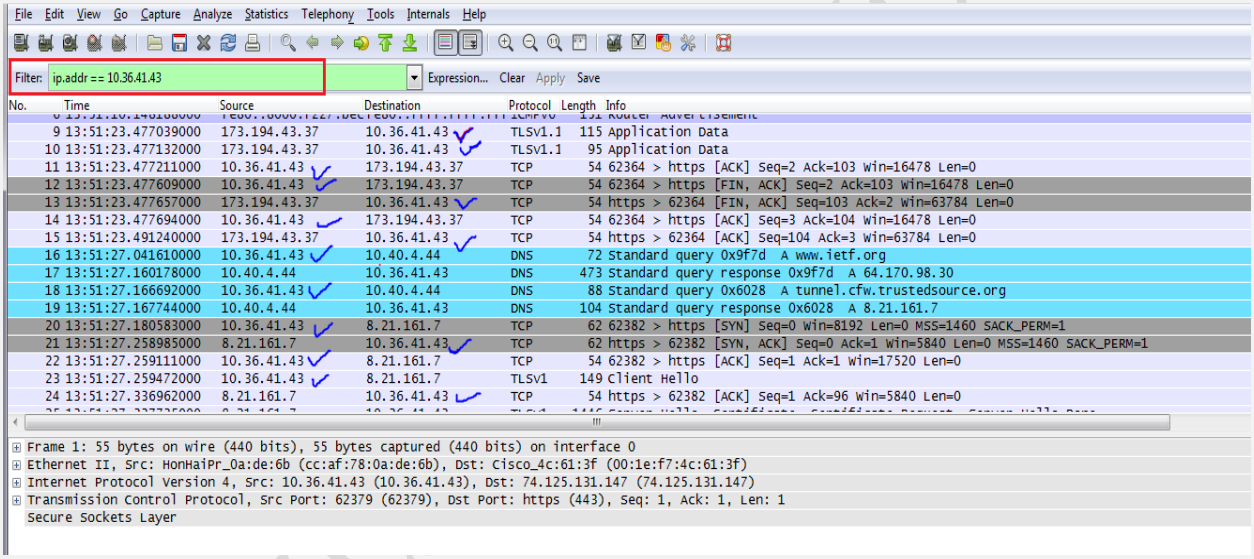
>>ipconfig/flushdns



PART.3 TRACING DNS TRAFFIC WITH "WIRESHARK"

To trace DNS using wireshark, follow the following steps;

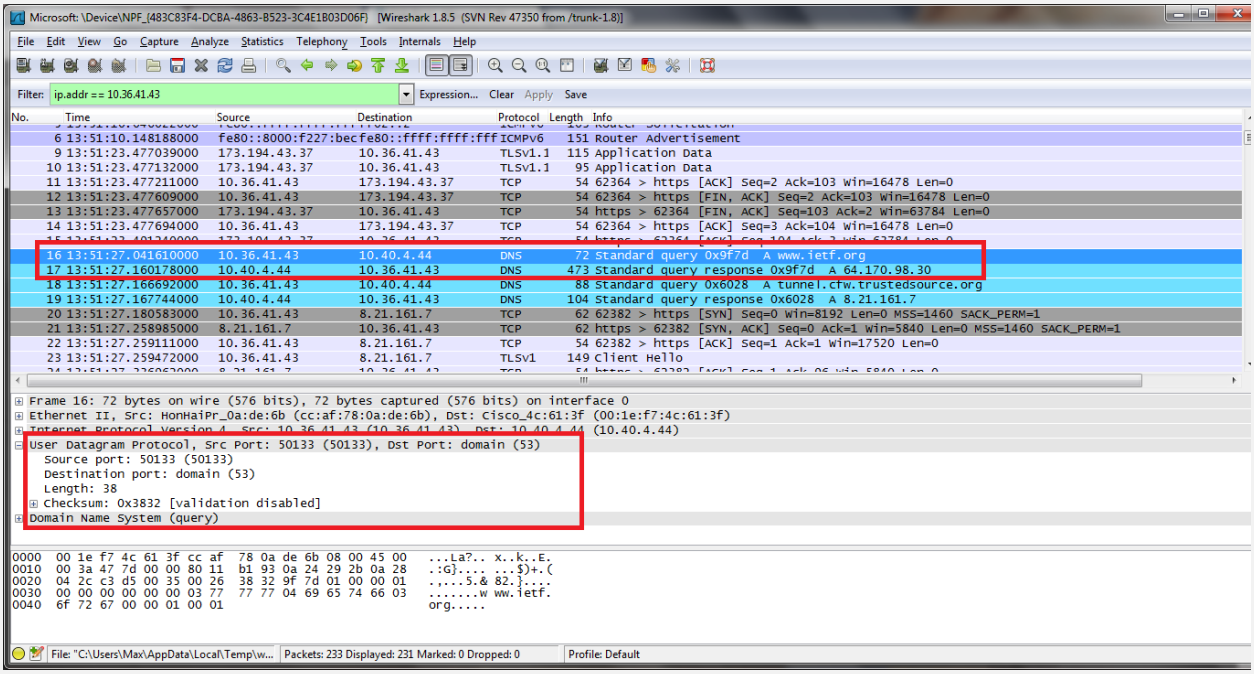
- Use ipconfig to empty the DNS cache in your host.
- Open your browser and empty your browser cache. (With Internet Explorer, go to Tools menu and select Internet Options; then in the General tab select Delete Files.)
- Open Wireshark and enter **"ip.addr == your_IP_address"** into the filter, where you obtain your_IP_address with ipconfig. This filter removes all packets that neither originate nor are destined to your host.
- Start packet capture in Wireshark.



QUESTIONS:

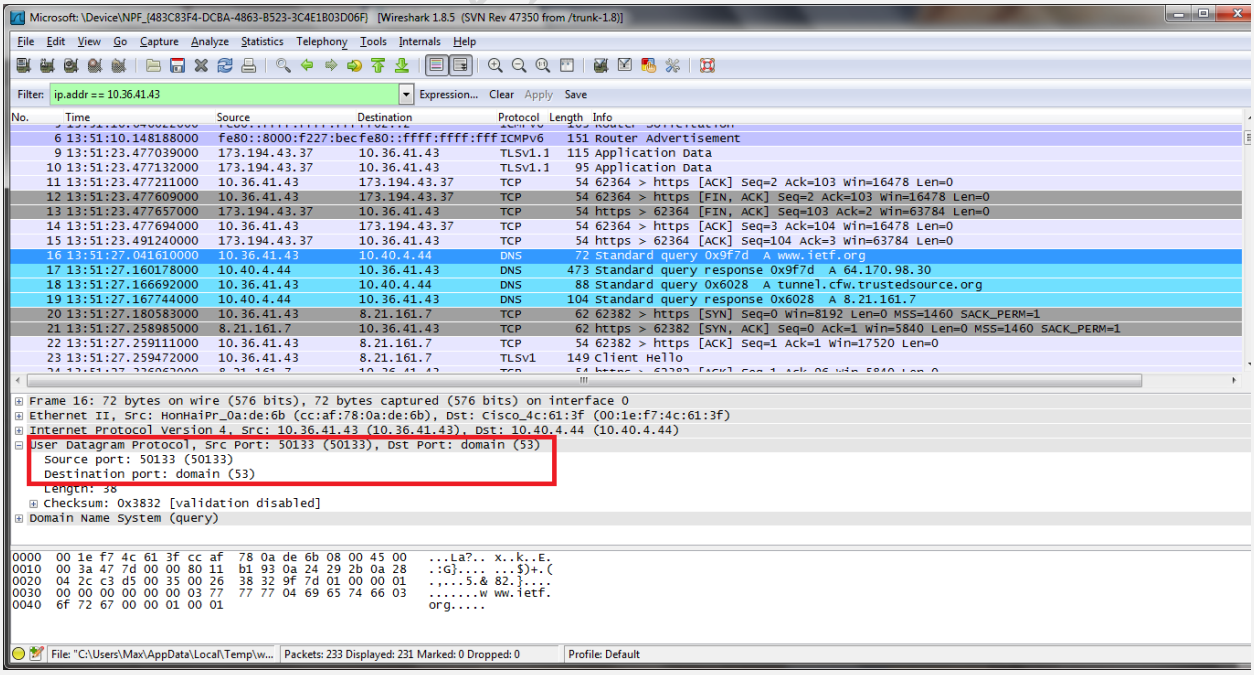
1) Locate the DNS query and response messages. Are then sent over UDP or TCP?

Ans: The DNS query and response messages are sent over UDP.



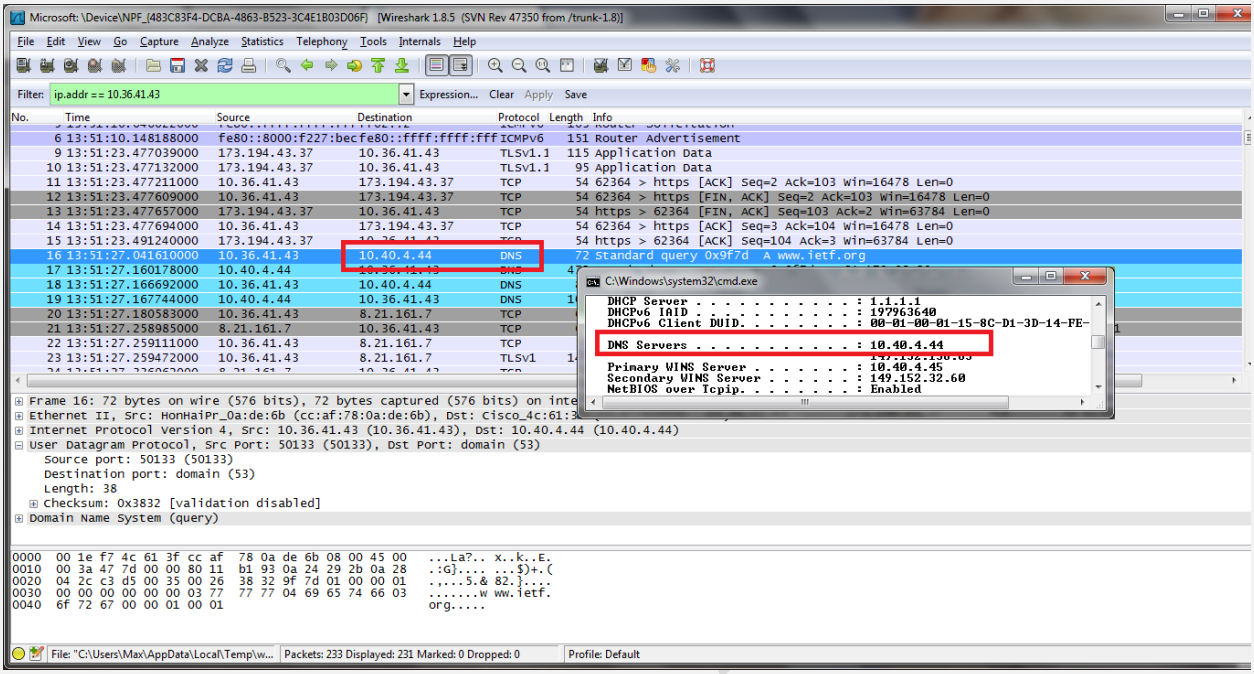
2) What is the destination port for the DNS query message? What is the source port of DNS response message?

Ans: The destination port is 53 & the source port is 50133.



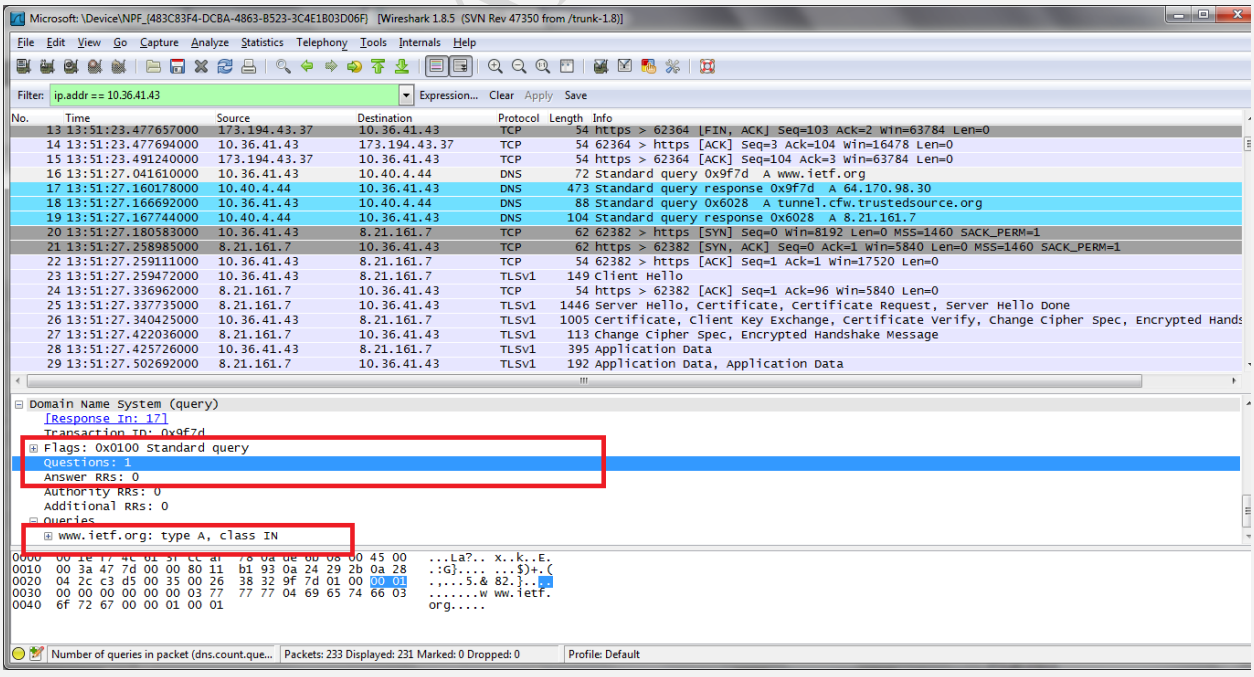
3) To what IP address is the DNS query message sent? Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same?

Ans: The DNS query was sent to IP address 10.40.4.44. Yes it is the same IP address as that of my local DNS server.



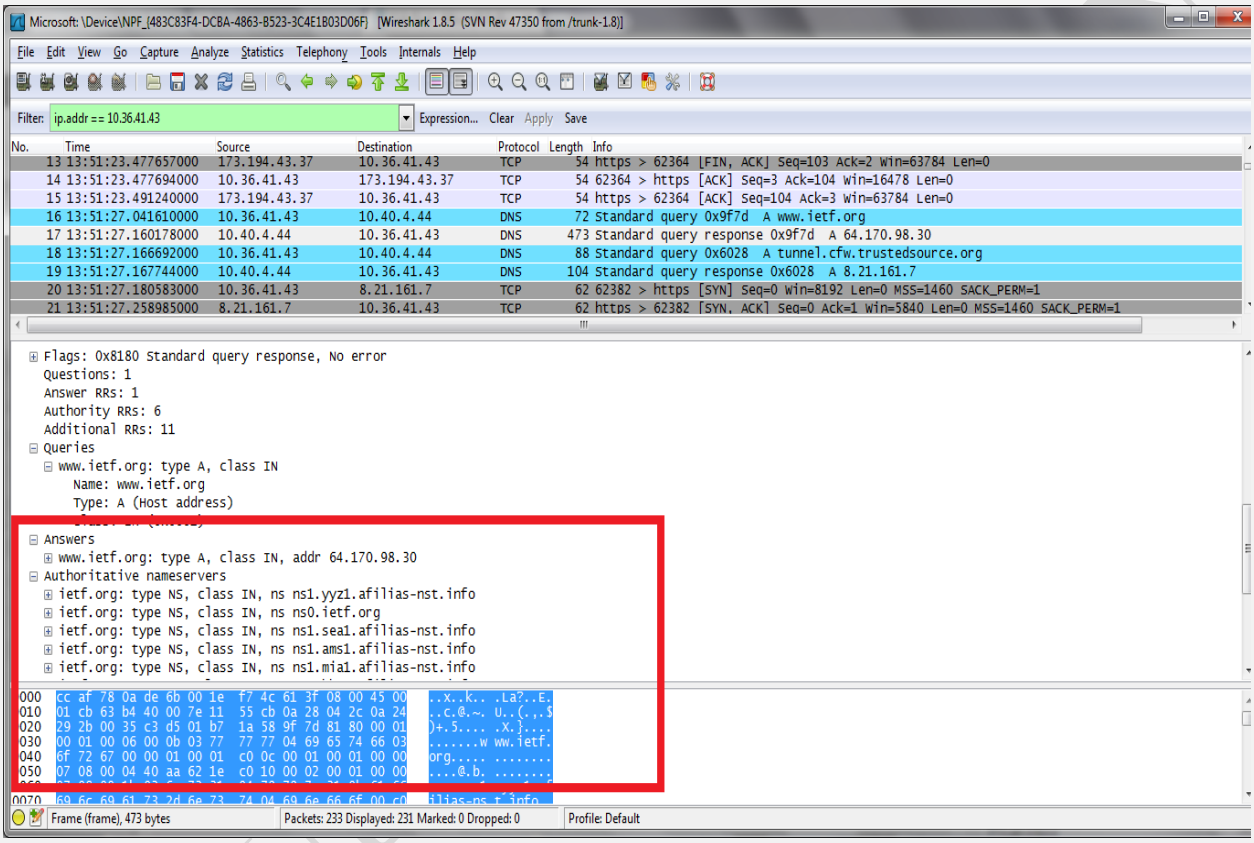
4) Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?

Ans: The query message was a type "A" query, but the message did not contain any "answers."



5) Examine the DNS response message. How many “answers” are provided? What do each of these answers contain?

Ans: The response message contained one answer to the query which was the sites address [64.170.98.30]. Although it also provided 6 authoritative nameservers, and 11 other responses containing additional information.

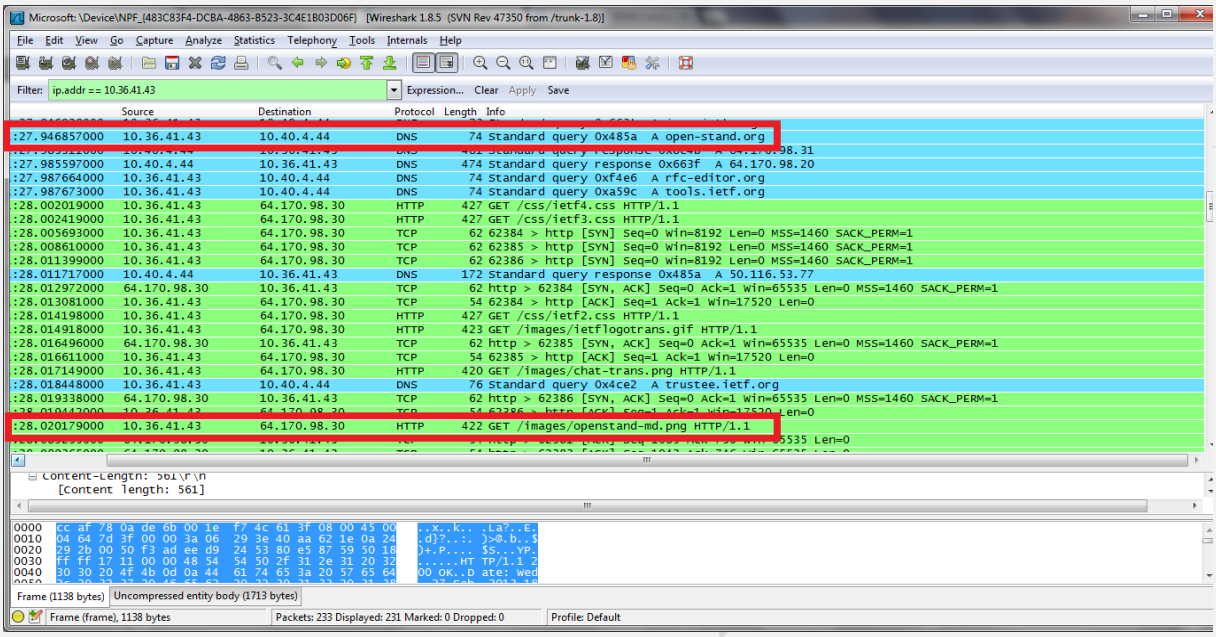


6) Consider the subsequent TCP SYN packet sent by your host. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message?

Ans: The destination of the SYN packet is 64.170.98.30, the same address that was provided in the DNS response message as the type “A” address of the webpage.

7) This web page contains images. Before retrieving each image, does your host issue new DNS queries?

Ans: Yes, my host did issue new DNS queries before the images were retrieved. For example, one such query was for an image from open-stand.org. The image corresponding to the page was not returned until this query was made.

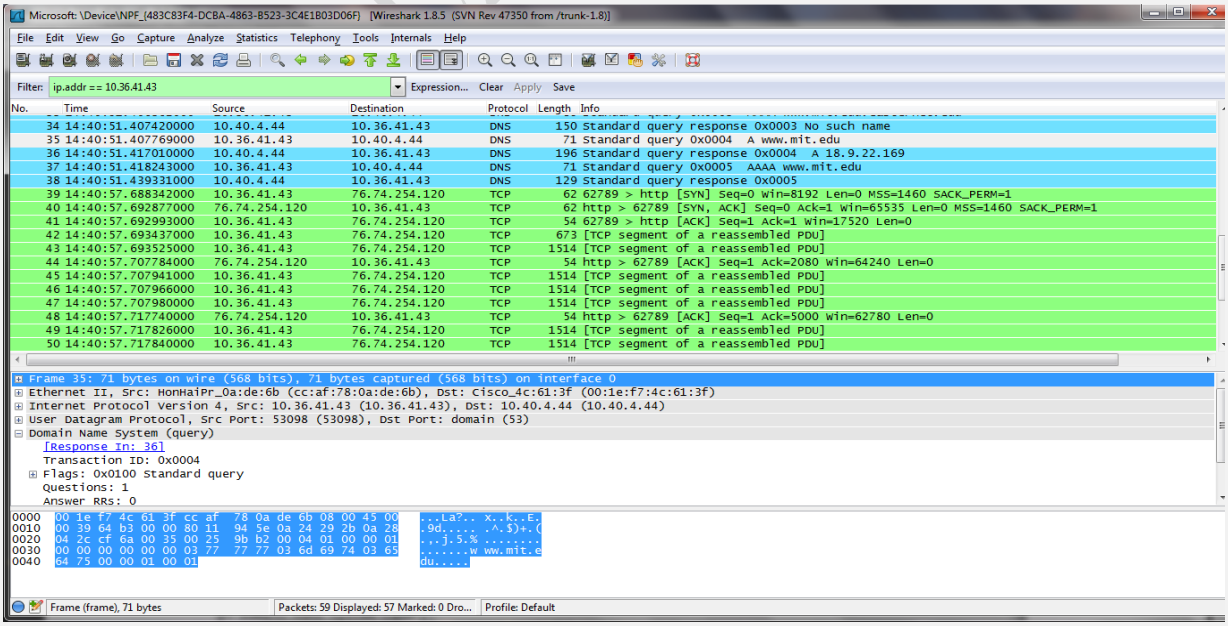


PART.4 TRACING DNS WITH 'NSLOOKUP' CMD

Step 1: Start packet capture.

