LAB MANUAL

Course: CSC339-Data Communication & Computer Networks

Department of Computer Science

COMSATS UNIVERSITY ISLAMABAD

Lahore Campus

Table of Contents

Lab 1 & 2 – An Introduction to Networking Devices 3
Lab 3 & 4– Introduction of Packet Tracer 12
Lab 5 & 6 – Routing Protocols implementation on Packet Tracer 23
Lab 7 – DHCP and DNS implementation using Packet Tracer 34
Lab 8 – VLAN on Packet Tracer 48
Lab 9 – STP and Sticky MAC addresses 60
Lab 10 – CDP, Telnet and SSH , PAP and CHAP 76
Lab 11 & 12 – VOIP, Wireless, ACL on Packet Tracer 102
Lab 13 – NS2 and Ubuntu Installation, NS2 introduction 124
Lab 14 & 15 – Traffic Flow on Nodes in NS2 136
Lab 16 & 17 – Dynamic Routing in NS2 145
Lab 18 – X-Graph in NS2 148
Lab 19 – Installation of IT Guru 153
Lab 20 – Simulation of a Small Office Network on IT Guru 160
Lab 21 – A Direct Link Network with Media Access Control 176
Lab 22 – A shared Media Network with Media Access Control 186
Lab 23 – A set of Local Area Network on IT Guru 202
Lab 24 – Planning a Network with different Users, Hosts and Services 212
Lab 25 – RIP on IT Guru 223
Lab 26 – OSPF on IT Guru 234
Lab 27 & 28 – TCP on IT Guru 245
Lab 29 & 30 – Network Security and Virtual Private Networks 258

Lab 1 & 2 – An Introduction to Networking Devices

Objectives

The lab objective is to explore some basic concepts related to computer networking, networking devices and cables.

Lab Tasks

1. Introduction of Networking Devices

Outcomes

In this task, students will learn about: Basic Networking concepts & Different Networking devices.

Since, we are going to do a series of tutorials on packet tracer. In this manner, we need to have a familiarity of various networking components and devices. We are going to discuss some important devices which are going to be used in networking.

All networks are made up of basic hardware building blocks to interconnect network nodes, such as Network Interface Cards (NICs), Bridges, Hubs, Switches, and Routers etc. These devices also need cables to connect them. In this tutorial, we are going to discuss these important devices.

1. Network interface card

A NIC (network interface card) is a piece of computer hardware designed to allow computers to communicate over a computer network. It provides physical access to a networking medium and often provides a low-level addressing system through the use of MAC addresses. It allows users to connect to each other either by using cables or wirelessly. The NIC provides the transfer of data in megabytes.

Figure 1: Network Interface Card

Every device on a network that needs to transmit and receive data must have a network interface card (NIC) installed. They are sometimes called network adapters and are usually installed into one of the computer's expansion slots in the same way as a sound or graphics card. The NIC includes a transceiver, (a transmitter and receiver combined). The transceiver allows a network device to transmit and receive data via the transmission medium. Each NIC has a unique 48-bit Media Access Control (MAC) address burned into its ROM during manufacture. The first 24 bits make up a block code known as the Organizationally Unique Identifier (OUI) that is issued to manufacturers of NICs and identify the manufacturer. The issue of OUIs to organizations is administered by the Institute of Electrical and Electronics Engineers (IEEE). The last 24 bits constitute a sequential number

issued by the manufacturer. The MAC address is sometimes called a hardware address or physical address, and uniquely identifies the network adapter. It is used by many data link layer communications protocols, including Ethernet, the 802.11 wireless protocol and Bluetooth. The use of a 48-bit address allows for 2^{48}(281,474,976,710,656) unique addresses. A MAC address is usually shown in hexadecimal format, with each octet separated by a dash or colon. For example: 00-60-55-93-R2-N7

2. Repeater

A repeater is an electronic device that receives a signal and retransmits it at a higher power level, or to the other side of an obstruction, so that the signal can cover longer distances without degradation. In most twisted pair ethernet configurations, repeaters are required for cable runs longer than 100 meters away from the computer. As signals travel along a transmission medium there will be a loss of signal strength i.e. attenuation. A repeater is a non-intelligent network device that receives a signal on one of its ports, regenerates the signal, and then retransmits the signal on all of its remaining ports. Repeaters can extend the length of a network (but not the capacity) by connecting two network segments. Repeaters cannot be used to extend a network beyond the limitations of its underlying architecture, or to connect network segments that use different network access methods. They can, however, connect different media types, and may be able to link bridge segments with different data rates.

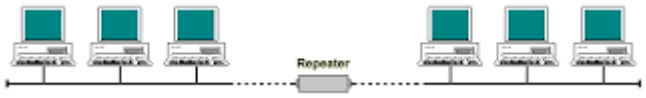


Figure 2: Repeater

Repeaters are used to boost signals in coaxial and twisted pair cable and in optical fiber lines. An electrical signal in a cable gets weaker the further it travels, due to energy dissipated in conductor resistance and dielectric losses. Similarly, a light signal traveling through an optical fiber suffers attenuation due to scattering and absorption. In long cable runs, repeaters are used to periodically regenerate and strengthen the signal.

