Table of Contents

Preface

This lab manual is designed for a one semester course in Data Communication and Computer Networks. The pre-requisite for students using this course and lab is the basic understanding of computer networks.

Lab # 01: Introduction to Wireshark

Introduction

One's understanding of network protocols can often be greatly deepened by "seeing protocols in action" and by "playing around with protocols" – observing the sequence of messages exchanged between two protocol entities, delving down into the details of protocol operation, and causing protocols to perform certain actions and then observing these actions and their consequences. This can be done in simulated scenarios or in a "real" network environment such as the Internet. In the Wireshark labs you'll be doing in this course, you'll be running various network applications in different scenarios using your own computer (or you can borrow a friend; let me know if you don't have access to a computer where you can install/run Wireshark). You'll observe the network protocols in your computer "in action," interacting and exchanging messages with protocol entities executing elsewhere on the Internet. Thus, you and your computer will be an integral part of these "live" labs. You'll observe, and you'll learn, by doing.

Getting Wireshark

In order to run Wireshark, you will need to have access to a computer that supports both Wireshark and the *libpcap* or *WinPCap* packet capture library. The *libpcap* software will be installed for you if it is not installed within your operating system when you install Wireshark.

See http://www.wireshark.org/download.html for a list of supported operating systems and download sites. Download and install the Wireshark software:

- Go to http://www.wireshark.org/download.html and download and install the Wireshark binary for your computer.

Running Wireshark

When you run the Wireshark program, you'll get a startup screen, as shown below:

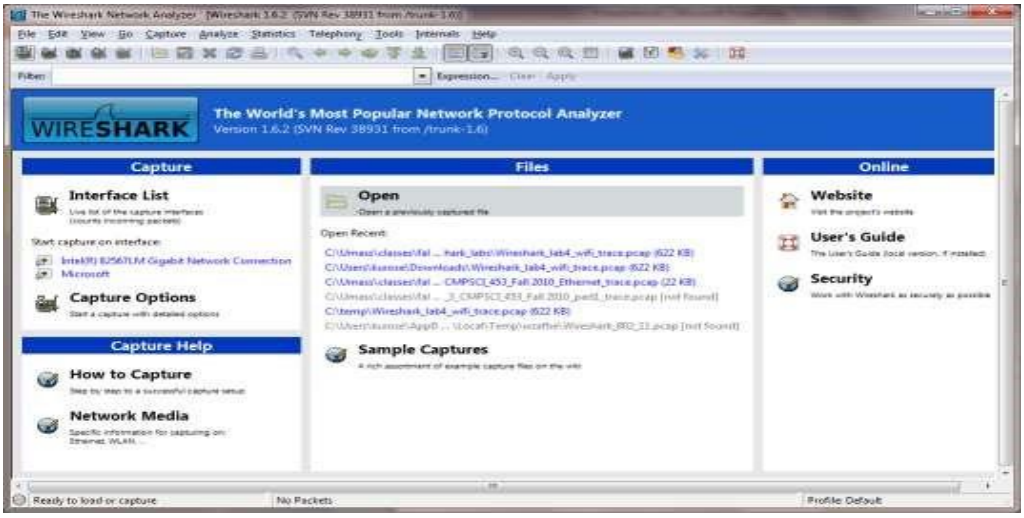


Figure 1: Initial Wireshark Screen

Take a look at the upper left-hand side of the screen – you'll see an "Interface list". This is the list of network interfaces on your computer. Once you choose an interface, Wireshark will capture all packets on that interface. In the example above, there is an Ethernet interface (Gigabit network Connection) and a wireless interface ("Microsoft").

If you click on one of these interfaces to start packet capture (i.e., for Wireshark to begin capturing all packets being sent to/from that interface), a screen like the one below will be displayed, showing information about the packets being captured. Once you start packet capture, you can stop it by using the Capture pull down menu and selecting Stop.

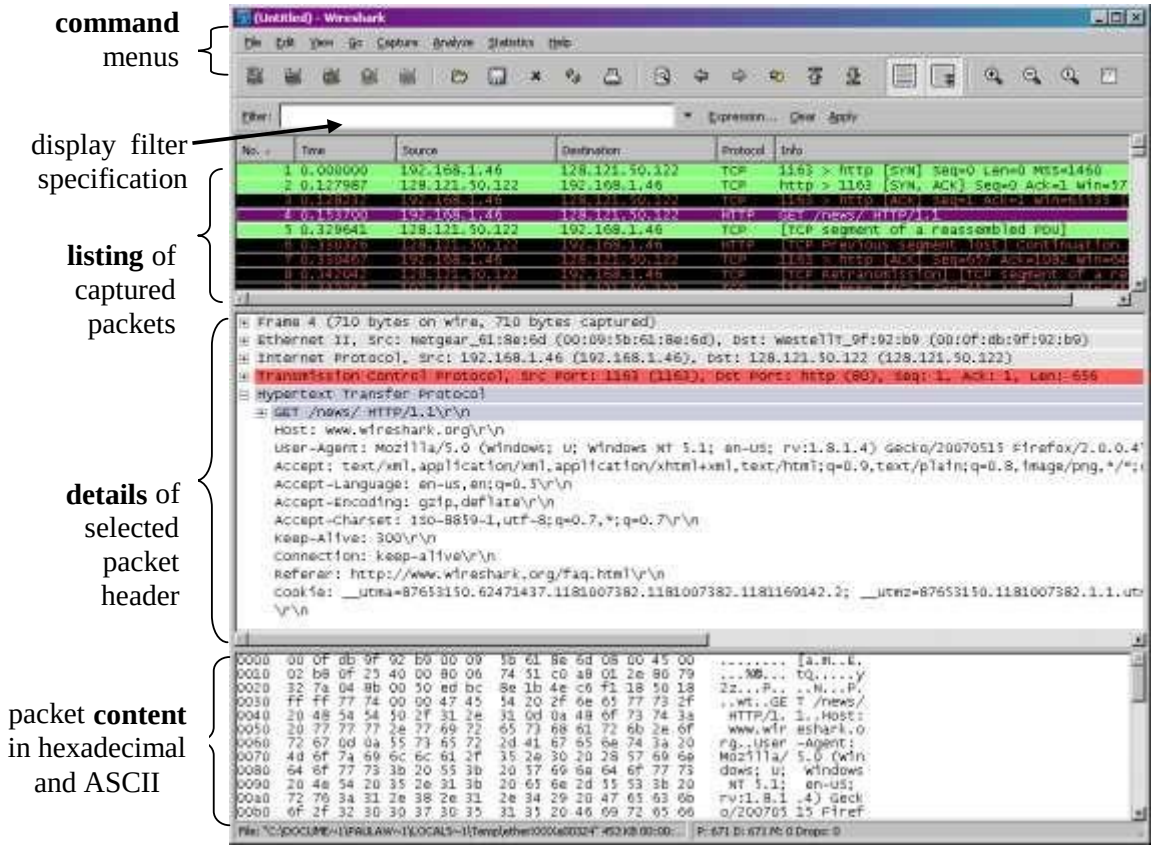


Figure 2: Wireshark Graphical User Interface, during packet capture and analysis

The Wireshark interface has five major components:

1. The **command menus** are standard pulldown menus located at the top of the window. Of interest to us now are the File and Capture menus. The File menu allows you to save captured packet data or open a file containing previously captured packet data and exit the Wireshark application. The Capture menu allows you to begin packet capture.
2. The **packet-listing window** displays a one-line summary for each packet captured, including the packet number (assigned by Wireshark; this is *not* a packet number contained in any

protocol's header), the time at which the packet was captured, the packet's source and destination addresses, the protocol type, and protocol-specific information contained in the packet. The packet listing can be sorted according to any of these categories by clicking on a column name. The protocol type field lists the highest-level protocol that sent or received this packet, i.e., the protocol that is the source or ultimate sink for this packet.

3. The **packet-header details window** provides details about the packet selected (highlighted) in the packet-listing window. (To select a packet in the packet-listing window, place the cursor over the packet's one-line summary in the packet-listing window and click with the left mouse button.). These details include information about the Ethernet frame (assuming the packet was sent/received over an Ethernet interface) and IP datagram that contains this packet. The amount of Ethernet and IP-layer detail displayed can be expanded or minimized by clicking on the plus minus boxes to the left of the Ethernet frame or IP datagram line in the packet details window. If the packet has been carried over TCP or UDP, TCP or UDP details will also be displayed, which can similarly be expanded or minimized. Finally, details about the highest-level protocol that sent or received this packet are also provided.
4. The **packet-contents window** displays the entire contents of the captured frame, in both ASCII and hexadecimal format.
   Towards the top of the Wireshark graphical user interface, is the **packet display filter field,** into which a protocol name or other information can be entered in order to filter the information displayed in the packet-listing window (and hence the packet-header and packet-contents windows). In the example below, we'll use the packet-display filter field to have Wireshark hide (not display) packets except those that correspond to HTTP messages.

Activity 1:

The best way to learn about any new piece of software is to try it out! We'll assume that your computer is connected to the Internet via a wired Ethernet interface. Indeed, I recommend that you do this first lab on a computer that has a wired Ethernet connection, rather than just a wireless connection. Do the following

1. Start up your favorite web browser, which will display your selected homepage.
2. Start up the Wireshark software. You will initially see a window similar to that shown in Figure 2. Wireshark has not yet begun capturing packets.
3. To begin packet capture, select the Capture pull down menu and select *Interfaces.* This will cause the "Wireshark: Capture Interfaces" window to be displayed, as shown in Figure 3.

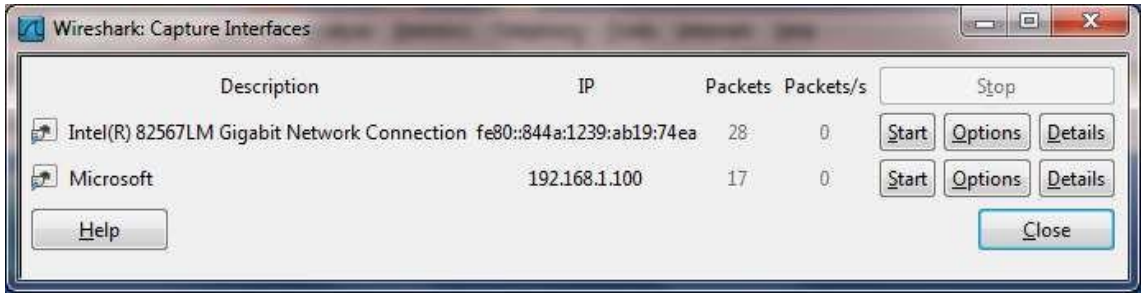


Figure 3: Wireshark Capture Interface Window

4. You'll see a list of the interfaces on your computer as well as a count of the packets that have been observed on that interface so far. Click on *Start* for the interface on which you want to begin packet capture (in the case, the Gigabit network Connection). Packet capture will now begin - Wireshark is now capturing all packets being sent/received from/by your computer!
5. Once you begin packet capture, a window similar to that shown in Figure 3 will appear. This window shows the packets being captured. By selecting *Capture* pulldown menu and selecting *Stop*, you can stop packet capture. But don't stop packet capture yet. Let's capture some interesting packets first. To do so, we'll need to generate some network traffic. Let's do so using a web browser, which will use the HTTP protocol that we will study in detail in class to download content from a website.
6. While Wireshark is running, enter the URL: http://gaia.cs.umass.edu/wireshark-labs/INTROwireshark-file1.html and have that page displayed in your browser. In order to display this page, your browser will contact the HTTP server at gaia.cs.umass.edu and exchange HTTP messages with the server in order to download this page, as discussed in section 2.2 of the text. The Ethernet frames containing these HTTP messages (as well as all other frames passing through your Ethernet adapter) will be captured by Wireshark.
7. After your browser has displayed the INTRO-wireshark-file1.html page (it is a simple one line of congratulations), stop Wireshark packet capture by selecting stop in the Wireshark capture window. The main Wireshark window should now look similar to Figure 3. You now have live packet data that contains all protocol messages exchanged between your computer and other network entities! The HTTP message exchanges with the gaia.cs.umass.edu web server should appear somewhere in the listing of packets captured. But there will be many other types of packets displayed as well (see, e.g., the many different protocol types shown in the *Protocol* column in Figure 3). Even though the only action you took was to download a web page, there were evidently many other protocols running on your computer that are unseen by the user. We'll learn much more about these protocols as we progress through the text! For now, you should just be aware that there is often much more going on than "meet's the eye"!
8. Type in "http" (without the quotes, and in lower case – all protocol names are in lower case in Wireshark) into the display filter specification window at the top of the main Wireshark window. Then select *Apply* (to the right of where you entered "http"). This will cause only HTTP message to be displayed in the packet-listing window.
9. Find the HTTP GET message that was sent from your computer to the gaia.cs.umass.edu HTTP server. (Look for an HTTP GET message in the "listing of captured packets" portion of the Wireshark window (see Figure 3) that shows "GET" followed by the gaia.cs.umass.edu URL that you entered. When you select the HTTP GET message, the Ethernet frame, IP datagram, TCP segment, and HTTP message header information will be displayed in the packet-header window[1]. By clicking on '+' and '-' right-pointing and down-

pointing arrowheads to the left side of the packet details window, *minimize* the amount of Frame, Ethernet, Internet Protocol, and Transmission Control Protocol information displayed. *Maximize* the amount information displayed about the HTTP protocol. Your Wireshark display should now look roughly (Note, in particular, the minimized amount of protocol information for all protocols except HTTP, and the maximized amount of protocol information for HTTP in the packet-header window).

The goal of this first lab was primarily to introduce you to Wireshark. The following questions will demonstrate that you've been able to get Wireshark up and running and have explored some of its capabilities. Answer the following questions, based on your Wireshark experimentation:

1. List 3 different protocols that appear in the protocol column in the unfiltered packet listing window in step 7 above.
2. How long did it take from when the HTTP GET message was sent until the HTTP OK reply was received? (By default, the value of the Time column in the packet-listing window is the amount of time, in seconds, since Wireshark tracing began. To display the Time field in time-of-day format, select the Wireshark *View* pull down menu, then select Time *Display Format*, then select *Time-of-day*.)
3. What is the Internet address of the gaia.cs.umass.edu (also known as wwwnet.cs.umass.edu)? What is the Internet address of your computer?
4. Print the two HTTP messages (GET and OK) referred to in question 2 above. To do so, select *Print* from the Wireshark *File* command menu, and select the "*Selected Packet Only"* and *"Print as displayed"* radial buttons, and then click OK.

Lab # 02: Introduction to Packet Tracer

Introduction

Packet Tracer provides a simulation-based environment for networking training. It offers a unique combination of visualization tools, complex assessment and activity authoring capabilities, and opportunities for multi-user collaboration and competition. For students, Packet Tracer offers extensive learning benefits:

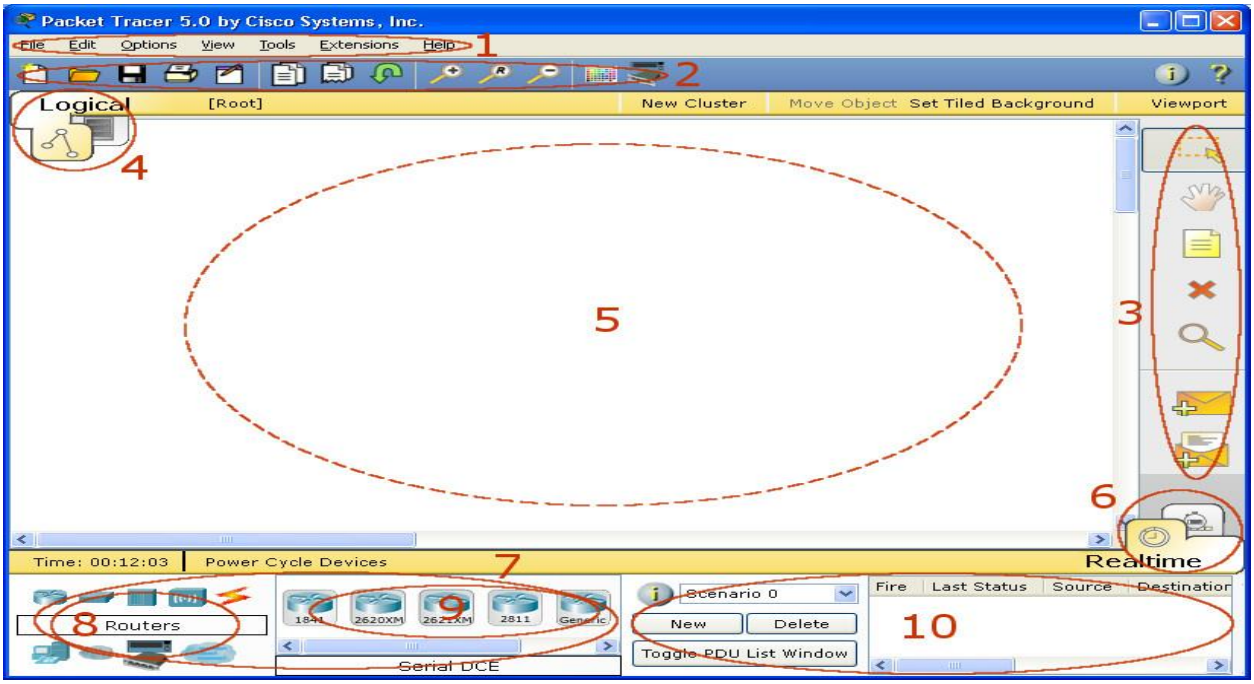
1. Provides a versatile practice and visualization environment for the design, configuration, and troubleshooting of network environments

2. Offers an exploratory development environment that enables users to design, build, and configure networks with drag-and-drop devices

Lab Activities:

When you open Packet Tracer 5.0, by default you will be presented with the following interface:

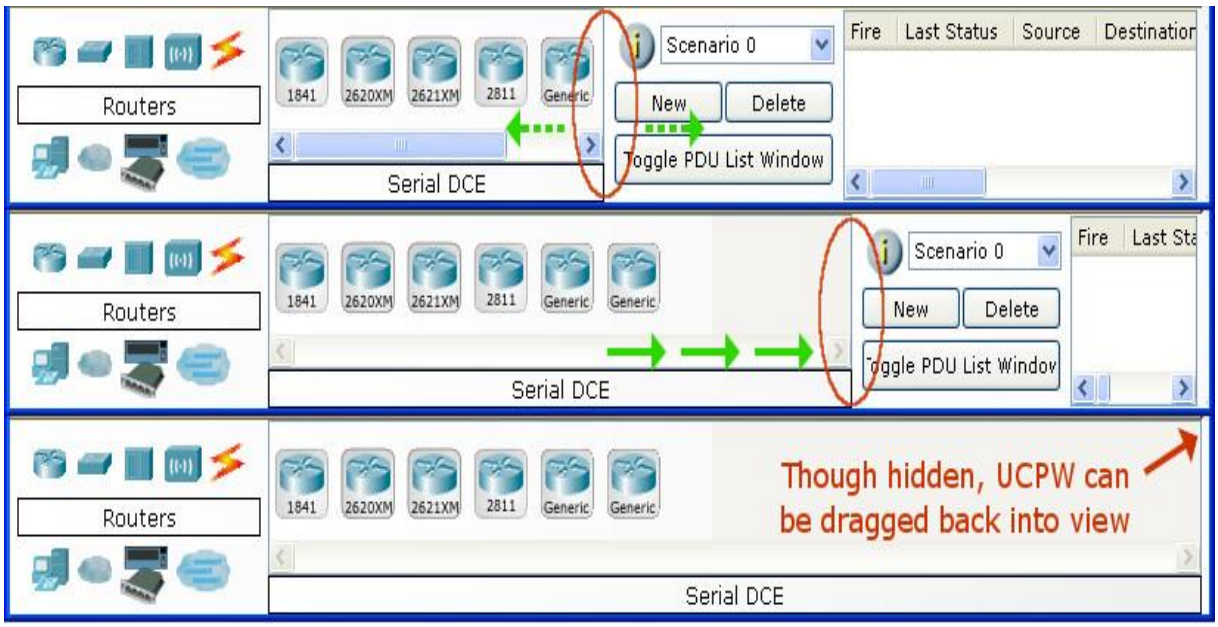
This initial interface contains ten components. If you are unsure of what a particular interface item does, move your mouse over the item and a help balloon will explain the item.



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 Box**	This box contains the type of devices and connections available in Packet Tracer 5.0. The **Device-Specific Selection** Box will change depending on which type of device you choose.
9	**Device-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.

You can freely resize the **User Created Packet Window** (UCPW) by placing the cursor near the left edge of the window (it will turn into a "resize" cursor) and then drag the cursor left or right. You can hide the window from view by dragging the edge all the way to the right. When the UCPW is hidden, you can bring it back by placing the cursor on the edge (notice when the resize cursor appears) and then dragging the edge back.



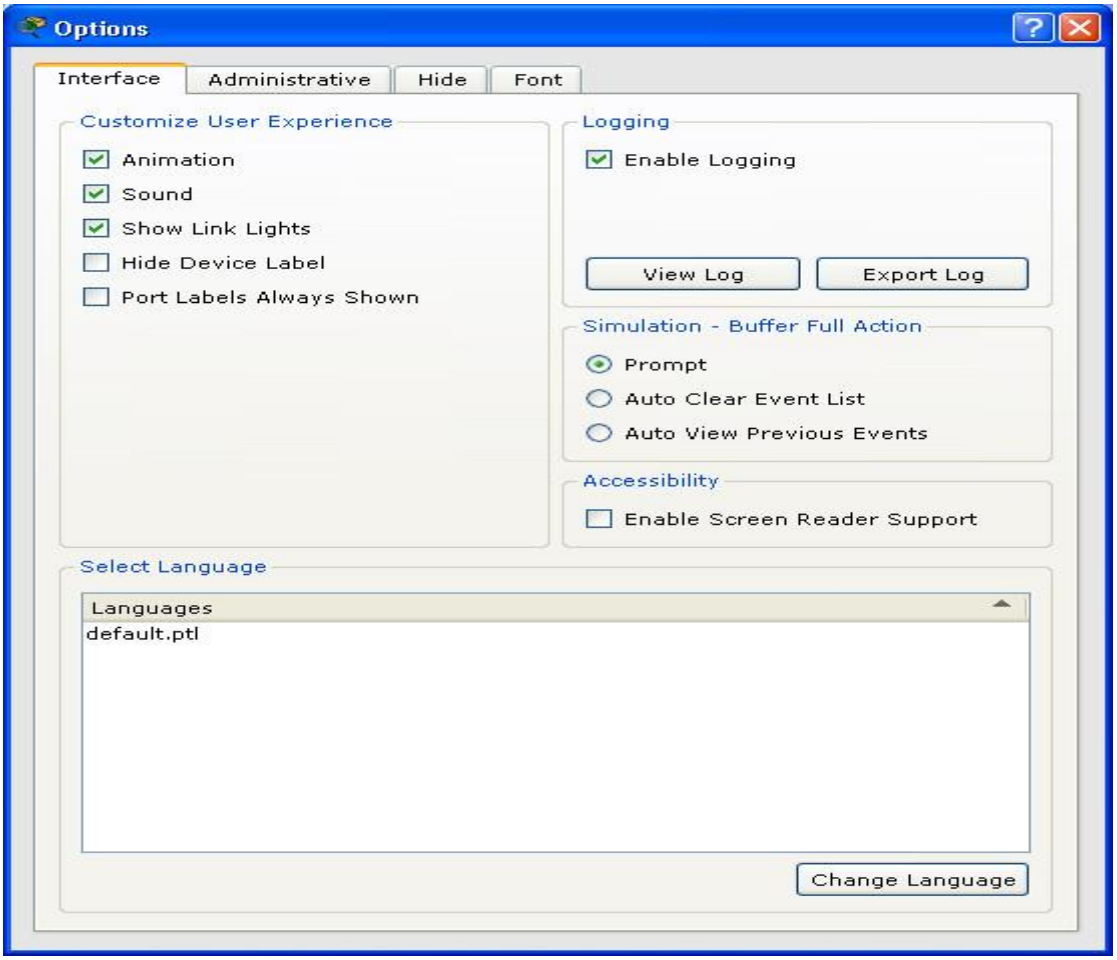
Workspaces and Modes

Packet Tracer has two workspaces (Logical and Physical) and two modes (Realtime and simulation). Upon startup, you are in the Logical Workspace in Realtime Mode. You can build your network and see it run in real time in this configuration. You can switch to Simulation mode to run controlled networking scenarios. You can also switch to the Physical Workspace to arrange the physical aspects (such as the location) of your devices. Note that you view a simulation while you are in the Physical Workspace. You should return to the Logical Workspace after you are done in the Physical Workspace.

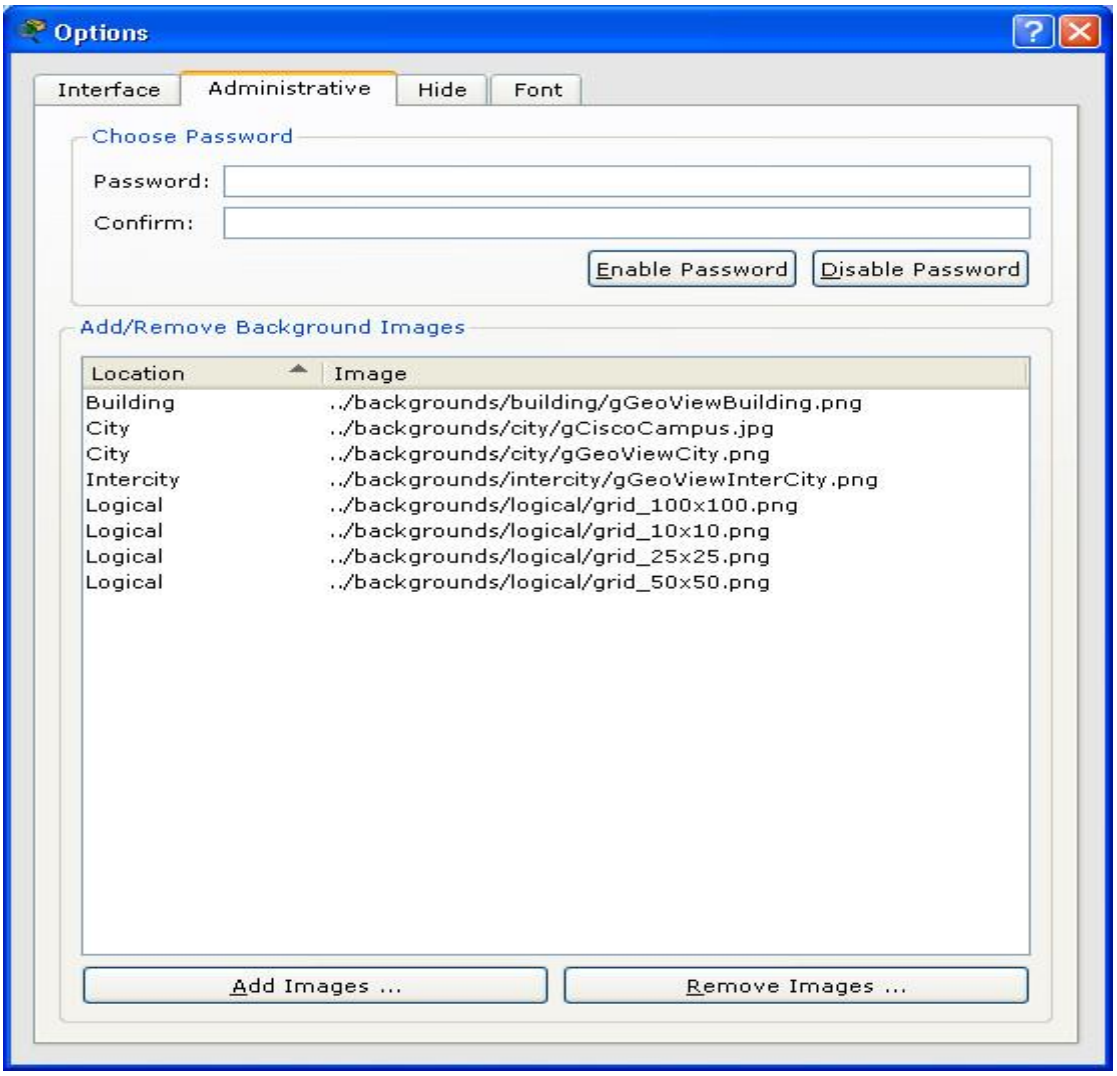
Setting Preferences

You can customize your Packet Tracer experience by setting your own preferences. From the **Menu Bar**, select **Options** > **Preferences** (or simply press **Ctrl + R**) to view the program settings.

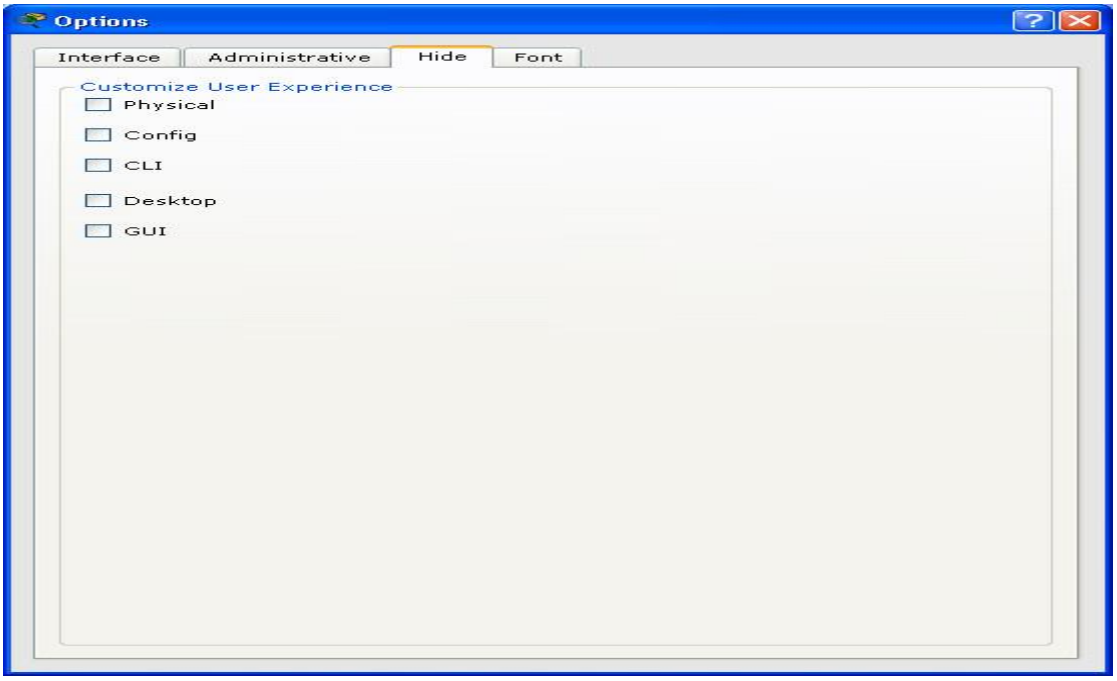
Under the **Interface** panel, you can toggle the **Animation**, **Sound**, and **Show Link Lights** settings to suit the performance of your system and your preferences. You can also manage information clutter with the **Hide Device Label** and **Port Labels Always Shown** settings. The **Logging** feature allows the program to capture all Cisco IOS commands that you enter and export them to a text file (refer to the "Configuring Devices" page for more information). The **Simulation - Buffer Full Action** feature allows you to set the preferred action that Packet Tracer 5.0 will perform. You can set the action to **Prompt** if you want to be prompted when the Simulation buffer is full. At the prompt, you can either **Clear Event List** or **View Previous Events**. Alternatively, you can set the action to either **Auto Clear Event List** to allow Packet Tracer 5.0 to automatically clear the Event List when the buffer is full, or you can set the action to **Auto View Previous Events** to automatically view the previous events. The **Enable Screen Reader Support** accessibility feature reads out all the titles and descriptions of the visible window that has the focus. Lastly, you can also change the base language of the program by choosing from the **Languages** list and then pressing the **Change Language** button.



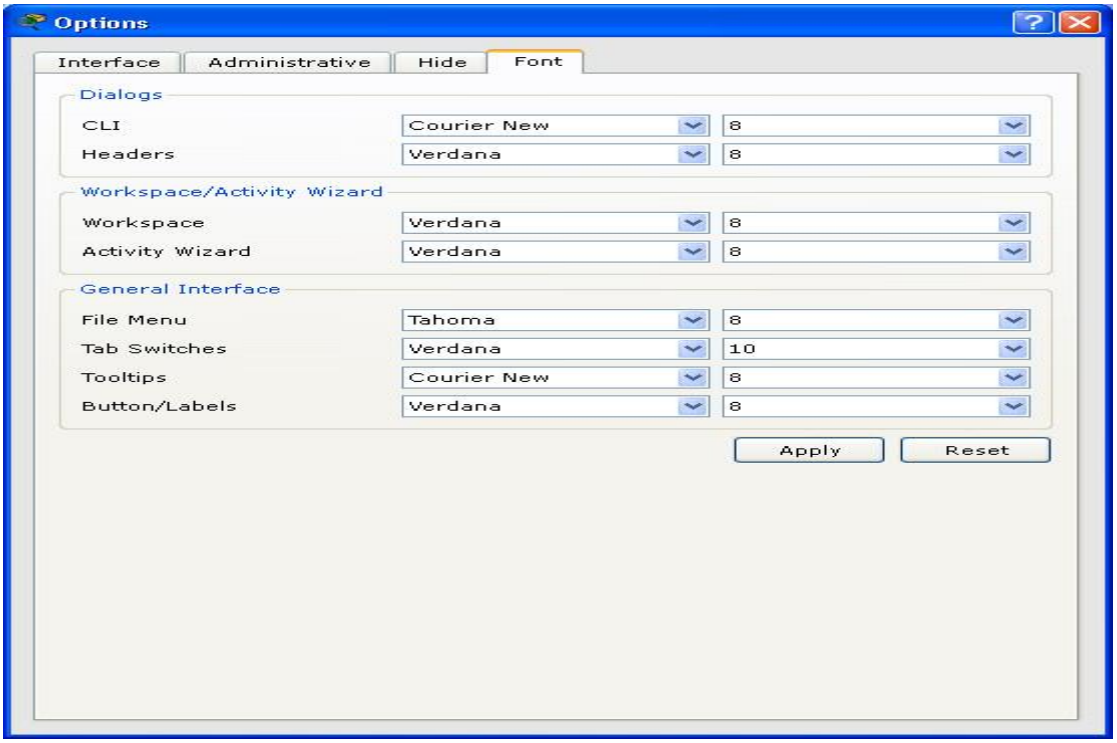
Under the Administrative panel, you can Add/Remove Background Images that are available in the program. You can also set a Password to prevent others from tampering with the images. Note that the password is case-sensitive.



Under the Hide panel, you can choose to hide or show the Physical, Config, CLI, Desktop, and GUI tabs in the device edit dialog.

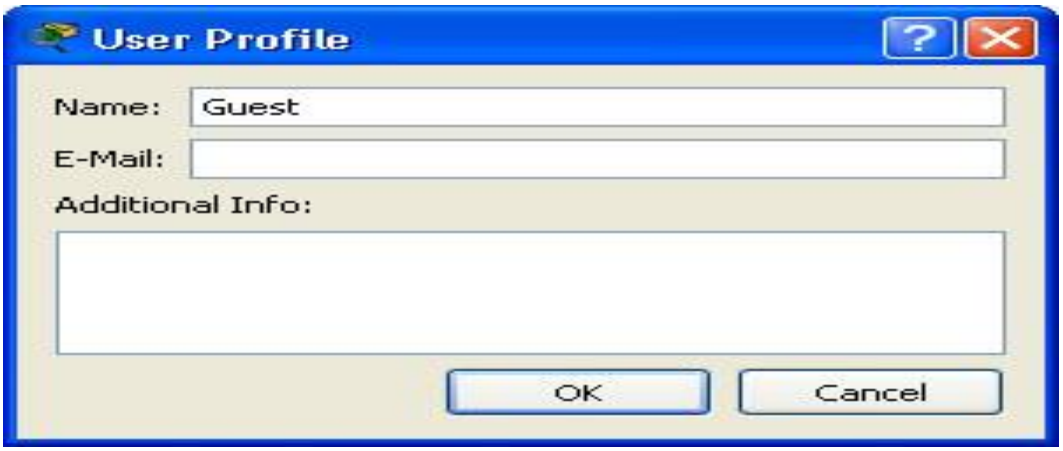


Under the Font panel, you can select different fonts and font sizes for the Dialogs, Workspace/Activity Wizard, and the General Interface.



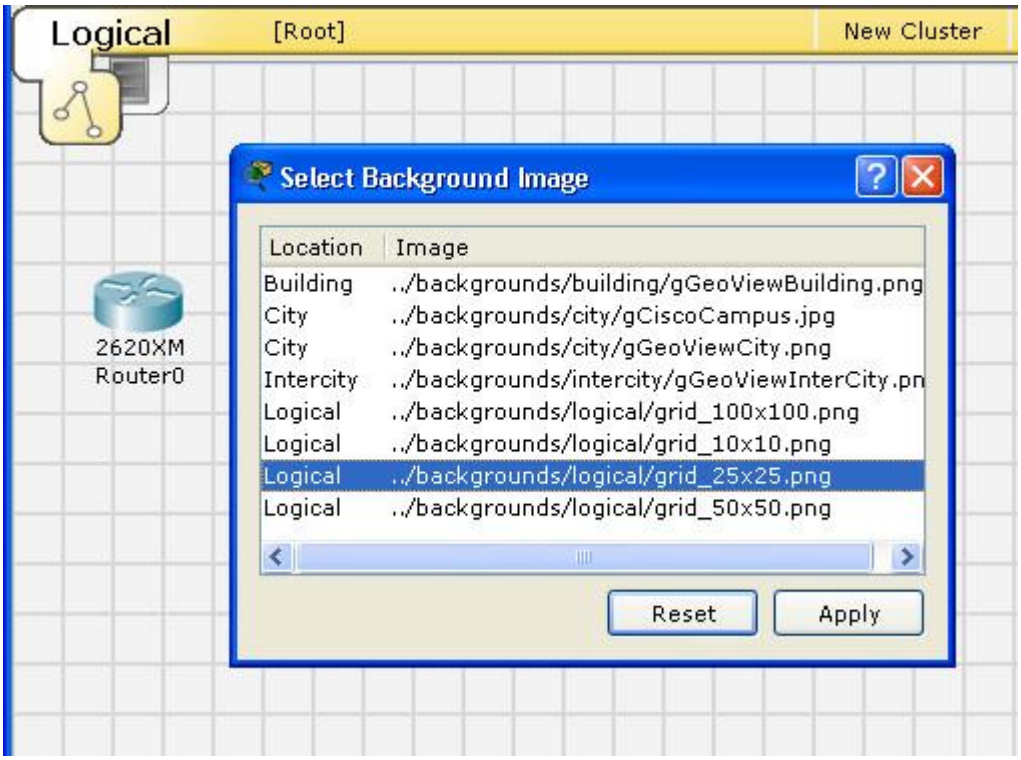
Setting a User Profile

You can set your user profile for activity assessment and Multi-user identification. From the Menu Bar, select Options > User Profile to view the User Profile dialog. In the User Profile dialog, you can enter your Name, E-Mail, and any Additional Info about yourself that you may want to share.



Setting a Background

You can replace the blank workspace with a background image of your choice. You can only set background images that are available in the Administrative panel in Preferences. To set a background, press the Set Tiled Background button in the Logical Workspace Bar. Choose from the list of available images from the Select Background Image window, and press the Apply button. You can revert to a blank background at any time by pressing the Reset button.



You can create or customize your own images and use them as backgrounds in the Logical Workspace. Just put image files in the ./backgrounds/logical folder of the program and add them to the Administrative panel list. Note that background images do not affect any network functions. They are simply visual aids.

The recommended format for background images is .png. Other supported file formats are .jpg and .bmp.

When adding photorealistic files, it is best to use .jpg format. For text or drawings, use .png or .bmp formats.

Lab # 03 (a): Introduction to IP Addressing (Classful addressing)

IP address:

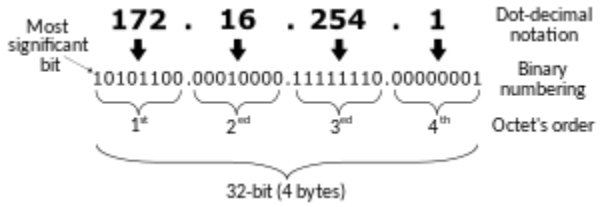
IP address is an address having information about how to reach a specific host, especially outside the LAN. It is a 32-bit address that identifies a connection to the Internet. The IP addresses are universally unique. The address space of IPv4 is 232 or 4,294,967,296.

IP address notation:

Generally, there are two notations in which IP address is written, dotted decimal notation and binary notation.

Typically, it is written in decimal digits, formatted as four 8-bit fields separated by periods. Each 8-bit field represents a byte of the IPv4 address, each ranging from 0 to 255. This form of representing the bytes of an IPv4 address is often referred to as the **dotted-decimal format.**

In **binary notation** each decimal number is represented by its 8-bit binary. The example of each notation is given below.



Classful addressing:

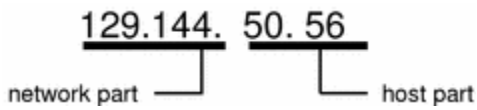
Classful addressing is a concept that divides the available address space of IPv4 into five classes namely A, B, C, D & E.

Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively. The order of bits in the first octet determines the classes of IP address.

IPv4 address is divided into two parts:

Network ID: This part specifies the unique number assigned to your network. It also identifies the class of network assigned.

Host ID: This is the part of the IPv4 address that you assign to each host. It uniquely identifies this machine on your network. Note that for each host on your network, the network part of the address will be the same, but the host part must be different.



The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that class.

Class A:

IP address belonging to class A are assigned to the networks that contain a large number of hosts.

The network ID is 8 bits long.

The host ID is 24 bits long.

The highest order bit of the first octet in class A is always set to 0. The remaining 7 bits in first octet are used to determine network ID. The 24 bits of host ID are used to determine the host in any network. The default subnet mask for class A is 255.0.0.0.

Class B:

IP address belonging to class B are assigned to the networks that ranges from medium-sized to large-sized networks.

The network ID is 16 bits long.

The host ID is 16 bits long.

The higher order bits of the first octet of IP addresses of class B are always set to 10. The remaining 14 bits are used to determine network ID. The 16 bits of host ID is used to determine the host in any network. The default sub-net mask for class B is 255.255.0.0.

Class C:

IP address belonging to class C are assigned to small-sized networks.

The network ID is 24 bits long.

The host ID is 8 bits long.

The higher order bits of the first octet of IP addresses of class C are always set to 110. The remaining 21 bits are used to determine network ID. The 8 bits of host ID is used to determine the host in any network. The default sub-net mask for class C is 255.255.255.0.

Class D:

IP address belonging to class D are reserved for multi-casting. The higher order bits of the first octet of IP addresses belonging to class D are always set to 1110. The remaining bits are for the address that interested hosts recognize. Class D does not possess any sub-net mask.

Class E:

IP addresses belonging to class E are reserved for experimental and research purposes. This class doesn't have any sub-net mask. The higher order bits of first octet of class E are always set to 1111.

Summary of classful addressing

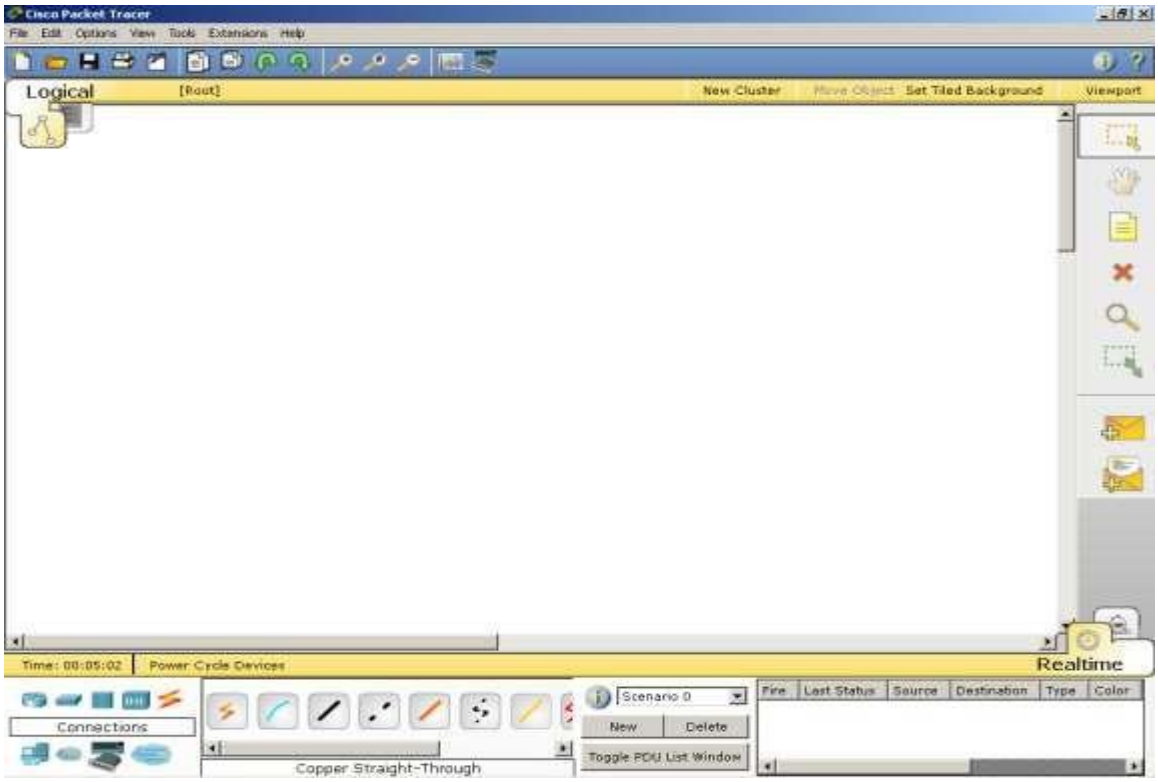
CLASS	LEADING BITS	NET ID BITS	HOST ID BITS	NO. OF NETWORKS	ADDRESSES PER NETWORK	START ADDRESS	END ADDRESS
CLASS A	0	8	24	2^7 (128)	2^{24} (16,777,216)	0.0.0.0	127.255.255.255
CLASS B	10	16	16	2^{14} (16,384)	2^{16} (65,536)	128.0.0.0	191.255.255.255
CLASS C	110	24	8	2^{21} (2,097,152)	2^8 (256)	192.0.0.0	223.255.255.255
CLASS D	1110	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	224.0.0.0	239.255.255.255
CLASS E	1111	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	240.0.0.0	255.255.255.255

Lab # 3 (b): Basic Network Configuration (connecting 2 nodes directly) using Packet Tracer

Introduction

Packet Tracer is a protocol simulator developed by Dennis Frezzo and his team at Cisco Systems. Packet Tracer (PT) is a powerful and dynamic tool that displays the various protocols used in networking, in either Real Time or Simulation mode. The purpose of this lab is to become familiar with the Packet Tracer interface. Learn how to use existing topologies and build your own. This activity will provide an opportunity to explore the standard lab setup using Packet Tracer simulator. Packet Tracer has two file formats it can create: .pkt files (network simulation model files) and .pka files (activity files for practice). When you create your own networks in Packet Tracer or modify existing files from your instructor or your peers, you will often use the .pkt file format. When you launched this activity from the curriculum, these instructions appeared. They are the result of the .pka, Packet Tracer activity file format. At the bottom of these instructions are two buttons: Check Results (which gives you feedback on how much of the activity you have completed) and Reset Activity (which starts the activity over if you want to clear your work or gain more practice).

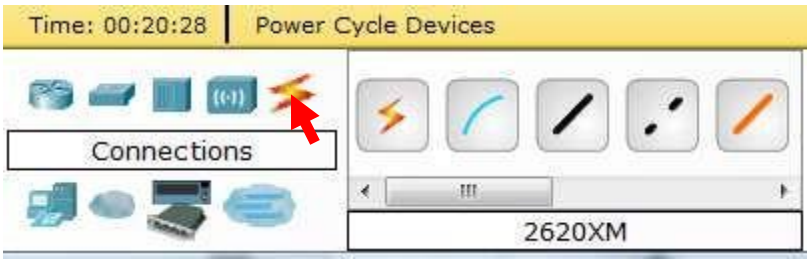
Step 1: Start Packet Tracer

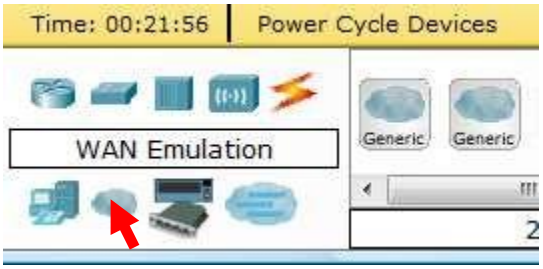


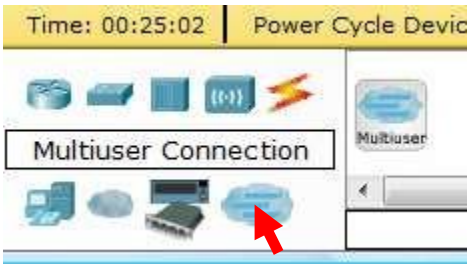
Step 2: Choosing Devices and Connections

We will begin building our network topology by selecting devices and the media in which to connect them. Several types of devices and network connections can be used. For this lab we will keep it simple by using End Devices and Connections.

Single click on each group of devices and connections to display the various choices. The devices you see may differ slightly.