Step 2: Do an nslookup on www.mit.edu

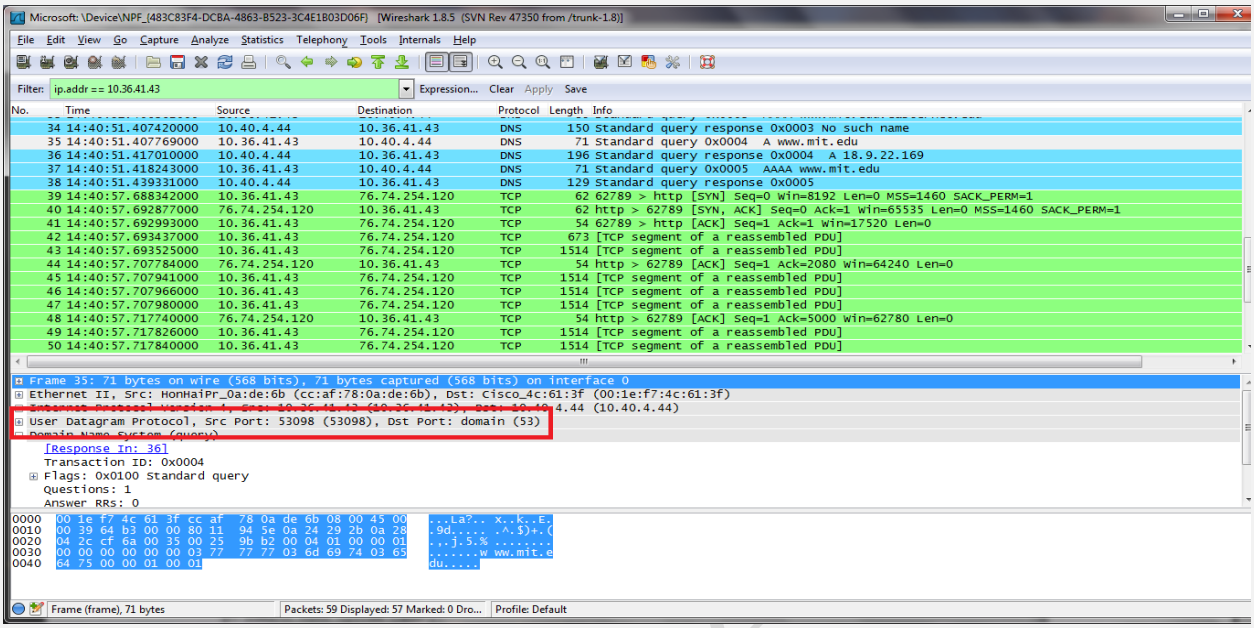
Step 3: Stop packet capture.



QUESTIONS;

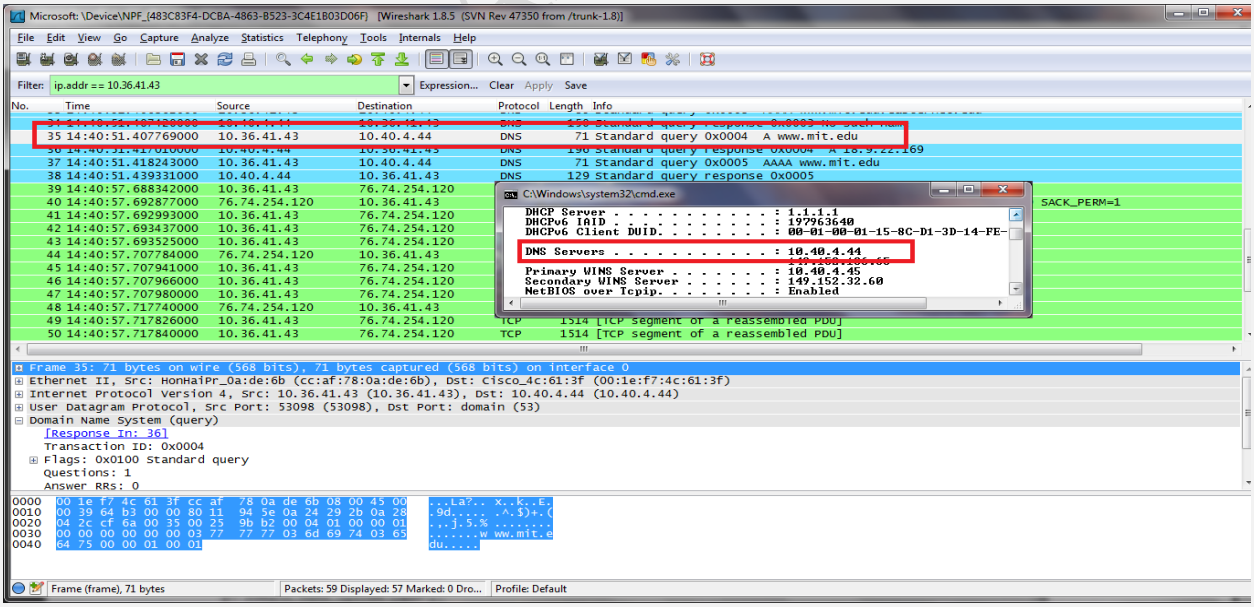
8) What is the destination port for the DNS query message? What is the source port of DNS response message?

Ans: Destination Port: 53 & Source Port: 53098



9) To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?

Ans: The DNS query message is sent to IP address 10.40.4.44, the same address as my default local DNS server.



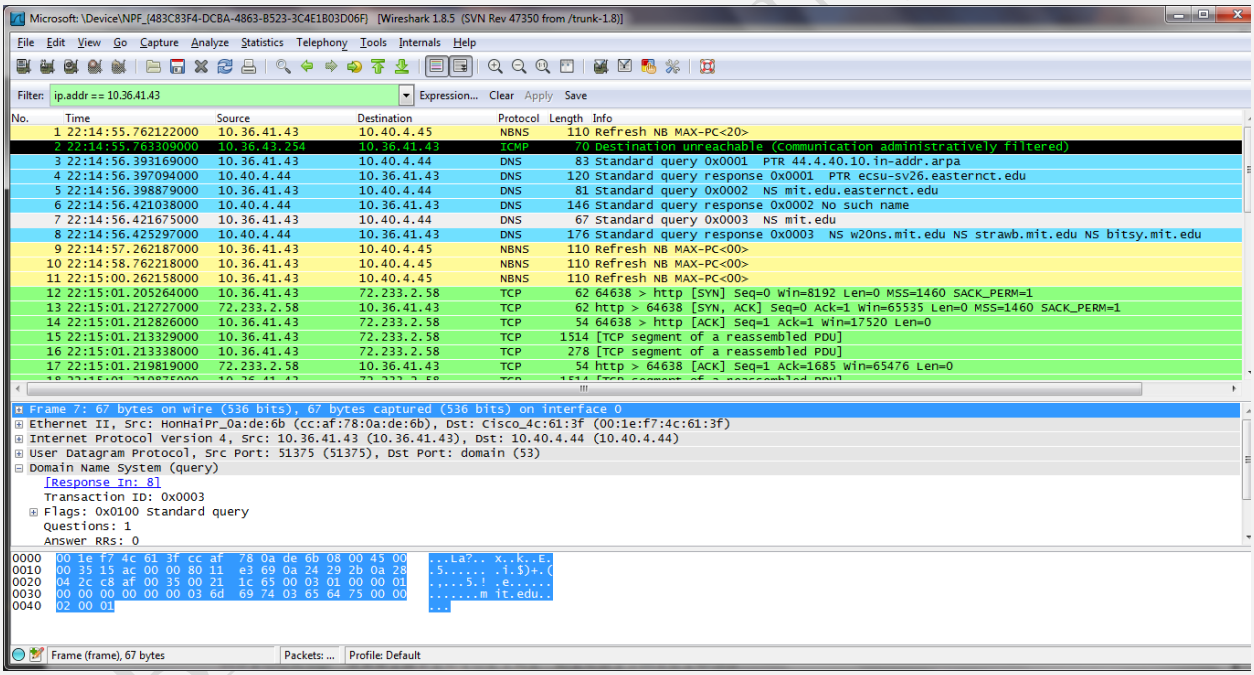
10) Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?

Ans: The DNS query message is a type "A" query, containing only one question and not containing any answers.

11) Examine the DNS response message. How many "answers" are provided? What do each of these answers contain?

Ans: The response message contains one answer to the aforementioned query which is the type "A" address of http://www.mit.edu or 18.9.22.169. It also contained information on 3 authoritative nameservers and 3 additional records.

- **Repeat the previous experiment but instead issue the command;**
  >>nslookup –type=NS mit.edu



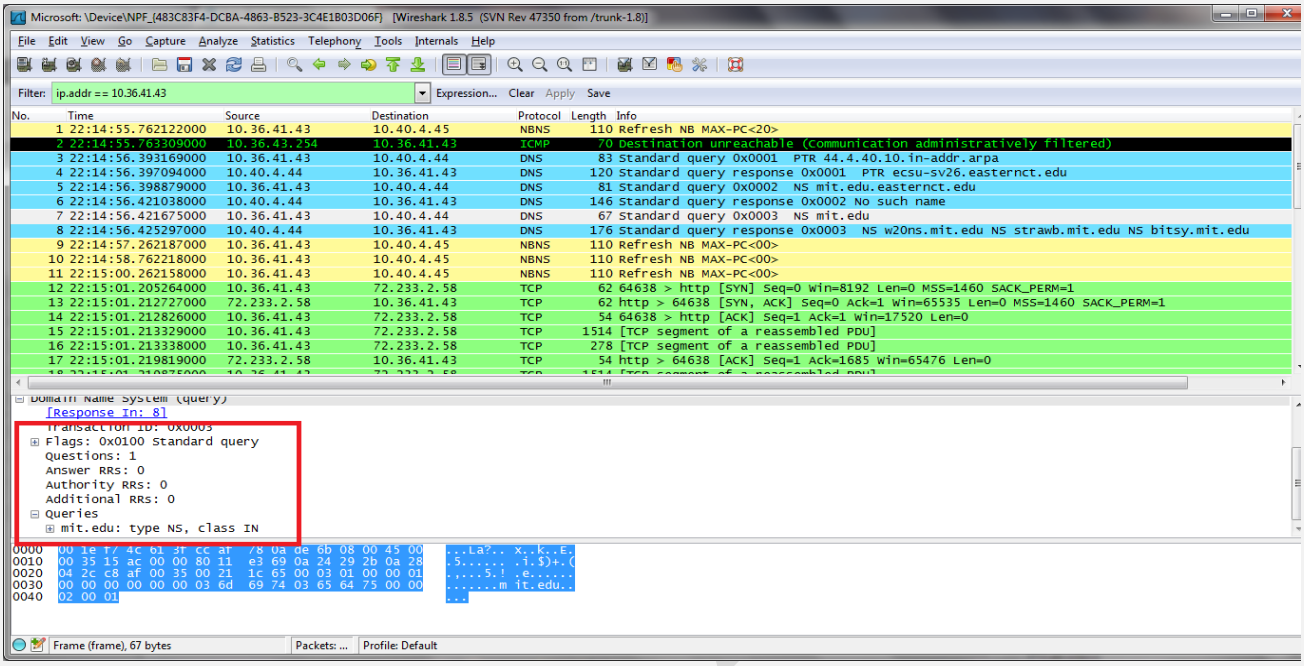
QUESTIONS:

12) To what IP address is the DNS query message sent? Is this the IP address of you default local DNS server?

ANS:The query is sent to 10.40.4.44, the same IP address as that of my default local DNS server.

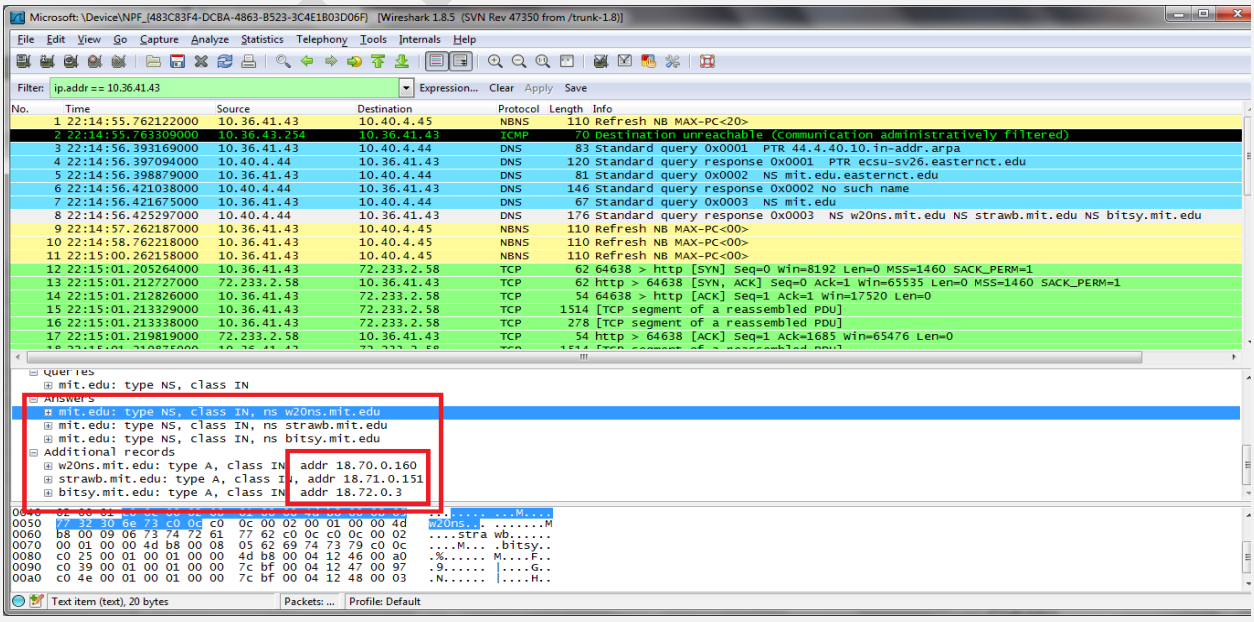
13) Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"

ANS:The DNS query is a type "NS" message including one question. The query message did not contain any answers.

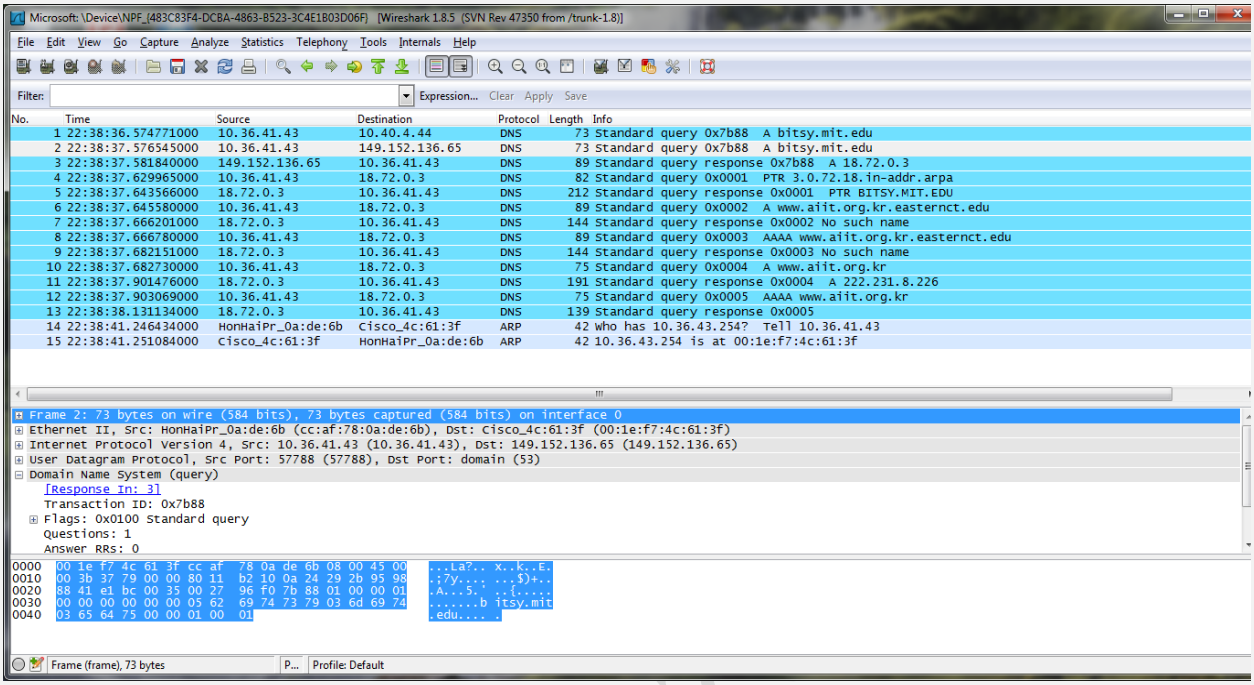


14) Examine the DNS response message. What MIT nameservers does the response message provide? Does this response message also provide the IP addresses of the MIT namesevers?

ANS:The response message provides 3 MIT nameservers: w20ns.mit.edu[18.70.0.160], strawb.mit.edu[18.71.0.150], and bitsy.mit.edu[18.72.0.3]. The IP addresses for the nameservers was included under the additional records category sent back as part of the response message.



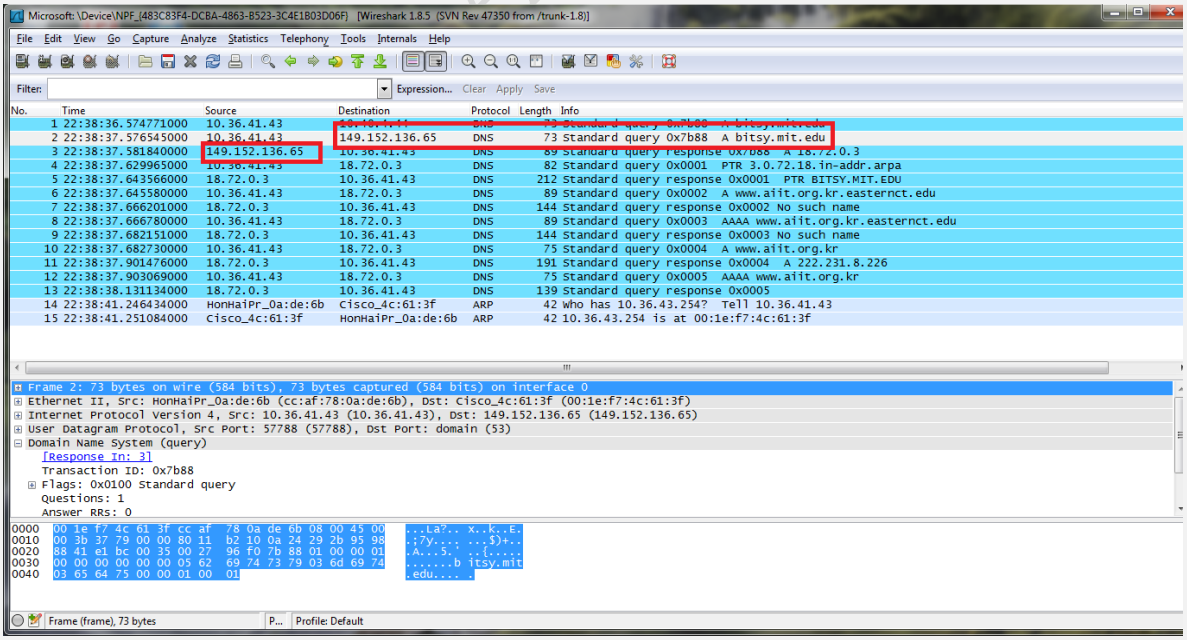
- **Now repeat the previous experiment, but instead issue the command:**
  >>nslookup www.aiit.or.kr bitsy.mit.edu



Answer the following questions;

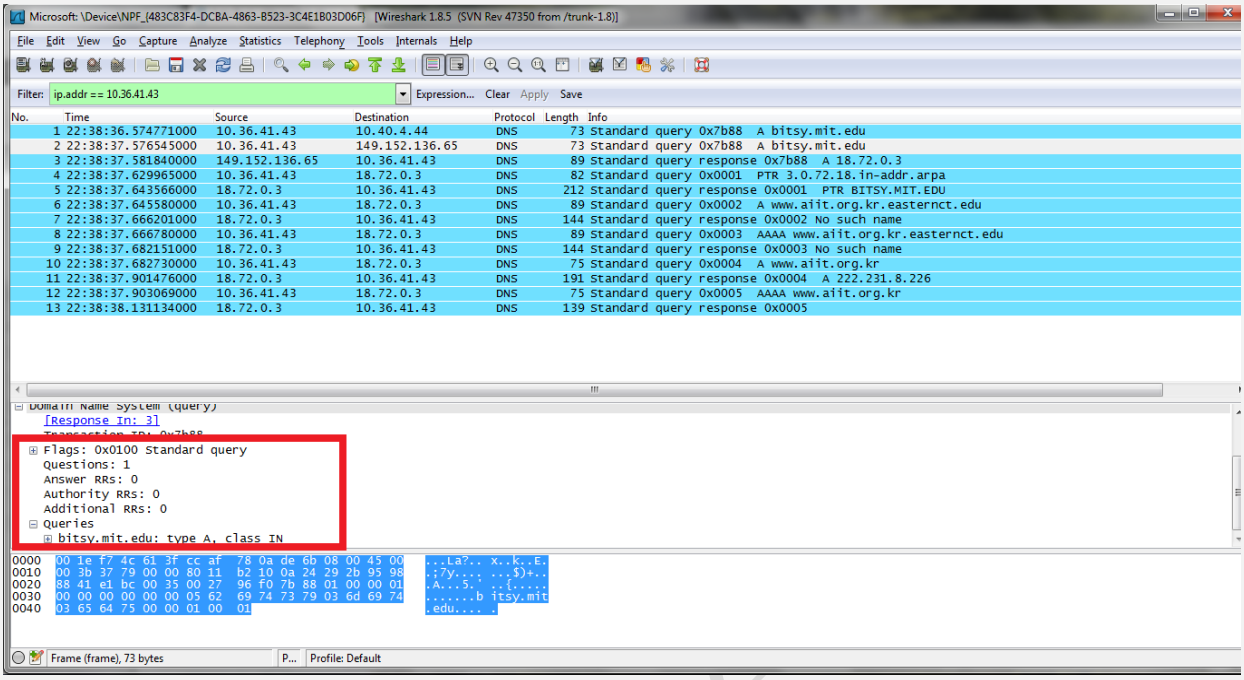
15) To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server? If not, what does the IP address correspond to?

Ans: This DNS query message is sent to 149.152.136.65 which is the IP address of the MIT DNS response sender.



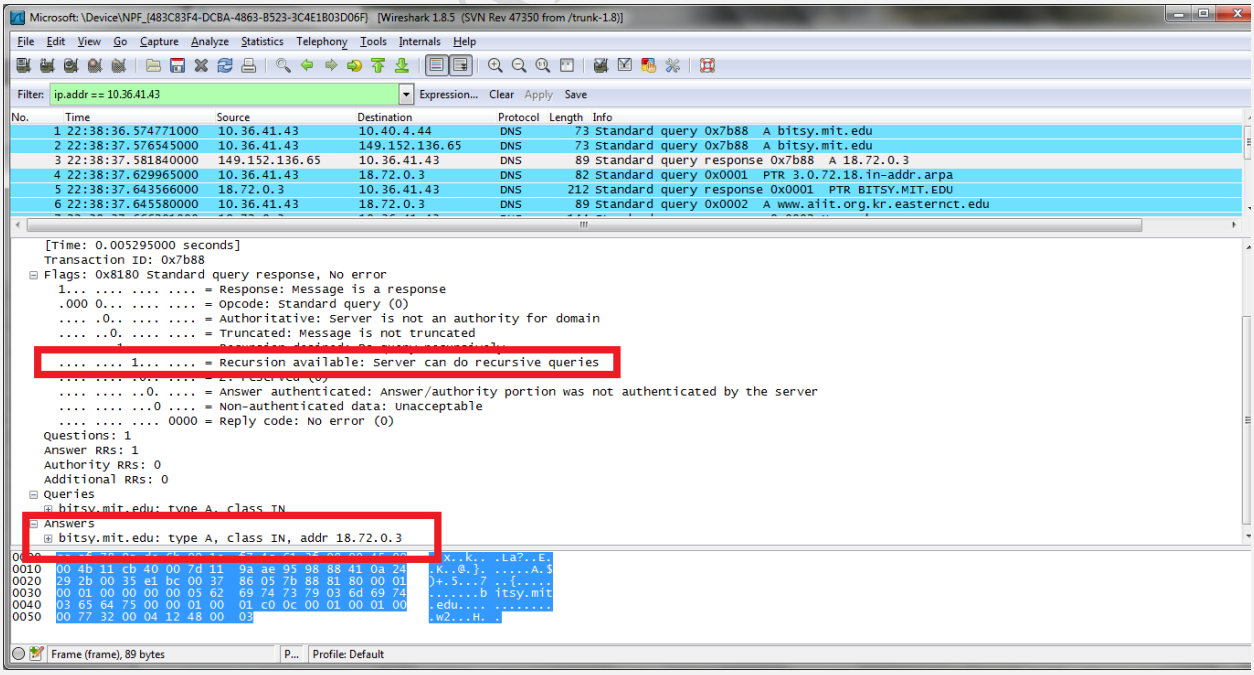
16) Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?