3. Hub

A hub contains multiple ports. When a packet arrives at one port, it is copied to all the ports of the hub for transmission. In a hub, a frame is passed along or "broadcast" to every one of its ports. It doesn't matter that the frame is only destined for one port. The hub has no way of distinguishing which port a frame should be sent to. Passing it along to every port ensures that it will reach its intended destination. This place a lot of traffic on the network and can lead to poor network response times. Additionally, a 10/100Mbps hub must share its bandwidth with each and every

Figure 3: Hub

one of its ports. So when only one PC is broadcasting, it will have access to the maximum available bandwidth. If, however, multiple PCs are broadcasting, then that bandwidth will need to be divided among all of those systems, which will degrade performance.

4. Bridge

A network bridge connects multiple network segments at the data link layer (layer 2) of the OSI model. Bridges do not copy traffic to all ports, as hubs do, but learn which MAC addresses are reachable through specific ports. Once the bridge associates a port and an address, it will send traffic for that address only to that port. Bridges do send broadcasts to all ports except the one on which the broadcast was received.

Bridges learn the association of ports and addresses by examining the source address of frames that it sees on various ports. Once a frame arrives through a port, its source address is stored, and the bridge assumes that MAC address is associated with that port. The first time that a previously unknown destination address is seen, the bridge will forward the frame to all ports other than the one on which the frame arrived.

Figure 4: Bridge

Bridges don't know anything about protocols, but just forward data depending on the destination address in the data packet. This address is not the IP address, but the MAC (Media Access Control) address that is unique to each network adapter card. The bridge is basically just to connect two local-area networks (LANs), or two segments of the same LAN that use the same protocol. Bridges can extend the length of a network, but unlike repeaters they can also extend the capacity of a network, since each port on a bridge has its own MAC address. When bridges are powered on in an Ethernet network, they start to learn the network's topology by analyzing the source addresses of incoming frames from all attached network segments (a process called backward learning). Over a period of time, they build up a routing table.

The bridge monitors all traffic on the segments it connects and checks the source and destination address of each frame against its routing table. When the bridge first becomes operational, the routing table is blank, but as data is transmitted back and forth, the bridge adds the source MAC address of any incoming frame to the routing table and associates the address with the port on which the frame arrives. In this way, the bridge quickly builds up a complete picture of the network topology. If the bridge does not know the destination segment for an incoming frame, it will forward the frame to all attached segments except the segment on which the frame was transmitted. Bridges reduce the amount of traffic on individual segments by acting as a filter, isolating intra-segment traffic. This can greatly improve response times.

5. Switch

The switch is a relatively new network device that has replaced both hubs and bridges in LANs. A switch uses an internal address table to route incoming data frames via the port associated with

their destination MAC address. Switches can be used to connect together a number of end-user devices such as workstations, or to interconnect multiple network segments. A switch that interconnects end-user devices is often called a workgroup switch. Switches provide dedicated full-duplex links for every possible pairing of ports, effectively giving each attached device its own network segment, This significantly reduces the number of intra-segment and inter-segment collisions. Strictly speaking, a switch is not capable of routing traffic based on IP address (layer 3) which is necessary for communicating between network segments or within a large or complex LAN. Some switches are capable of routing based on IP addresses but are still called switches as a marketing term. A switch normally has numerous ports, with the intention being that most or all of the network is connected directly to the switch, or another switch that is in turn connected to a switch.

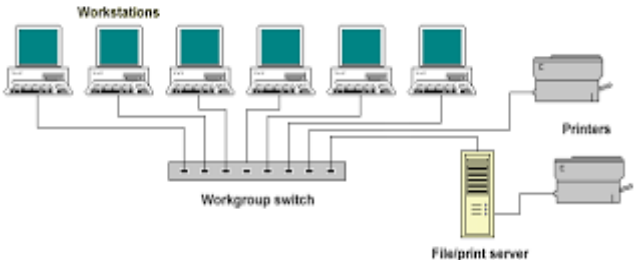


Figure 5: Switch

6. Router

Routers are networking devices that forward data packets between networks using headers and forwarding tables to determine the best path to forward the packets. A network environment that consists of several interconnected networks employing different network protocols and architectures requires a sophisticated device to manage the flow of traffic between these diverse networks. Such a device, sometimes referred to as an intermediate system, but more commonly called a router, must be able to determine how to get incoming packets (or datagrams) to the destination network by the most efficient route. Routers gather information about the networks to which they are connected and can share this information with routers on other networks. The information gathered is stored in the router's internal routing table and includes both the routing information itself and the current status of various network links. Routers exchange this routing information using special routing protocols.