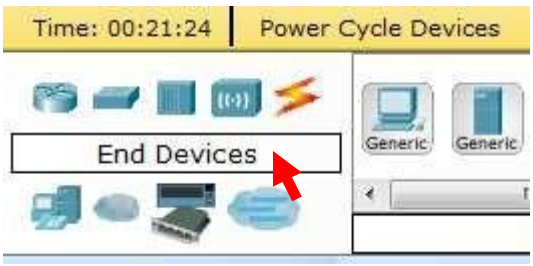






Step 3: Building the Topology – Adding Hosts

Single click on the End Devices.

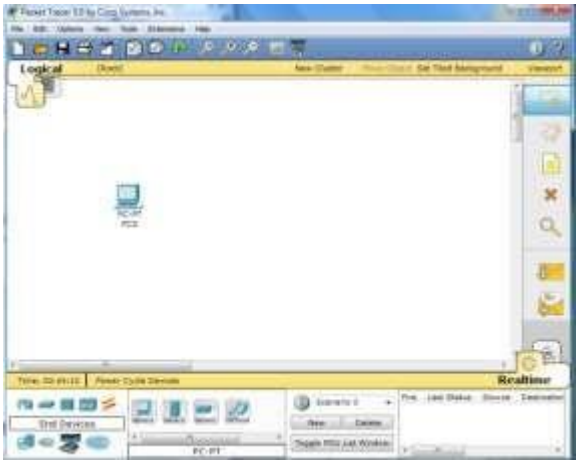


Single click on the Generic host.

Move the cursor into topology area. You will notice it turns into a plus "+" sign.

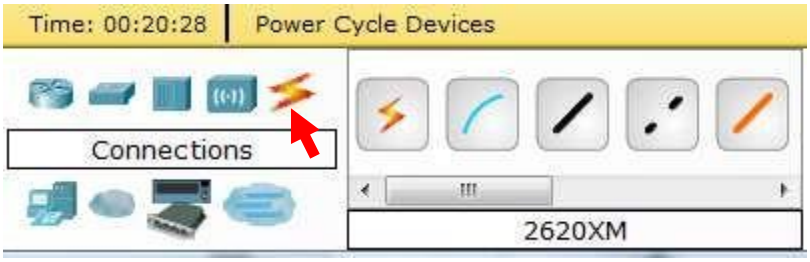
+

Single click in the topology area and it copies the device.



Add another node by following the same steps.

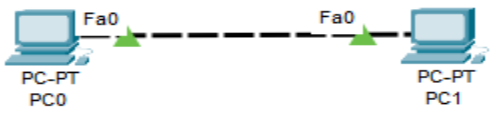
Connect PC0 to PC1:

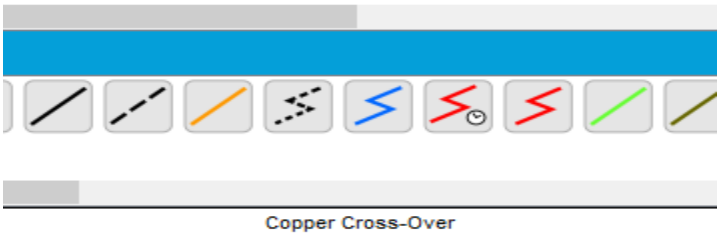


Click once on the copper cross over cable

Perform the following steps to connect PC0 to PC1:

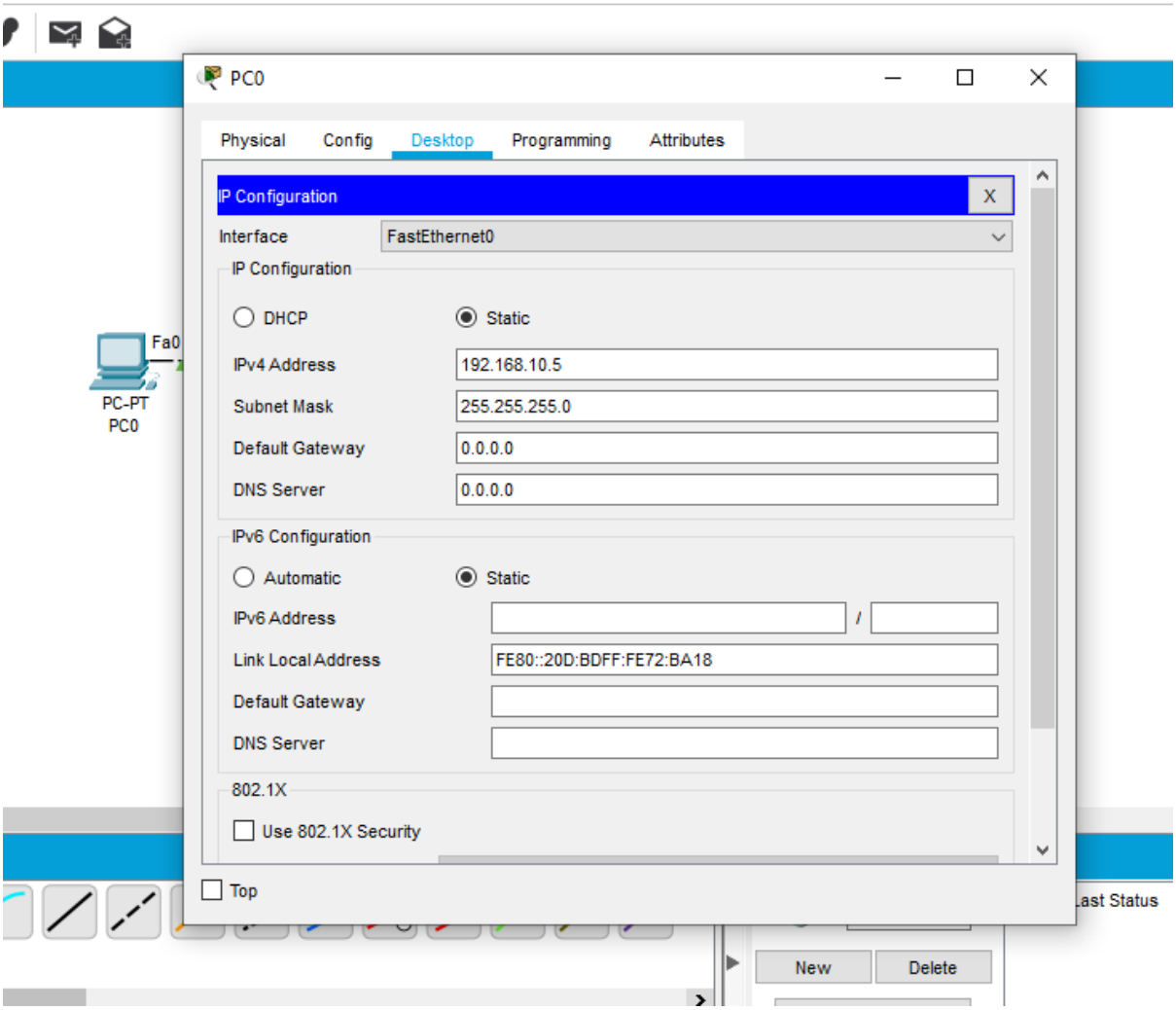
1. Click once on PC0
2. Choose FastEthernet
3. Drag the cursor to PC1
4. Click once on PC1
5. Choose FastEthernet





Step 4: Configuring IP Addresses and Subnet Masks on the Hosts

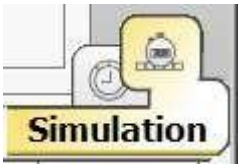
Before we can communicate between the hosts, we need to configure IP Addresses and Subnet Masks on the devices. Click once on PC0. Choose the desktop tab and click on IP configuration. It is also here where you would enter a Gateway IP Address, also known as the default gateway and the DNS Server IP Address. We will discuss this later.



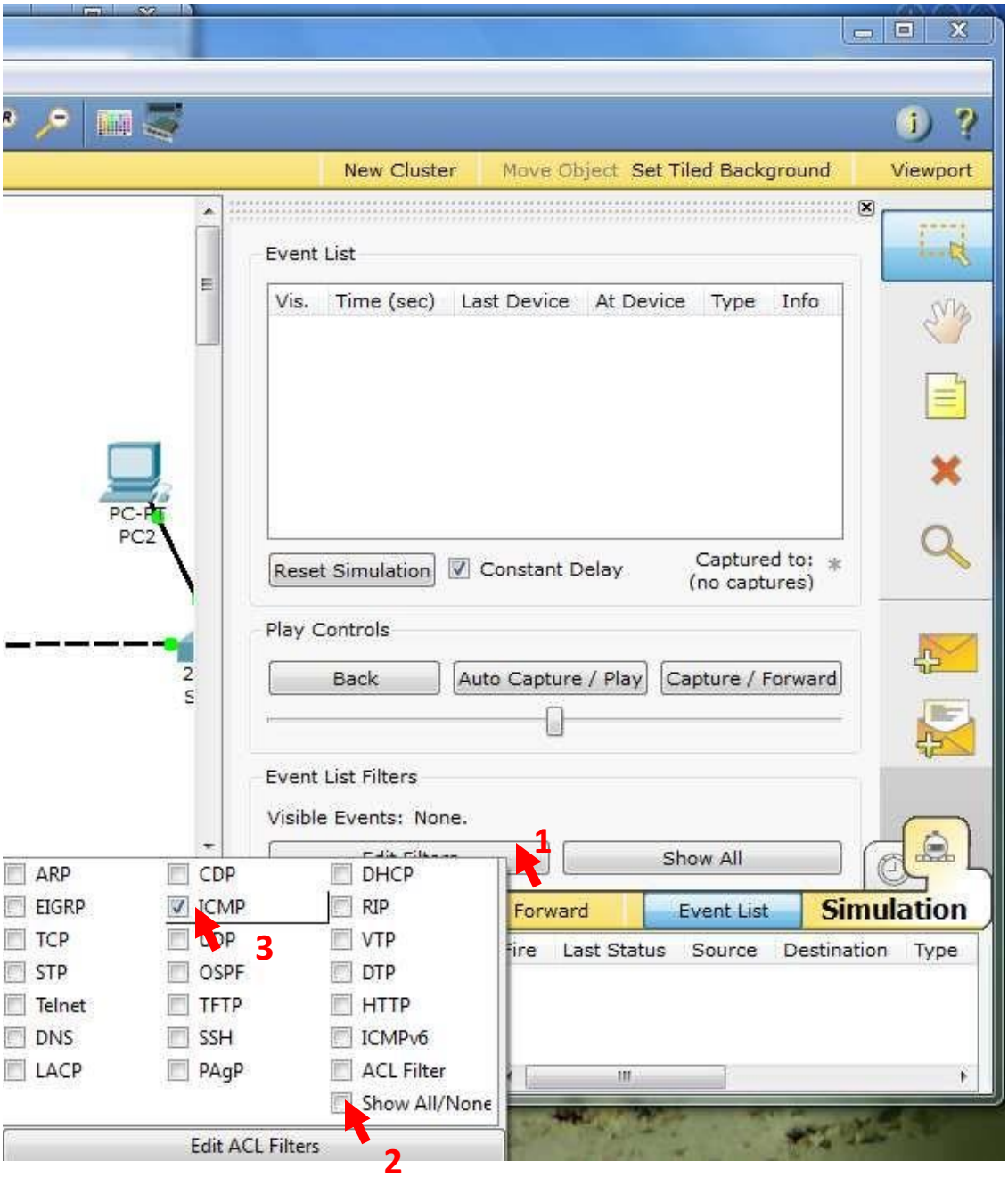
For now, we will add IP address and subnet mask. We will use class C IP address of network 192.168.10.0. Add the IP address as given in the above image and the subnet mask. Repeat the same for PC1 using different IP address from same network. Use 192.168.10.6 for practice.

Step 5: Verifying Connectivity in Simulation Mode

Be sure you are in Simulation mode.



Deselect all filters (All/None) and select only ICMP.



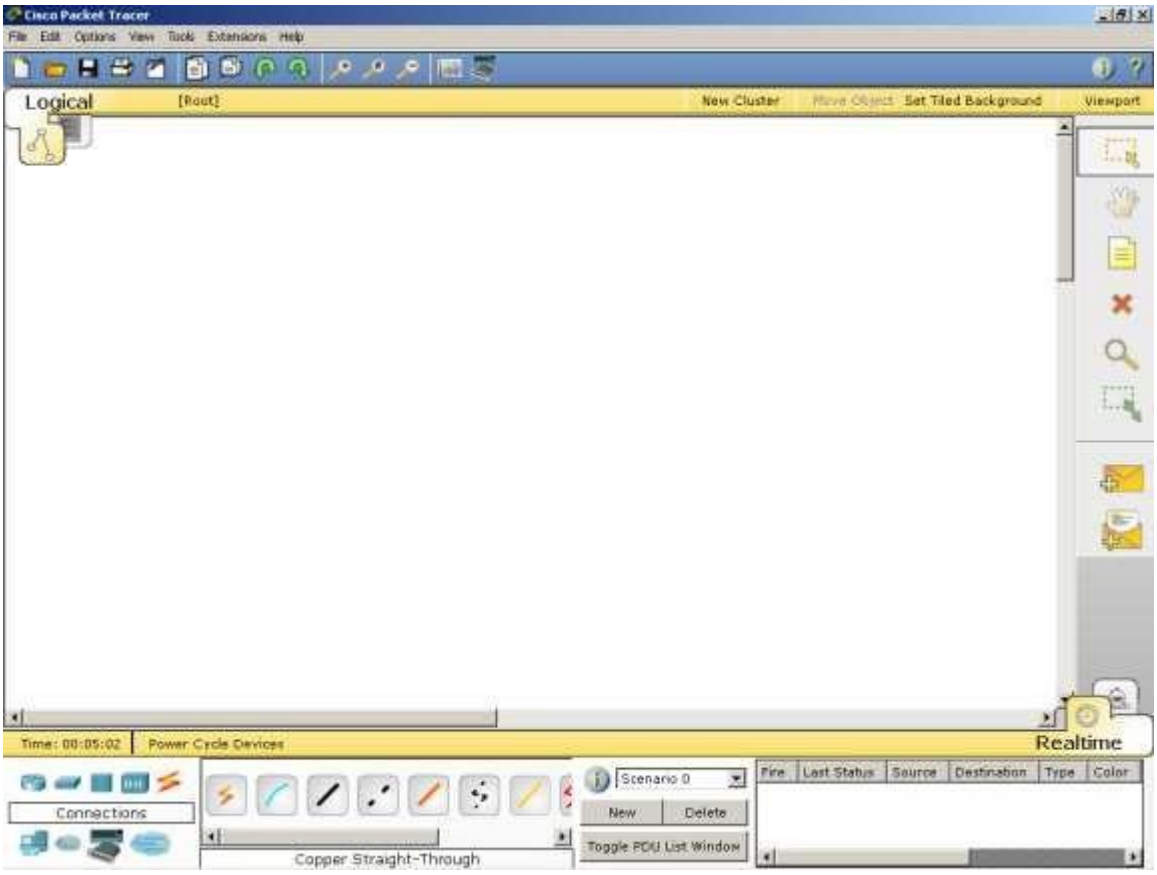
Select the Add Simple PDU tool used to ping devices.

Click once on PC0, then once on PC1. Continue clicking Capture/Forward button until the ICMP ping is completed. You should see the ICMP messages move. The PDU Last Status should show as Successful.

The configuration is complete.

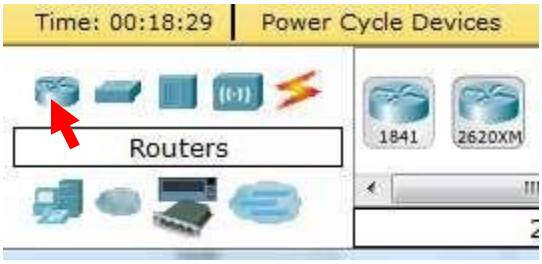
Lab # 3 (c): Basic Network Configuration (connecting multiple end devices through switch) using Packet Tracer

Topology Step 1: Start Packet Tracer

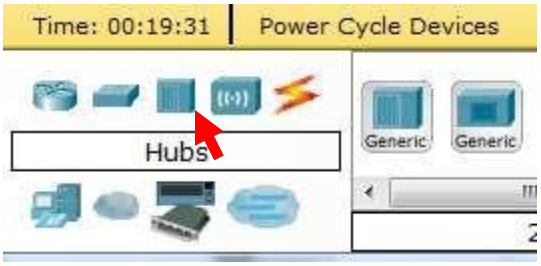


Step 2: Choosing Devices and Connections

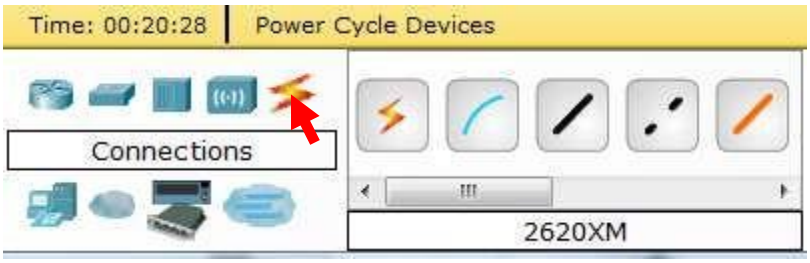
We will begin building our network topology by selecting devices and the media in which to connect them. Several types of devices and network connections can be used. For this lab we will keep it simple by using End Devices, Switches, and Connections. Single click on each group of devices and connections to display the various choices. The devices you see may differ slightly.


Time: 00:18:29
Power Cycle Devices
Routers
1841
2620XM

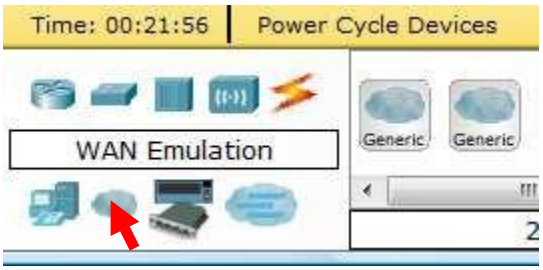

Time: 00:18:55
Power Cycle Devices
Switches
2950-24
2950T


Time: 00:19:31
Power Cycle Devices
Hubs
Generic
Generic

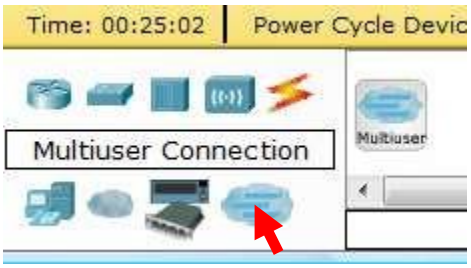

Time: 00:19:56
Power Cycle Devices
Wireless Devices
Generic
Linksys


Time: 00:20:28
Power Cycle Devices
Connections
2620XM


Time: 00:21:24
Power Cycle Devices
End Devices
Generic
Generic

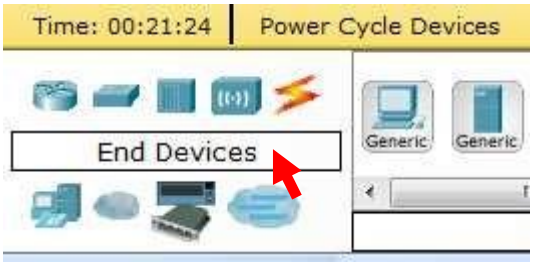

Time: 00:21:56
Power Cycle Devices
WAN Emulation
Generic
Generic


Time: 00:24:29
Power Cycle Devices
Custom Made Devices
1841
2621XM


Time: 00:25:02
Power Cycle Devic
Multiuser Connection
Multiuser

Step 3: Building the Topology – Adding Hosts

Single click on the End Devices.

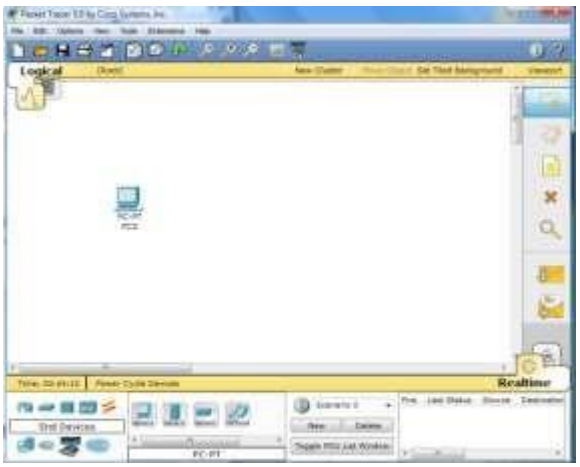


Single click on the Generic host.

Move the cursor into topology area. You will notice it turns into a plus "+" sign.

+

Single click in the topology area and it copies the device.



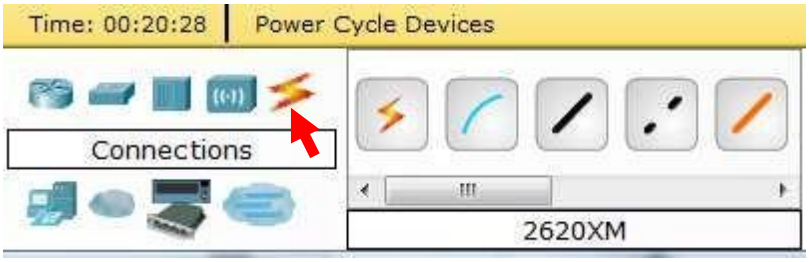
Add three more hosts.

Building the Topology – Connecting the Hosts to Switch.

Adding a Switch. Select a switch, by clicking once on Switches and once on a 2950-24 switch.

Add the switch by moving the plus sign "+" below PC2 and PC3 and click once.

Connect PC2 to switch by first choosing Connections.

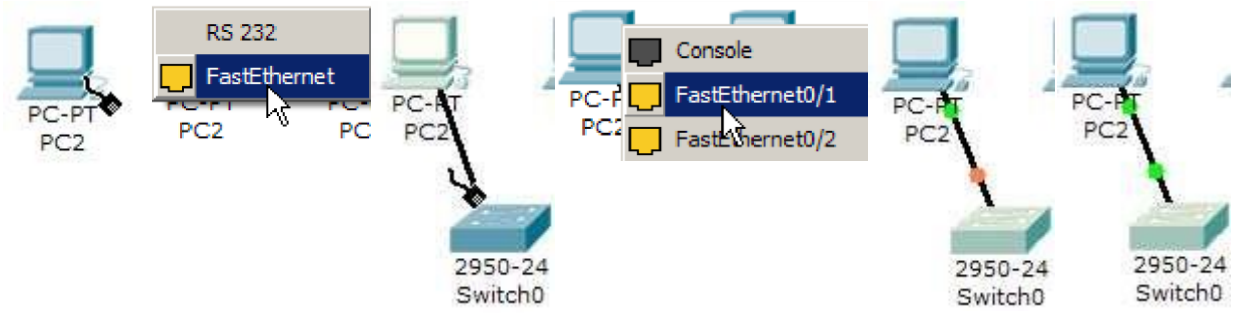


Click once on the Copper Straight-through cable.

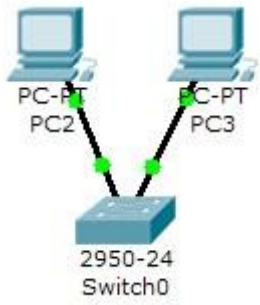
Perform the following steps to connect PC2 to Switch0:

1. Click once on PC2
2. Choose FastEthernet
3. Drag the cursor to Switch0
4. Click once on Switch0 and choose FastEthernet0/1
5. Notice the green link lights on PC2 Ethernet NIC and amber light Switch0 FastEthernet0/1 port. The switch port is temporarily not forwarding frames, while it goes through the stages for the Spanning Tree Protocol (STP) process.
6. After a about 30 seconds the amber light will change to green indicating that the port has entered the forwarding stage. Frames can now forward out the switch port.

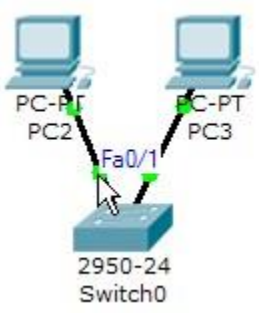
1 2 3 4 5 6



Repeat the steps above for PC3 connecting it to Port 3 on Switch0 on port FastEtherent0/2. (The actual switch port you choose does not matter.)



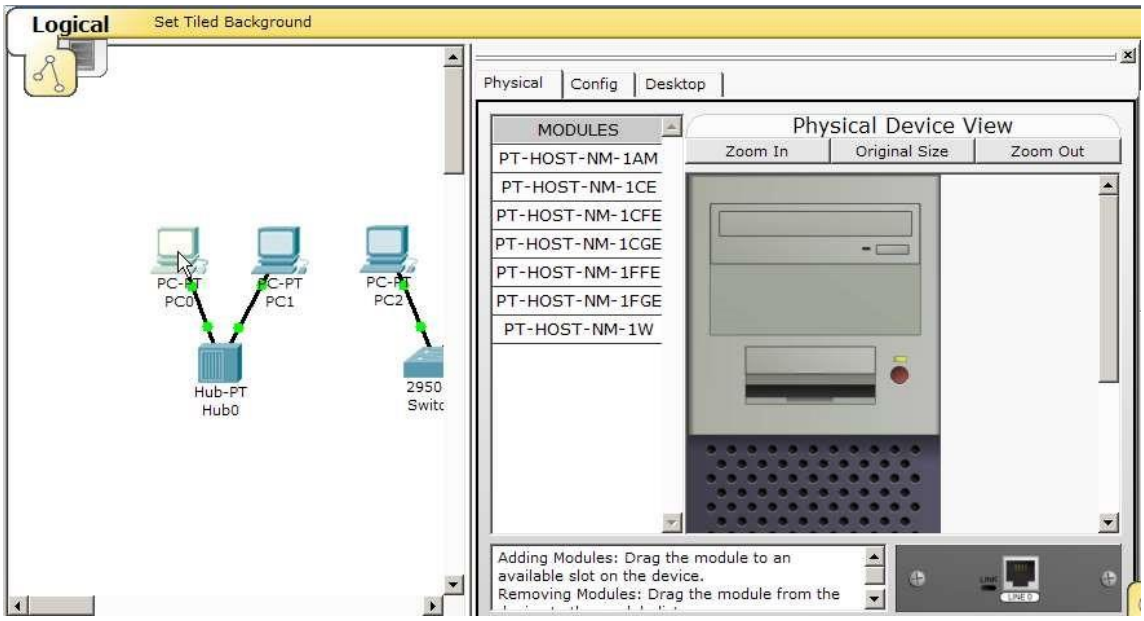
Move the cursor over the link light to view the port number. Fa means FastEthernet, 100 Mbps Ethernet.



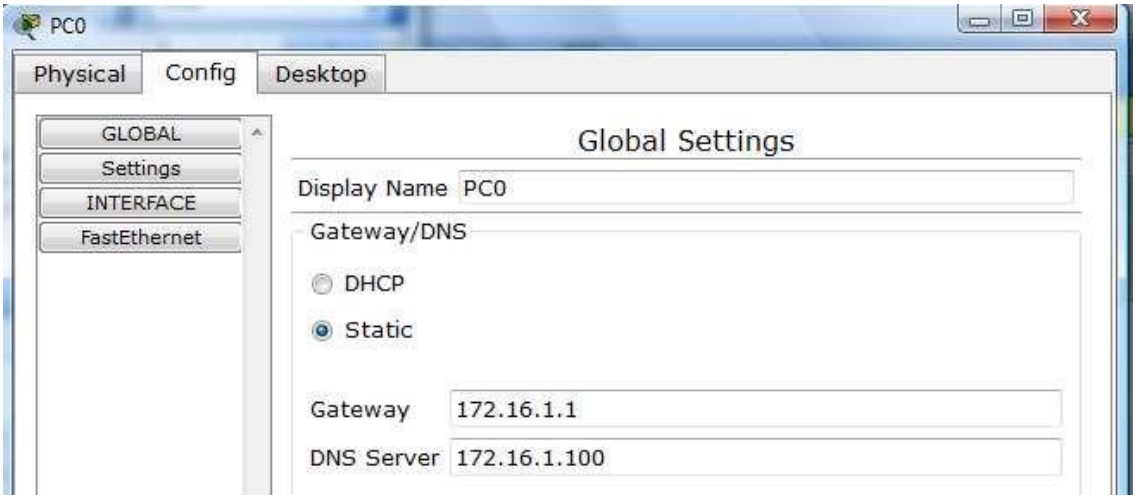
Repeat same steps to connect other PCs with switch.

Step 5: Configuring IP Addresses and Subnet Masks on the Hosts

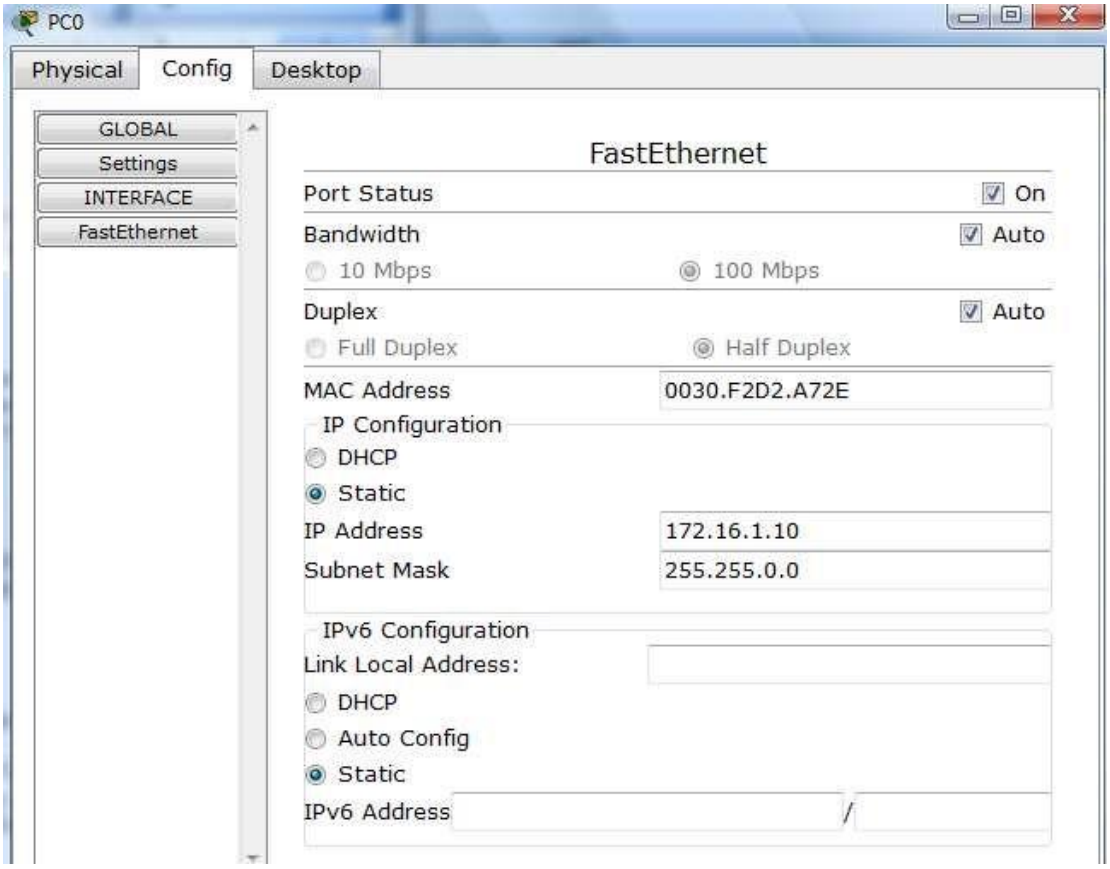
Before we can communicate between the hosts, we need to configure IP Addresses and Subnet Masks on the devices. Click once on PC0.



Choose the Config tab and click on Settings. It is here that you can change the name of PC0. It is also here where you would enter a Gateway IP Address, also known as the default gateway and the DNS Server IP Address. We will discuss this later, but this would be the IP address of the local router. If you want, you can enter the Gateway IP Address 172.16.1.1 and DNS Server IP Address 172.16.1.100, although it will not be used in this lab.



Click on Interface and then FastEthernet. Add the IP Address to 172.16.1.10. Click once in the Subnet Mask field to enter the default Subnet Mask. You can leave this at 255.255.0.0.



Also, notice this is where you can change the Bandwidth (speed) and Duplex of the Ethernet NIC

(Network Interface Card). The default is Auto (autonegotiation), which means the NIC will negotiate with the hub or switch. The bandwidth and/or duplex can be manually set by removing the check from the Auto box and choosing the specific option.

Bandwidth - Auto

If the host is connected to a hub or switch port which can do 100 Mbps, then the Ethernet NIC on the host will choose 100 Mbps (Fast Ethernet). Otherwise, if the hub or switch port can only do 10 Mbps, then the Ethernet NIC on the host will choose 10 Mbps (Ethernet).

Duplex - Auto

Switch: If the host is connected to a switch, and the switch port is configured as Full Duplex (or Autonegotiation), then the Ethernet NIC on the host will choose Full Duplex. If the switch port is configured as Half Duplex, then the Ethernet NIC on the host will choose Half Duplex. (Full Duplex is a much more efficient option.)

The information is automatically saved when entered.

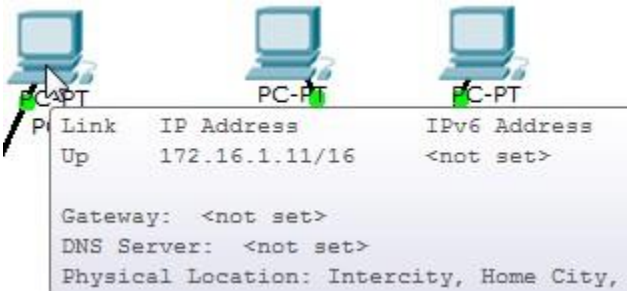
To close this dialog box, click the "X" in the upper right.

Repeat these steps for the other hosts. Use the information below for IP Addresses and Subnet Masks.

Host	IP Address	Subnet Mask
PC0	172.16.1.10	255.255.0.0
PC1	172.16.1.11	255.255.0.0
PC2	172.16.1.12	255.255.0.0
PC3	172.16.1.13	255.255.0.0

Verify the information:

To verify the information that you entered, move the Select tool (arrow) over each host.



Deleting a Device or Link

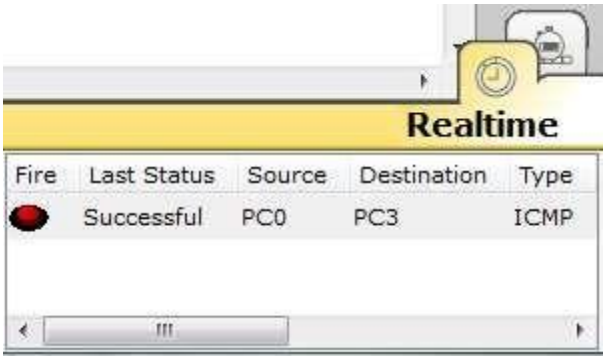
To delete a device or link, choose the Delete tool and click on the item you wish to delete.

Step 7: verify connectivity in Realtime mode

Be sure you are in Realtime mode.

Select the Add Simple PDU tool used to ping devices.

Click once on PC0, then once on PC3.



Change the IP address of PC3 to 172.16.2.13. Perform a ping from PC0 to PC3. What is the ping result?

_____

Return the IP address of PC3 to 172.16.1.13. Change the IP address of PC2 to 172.17.1.12. Perform a ping from PC0 to PC2. What is the ping result?

_____

Resetting the Network

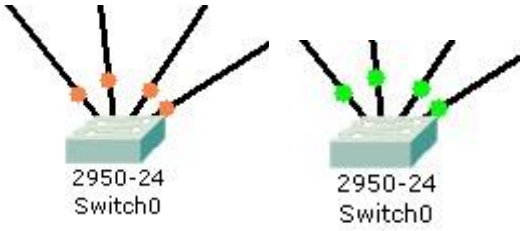
At this point we will want to reset the network, whenever you want to reset the network and begin the simulation again, perform the following tasks:

Click Delete in the PDU area.

Now, Power Cycle Devices and confirm the action.

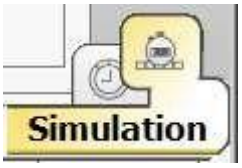
Waiting for Spanning Tree Protocol (STP)

Note: Because Packet Tracer also simulates the Spanning Tree Protocol, at times the switch may show amber lights on its interfaces. You will need to wait for the lights to turn green on the switches before they will forward any Ethernet frames.

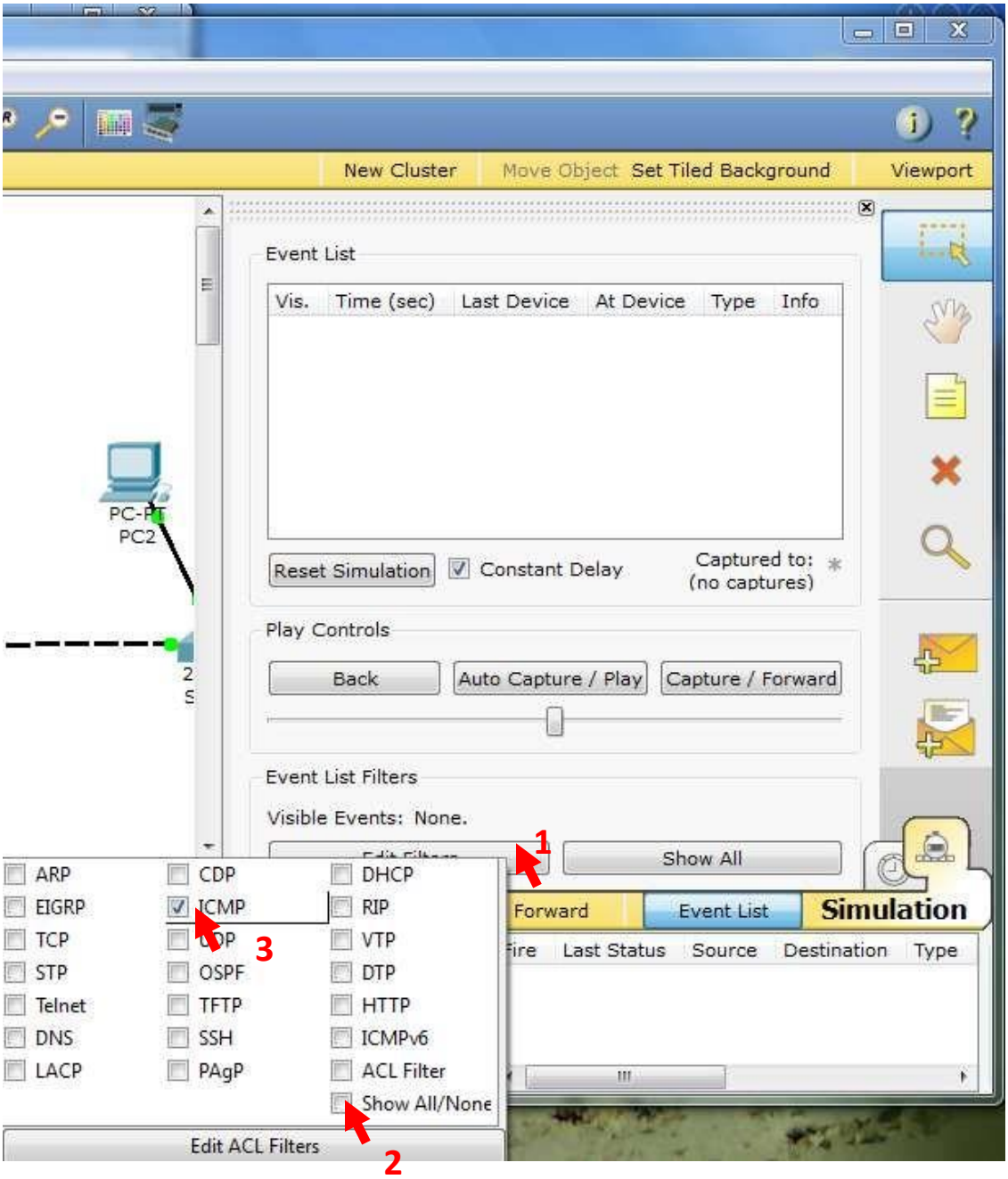


Step 8: Verifying Connectivity in Simulation Mode

Be sure you are in Simulation mode.



Deselect all filters (All/None) and select only ICMP.



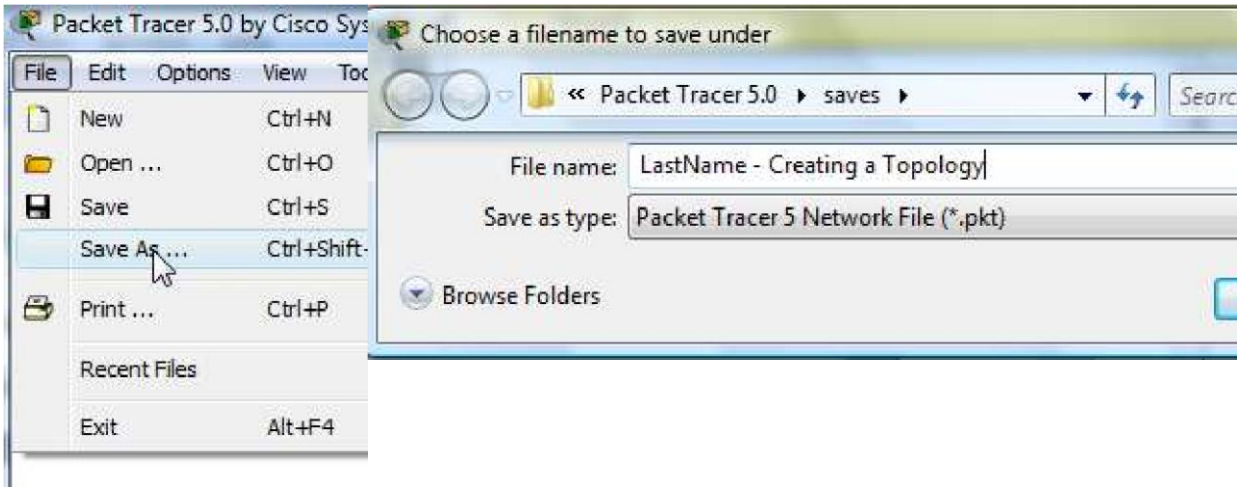
Select the Add Simple PDU tool used to ping devices.

Click once on PC0, then once on PC3.

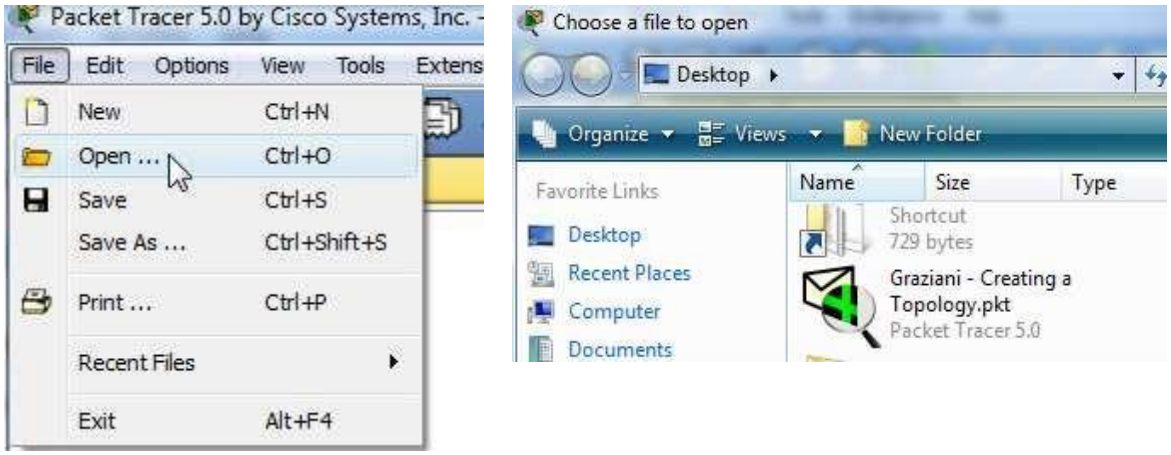
Continue clicking Capture/Forward button until the ICMP ping is completed. You should see the ICMP messages move. The PDU Last Status should show as Successful. Click on Clear Event List if you do not want to look at the events or click Preview Previous Events if you do. For this exercise it does not matter.

Step 9: Saving the Topology

Perform the following steps to save the topology (uses .pkt file extension).



Opening Existing Topologies



Lab 04 (a): Hyper Text Transfer Protocol (HTTP) using Wireshark

Introduction

In this lab, we'll explore several aspects of the HTTP protocol: the basic GET/response interaction, HTTP message formats, retrieving large HTML files, retrieving HTML files with embedded objects, and HTTP authentication and security.

Lab Activities:

Activity 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 Wireshark 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 displayfilter-specification 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 Wireshark packet capture.
4. Enter the following to your browser http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html Your browser should display the very simple, one-line HTML file.
5. Stop Wireshark packet capture.

Your Wireshark window should look similar to the window shown in Figure 1. If you are unable to run Wireshark on a live network connection, you can download a packet trace that was created when the steps above were followed.[2]

2 Download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the file httpethereal-trace-1. The traces in this zip file were collected by Wireshark running on one of the author's computers, while performing the steps indicated in the Wireshark lab. Once you have downloaded the trace, you can load it into Wireshark and view the trace using the *File* pull down menu, choosing *Open*, and then selecting the http-ethereal-trace-1 trace file. The resulting display should look similar to Figure 1. (The Wireshark user interface displays just a bit differently on different operating systems, and in different versions of Wireshark).

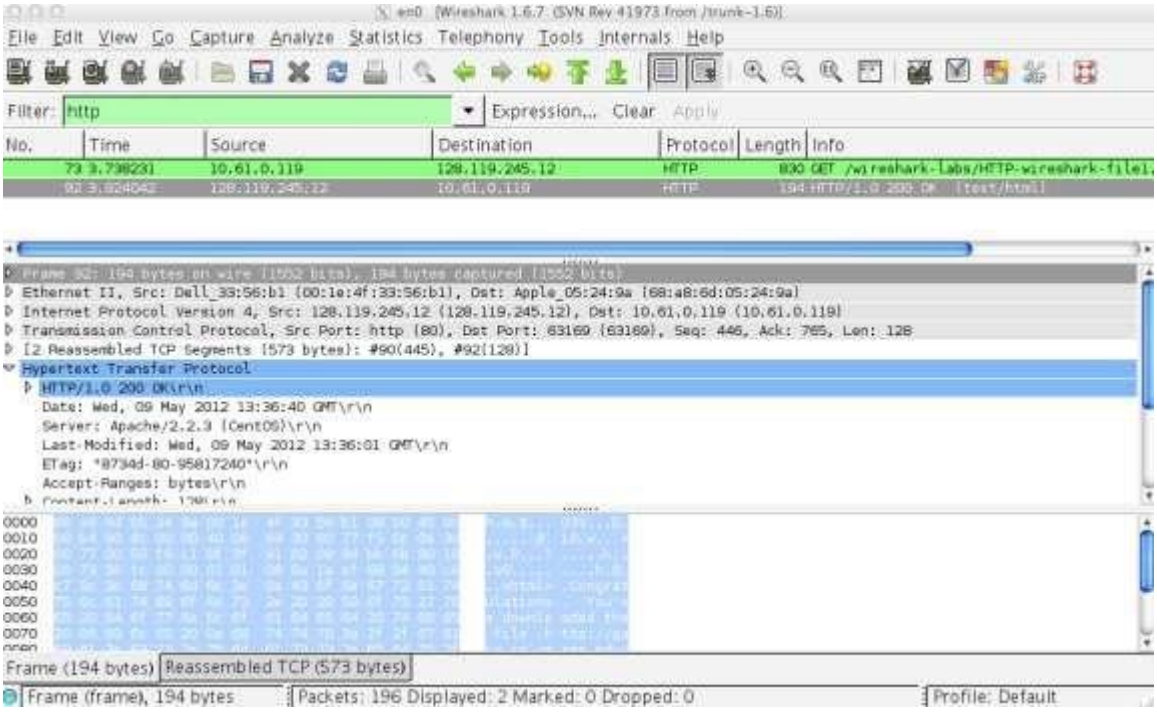


Figure 1: Wireshark Display after http://gaia.cs.umass.edu/wireshark-labs/ HTTP-wiresharkfile1.html has been retrieved by your browser

The example in Figure 1 shows in the packet-listing window that two HTTP messages were captured: the GET message (from your browser to the gaia.cs.umass.edu web server) and the response message from the server to your browser. The packet-contents window shows details of the selected message (in this case the HTTP OK message, which is highlighted in the packet-listing window). Recall that since the HTTP message was carried inside a TCP segment, which was carried inside an IP datagram, which was carried within an Ethernet frame, Wireshark displays the Frame, Ethernet, IP, and TCP packet information as well. We want to minimize the amount of non-HTTP data displayed (we're interested in HTTP here, and will be investigating these other protocols is later labs), so make sure the boxes at the far left of the Frame, Ethernet, IP and TCP information have a plus sign or a right-pointing triangle (which means there is hidden, undisplayed information), and the HTTP line has a minus sign or a down-pointing triangle (which means that all information about the HTTP message is displayed).

(*Note:* You should ignore any HTTP GET and response for favicon.ico. If you see a reference to this file, it is your browser automatically asking the server if it (the server) has a small icon file that should be displayed next to the displayed URL in your browser. We'll ignore references to this pesky file in this lab.).

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 Wireshark 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. When you hand in your assignment, annotate the output so that it's clear where in the output you're getting the information for your answer (e.g., for our classes, we ask that students markup paper copies with a pen, or annotate electronic copies with text in a colored font).

1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?
2. What languages (if any) does your browser indicate that it can accept to the server?
3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?
4. What is the status code returned from the server to your browser?
5. When was the HTML file that you are retrieving last modified at the server?
6. How many bytes of content are being returned to your browser?
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.