Ans: This DNS query is a type "A" query. The message does not contain any answers.



17) Examine the DNS response message. How many "answers" are provided? What does each of these answers contain?

ANS: It only provided one "answer" containing the servers IP address, however, the server also returned a flag that stated that it could complete a recursive query.



LAB NO.10

NAME *: ABDULLAH ZUNORAIN*

REG NO: *19JZELE0338*

SUBJECT*: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO: *SYED UZAIR GILLANI*

SECTION: *A*

TITLE: ANALYSIS OF FTP PROTOCOL USING CMD & WIRESHARK

OBJECTIVES:

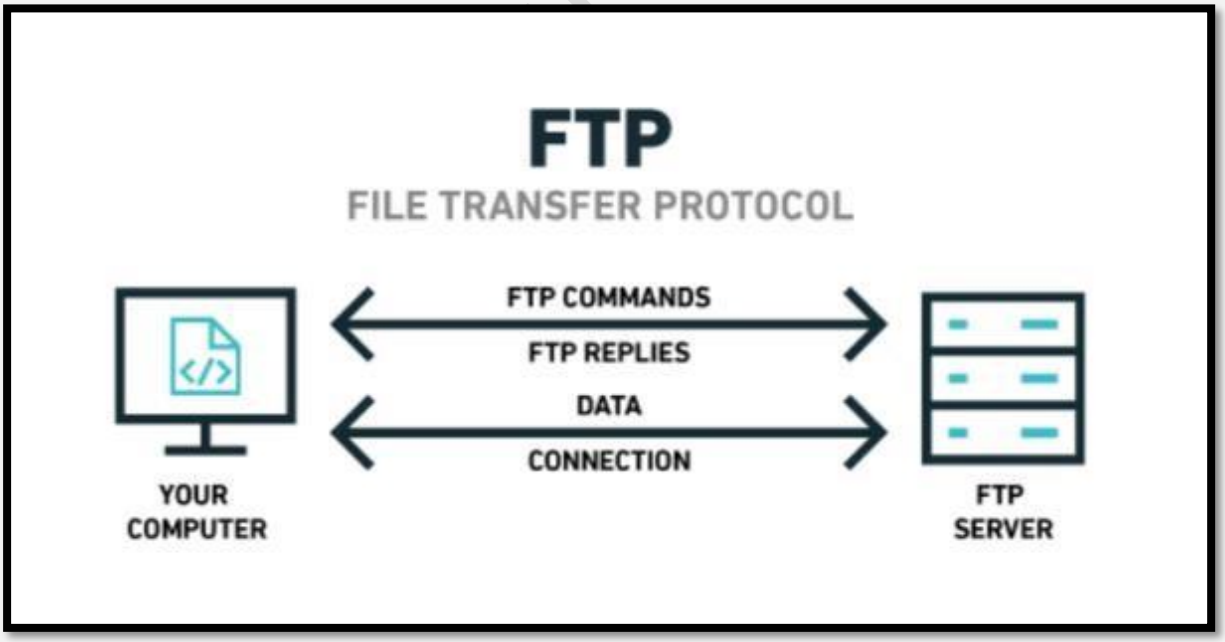
- Understanding how File Transfer Protocol works.
- Capturing Packets while working on FTP.
- Using Filters analyze the FTP packets.

THEORY:

FTP PROTOCOL:

FTP (File Transfer Protocol) is a network protocol for transmitting files between computers over Transmission Control Protocol/Internet Protocol (TCP/IP) connections. Within the TCP/IP suite, FTP is considered an application layer protocol.

In an FTP transaction, the end user's computer is typically called the *local host*. The second computer involved in FTP is a *remote host*, which is usually a server. Both computers need to be connected via a network and configured properly to transfer files via FTP. Servers must be set up to run FTP services, and the client must have FTP software installed to access these services.

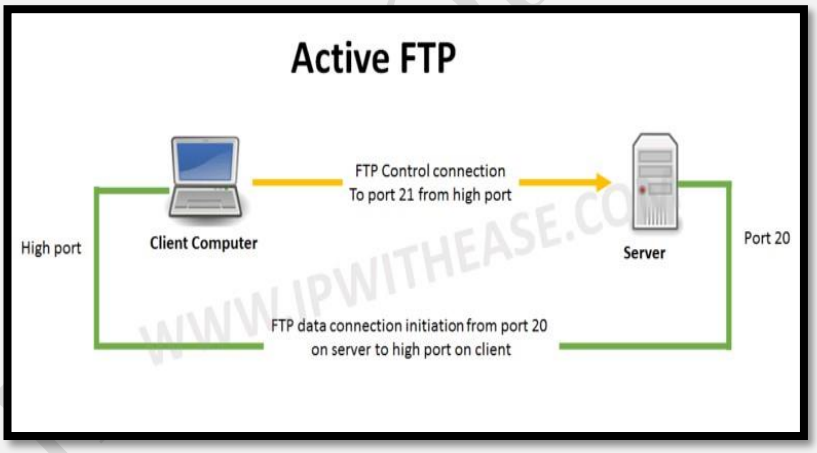


Working principle:

FTP is a client-server protocol that replies on two communications channels between the client and server: a command channel for controlling the conversation and a data channel for transmitting file content. Here is how a typical FTP transfer works:

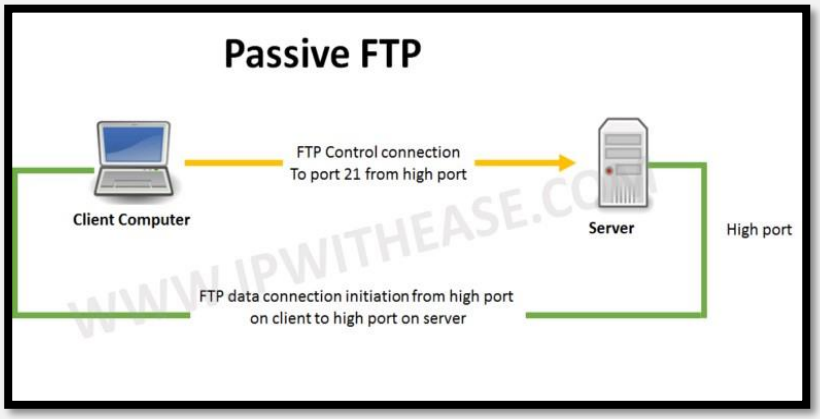
1. A user typically needs to log on to the FTP server, although some servers make some or all of their content available without a login, a model known as *anonymous FTP*.
2. The client initiates a conversation with the server when the user requests to download a file.
3. Using FTP, a client can upload, download, delete, rename, move and copy files on a server.

File Transfer Protocol may run in **active** or **passive** mode, which determines how the data connection is established.



In an **Active FTP** mode, the client opens a port and listens. It sends the FTP command PORT M to inform the server on which port it is listening. The server actively connects to the client from its port 20, the FTP server data port.

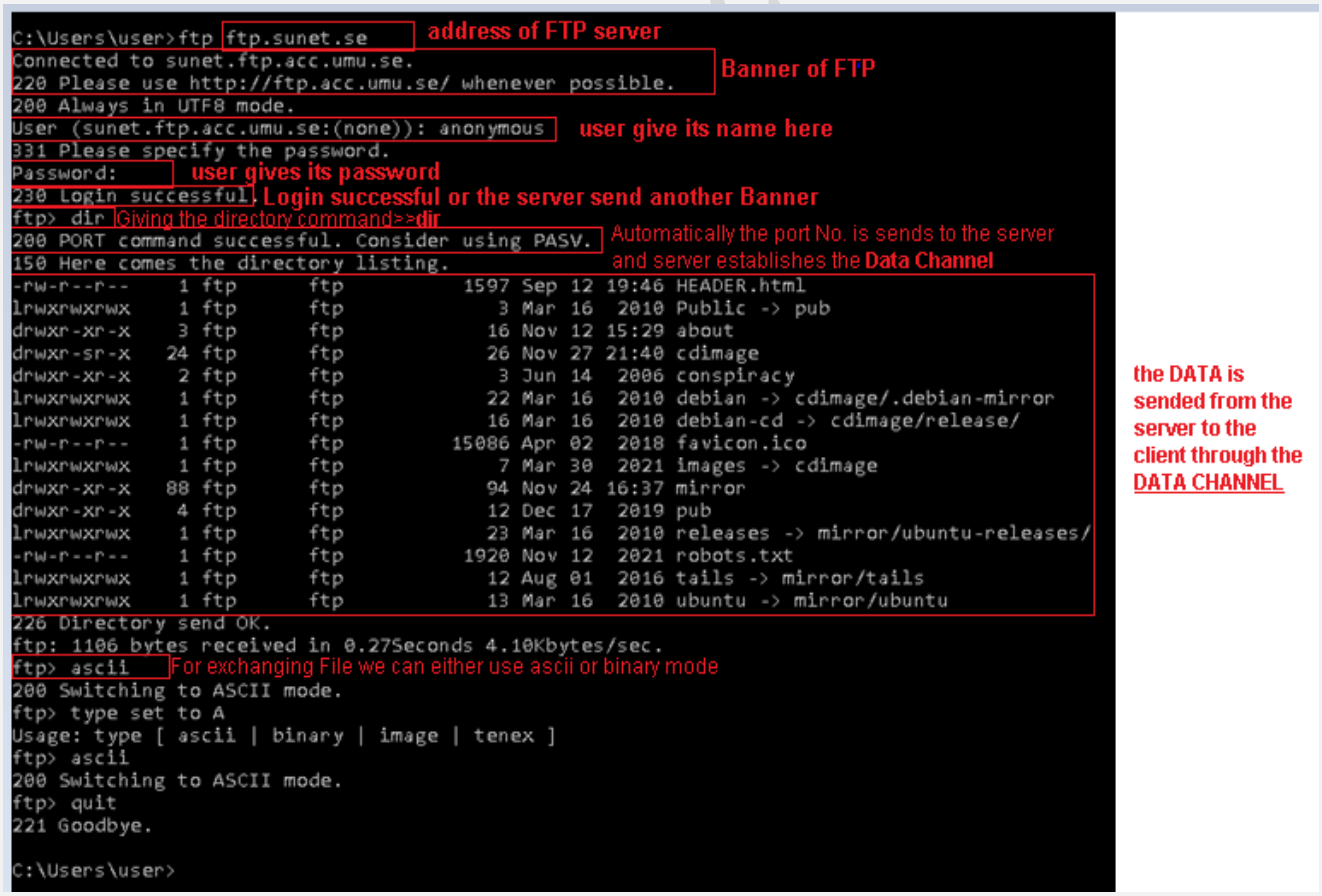
In a **Passive FTP** mode, the server opens a port and listens (passively) and the client uses the control connection to send a **PASV** command to the server and then receives a server IP address and server port number from the server for the client connect to it.



Passive mode is used generally where the client is behind a firewall and unable to accept incoming TCP connections. From company It security perspective, passive FTP mode is preferred safety measure. **FTP client** programs select passive connection mode by default because server administrators prefer it as a safety measure. Firewalls generally block connections that are "initiated" from the outside

1. USING FTP COMMAND IN CMD (ACTIVE MODE)

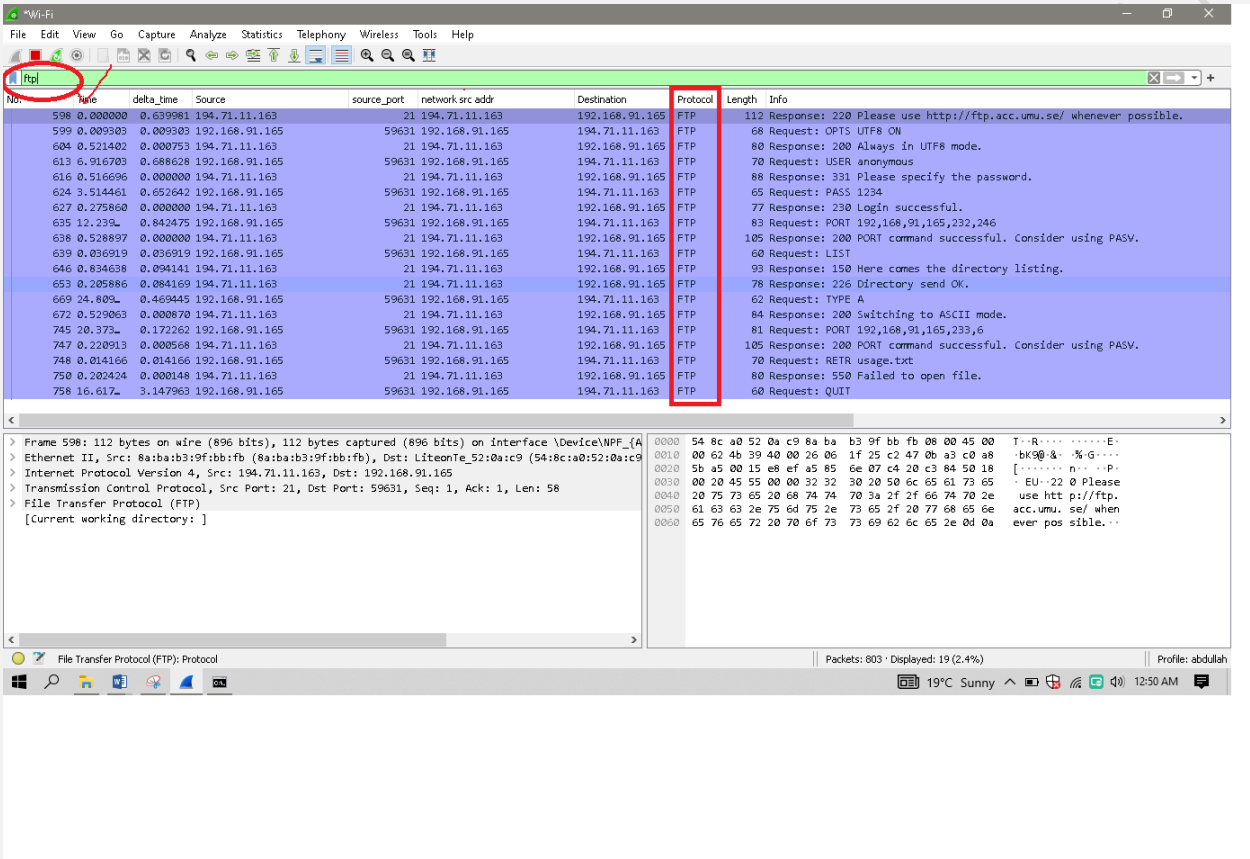
- First of all, established the Internet Connection and run the command prompt as administrator.
- If FTP server connection with the Client succeeds, then server gives you a brief info about itself and send back a banner.
- The client request from the server on the command line. If connection succeeds, you will be asked for username and password. In our case, username will be anonymous and gives password whatever you want.
- After that, open wire-shark and start capturing packets while connecting to the host server directory.
- Use the FTP filter in wire-shark to capture packets.
- Type dir (directory) command in the ftp protocol using command prompt to see the files and folders on the host server.
- Now, client can tell the server that in which type you want to read data asci or binary form.
- Now, a client can edit, move, copy, rename, delete files from the server.
- To exit from the ftp host server type quit command and press enter.



FIG_a (Using FTP protocol in Active Mode)

➢ ANALYZE THE FTP NETWORK TRAFFIC IN WIRESHARK:

Here, we open the Wireshark and click on the start capturing. So, it will automatically capture all the network traffic working during the CMD while making a connection between client and server. Given below is the capture network traffic of FTP ;



FIG_b (CAPTURING FTP TRAFFIC USING WIRESHARK)

In the above fig, first of all sending a request from the client to the server on the Command line, then it requires a login with USER NAME and PASSWORD.

Now, if we want to check the file list on the server, type DIR or List.

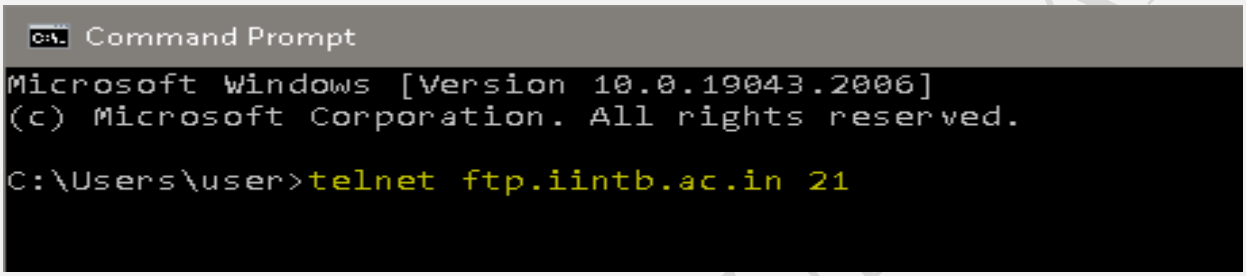
Finally, we get a list of files on the server and we can easily access through it data connection port.

Now, we can easily transfer data to each other.

2. USING FTP COMMAND IN CMD (PASSIVE MODE)

In a **Passive FTP** mode, the server opens a port and listens (passively) and the client uses the control connection to send a **PASV** command to the server and then receives a server IP address and server port number from the server for the client connect to it.
Now, the client got the Port number of server so, client sends a outgoing message to the message and as we know that the Outgoing message is always allow by firewall that's why passive mode is used.

➢ CREATION OF COMMAND CHANNEL:



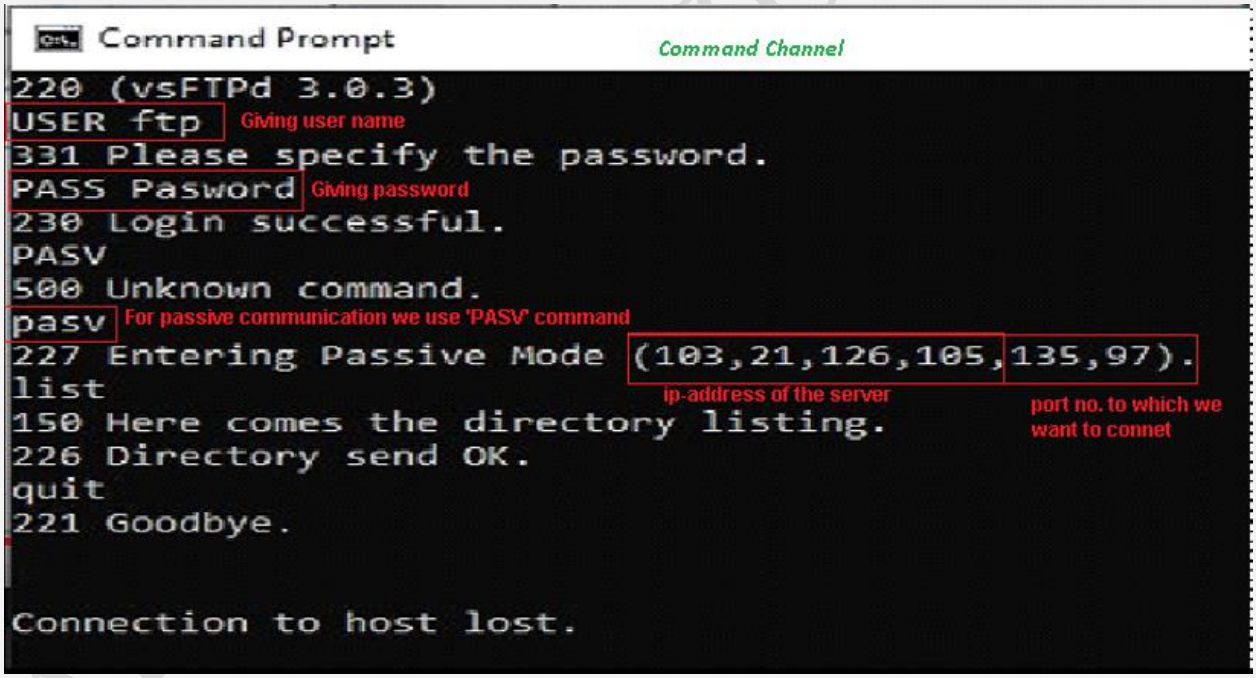


FIG-Using FTP protocol in PASSIVE MODE

PROCEDURE:

First of all, we should TURN ON the telnet to communicate the client and server in passive mode.

Type telnet **ftp.iitb.ac.in 21** in the Command Window Prompt.

Here, **telnet** is the protocol and **ftp.iitb.ac.in** is the server which we want to communicate with it and **21** is the port number for connection line.

Now, it requires the login info after that type the mode in which you want to communicate. So, here we wants to use the Passive mode that's why type **pasv** in cmd.

In return, the server sends its ip address and port number in control connection.

➢ CREATION OF DATA CONNECTION:

Now, we create another CMD and type telnet IP Address of Server and Port number.

telnet 103.21.126.105 34657

Here, if we send a **List command** from the control connection. So, we receive the list directory command on the data connection on the server side.

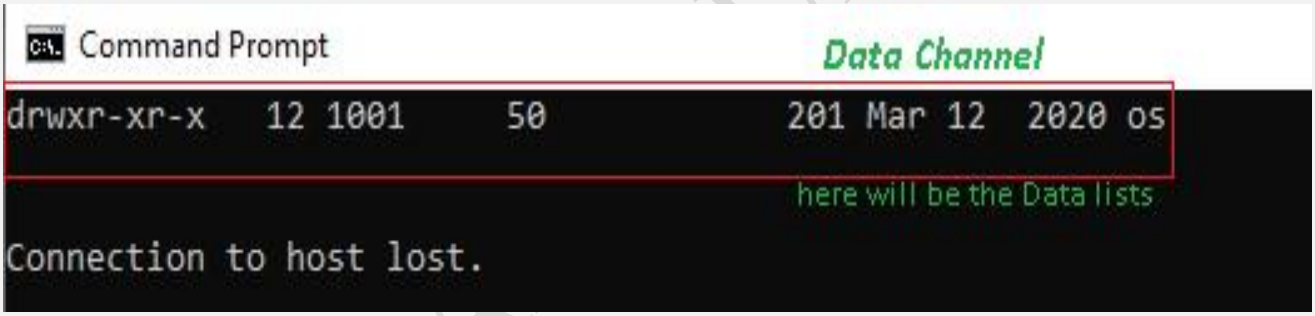


FIG-List Directory of Server IP address

➢ ANALYZE THE FTP NETWORK TRAFFIC IN WIRESHARK USING PASSIVE MODE:

Here, we open the Wireshark and click on the start capturing. So, it will automatically capture all the network traffic working during the CMD while making a connection between client and server. Given below is the capture network traffic of FTP.