A router is connected to at least two networks, commonly two LANs or WANs or a LAN and its ISP's network. Routers are located at gateways, the places where two or more networks connect, and are the critical device that keeps data flowing between networks and keeps the networks connected to the Internet. When data is sent between locations on one network or from one network to a second network the data is always seen and directed to the correct location by the router. The router accomplishes this by using headers and forwarding tables to determine the best path for forwarding the data packets, and they also use protocols such as ICMP to communicate with each other and configure the best route between any two hosts. The Internet itself is a global network connecting millions of computers and smaller networks. There are various routing protocols which are helpful for various different environments and will be discussed later.

Figure 6: Router

Lab Tasks

2. Introduction to different cables and connections, which type of cables are used for networking?

Outcomes

In this task, students will learn about:

- Basic concepts of cables
- Different types of cables
- Categories of cables
- Manufacturing of cables

In order for the communication to take place, cables play important role. Cable is the medium through which information usually moves from one network device to another. There are several types of cable which are commonly used with LANs. The type of cable chosen for a network is related to the network's topology, protocol, and size.

There are various types of cables used in networks as follows.

- Unshielded Twisted Pair (UTP) Cable
- Shielded Twisted Pair (STP) Cable
- Coaxial Cable
- Fiber Optic Cable

Twisted Pair Cables:

Twisted pair cabling is a type of wiring in which two conductors of a single circuit are twisted together for the purposes of canceling out electromagnetic interference (EMI) from external sources; for instance, electromagnetic radiation from unshielded twisted pair (UTP) cables, and crosstalk between neighboring pairs. In balanced pair operation, the two wires carry equal and opposite signals and the destination detects the difference between the two. This is known as differential mode transmission. Noise sources introduce signals into the wires by coupling of electric or magnetic fields and tend to couple to both wires equally. The noise thus produces a common-mode signal which is cancelled at the receiver when the difference signal is taken.

Categories of UTP Cable:

It has been categorized into three categories based on the equipment that are being connected through these wires.

1) Straight Through Cable
2) Cross Over Cable
3) Roll Over Cable

Explanation:

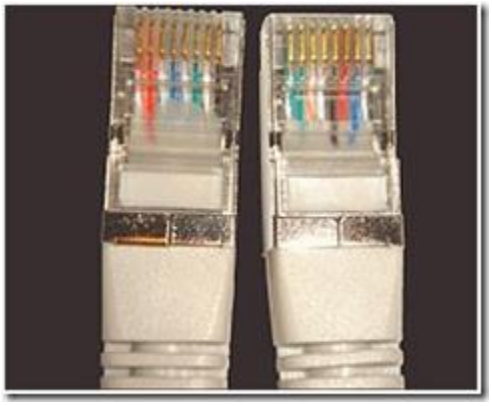
Straight Through Cable:
Straight through cables are used to connect different devices like Switch to PC. Switch to Router. Router to Switch etc. Straight-through cables are used when each end of the communication transmits and receives on different pairs.

Cross Over Cable:
In a cross over the cable, the send and receive wires are "crossed over", meaning the wires are opposite on each end. This allows two PCs to talk to each other, has it connected the send of one computer to the receiver of the other. Hence, the cross over cables are used to connect similar devices like PC to PC, Router to Router, Switch to Switch, Hub to Hub etc.

Roll Over Cable:
Roll over cables are used to connect to the console port of the device. It gets the name rollover because the pin outs on one end are reversed from the other, as if the wire had been rolled over and you were viewing it from the other side.

Transmission Pins:
Devices that transmit on 1,2 and receive on 3,6:
1) PC
2) Router
3) Wireless Access Point AP
4) Networked printers

Devices that transmit on 3,6 and receive on 1,2:
1) Switch
2) Bridge
3) Hub



Required Equipment:
In order to make a network cable you need the following equipment.

A. Cat5, Cat5e cable:
CAT5 cable usually contains four pairs of copper wire, Fast Ethernet communications only utilize two pairs. A newer specification for CAT5 cable -CAT5 enhanced ("CAT5e" or "CAT 5e")- supports networking at Gigabit Ethernet [speeds (up to 1000 Mbps) over short distances by utilizing all four wire pairs, and it is backward-compatible with ordinary CAT5.

B. A connector named RJ-45:
RJ45 connectors feature eight pins to which the wire strands of a cable interface electrically. Standard RJ-45 pin outs define the arrangement of the individual wires needed when attaching connectors to a cable.

C. **Crimping tool:**
Use to crimp the cable inside RJ 45 connector.

D. Wire stripper or Knife:
You can use a knife too to cut the wire open. In order to make different combinations of it. we will have to cut the upper protective coating and bring out the eight wires.