In your answer to question 5 above, you might have been surprised to find that the document you just retrieved was last modified within a minute before you downloaded the document. That's because (for this particular file), the gaia.cs.umass.edu server is setting the file's last-modified time to be the current time and is doing so once per minute. Thus, if you wait a minute between accesses, the file will appear to have been recently modified, and hence your browser will download a "new" copy of the document.

Activity 2: The HTTP CONDITIONAL GET/response interaction

Most web browsers perform object caching and thus perform a conditional GET when retrieving an HTTP object. Before performing the steps below, make sure your browser's cache is empty. (To do this under Firefox, select *Tools->Clear Recent History* and check the Cache box, or for Internet Explorer, select *Tools->Internet Options->Delete File;* these actions will remove cached files from your browser's cache.) Now do the following:

1. Start up your web browser, and make sure your browser's cache is cleared, as discussed above.
2. Start up the Wireshark packet sniffer
3. Enter the following URL into your browser http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html Your browser should display a very simple five-line HTML file.
4. Quickly enter the same URL into your browser again (or simply select the refresh button on your browser)
5. Stop Wireshark 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.
6. (*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-2 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one of the author's computers.)

Answer the following questions:

1. 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?
2. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?
3. 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?
4. 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.

Activity 3: Retrieving Long Documents

In our examples thus far, the documents retrieved have been simple and short HTML files. Let's next see what happens when we download a long HTML file. Do the following:

1. Start up your web browser, and make sure your browser's cache is cleared, as discussed above.
2. Start up the Wireshark packet sniffer
3. Enter the following URL into your browserhttp://gaia.cs.umass.edu/wireshark-labs/HTTPwireshark-file3.html

Your browser should display the rather lengthy US Bill of Rights.

4. Stop Wireshark packet capture, and enter "http" in the display-filter-specification window, so that only captured HTTP messages will be displayed.
5. (*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-3 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one of the author's computers.)

In the packet-listing window, you should see your HTTP GET message, followed by a multiple-packet TCP response to your HTTP GET request. This multiple-packet response deserves a bit of explanation. HTTP response message consists of a status line, followed by header lines, followed by a blank line, followed by the entity body. In the case of our HTTP GET, the entity body in the response is the *entire* requested HTML file. In our case here, the HTML file is rather long, and at 4500 bytes is too large to fit in one TCP packet. The single HTTP response message is thus broken into several pieces by TCP, with each piece being contained within a separate TCP segment (see Figure 1.24 in the text). In recent versions of Wireshark, Wireshark indicates each TCP segment as a separate packet, and the fact that the single HTTP response was fragmented across multiple TCP packets is indicated by the "TCP segment of a reassembled PDU" in the Info column of the Wireshark display. Earlier versions of Wireshark used the "Continuation" phrase to indicated that the entire content of an HTTP message was broken across multiple TCP segments.. We stress here that there is no "Continuation" message in HTTP!

Answer the following questions:

1. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill or Rights?
2. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?

3. What is the status code and phrase in the response?

How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?

Activity 4: HTML Documents with Embedded Objects

Now that we've seen how Wireshark displays the captured packet traffic for large HTML files, we can look at what happens when your browser downloads a file with embedded objects, i.e., a file that includes other objects (in the example below, image files) that are stored on another server(s).

Do the following:

1. Start up your web browser, and make sure your browser's cache is cleared, as discussed above.
2. Start up the Wireshark packet sniffer
3. Enter the following URL into your browserhttp://gaia.cs.umass.edu/wireshark-labs/HTTPwireshark-file4.html

Your browser should display a short HTML file with two images. These two images are referenced in the base HTML file. That is, the images themselves are not contained in the HTML; instead the URLs for the images are contained in the downloaded HTML file. As discussed in the textbook, your browser will have to retrieve these logos from the indicated web sites. Our publisher's logo is retrieved from the www.aw-bc.com web site. The image of the cover for our 5th edition (one of our favorite covers) is stored at the manic.cs.umass.edu server.

4. Stop Wireshark packet capture, and enter "http" in the display-filter-specification window, so that only captured HTTP messages will be displayed.
5. (*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-4 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one of the author's computers.) Answer the following questions:

1. How many HTTP GET request messages did your browser send? To which Internet addresses were these GET requests sent?
2. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.

Lab # 04 (b): Hyper Text Transfer Protocol (HTTP) using Packet Tracer

Instructions:

1. Start Packet Tracer using **Realtime** mode. Options -> Preferences o Enable "Show Link Lights" o Disable "Hide Device Label"

2. Configuring the www.mapua.edu Web Server

Add a server.

Global Settings:

1. Change the Display Name to "**Web Server: www.mapua.edu**"
2. Set the Gateway to **172.16.0.1** FastEthernet:
3. Set the IP address to **172.16.0.20**
4. Set the Subnet Mask to **255.255.0.0**

DHCP:

5. Set the Service to **Off** DNS:
6. Set the Service to **Off** HTTP
7. Set the both the HTTP and HTTPS Service to **On**
8. Change the sentence, "<hr> Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open." to "<hr> Welcome to MapuaInstitute's of Technology's public web page!" You may add other information as well. Email:
9. Set the SMTP Service and POP3 Service to **Off**

3. Configuring the www.internal.com Web Server

Add a server.

Global Settings:

1. Change the Display Name to "**Web Server: www.internal.com**"
2. Set the Gateway to **172.16.0.1** FastEthernet:
3. Set the IP address to **172.16.0.30**
4. Set the Subnet Mask to **255.255.0.0** DHCP:

5. Set the Service to **Off** DNS:

6. Set the Service to **Off** HTTP:

7. Change the sentence, "<hr> Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open." to "<hr>This is the corporate internal network!" You may add other information as well.

Activity 1: HTTP Authentication

Finally, let's try visiting a web site that is password-protected and examine the sequence of HTTP message exchanged for such a site. The URL http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html is password protected. The username is "wireshark-students" (without the quotes), and the password is "network" (again, without the quotes). So let's access this "secure" password-protected site. Do the following:

1. Make sure your browser's cache is cleared, as discussed above, and close down your browser. Then, start up your browser

2. Start up the Wireshark packet sniffer

3. Enter the following URL into your browserhttp://gaia.cs.umass.edu/wiresharklabs/protected_pages/HTTP-wireshark-file5.html

Type the requested user name and password into the pop up box.

4. Stop Wireshark 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.

5. (*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-5 packet trace to answer the questions below; see footnote 2. This trace file was gathered while performing the steps above on one of the author's computers.)

6. Now let's examine the Wireshark output. You might want to first read up on HTTP authentication by reviewing the easy-to-read material on "HTTP Access Authentication Framework" at http://frontier.userland.com/stories/storyReader$2159 Answer the following questions:

7. What is the server's response (status code and phrase) in response to the initial HTTP GET message from your browser?

8. When your browser's sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?

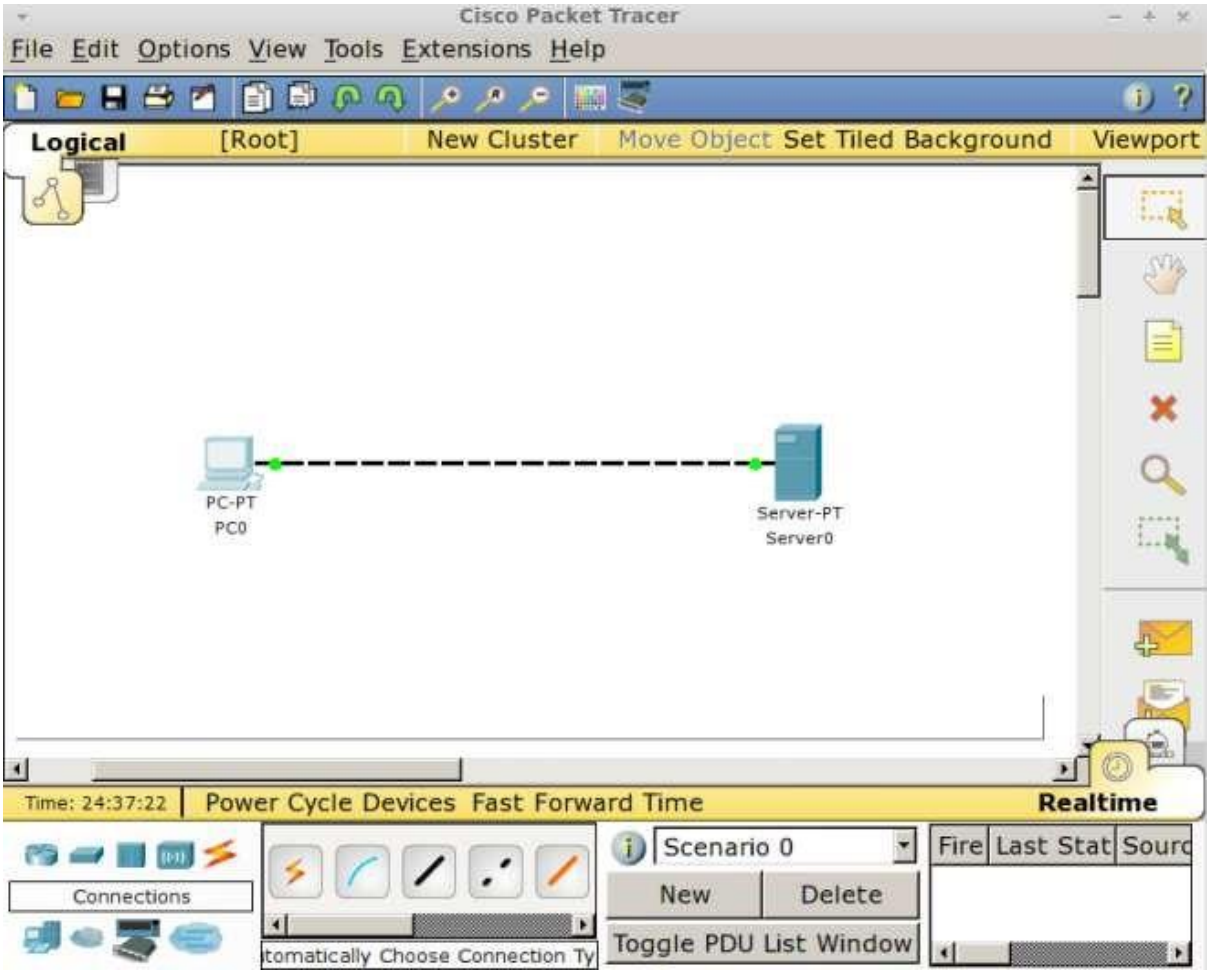
The username (wireshark-students) and password (network) that you entered are encoded in the string of characters (d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=) following the "Authorization: Basic" header in the client's HTTP GET message. While it may appear that your username and password are encrypted, they are simply encoded in a format known as Base64 format. The username and password are *not*encrypted! To see this, go to http://www.motobit.com/util/base64decoder-encoder.asp and enter the base64-encoded string d2lyZXNoYXJrLXN0dWRlbnRz and decode. *Voila!* You have translated from Base64 encoding to ASCII encoding, and thus should see your username! To view the password, enter the remainder of the string Om5ldHdvcms= and press decode. Since anyone can download a tool like Wireshark and sniff packets (not just their own) passing by their network adaptor, and anyone can translate from Base64 to ASCII (you just did it!), it should be clear to you that simple passwords on WWW sites are not secure unless additional measures are taken.

Activity 2:

Provide web services in the said topology using this information:

1. Set the server ip into "192.168.78.1"

2. Set the DNS "Name" into whatever you wish, mine is "http://www.hesemeleh.com"


Cisco Packet Tracer
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT
PC0
Server-PT
Server0
Time: 24:37:22
Power Cycle Devices
Fast Forward Time
Realtime
Connections
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Stat
Sourc

Lab # 04 (c): Domain Name Server (DNS) using Wireshark

Statement Purpose:

The purpose of this lab is to understand basic concept of DNS and examine DNS Query resolution closely.

Activity outcomes:

Students will be able to better understand the working of DNS

Instructor Note:

In this Wireshark lab, you'll captures some DNS packets using Wireshark and make some observations on them. The Second part of the lab is based on Packet Tracer based implementation of DNS where the students will be able to configure and simulate basic query/response messages of DNS.

Introduction

Domain Name System (DNS):

The DNS is the Internet's system for mapping alphabetic domain names to numeric Internet Protocol (IP) addresses like a phone book maps a person's name to a phone number.
DNS provides the following functions:

1. Server: Configures DNS servers and default domain names for a network resource (server, host, website, etc.).
2. Proxy: The server acts as a DNS proxy and provides proxy service for the connected PCs and other clients. Besides, the proxy server can also choose different DNS servers according to domain names.
3. Resolver: Sets retry times and timeout for DNS service.
4. Cache: Stores DNS mappings to cache to speed up query. You can create, edit and delete DNS mappings.
5. NBT Cache: Displays NBT cache information.

Some basic types of DNS records are:

nslookup:

nslookup (from name server lookup) is a network administration command-line tool for **querying the DNS to obtain the mapping between domain name and IP address, or other DNS records**. To run *nslookup* in Linux/Unix, you just type the *nslookup* command on the command line. To run it in Windows, open the Command Prompt and run *nslookup* on the command line. In it is most basic operation, *nslookup* tool allows the host running the tool to query any specified DNS server for a DNS record. 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.
In this lab we will look into basic commands of nslookup.

1. nslookup www.google.com

In other words, this command is like **"please send me the IP address for the host www.google.com".**
As a response of this command, we will get two pieces of information i.e.

2. The name and IP address of the DNS server that provides the answer.
3. The answer itself, which is the host name and IP address of w ww.google.com.

4. 192.168.1.5om

In this command we will provide "type" of record we want to fetch and the "domain name".

In words, the query is saying, **"please send me the host names of the authoritative DNS for mit.edu"** (When –type is not used, nslookup uses default)

Non-authoritative name servers **do not contain original source files of domain's zone**. They have a

```
C:\>nslookup www.google.com
Server:  UnKnown
Address:  192.168.43.103

Non-authoritative answer:
Name:    www.google.com
Addresses:  2a00:1450:4019:80c::2004
          142.250.181.100
```

```
C:\>nslookup -type=NS google.com
Server:  UnKnown
Address:  192.168.43.103

Non-authoritative answer:
google.com      nameserver = ns2.google.com
google.com      nameserver = ns4.google.com
google.com      nameserver = ns1.google.com
google.com      nameserver = ns3.google.com
```

cache file for the domains that is constructed from all the DNS lookups done previously.

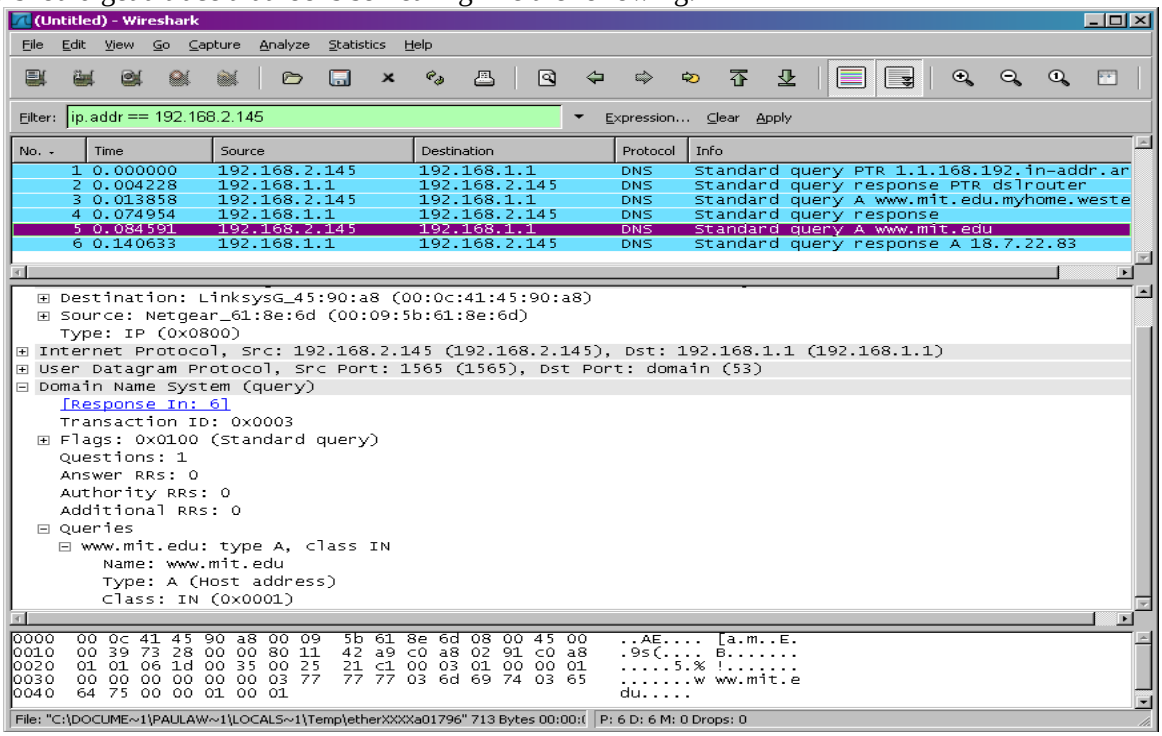
Lab Activities:

Activity 1:

Now let's play with *nslookup*5.

1. Start packet capture.
2. Do an *nslookup* on www.mit.edu
3. Stop packet capture.

You should get a trace that looks something like the following:



We see from the above screenshot that *nslookup* actually sent three DNS queries and received three DNS responses. For the purpose of this assignment, in answering the following questions, ignore the first two

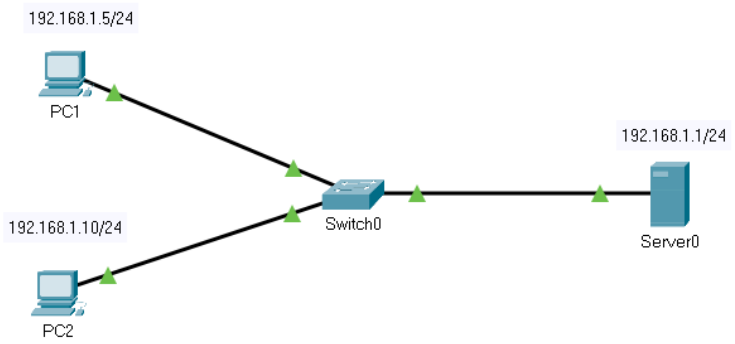
sets of queries/responses, as they are specific to *nslookup* and are not normally generated by standard Internet applications. You should instead focus on the last query and response messages.

1. What is the destination port for the DNS query message? What is the source port of DNS response message?
2. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?
3. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
4. Examine the DNS response message. How many "answers" are provided? What do each of these answers contain?

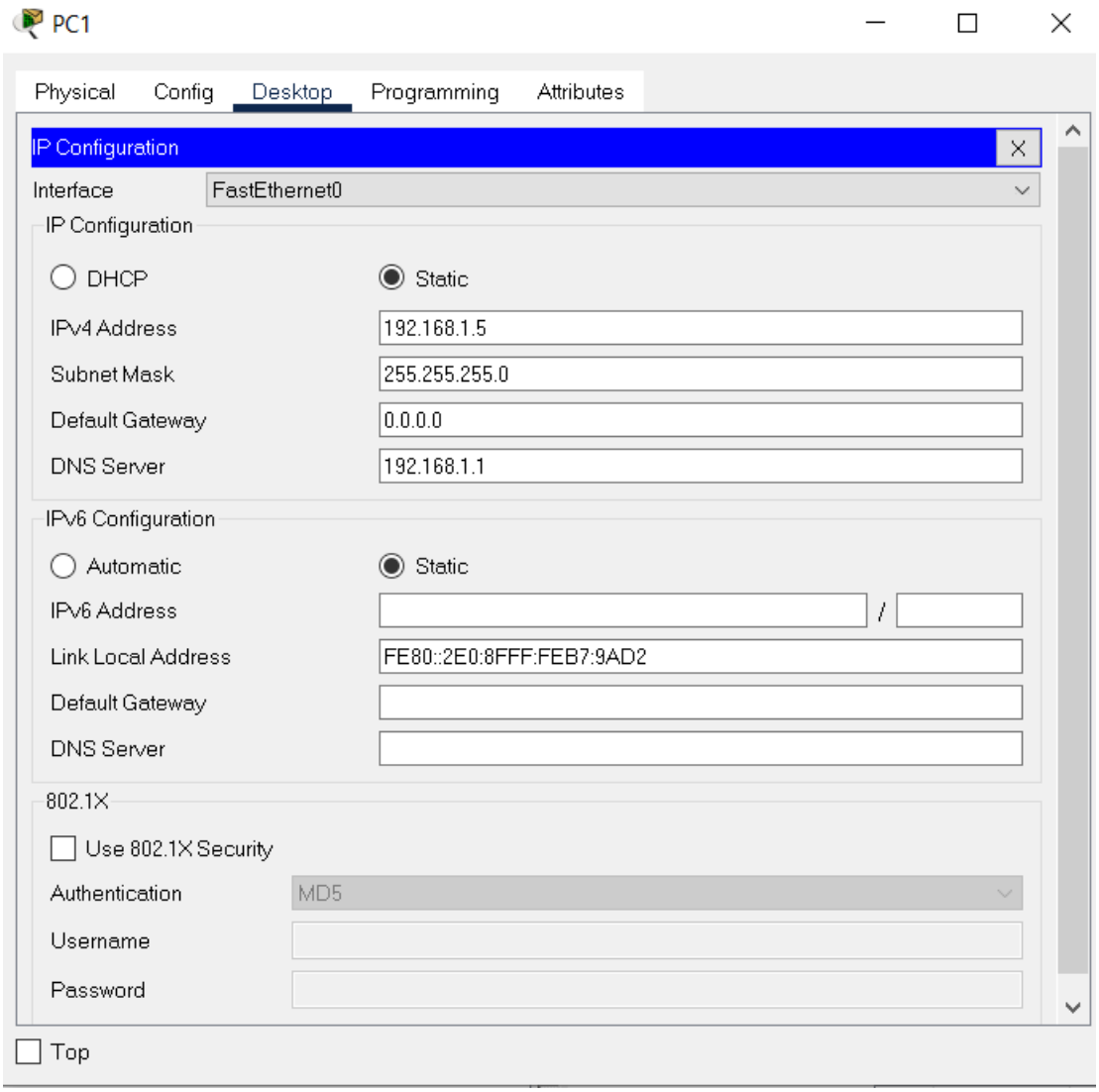
Lab # 04 (d): Domain Name Server (DNS) using Packet Tracer

1. Build and configure the network topology using the following configuration.

    1. DNS_Server IP address: 192.168.1.1/24
    2. PC1 IP address: 192.168.1.5/24, **DNS Server**: 192.168.1.1
    3. PC2 IP address: 192.168.1.10/24, **DNS Server**: 192.168.1.1



Configuration of PC1 is provided below as an example.



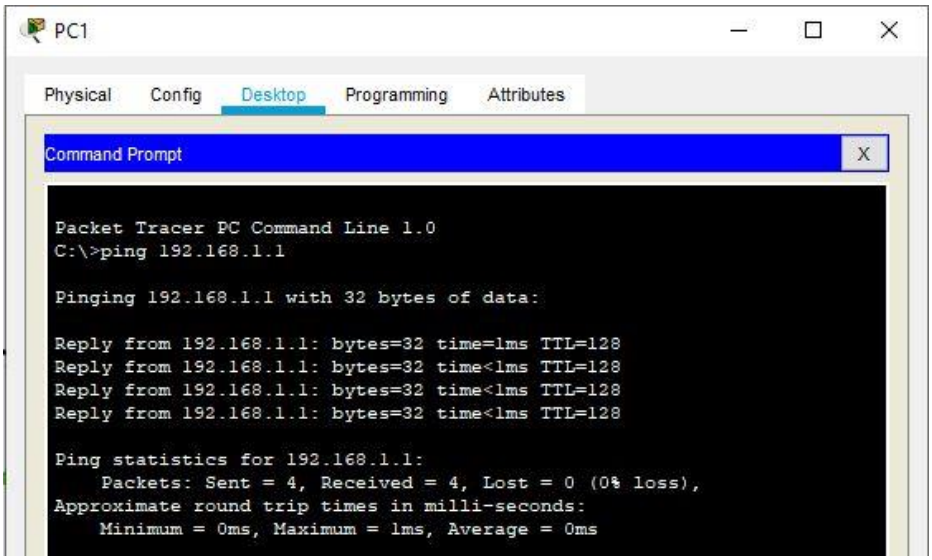
ACTIVITY 1:

PING THE SERVER USING IP:

2. Go to the desktop tab of any PC and open command prompt.

3. Write a command “ping 192.168.1.1”, we are pinging with 32 bytes of data.

4. Try to understand the response we are getting in ping statistics.



Ping statistics shows that 4 packets are being sent and received successfully.

ACTIVITY 2:

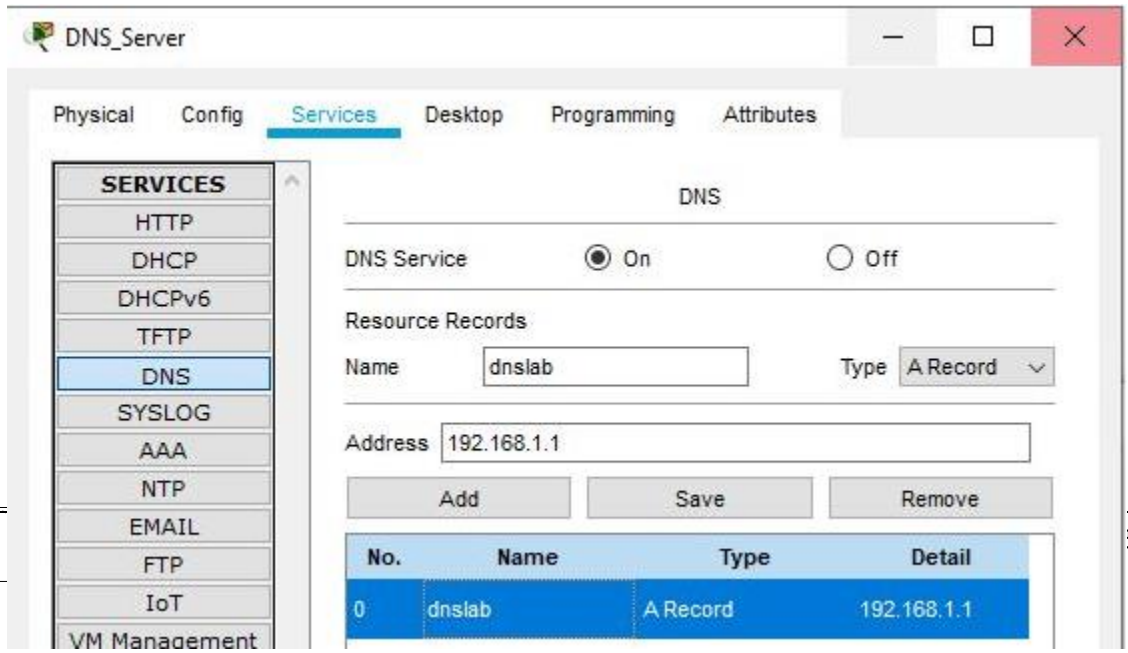
ADD A RECORD ON DNS SERVER.

Now we want to ping DNS server using name, for this purpose we need to add an address record.

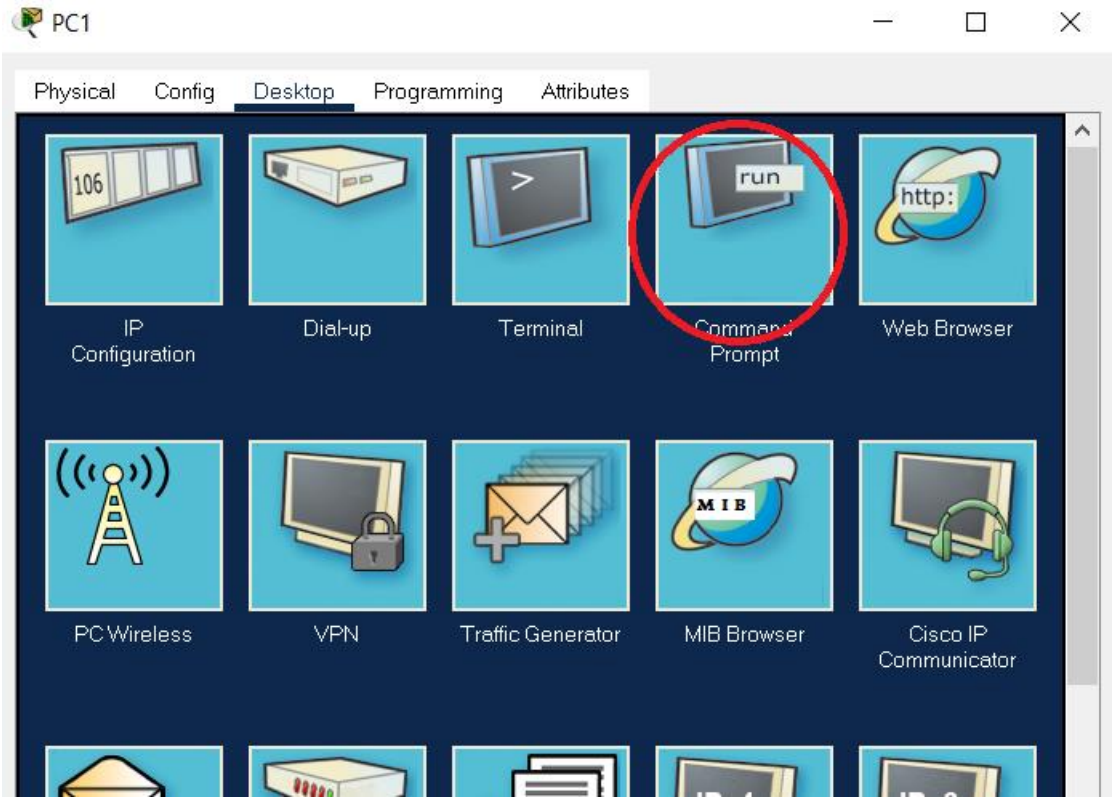
5. Go to the services tab of DNS_Server and select DNS from the SERVICES menu. Make sure the DNS service is **ON**.