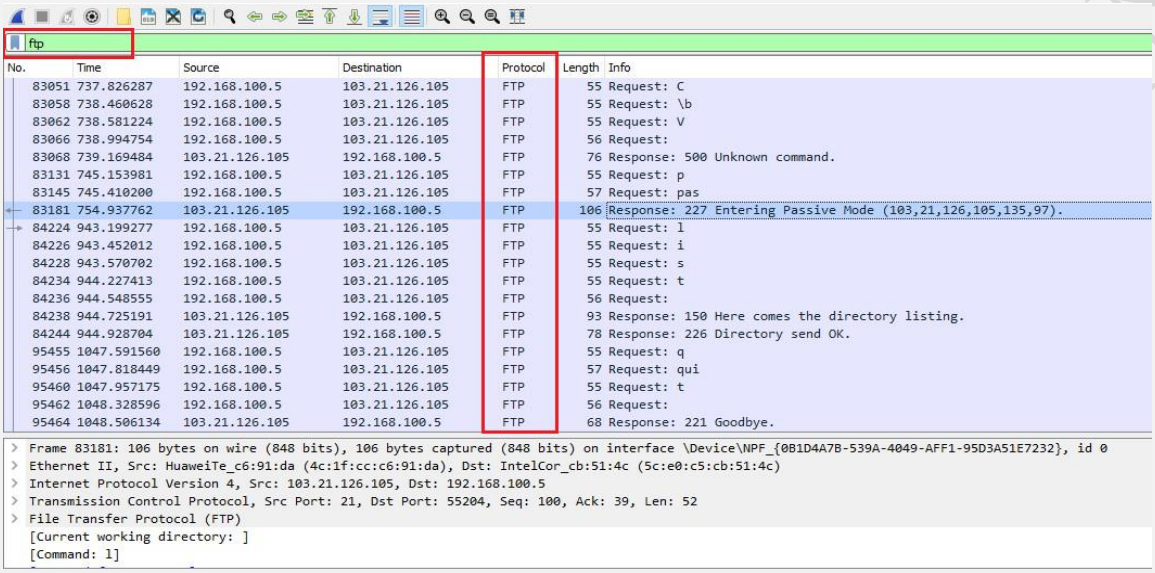


FIG-CAPTURING FTP TRAFFIC USING WIRESHARK IN PASSIVE MODE

Here, we first enter USERNAME and PASSWORD for login. After that in which mode you want to communicate so that's why we capture Passive Mode packet in Wireshark.

LAB NO. 11

NAME *: ABDULLAH ZUNORAIN*

REG NO: *19JZELE0338*

SUBJECT*: COMPUTER COMMUNICATION NETWORKING*

SUBMITTED TO: *SYED UZAIR GILLANI*

SECTION: *A*

DEPT: *ELECTRICAL COMMUNICATION*

CAMPUS*: JALOZAI*

TITLE: ANALYSIS OF SMTP PROTOCOL USING CMD & Wireshark

OBJECTIVES:

- Understanding how Simple Mail Transfer Protocol (SMTP) works.
- Capturing Packets while working on SMTP.
- Using Filters analyze the SMTP packets.

THEORY:

SMTP PROTOCOL

SMTP is an application layer protocol. The client who wants to send the mail opens a TCP connection to the SMTP server and then sends the mail across the connection. The SMTP server is an always-on listening mode. As soon as it listens for a TCP connection from any client, the SMTP process initiates a connection through port 25. After successfully establishing a TCP connection the client process sends the mail instantly.

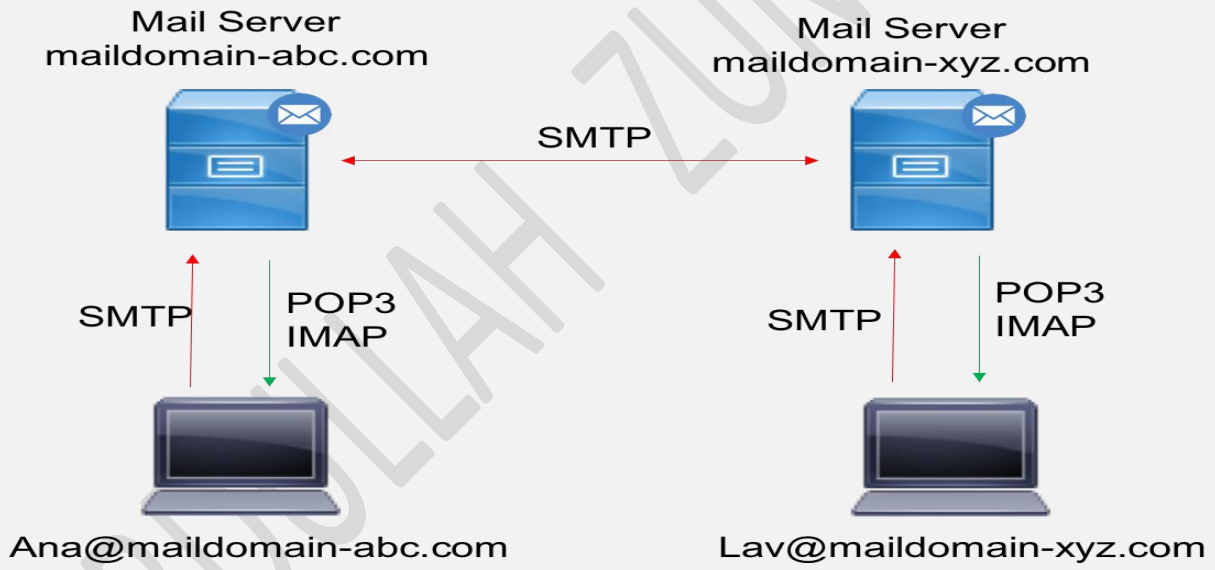
The SMTP model is of two types:

- End-to-end method
- Store-and- forward method

The end-to-end model is used to communicate between different organizations whereas the store and forward method is used within an organization. An SMTP client who wants to send the mail will contact the destination's host SMTP directly, in order to send the mail to the destination. The SMTP server will keep the mail to itself until it is successfully copied to the receiver's SMTP.
The client SMTP is the one that initiates the session so let us call it client- SMTP and the server SMTP is the one that responds to the session request so let us call it receiver-SMTP. The client- SMTP will start the session and the receiver-SMTP will respond to the request.

Model of SMTP system:

In the SMTP model user deals with the user agent (UA), for example, Microsoft Outlook, Netscape, Mozilla, etc. In order to exchange the mail using TCP, MTA is used. The user sending the mail doesn't have to deal with MTA as it is the responsibility of the system admin to set up a local MTA. The MTA maintains a small queue of mails so that it can schedule repeat delivery of mails in case the receiver is not available. The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.



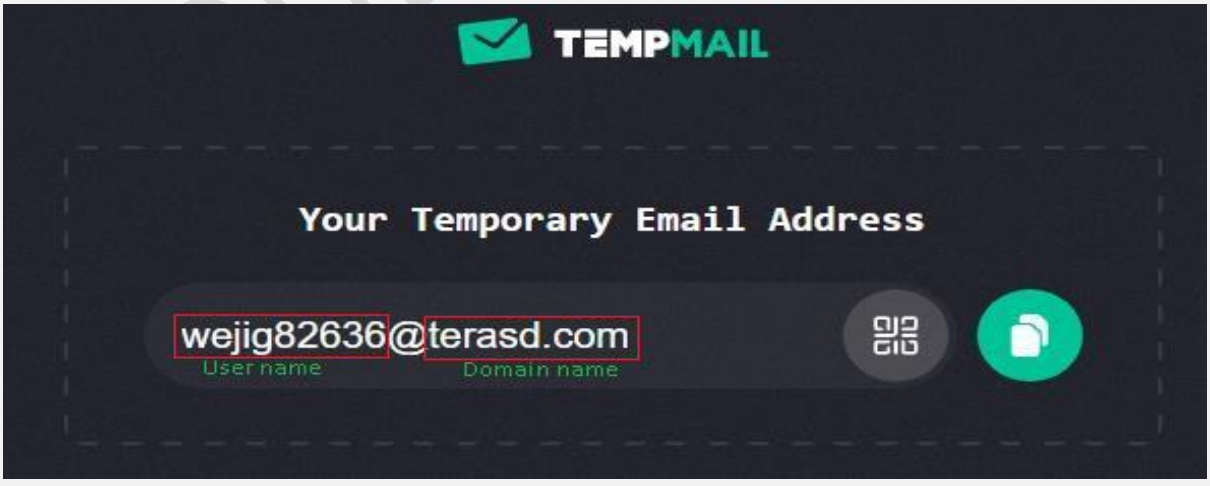
As the diagram above displays, the client Ana@maildomain-abc.com sends email to the MTU server via SMTP and retrieves email via either POP3 or IMAP. The same is true for the other client, Lav@maildomain-xyz.com. Communication between the email servers or MTU's is exclusively SMTP on port 25. POP3 uses port 110 amd IMAP uses port 143.

WORKING PRINCIPLE (In detail):

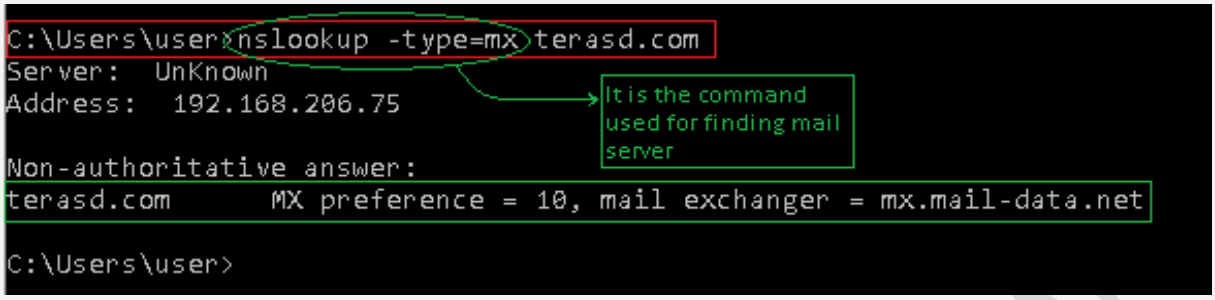
1. **Composition of Mail:** A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.
2. **Submission of Mail:** After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port 25.
3. **Delivery of Mail:** E-mail addresses contain two parts: username of the recipient and domain name. For example, vivek@gmail.com, where "vivek" is the username of the recipient and "gmail.com" is the domain name. If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.
4. **Receipt and Processing of Mail:** Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.
5. **Access and Retrieval of Mail:** The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.

PROCEDURE OF LAB TASK:

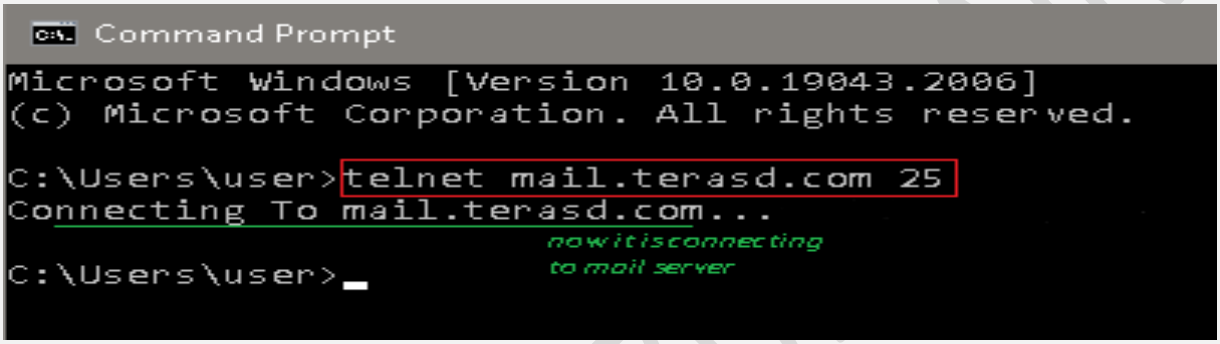
1. First of all open the **Telnet Client** and then open **wire shark** and start *capturing* **smpt** packets.
2. Open **temp-mail.org website** it will assign you a *temporary email*.



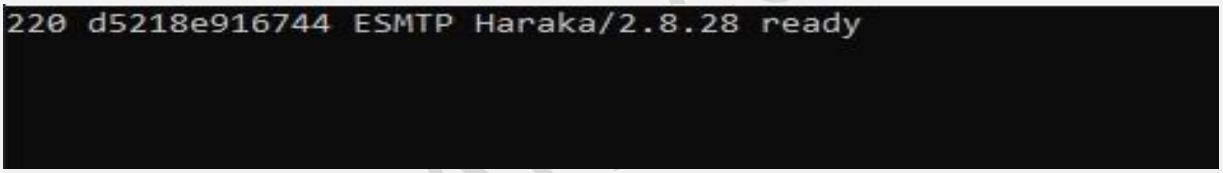
3. Now open **cmd** and enter **domain name** of the above email using **nslookup** command.



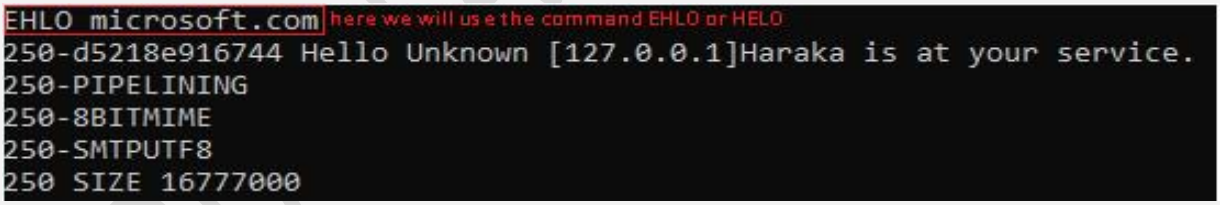
4. Now enter following command in cmd.
>>telnet name of the mail-server



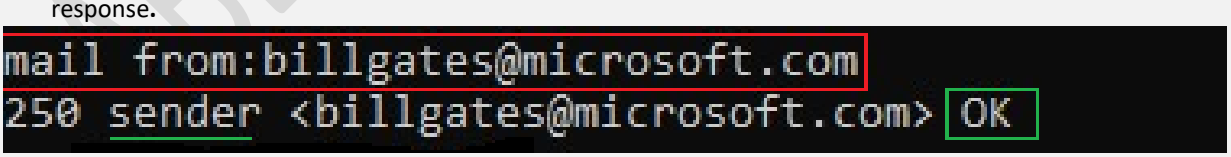
5. It will show you the below screen.



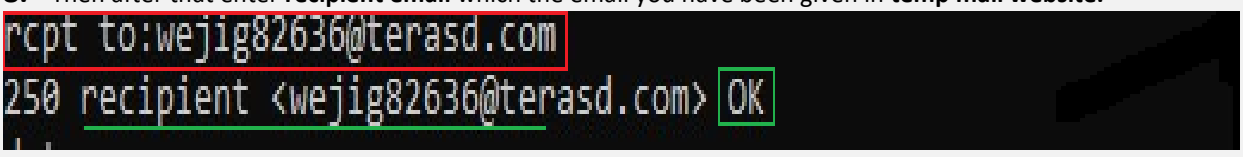
6. Then enter the **EHLO mcirsoft.com** and you will see the following response



7. Now we have to give the **email** through which we will send **message** and in response you get following response.

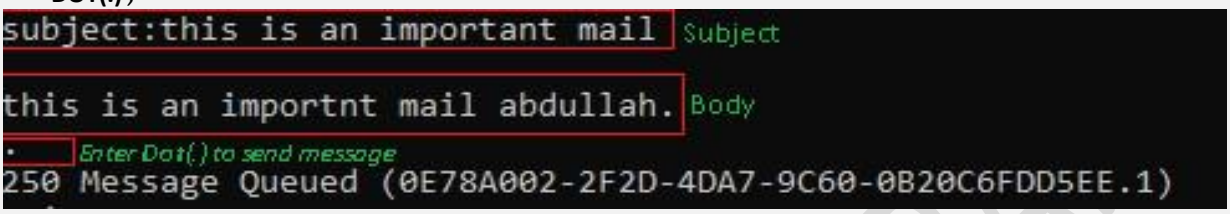


8. Then after that enter **recipient email** which the email you have been given in **temp mail website.**

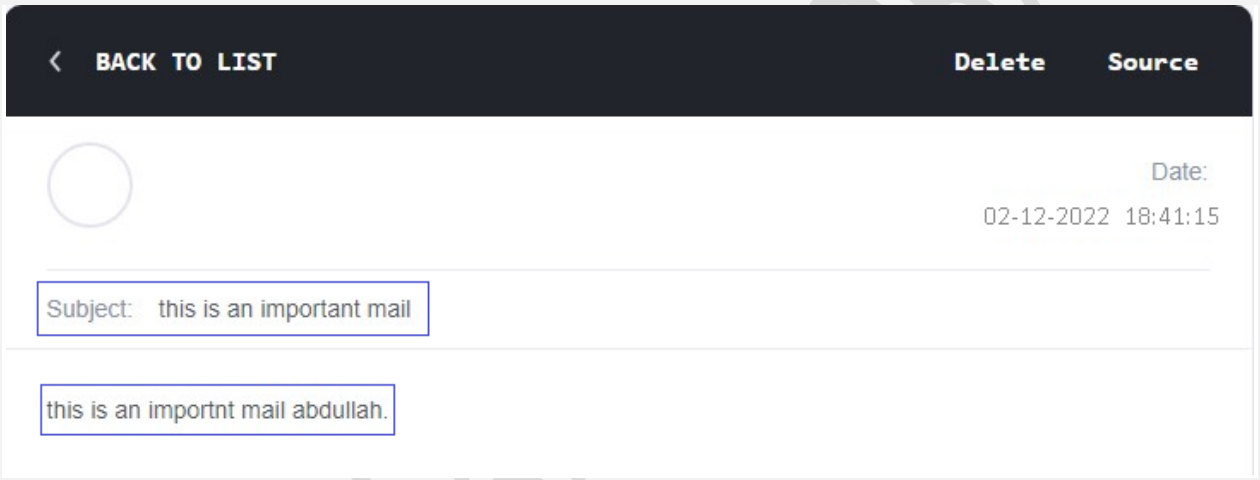


9. Now we will enter **data** and in response we will get.

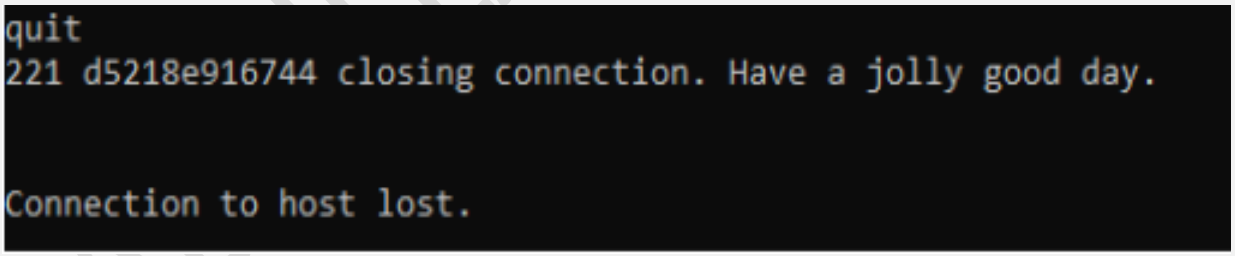
10. Now we can enter the **subject of the message** and **body** as well just as follow and at last we have to enter **DOT(.)** ;



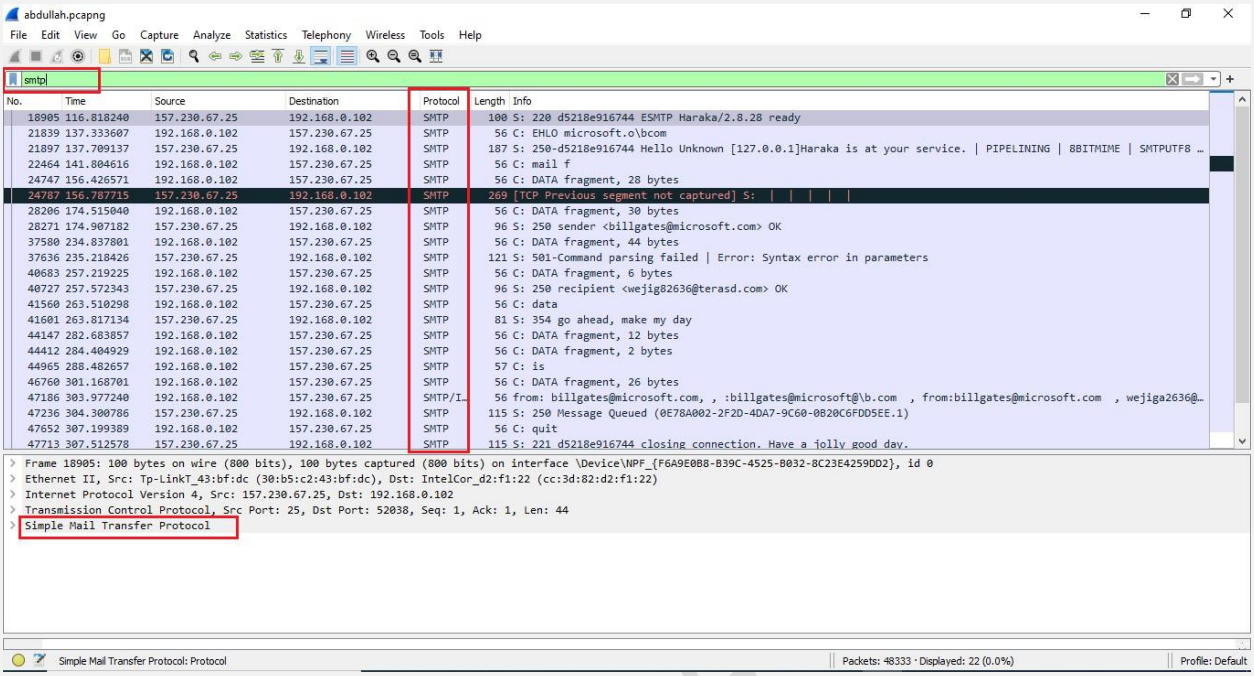
11. After that check the **inbox** of the recipient.



12. After that we will quit and get the following response.

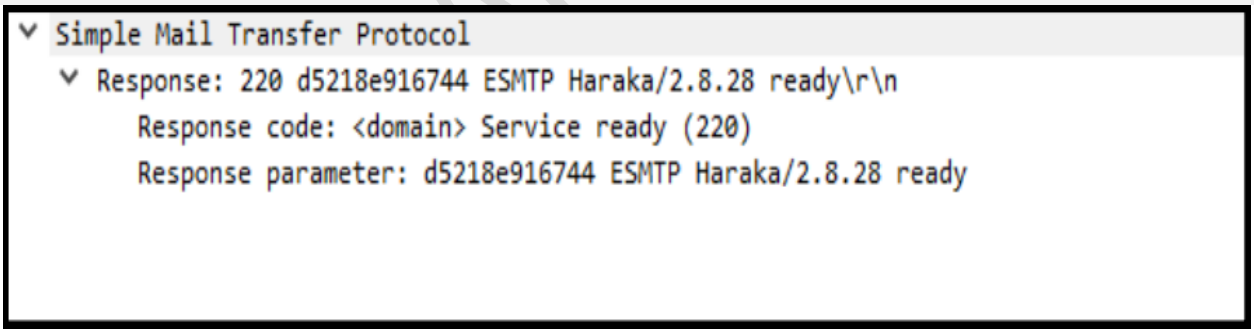


Wireshark Capturing SMTP Packets:



Now we will start to analyze the **smtp traffic**;

First of all we have tried to enter **telnet mail.servergem.com** and get.