Category	Speed	Use
1	1 Mbps	Voice Only (Telephone Wire)
2	4 Mbps	Local Talk & Telephone (Rarely used)
3	16 Mbps	10BaseT Ethernet
4	20 Mbps	Token Ring (Rarely used)
5	100 Mbps (2 pair)	100BaseT Ethernet
5e	1,000 Mbps	Gigabit Ethernet
6	10,000 Mbps	Gigabit Ethernet

Table 1: Cable Categories

Auto-MDIX:
Auto-MDIX (automatic medium-dependent interface crossover) is a computer networking technology that automatically detects the required cable connection type (straight-through or crossover) and configures the connection appropriately, thereby removing the need for crossover cables to interconnect switches or connecting PCs peer-to-peer. When it is enabled, either type of cable can be used, or the interface automatically corrects any incorrect cabling. For Auto-MDIX to operate correctly, the speed on the interface and duplex setting must be set to **"auto".**

Color Schemes of different pairs:
There are 2 different standards that are followed i.e. T568 A and t568 B. **TIA/EIA-568** is a set of telecommunications standards from the Telecommunications Industry Association (TIA), an

offshoot of the Electronic Industries Alliance (EIA). The standards address commercial building cabling for telecommunications products and services. Perhaps the widest known and most discussed feature of TIA/EIA-568 is the definition of pin/pair assignments for eight-conductor 100-ohm balanced twisted-pair cabling, such as Category 3, Category 5 and Category 6 unshielded twisted-pair (UTP) cables. These assignments are named T568A and T568B and they define the pinout, or order of connections, for wires in 8P8C (often incorrectly referred to as RJ45) eight-pin modular connector plugs and sockets.

So, here is the diagram to show the color scheme of A standard.

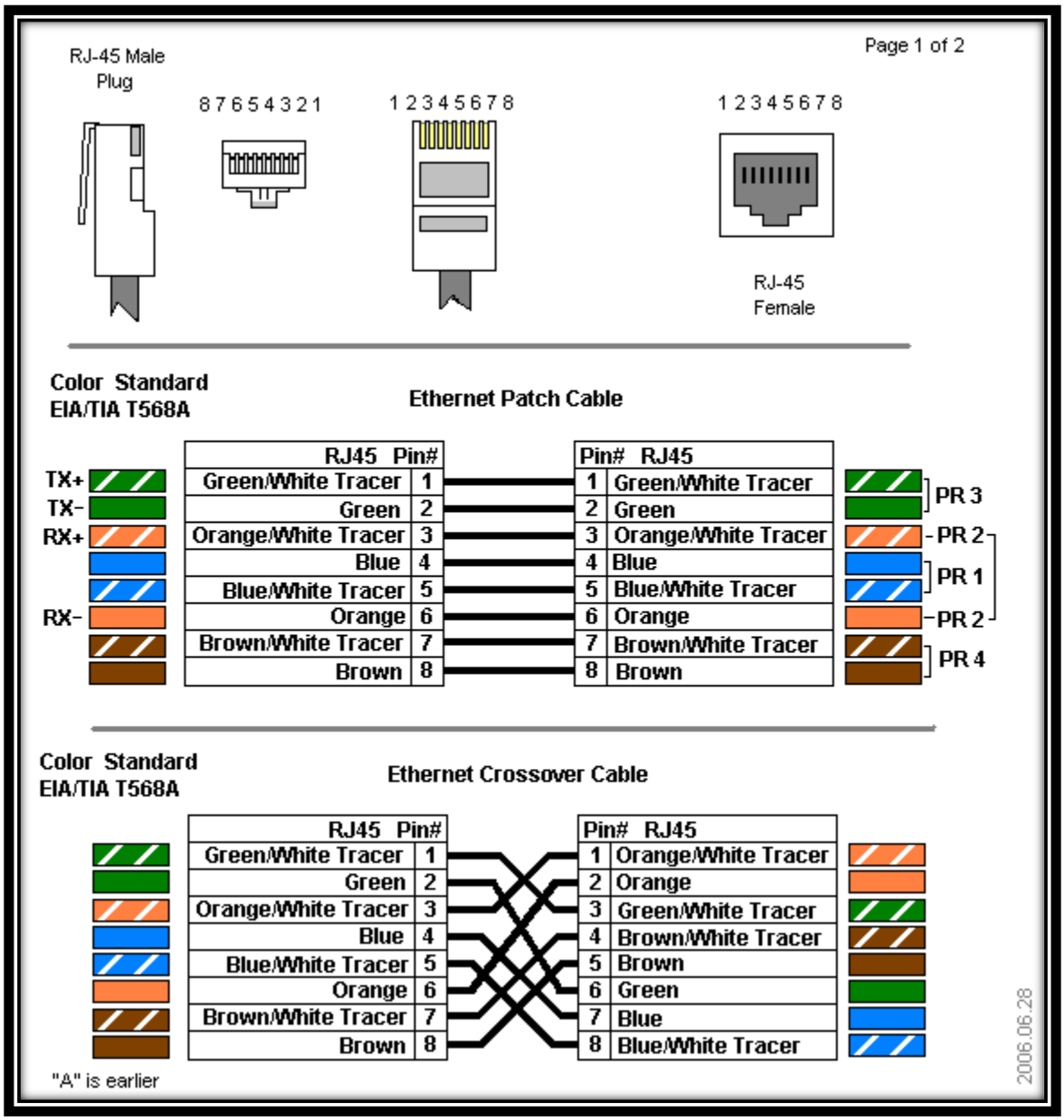


Figure 7: Standard A Color Scheme

Here is the diagram for B standard.