6. Add a new address record (Type: A Record)

    1. Ser **Name** to dnslab and **Address** to the DNS_Server IP Address i.e. 192.168.1.1



7. Go to the command prompt of any PC.



8. Enter the command: *ping dnslab* and observe the *output* of the command.

    1. Note: The name ***dnslab*** is resolved to the IP address **192.168.1.1** by consulting the DNS server configured during IP address configuration of the PC. You can observe this communication if the ping command is initiated while in **Simulation** mode instead of **Realtime** mode.

```
C:\>ping dnslab

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=128
Reply from 192.168.1.1: bytes=32 time<1ms TTL=128
Reply from 192.168.1.1: bytes=32 time=1ms TTL=128
Reply from 192.168.1.1: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

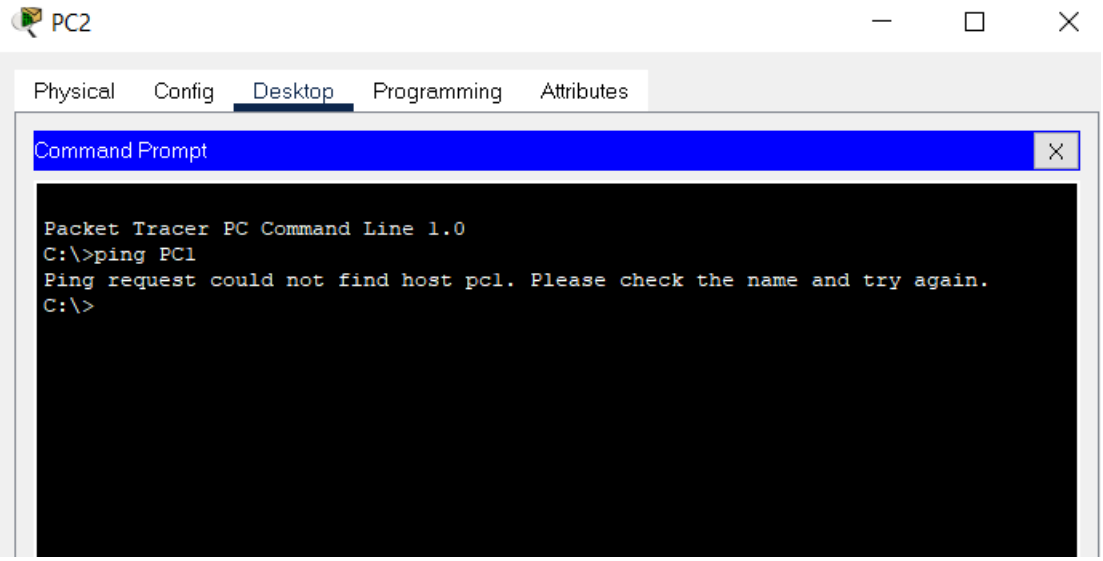
Ping statistics shows that 4 packets are being sent and received successfully.

ACTIVITY: 3

PING A PC BY NAME:

Name PCs as PC1 and PC2 respectively. Ping one pc from another using the name.

9. Initially when we execute the command "*ping PC1*" on PC2, we will get the response **"Ping request cannot find the host pc1. Please check the name and try again".** The reason behind this is, it doesn't know what exactly is PC1 as there is no record on the DNS server for PC1.

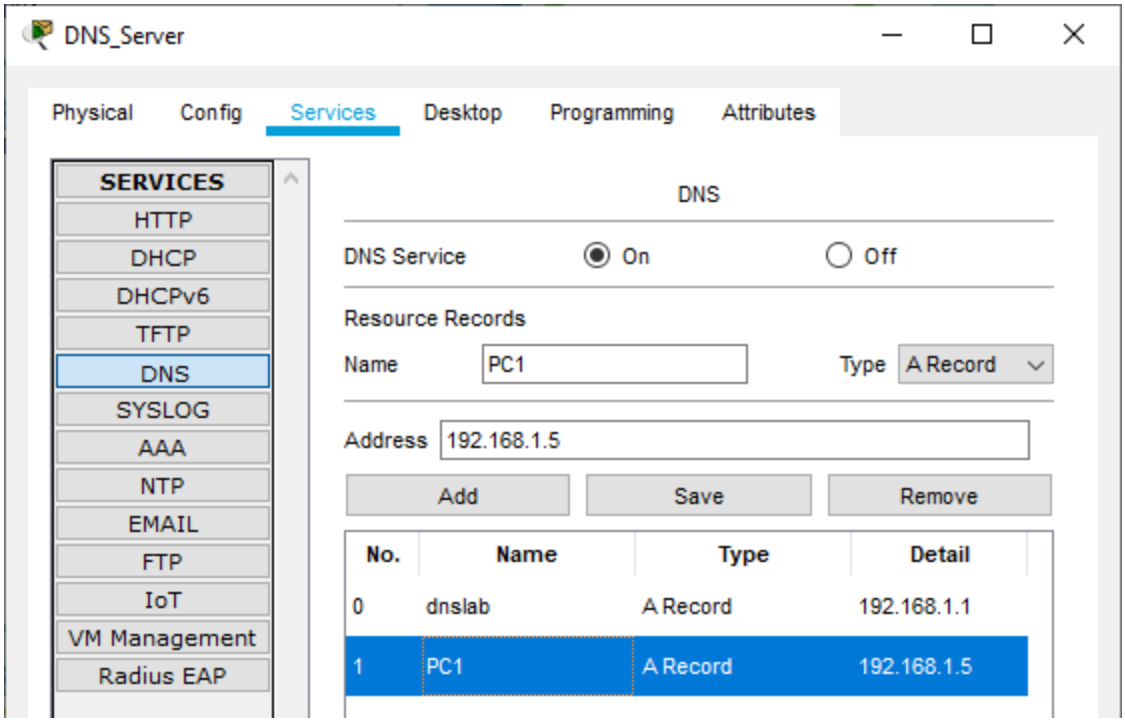


10. To add the record for PC1, repeat the steps of ACTIVITY 2 with the following setting:

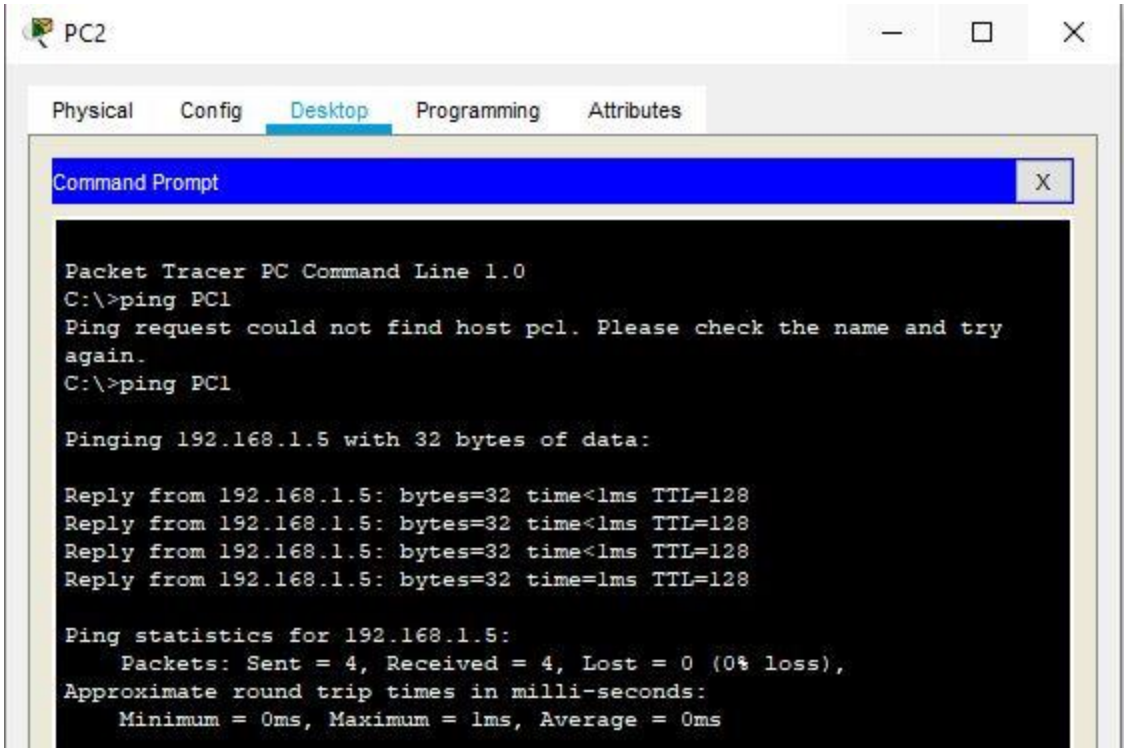
Name: PC1

Address: 192.168.1.5

Type: A Record



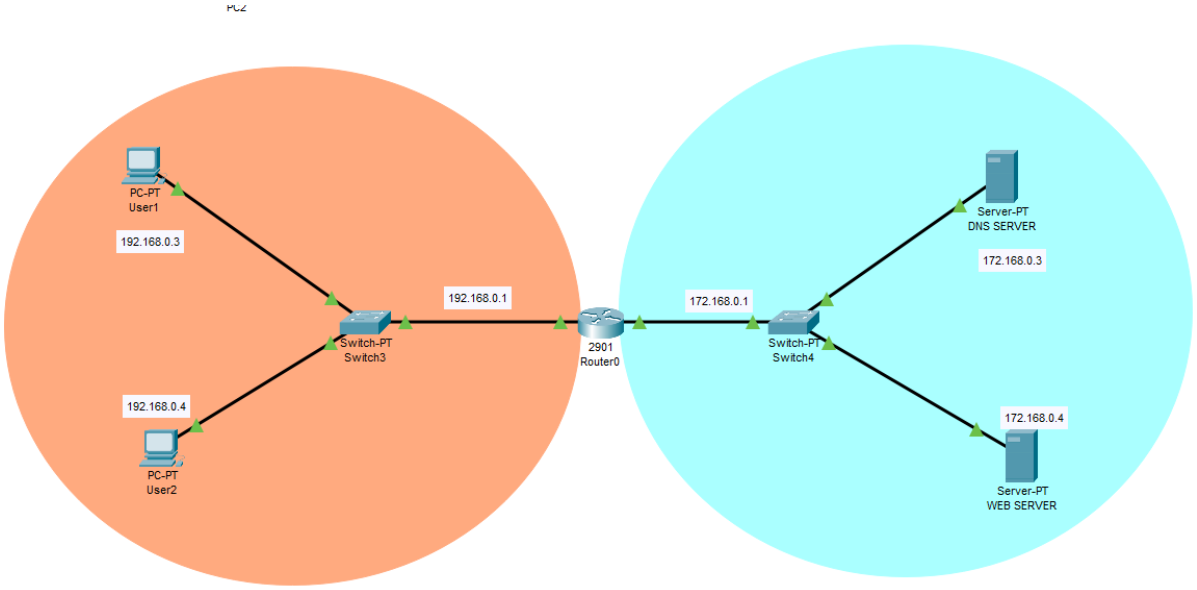
11. Now go to command prompt of PC2 and execute the command "*ping PC1*" again.



You can observe the name PC1 is now resolved to IP address of the PC and that statistics shows that 4 packets are being sent and received successfully.

ACTIVITY : 4

DNS AND HTTP SERVER:



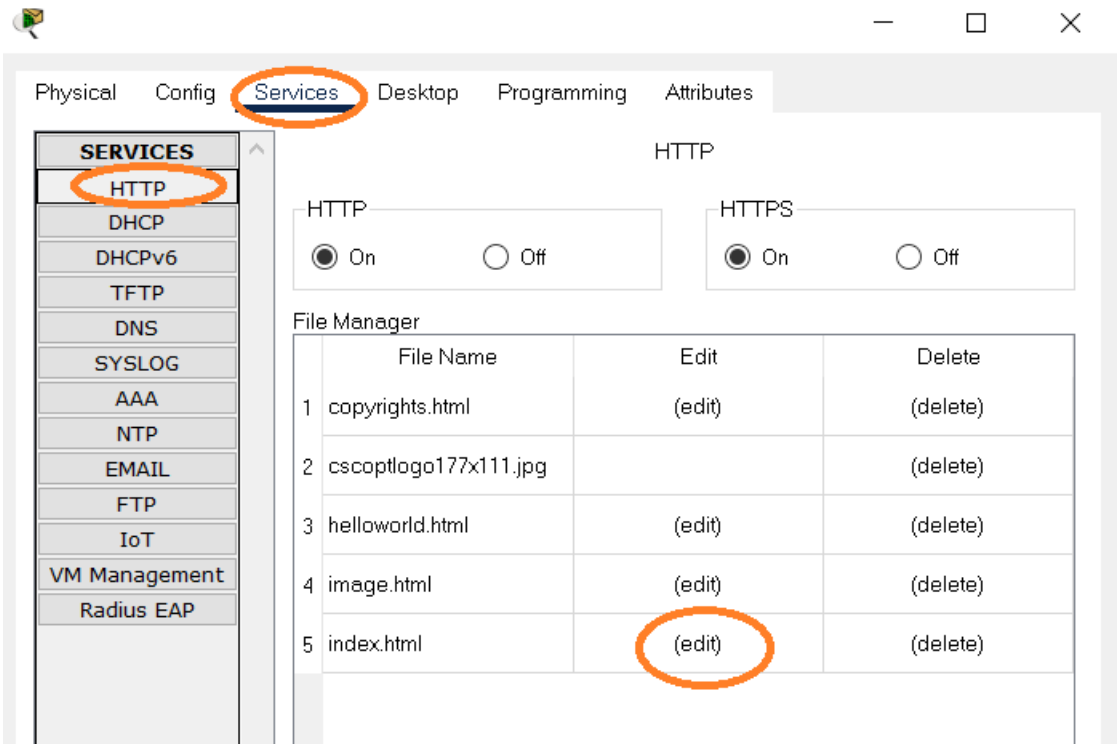
12. Connect 2 PCs to the router.

13. Also connect 2 servers i.e. DNS server AND HTTP server.

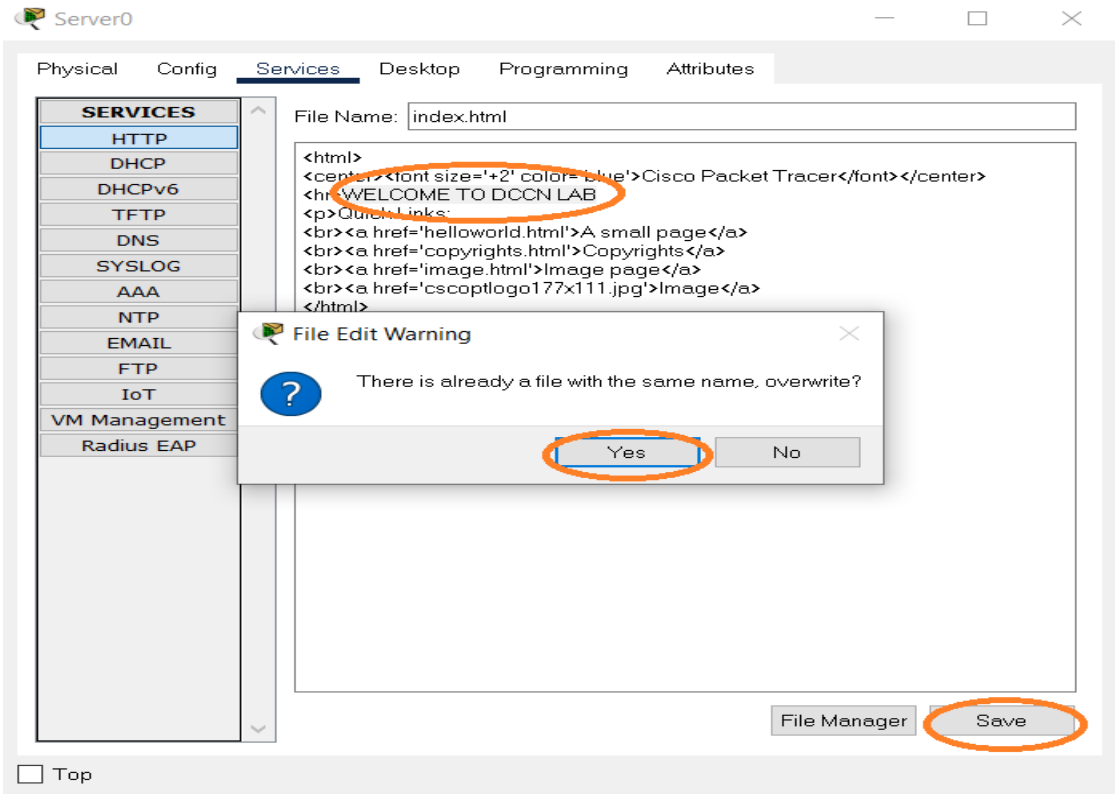
14. Configure DNS server

15. Go to services of web server (http server) and set the http **ON**

    1. In order to make sure that we are receiving response from the desired web server, we will make some changes in the **index.html** page of the WEB SERVER.

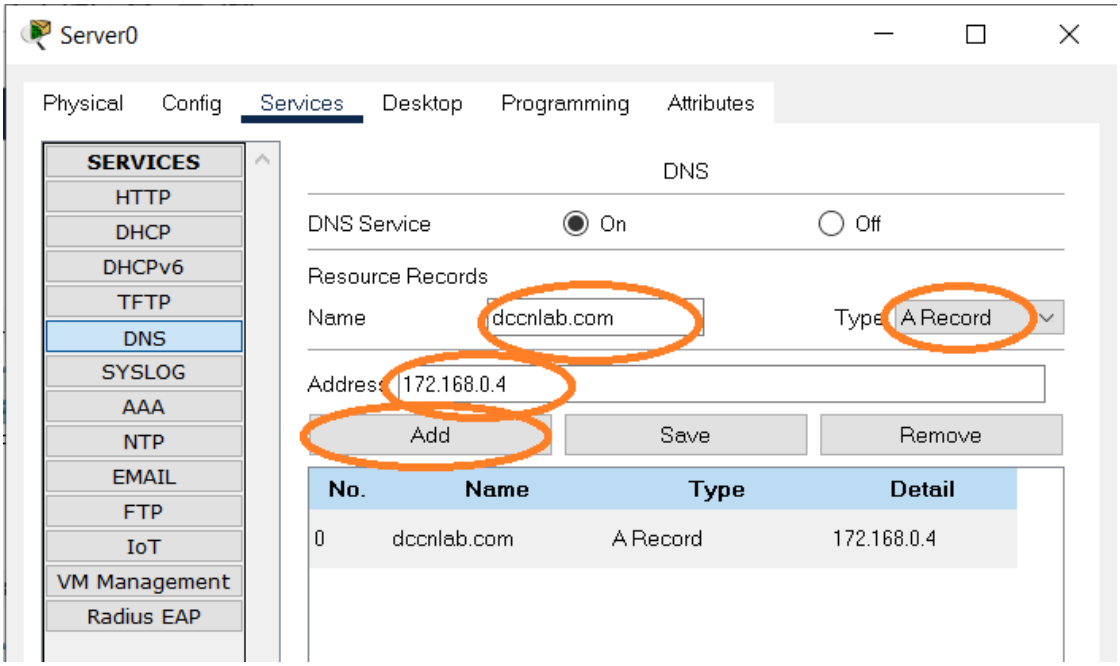


    2. Change the Greetings line form "***Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open***." to "***WELCOME TO DCCN LAB***". You may add any line of your choice.



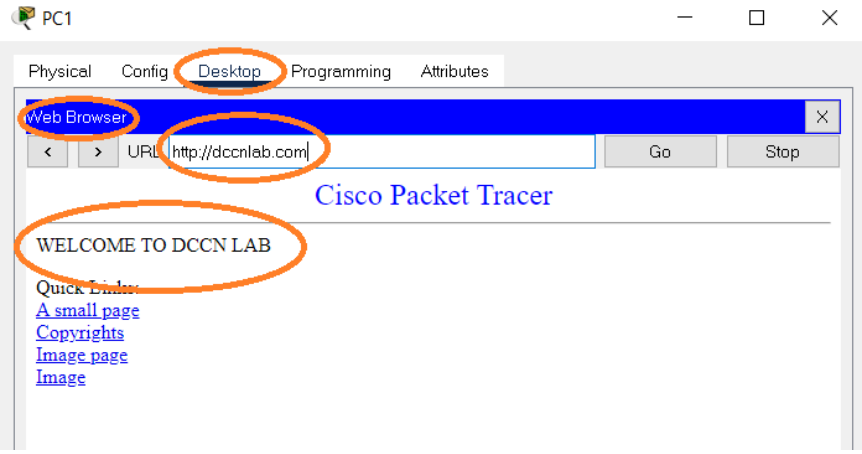
16. Set the Resource record for the Web Server in the DNS server by following steps of ACTIVITY 2: with following configuration.

    1. **Name:** dccnlab.com (set a url for the web resource)
    2. **Address:** 172.168.0.4 (IP address of the web server)
    3. **Type:** A Record



17. Open the web browser on a PC and type *dccnlab.com* in the URL field.

    1. Note the the url is first resolved into the IP address of the web server using DNS before the actual HTTP communication takes place. You can observe this DNS traffic while in **Simulation** mode.



DNS QUERY AND RESPONSE MESSAGES.

To understand the DNS query resolution process, we need to observe the packets in Simulation Mode. Follow the steps given bellow to set up simmulation envirement.

Note: Use the network topology of ACTIVITY1-to-ACTIVITY 3 for this activity.

Step 1: Select Simmulation mode.

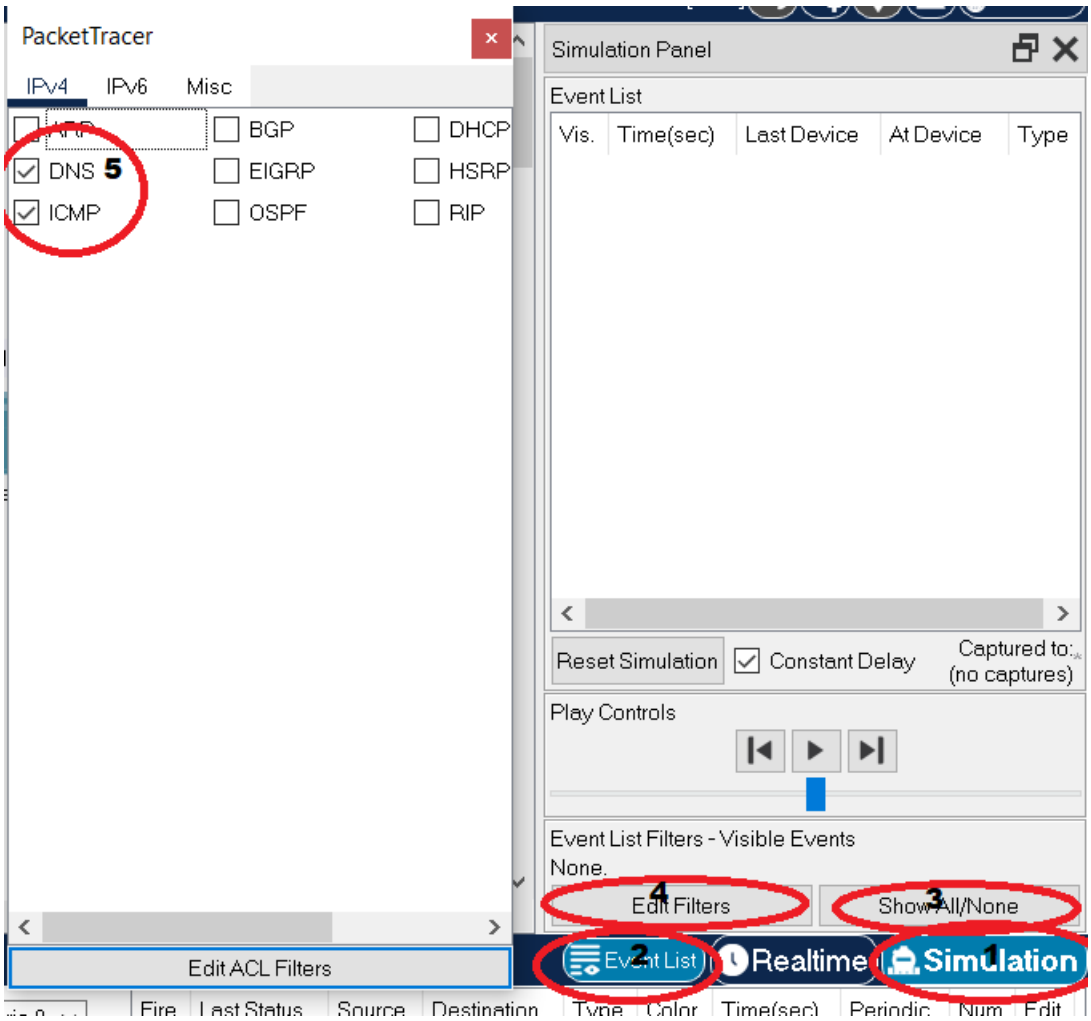
Step 2: Open Event List

Step 3: Clear all event list filters by clicking “Show All/None” button.

Step 4: Click “Edit Filters” button to open “Edit ECL Filters” window.

Step 5: Tick the DNS and ICMP checkboxes.

Note: By following these steps your simmulation environment is set to simmulate only ICMP (ping) and DNS packets.

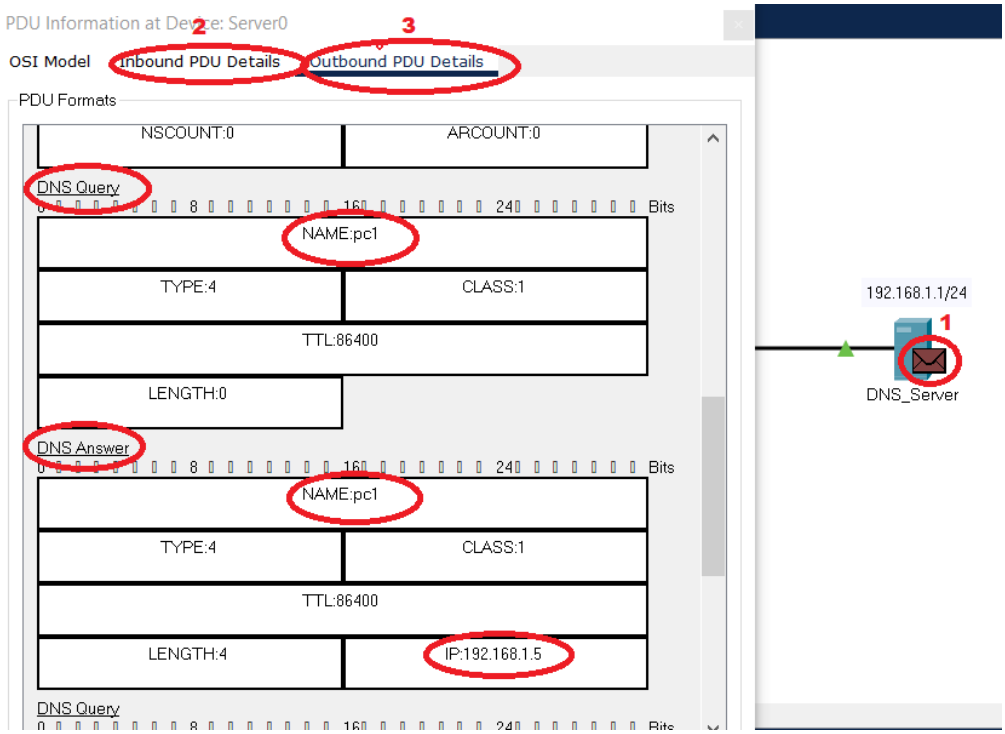


1. ping PC1 from PC2 as you did in ACTIVITY 3, while in summulation mode. You will see a DNS packet on PC2.

2. Step forward the simmulation until the packet reached the DNS_Server.

3. Click on the packet and check the **Inbound** (query received by the server), and **Outbound PDU details** (response by the server).

4. Observe the DNS Query and DNS Anwer Header fields.



1 Repeat the Activity 4 of Packet tracer your own web and domain settings and Network topologies.

2 Repeat the experiments of *nslookup* with two different domain sites and take a note of the following information:

1. 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?
2. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
3. Examine the DNS response message. How many "answers" are provided? What does each of these answers contain?

Lab # 05 (a): Simple Message Transfer Protocol (SMTP) using Wireshark and Packet Tracer

Statement Purpose:

The purpose of this lab is to study the steps involved in email communication. We will use Simple Mail Transfer Protocol (SMTP) to send mail from host to destination address. SMTP protocol is used for sending mail and POP3 is used for receiving. Every server provides the services of SMTP to clients by just enabling feature and doing basic configurations. We will perform activities both on packet tracer and Wireshark.

Activity outcomes:

Students will be able to better understand the working of electronic mails.

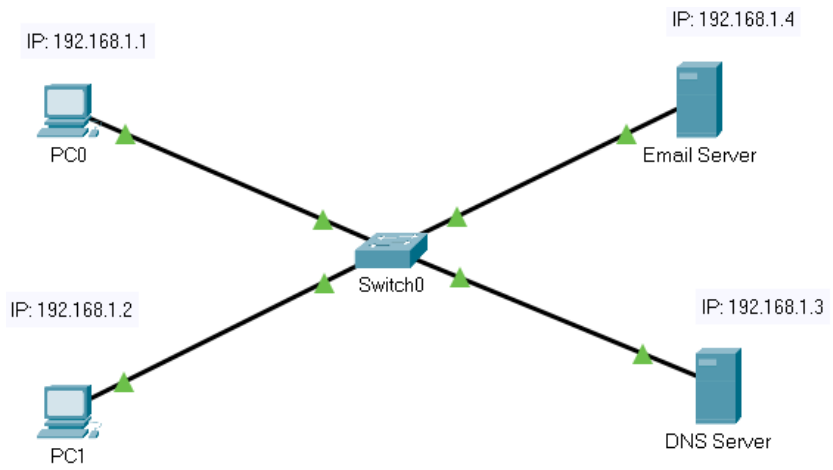
Introduction

SMTP stands for Simple Mail Transfer Protocol and its primary purpose is to handle the sending, receiving, and relaying of email. An email server, such as Gmail stores and sends email messages to email clients on request. We often send and receive emails on our mobile devices or computers. Have you ever imagined how this happens? Well, whenever you compose and send an email to another person, the message you send first goes to a mail server. It's the mail server which then sends the email when it is requested from the email client (e.g., Gmail App) of the recipient's device.

Lab Activities Using Packet Tracer:

Activity 1: Configuration of the Simulation Environment

Step 1: Open packet tracer and build the network topology.



Step 2: Configure the PCs, EMail Server and DNS Server.

Mail Server IP address: 192.168.1.4/24, DNS Server 192.168.1.3

PC0 IP address: 192.168.1.1/24, DNS Server: 192.168.1.3

PC1 IP address: 192.168.1.2/24, DNS Server: 192.168.1.3

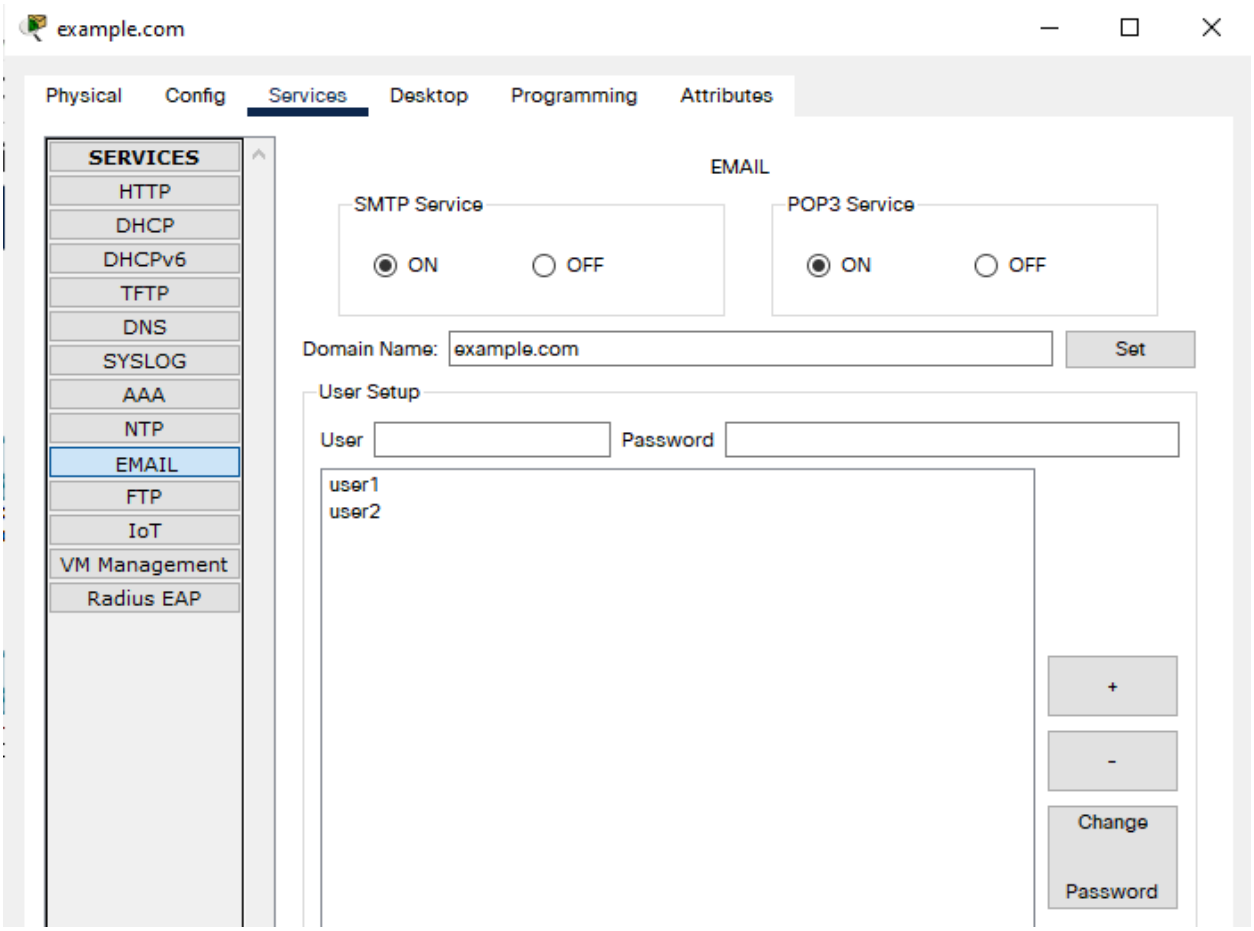
DNS server IP address: 192.168.1.3/24

Step 3: Now configure mail clients on the PCs and mail service on the Email Server.

SERVER

First, we'll configure the email server. To do this, click on the server, then click Services tab, pick email server from the menu. Make sure that both SMTP and POP3 services are on. Provide the Domain name of the server then click on Set. In this example we've used the name 'example.com'. Proceed and add users and provide their passwords. We have two email clients(users) with usernames 'user1' and 'user2' with a common password '123'.

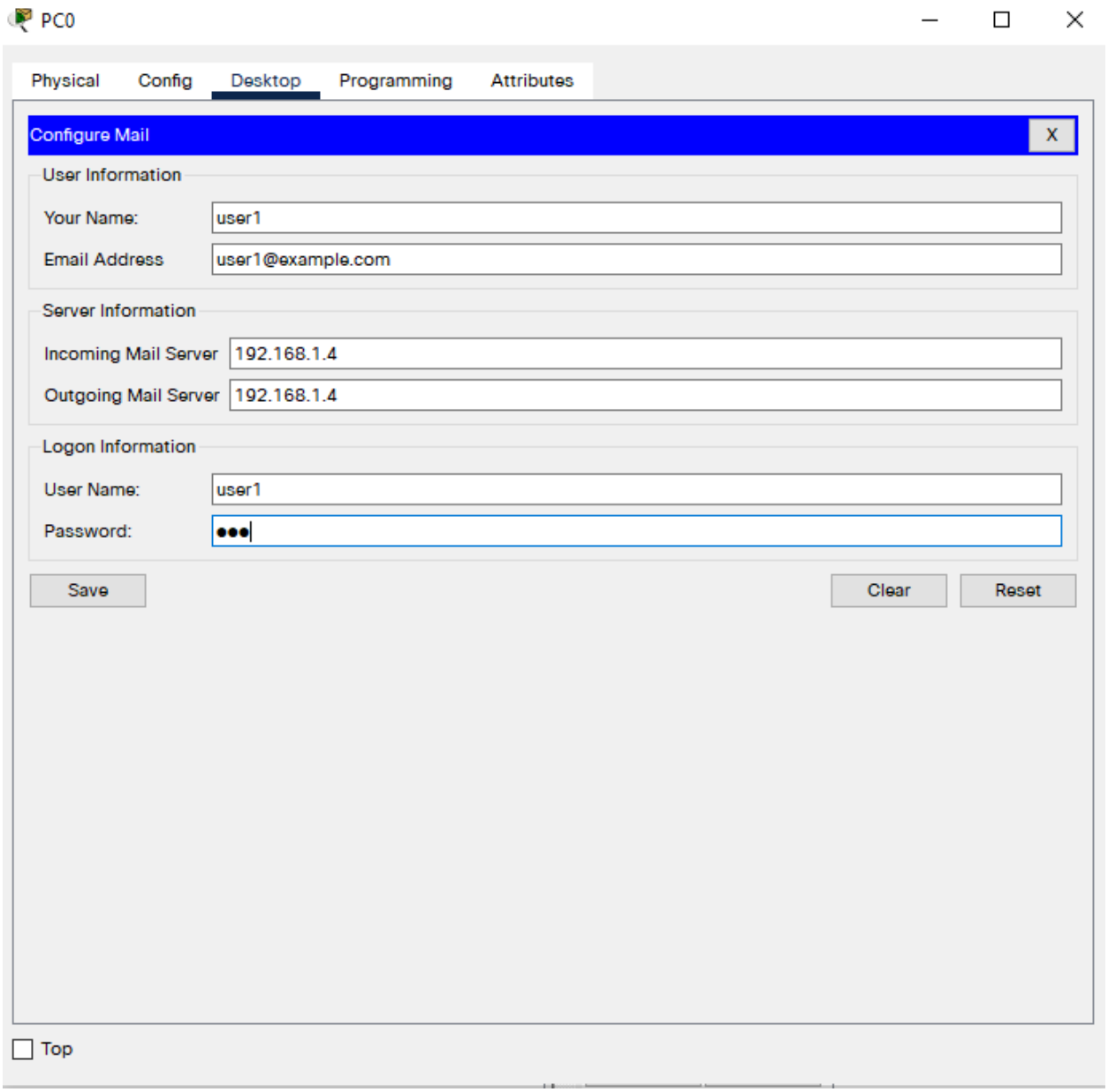
After entering a username and password, click on Add(+) to add the user to the server. You can optionally remove a user by clicking on Remove (-). You can change a user's password by clicking on change password.



Mail Clients

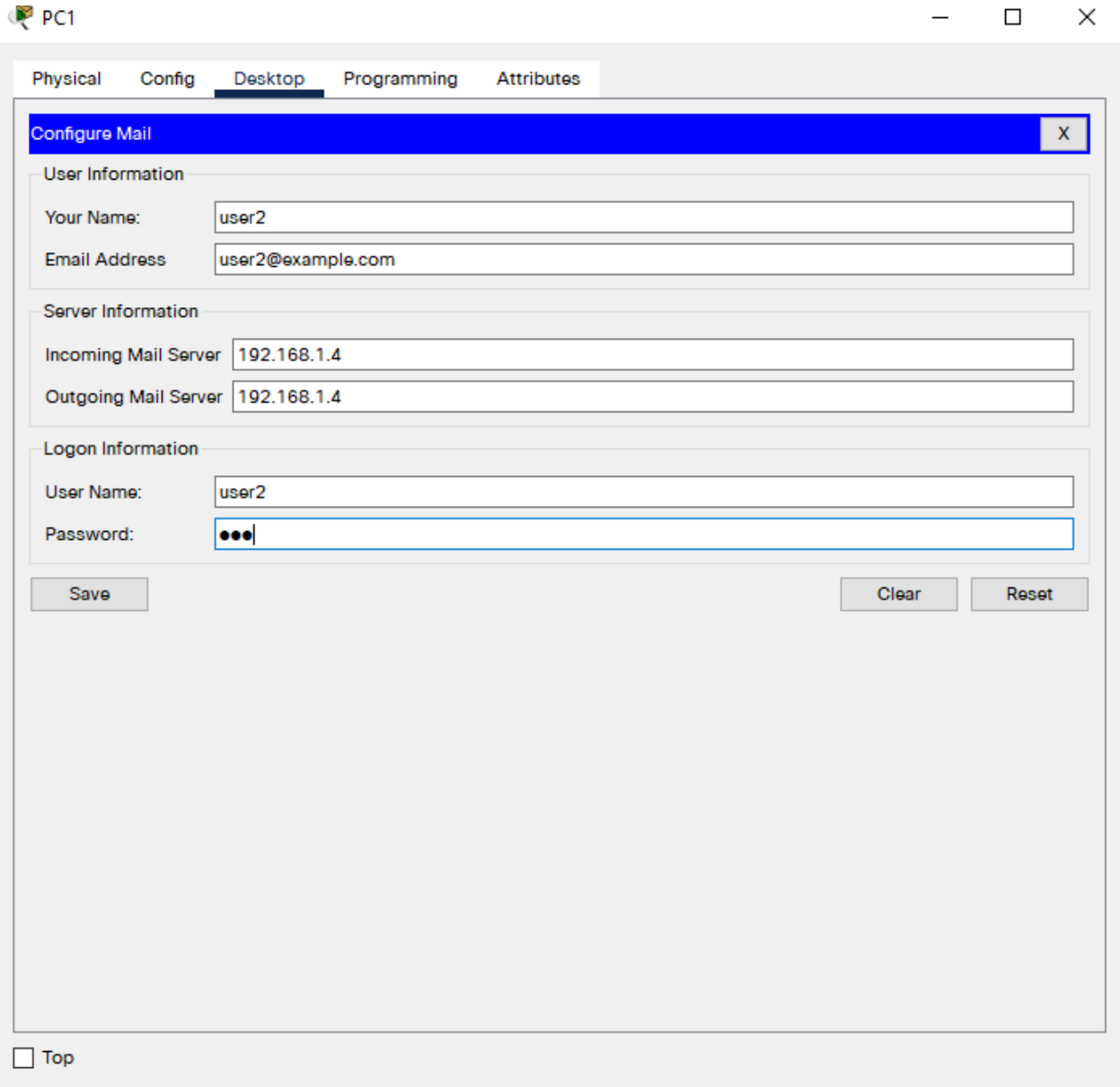
PC0:

Click on PC0. Go to its Desktop tab, and click on Email. Configure the email client by filling in the user, server and login information. Save the configuration. You can used domain name (example.com in our example) instead of IP address of the incoming and outgoing mail servers.

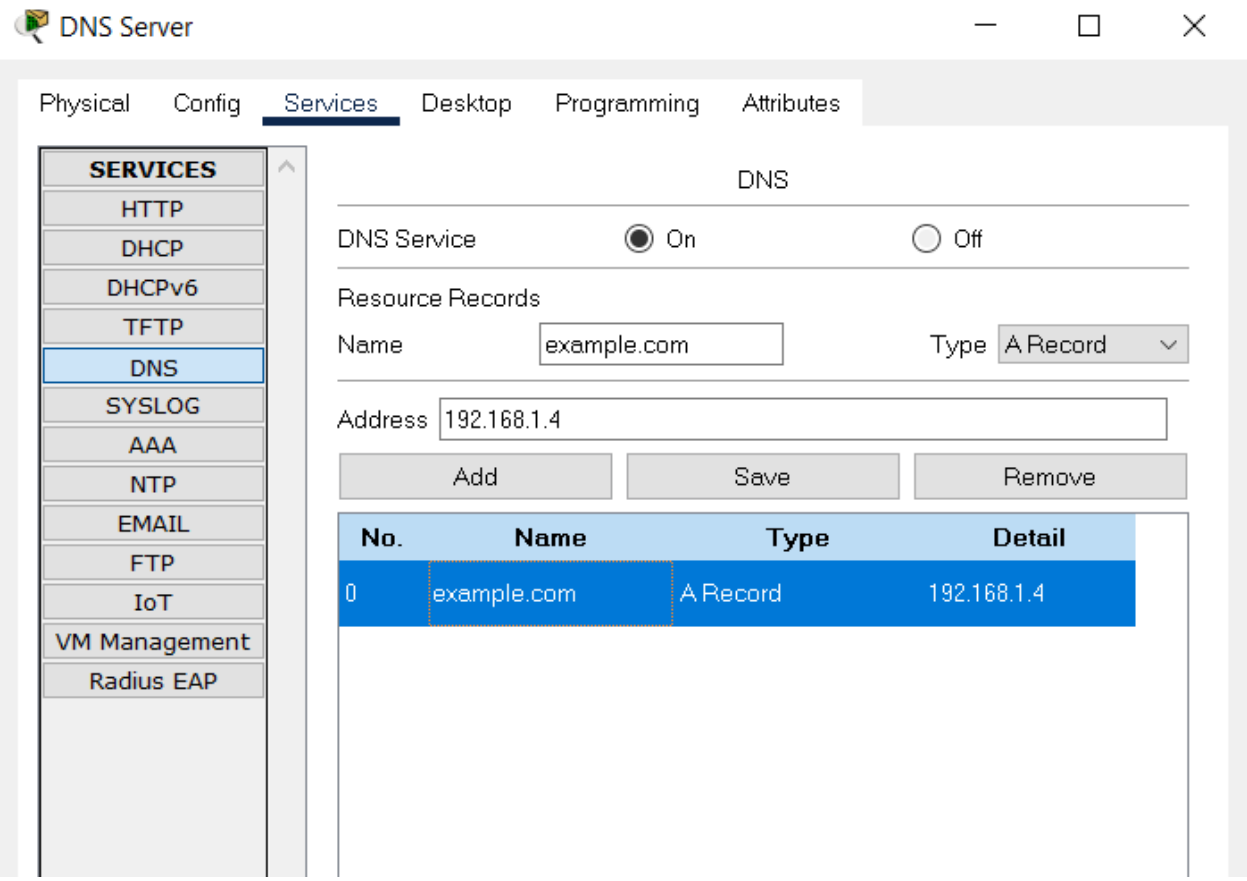


PC1:

Configure PC1for second user in the same way as you have configured PC0.

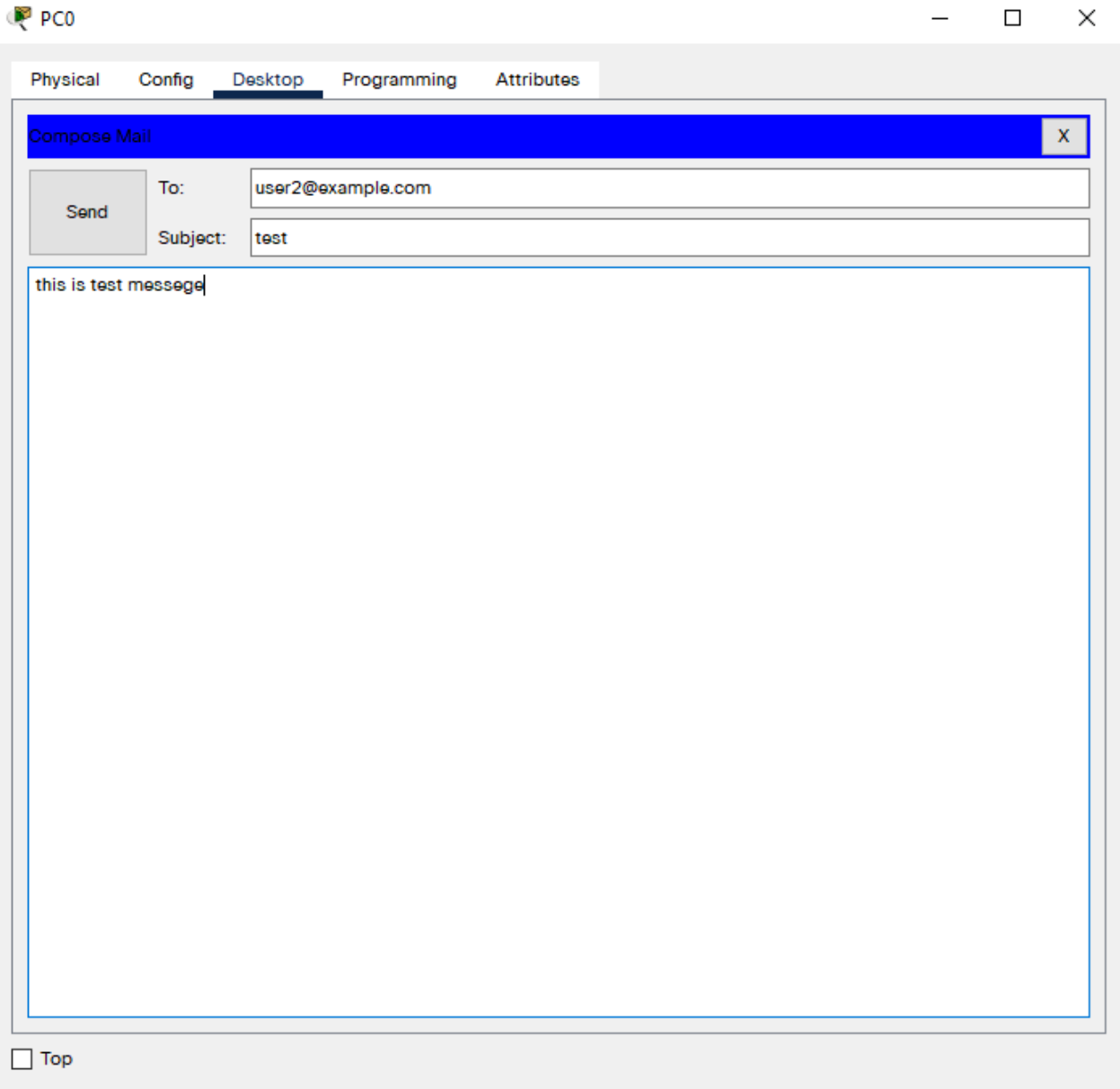


As the emails are sent using domains names of the email server of the receiver, we need a DNS server configured to resolve this domain name (plus other domain names if any). let's configure a DNS server. Click DNS server, click Services tab, then pick DNS. Turn the service ON. Add the address record for the mail server. You can refer to the DNS entry below.

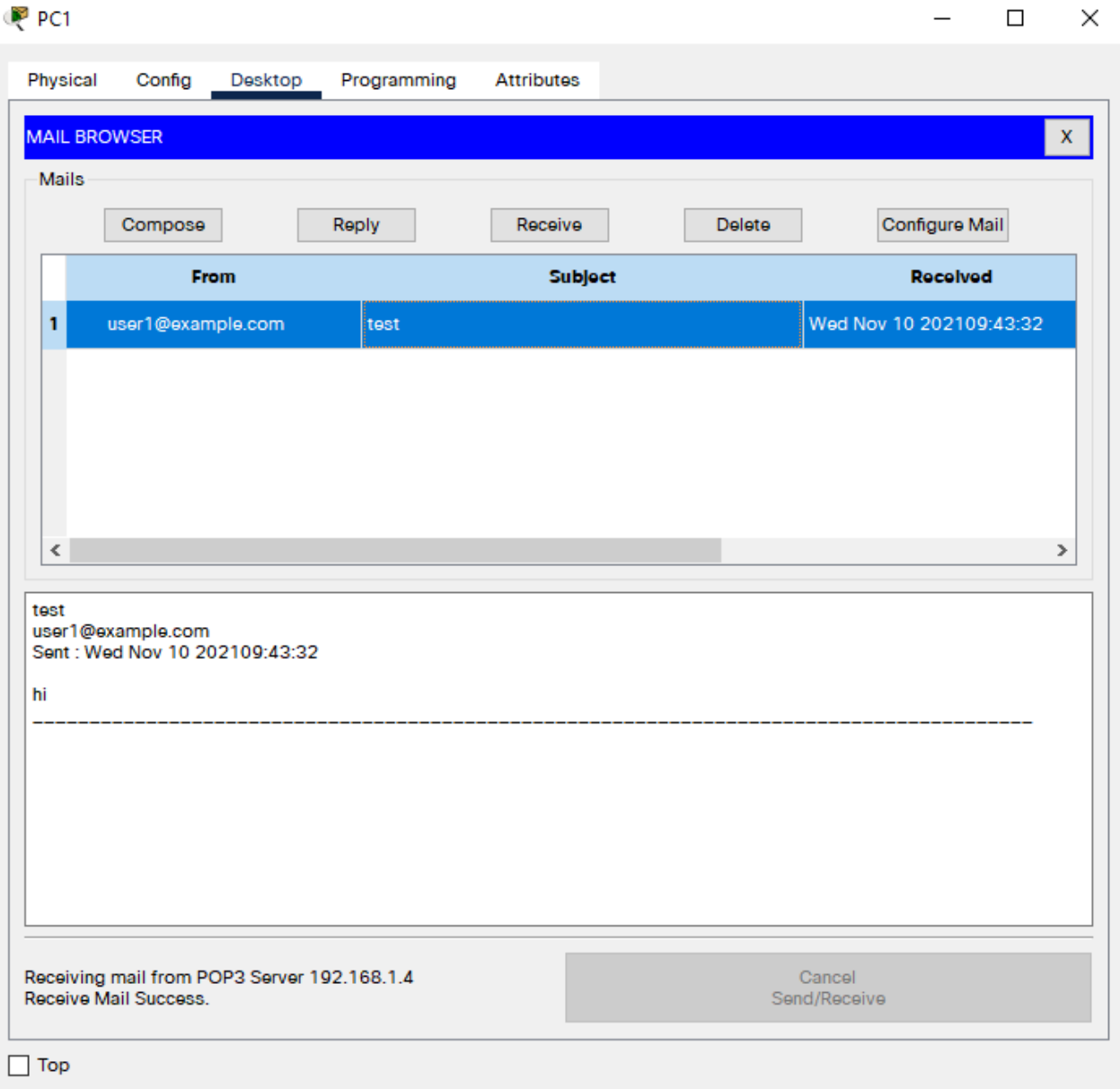


Lab Activity 1: Send and receive emails

Step 1: Test the email service. Go to PC0 email client, compose an email and send it to the email address of other user (user2@example.com) configured on PC1.



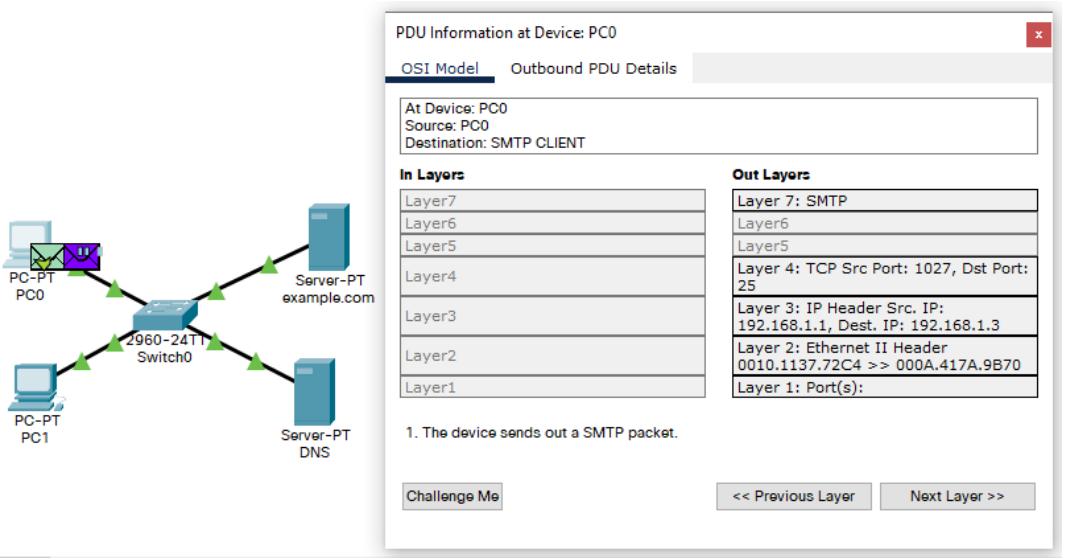
Step 1: Try to see whether the email from PC0 is received on PC1. On the email client of PC1, click on Receive.



If everything is well configured, the email from PC0 will be well received on PC1.

Lab Activity 2: Simulate the SMTP and POP3 Messages

Repeat the previous activity of sending/ receiving emails while in simulation mode.



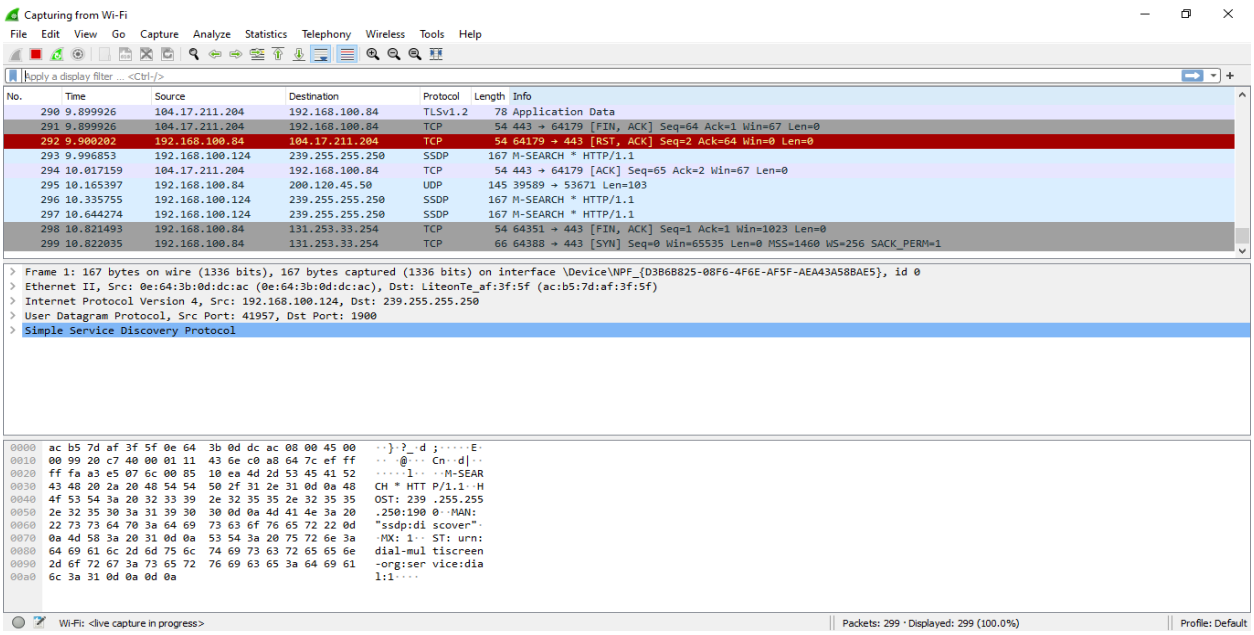
Observe the SMTP communication between sending node (PC0) and the Email server after sending the email.

Observe the POP3 communication between the email Server and the receiving node (PC1) when you click the Receive button on the receiving user agent.

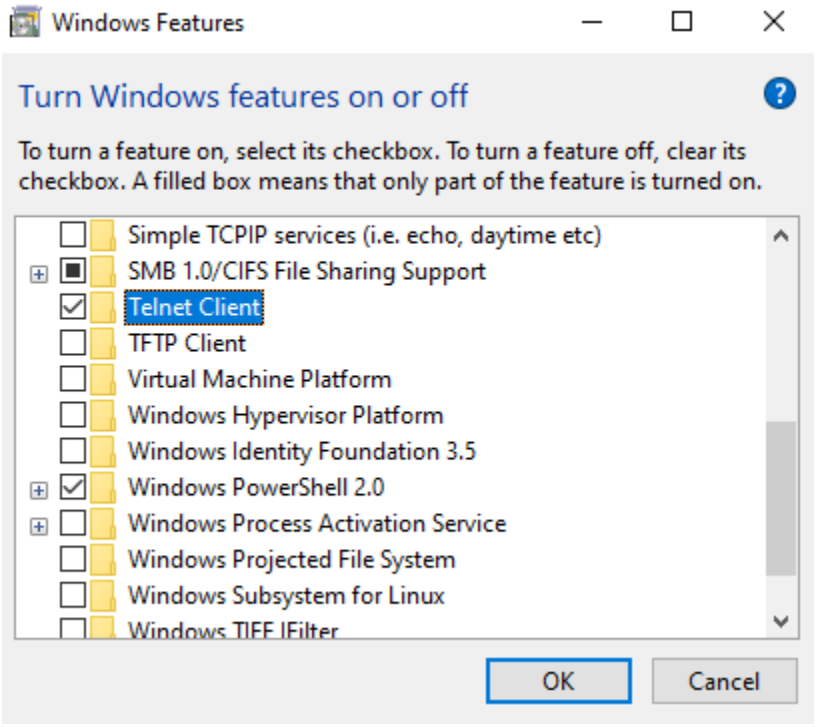
Lab Activities Using Wireshark:

ACTVITIY 1: Capture SMTP Traffic:

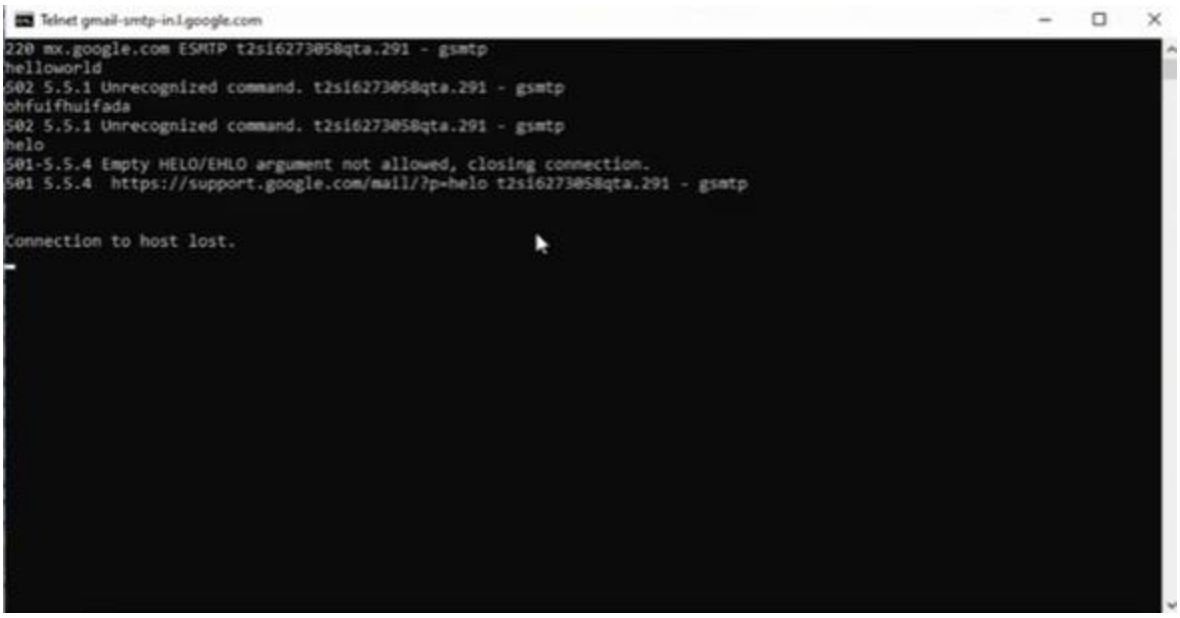
Step.1: Start Wireshark Capture.



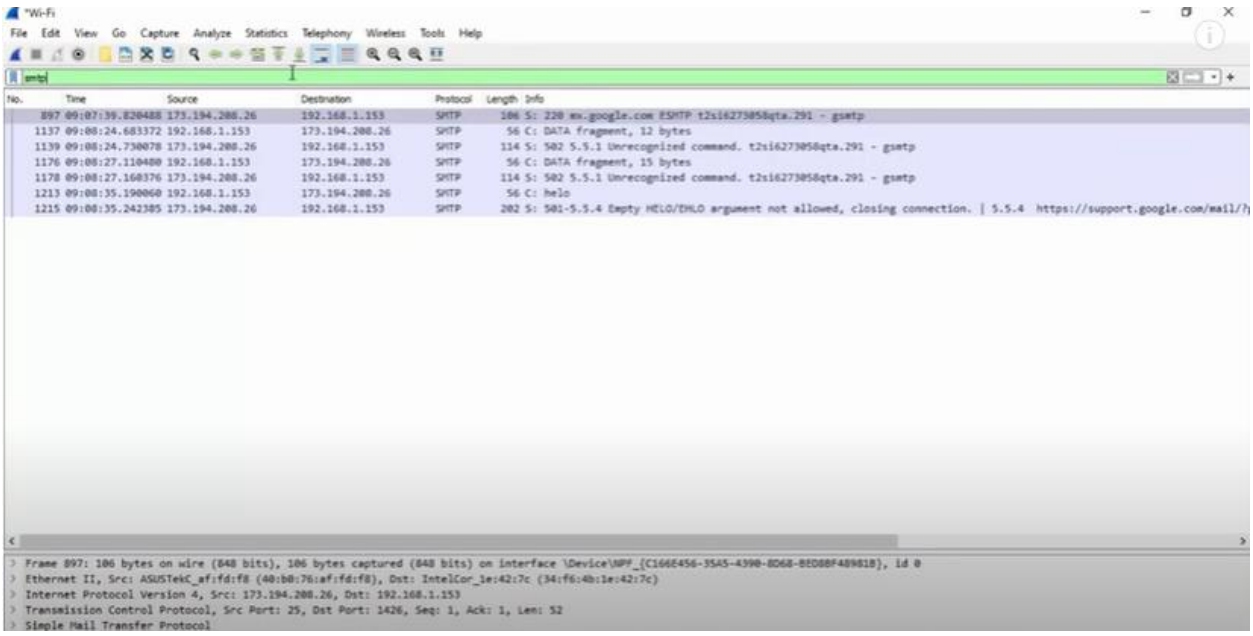
Step.2 Open windows features from start menu and check telnet client feature ON. Then Open command Prompt and type **telnet gmail-smtp-in.l.google.com 25** and press **Enter**. If this does not work, your ISP may be blocking outbound traffic on port 25. You can try **telnet smtp.gmail.com 587** instead to generate SMTP traffic and then filter on port 587 in the next activity.



Step 3: Observe the server response and type *hello* and enter. Observe the server response. Note that at this point you could enter mail, rcpt and data to send an SMTP message, but this only works on servers configured to allow clear text relay without authentication.



Step.4 Type **quit** and press **Enter** to close the connection. Observe the server response and close the command prompt, stop capture wireshark.



ACTVITIY 1: Select Destination Traffic

1. Observe the traffic captured in the top Wireshark packet list pane. To view only SMTP traffic, type **smtp** (lower case) in the Filter box and press **Enter.**
2. Select the first SMTP packet labeled **220 ...**.
3. Observe the destination IP address.
4. To view all related traffic for this connection, change the filter to **ip.addr == <*destination*>**, where <*destination*> is the destination address of the SMTP packet.

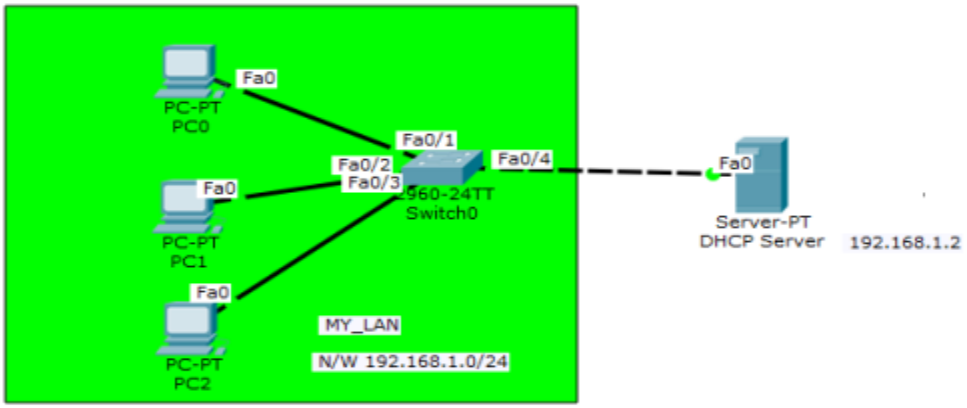
4) Stage a2 (assess)

1. Configure two different email servers for two users of different domains using Packet Tracer.

Lab # 05 (b): Dynamic Host Configuration Protocol (DHCP) Configuration on a Server using Packet Tracer

Configuring DHCP service on a generic server in Packet Tracer.

1. Build the network topology in packet tracer



2. Configure static IP address on the server (192.168.1.2/24).

3. Now configure DHCP service on the generic server.

To do this, click on the server, then click on **Services tab**. You will pick **DHCP** on the menu. Then proceed to define the DHCP network parameters as follows:

Pool name: MY_LAN

Default Gateway: 192.168.1.1

DNS Server: 192.168.1.2

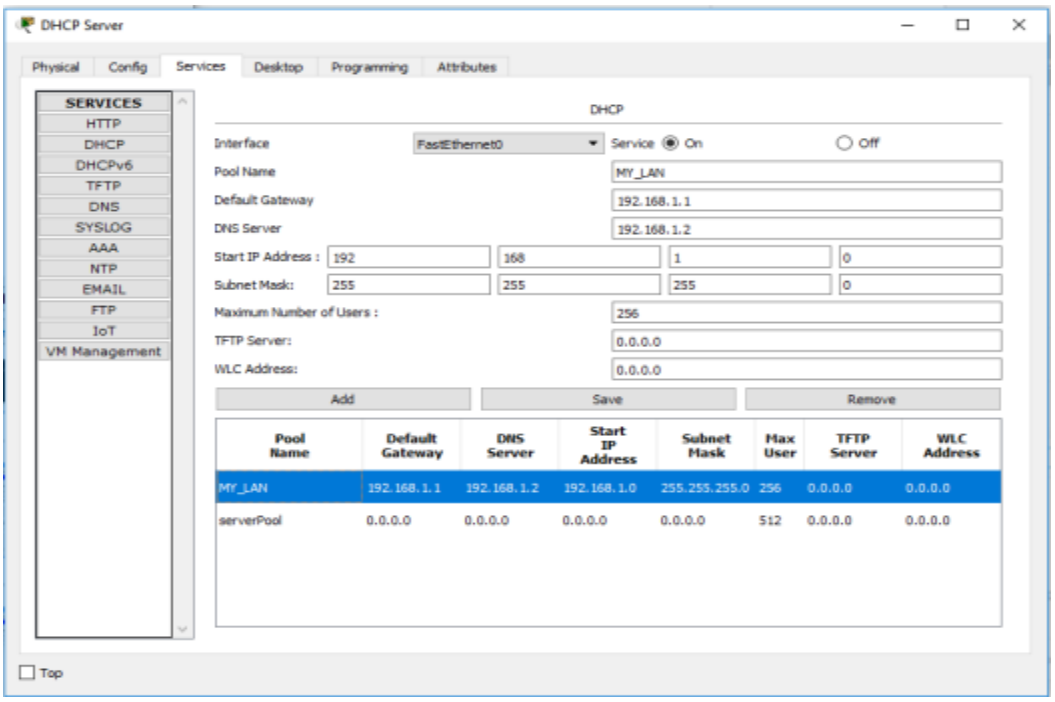
Start IP Address: 192.168.1.0

Subnet Mask: 255.255.255.0

Maximum Number of users: 256

Click on **add** then **Save.**The DHCP entry is included in the list.

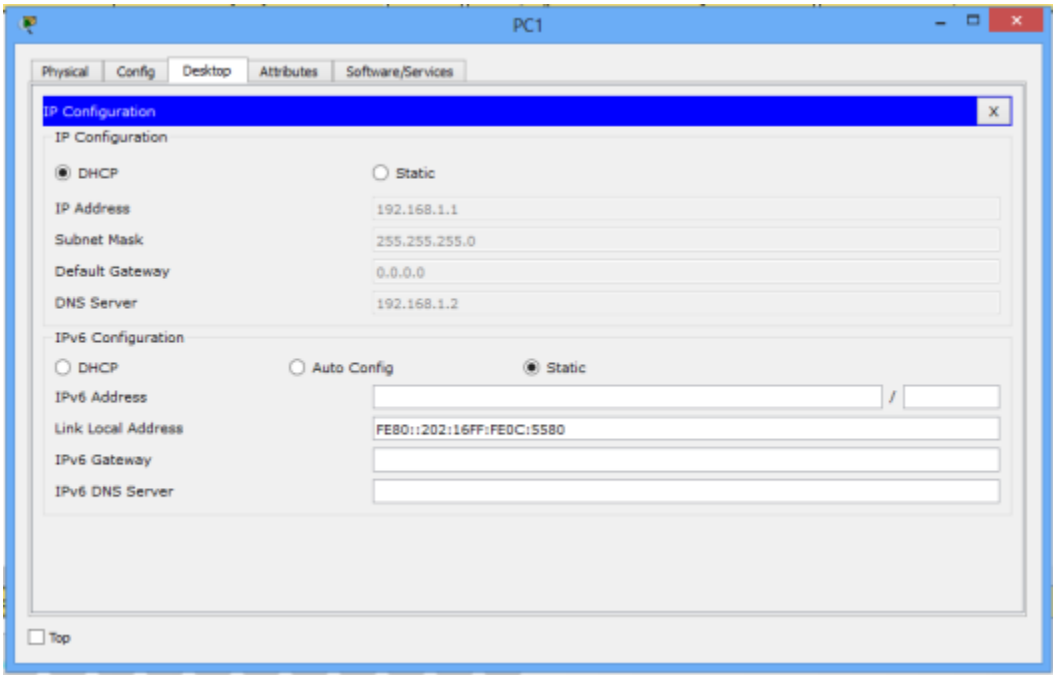
Here are the configurations on the server:



Once you've configured everything, turn **ON** the DHCP service.

4. Finally, enable DHCP configuration on each PC. The three PCs should get automatically configured.

As an example, here is the DHCP configuration on PC1:



Addendum: You can define a DHCP server on one broadcast domain to serve hosts in a **different** broadcast domain. If you want to do this, then you should consider using ip helper-address command. To learn more about this, you can read my article on IP helper address configuration**.**

Lab # 06 (a): Router Configuration through Command Line using Packet Tracer

Basic Settings on a Router

When initially configuring a Cisco switch or router, the following steps should be executed:

1. **Step 1.** Name the device. This changes the router prompt and helps distinguish the device from others.
2. **Step 2.** Secure management access. Specifically, secure the privileged EXEC, user EXEC, and Telnet access, and encrypt passwords to their highest level.
3. **Step 3.** Configure a banner. Although optional, this is a recommended step to provide legal notice to anyone attempting to access the device.
4. **Step 4.** Save the configuration.

For example, the following commands would configure the basic settings for router R1 shown in Figure 1

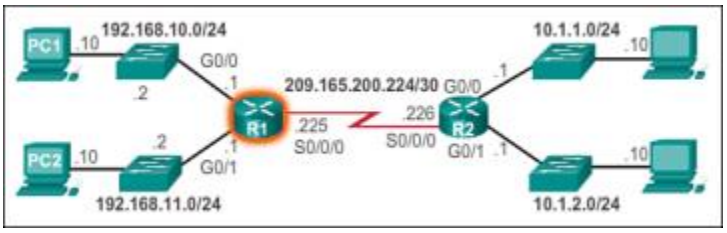


Figure 1 : Configuring the Basic Settings of R1

Router# **configure terminal**

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)# **hostname R1**

R1(config)#

R1(config)# **enable secret class**

R1(config)#

R1(config)# **line console 0**

R1(config-line)# **password cisco**

R1(config-line)# **login**

R1(config-line)# **exit**

R1(config)#

R1(config)# **line vty 0 4**

R1(config-line)# **password cisco**

R1(config-line)# **login**

R1(config-line)# **exit**

R1(config)#

R1(config)# **service password-encryption**

R1(config)#

R1(config)# **banner motd $ Authorized Access Only! $**

R1(config)# **end**

R1#

R1# **copy running-config startup-config**

Destination filename [startup-config]?

Building configuration...

[OK]

R1#

Configure an IPv4 Router Interface

One distinguishing feature between switches and routers is the type of interfaces supported by each. For example, Layer 2 switches support LANs and, therefore, have multiple FastEthernet or Gigabit Ethernet ports.

Routers support LANs and WANs and can interconnect different types of networks; therefore, they support many types of interfaces. For example, G2 ISRs have one or two integrated Gigabit Ethernet interfaces and ***High-Speed WAN Interface Card (HWIC)*** slots to accommodate other types of network interfaces, including serial, DSL, and cable interfaces.

To be available, an interface must be:

1. **If using IPv4, configured with an address and a subnet mask**: Use the **ip address** *ip-address subnet-mask* interface configuration command.

2. **Activated**: By default, LAN and WAN interfaces are not activated (**shutdown**). To enable an interface, it must be activated using the **no shutdown** command. (This is similar to powering on the interface.) The interface must also be connected to another device (a hub, a switch, or another router) for the physical layer to be active.

Optionally, the interface could also be configured with a short description. It is good practice to configure a description on each interface. The description text is limited to 240 characters. On production networks, a description can be helpful in troubleshooting by providing information about the type of network to which the interface is connected. If the interface connects to an ISP or service carrier, it is helpful to enter the third-party connection and contact information.

Depending on the type of interface, additional parameters may be required. For example, in the lab environment, the serial interface connecting to the serial cable end labeled DCE must be configured with the **clock rate** command.

The steps to configure an IPv4 interface on a router are:

1. **Step 1.** Add a description. Although optional, it is a necessary component for documenting a network.

2. **Step 2.** Configure the IPv4 address.

3. **Step 3.** Configure a clock rate on Serial interfaces. This is only necessary on the DCE device in our lab environment and does not apply to Ethernet interfaces.

4. **Step 4.** Enable the interface.

For example, the following commands would configure the three directly connected interfaces of router R1 shown in Figure 1-14 (in the previous section):

R1(config)# **interface gigabitethernet** 0/0

R1(config-if)# **description Link to LAN 1**

R1(config-if)# **ip address 192.168.10.1 255.255.255.0**

R1(config-if)# **no shutdown**

R1(config-if)# **exit**

R1(config)#

R1(config)# **interface gigabitethernet 0/1**

R1(config-if)# **description Link to LAN 2**

R1(config-if)# **ip address 192.168.11.1 255.255.255.0**

R1(config-if)# **no shutdown**

R1(config-if)# **exit**

R1(config)#

R1(config)# **interface serial 0/0/0**

R1(config-if)# **description Link to R2**

R1(config-if)# **ip address 209.165.200.225 255.255.255.252**

R1(config-if)# **clock rate 128000**

R1(config-if)# **no shutdown**

R1(config-if)# **exit**

R1(config)#

Configure an IPv6 Router Interface

Configuring an IPv6 interface is similar to configuring an interface for IPv4. Most IPv6 configuration and verification commands in the Cisco IOS are very similar to their IPv4 counterparts. In many cases, the only difference uses **ipv6** in place of **ip** in commands.

An IPv6 interface must be:

1. **Configured with IPv6 address and subnet mask**: Use the **ipv6 address** *ipv6-address*/*prefix-length* [**link-local** | **eui-64**] interface configuration command.

2. **Activated**: The interface must be activated using the **no shutdown** command.

NOTE

An interface can generate its own IPv6 link-local address without having a global unicast address by using the **ipv6 enable** interface configuration command.

Unlike IPv4, IPv6 interfaces will typically have more than one IPv6 address. At a minimum, an IPv6 device must have an IPv6 link-local address but will most likely also have an IPv6 global unicast address. IPv6 also supports the ability for an interface to have multiple IPv6 global unicast addresses from the same subnet. The following commands can be used to statically create a global unicast or link-local IPv6 address:

1. **ipv6 address** *ipv6-address/prefix-length*: Creates a global unicast IPv6 address as specified.

2. **ipv6 address** *ipv6-address/prefix-length* **eui-64**: Configures a global unicast IPv6 address with an interface identifier (ID) in the low-order 64 bits of the IPv6 address using the EUI-64 process.

3. **ipv6 address** *ipv6-address/prefix-length* **link-local**: Configures a static link-local address on the interface that is used instead of the link-local address that is automatically configured when the global unicast IPv6 address is assigned to the interface or enabled using the **ipv6 enable** interface command. Recall, the **ipv6 enable** interface command is used to automatically create an IPv6 link-local address whether or not an IPv6 global unicast address has been assigned.

The steps to configure an IPv6 interface on a router are:

1. **Step 1.** Add a description. Although optional, it is a necessary component for documenting a network.

2. **Step 2.** Configure the IPv6 global unicast address. Configuring a global unicast address automatically creates a link-local IPv6 address.

3. **Step 3.** Configure a link-local unicast address which automatically assigns a link-local IPv6 address and overrides any previously assigned address.

4. **Step 4.** Configure a clock rate on Serial interfaces. This is only necessary on the DCE device in our lab environment and does not apply to Ethernet interfaces.

5. **Step 5.** Enable the interface.

In the example topology shown in Figure 2 below, R1 must be configured to support the following IPv6 global network addresses:

1. 2001:0DB8:ACAD:0001:/64 (2001:DB8:ACAD:1::/64)

2. 2001:0DB8:ACAD:0002:/64 (2001:DB8:ACAD:2::/64)

3. 2001:0DB8:ACAD:0003:/64 (2001:DB8:ACAD:3::/64)

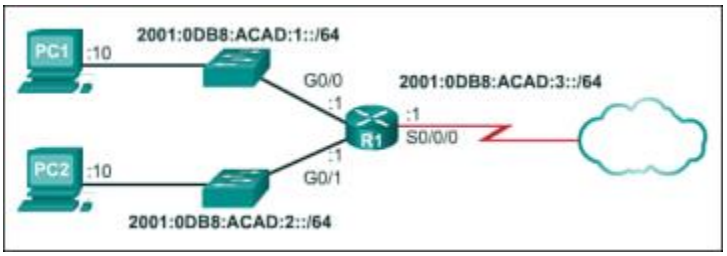


Figure 2: IPv6 Topology

When the router is configured using the **ipv6 unicast-routing** global configuration command, the router begins sending ICMPv6 Router Advertisement messages out the interface. This enables a PC connected to the interface to automatically configure an IPv6 address and to set a default gateway without needing the services of a DHCPv6 server. Alternatively, a PC connected to the IPv6 network can get its IPv6 address statically assigned, as shown in Figure 3. Notice that the default gateway address configured for PC1 is the IPv6 global unicast address of the R1 Gigabit Ethernet 0/0 interface.

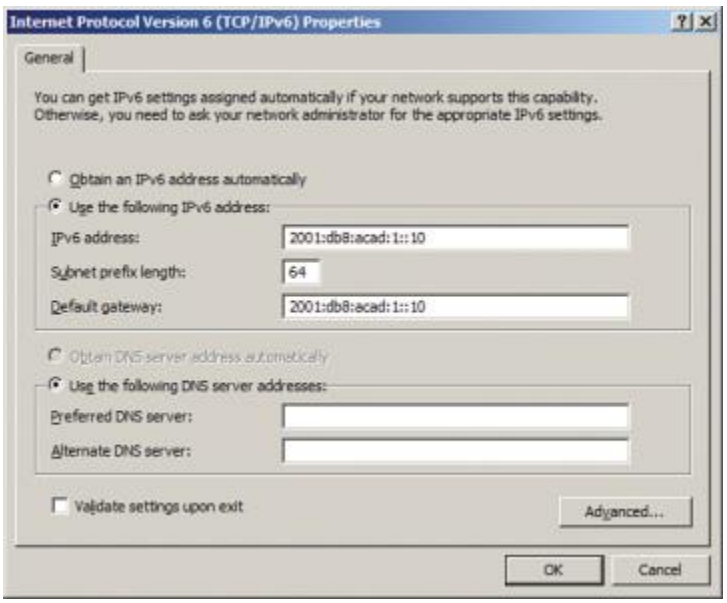


Figure 3 Statically Assign an IPv6 Address to PC1

For example, the following commands would configure the IPv6 global unicast addresses of the three directly connected interfaces of the R1 router shown in Figure 2:

R1# **configure terminal**

R1(config)# **interface gigabitethernet 0/0**

R1(config-if)# **description Link to LAN 1**

R1(config-if)# **ipv6 address 2001:db8:acad:1::1/64**

R1(config-if)# **no shutdown**

R1(config-if)# **exit**

R1(config)#

R1(config)# **interface gigabitethernet 0/1**

R1(config-if)# **description Link to LAN 2**

R1(config-if)# **ipv6 address 2001:db8:acad:2::1/64**

R1(config-if)# **no shutdown**

R1(config-if)# **exit**

R1(config)#

R1(config)# **interface serial 0/0/0**

R1(config-if)# **description Link to R2**

R1(config-if)# **ipv6 address 2001:db8:acad:3::1/64**

R1(config-if)# **clock rate 128000**

R1(config-if)# **no shutdown**

R1(config-if)#

Configure an IPv4 Loopback Interface

Another common configuration of Cisco IOS routers is enabling a loopback interface.

The ***loopback interface*** is a logical interface internal to the router. It is not assigned to a physical port and can therefore never be connected to any other device. It is considered a software interface that is automatically placed in an "up/up" state, as long as the router is functioning.

The loopback interface is useful in testing and managing a Cisco IOS device because it ensures that at least one interface will always be available. For example, it can be used for testing purposes, such as testing internal routing processes, by emulating networks behind the router.

Additionally, the IPv4 address assigned to the loopback interface can be significant to processes on the router that use an interface IPv4 address for identification purposes, such as the Open Shortest Path First (OSPF) routing process. By enabling a loopback interface, the router will use the always available loopback interface address for identification, rather than an IP address assigned to a physical port that may go down.

The steps to configure a loopback interface on a router are:

1. **Step 1.** Create the loopback interface using the **interface loopback** *number* global configuration command.

2. **Step 2.** Add a description. Although optional, it is a necessary component for documenting a network.

3. **Step 3.** Configure the IP address.

For example, the following commands configure a loopback interface of the R1 router shown in Figure 1

R1# **configure terminal**

R1(config)# **interface loopback 0**

R1(config-if)# **ip address 10.0.0.1 255.255.255.0**

R1(config-if)# **exit**

R1(config)#

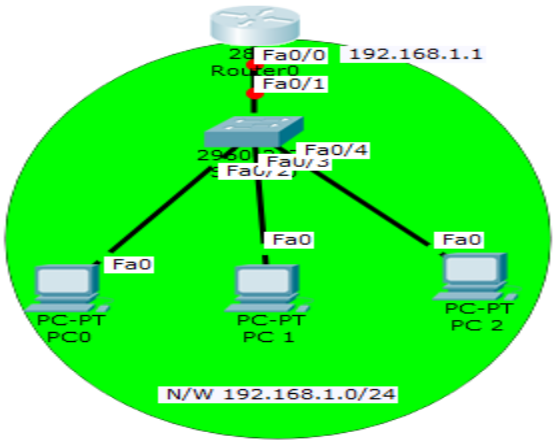
A loopback interface is always enabled and therefore does not require a **no shutdown** command. Multiple loopback interfaces can be enabled on a router. The IPv4 address for each loopback interface must be unique and unused by any other interface.

Lab # 06 (b): Dynamic Host Configuration Protocol (DHCP) Configuration on a Router using Packet Tracer

Introduction

Configuring DHCP server on a Router.

1. Build the network topology:



2. On the router, configure *interface fa0/0* to act as the default gateway for our LAN.

Router>enable

Router#config terminal

Router(config)#int fa0/0

Router(config-if)#ip add 192.168.1.1 255.255.255.0

Router(config-if)#no shutdown

Router(config-if)#exit

3. Configure DHCP server on the Router. In the server we will define a **DHCP pool** of IP addresses to be assigned to hosts, a **Default gateway** for the LAN and a **DNS Server**.

Router(config)#

Router(config)#ip dhcp pool MY_LAN

Router(dhcp-config)#network 192.168.1.0 255.255.255.0

Router(dhcp-config)#default-router 192.168.1.1

Router(dhcp-config)#dns-server 192.168.1.10

We can add ip dhcp excluded-address command to our configuration so as to configure the router to exclude addresses 192.168.1.1 through 192.168.1.10 when assigning addresses to clients. The **ip dhcp excluded-address** command may be used to reserve addresses that are statically assigned to key hosts.

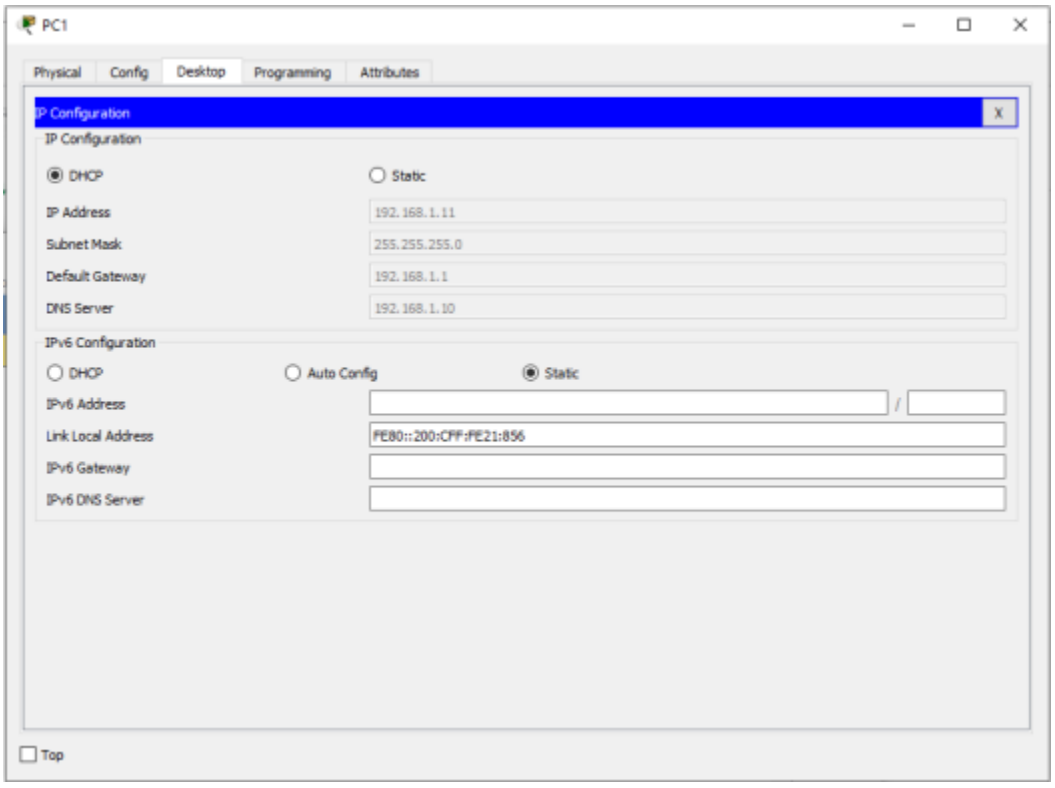
So add the above command under the **global configuration mode.**

Router(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.10

4. Now go to every PC and on their **IP configuration** tabs, enable **DHCP**. Every PC should be able to obtain an IP address, default gateway and DNS server, as defined in step 2.

For example, to enable DHCP on PC1:

Click **PC1->Desktop->IP configuration.** Then enable DHCP:



Do this for the other PCs.

You can test the configuration by pinging PC2 from PC1. Ping should succeed.

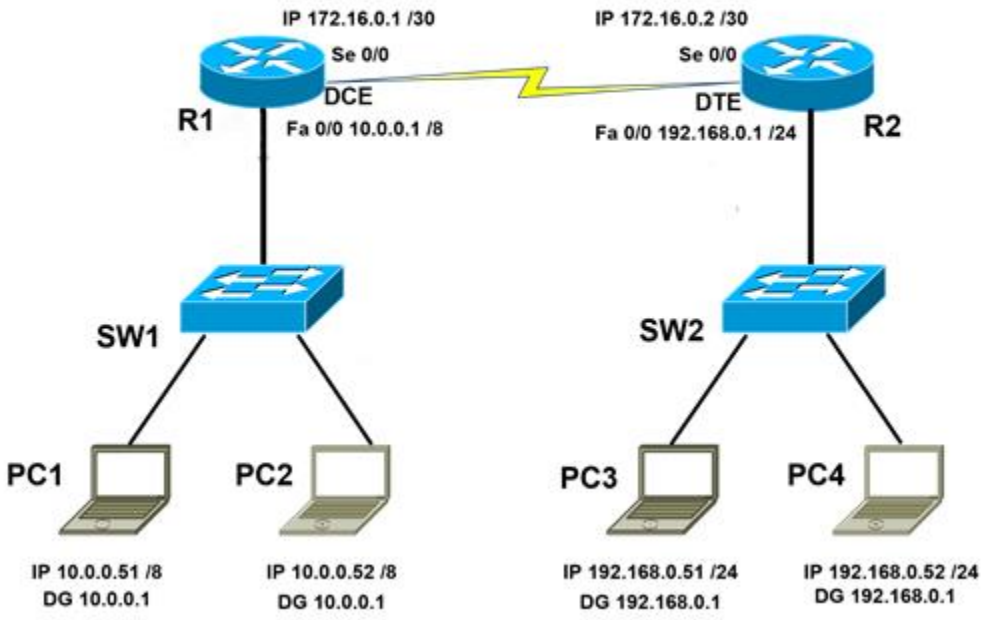
Lab # 07: Static Routing using Packet Tracer

Introduction

The objective of this lab is to:

1. To configure & implement Static Route successfully on said devices and test its all functionalities practically.

TOPOLOGY: Setup your lab topology as below.



Procedure:

1. Configure the hostnames for switches as SW1, SW2 & for Routers as R1, R2 as shown in above topology
2. Set the ip address of interface Fa 0/0 for R1 and R2 as shown in above topology
3. Set the ip address of interface Se 0/0 for R1, set clock rate for this interface & also set ip address of interface Se 0/0 for R2 as shown in above topology
4. Configure static route on R1 & R2

5. Set the ip address and default gateway for pc's PC1, PC2, PC3 & PC4 as shown in above topology
6. Verify the static route configured on R1 & R2 by using **show ip route** command
7. Check the connectivity between all PC's with each other

Configuration:

Step 1:

Configure the hostnames for switches as SW1, SW2 & for Routers as R1,R2 as shown in above topology

1.1: For SW1

Switch>enable

Switch#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname SW1

SW1(config)#

1.2: For SW2

Switch>enable

Switch#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname SW2

SW2(config)#

1.3: For R1

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R1

R1(config)#

1.4: For R2

Router>enable

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R2

R2(config)#

Step 2:

Set the ip address of interface Fa 0/0 for R1 and R2 as shown in above topology

2.1: For R1

R1(config)#interface fastEthernet 0/0

R1(config-if)#ip address 10.0.0.1 255.0.0.0

R1(config-if)#no shutdown

%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

2.2: For R2

R2(config)#interface fastEthernet 0/0

R2(config-if)#ip address 192.168.0.1 255.255.255.0

R2(config-if)#no shutdown

%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

Step 3:

Set the ip address of interface Se 0/0 for R1, set clock rate for this interface & also set ip address of interface Se 0/0 for R2 as shown in above topology

3.1: For R1

R1(config)#interface serial 0/0

R1(config-if)#ip address 172.16.0.1 255.255.255.252

R1(config-if)#clock rate 64000

R1(config-if)#no shutdown

3.2: For R2

R2(config)#interface serial 0/0

R2(config-if)#ip address 172.16.0.2 255.255.255.252

R2(config-if)#no shutdown

Step 4:

Configure static route on R1 & R2

4.1: For R1

R1(config)#ip route 192.168.0.0 255.255.255.0 172.16.0.2

OR

R1(config)#ip route 192.168.0.0 255.255.255.0 serial0/0

4.2: For R2

R2(config)#ip route 10.0.0.0 255.0.0.0 172.16.0.1

OR

R2(config)#ip route 10.0.0.0 255.0.0.0 serial0/0

Step 5:

Verify the static route configured on R1 & R2 by using show ip route command

5.1: For R1

R1#show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 10.0.0.0/8 is directly connected, FastEthernet0/0
172.16.0.0/30 is subnetted, 1 subnets
C 172.16.0.0 is directly connected, Serial0/0
S 192.168.0.0/24 [1/0] via 172.16.0.2

5.2: For R2

R2#show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

S 10.0.0.0/8 [1/0] via 172.16.0.1
172.16.0.0/30 is subnetted, 1 subnets
C 172.16.0.0 is directly connected, Serial0/0
C 192.168.0.0/24 is directly connected, FastEthernet0/0

Step 6:

Check the connectivity between all PC's with each other

6.1: Check the connectivity between PC1 & PC3

PC1>ping 192.168.0.51

Pinging 192.168.0.51 with 32 bytes of data:

Request timed out.
Reply from 192.168.0.51: bytes=32 time=141ms TTL=126
Reply from 192.168.0.51: bytes=32 time=140ms TTL=126
Reply from 192.168.0.51: bytes=32 time=140ms TTL=126

Ping statistics for 192.168.0.51:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 140ms, Maximum = 141ms, Average = 140ms

6.2: Check the connectivity between PC2 & PC4

PC2>ping 192.168.0.52

Pinging 192.168.0.52 with 32 bytes of data:

Request timed out.
Reply from 192.168.0.52: bytes=32 time=141ms TTL=126
Reply from 192.168.0.52: bytes=32 time=140ms TTL=126
Reply from 192.168.0.52: bytes=32 time=140ms TTL=126

Ping statistics for 192.168.0.52:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 140ms, Maximum = 141ms, Average = 140ms

Lab # 08 (a): Transport Control Protocol (TCP) using Wireshark

Statement Purpose:

8. Investigate the behavior of the celebrated TCP protocol in detail
9. Analyze a trace of the TCP segments sent and received in transferring a 150KB file from your computer to a remote server.
10. Study TCP's use of sequence and acknowledgement numbers for providing reliable data transfer
11. Study TCP's congestion control algorithm – slow start and congestion avoidance in action; and we'll look at TCP's receiver-advertised flow control mechanism.