Then we have enter **EHLO Microsoft.com**

```
Simple Mail Transfer Protocol
  Command Line: EHLO microsoft.o\bcom\r\n
      Command: EHLO
      Request parameter: microsoft.o\bcom
```

Then we received following response.

```
Simple Mail Transfer Protocol
  Response: 250-d5218e916744 Hello Unknown [127.0.0.1]Haraka is at your service.\r\n
      Response code: Requested mail action okay, completed (250)
      Response parameter: d5218e916744 Hello Unknown [127.0.0.1]Haraka is at your service.
      Response parameter: PIPELINING
      Response parameter: 8BITMIME
      Response parameter: SMTPUTF8
      Response parameter: SIZE 16777000
```

After that we have given following command.

```
Simple Mail Transfer Protocol
  Command Line: mail from:billgates@microsoft.com\r\n
      Command: mail
      Request parameter: from:billgates@microsoft.com
```

And received following response.

```
Simple Mail Transfer Protocol
  Response: 250 sender <billgates@microsoft.com> OK\r\n
      Response code: Requested mail action okay, completed (250)
      Response parameter: sender <billgates@microsoft.com> OK
```

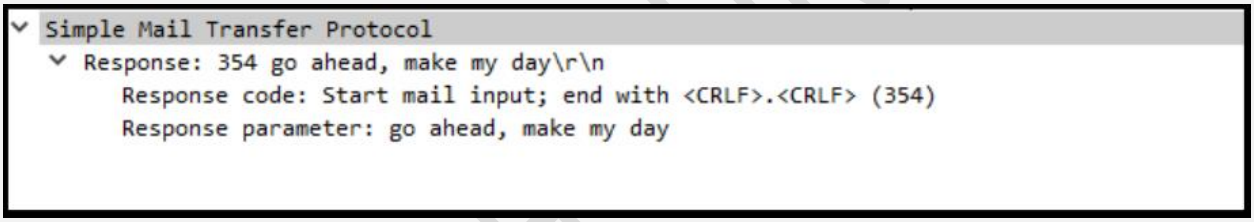
After that we have given **rcpt address**.

```
Simple Mail Transfer Protocol
  Response: 250 recipient <wejig82636@terasd.com> OK\r\n
```

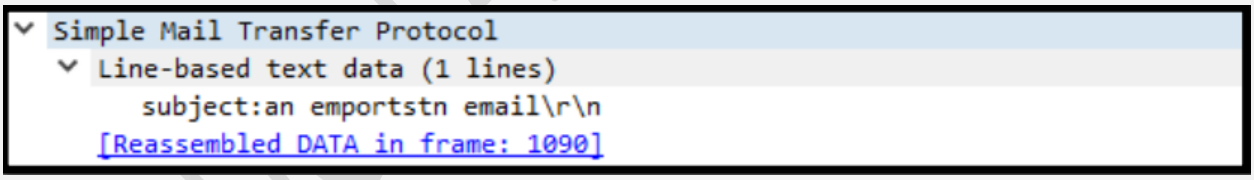
And got following response.

After that we have entered **data command**.

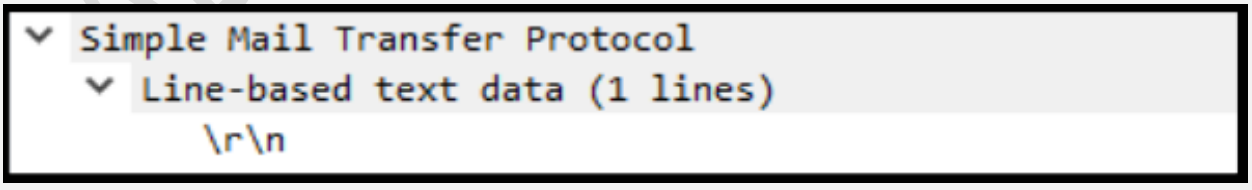
And then got following response



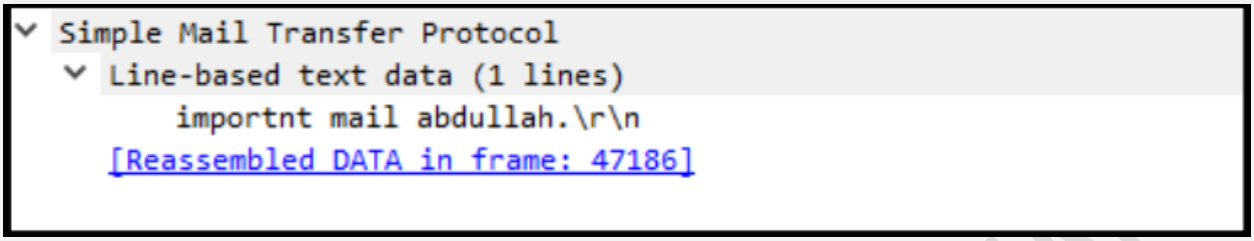
And then we type **subject header**.



And then given one enter.

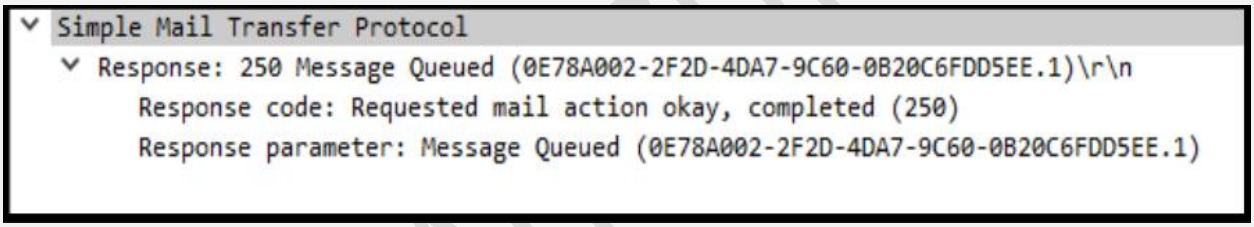


And then typed **message body.**

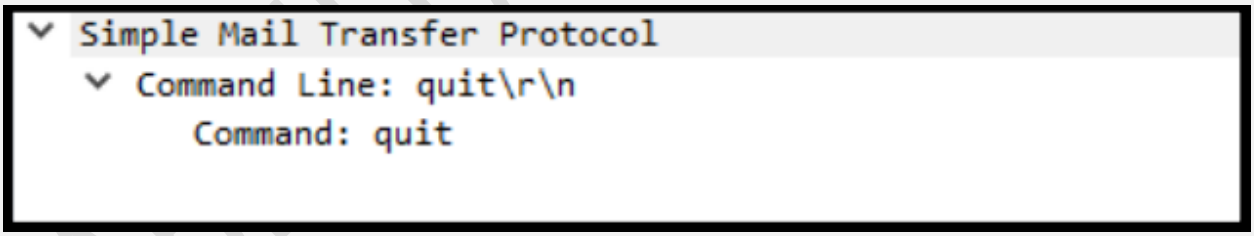


And send the **entire message** as below.

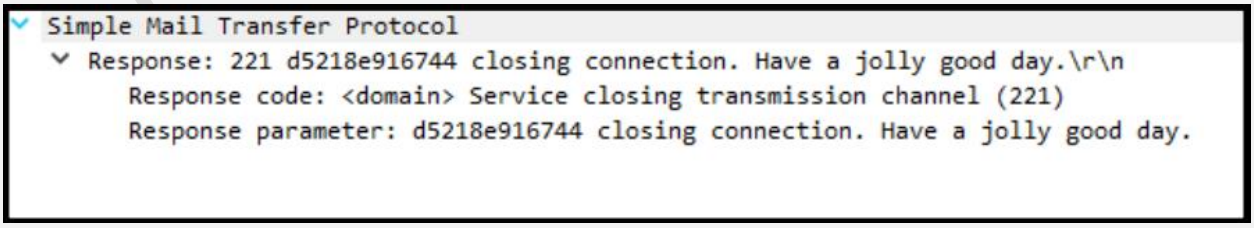
And then got response of



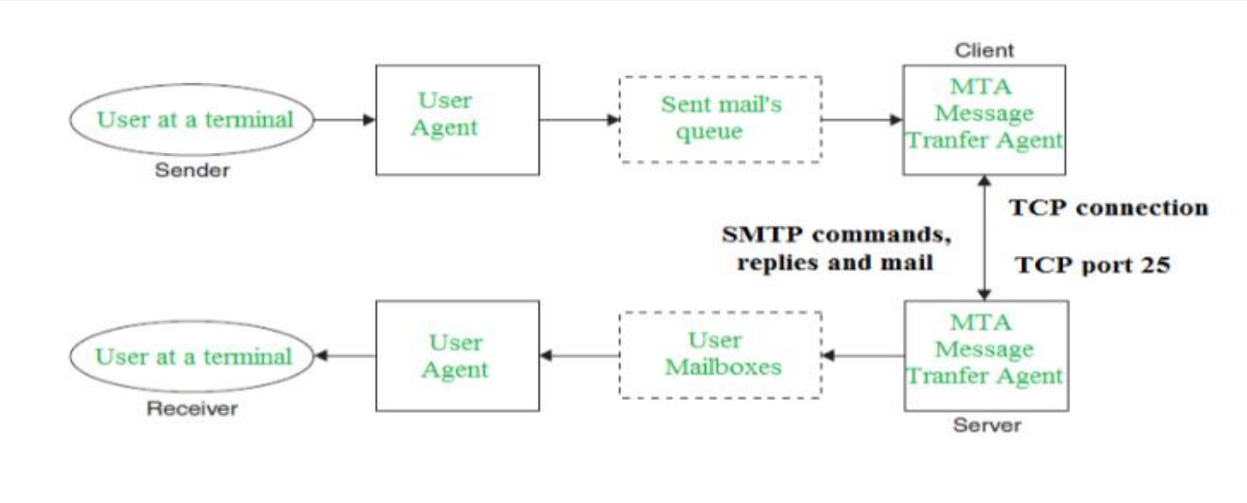
After that we give **quit** command.



And got following response.



❖FURTHER THEORY:



Both the SMTP-client and SMTP-server should have 2 components:

- User-agent (UA)
- Local MTA

Communication between sender and the receiver :

The sender's user agent prepares the message and sends it to the MTA. The MTA's responsibility is to transfer the mail across the network to the receiver's MTA. To send mails, a system must have a client MTA, and to receive mails, a system must have a server MTA.

SENDING EMAIL:

Mail is sent by a series of request and response messages between the client and the server. The message which is sent across consists of a header and a body. A null line is used to terminate the mail header and everything after the null line is considered as the body of the message, which is a sequence of ASCII characters. The message body contains the actual information read by the receipt.

RECEIVING EMAIL:

The user agent at the server-side checks the mailboxes at a particular time of intervals. If any information is received, it informs the user about the mail. When the user tries to read the mail it displays a list of emails with a short description of each mail in the mailbox. By selecting any of the mail users can view its contents on the terminal.

Essential SMTP commands in the order they may be used

Each SMTP command defines a particular function within the SMTP session, which consists of three steps:

- handshake – establishing a TCP connection
- email transfer – manipulations with the email
- termination – closing a TCP connection

Therefore, we decided to list the SMTP commands according to this flow.

➢ HELO/EHLO

The HELO command initiates the SMTP session conversation. The client greets the server and introduces itself. As a rule, HELO is attributed with an argument that specifies the domain name or IP address of the SMTP client.

Example: HELO client.net

EHLO is an alternative to HELO for servers that support the SMTP service extensions (ESMTP). If the server does not support ESMTP, it will reply with an error.

Example: EHLO client.net

In any case, HELO or EHLO is a MUST command for the SMTP client to commence a mail transfer.

➢ MAIL FROM

The MAIL FROM command initiates a mail transfer. As an argument, MAIL FROM includes a sender mailbox (reverse-path). For some types of reporting messages like non-delivery notifications, the reverse-path may be void. Optional parameters may also be specified.

Example: MAIL FROM "test@client.net"

➢ RCPT TO

The RCPT TO command specifies the recipient. As an argument, RCPT TO includes a destination mailbox (forward-path). In case of multiple recipients, RCPT TO will be used to specify each recipient separately.

Example: RCPT TO "user@recipient.net"

➢ DATA

With the DATA command, the client asks the server for permission to transfer the mail data. The response code 354 grants permission, and the client launches the delivery of the email contents line by line. This includes the date, from header, subject line, to header, attachments, and body text.

A final line containing a period (“.”) terminates the mail data transfer. The server responses to the final line.

Example:
DATA
354 (server response code)
Date: Wed, 30 July 2019 06:04:34
From: test@client.net
Subject: How SMTP works
To: user@recipient.net
Body text
.

➢ NOOP

The NOOP command is used only to check whether the server can respond. “250 OK” reply in response

Example: NOOP

➢ HELP

With the HELP command, the client requests a list of commands the server supports. HELP may be used with an argument (a specific command). If the server supports this, it will provide the information accordingly to this request.

Example: HELP

➢ VRFY and EXPN

VRFY is used to verify whether a mailbox in the argument exists on the local host. The server response includes the user’s mailbox and may include the user’s full name.

Example:
VRFY user2
250 Samantha Smith user2@client.net (server response)

EXPN is used to verify whether a mailing list in the argument exists on the local host. The positive response will specify the membership of the recipients.

Example:
EXPN mail-list
250-user1@client.net (server response)
250-user2@client.net (server response)
250-user3@client.net (server response)

The hyphen (-) between the numerical code and the user’s mailbox indicates that the response is continued on the next line.

VRFY and EXPN implement SMTP authentication. Also, they are useful to perform an internal audit of the server. On the other hand, these commands are considered a security risk. Spammers can use them to harvest valid email addresses from the server. Therefore, messaging systems either install corresponding protections or disable the commands.

➢ RSET

The RSET command resets the SMTP connection to the initial state. It erases all the buffers and state tables (both sender and recipient). RSET gets only the positive server response – 250. At the same time, the SMTP connection remains open and is ready for a new mail transaction.

Example: RSET

➢ QUIT

The QUIT command send the request to terminate the SMTP session. Once the server responses with 221, the client closes the SMTP connection. This command specifies that the receiver MUST send a “221 OK” reply and then closes the transmission channel.

Example: QUIT

Extended SMTP commands that some SMTP servers may support

➢ STARTTLS

The STARTTLS command is used to start a TLS handshake for a secure SMTP session. STARTTLS resets the SMTP protocol to the initial state. Once the response 220 is received from the server, the SMTP client should send HELO or EHLO to launch the session. In the case of a negative response (454), the client must decide whether to continue the SMTP session or not.

Example: STARTTLS

➢ AUTH

The AUTH command is used to authenticate the client to the server. For this, it uses an argument that specifies different levels of security and login methods: PLAIN, LOGIN, and CRAM-MD5. The session is considered authenticated once the server provided a positive response. For more on this, read the SMTP authentication blog post.

Example: AUTH CRAM-MD5

➢ ATRN

The ATRN command replaced the obsolete TURN command. It was used to reverse the connection between the local and external SMTP servers (sender and receiver). TURN lacked authentication and hence was deprecated. ATRN is devoid of this drawback. Besides, it is available for dynamically assigned IP addresses.

Example:
ATRN client.net,client.com
250 OK now reversing the connection (server response)

➢ BDAT

The BDAT command is used to submit mail contents. It can be an alternative to the DATA command. BDAT has two arguments. The first one defines the length of the data chunk in octets. The second one indicates that the data chunk is terminating. No need for a period to terminate mail transfer as it is in the DATA command. BDAT is widely used in Microsoft Exchange Server. At the same time, DATA is a must to support command for all servers.

Example:
BDAT 67 LAST
To: user@recipient.net
From: test@client.net
Subject: How SMTP works
250 Message OK, 67 octets received (server response)

➢ ETRN

The ETRN command is the request to start SMTP queue processing of a specified server host.

Example:
ETRN client.com
250 OK, queuing for client.com started (server response)

Private-use SMTP commands

The client and the server may use private-use SMTP commands through a bilateral agreement. These are proprietary service extensions and should start with “X”. They must not be registered or standardized. Here are some of them:

- **XADR** – releases status of the channel an address matches (how an address is routed internally)
- **XCIR** – releases status of the circuit checking facility.

- **XSTA** – releases status of the number of messages in channel queues
- **XGEN** – releases status of whether a compiled configuration and character set are in use.

Obsolete SMTP commands

On the web, there are many outdated sources where you can encounter obsolete SMTP commands. Not to waste your time for invalid options, here is a list of the commands you can't use anymore:

- SEND
- SOML
- SAML
- RELAY
- TLS
- TURN

SMTP response codes

The SMTP server responses to the client using a three-digit code. Each digit has a special significance:

- First (2 to 5) – denotes whether the request is accepted, incomplete, or declined
- Second (0 to 5) – denotes the type of error occurred (syntax, information, connections, mail system, or unspecified (two options)).
- Third (0 to 5) – provides finest description (together with textual explanation)

The numerical code is followed by a text meant for a human user to get the point. Different servers can use a modified textual description of the response, while the numerical code is permanent. So, here is what your SMTP server can reply with:

Code	What it means
101	Server connection error (wrong server name or connection port)
211	System status (response to HELP)
214	Help message (response to HELP)
220	The server is ready (response to the client's attempt to establish a TCP connection)
221	The server closes the transmission channel
235	Authentication successful (response to AUTH)
250	The requested command is completed. As a rule, the code is followed by OK

251	User is not local, but the server will forward the message to <forward-path>
252	The server cannot verify the user (response to VRFY). The message will be accepted and attempted for delivery
334	Response to the AUTH command when the requested security mechanism is accepted
354	The server confirms mail content transfer (response to DATA). After that, the client starts sending the mail. Terminated with a period (“ . ”)
421	The server is not unavailable because it closes the transmission channel
422	The recipient’s mailbox has exceeded its storage limit
431	File overload (too many messages sent to a particular domain)
441	No response from the recipient’s server
442	Connection dropped
446	Internal loop has occurred
450	Mailbox unavailable (busy or temporarily blocked). Requested action aborted
451	The server aborted the command due to a local error
452	The server aborted the command due to insufficient system storage
454	TLS not available due to a temporary reason (response to STARTTLS)
455	Parameters cannot be accommodated
471	Mail server error due to the local spam filter
500	Syntax error (also a command line may be too long). The server cannot recognize the command
501	Syntax error in parameters or arguments
502	The server has not implemented the command
503	Improper sequence of commands
504	The server has not implemented a command parameter
510	Invalid email address
512	A DNS error (recheck the address of your recipients)
523	The total size of your mailing exceeds the recipient server limits
530	Authentication problem that mostly requires the STARTTLS command to run
535	Authentication failed

538	Encryption required for a requested authentication mechanism
541	Message rejected by spam filter
550	Mailbox is unavailable. Server aborted the command because the mailbox was not found or for policy reasons. Alternatively: Authentication is required for relay
551	User not local. The <forward-path> will be specified
552	The server aborted the command because the mailbox is full
553	Syntactically incorrect mail address
554	The transaction failed due to an unknown error or No SMTP service here as a response to the client's attempts to establish a connection
555	Parameters not recognized/ not implemented (response to `MAIL FROM` or `RCPT TO`)

Command-response

As you may have noticed above, some codes are command-specific. Actually, only three of them, 500, 501, and 421 can be a response to any SMTP command. Others can be categorized as positive and negative (code 354 can be considered as an intermediate response). Let's see which commands they can refer to.

Command	Positive response	Negative response
SMTP handshake (establishing a connection)	220	554
`STARTTLS`	220	454
`EHLO` or `HELO`	250	502 (response to EHLO for old-time servers)

		504 550
AUTH	235 334	530 535 538
MAIL FROM	250	451 452 455 503 550 552 553 555
RCPT TO	250 251	450 451 452 455 503 550 551 552 553 555
DATA	250 354 (intermediate response)	450 451 452 503 550 (rejection for policy reasons)

		552 554
RSET	250	–
VRFY	250 251 252	502 504 550 551 553
EXPN	250 252	502 504 550
HELP	211 214	502 504
NOOP	250	–
QUIT	221	–

This is the list of standard response codes. It should be also mentioned that some SMTP servers can generate other three-digit codes. In this case, the SMTP client will have to interpret the first digit that must be in a range from 2 to 5 inclusive. It denotes the essence of the response (successful or not).

LAB NO.12

NAME : ABDULLAH ZUNORAIN

REG NO : 19JZELE0338

SUBJECT : COMPUTER COMMUNICATION NETWORK

SUBMITTED TO : DR. UZAIR GILLANI

SECTION : A

TITLE: ANALYSIS OF UDP USING WIRESHARK

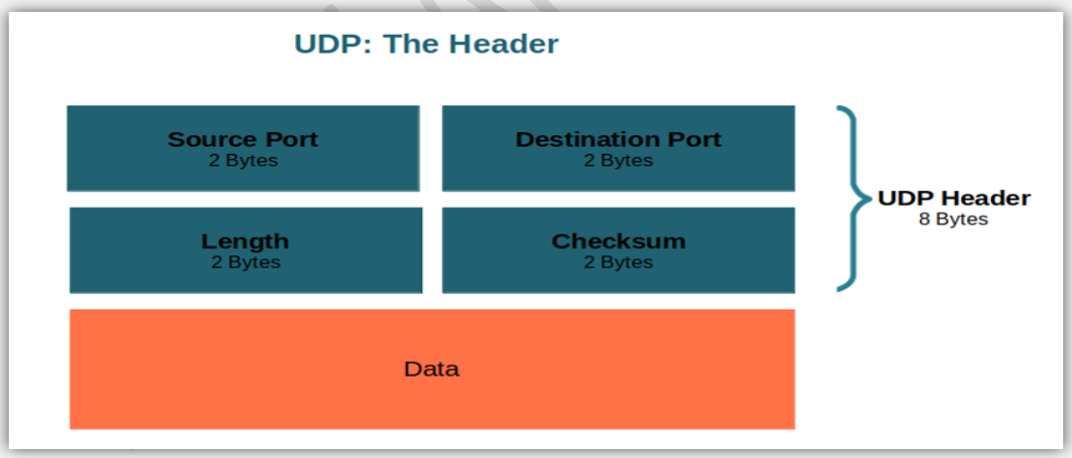
Objectives:

- Capturing UDP packets traffic on wireshark
- To understand how UDP protocol works

Introduction:

User Datagram Protocol (UDP) is a Transport Layer protocol. UDP is a part of the Internet Protocol suite, referred to as UDP/IP suite. Unlike TCP, it is an unreliable and connectionless protocol. So, there is no need to establish a connection prior to data transfer. Though Transmission Control Protocol (TCP) is the dominant transport layer protocol used with most of the Internet services; provides assured delivery, reliability, and much more but all these services cost us additional overhead and latency. Here, UDP comes into the picture. For real-time services like computer gaming, voice or video communication, live conferences; we need UDP. Since high performance is needed, UDP permits packets to be dropped instead of processing delayed packets. There is no error checking in UDP, so it also saves bandwidth.So User Datagram Protocol (UDP) is more efficient in terms of both latency and bandwidth.

UDP header is an **8-bytes** fixed and simple header, while for TCP it may vary from 20 bytes to 60 bytes. The first 8 Bytes contains all necessary header information and the remaining part consist of data. UDP port number fields are each 16 bits long, therefore the range for port numbers is defined from **0 to 65535**; port number 0 is reserved. Port numbers help to distinguish different user requests or processes.



Source Port: Source Port is a 2 Byte long field used to identify the port number of the source.

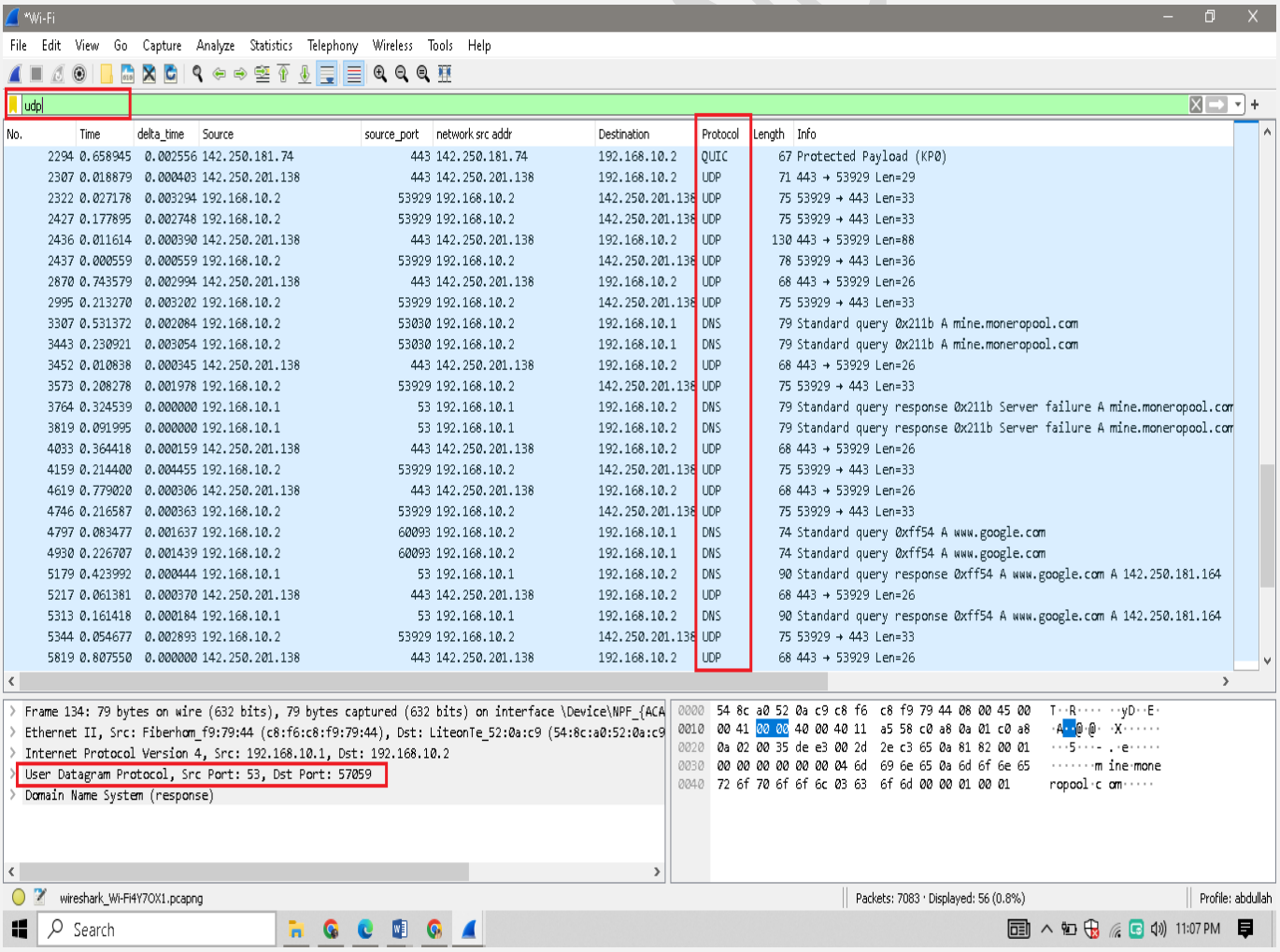
Destination Port: It is a 2 Byte long field, used to identify the port of the destined packet.