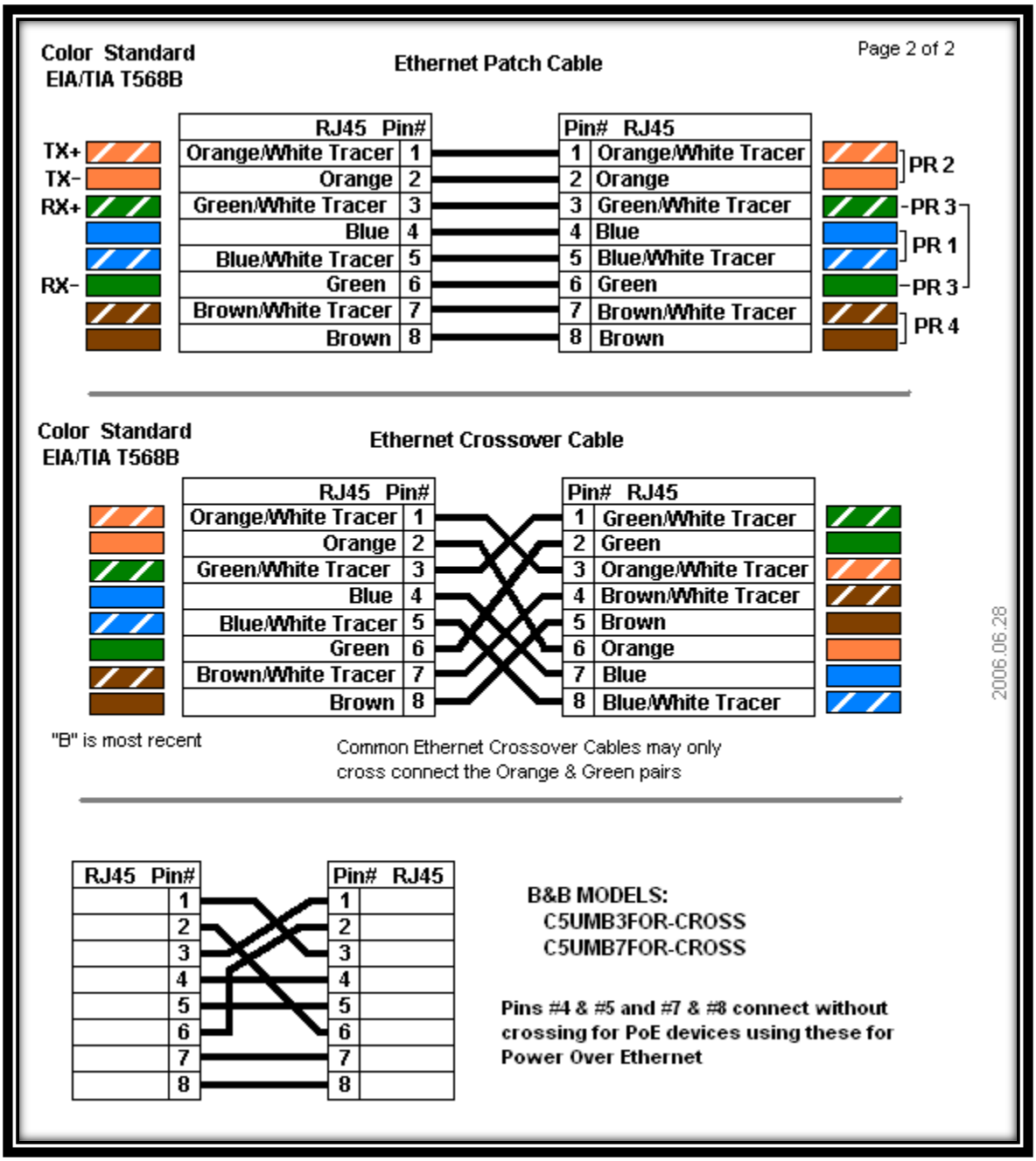


Figure 8: Standard B Color Scheme

For rollover cable, we just invert the wires of straight through combination on one end.