12. Study TCP connection setup and investigate the performance (throughput and round-trip time) of the TCP connection between student's computer and the server.

Activity outcomes:

1. Students will gain better understanding of the TCP protocol

Introduction

Capturing a bulk TCP transfer from your computer to a remote server Before beginning our exploration of TCP, we'll need to use Wireshark to obtain a packet trace of the TCP transfer of a file from your computer to a remote server. You'll do so by accessing a Web page that will allow you to enter the name of a file stored on your computer (which contains the ASCII text of *Alice in Wonderland*), and then transfer the file to a Web server using the HTTP POST method. We're using the POST method rather than the GET method as we'd like to transfer a large amount of data *from* your computer to another computer. Of course, we'll be running Wireshark during this time to obtain the trace of the TCP segments sent and received from your computer.

Lab Activities:

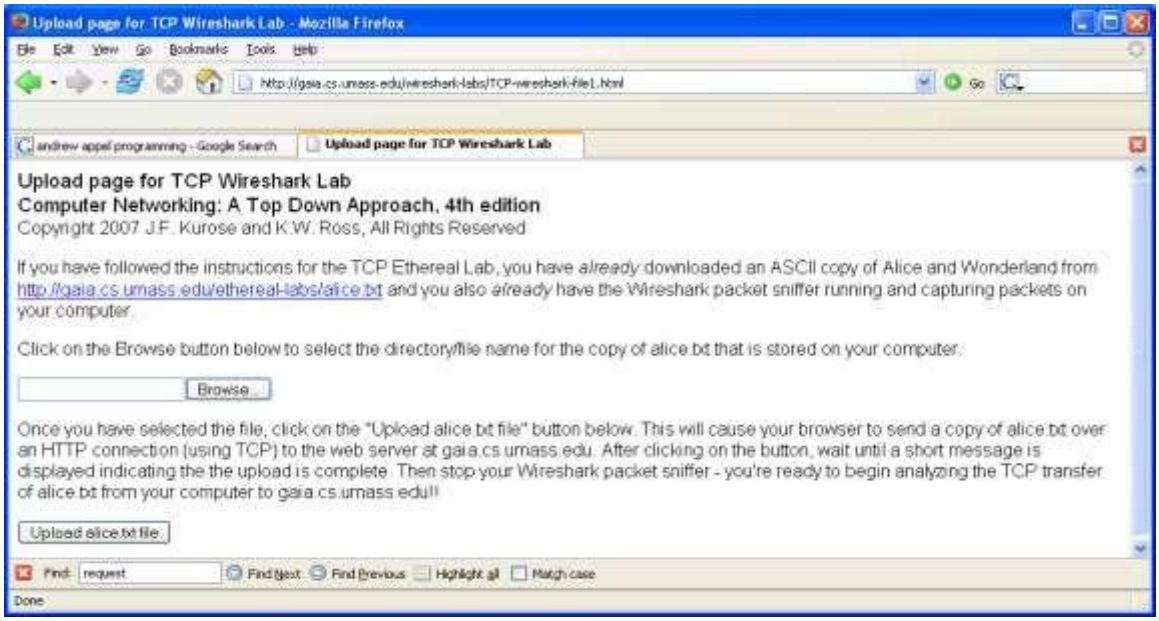
Activity 1:

1. Examples

Do the following:

2. Start up your web browser. Go the http://gaia.cs.umass.edu/wireshark-labs/alice.txt and retrieve an ASCII copy of *Alice in Wonderland.* Store this file somewhere on your computer.
3. Next go to http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html.

4. You should see a screen that looks like:

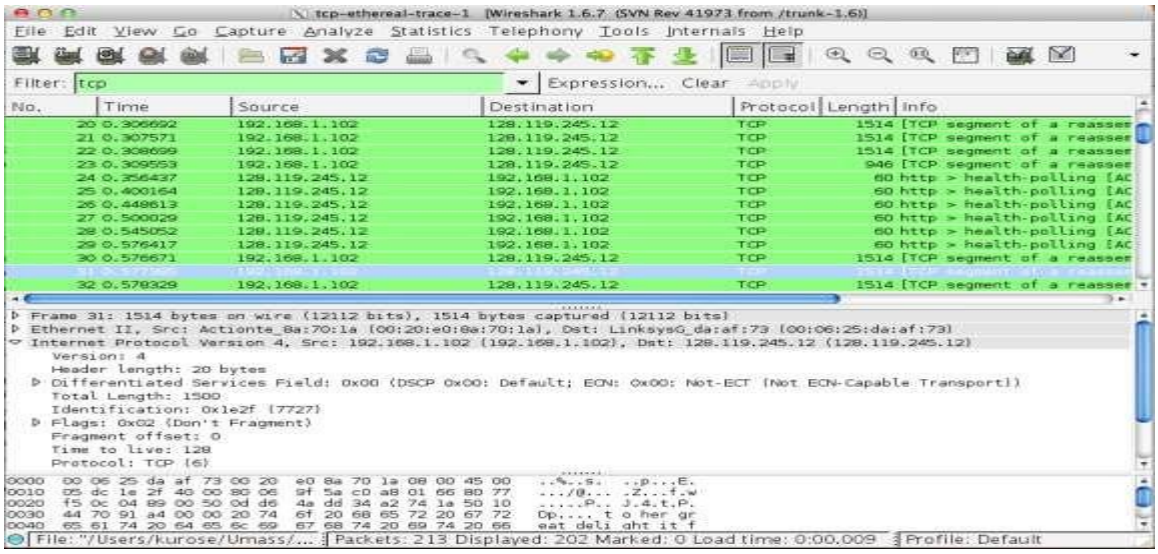


1. Use the *Browse* button in this form to enter the name of the file (full path name) on your computer containing *Alice in Wonderland* (or do so manually).Don't yet press the "*Upload alice.txt file*" button.

2. Now start up Wireshark and begin packet capture *(Capture->Start)* and then press *OK* on the Wireshark Packet Capture Options screen (we'll not need to select any options here).

3. Returning to your browser, press the "*Upload alice.txt file*" button to upload the file to the gaia.cs.umass.edu server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.

4. Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.



If you are unable to run Wireshark on a live network connection, you can download a packet trace file that was captured while following the steps above on one of the author's computers[3]. You may well find it valuable to download this trace even if you've captured your own trace and use it, as well as your own trace, when you explore the questions below.

A first look at the captured trace

Before analyzing the behavior of the TCP connection in detail, let's take a high level view of the trace.

First, filter the packets displayed in the Wireshark window by entering "tcp" (lowercase, no quotes, and don't forget to press return after entering!) into the display filter specification window towards the top of the Wireshark window.

What you should see is series of TCP and HTTP messages between your computer and gaia.cs.umass.edu. You should see the initial three-way handshake containing a SYN message. You should see an HTTP POST message. Depending on the version of Wireshark you are using, you might see a series of "HTTP Continuation" messages being sent from your computer to gaia.cs.umass.edu. Recall from our discussion in the earlier HTTP Wireshark lab, that is no such thing as an HTTP Continuation message – this is Wireshark's way of indicating that there are multiple TCP segments being used to carry a single HTTP message. In more recent versions of Wireshark, you'll see "[TCP segment of a reassembled PDU]" in the Info column of the Wireshark display to indicate that this TCP segment contained data that belonged to an upper

layer protocol message (in our case here, HTTP). You should also see TCP ACK segments being returned from gaia.cs.umass.edu to your computer.

Answer the following questions, by opening the Wireshark captured packet file *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip (that is download the trace and open that trace in Wireshark; see footnote 2). Whenever possible, when answering a question you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout[4] to explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line*, and select the minimum amount of packet detail that you need to answer the question.

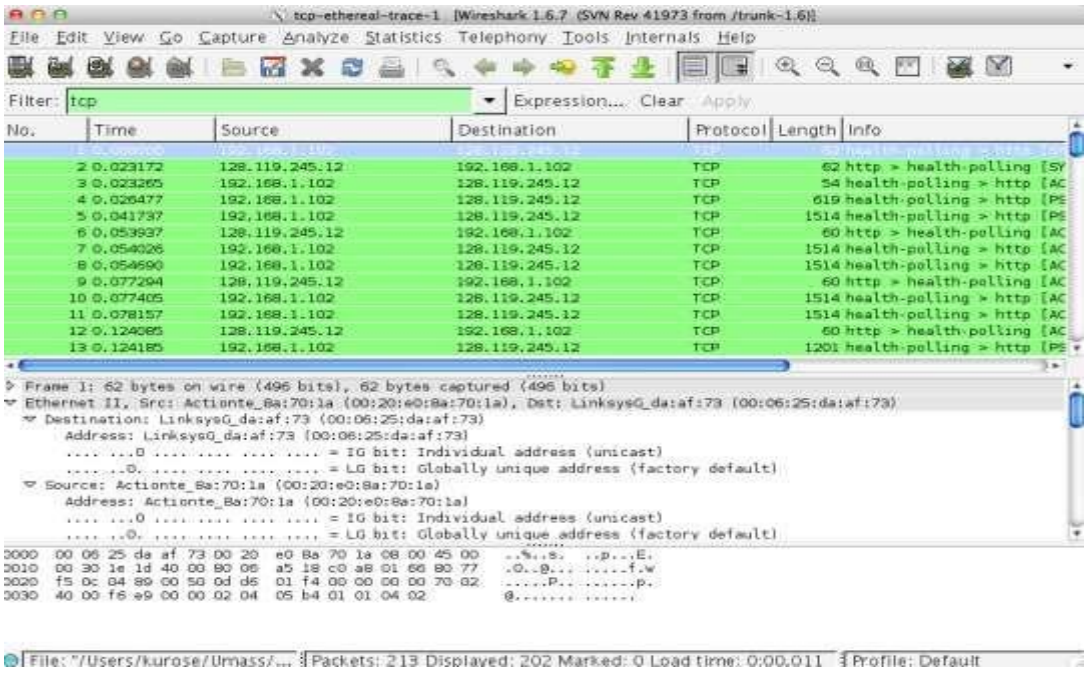
1. What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows.

2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

3. If you have been able to create your own trace, answer the following question:

4. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?

5. Since this lab is about TCP rather than HTTP, let's change Wireshark's "listing of captured packets" window so that it shows information about the TCP segments containing the HTTP messages, rather than about the HTTP messages. To have Wireshark do this, select *Analyze->Enabled Protocols.* Then uncheck the HTTP box and select *OK*. You should now see a Wireshark window that looks like:



This is what we're looking for - a series of TCP segments sent between your computer and gaia.cs.umass.edu. We will use the packet trace that you have captured (and/or the packet trace *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip;

Activity 2:
TCP Basics

Answer the following questions for the TCP segments:

1. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?

2. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

3. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

4. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection

(including the segment containing the HTTP POST)? At what time was each segment sent?

When was the ACK for each segment received? Given the difference between when each

TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value (see Section 3.5.3, page 239 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 239 for all subsequent segments.

Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the "listing of captured packets" window that is being sent from the client to the gaia.cs.umass.edu server. Then select: *Statistics->TCP Stream Graph->Round Trip Time Graph.*

1. What is the length of each of the first six TCP segments?[10]
2. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?
3. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
4. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

Lab # 08 (b): User Datagram Protocol (UDP) using Wireshark

Statement Purpose:

5. Investigate the behavior of the celebrated UDP protocol in detail
6. After this lab, the students will get better understanding of the UDP protocol.
7. Explore several aspects of UDP protocol.

Activity outcomes:

8. Students will gain better understanding of the UDP protocol

Introduction

In this lab, we'll take a quick look at the UDP transport protocol. UDP is a streamlined, no-frills protocol. You may want to re-read section 3,3 in the text before doing this lab. Because UDP is simple and sweet, we'll be able to cover it pretty quickly in this lab.

Lab Activities:

Activity 1:

Start capturing packets in Wireshark and then do something that will cause your host to send and receive several UDP packets. It's also likely that just by doing nothing (except capturing packets via Wireshark) that some UDP packets sent by others will appear in your trace. In particular, the Simple Network Management Protocol sends SNMP messages inside of UDP, so it's likely that you'll find some SNMP messages (and therefore UDP packets) in your trace.

After stopping packet capture, set your packet filter so that Wireshark only displays the UDP packets sent and received at your host. Pick one of these UDP packets and expand the UDP fields in the details window. If you are unable to find UDP packets or are unable to run Wireshark on a live network connection, you can download a packet trace containing some UDP packets.[5]

Whenever possible, when answering a question below, you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout[12] to

5 Download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the file httpethereal-trace-5, which contains some UDP packets carrying SNMP messages. The traces in this zip file were collected by Wireshark running on one of the author's computers. Once you have downloaded the trace, you can load it into Wireshark and view the trace using the *File* pull down menu, choosing *Open*, and then selecting the http-ethereal-trace-5 trace file.

explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line,* and select the minimum amount of packet detail that you need to answer the question.

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.

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.

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.

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)

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

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

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.

Lab # 09: Transport Control Protocol (TCP) and User Datagram Protocol (UDP) using Packet Tracer

Packet Tracer - TCP and UDP Communications

Objectives

Part 1: Generate Network Traffic in Simulation Mode

Part 2: Examine the Functionality of the TCP and UDP Protocols

Background

This simulation activity is intended to provide a foundation for understanding TCP and UDP in detail. Packet Tracer simulation mode provides you the ability to view the state of different PDUs as they travel through the network. Open the provided file named as TCP&UDP.pka

Packet Tracer Simulation mode enables you to view each of the protocols and the associated PDUs. The steps outlined below lead you through the process of requesting network services using various applications that are available on a client PC. You will explore the functionality of the TCP and UDP protocols, multiplexing, and the function of port numbers in determining which local application requested the data or is sending the data.

Instructions

Part 1: Generate Network Traffic in Simulation Mode and View Multiplexing

Step 1: Generate traffic to populate Address Resolution Protocol (ARP) tables.

Perform the following task to reduce the amount of network traffic viewed in the simulation.

a. Click **MultiServer** and click the **Desktop** tab > **Command Prompt**.

b. Enter the **ping -n 1 192.168.1.255** command. You are pinging the broadcast address for the client LAN. The command option will send only one ping request rather than the usual four. This will take a few seconds as every device on the network responds to the ping request from MultiServer.

c. Close the **MultiServer** window.

Step 2: Generate web (HTTP) traffic.

a. Switch to Simulation mode.

b. Click **HTTP Client** and open the **Web Browser** from the desktop.

c. In the URL field, enter **192.168.1.254** and click **Go**. Envelopes (PDUs) will appear in the topology window.

d. Minimize, but do not close, the **HTTP Client** configuration window.

Step 3: Generate FTP traffic.

a. Click **FTP Client** and open the **Command Prompt** from the desktop

b. Enter the **ftp 192.168.1.254** command. PDUs will appear in the simulation window.

c. Minimize, but do not close, the **FTP Client** configuration window.

Step 4: Generate DNS traffic.

a. Click DNS Client and open the **Command Prompt**.

b. Enter the **nslookup multiserver.pt.ptu** command. A PDU will appear in the simulation window.

c. Minimize, but do not close, the **DNS Client** configuration window.

Step 5: Generate Email traffic.

a. Click **E-Mail Client** and open the **E Mail** tool from the Desktop.

b. Click **Compose** and enter the following information:

1) **To:** user@multiserver.pt.ptu

2) **Subject:** personalize the subject line

3) **E-Mail Body:** personalize the Email

c. Click **Send**.

d. Minimize, but do not close, the **E-Mail Client** configuration window.

Step 6: Verify that the traffic is generated and ready for simulation.

There should now be PDU entries in the simulation panel for each of the client computers.

Step 7: Examine multiplexing as the traffic crosses the network.

You will now use the **Capture/Forward button** in the Simulation Panel to observe the different protocols travelling on the network.

Note: The **Capture/Forward** button ‘ >| ‘ is a small arrow pointing to the right with a vertical bar next to it.

a. Click **Capture/Forward** once. All of the PDUs travel to the switch.

b. Click **Capture/Forward** six times and watch the PDUs from the different hosts as they travel on the network. Note that only one PDU can cross a wire in each direction at any given time.

Questions:

What is this called?

A variety of PDUs appears in the event list in the Simulation Panel. What is the meaning of the different colors?

Part 2: Examine Functionality of the TCP and UDP Protocols

Step 1: Examine HTTP traffic as the clients communicate with the server.

a. Click **Reset Simulation**.

b. Filter the traffic that is currently displayed to only **HTTP** and **TCP** PDUs. To filter the traffic that is currently displayed:

1) Click **Edit Filters** and toggle the **Show All/None** button.

2) Select **HTTP** and **TCP**. Click the red "x" in the upper right-hand corner of the Edit Filters box to close it. Visible Events should now display only **HTTP** and **TCP** PDUs.

c. Open the browser on HTTP Client and enter **192.168.1.254** in the URL field. Click **Go** to connect to the server over HTTP. Minimize the HTTP Client window.

d. Click **Capture/Forward** until you see a PDU appear for HTTP. Note that the color of the envelope in the topology window matches the color code for the HTTP PDU in the Simulation Panel.

Question:

Why did it take so long for the HTTP PDU to appear?

e. Click the PDU envelope to show the PDU details. Click the **Outbound PDU Details** tab and scroll down to the second to the last section.

Questions:

What is the section labeled?

Are these communications considered to be reliable?

Record the **SRC PORT**, **DEST PORT**, **SEQUENCE NUM**, and **ACK NUM** values.

f. Look at the value in the Flags field, which is located next to the Window field. The values to the right of the "b" represent the TCP flags that are set for this stage of the data conversation. Each of the six places corresponds to a flag. The presence of a "1" in any place indicates that the flag is set. More than one flag can be set at a time. The values for the flags are shown below.

Flag Place	**6**	**5**	**4**	**3**	**2**	**1**
Value	URG	ACK	PSH	RST	SYN	FIN

Question:

Which TCP flags are set in this PDU?

g. Close the PDU and click **Capture/Forward** until a PDU with a checkmark returns to the **HTTP Client**.

h. Click the PDU envelope and select **Inbound PDU Details**.

Question:

How are the port and sequence numbers different than before?

i. Click the HTTP PDU which **HTTP Client** has prepared to send to **MultiServer**. This is the beginning of the HTTP communication. Click this second PDU envelope and select **Outbound PDU Details**.

Question:

What information is now listed in the TCP section? How are the port and sequence numbers different from the previous two PDUs?

j. Reset the simulation.

Step 2: Examine FTP traffic as the clients communicate with the server.

a. Open the command prompt on the FTP Client desktop. Initiate an FTP connection by entering **ftp 192.168.1.254**.

b. In the Simulation Panel, change **Edit Filters** to display only **FTP** and **TCP**.

c. Click **Capture/Forward**. Click the second PDU envelope to open it.

Click the **Outbound PDU Details** tab and scroll down to the TCP section.

Question:

Are these communications considered to be reliable?

d. Record the **SRC PORT**, **DEST PORT**, **SEQUENCE NUM**, and **ACK NUM** values.

Question:

What is the value in the flag field?

e. Close the PDU and click **Capture/Forward** until a PDU returns to the **FTP Client** with a checkmark.

f. Click the PDU envelope and select **Inbound PDU Details**.

Question:

How are the port and sequence numbers different than before?

g. Click the **Outbound PDU Details** tab.

Question:

How are the port and sequence numbers different from the previous results?

h. Close the PDU and click **Capture/Forward** until a second PDU returns to the **FTP Client**. The PDU is a different color.

i. Open the PDU and select **Inbound PDU Details**. Scroll down past the TCP section.

Question:

What is the message from the server?

j. Click Reset Simulation.

Step 3: Examine DNS traffic as the clients communicate with the server.

a. Repeat the steps in Part 1 to create DNS traffic.

b. In the Simulation Panel, change **Edit Filters** to display only **DNS** and **UDP**.

c. Click the PDU envelope to open it.

d. Look at the OSI Model details for the outbound PDU.

Question:

What is the Layer 4 protocol?

Are these communications considered to be reliable?

e. Open the Outbound PDU Details tab and find the UDP section of the PDU formats. Record the **SRC PORT** and **DEST PORT** values.

Question:

Why are there no sequence and acknowledgement numbers?

f. Close the **PDU** and click **Capture/Forward** until a PDU with a check mark returns to the **DNS Client**.

g. Click the PDU envelope and select **Inbound PDU Details**.

Question:

How are the port and sequence numbers different than before?

What is the last section of the **PDU** called? What is the IP address for the name **multiserver.pt.ptu**?

h. Click Reset Simulation.

Step 4: Examine email traffic as the clients communicate with the server.

a. Repeat the steps in Part 1 to send an email to **user@multiserver.pt.ptu**.

b. In the Simulation Panel, change **Edit Filters** to display only **POP3, SMTP** and **TCP**.

c. Click the first PDU envelope to open it.

d. Click the **Outbound PDU Details** tab and scroll down to the last section.

Questions:

What transport layer protocol does email traffic use?

Are these communications considered to be reliable?

e. Record the **SRC PORT**, **DEST PORT**, **SEQUENCE NUM**, and **ACK NUM** values. What is the flag field value?

f. Close the **PDU** and click **Capture/Forward** until a PDU returns to the **E-Mail Client** with a checkmark.

g. Click the TCP PDU envelope and select **Inbound PDU Details**.

Question:

How are the port and sequence numbers different than before?

h. Click the **Outbound PDU Details** tab.

Question:

How are the port and sequence numbers different from the previous two results?

i. There is a second **PDU** of a different color that **E-Mail Client** has prepared to send to **MultiServer**. This is the beginning of the email communication. Click this second PDU envelope and select **Outbound PDU Details**.

Questions:

How are the port and sequence numbers different from the previous two **PDU**s?

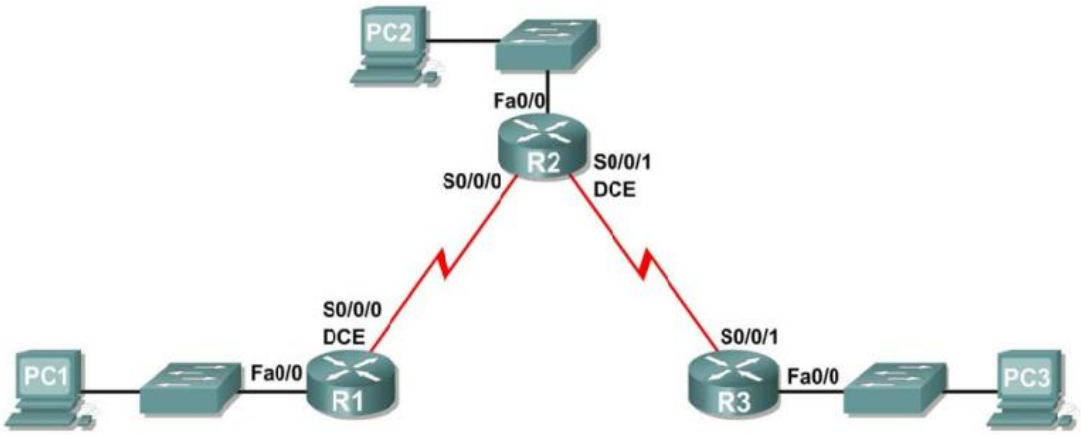
What email protocol is associated with TCP port 25? What protocol is associated with TCP port 110?

Lab # 10: Dynamic Routing using Packet Tracer

Upon completion of this lab, you will be able to:

8. Cable a network according to the Topology Diagram.
9. Perform basic configuration tasks on a router.
10. Configure and activate interfaces.
11. Configure RIP routing on all routers.
12. Verify RIP routing using show and debug commands.
13. Reconfigure the network to make it contiguous.
14. Observe automatic summarization at boundary router.
15. Gather information about RIP processing using the debug ip rip command.

Topology Diagram



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.2.1	255.255.255.0	N/A
R2	Fa0/0	192.168.3.1	255.255.255.0	N/A
	S0/0/0	192.168.2.2	255.255.255.0	N/A
	S0/0/1	192.168.4.2	255.255.255.0	N/A
R3	Fa0/0	192.168.5.1	255.255.255.0	N/A
	S0/0/1	192.168.4.1	255.255.255.0	N/A
PC1	NIC	192.168.1.10	255.255.255.0	192.168.1.1
PC2	NIC	192.168.3.10	255.255.255.0	192.168.3.1
PC3	NIC	192.168.5.10	255.255.255.0	192.168.5.1

Task 1: Prepare the Network.

Step 1: Cable a network that is similar to the one in the Topology Diagram.

You can use any current router in your lab as long as it has the required interfaces shown in the topology.

Task 2: Perform Basic Router Configurations.

Perform basic configuration of the R1, R2, and R3 routers according to the following guidelines:

1. Configure the router hostname.

Task 3: Configure and Activate Serial and Ethernet Addresses.

Step 1: Configure interfaces on R1, R2, and R3.

Configure the interfaces on the R1, R2, and R3 routers with the IP addresses from the table under the Topology Diagram.

Step 2: Verify IP addressing and interfaces.

Use the **show ip interface brief** command to verify that the IP addressing is correct and that the interfaces are active.

When you have finished, be sure to save the running configuration to the NVRAM of the router.

Step 3: Configure Ethernet interfaces of PC1, PC2, and PC3.

Configure the Ethernet interfaces of PC1, PC2, and PC3 with the IP addresses and default gateways from the table under the Topology Diagram.

Step 4: Test the PC configuration by pinging the default gateway from the PC.

Task 4: Configure RIP.

Step 1: Enable dynamic routing.

To enable a dynamic routing protocol, enter configuration mode and use the **router** command.

Enter **router ?** at the configuration prompt to a see a list of available routing protocols on your router.

To enable RIP, enter the command **router rip** in configuration mode.

R1(config)#**router rip**

R1(config-router)#

Step 2: Enter classful network addresses.

Once you are in routing configuration mode, enter the classful network address for each directly connected network, using the **network** command.

R1(config-router)#**network 192.168.1.0**

R1(config-router)#**network 192.168.2.0**

R1(config-router)#

The **network** command:

1. Enables RIP on all interfaces that belong to this network. These interfaces will now both send and receive RIP updates.
2. Advertises this network in RIP routing updates sent to other routers every 30 seconds.

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

R1(config-router)#**end**

%SYS-5-CONFIG_I: Configured from console by console

R1#**copy run start**

Step 3: Configure RIP on the R2 router using the router rip and network commands.

R2(config)#**router rip**

R2(config-router)#**network 192.168.2.0**

R2(config-router)#**network 192.168.3.0**

R2(config-router)#**network 192.168.4.0**

R2(config-router)#**end**

%SYS-5-CONFIG_I: Configured from console by console

R2#**copy run start**

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Step 4: Configure RIP on the R3 router using the router rip and network commands.

R3(config)#**router rip**

R3(config-router)#**network 192.168.4.0**

R3(config-router)#**network 192.168.5.0**

R3(config-router)#**end**

%SYS-5-CONFIG_I: Configured from console by console

R3# **copy run start**

When you are finished with the RIP configuration, return to privileged EXEC mode and save the current configuration to NVRAM.

Task 5: Verify RIP Routing.

Step 1: Use the show ip route command to verify that each router has all of the networks in the topology entered in the routing table.

Routes learned through RIP are coded with an **R** in the routing table. If the tables are not converged as shown here, troubleshoot your configuration. Did you verify that the configured interfaces are active? Did you configure RIP correctly? Return to Task 3 and Task 4 to review the steps necessary to achieve convergence.

R1#**show ip route**

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

* - candidate default, U - per-user static route, o - ODR

P - periodic downloaded static route

Gateway of last resort is not set

C 192.168.1.0/24 is directly connected, FastEthernet0/0

C 192.168.2.0/24 is directly connected, Serial0/0/0

R 192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:04, Serial0/0/0

R 192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:04, Serial0/0/0

R 192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:04, Serial0/0/0

R1#

R2#**show ip route**

<Output omitted>

R 192.168.1.0/24 [120/1] via 192.168.2.1, 00:00:22, Serial0/0/0

C 192.168.2.0/24 is directly connected, Serial0/0/0

C 192.168.3.0/24 is directly connected, FastEthernet0/0

C 192.168.4.0/24 is directly connected, Serial0/0/1

R 192.168.5.0/24 [120/1] via 192.168.4.1, 00:00:23, Serial0/0/1

R2#

R3#**show ip route**

<Output omitted>

R 192.168.1.0/24 [120/2] via 192.168.4.2, 00:00:18, Serial0/0/1

R 192.168.2.0/24 [120/1] via 192.168.4.2, 00:00:18, Serial0/0/1

R 192.168.3.0/24 [120/1] via 192.168.4.2, 00:00:18, Serial0/0/1

C 192.168.4.0/24 is directly connected, Serial0/0/1

C 192.168.5.0/24 is directly connected, FastEthernet0/0

R3#

Step 2: Use the show ip protocols command to view information about the routing processes.

The **show ip protocols** command can be used to view information about the routing processes that are occurring on the router. This output can be used to verify most RIP parameters to confirm that:

1. RIP routing is configured
2. The correct interfaces send and receive RIP updates
3. The router advertises the correct networks
4. RIP neighbors are sending updates

R1#**show ip protocols**

Routing Protocol is "rip"

Sending updates every 30 seconds, next due in 16 seconds

Invalid after 180 seconds, hold down 180, flushed after 240

Outgoing update filter list for all interfaces is not set

Incoming update filter list for all interfaces is not set

Redistributing: rip

Default version control: send version 1, receive any version

Interface	Send	Recv	Triggered RIP	Key-chain
FastEthernet0/0	1	2	1	
Serial0/0/0	1	2	1	

Automatic network summarization is in effect

Maximum path: 4

Routing for Networks:

192.168.1.0

192.168.2.0

Passive Interface(s):

Routing Information Sources:

Gateway Distance Last Update

192.168.2.2 120

Distance: (default is 120)

R1#

R1 is indeed configured with RIP. R1 is sending and receiving RIP updates on FastEthernet0/0 and Serial0/0/0. R1 is advertising networks 192.168.1.0 and 192.168.2.0. R1 has one routing information source. R2 is sending R1 updates.

Step 3: Use the debug ip rip command to view the RIP messages being sent and received.

Rip updates are sent every 30 seconds so you may have to wait for debug information to be displayed.

R1#**debug ip rip**

R1#RIP: received v1 update from 192.168.2.2 on Serial0/0/0

192.168.3.0 in 1 hops

192.168.4.0 in 1 hops

192.168.5.0 in 2 hops

RIP: sending v1 update to 255.255.255.255 via FastEthernet0/0 (192.168.1.1) RIP: build update entries network 192.168.2.0 metric 1 network 192.168.3.0 metric 2 network 192.168.4.0 metric 2 network 192.168.5.0 metric 3

RIP: sending v1 update to 255.255.255.255 via Serial0/0/0 (192.168.2.1) RIP: build update entries network 192.168.1.0 metric 1

The debug output shows that R1 receives an update from R2. Notice how this update includes all the networks that R1 does not already have in its routing table. Because the FastEthernet0/0 interface belongs to the 192.168.1.0 network configured under RIP, R1 builds an update to send out that interface. The update includes all networks known to R1 except the network of the interface. Finally, R1 builds an update to send to R2. Because of split horizon, R1 only includes the 192.168.1.0 network in the update.

Step 4: Discontinue the debug output with the undebug all command.

R1#**undebug all**

All possible debugging has been turned off

Lab # 11: NAT (Network Address Translation Protocol) using Packet Tracer

Objectives

1. Part 1: Build the Network and Verify Connectivity
2. Part 2: Configure and Verify NAT Pool Overload

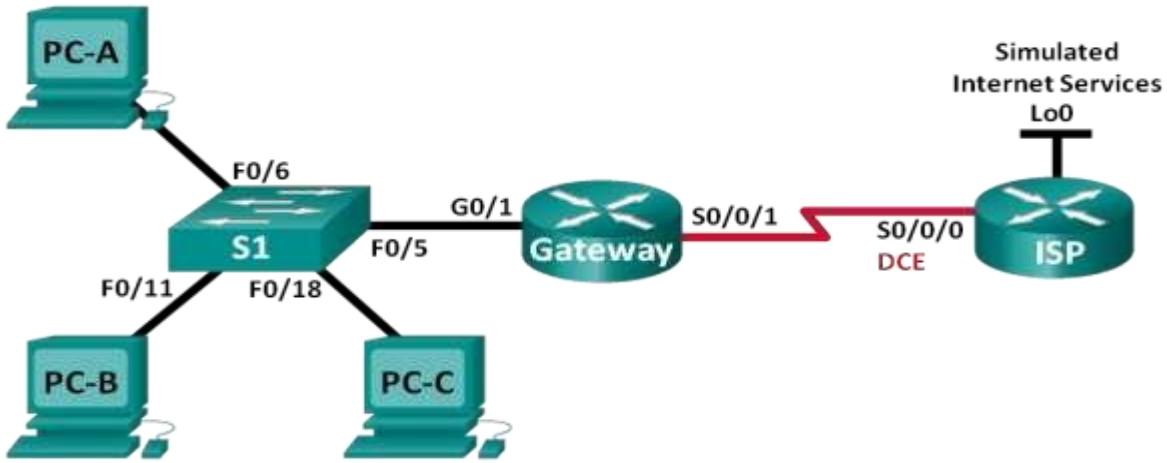
Background / Scenario

In this lab, your company is allocated the public IP address range of 209.165.200.224/29 by the ISP. This provides the company with six public IP addresses. Dynamic NAT pool overload uses a pool of IP addresses in a many-to-many relationship. The router uses the first IP address in the pool and assigns connections using the IP address plus a unique port number. After the maximum number of translations for a single IP address have been reached on the router (platform and hardware specific), it uses the next IP address in the pool.

Required Resources

1. 2 Routers (Cisco 1941)
2. 1 Switch (Cisco 2960)
3. 3 PCs (Windows 7, Vista, or XP with terminal emulation program, such as Tera Term)
4. Console cables to configure the Cisco IOS devices via the console ports
5. Ethernet and serial cables as shown in the topology

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
Gateway	G0/1	192.168.1.1	255.255.255.0	N/A
	S0/0/1	209.165.201.18	255.255.255.252	N/A
ISP	S0/0/0 (DCE)	209.165.201.17	255.255.255.252	N/A
	Lo0	192.31.7.1	255.255.255.255	N/A
PC-A	NIC	192.168.1.20	255.255.255.0	192.168.1.1
PC-B	NIC	192.168.1.21	255.255.255.0	192.168.1.1
PC-C	NIC	192.168.1.22	255.255.255.0	192.168.1.1

Part 1: Build the Network and Verify Connectivity

In Part 1, you will set up the network topology and configure basic settings, such as the interface IP addresses, static routing, device access, and passwords.

Step 1: Cable the network as shown in the topology.

Step 2: Configure PC hosts.

Step 3: Initialize and reload the routers and switches.

Step 4: Configure basic settings for each router.

1. Disable DNS lookup.
2. Configure IP addresses for the routers as listed in the Addressing Table.
3. Set the clock rate to **128000** for DCE serial interface.
4. Configure device name as shown in the topology.
5. Assign **cisco** as the console and vty passwords.
6. Assign **class** as the encrypted privileged EXEC mode password.

7. Configure **logging synchronous** to prevent console messages from interrupting the command entry.

Step 5: Configure static routing.

1. Create a static route from the ISP router to the Gateway router.