Length: Length is the length of UDP including the header and the data. It is a 16-bits field.

Checksum: Checksum is 2 Bytes long field. It is the 16-bit one's complement of the one's complement sum of the UDP header, the pseudo-header of information from the IP header, and the data, padded with zero octets at the end (if necessary) to make a multiple of two octets.

Procedure:

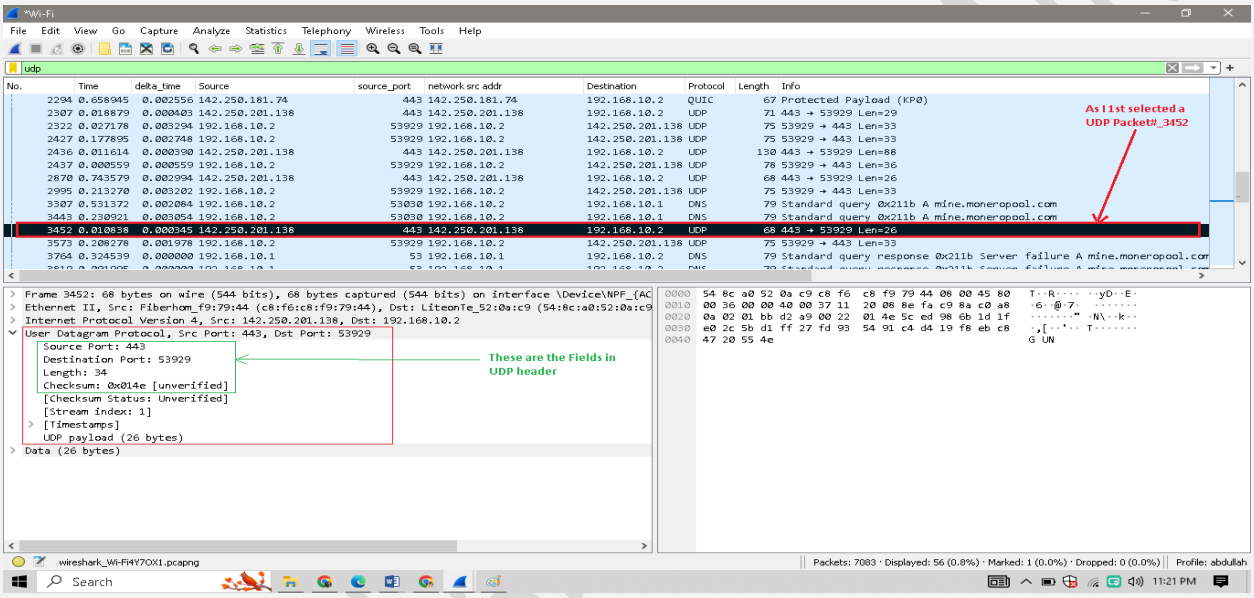
- First we have to open the wireshark and select Wifi interaface.
- After that we have to start capturing the packets.
- We have to go to our browser and type there any URL so that packets is captured.
- We have to stop the capturing process.
- For filter out only UDP packets we have to search UDP in current filter tab.
- Now we have to analyse it as shown below;



LAB TASKS:

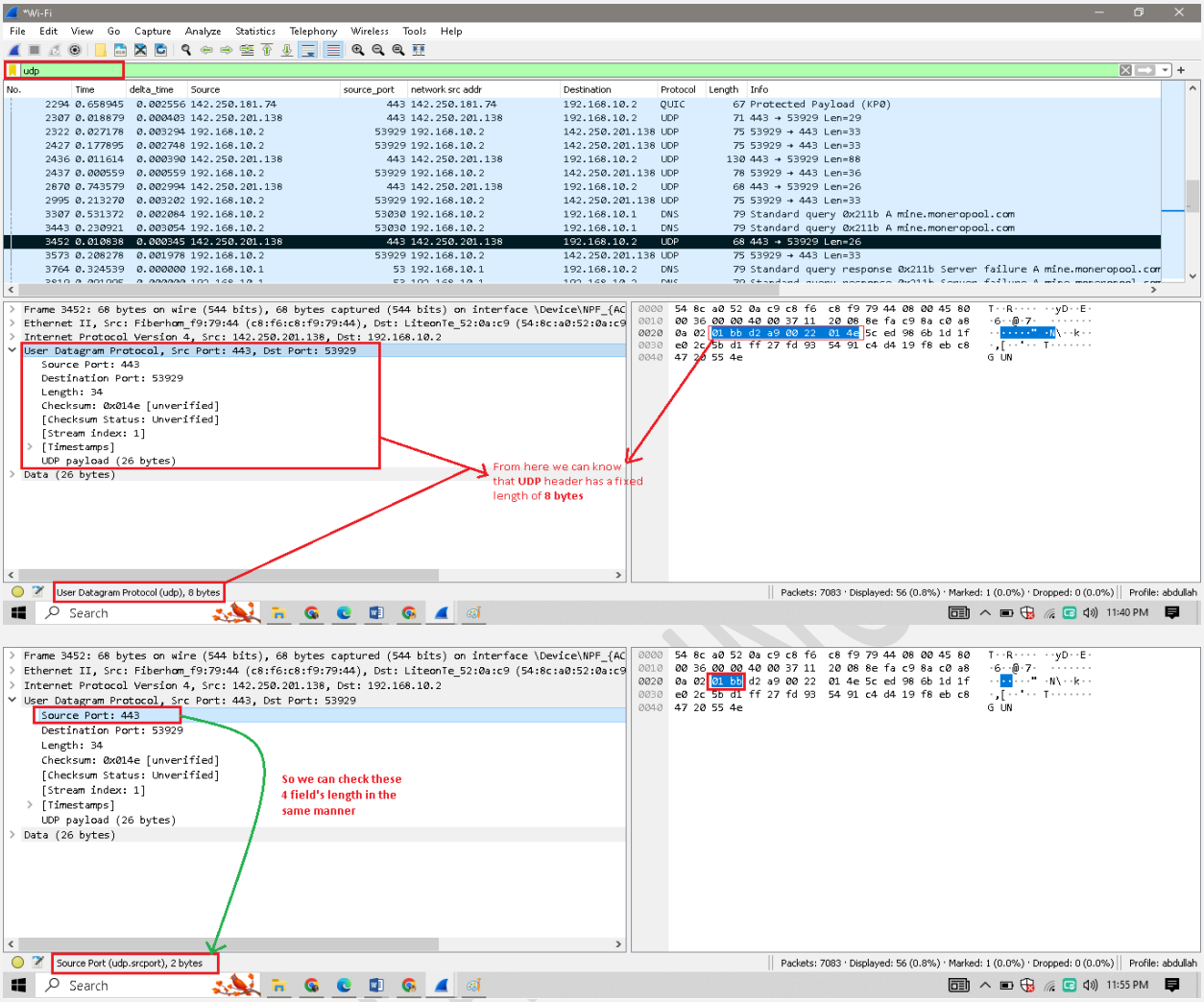
1. **Select *one* UDP packet from your trace. From this packet, determine how many fields there are in the UDP header. (You shouldn't look in the textbook! Answer these questions directly from what you observe in the packet trace.) Name these fields.**

ANS: The **UDP header** only contains **4** fields: the **source port**, **destination port**, **length**, and **checksum**, And the further which is enclosed in the square brackets are **optional fields.**



2. **By consulting the displayed information in Wireshark's packet content field for this packet, determine the length (in bytes) of each of the UDP header fields.**

ANS: The **UDP header** has a *fixed length of **8-bytes***. Each of these **4-header fields are of 2 bytes** long as shown below;



3. The value in the Length field is the length of what? (You can consult the text for this answer). Verify your claim with your captured UDP packet.

ANS: Length is the length of UDP including the **header** and the **data**.

It is a **16-bits long field**. There is **16 bits** which is used to represent the size but the **actual size is 2^16(0-65535) bytes.**

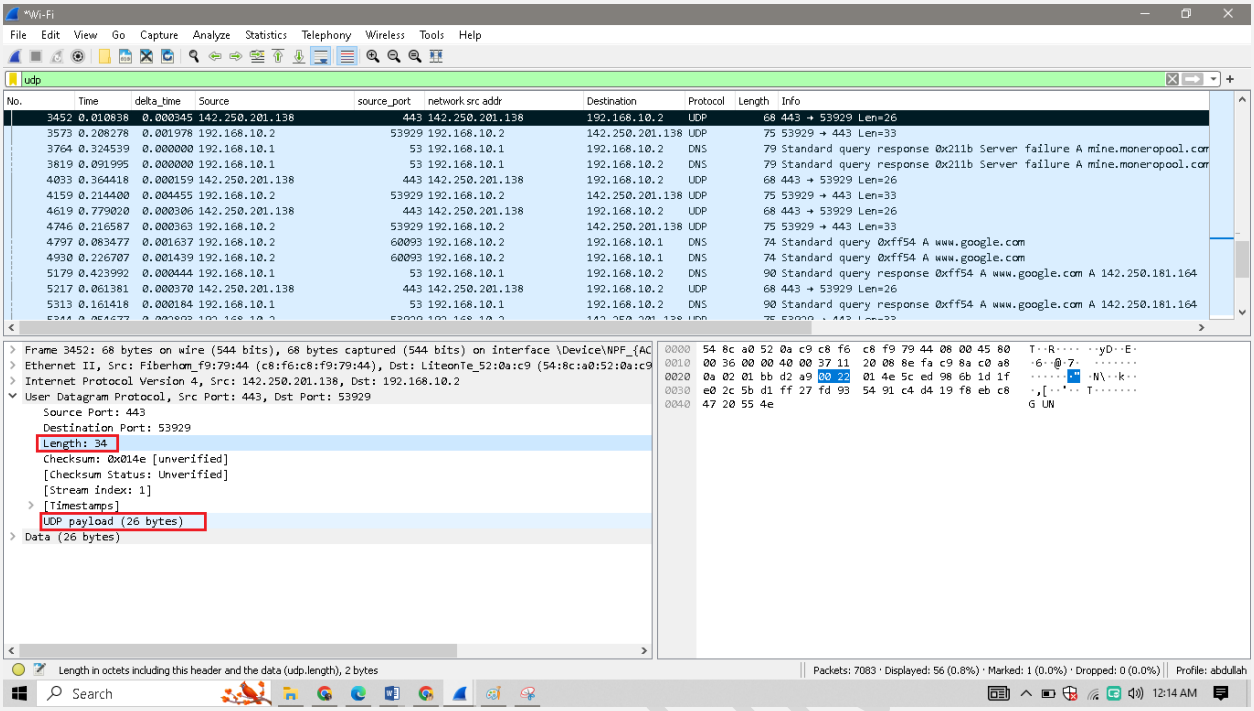
Header = 8 bytes

Payload(actual data)= 65527 bytes

The length field specifies the number of bytes in the UDP segment (header plus data).

An explicit length value is needed since the size of the data field may differ from one UDP segment to the next.

The length of UDP payload for selected packet is **26 bytes; *34 bytes - 8 bytes = 33bytes.***



4. **What is the maximum number of bytes that can be included in a UDP payload? (Hint: the answer to this question can be determined by your answer to 2. above)**

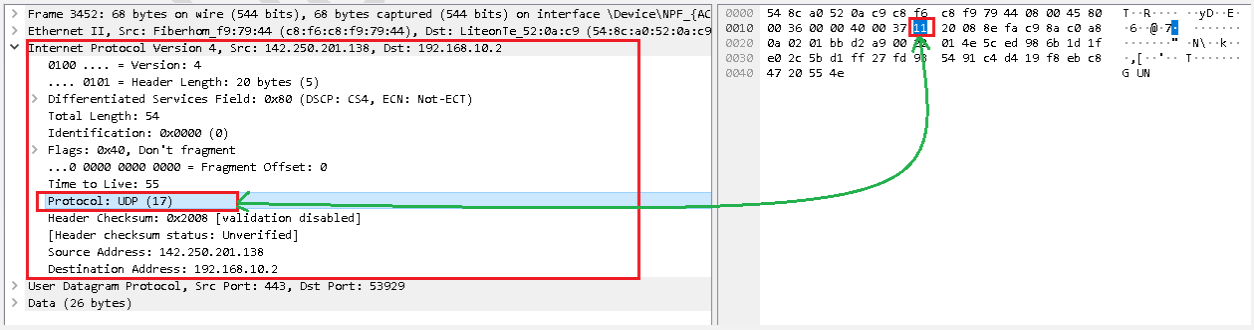
ANS: The maximum number of bytes that can be in the payload is **2^16- the bytes** already being used by the **header field (8 bytes)**. **Maximum payload** is 65535-8= **65527 bytes**.

5. **What is the largest possible source port number? (Hint: see the hint in 4.)**

ANS: The largest source port number is 2 16 -1=**65535**

6. **What is the protocol number for UDP? Give your answer in both hexadecimal and decimal notation. To answer this question, you'll need to look into the Protocol field of the IP datagram containing this UDP segment (see Figure 4.13 in the text, and the discussion of IP header fields).**

ANS: The IP protocol number for UDP is **0x11** in hexadecimal, which is **17** in *decimal.*



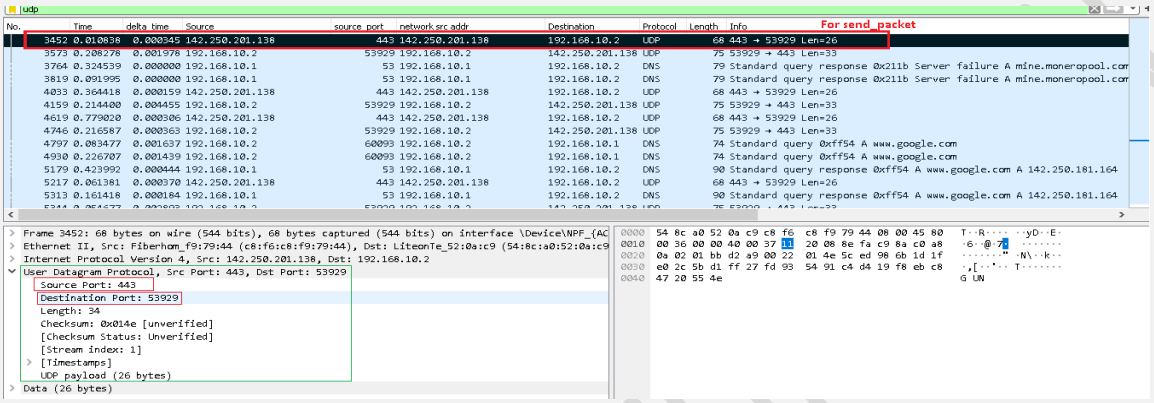
7. **Examine a pair of UDP packets in which your host sends the first UDP packet and the second UDP packet is a reply to this first UDP packet. (Hint: for a second packet to be sent in response to a first packet, the sender of the first packet should be the destination of the second packet). Describe the relationship between the port numbers in the two packets.**

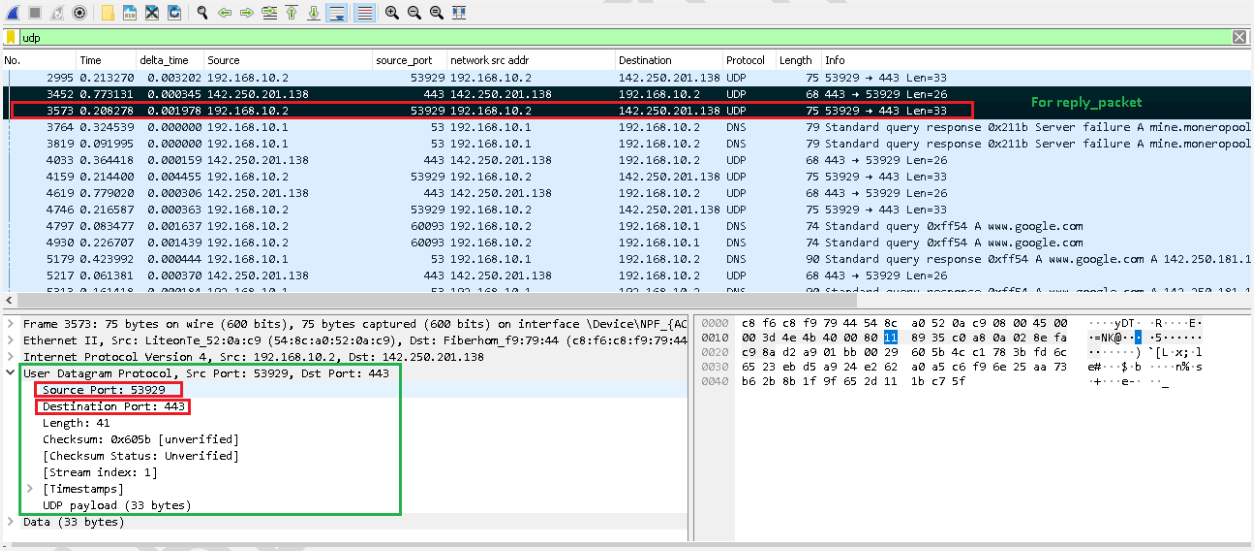
ANS: The source port of the UDP packet sent by the host is the same as the destination port of the reply packet, and conversely the destination port of the UDP packet sent by the host is the same as the source port of the reply packet.

Send packet:

Source port:

Destination port:





Reply packet:

Source port:

Destination port:

LAB NO.13

NAME : ABDULLAH ZUNORAIN

REG NO: 19JZELE0338

SUBJECT: COMPUTER COMMUNICATION NETWORKING

SUBMITTED TO: SYED UZAIR GILLANI

SECTION: A

DEPT: ELECTRICAL COMMUNICATION

TITLE: ILLUSTRATION OF THE WORKING OF A SWITCHED BASED LOCAL AREA NETWORK (LAN) USING PACKET TRACER.

Objectives:

- To know that how to make a *Local Area Network* on the base of switch in Packet tracer.
- To Identify tools used for making a LAN based on switch.

Software Used:

- Cisco Packet Tracer

Devices Used:

- 2960 network switches.
- Three PC's
- Copper Straight-Through cable

About switches(Working of switches) :

Switches are key building blocks for any network. They connect multiple devices, such as computers, wireless access points, printers, and servers; on the same network within a building or campus. A switch enables connected devices to share information and talk to each other.

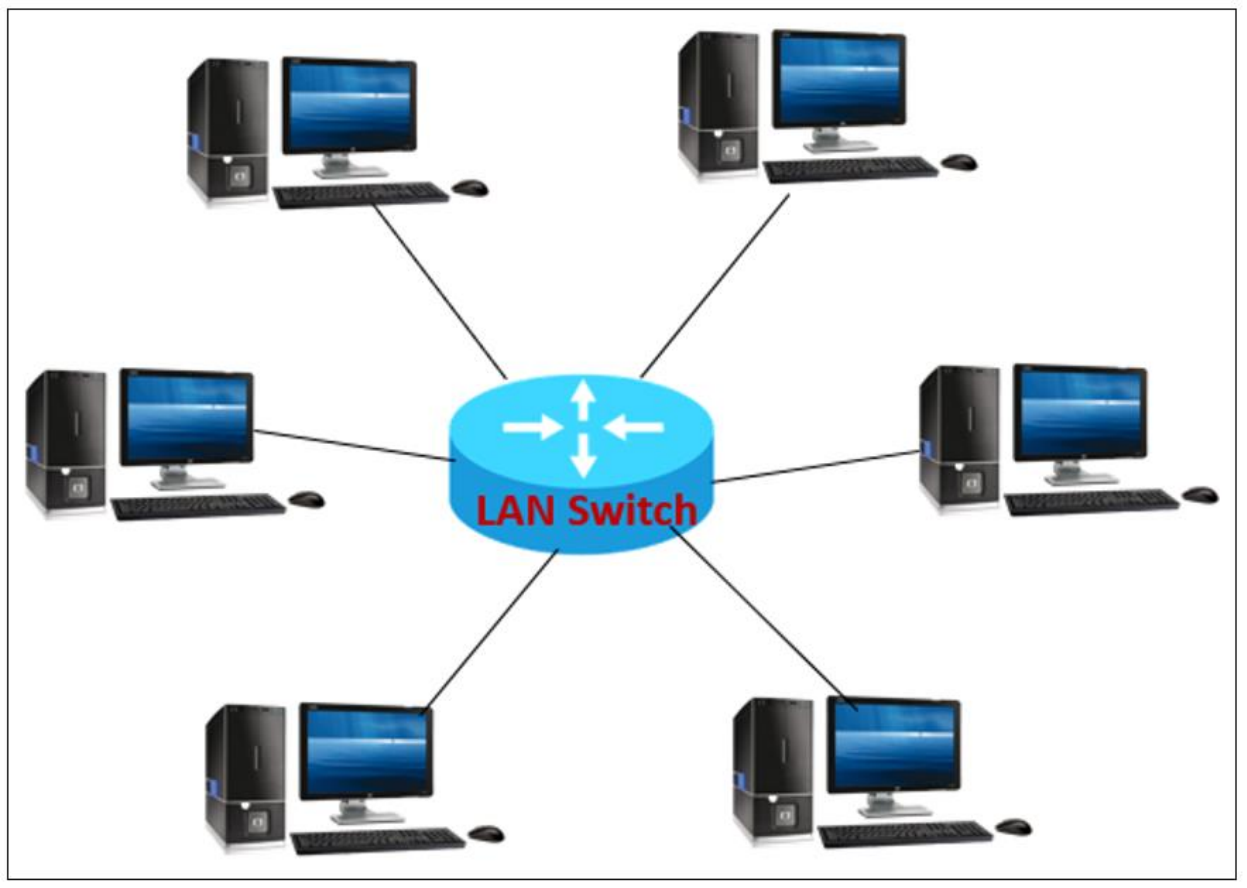
And also, it allows to directly communicate with multiple users. The creation of switching provides free collision, high-speed in networking and makes the system parallel, quick, and peer-to-peer connection is available between two devices.

1. ***Unmanaged switches:***

An unmanaged network switch is designed so that you can simply plug them in and they work, no configuration required. Unmanaged switches are typically for basic connectivity. You'll often see them used in home networks or wherever a few more ports are needed, such as at your desk, in a lab, or in a conference room.