Lab 3 & 4– Introduction of Packet Tracer

Objectives

In this lab, students will learn:

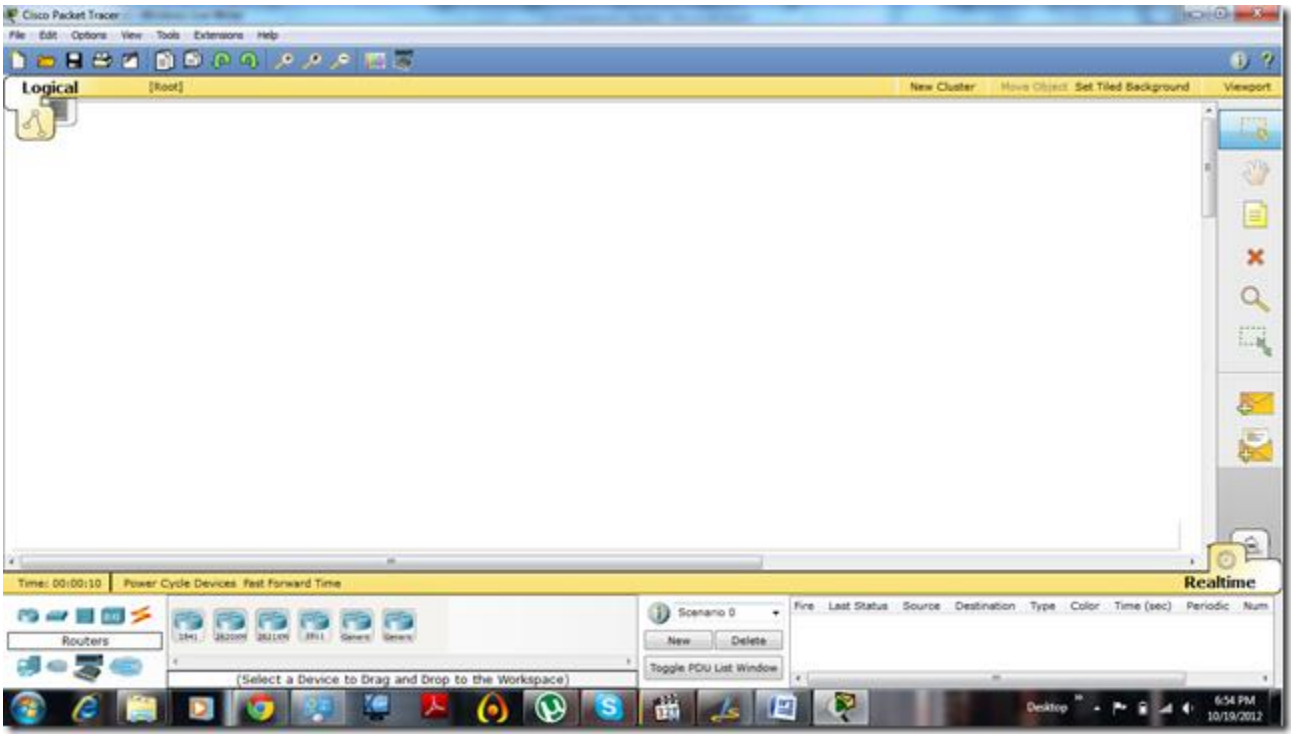
- How to use packet tracer.
- How to use different commands on packet tracer CLI mode.
- How to configure different devices on packet tracer.

Lab Tasks

1. **Explore Packet Tracer and enable communication on packet tracer by using different devices.**

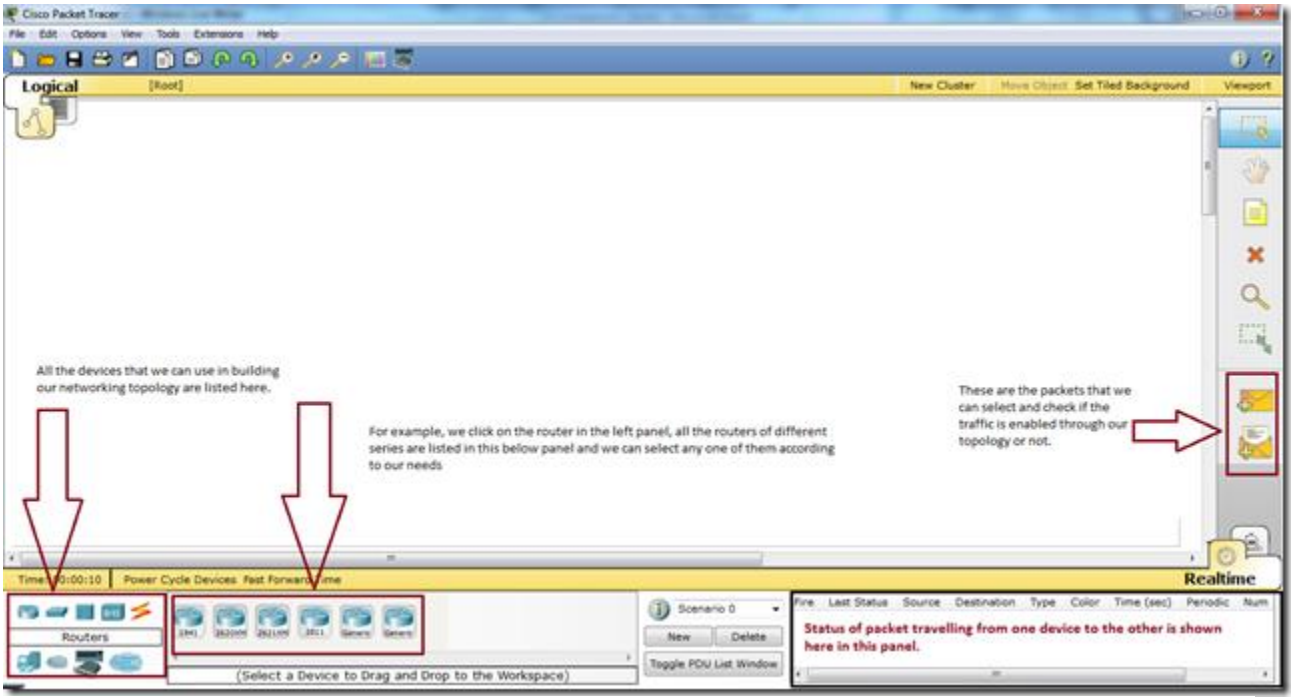
Packet Tracer is a powerful network simulator that can be utilized in training for CCNA and CCNP certification exam by allowing students to create networks with an almost unlimited number of devices and to experience troubleshooting without having to buy real Cisco routers or switches. The tool is created by Cisco Systems. The purpose of Packet Tracer is to offer students a tool to learn the principles of networking as well as develop Cisco technology specific skills. However, it is not be used as a replacement for Routers or Switches.