```
ISP(config)# ip route 209.165.200.224 255.255.255.248
209.165.201.18
```

2. Create a default route from the Gateway router to the ISP router.

```
Gateway(config)# ip route 0.0.0.0 0.0.0.0
```

```
209.165.201.17
```

Step 6: Verify network connectivity.

1. From the PC hosts, ping the G0/1 interface on the Gateway router. Troubleshoot if the pings are unsuccessful.
2. Verify that the static routes are configured correctly on both routers.

Part 2: Configure and Verify NAT Pool Overload

In Part 2, you will configure the Gateway router to translate the IP addresses from the 192.168.1.0/24 network to one of the six usable addresses in the 209.165.200.224/29 range.

Step 1: Define an access control list that matches the LAN private IP addresses.

ACL 1 is used to allow the 192.168.1.0/24 network to be translated.

```
Gateway(config)# access-list 1 permit 192.168.1.0
```

```
0.0.0.255
```

Step 2: Define the pool of usable public IP addresses.

```
Gateway(config)# ip nat pool public_access 209.165.200.225
209.165.200.230 netmask 255.255.255.248
```

Step 3: Define the NAT from the inside source list to the outside pool.

```
Gateway(config)# ip nat inside source list 1 pool public_access
```

```
overload
```

Step 4: Specify the interfaces.

Issue the **ip nat inside** and **ip nat outside** commands to the interfaces.

```
Gateway(config)# interface g0/1
Gateway(config-if)# ip nat inside
Gateway(config-if)#         interface         s0/0/1
Gateway(config-if)# ip nat outside
```

Step 5: Verify the NAT pool overload configuration.

1. From each PC host, ping the 192.31.7.1 address on the ISP router.
2. Display NAT statistics on the Gateway router.

```
Gateway# show ip nat statistics
Total active translations: 3 (0 static, 3 dynamic; 3
extended)
Peak translations: 3, occurred
00:00:25 ago Outside interfaces:
  Serial0/0/1
Inside
interfaces:
GigabitEthern
et0/1
Hits: 24  Misses: 0
CEF Translated packets: 24, CEF Punted
packets: 0  Expired translations: 0
Dynamic mappings:
-- Inside Source
[Id: 1] access-list 1 pool public_accessrefcount3
poolpublic_access: netmask 255.255.255.248start
209.165.200.225 end 209.165.200.230type generic, total
addresses 6, allocated 1 (16%), misses 0
Total doors: 0
Appl
door
s: 0
Norm
al
door
s: 0
Queued Packets: 0
```

c. Display NATs on the Gateway router.