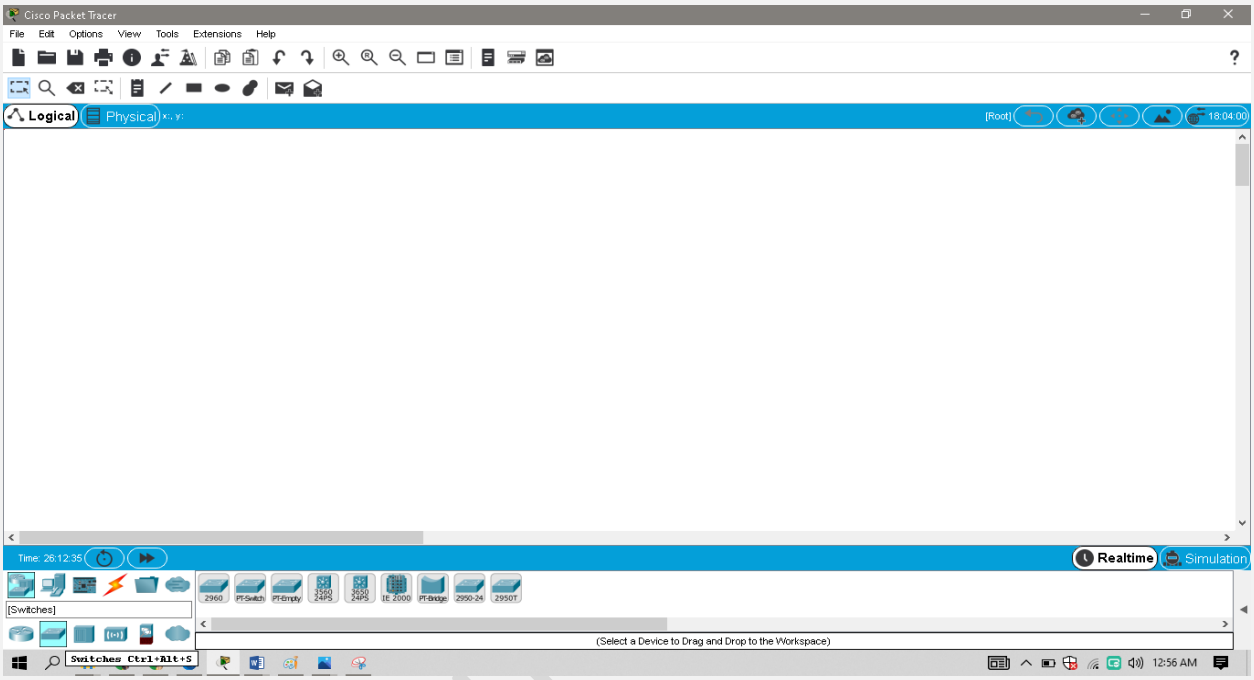
2. ***Managed switches:***

Managed switches give you greater security and more features and flexibility because you can configure them to custom-fit your network. With this greater control, you can better protect your network and improve the quality of service for those who access the network.

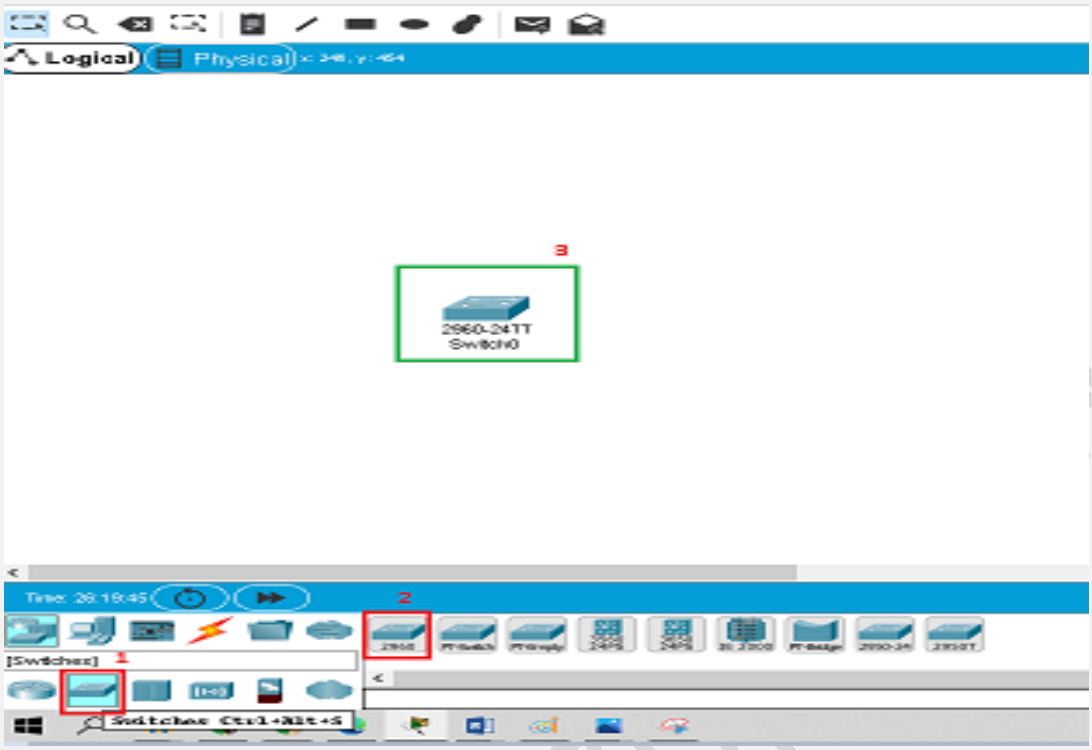


PROCEDURE FOR MAKING SWITCHED BASED LAN IN PACKET TRACER:

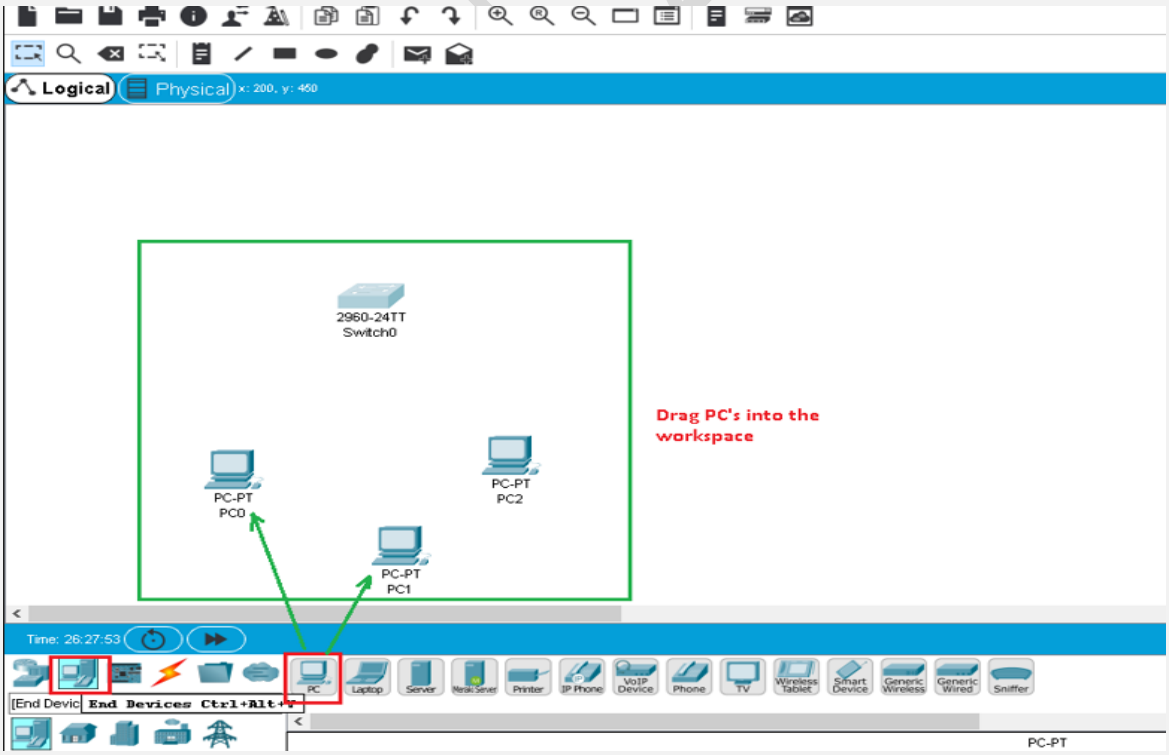
First of all we have to open Packet tracer software.



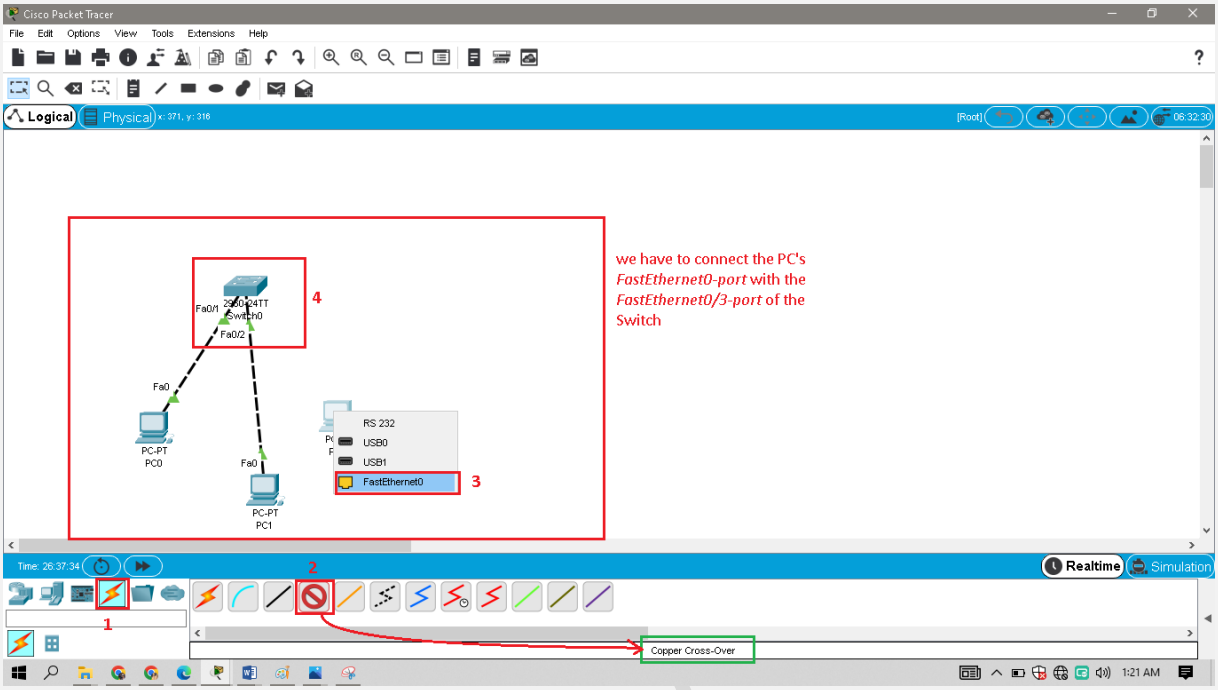
Then we have to select the ***Switches*** from **Device-type selection** and then select **2960-switch** and will drag it to the *work space as shown below;*



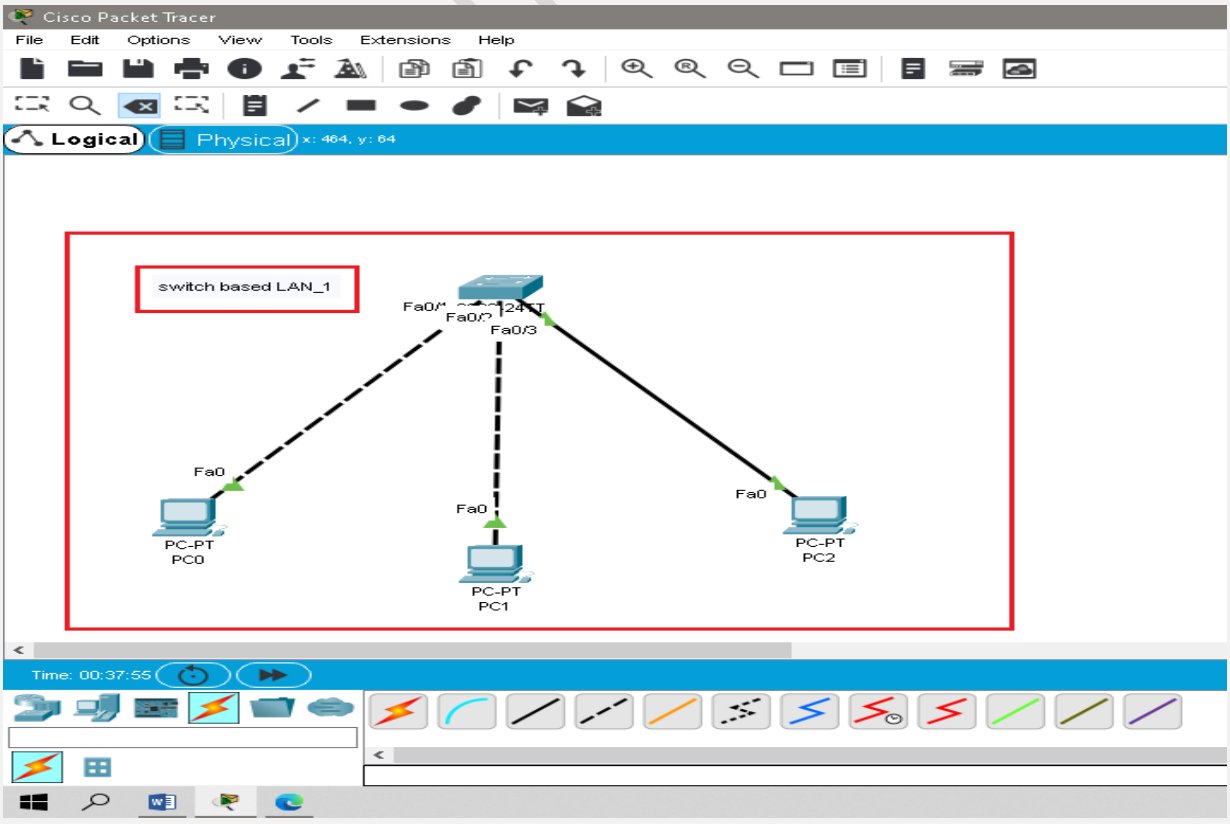
Now we have to go to **End-devices**, from there we have to select the **3-PC's** and drag these PC's to workspace as shown below;



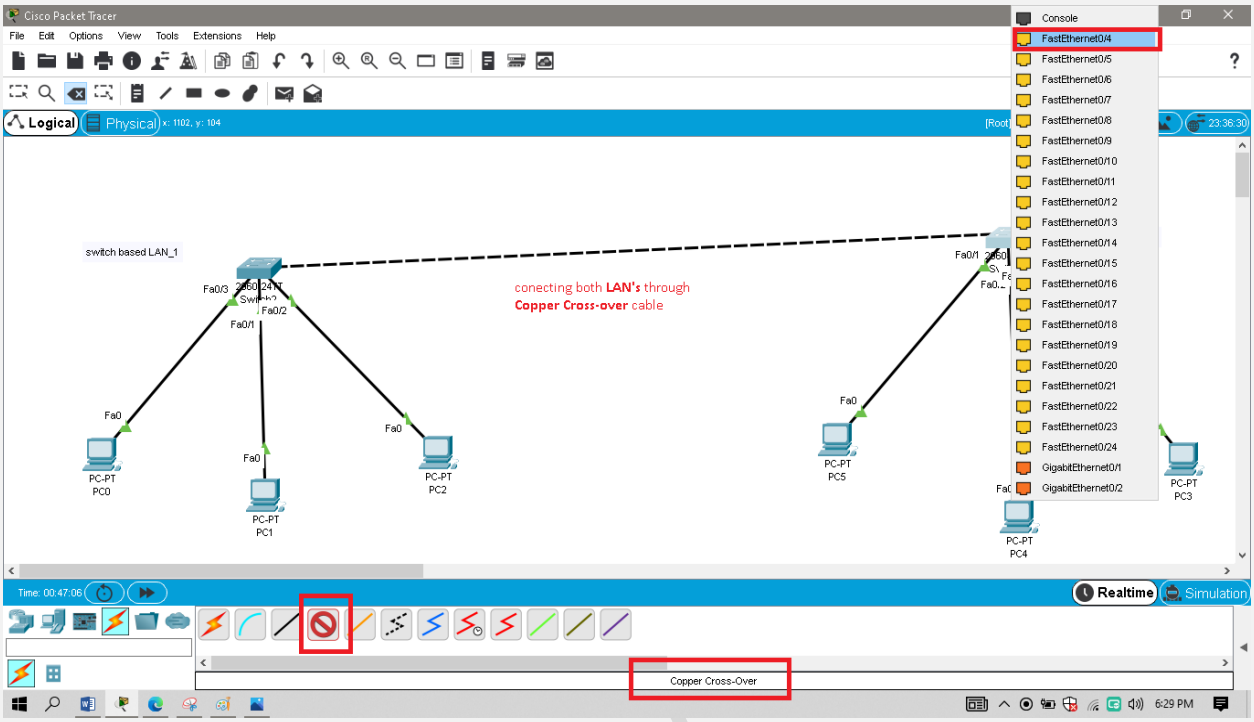
Now for connecting these PC's with the switch we have to go to the **Connection** and then in the connection we have to select the **copper cross-over cable** for different devices *or* we can choose **automatically-choose-connection-type** as shown below;



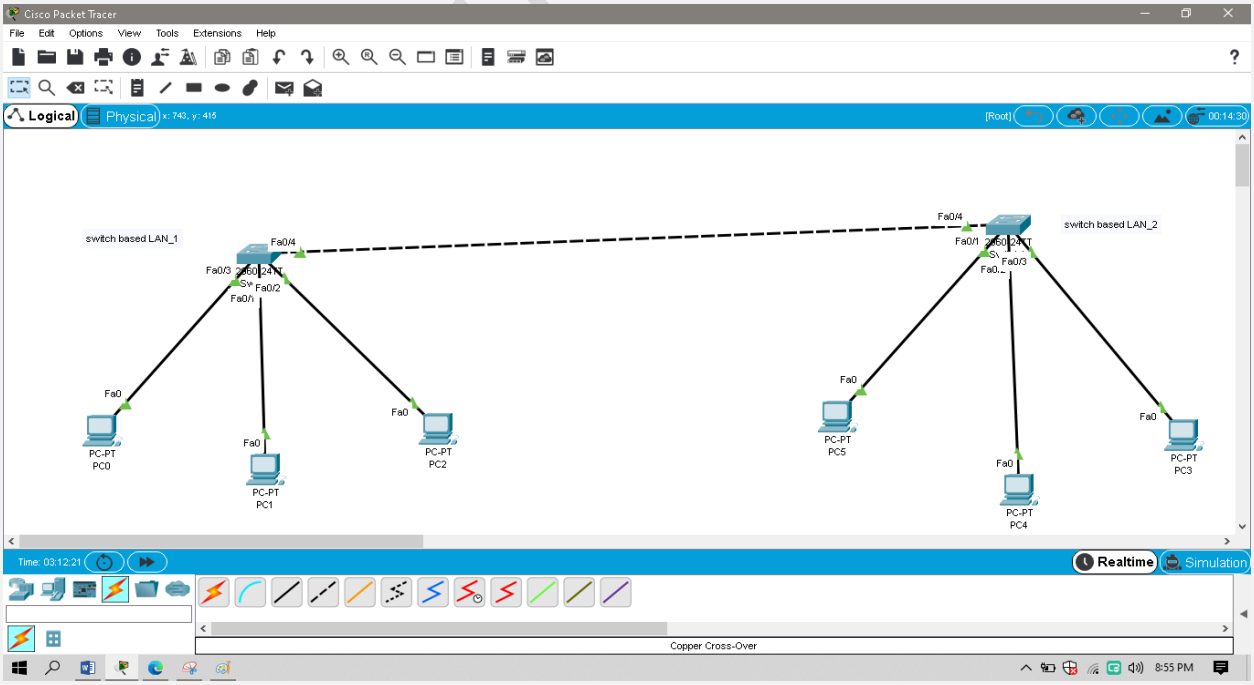
So now we have successfully created a **switch based LAN_1** as shown below;



Now we have to create another **LAN_2** and *connects* both LAN's through **Copper Cross over cable** as shown below;



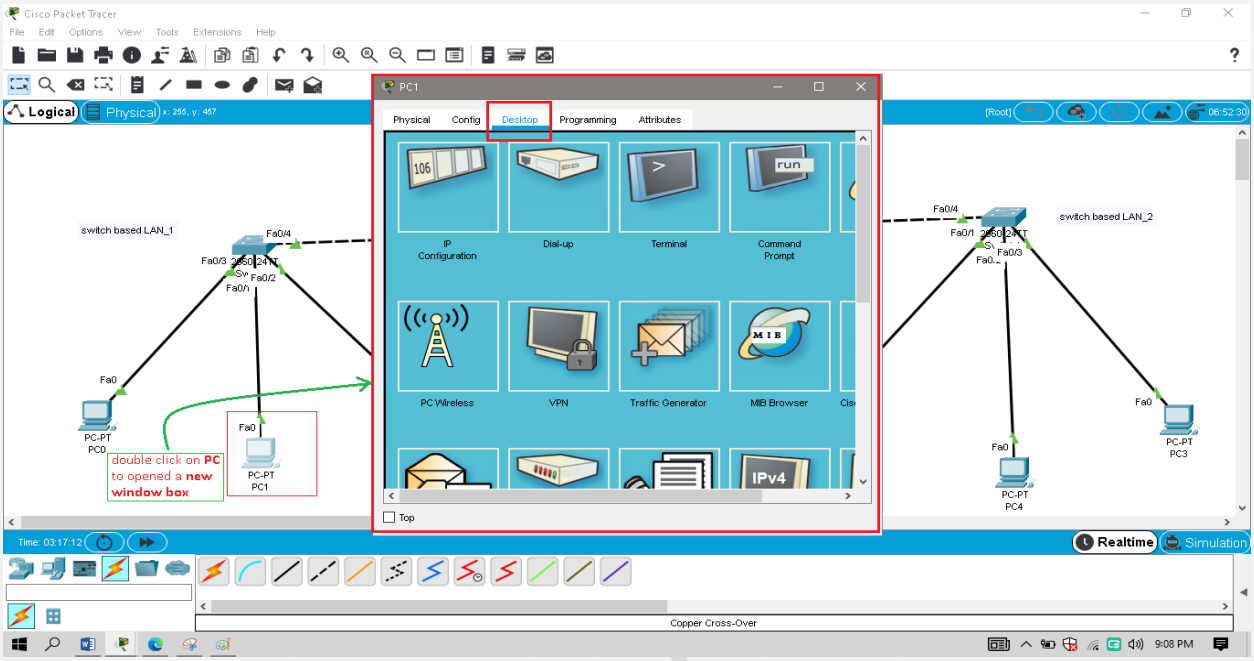
So we had successfully **configured** both switch based LAN's as shown below;



ASSIGNING IP-ADDRESSES TO ALL THE PC'S IN BOTH LAN_1 & LAN_2

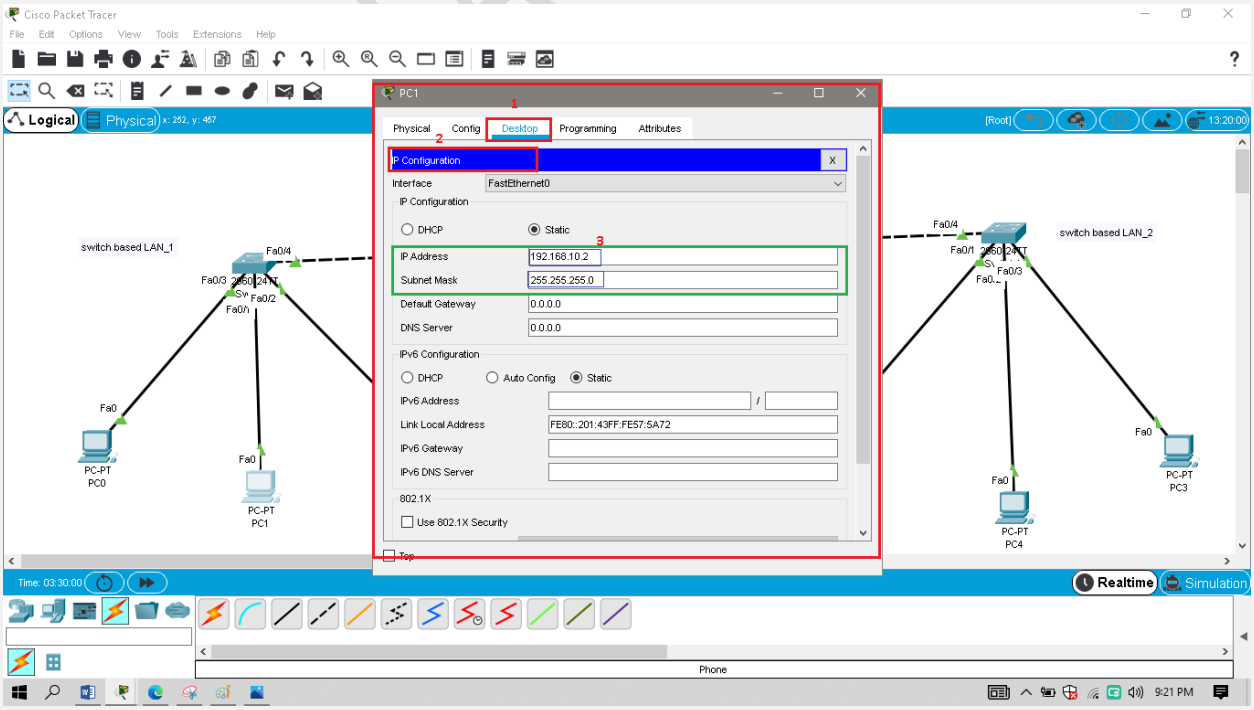
Now we have to assign Ip-addresses to all of PC's in both LAN's .