Here how it looks like after we start it.

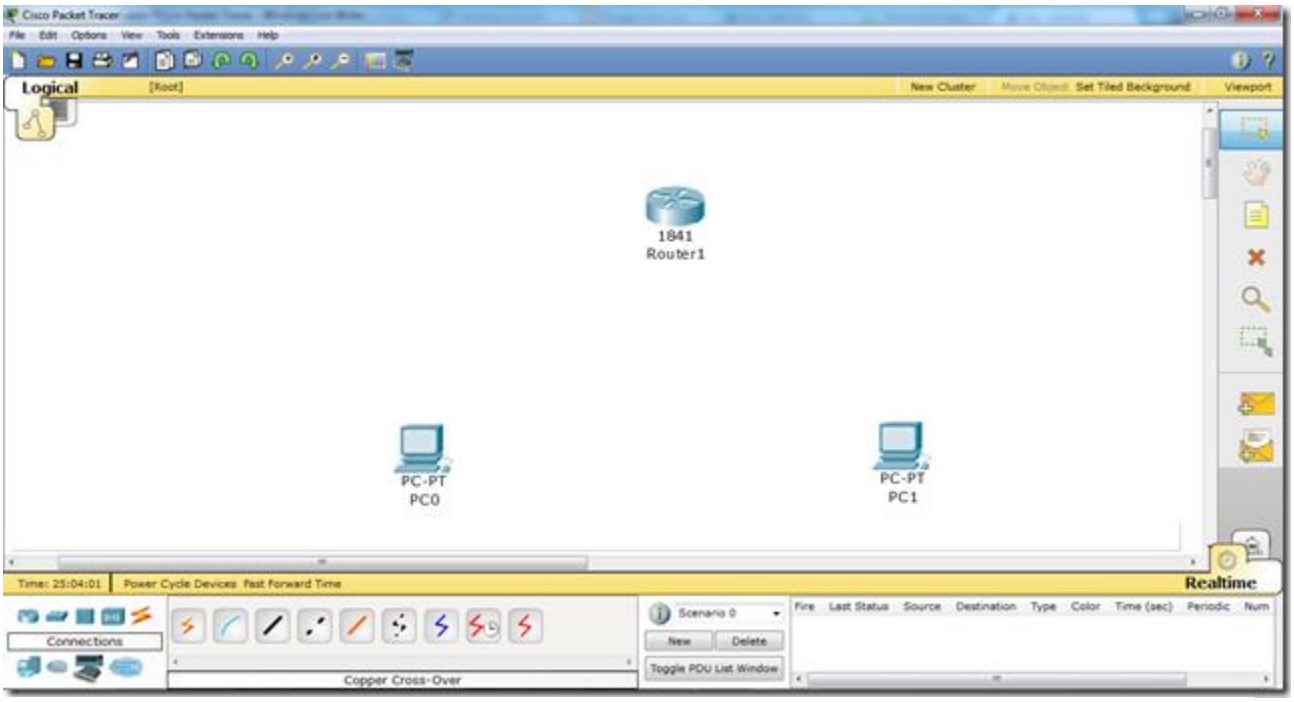


We are different modules and panels available in the packet tracer. Some important modules, which are