```
Gateway# show ip nat translations
Pro Inside global      Inside local       Outside local      Outside
global icmp 209.165.200.225:0 192.168.1.20:1     192.31.7.1:1
192.31.7.1:0 icmp 209.165.200.225:1 192.168.1.21:1     192.31.7.1:1
192.31.7.1:1 icmp 209.165.200.225:2 192.168.1.22:1     192.31.7.1:1
192.31.7.1:2
```

Note: Depending on how much time has elapsed since you performed the pings from each PC, you may not see all three translations. ICMP translations have a short timeout value.

How many Inside local IP addresses are listed in the sample output above? _____

How many Inside global IP addresses are listed? _____

How many port numbers are used paired with the Inside global addresses _____

What would be the result of pinging the Inside local address of PC-A from the ISP router?
Why?

_____
_____ _____

Lab # 12: Configuring VLAN using Packet Tracer

Statement Purpose:

1. Configuring VLAN using Layer 2 device
2. Make some simple Packet Tracer scenarios

Activity outcomes:

1. Students will have gained the understanding of Virtual LANs that is partitioned and isolated broadcast domain at layer 2.
2. Students will be able to overcome the broadcast problem in LAN environment.

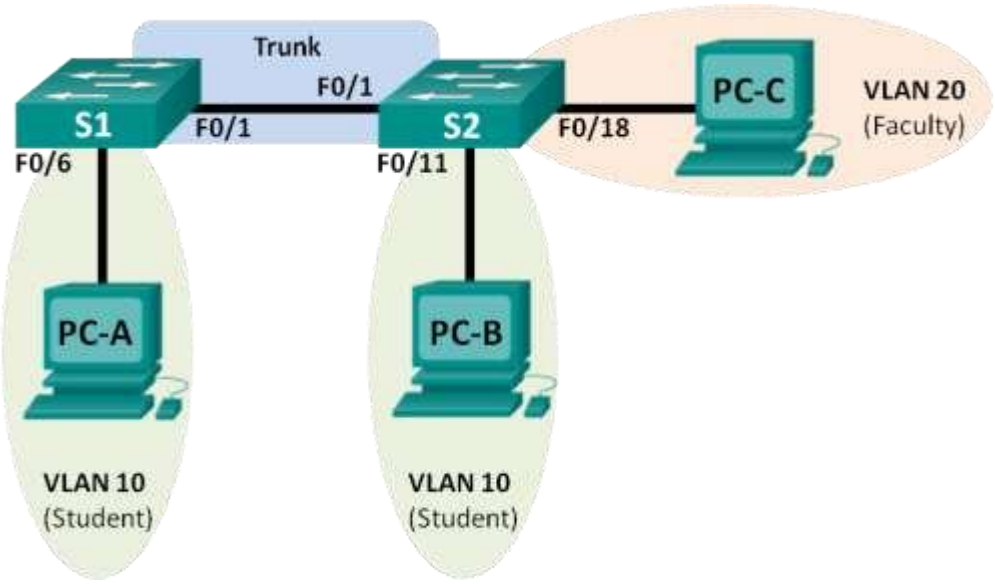
Introduction

Modern switches use virtual local-area networks (VLANs) to improve network performance by separating large Layer 2 broadcast domains into smaller ones. VLANs can also be used as a security measure by controlling which hosts can communicate. In general, VLANs make it easier to design a network to support the goals of an organization. VLAN trunks are used to span VLANs across multiple devices. Trunks allow the traffic from multiple VLANS to travel over a single link, while keeping the VLAN identification and segmentation intact. In this lab, you will create VLANs on both switches in the topology, assign VLANs to switch access ports, verify that VLANs are working as expected, and then create a VLAN trunk between the two switches to allow hosts in the same VLAN to communicate through the trunk, regardless of which switch the host is actually attached to.

Activity 1:

Configuring VLANs (Subnets) using Layer 2 device.

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
S1	VLAN 1	192.168.1.11	255.255.255.0	N/A
S2	VLAN 1	192.168.1.12	255.255.255.0	N/A
PC-A	NIC	192.168.10.3	255.255.255.0	192.168.10.1
PC-B	NIC	192.168.10.4	255.255.255.0	192.168.10.1
PC-C	NIC	192.168.20.3	255.255.255.0	192.168.20.1

Objectives

Part 1: Build the Network and Configure Basic Device Settings

Part 2: Create VLANs and Assign Switch Ports

Part 3: Maintain VLAN Port Assignments and the VLAN Database

Part 4: Configure an 802.1Q Trunk between the

Switches Part Part 5: Delete the VLAN Database

Required Resources

1. 2 Switches (Cisco 2960)
2. 3 PCs (Windows 7, Vista, or XP with terminal emulation program, such as Tera Term/Putty)

3. Console cables to configure the Cisco IOS devices via the console ports
4. Ethernet cables as shown in the topology

Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings on the PC hosts and switches.

Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

Step 2: Initialize and reload the switches as necessary.

Step 3: Configure basic settings for each switch.

1. Disable DNS lookup.
2. Configure device name as shown in the topology.
3. Assign **class** as the privileged EXEC password.
4. Assign **cisco** as the console and vty passwords and enable login for console and vty lines.
5. Configure **logging synchronous** for the console line.
6. Configure a MOTD banner to warn users that unauthorized access is prohibited.
7. Configure the IP address listed in the Addressing Table for VLAN 1 on both switches.
8. Administratively deactivate all unused ports on the switch.
9. Copy the running configuration to the startup configuration.

Step 4: Configure PC hosts.

Refer to the Addressing Table for PC host address information.

Step 5: Test connectivity.

Verify that the PC hosts can ping one another.

Note: It may be necessary to disable the PCs firewall to ping between PCs.

Can PC-A ping PC-B? _____

Can PC-A ping PC-C? _____

Can PC-A ping S1? _____

Can PC-B ping PC-C? _____

Can PC-B ping S2? _____

Can PC-C ping S2? _____

Can S1 ping S2? _____

If you answered no to any of the above questions, why were the pings unsuccessful?

Part 2: Create VLANs and Assign Switch Ports

In Part 2, you will create student, faculty, and management VLANs on both switches. You will then assign the VLANs to the appropriate interface. The **show vlan** command is used to verify your configuration settings.

Step 1: Create VLANs on the switches.

1. Create the VLANs on S1.

   S1(config)# **vlan 10**

   S1(config-vlan)# **name**

   Student

   S1(config-vlan)# **vlan 20**

   S1(config-vlan)# **name**

   Faculty

   S1(config-vlan)# **vlan 99**

   S1(config-vlan)# **name Management**

   S1(config-vlan)# **end**

2. Create the same VLANs on S2.

3. Issue the **show vlan** command to view the list of VLANs on S1.

   S1# **show vlan**

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/1, Gi0/2
10	Student	active	
20	Faculty	active	
99	Management	active	

What is the default VLAN? _____

What ports are assigned to the default VLAN? _____

Step 2: Assign VLANs to the correct switch interfaces.

1. Assign VLANs to the interfaces on S1.

1. Assign PC-A to the Student VLAN.

S1(config)# **interface f0/6**
S1(config-if)# **switchport mode access**

S1(config-if)# **switchport access vlan 10**

2. Move the switch IP address VLAN 99.

S1(config)# **interface vlan 1**
S1(config-if)# **no ip address**
S1(config-if)# **interface vlan 99**
S1(config-if)# **ip address 192.168.1.11 255.255.255.0**
S1(config-if)# **end**

2. Issue the **show vlanbrief** command and verify that the VLANs are assigned to the correct interfaces.

S1# **show vlan brief**

VLAN Name Status Ports
---- ---------- --------- ----------

1. default active Fa0/1, Fa0/2, Fa0/3, Fa0/4
Fa0/5, Fa0/7, Fa0/8, Fa0/9
Fa0/10, Fa0/11, Fa0/12, Fa0/13
Fa0/14, Fa0/15, Fa0/16, Fa0/17
Fa0/18, Fa0/19, Fa0/20,
Fa0/21 Fa0/22, Fa0/23, Fa0/24, Gi0/1
Gi0/2
10 Student active Fa0/6
20 Faculty active
99 Management active

3. Issue the **show ip interfaces brief** command.

What is the status of VLAN 99? Why? _____

1. Use the Topology to assign VLANs to the appropriate ports on S2.
2. Remove the IP address for VLAN 1 on S2.
3. Configure an IP address for VLAN 99 on S2 according to the Addressing Table.

4. Use the **show vlan brief** command to verify that the VLANs are assigned to the correct interfaces.

S2# **show vlan brief**

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/12, Fa0/13 Fa0/14, Fa0/15, Fa0/16, Fa0/17 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gi0/1, Gi0/2
10	Student	active	Fa0/11
20	Faculty	active	Fa0/18
99	Management	active	

Is PC-A able to ping PC-B? Why? _____

Is S1 able to ping S2? Why? _____

Part 3: Maintain VLAN Port Assignments and the VLAN Database

In Part 3, you will change VLAN assignments to ports and remove VLANs from the VLAN database.

Step 1: Assign a VLAN to multiple interfaces.

1. On S1, assign interfaces F0/11 – 24 to VLAN 10.

   S1(config)# **interface range f0/11-24**
   S1(config-if-range)# **switchport mode access**
   S1(config-if-range)# **switchport access vlan 10**
   S1(config-if-range)# **end**

2. Issue the **show vlan brief** command to verify VLAN assignments.
3. Reassign F0/11 and F0/21 to VLAN 20.
4. Verify that VLAN assignments are correct.

Step 2: Remove a VLAN assignment from an interface.

1. Use the **noswitchport access vlan** command to remove the VLAN 10 assignment to F0/24.

   S1(config)# **interface f0/24**

S1(config-if)# **no switchport access vlan**
S1(config-if)# **end**

2. Verify that the VLAN change was made.

Which VLAN is F0/24 is now associated with? _____

Step 3: Remove a VLAN ID from the VLAN database.

1. Add VLAN 30 to interface F0/24 without issuing the VLAN command.

S1(config)# **interface f0/24**
S1(config-if)# **switchport access vlan 30**

2. Verify that the new VLAN is displayed in the VLAN table.

S1# **show vlan brief**

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4
			Fa0/5, Fa0/6, Fa0/7, Fa0/8
			Fa0/9, Fa0/10, Gi0/1, Gi0/2
10	Student	active	Fa0/12, Fa0/13, Fa0/14, Fa0/15
			Fa0/16, Fa0/17, Fa0/18, Fa0/19
			Fa0/20, Fa0/22, Fa0/23
20	Faculty	active	Fa0/11, Fa0/21
30	VLAN0030	active	Fa0/24
99	Management	active	

What is the default name of VLAN 30? _____

1. Use the **no vlan 30** command to remove VLAN 30 from the VLAN database.

S1(config)# **no vlan 30**
S1(config)# **end**

2. Issue the **show vlan brief** command. F0/24 was assigned to VLAN 30.

After deleting VLAN 30, what VLAN is port F0/24 assigned to? What happens to the traffic destined to the host attached to F0/24? _____

S1# **show vlan brief**

VLAN	Name	Status	Ports

---- -------------------------------- --------- -------------------------------

1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4
Fa0/5, Fa0/6, Fa0/7, Fa0/8
Fa0/9, Fa0/10, Gi0/1, Gi0/2

10 Student active Fa0/12, Fa0/13, Fa0/14, Fa0/15
Fa0/16, Fa0/17, Fa0/18, Fa0/19
Fa0/20, Fa0/22, Fa0/23

20 Faculty active Fa0/11, Fa0/21

99 Management active

3. Issue the **no switchport access vlan** command on interface F0/24.

Issue the **show vlan brief** command to determine the VLAN assignment for F0/24. To which

VLAN is F0/24 assigned? _____

Note: Before removing a VLAN from the database, it is recommended that you reassign all the ports assigned to that VLAN.

Why should you reassign a port to another VLAN before removing the VLAN from the VLAN database?

_____
__ _____

Part 4: Configure an 802.1Q Trunk Between the Switches

Note: By default, all VLANs are allowed on a trunk. The **switchport trunk** command allows you to control what VLANs have access to the trunk. For this lab, keep the default settings which allows all VLANs to traverse F0/1.

Manually configure trunk interface F0/1.

The **switchport mode trunk** command is used to manually configure a port as a trunk. This command should be issued on both ends of the link.

1. Change the switchport mode on interface F0/1 to force trunking. Make sure to do this on both switches.

S1(config)# **interface f0/1**
S1(config-if)# **switchport mode trunk**

2. Issue the **show interfaces trunk** command to view the trunk mode. Notice that the mode changed from **desirable** to **on**.

S2# **show interfaces trunk**

Port Mode Encapsulation Status Native vlan Fa0/1 on
802.1q trunking 99

Port Vlans allowed on trunk
Fa0/1 1-4094

Port Vlans allowed and active in management domain
Fa0/1 1,10,20,99

Port Vlans in spanning tree forwarding state and not pruned Fa0/1
1,10,20,99

Part 5: Delete the VLAN Database

In Part 5, you will delete the VLAN Database from the switch. It is necessary to do this when initializing a switch back to its default settings.

Step 1: Determine if the VLAN database exists.

Issue the **show flash** command to determine if a **vlan.dat** file exists in flash.

S1# **show flash**

Directory of flash:/

1. -rwx 1285 Mar 1 1993 00:01:24 +00:00 config.text
2. -rwx 43032 Mar 1 1993 00:01:24 +00:00 multiple-fs
3. -rwx 5 Mar 1 1993 00:01:24 +00:00 private-config.text
4. -rwx 11607161 Mar 1 1993 02:37:06 +00:00 c2960-lanbasek9-mz.150-2.SE.bin
5. -rwx 736 Mar 1 1993 00:19:41 +00:00 vlan.dat

32514048 bytes total (20858880 bytes free)

Note: If there is a **vlan.dat** file located in flash, then the VLAN database does not contain its default settings.

Step 2: Delete the VLAN database.

1. Issue the **delete vlan.dat** command to delete the vlan.dat file from flash and reset the VLAN database back to its default settings. You will be prompted twice to confirm that you want to delete the vlan.dat file. Press Enter both times.

S1# delete vlan.dat
Delete filename [vlan.dat]? Delete
flash:/vlan.dat? [confirm]
S1#

2. Issue the **show flash** command to verify that the vlan.dat file has been deleted.

S1# **show flash**

Directory of flash:/

1. -rwx 1285 Mar 1 1993 00:01:24 +00:00 config.text
2. -rwx 43032 Mar 1 1993 00:01:24 +00:00 multiple-fs
3. -rwx 5 Mar 1 1993 00:01:24 +00:00 private-config.text
4. -rwx 11607161 Mar 1 1993 02:37:06 +00:00 c2960-lanbasek9-mz.150-2.SE.bin

32514048 bytes total (20859904 bytes free)

To initialize a switch back to its default settings, what other commands are needed?

_____

Lab # 13: Configuring WLAN (802.11) using Wireshark and basic Wireless AP setting

Statement Purpose:
Investigate the 802.11 wireless network protocols.

Activity Outcomes:
Students will have better understanding of the 802.11 protocol in terms of its various frames, data transfer mechanism and association/disassociation mechanism.

Introduction

A wireless LAN (WLAN or WiFi) is a data transmission system designed to provide locationindependent network access between computing devices by using radio waves rather than a cable infrastructure. In the corporate enterprise, wireless LANs are usually implemented as the final link between the existing wired network and a group of client computers, giving these users wireless access to the full resources and services of the corporate network across a building or campus setting. The widespread acceptance of WLANs depends on industry standardization to ensure product compatibility and reliability among the various manufacturers.

The 802.11 specification [**IEEE Std 802.11 (ISO/IEC 8802-11: 1999)**] as a standard for wireless LANS was ratified by the Institute of Electrical and Electronics Engineers (IEEE) in the year 1997. This version of 802.11 provides for 1 Mbps and 2 Mbps data rates and a set of fundamental signaling methods and other services. Like all IEEE 802 standards, the 802.11 standards focus on the bottom two levels the ISO model, the physical layer and link layer (see figure below). Any LAN application, network operating system, protocol, including TCP/IP and Novell NetWare, will run on an 802.11-compliant WLAN as easily as they run over Ethernet. The major motivation and benefit from Wireless LANs is increased mobility. Untethered from conventional network connections, network users can move about almost without restriction and access LANs from nearly anywhere.

The other advantages for WLAN include cost-effective network setup for hard-to-wire locations such as older buildings and solid-wall structures and reduced cost of ownership-particularly in dynamic environments requiring frequent modifications, thanks to minimal wiring and installation costs per device and user. WLANs liberate users from dependence on hard-wired access to the network backbone, giving them anytime, anywhere network access. This freedom to roam offers numerous user benefits for a variety of work environments, such as:

1. Immediate bedside access to patient information for doctors and hospital staff
2. Easy, real-time network access for on-site consultants or auditors
3. Improved database access for roving supervisors such as production line managers, warehouse auditors, or construction engineers
4. Simplified network configuration with minimal MIS involvement for temporary setups such as trade shows or conference rooms

5. Faster access to customer information for service vendors and retailers, resulting in better service and improved customer satisfaction
6. Location-independent access for network administrators, for easier on-site troubleshooting and support
7. Real-time access to study group meetings and research links for students

Lab Activities:

Activity 1:

Download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the file Wireshark_802_11.pcap. This trace was collected using AirPcap and Wireshark running on a computer in the home network of one of the authors, consisting of a Linksys 802.11g combined access point/router, with two wired PCs and one wireless host PC attached to the access point/router. The author is fortunate to have other access points in neighboring houses available as well. In this trace file, we'll see frames captured on channel 6. Since the host and AP that we are interested in are not the only devices using channel 6, we'll see a lot of frames that we're not interested in for this lab, such as beacon frames advertised by a neighbor's AP also operating on channel 6. The wireless host activities taken in the trace file are:

•The host is already associated with the 30 Munroe St AP when the trace begins.
•At t = 24.82, the host makes an HTTP request to http://gaia.cs.umass.edu/wiresharklabs/alice.txt. The IP address of gaia.cs.umass.edu is 128.119.245.12.
•At t=32.82, the host makes an HTTP request to http://www.cs.umass.edu, whose IP address is 128.119.240.19.
•At t = 49.58, the host disconnects from the 30 Munroe St AP and attempts to connect to the linksys_ses_24086. This is not an open access point, and so the host is eventually unable to connect to this AP.
•At t=63.0 the host gives up trying to associate with the linksys_ses_24086 AP, and associates again with the 30 Munroe St access point.
Once you have downloaded the trace, you can load it into Wireshark and view the trace using the File pull down menu, choosing Open, and then selecting the Wireshark_802_11.pcap trace file.
Recall that beacon frames are used by an 802.11 AP to advertise its existence. To answer some of the questions below, you'll want to look at the details of the "IEEE 802.11" frame and subfields in the middle Wireshark window. Whenever possible, when answering a question below, you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout27 to explain your answer. To print a packet, use File->Print, choose Selected packet only, choose Packet summary line, and select the minimum amount of packet detail that you need to answer the question.
1. What are the SSIDs of the two access points that are issuing most of the beacon frames in this trace?

2. What are the intervals of time between the transmissions of the beacon frames the linksys_ses_24086 access point? From the 30 Munroe St. access point? (Hint: this interval of time is contained in the beacon frame itself).