For doing so we have to 1st ***double click*** on each of one PC , so a ***new Box*** will be appears as shown below;



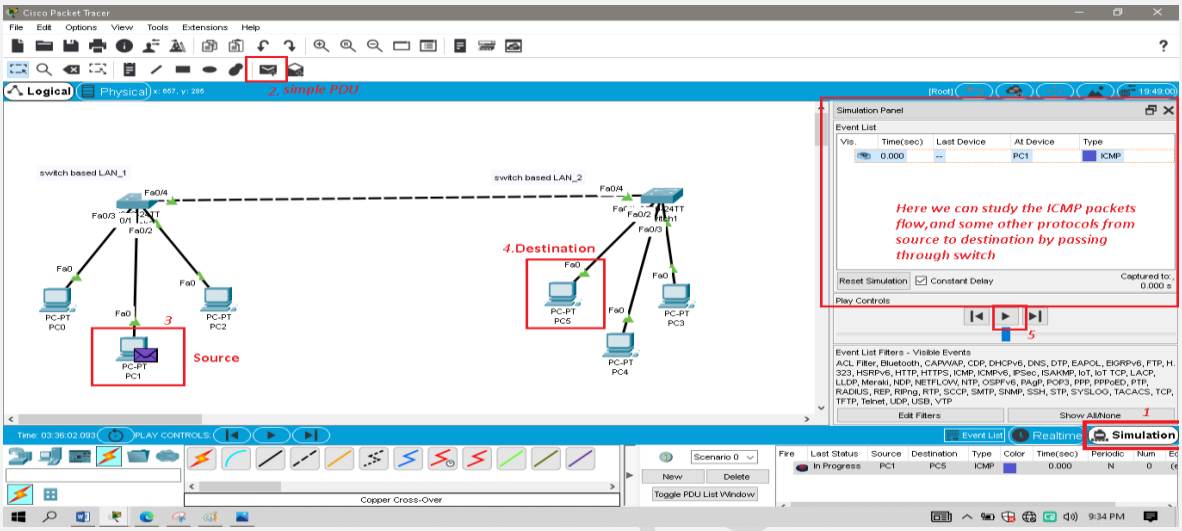
Then we have to go to the **Ip-configuration** where we have to assign **ip-addresses** and **subnet mask** to each of the PC's ;

NOTE: IN THIS CASE WHERE WE ARE CONNECTING BOTH LAN's , SO WE HAVE TO KEPT SUBNET-MASK SAME FOR PROPER TRANSFERRING OF SIMPLE PDU.

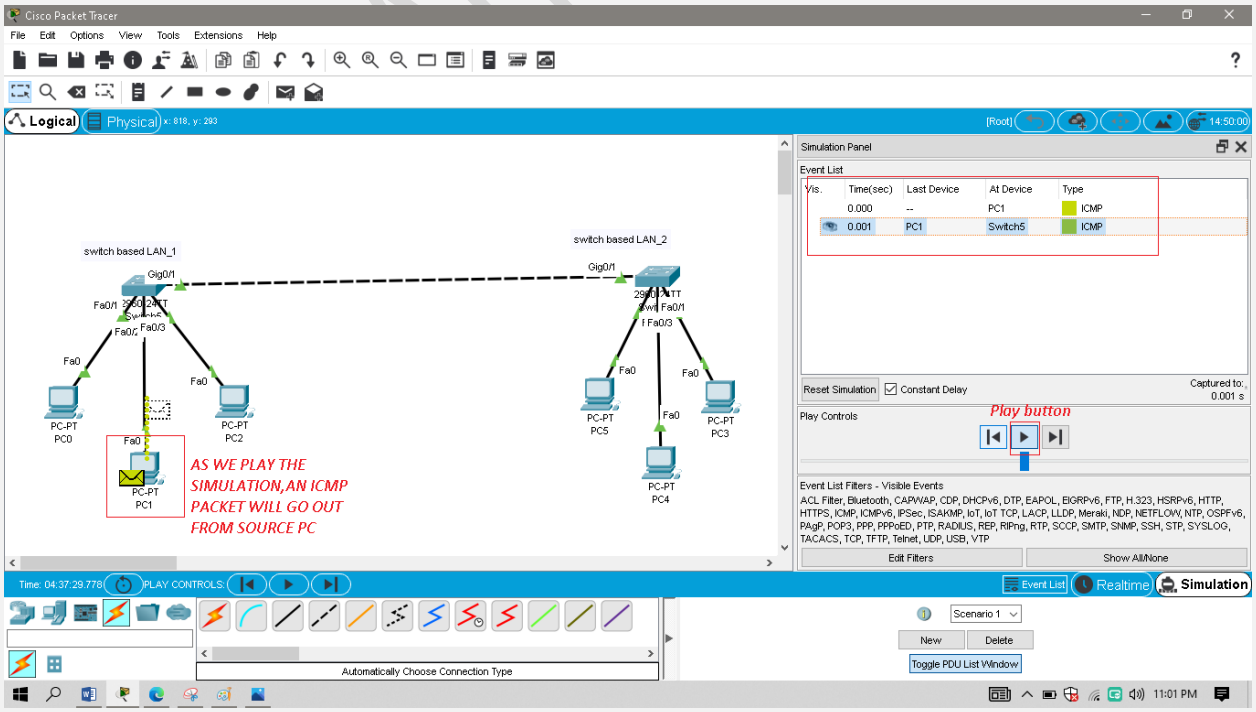


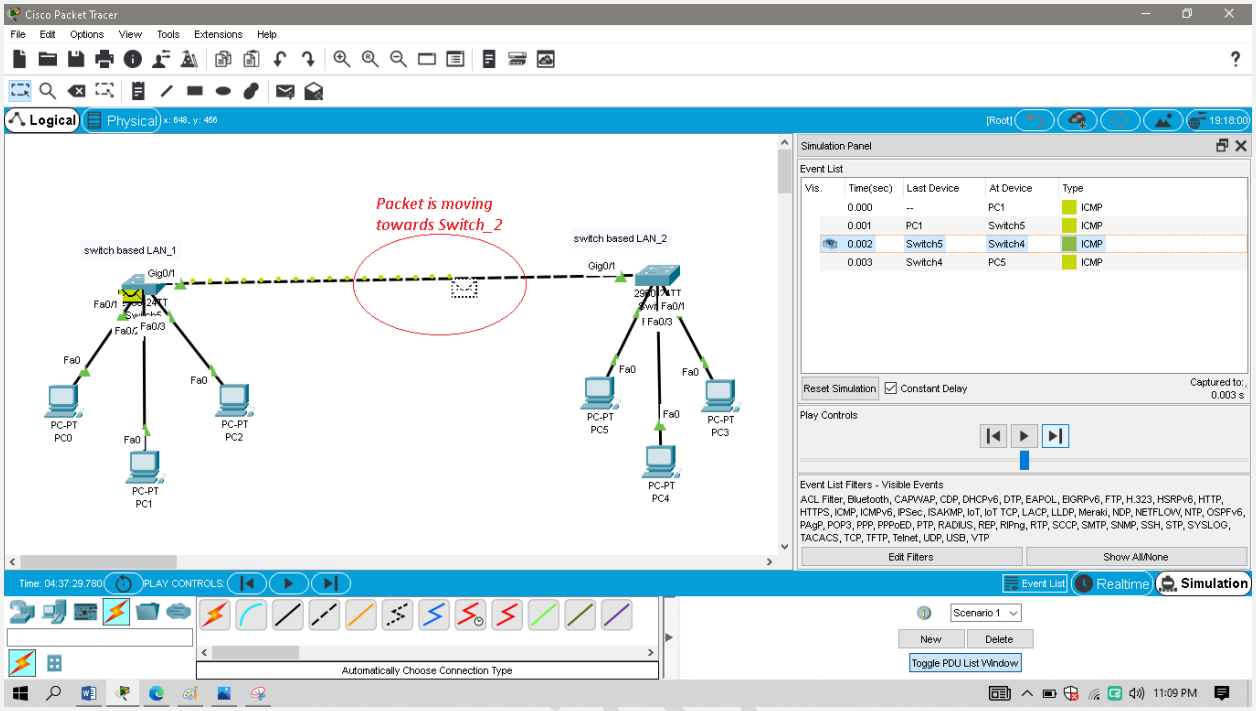
SIMULATION MODE:

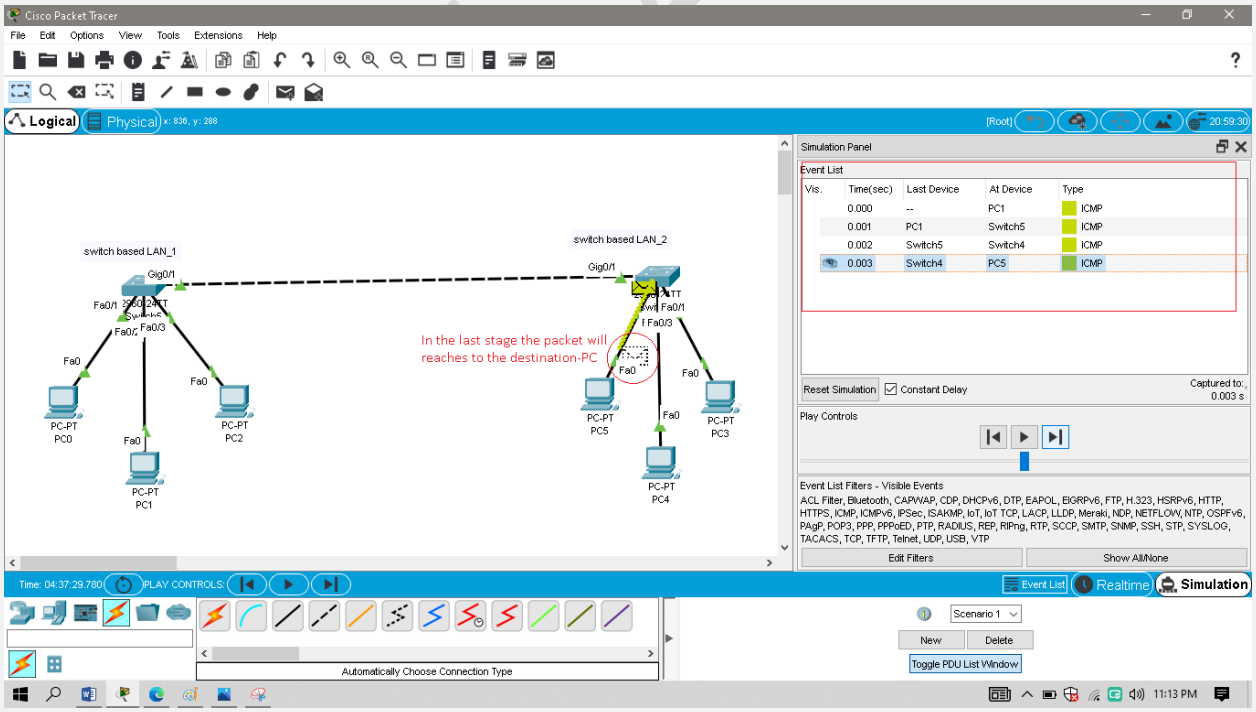
Now to study in the Simulation Mode we have to select the **SIMULATION MODE** and the select ***simple PDU*** and drag into the **workspace** to paste on any of the PC to which we want as **Source-PC** and then we have to paste on other PC to which we want as **Destination-PC** as shown in the below screenshot;

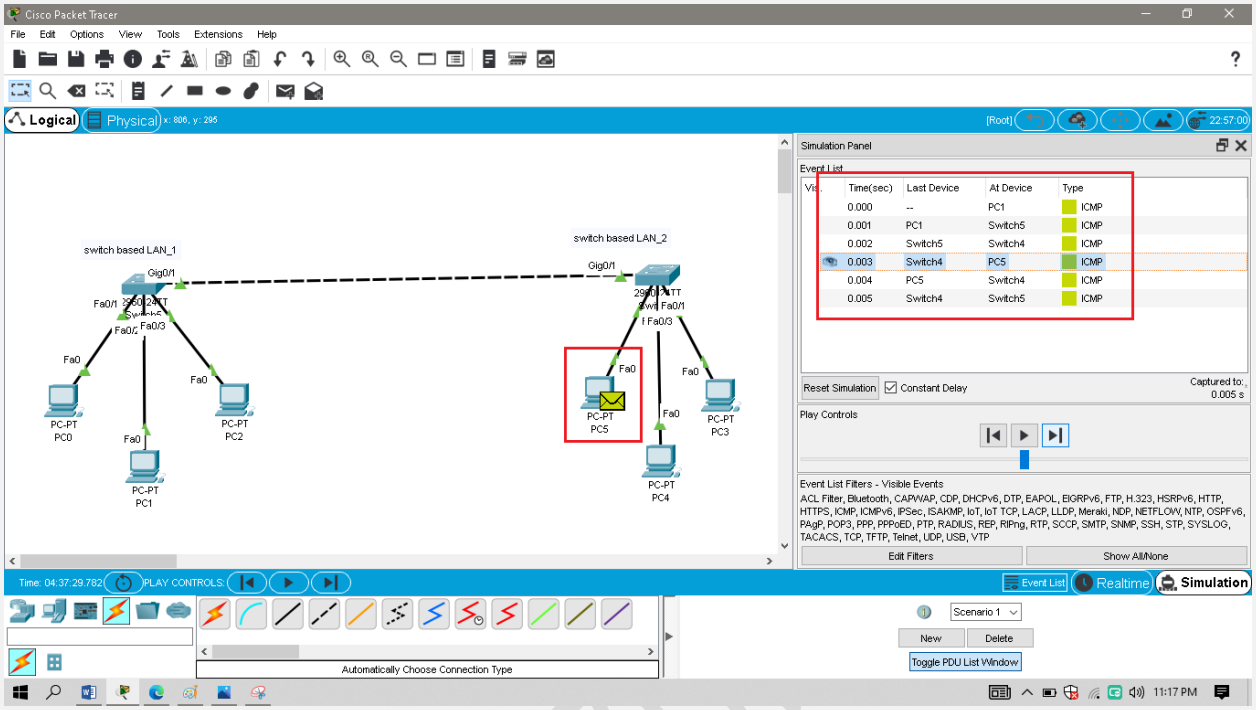


Now when we push the **Play button** so ***Simple PDU packet*** will be sends from **source PC** and will be go out through ***switch_1*** and will be enters into the ***switch_2*** to reaches to its **destination PC** as shown below;

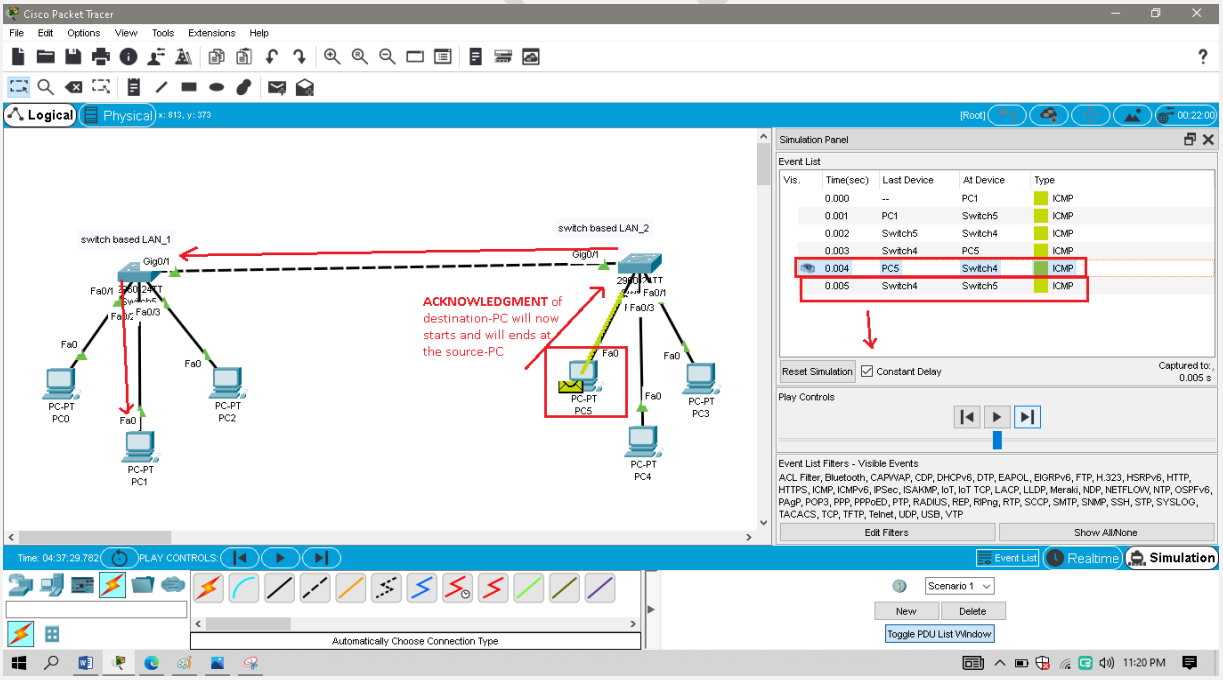



Cisco Packet Tracer
File Edit Options View Tools Extensions Help
Logical
Physical
Packet is moving towards Switch_2
switch based LAN_1
switch based LAN_2
Gig0/1
Fa0/1
Fa0/2
Fa0/3
Fa0
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type
0.000 -- PC1 ICMP
0.001 PC1 Switch5 ICMP
0.002 Switch5 Switch4 ICMP
0.003 Switch4 PC5 ICMP
Reset Simulation
Constant Delay
Captured to: 0.003 s
Play Controls
Event List Filters - Visible Events
ACL Filter, Bluetooth, CAPWAP, CDP, DHCPv6, DTP, EAPOL, EIGRPv6, FTP, H.323, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, Meraki, NDP, NETFLOW, NTP, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP
Edit Filters
Show All/None
Time: 04:37:29.780
PLAY CONTROLS:
Event List
Realtime
Simulation
Scenario 1
New
Delete
Toggle PDU List Window
Automatically Choose Connection Type
11:09 PM


Cisco Packet Tracer
File Edit Options View Tools Extensions Help
Logical
Physical
In the last stage the packet will reaches to the destination-PC
switch based LAN_1
switch based LAN_2
Gig0/1
Fa0/1
Fa0/2
Fa0/3
Fa0
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
PC-PT PC5
Simulation Panel
Event List
Vis. Time(sec) Last Device At Device Type
0.000 -- PC1 ICMP
0.001 PC1 Switch5 ICMP
0.002 Switch5 Switch4 ICMP
0.003 Switch4 PC5 ICMP
Reset Simulation
Constant Delay
Captured to: 0.003 s
Play Controls
Event List Filters - Visible Events
ACL Filter, Bluetooth, CAPWAP, CDP, DHCPv6, DTP, EAPOL, EIGRPv6, FTP, H.323, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, Meraki, NDP, NETFLOW, NTP, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP
Edit Filters
Show All/None
Time: 04:37:29.780
PLAY CONTROLS:
Event List
Realtime
Simulation
Scenario 1
New
Delete
Toggle PDU List Window
Automatically Choose Connection Type
11:13 PM

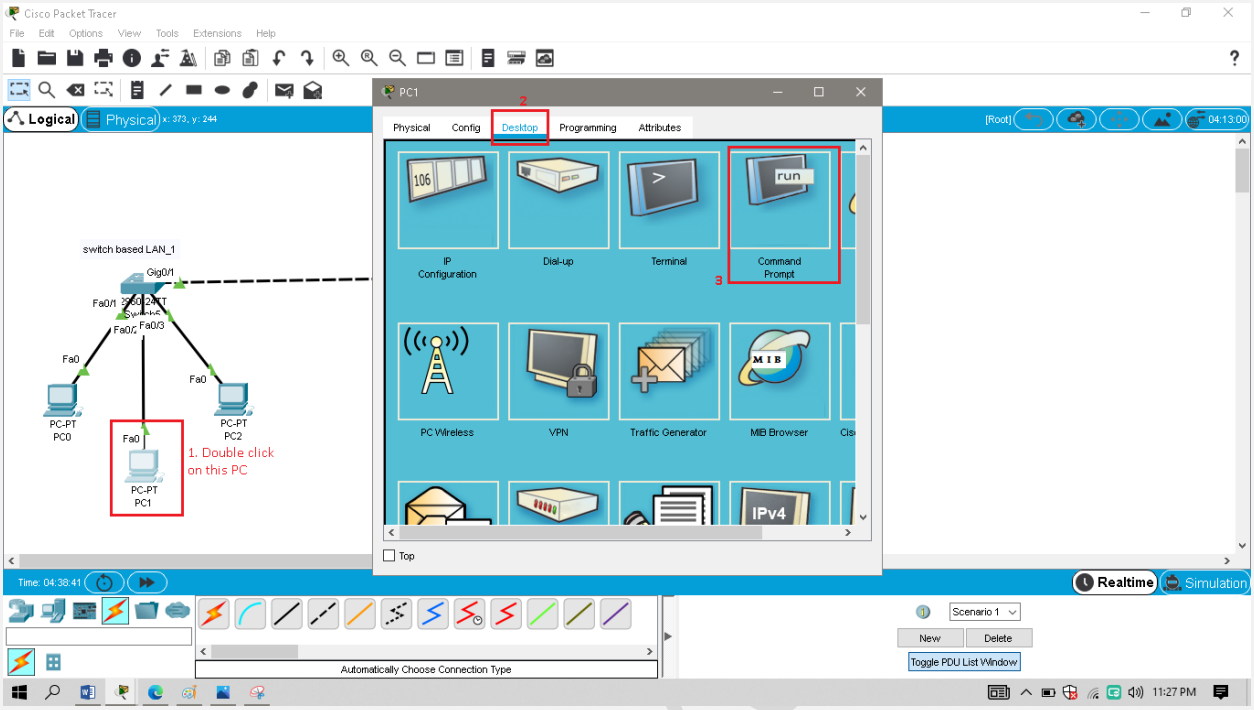


After this process ***Acknowledgment*** starts in the reverse way as shown below;



USING PING COMMAND IN CMD_PROMPT TO PING A PC:

1ST of all we have to select **Real-time Mode** then we will **double click** on the a PC from which we want to **ping** other PC as shown below;



so in **Command Prompt** we will use the *ping command* to ping a PC(*ping ip-address of that PC which we want to ping*) as shown below;