3. What (in hexadecimal notation) is the source MAC address on the beacon frame from 30 Munroe St? Recall from Figure 6.13 in the text that the source, destination, and BSS are three addresses used in an 802.11 frame. For a detailed discussion of the 802.11 frame structure, see section 7 in the IEEE 802.11 standards document (cited above).
4. What (in hexadecimal notation) is the destination MAC address on the beacon frame from 30 Munroe St??
5. What (in hexadecimal notation) is the MAC BSS id on the beacon frame from 30 Munroe St?
6. The beacon frames from the 30 Munroe St access point advertise that the access point can support four data rates and eight additional "extended supported rates." What are these rates?

Activity 2:

Since the trace starts with the host already associated with the AP, let first look at data transfer over an 802.11 association before looking at AP association/disassociation. Recall that in this trace, at *t = 24.82*, the host makes an HTTP request to http://gaia.cs.umass.edu/wireshark-labs/alice.txt. The IP address of gaia.cs.umass.edu is 128.119.245.12. Then, at *t=32.82*, the host makes an HTTP request to http://www.cs.umass.edu.

1. Find the 802.11 frame containing the SYN TCP segment for this first TCP session (that downloads alice.txt). What are three MAC address fields in the 802.11 frame? Which MAC address in this frame corresponds to the wireless host (give the hexadecimal representation of the MAC address for the host)? To the access point? To the first-hop router? What is the IP address of the wireless host sending this TCP segment? What is the destination IP address? Does this destination IP address correspond to the host, access point, first-hop router, or some other network-attached device? Explain.

   Find the 802.11 frame containing the SYNACK segment for this TCP session. What are three MAC address fields in the 802.11 frame? Which MAC address in this frame corresponds to the host? To the access point?To the first-hop router? Does the sender MAC address in the frame correspond to the IP address of the device that sent the TCP segment encapsulated within this datagram?

Lab # 14: SSL using Wireshark

Statement Purpose:

1. Investigate the Secure Sockets Layer (SSL) protocol, focusing on the SSL records sent over a TCP connection.
2. We'll do so by analyzing a trace of the SSL records sent between your host and an ecommerce server.
3. We'll investigate the various SSL record types as well as the fields in the SSL messages.

Activity Outcomes:

Students will gain better understanding of SSL.

Introduction

The Transmission Control Protocol/Internet Protocol (TCP/IP) governs the transport and routing of data over the Internet. Other protocols, such as the HyperText Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP), or Internet Messaging Access Protocol (IMAP), run "on top of" TCP/IP in the sense that they all use TCP/IP to support typical application tasks such as displaying web pages or running email servers. The SSL protocol runs above TCP/IP and below higher-level protocols such as HTTP or IMAP. It uses TCP/IP on behalf of the higher-level protocols, and in the process allows an SSL-enabled server to authenticate itself to an SSL-enabled client, allows the client to authenticate itself to the server, and allows both machines to establish an encrypted connection.

These capabilities address fundamental concerns about communication over the Internet and other TCP/IP networks: SSL server authentication allows a user to confirm a server's identity. SSL-enabled client software can use standard techniques of public-key cryptography to check that a server's certificate and public ID are valid and have been issued by a certificate authority (CA) listed in the client's list of trusted CAs. This confirmation might be important if the user, for example, is sending a credit card number over the network and wants to check the receiving server's identity. SSL client authentication allows a server to confirm a user's identity. Using the same techniques as those used for server authentication, SSL-enabled server software can check that a client's certificate and public ID are valid and have been issued by a certificate authority (CA) listed in the server's list of trusted CAs. This confirmation might be important if the server, for example, is a bank sending confidential financial information to a customer and wants to check the recipient's identity.

An encrypted SSL connection requires all information sent between a client and a server to be encrypted by the sending software and decrypted by the receiving software, thus providing a high degree of confidentiality. Confidentiality is important for both parties to any private transaction. In addition, all data sent over an encrypted SSL connection is protected with a mechanism for

detecting tampering-that is, for automatically determining whether the data has been altered in transit. The SSL protocol includes two sub-protocols: the SSL record protocol and the SSL handshake protocol. The SSL record protocol defines the format used to transmit data. The SSL handshake protocol involves using the SSL record protocol to exchange a series of messages between an SSL-enabled server and an SSL-enabled client when they first establish an SSL connection.

This exchange of messages is designed to facilitate the following actions:

1. Authenticate the server to the client.
2. Allow the client and server to select the cryptographic algorithms, or ciphers, that they both support.
3. Optionally authenticate the client to the server.
4. Use public-key encryption techniques to generate shared secrets.
5. Establish an encrypted SSL connection.
6. For more information about the handshake process, see "The SSL Handshake."

Lab Activities:

Activity 1:

The first step is to capture the packets in an SSL session. To do this, you should go to your favorite e-commerce site and begin the process of purchasing an item (but terminating before making the actual purpose!). After capturing the packets with Wireshark, you should set the filter so that it displays only the Ethernet frames that contain SSL records sent from and received by your host. (An SSL record is the same thing as an SSL message.) You should obtain something like screenshot on the previous page.

Your Wireshark GUI should be displaying only the Ethernet frames that have SSL records. It is important to keep in mind that an Ethernet frame may contain one or more SSL records. (This is very different from HTTP, for which each frame contains either one complete HTTP message or a portion of a HTTP message.) Also, an SSL record may not completely fit into an Ethernet frame, in which case multiple frames will be needed to carry the record. Whenever possible, when answering a question below, you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout28 to explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line,* and select the minimum amount of packet detail that you need to answer the question

1. For each of the first 8 Ethernet frames, specify the source of the frame (client or server), determine the number of SSL records that are included in the frame, and list the SSL record types that are included in the frame. Draw a timing diagram between client and server, with one arrow for each SSL record.
2. Each of the SSL records begins with the same three fields (with possibly different values). One of these fields is “content type” and has length of one byte. List all three fields and their lengths.

Activity 2:

ClientHello Record:

1. Expand the ClientHello record. (If your trace contains multiple ClientHello records, expand the frame that contains the first one.) What is the value of the content type?

2. Does the ClientHello record contain a nonce (also known as a “challenge”)? If so, what is the value of the challenge in hexadecimal notation?

3. Does the ClientHello record advertise the cyber suites it supports? If so, in the first listed suite, what are the public-key algorithm, the symmetric-key algorithm, and the hash algorithm?

Lab # 15: Edge Firewall TMG (Threat Management Gateway) Installation and Configuration

Statement Purpose:

1. Connecting servers and clients virtually using Virtual Machine's (VM's). Using the features of Edge Firewall Fore Front TMG. The role of firewall for an organization is important to stop unwanted traffic.

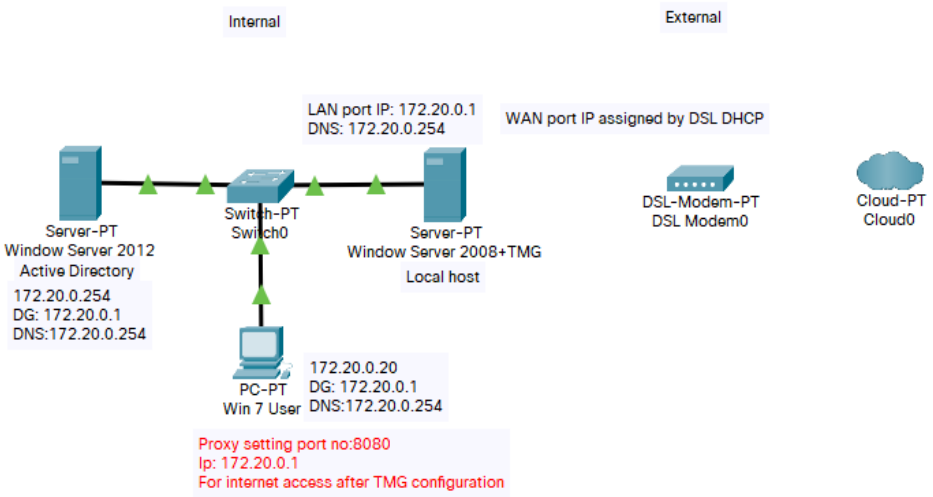
Activity outcomes:

2. Students will gain knowledge about virtualization, VM's and will know about the firewall that how a firewall works within organization.

3. After this lab student will develop basic understanding about virtualization and using firewall from server point of view restricting users from different activities.

Instructor Note:

1. In this lab you will learn about how to use VM's, installation and configurations of domain server and using firewall.



1. Stage J(Journey)

Introduction

Once you will learn to set up a domain network by configuring a server window and making it domain. Then installing different features of server in domain network, connecting sub-domains and clients. As a network administrator you must control whole network within organization. This lab will provide you virtual environment that how through Virtualization you can connect many devices as much you can and applying rules on different groups of clients. In addition, in this lab we will use the feature of firewall on secondary domain.

2. Stage a1 (apply)

Lab Activities:

Getting Oracle VM Virtual Box

To run VM virtual Box you must have a PC or laptop. It will be installed for you on operating systems if not installed visit on following link and install latest version of Oracle Virtual Box.

https://www.virtualbox.org/wiki/Downloads

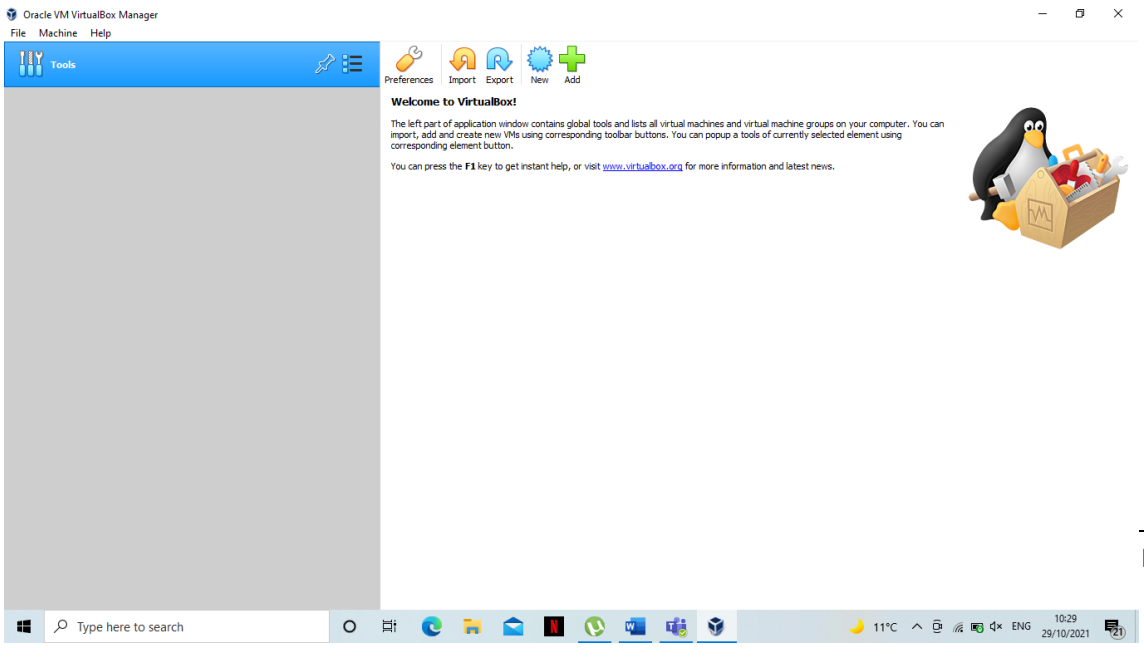
Running VM Virtual Box

Activity 1:

The best way to learn about any new software run it and try it out. We will assume that your computer is connected to internet. For doing your first lab once you install VM Virtual Box, it is not mandatory you will need internet for setting up servers and client. To provide internet facility to firewall, server and client window then you will need internet connection. You will also need ISO images of windows.

STEP 1:

1. Install latest version of Oracle VM Virtual Box.

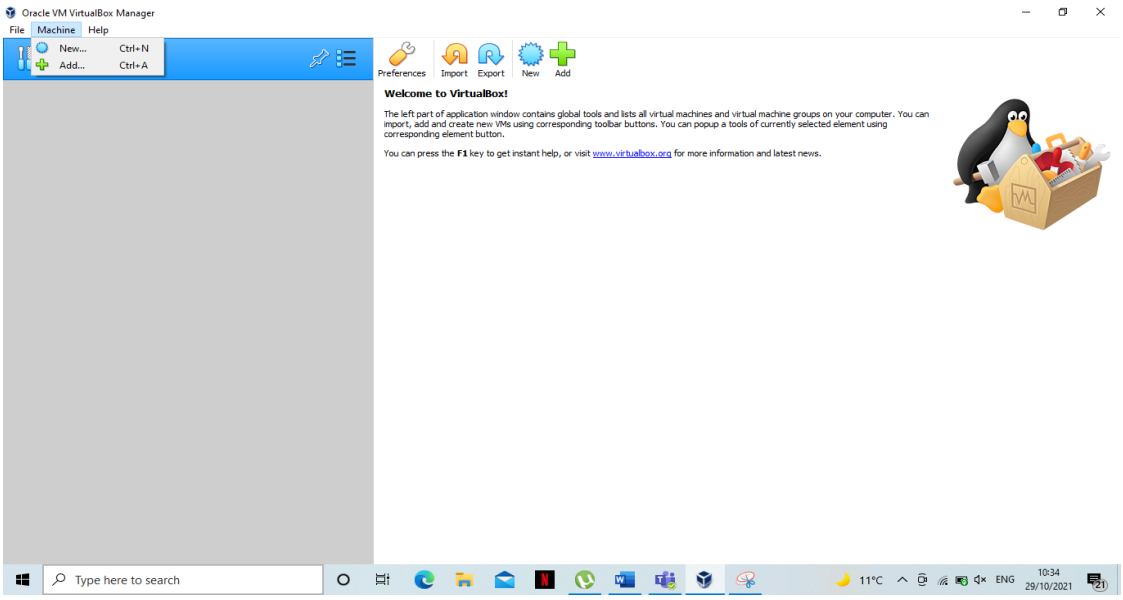


STEP 2:

2. Download ISO images of Window Server 2012, 2008 R2 and client window for users like win7.

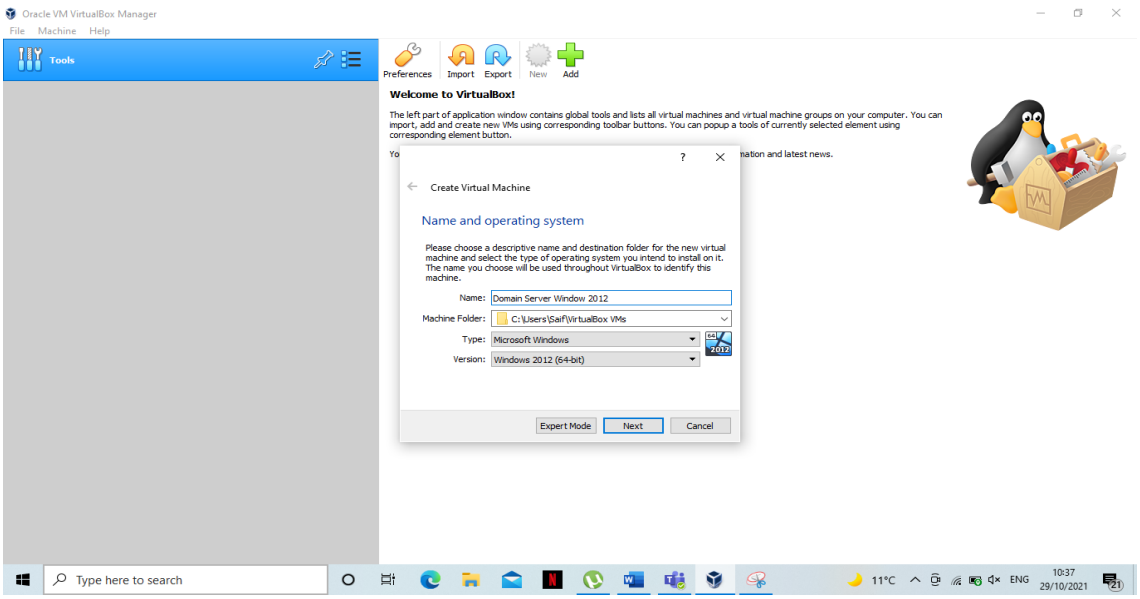
STEP 3:

3. Open VM Virtual Box and click on Machine to create virtual space for Server window.



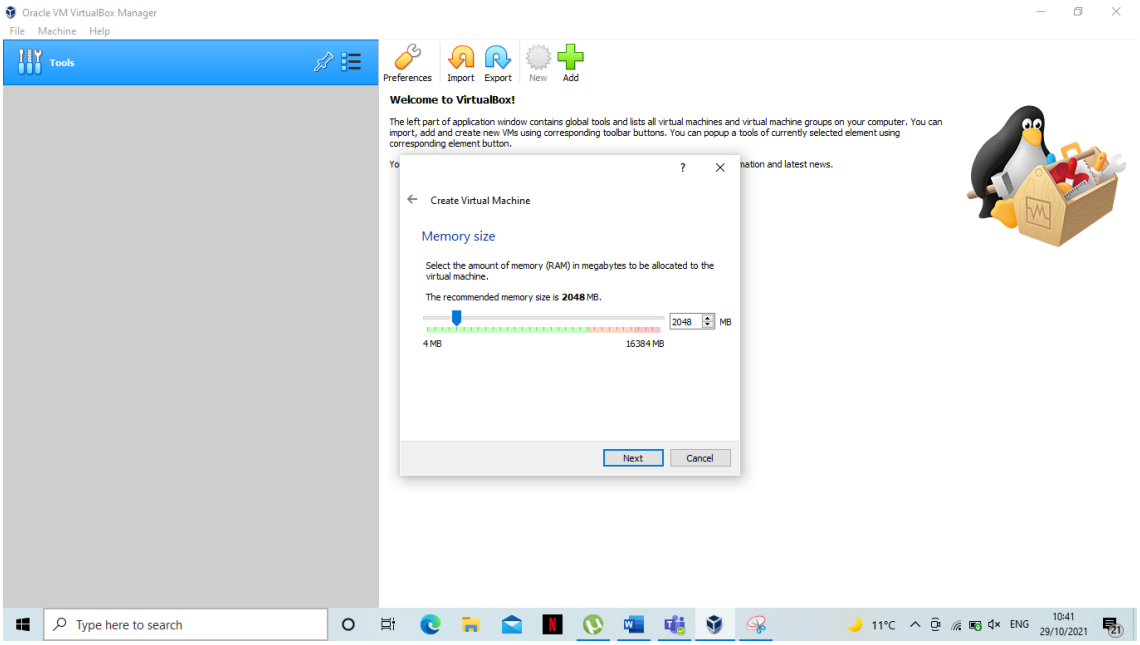
STEP 4:

4. Give proper name to virtual space for domain which will be window server 2012 and click on next.



STEP 5:

5. Now allocate memory space to Virtual Machine minimum space 2048 MB and click on next.

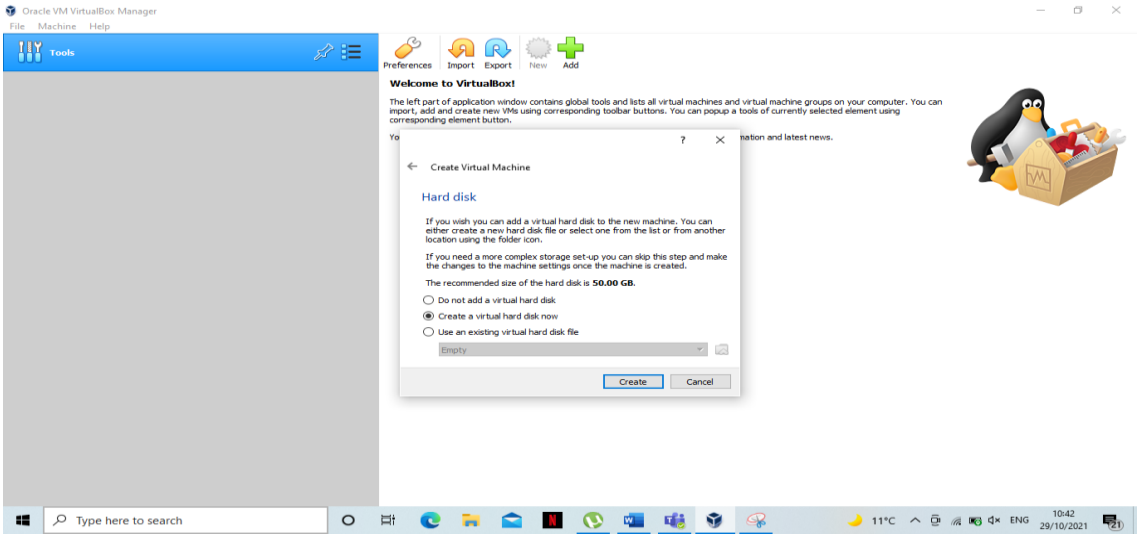


STEP 6:

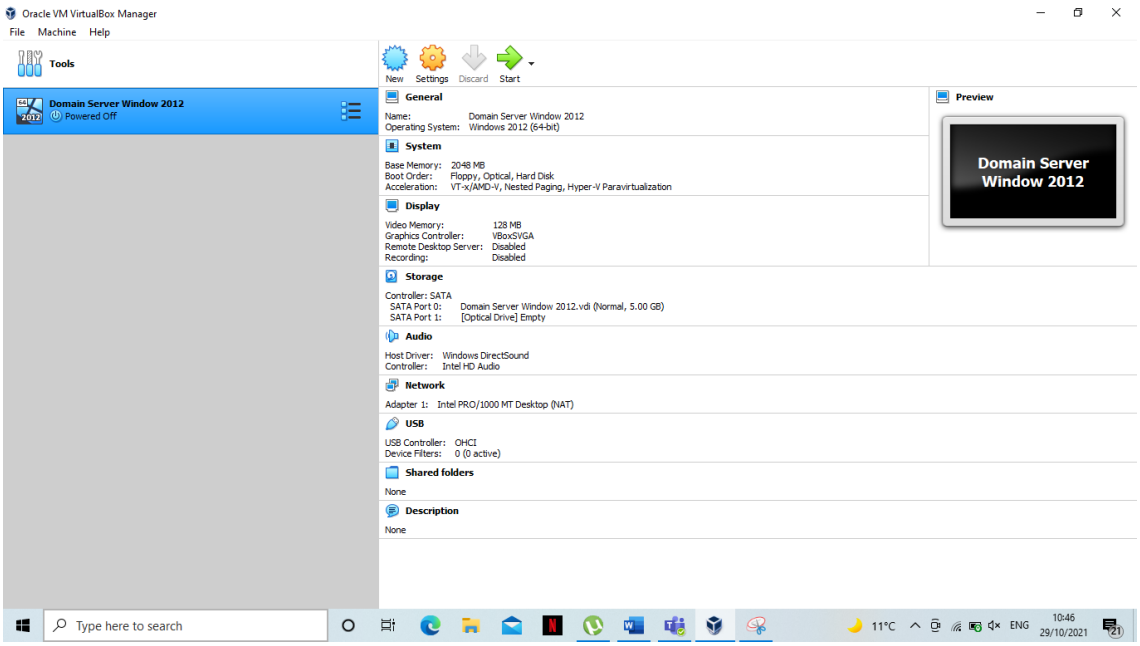
6. Then click on Create a virtual hard drive and click next.

7. Click on VDI option.

8. After this click on Dynamically allocation space option and allocate minimum 10 GB space for window in hard drive.

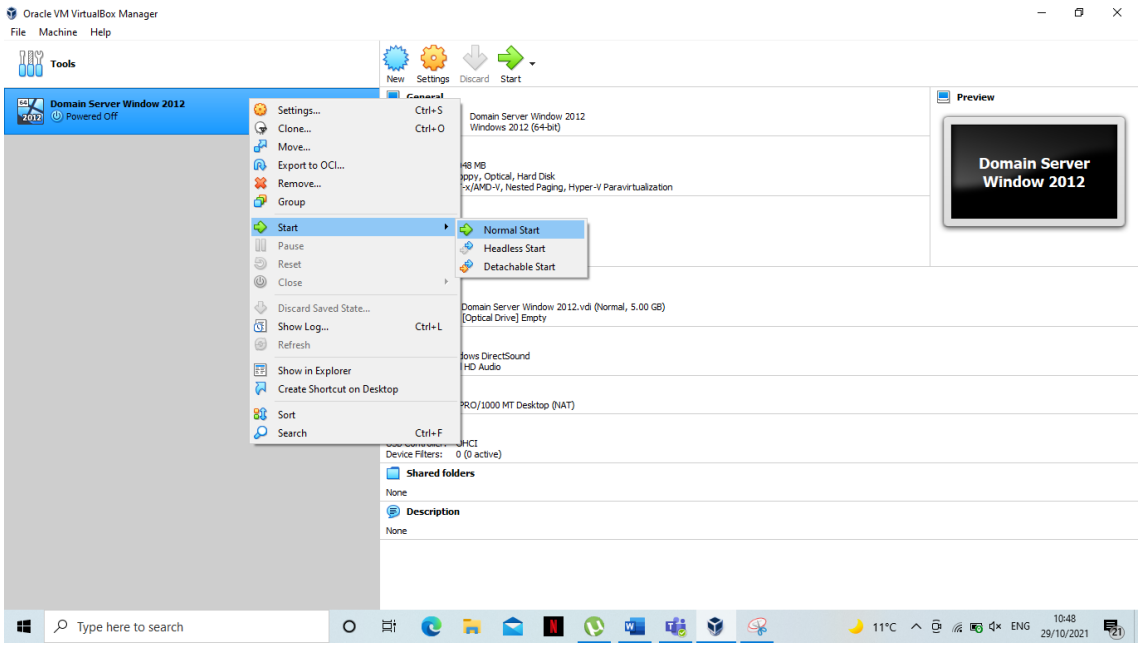


1. Now your VM is ready to install window server by default it is power off.

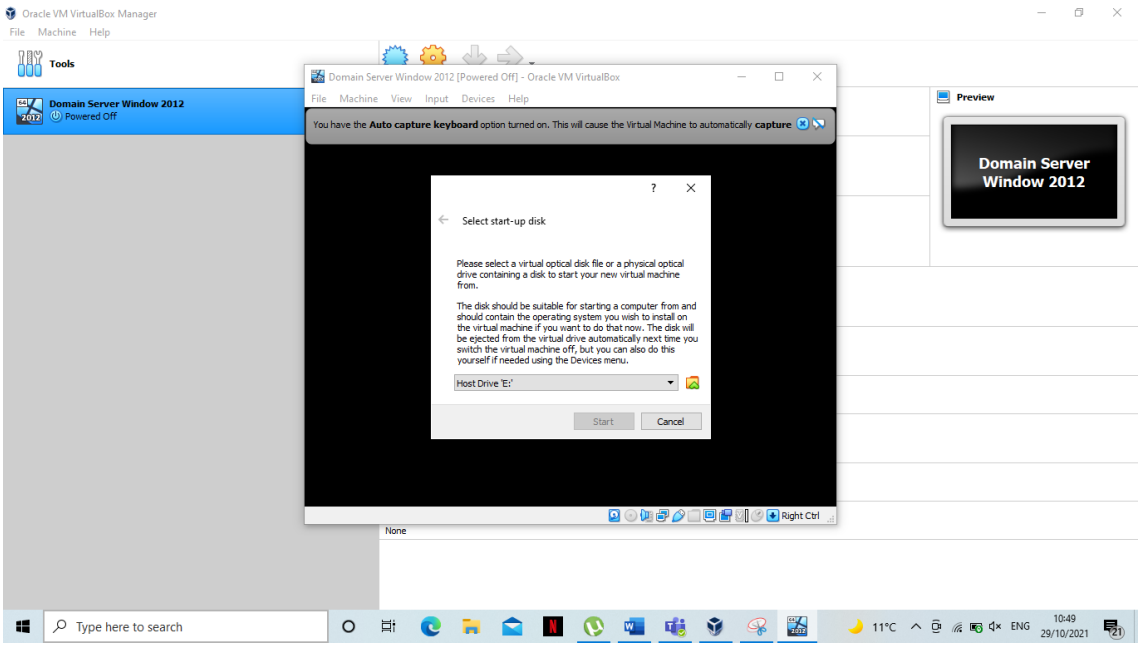


STEP 7:

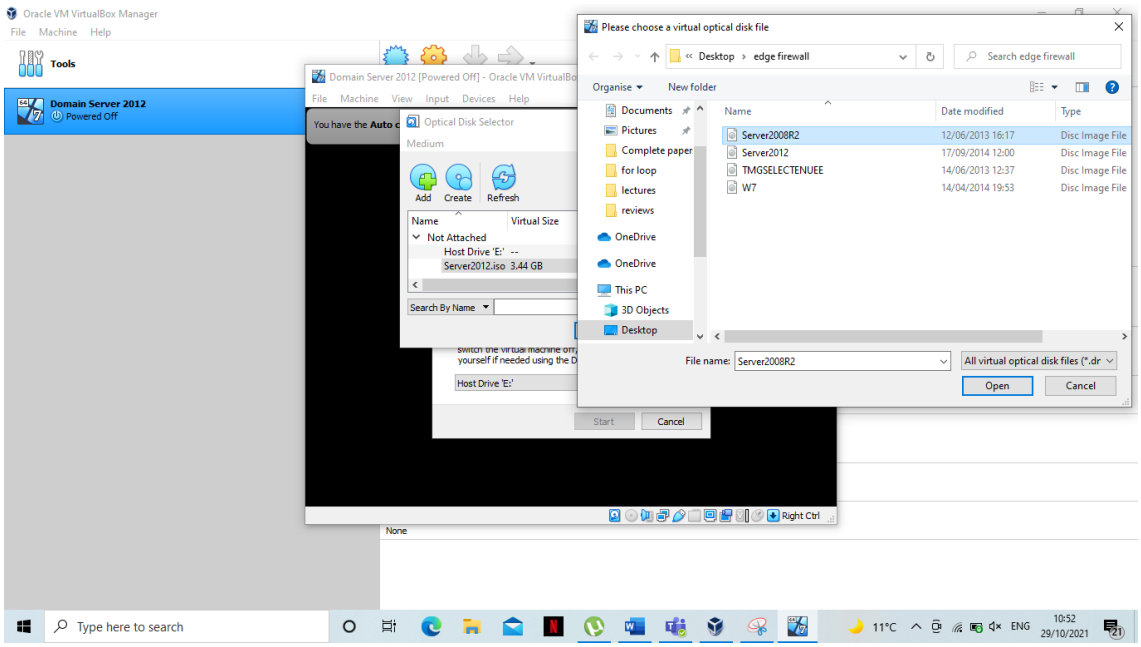
2. Right click on VM space and try to power on it by clicking start and the new start.



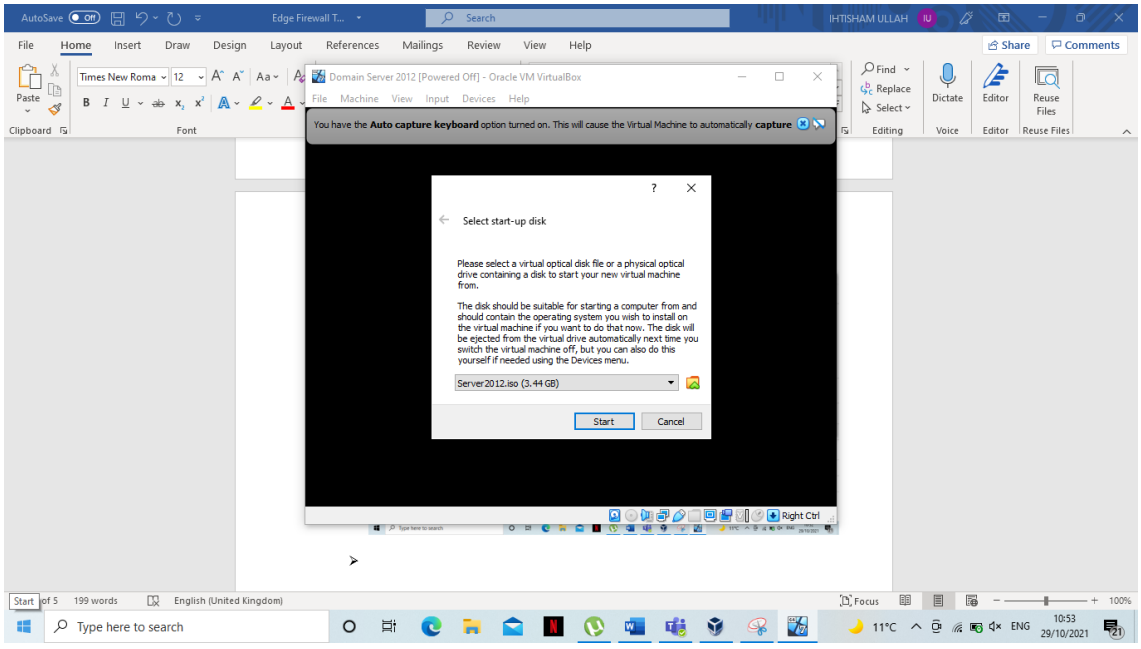
3. Now VM is ready for window Installation.



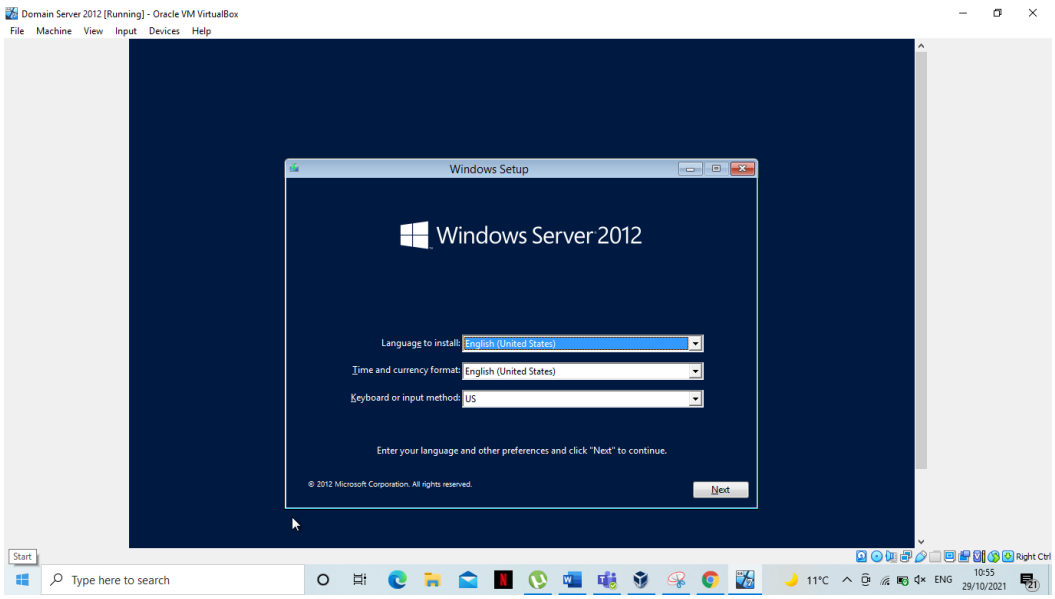
4. Choose ISO image of window server 2012 by clicking on folder then click on add and open it.



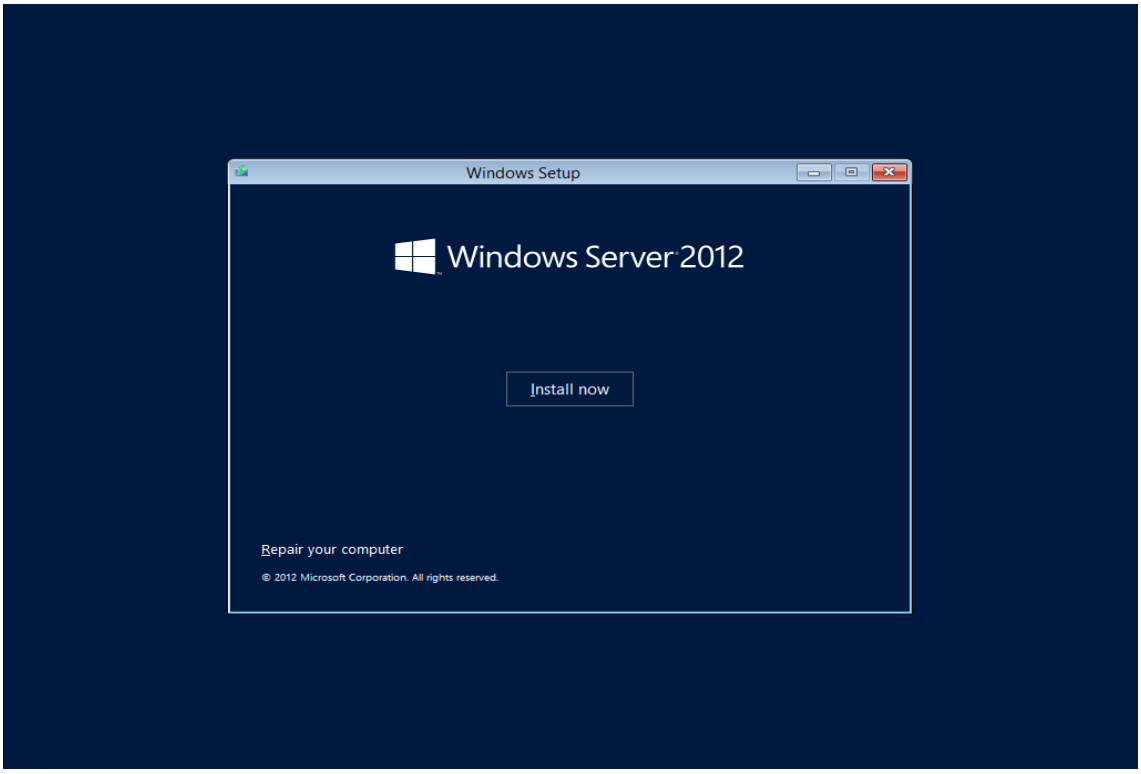
5. Choose ISO image on click on start.



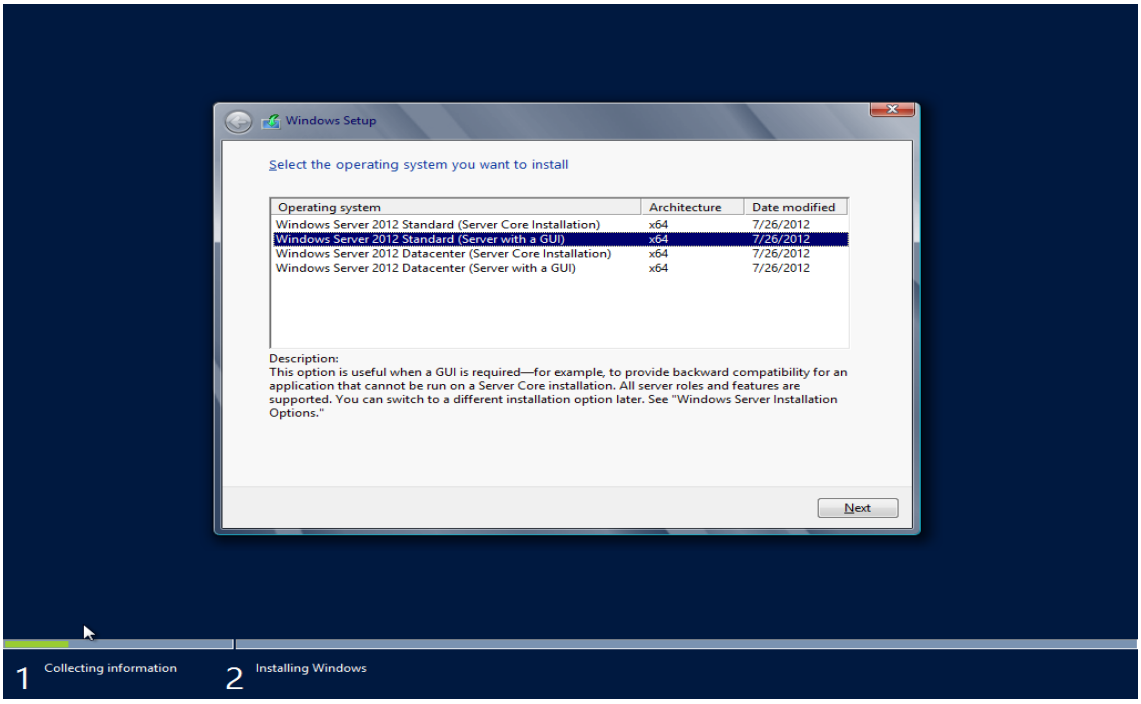
6. Window is ready for installation



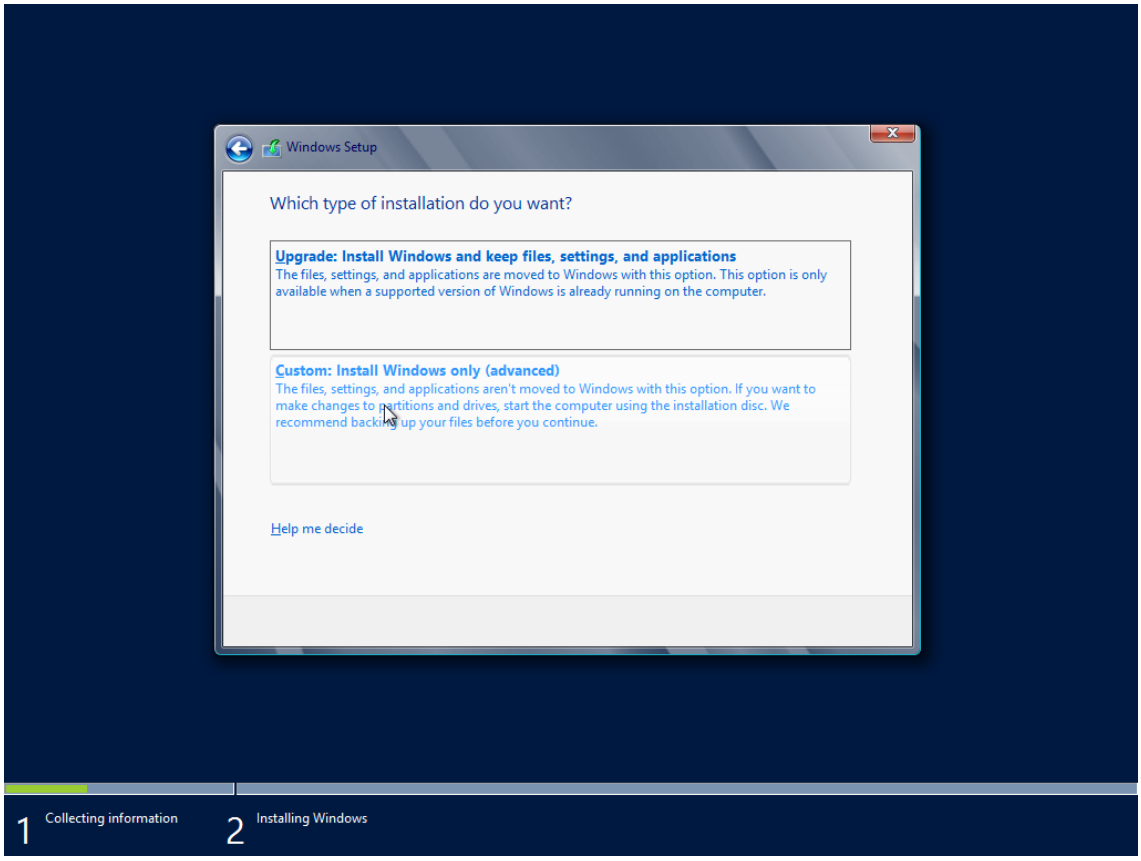
7. For full screen mode click right crtl+F.
8. Click on next and then click on install now.



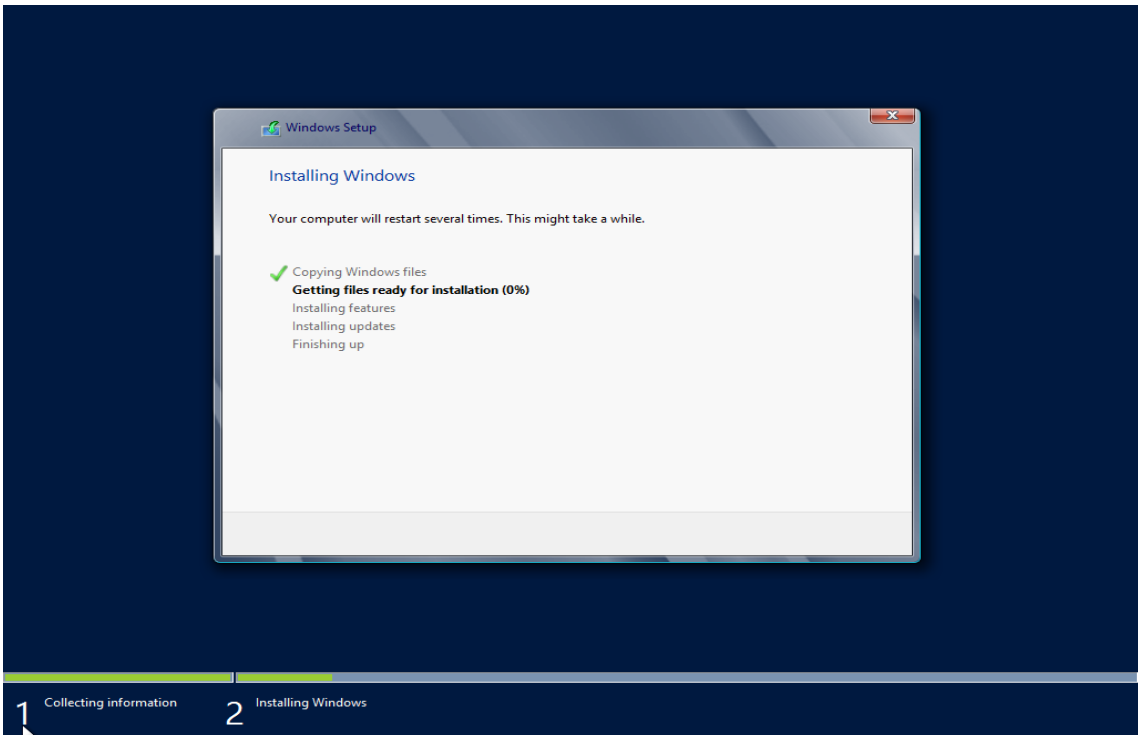
9. Then select window standard server core with GUI option.



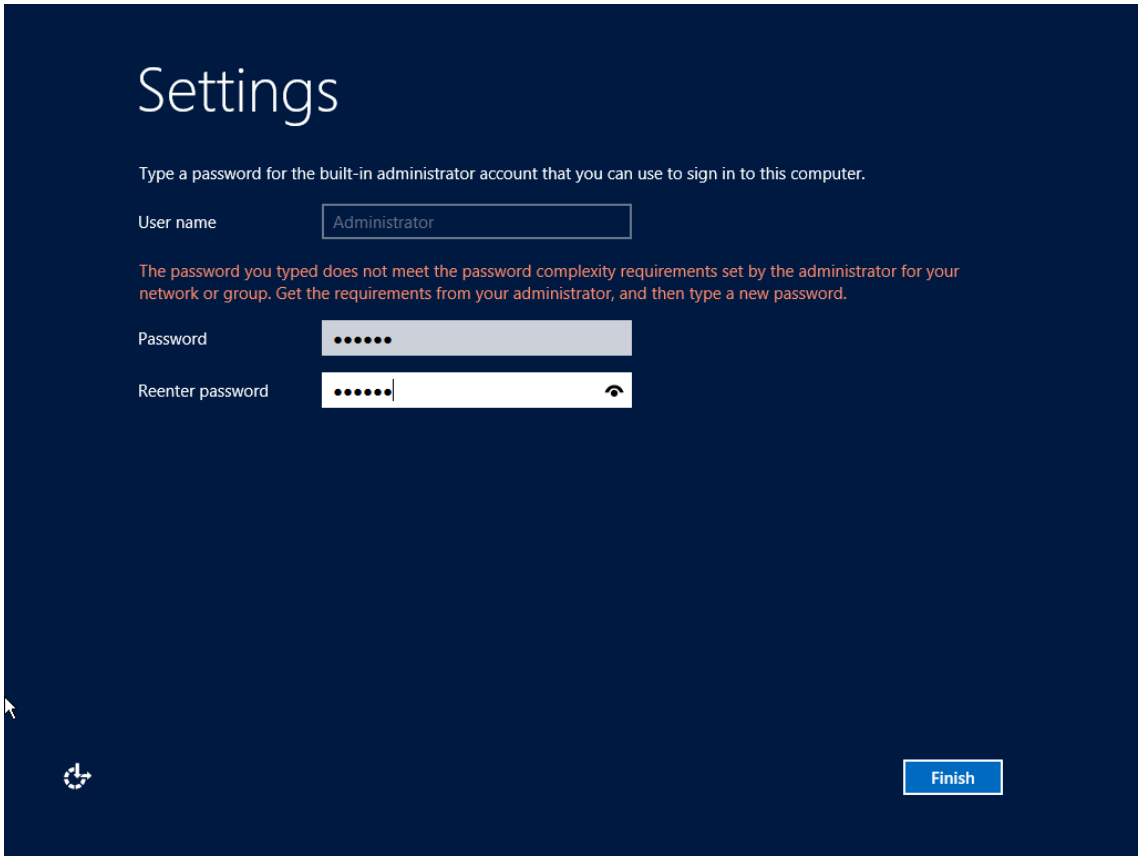
10. Select Custom Installation option.



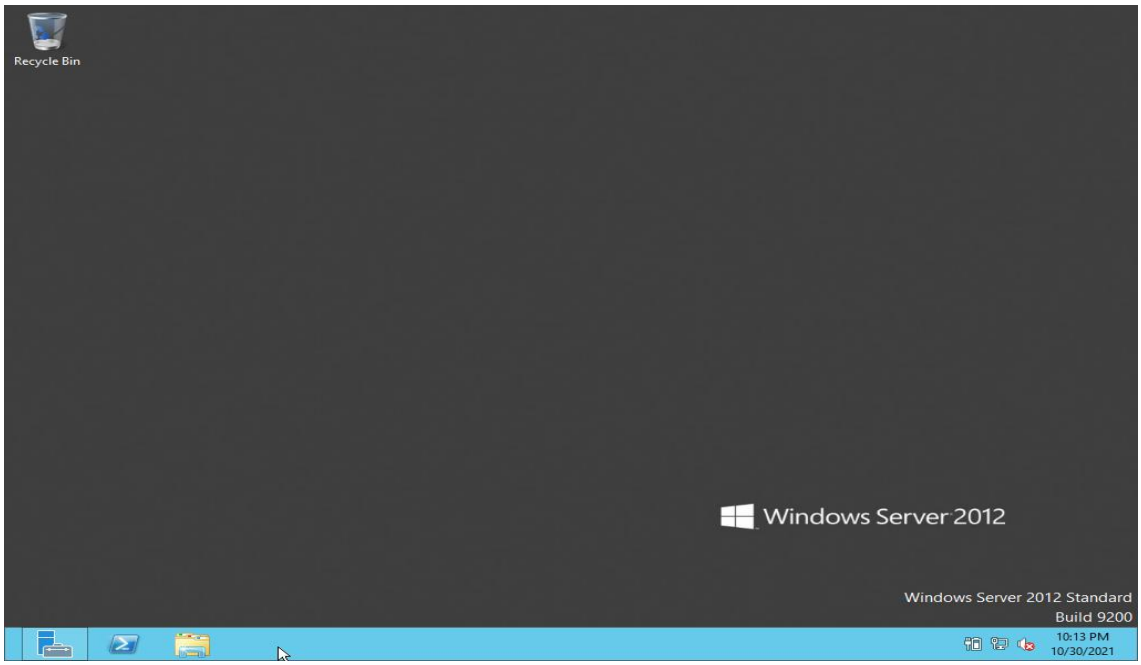
11. Installation gets started.



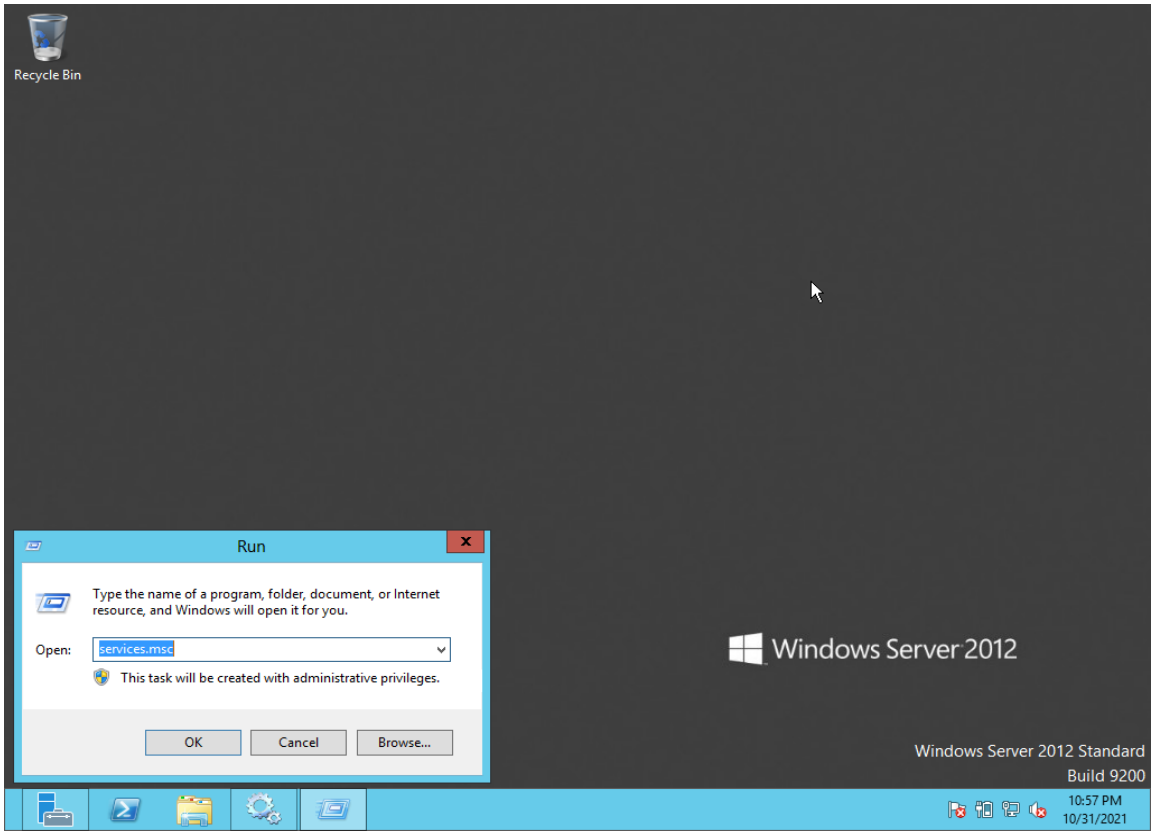
12. After Installation set password.



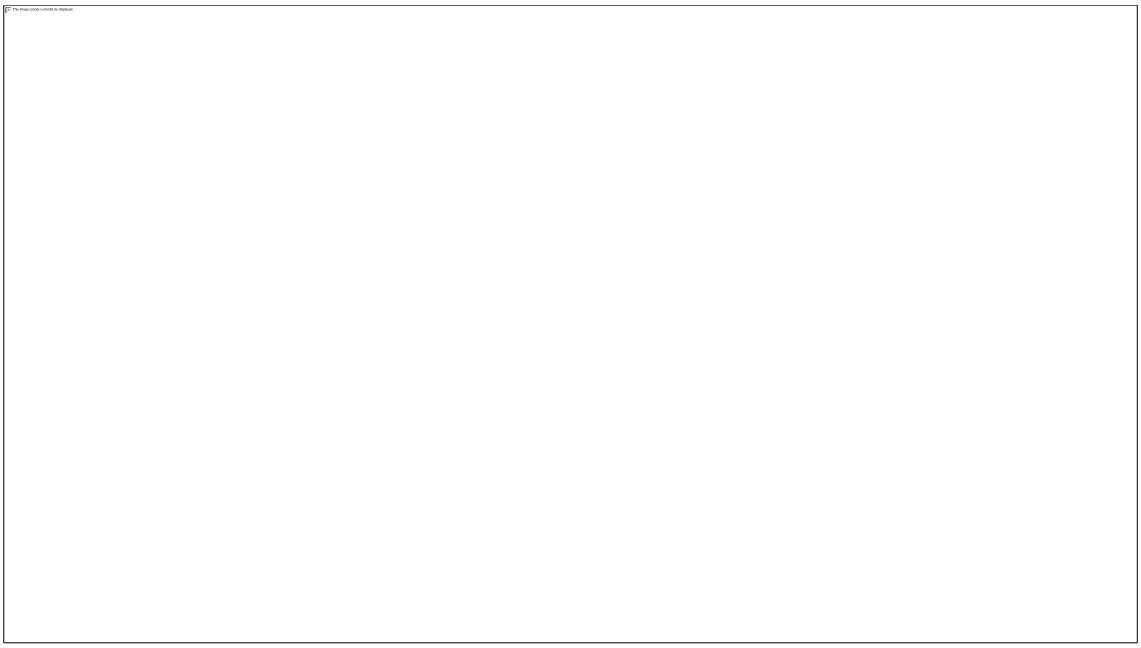
13. Finally window server 2012 is ready for configuration.



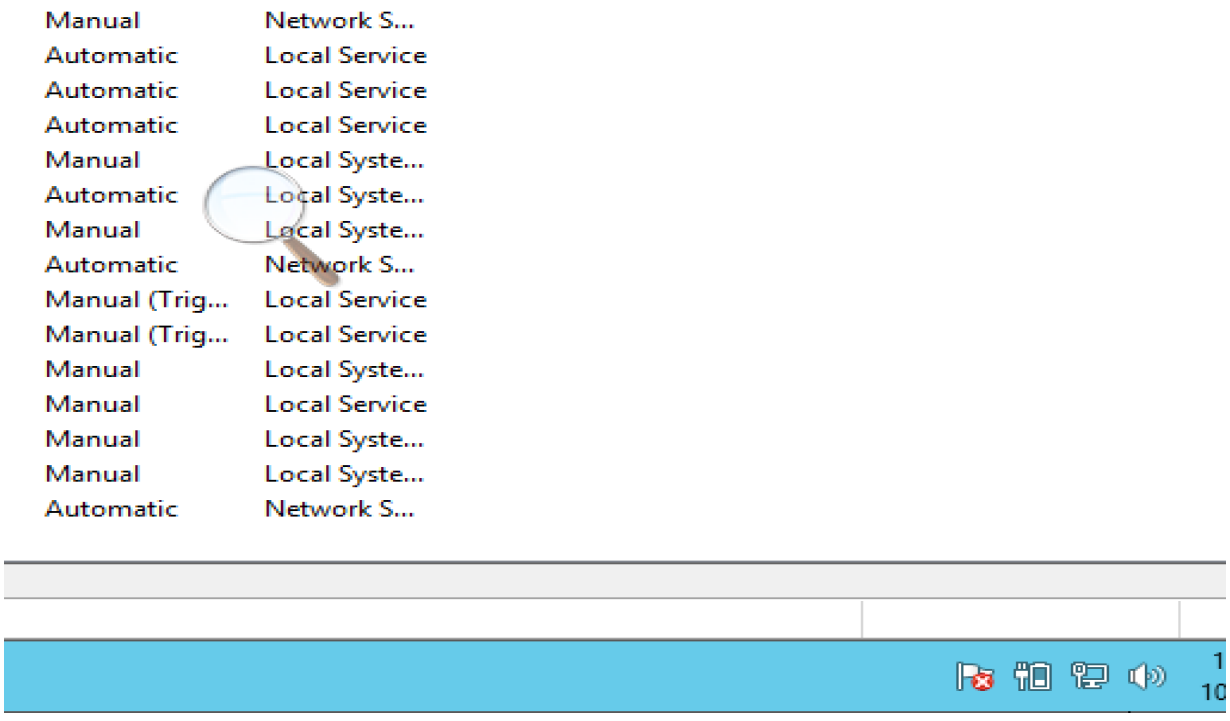
1. Initially in server window the audio is disabled by default. Open Run and type services.msc.



2. Now in servers select option window audio and changed it from manual to automatic.



3. Click on start and after finalizing the setting the audio will be enabled.

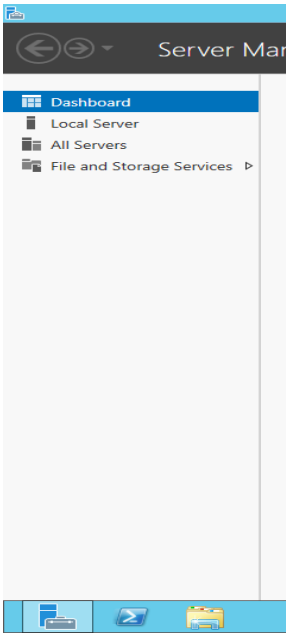


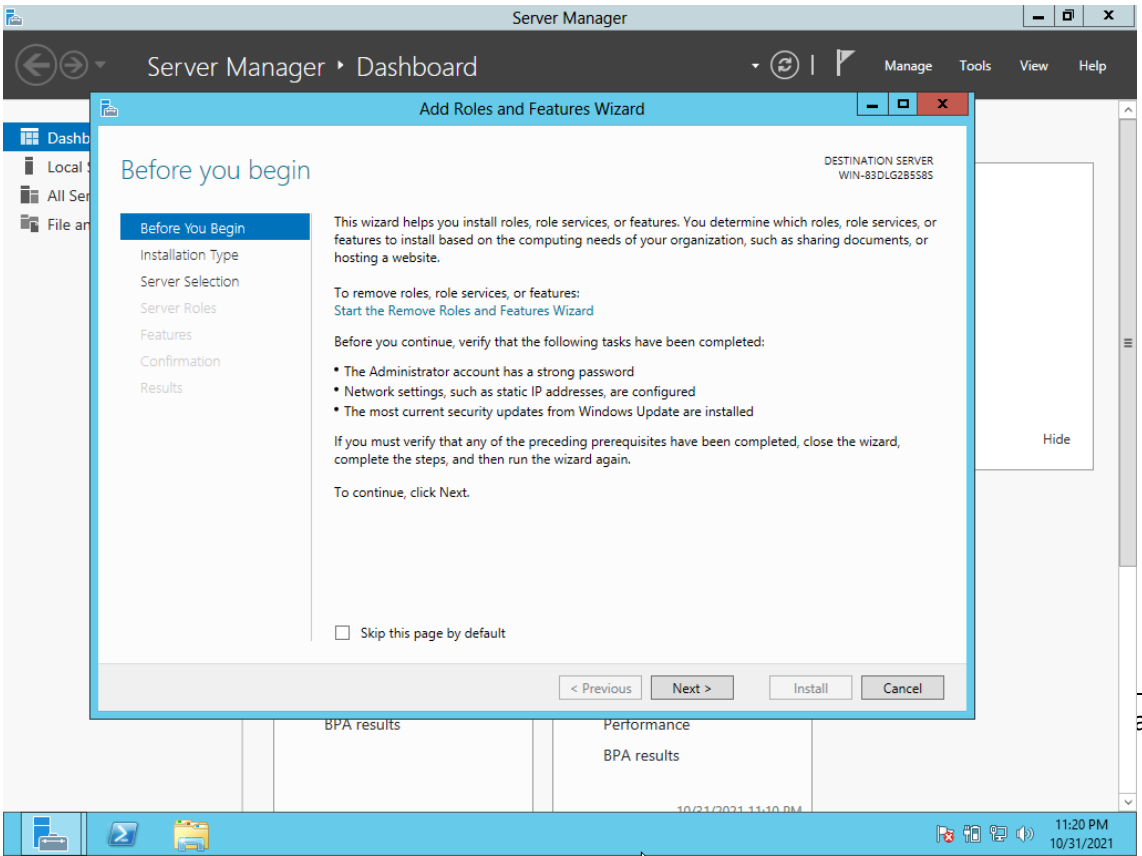
<u>STEP 8</u>: INSTALLING WIRELESS CONNECTION, DESKTOP EXPERIENCE AND ACTIVE DIRECTORY ROLES AND FEAURES

4. First adding roles and features of wireless connection for internet connectivity and desktop experience for adding icons on desktop.

5. Click on server then click on add roles and features.

6. Then after Installation setting up both Active Directory and DNS for domain.

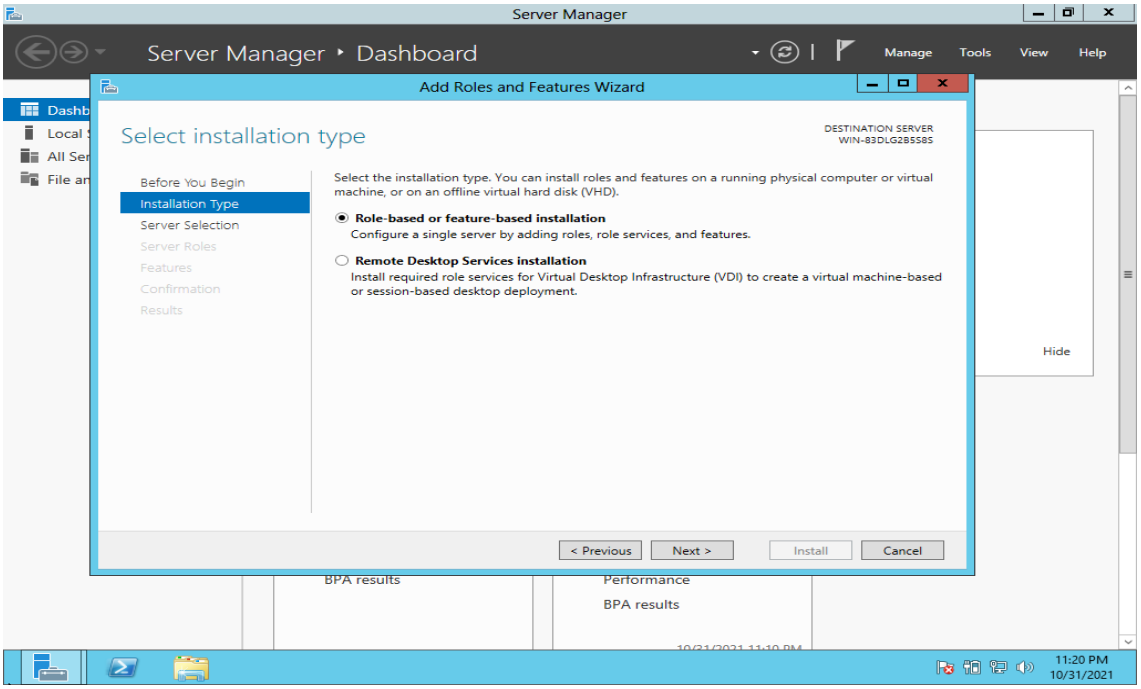




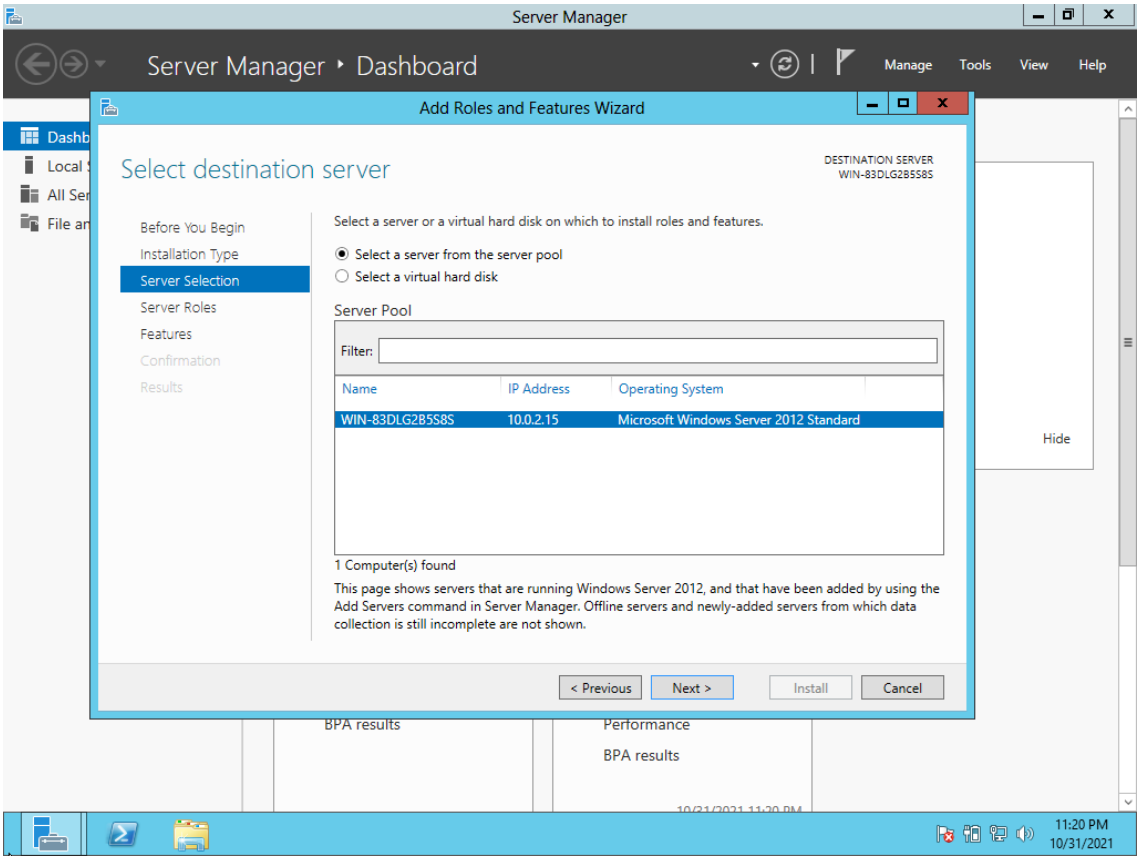
1. Click on next.

2. Now click on Role - based

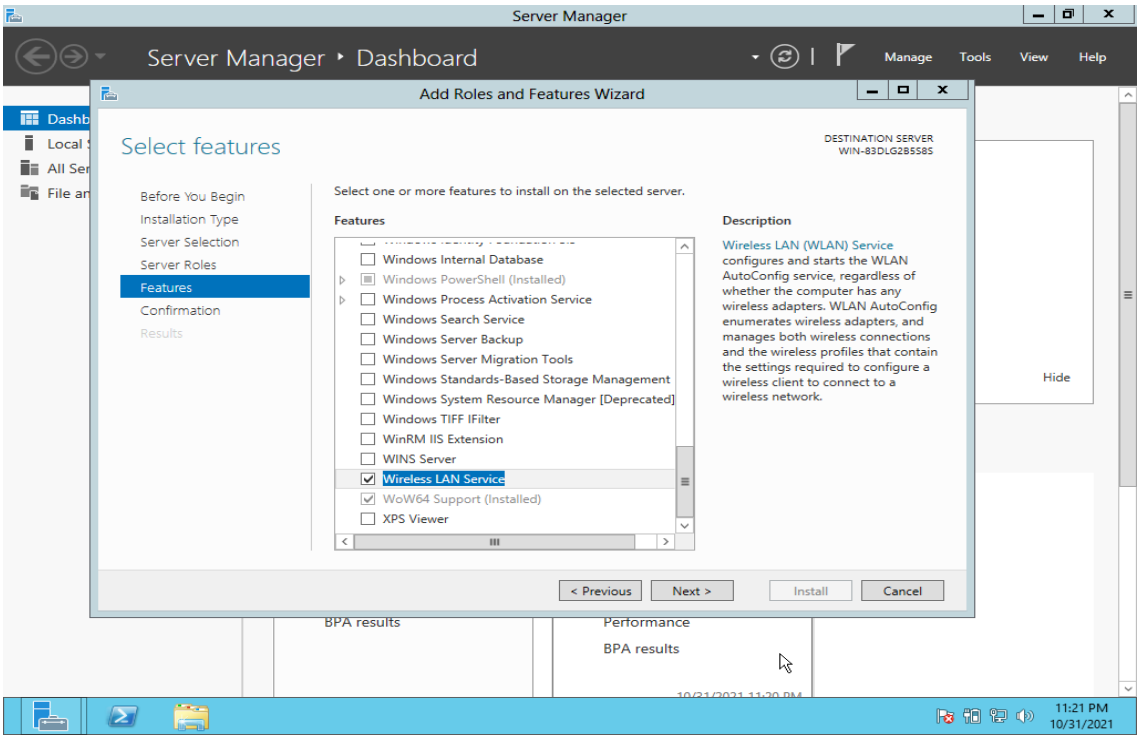
Installation.

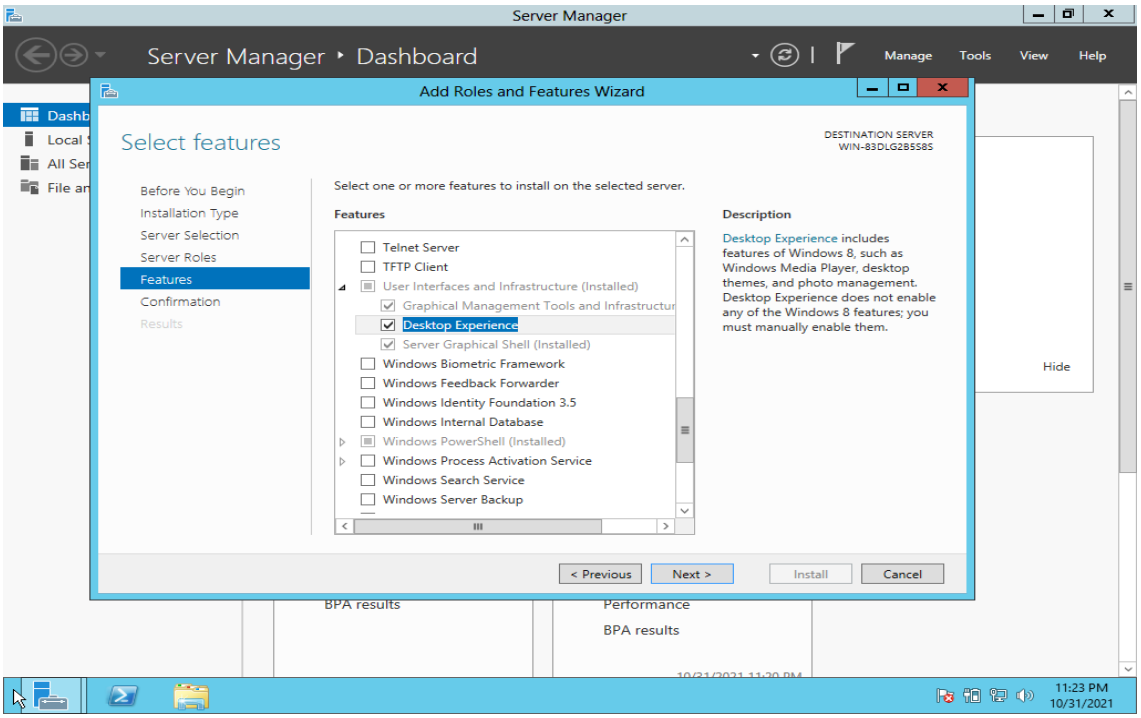


3. Then click on server from pool.

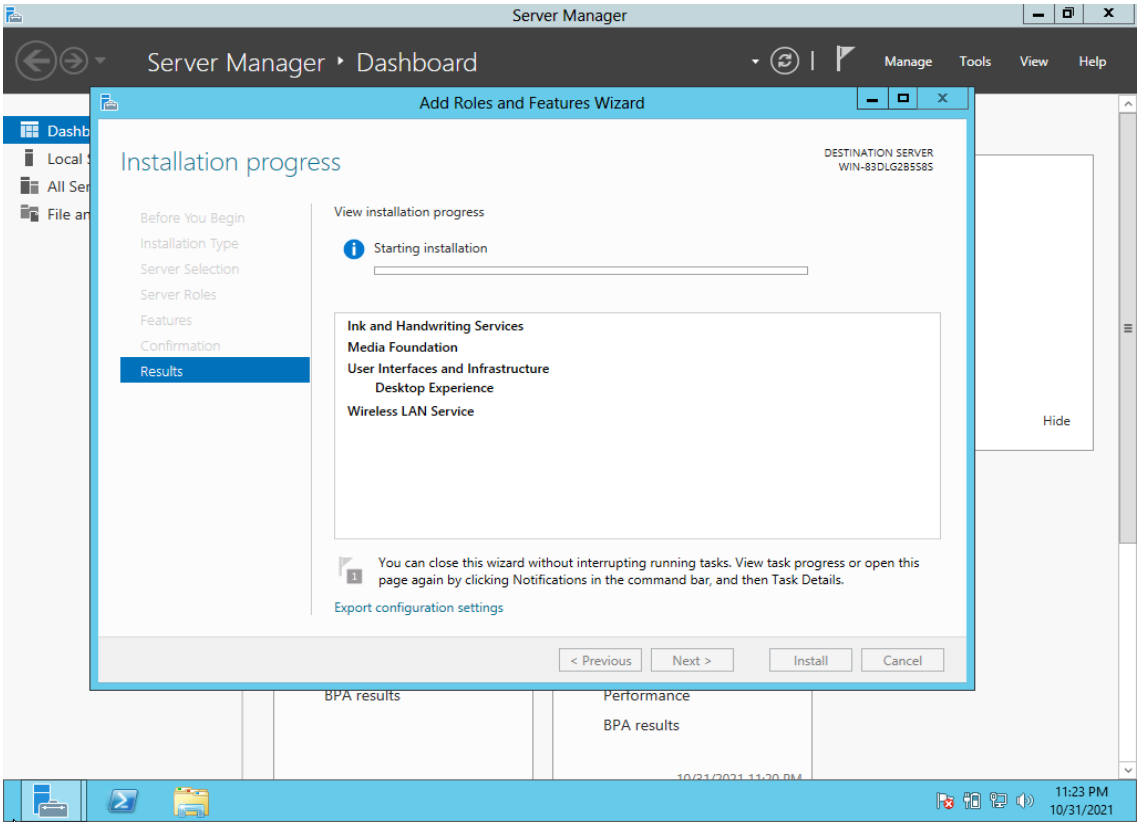


4. In roles and features click on wireless Lan service to enable it and add the features of Desktop experience to enable it.



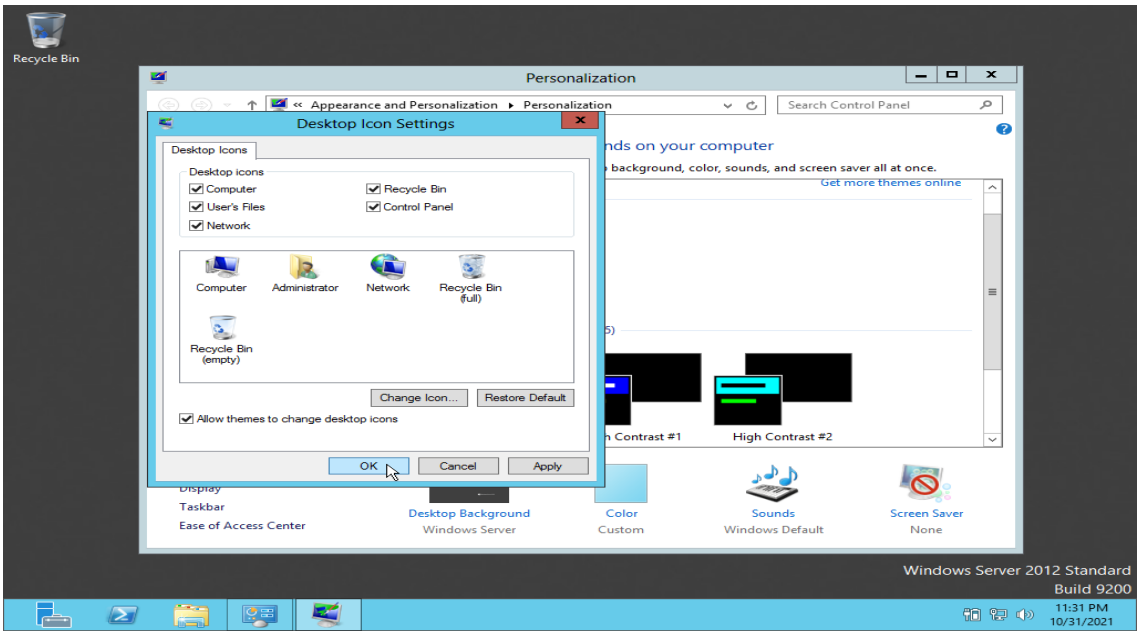


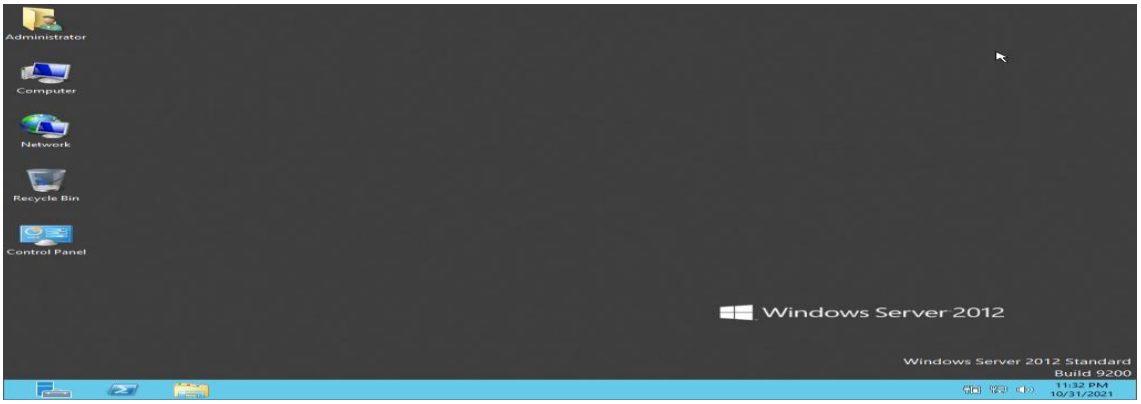
5. Start Installation.



6. After completing installation restart server.

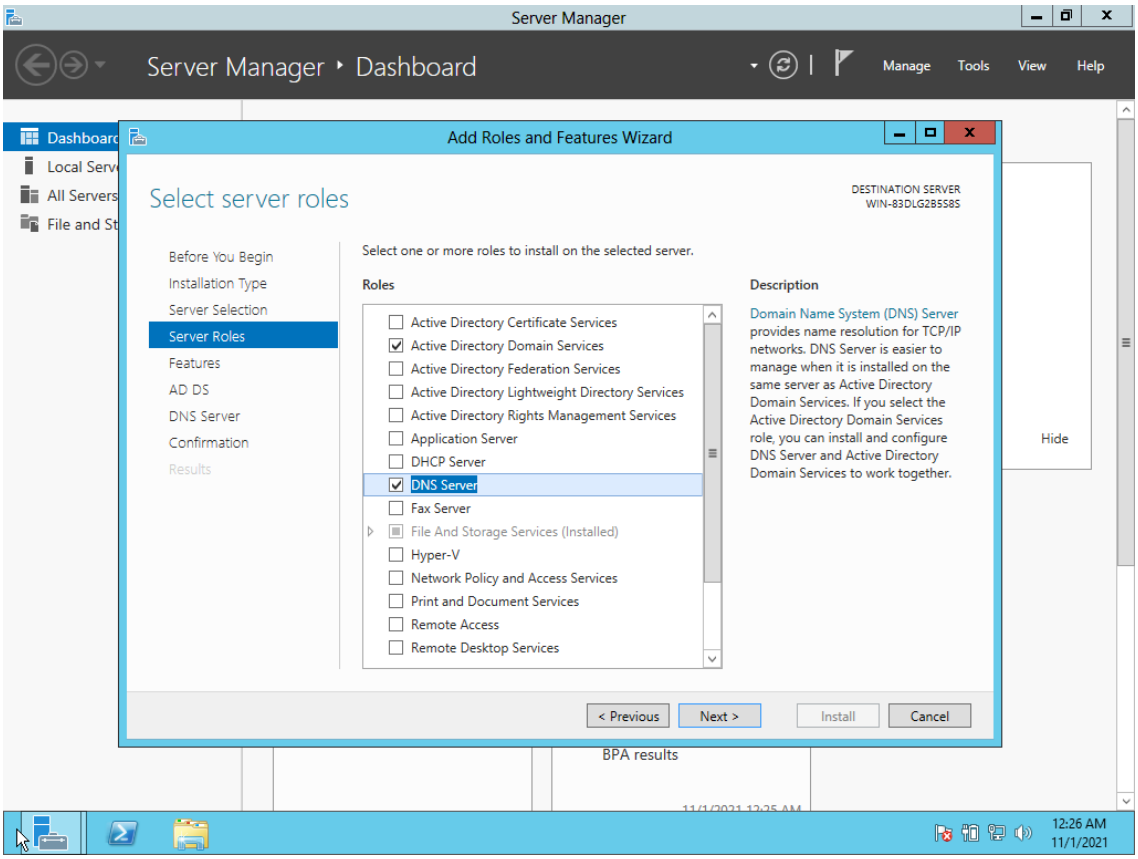
7. Now you can add desktop icons and connect the server to internet.



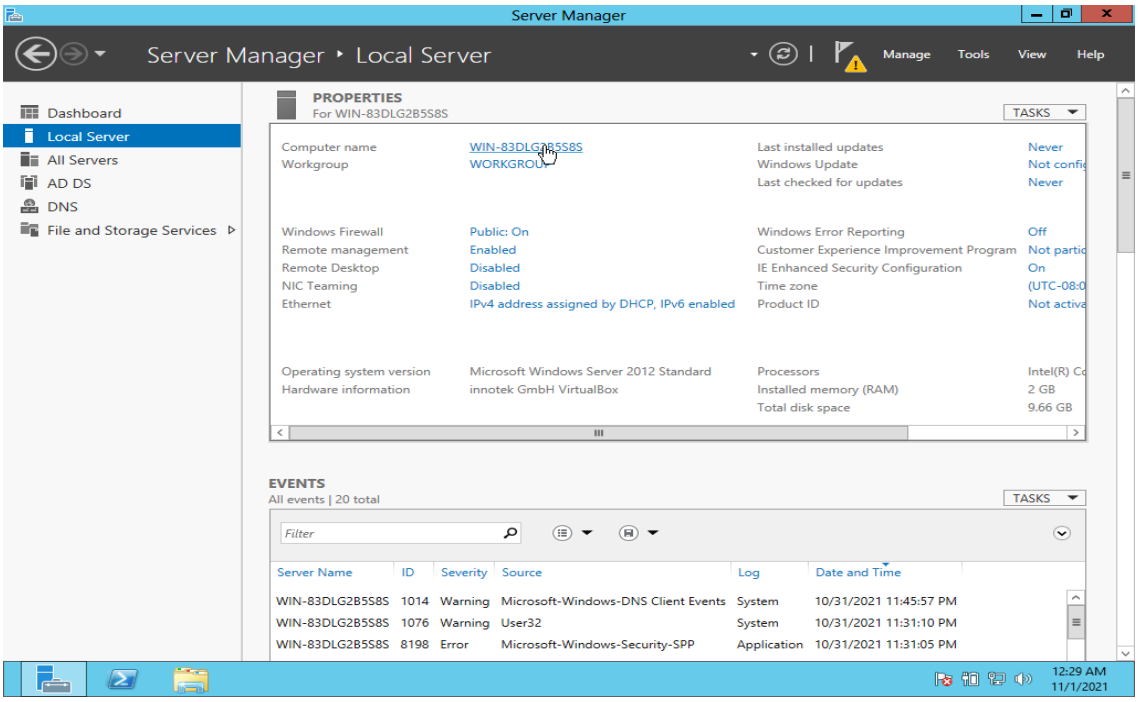


1. Now configuring active directory and DNS.

2. First install active directory and DNS features.

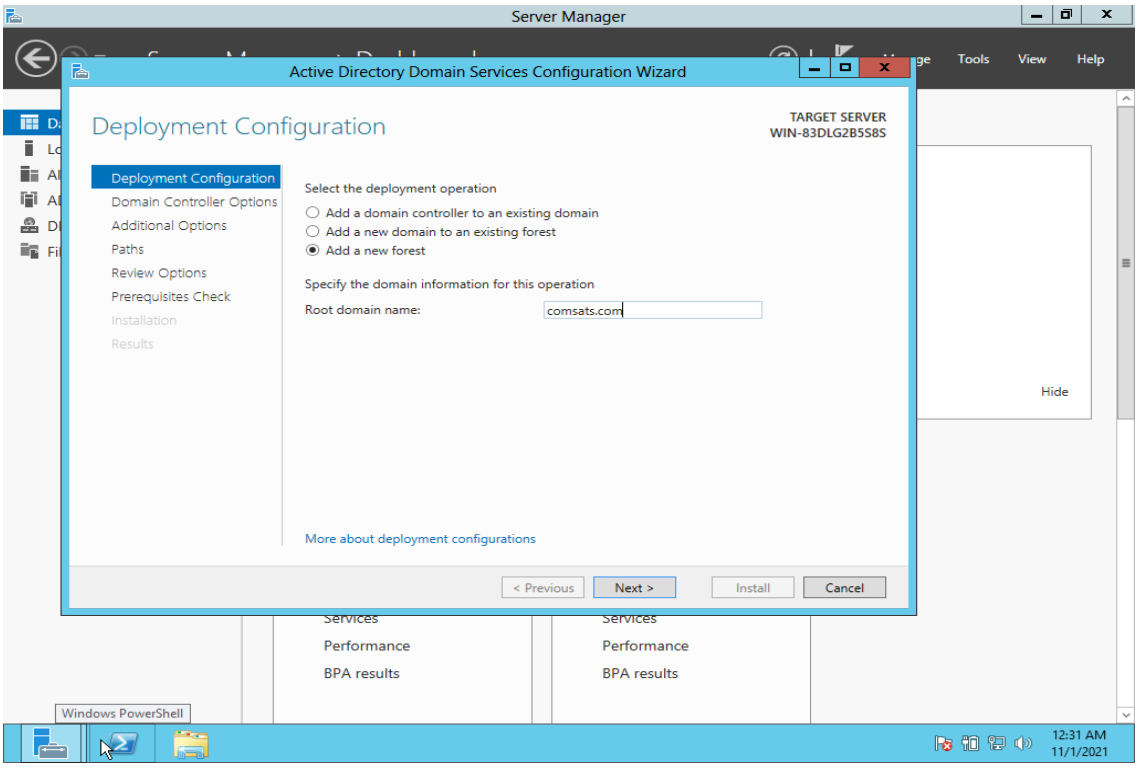


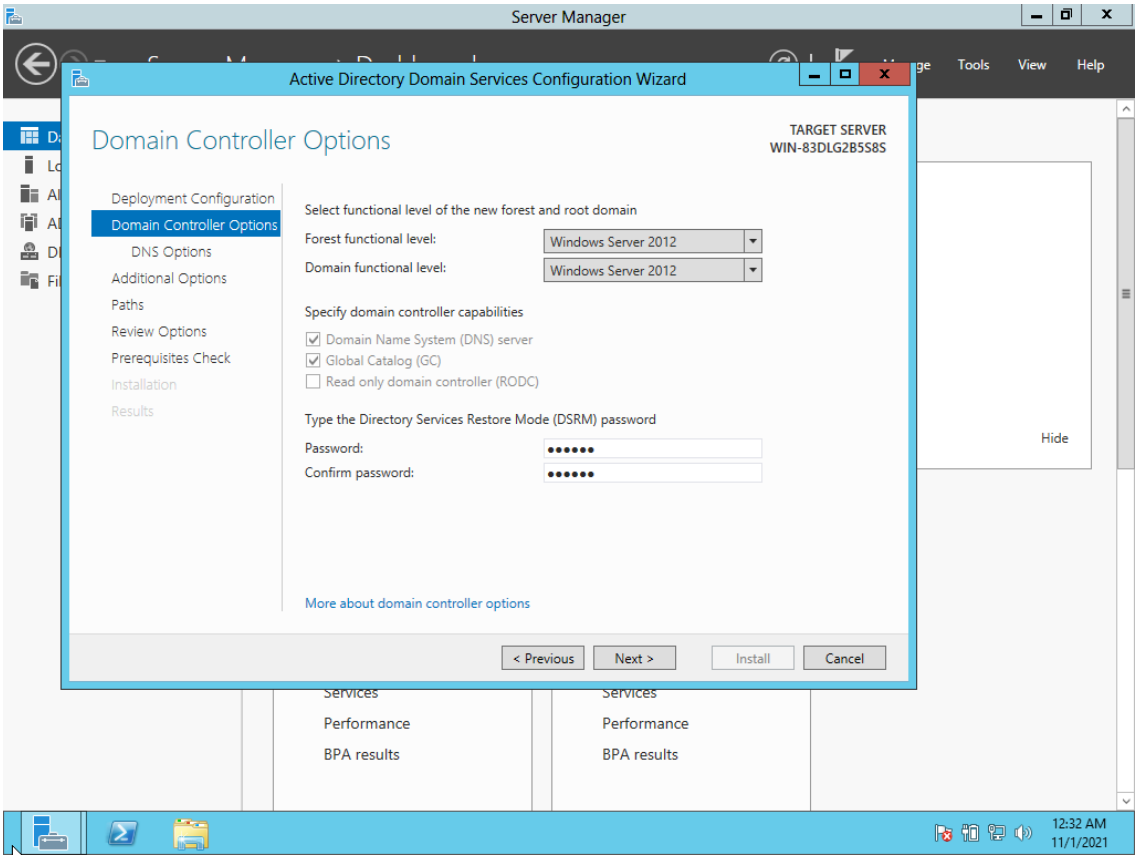
1. After installation change computer name and keep it simple.



2. Now configuring Active directory. Add a new forest and give it name like comsats.com.

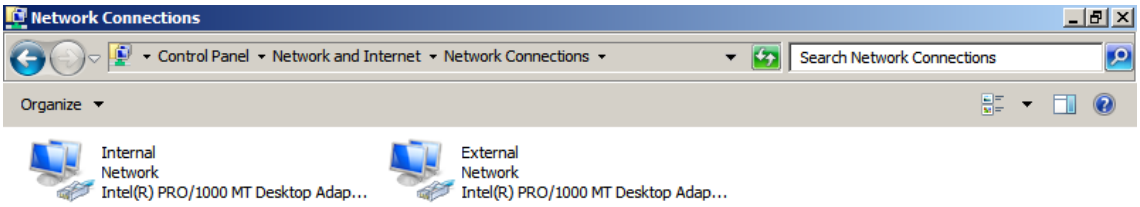
3. Click on next and give it simple password. Finalize Installation and restart server. Also configure DNS.


Server Manager
Tools
View
Help
Active Directory Domain Services Configuration Wizard
Deployment Configuration
TARGET SERVER
WIN-83DLG2B5S8S
Deployment Configuration
Domain Controller Options
Additional Options
Paths
Review Options
Prerequisites Check
Installation
Results
Select the deployment operation
Add a domain controller to an existing domain
Add a new domain to an existing forest
Add a new forest
Specify the domain information for this operation
Root domain name:
comsats.com
More about deployment configurations
< Previous
Next >
Install
Cancel
Hide
Services
Performance
BPA results
Services
Performance
BPA results
Windows PowerShell
12:31 AM
11/1/2021

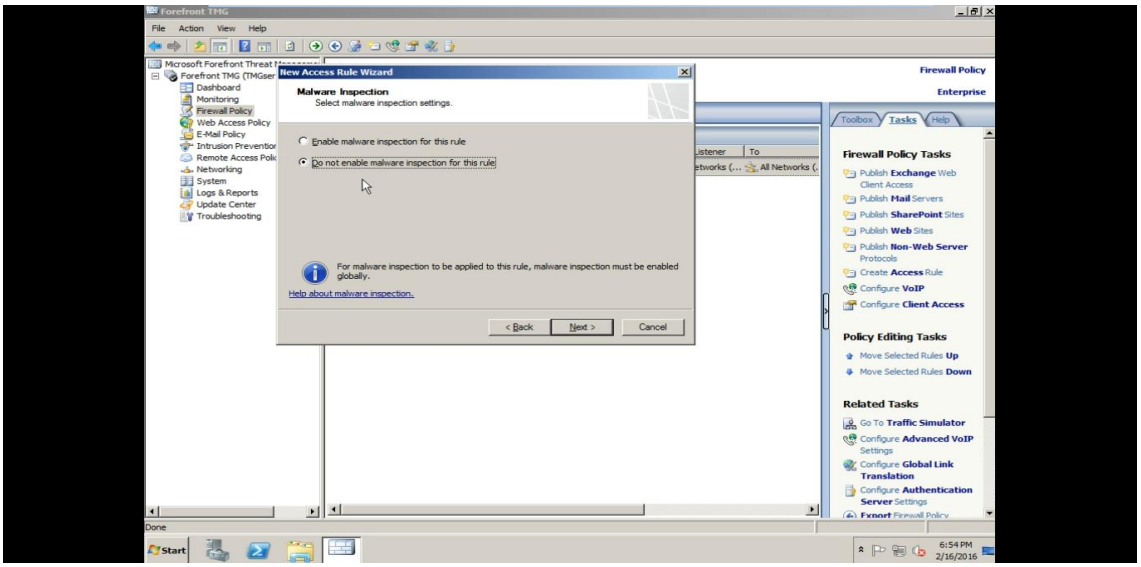

Server Manager
Tools
View
Help
Active Directory Domain Services Configuration Wizard
Domain Controller Options
TARGET SERVER
WIN-83DLG2B5S8S
Deployment Configuration
Domain Controller Options
DNS Options
Additional Options
Paths
Review Options
Prerequisites Check
Installation
Results
Select functional level of the new forest and root domain
Forest functional level:
Windows Server 2012
Domain functional level:
Windows Server 2012
Specify domain controller capabilities
Domain Name System (DNS) server
Global Catalog (GC)
Read only domain controller (RODC)
Type the Directory Services Restore Mode (DSRM) password
Password:
Confirm password:
More about domain controller options
< Previous
Next >
Install
Cancel
Hide
Services
Performance
BPA results
Services
Performance
BPA results
12:32 AM
11/1/2021

STEP 9: **WINDOW 2008 R2, WIN 7 and TMG Installation.**

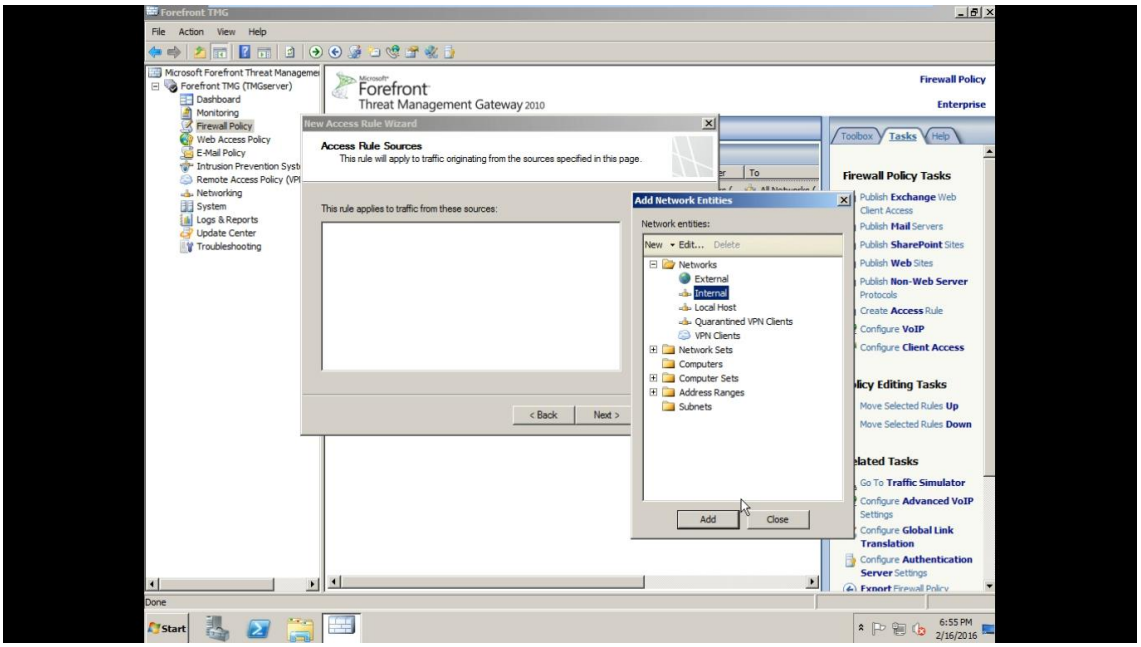
1. Install window server 2008.
2. Install win 7 and make it client of domain.
3. Add to main domain forest comsats.com.
4. Add two NIC cards from virtual machine one for internal Lan environment and one for external WAN environment.



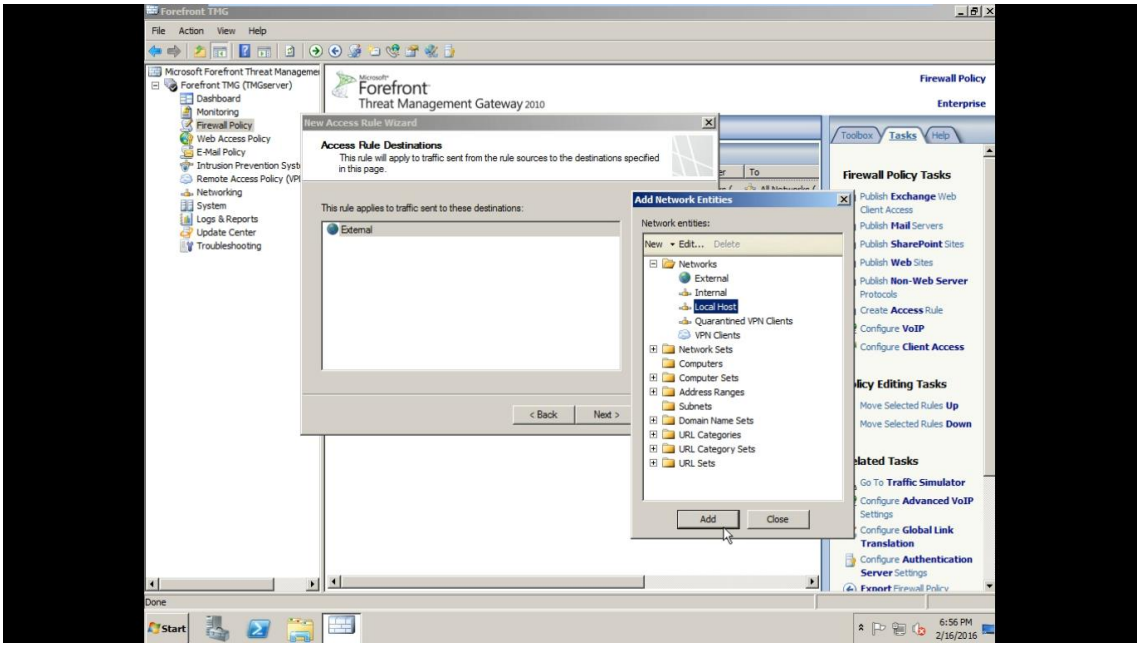
1. After TMG installation, configuring TMG.
2. First click on TMG manager and then click on do not enable malware.



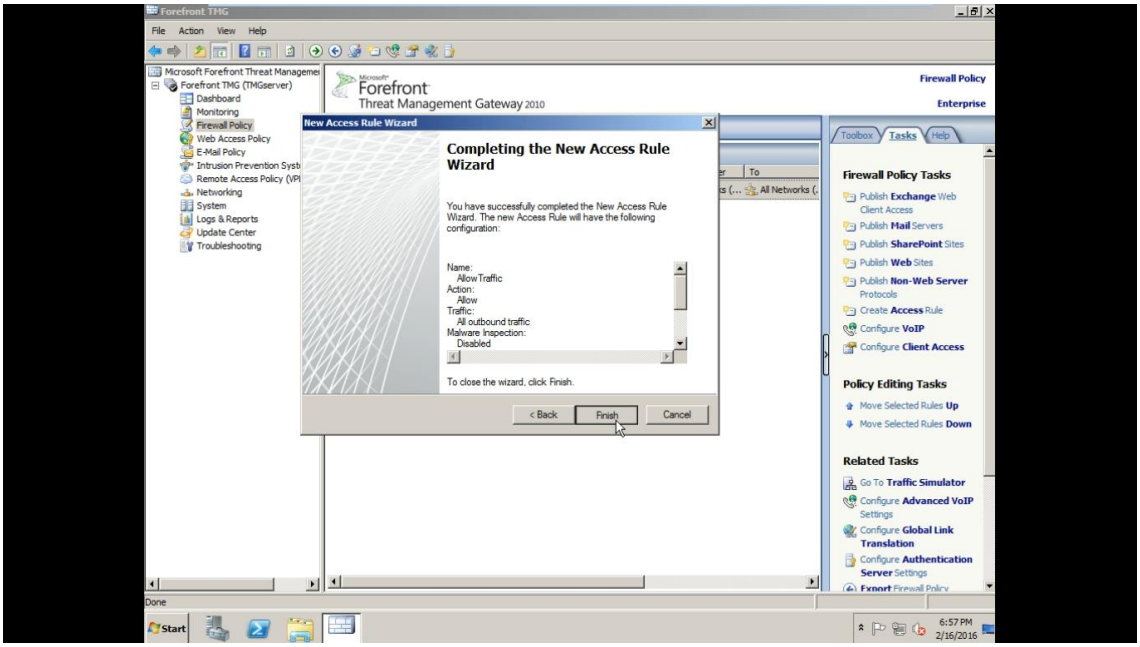
1. Next click on add source rule, the source will be internal select internal.



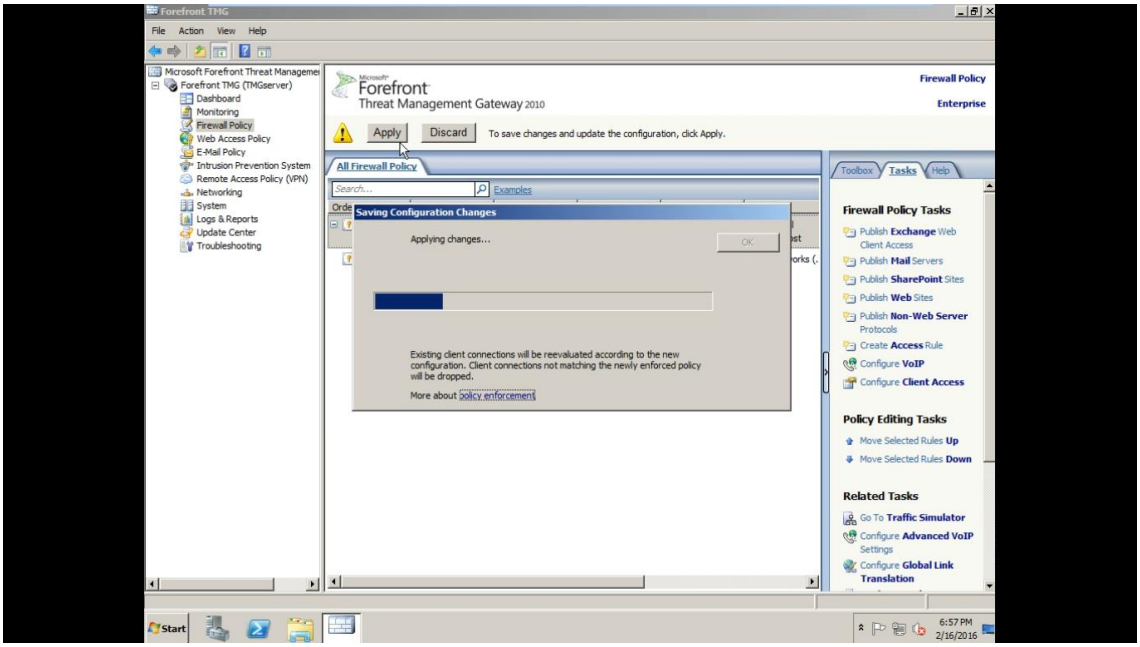
2. In destination add local host and external both.



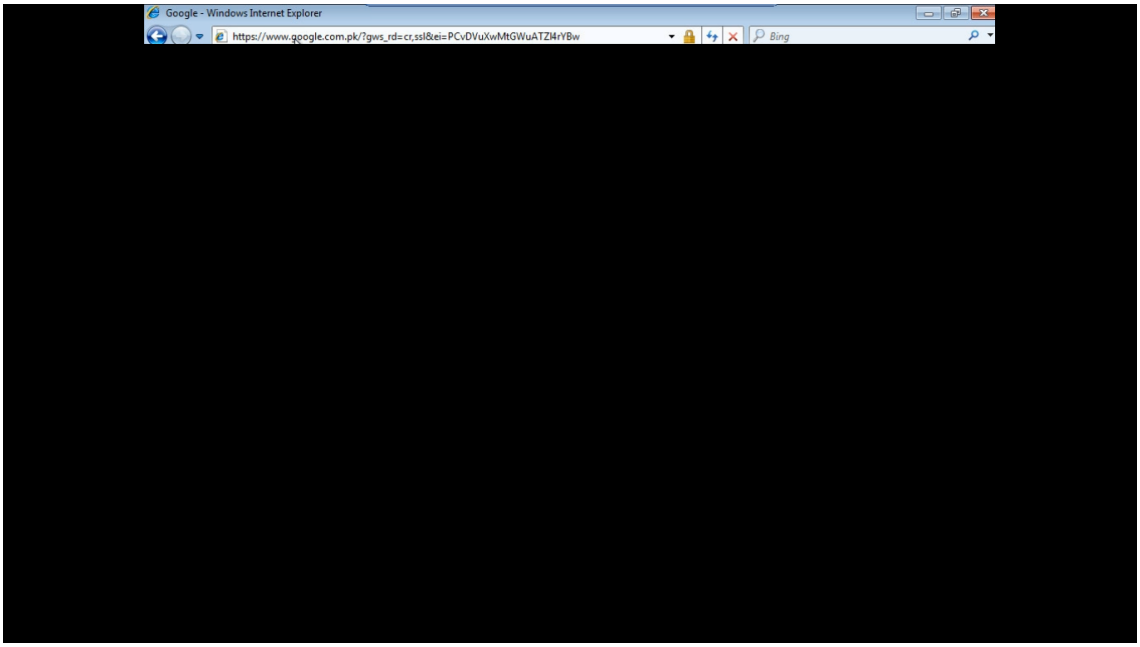
3. Then click on all users and add and finally click on finish.



4. Click on apply new rule.



5. In client window you can access internet now from external world under TMG new rule.



<u>STEP 10</u>: Add group policy management feature in window 2008.

6. Open group policy management in server 2008 and add proxy setting.

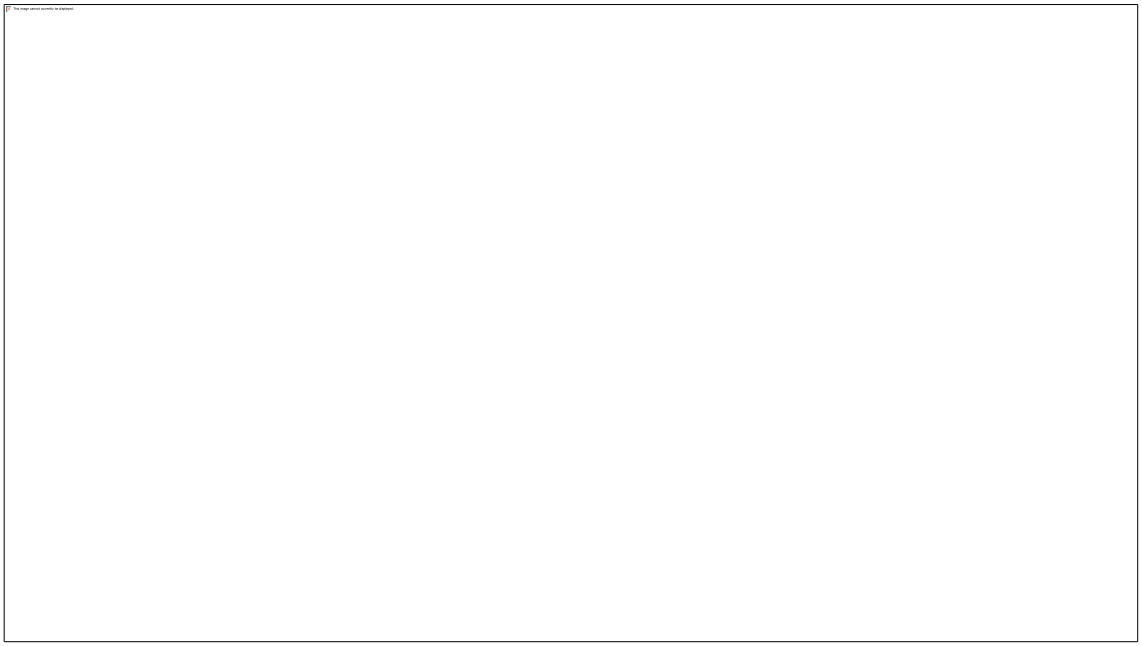
7. After this in group policy management click on user configuration then policies then administrative templates. Then click on windows components. In internet explorer click on disabled connection settings. Enable the policy.

8. Now to implement the policy open command prompt in window server 2008 and type gpudate/force to update policy for client. And also update policy in client window through same command.

9. After updating policy logged into client window and open browser in settings the LAN setting will be disabled under the applied rule.

10. To allow the internet to client add default gateway IP address which is 172.20.0.1 in proxy setting and enable the same rule. After enabling the rule update on both server and client side.

1. After enabling the rule again and saving proxy setting the TMG server will allow access to internet.



2. Stage v (verify)

Home Activities:

Try out different rules using TMG firewall on clients. In this lab we have only blocked access to internet. You can use different rules and different filters. You can apply different rules on many clients at a time.

3. Stage a2 (assess)

Submit the home activity before next lab.