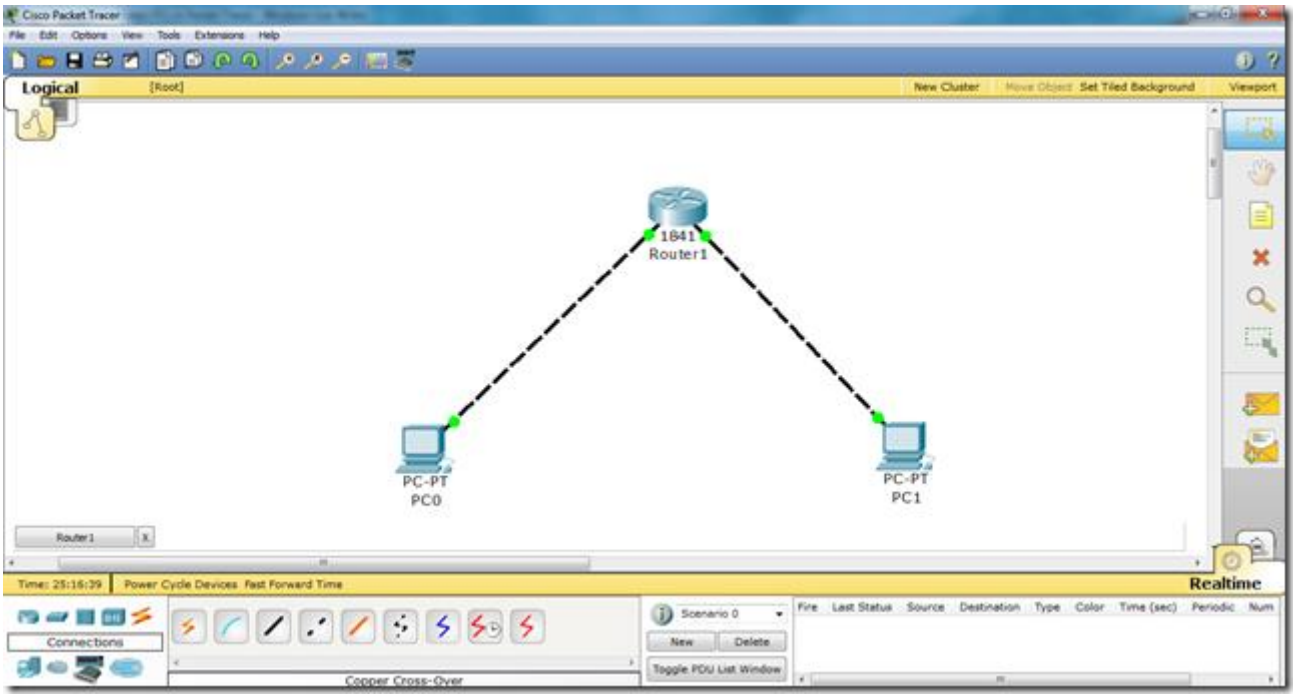
important to understand for the working in Packet Tracer, are mentioned in the following diagram.



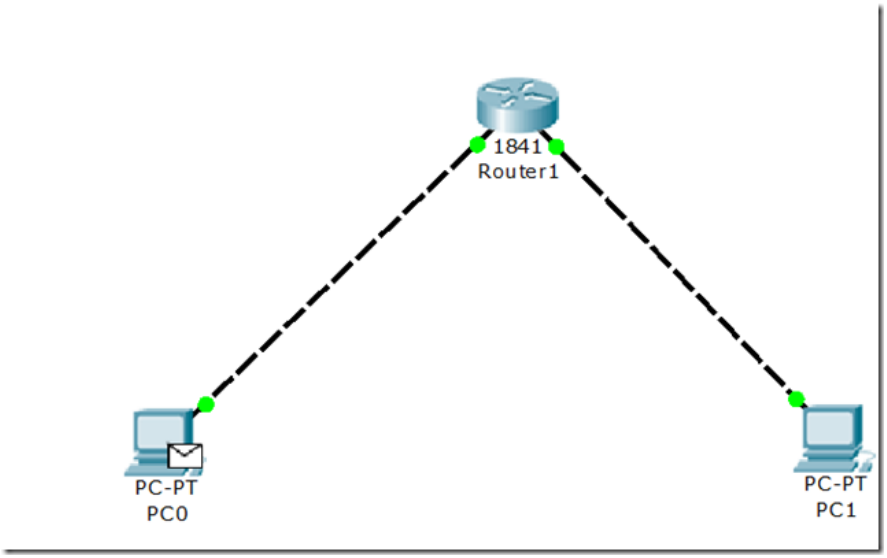
Now, in order to create a topology, we will have to select some of the devices and put them in our main window i.e. the white portion of packet tracer. and here how it looks after we add the devices.



Now, we will have to connect these devices and for that we use cables. To understand the cables, please refer to my following article.



And after you successfully create the topology, you can check either the traffic is flowing or not by selecting the packet from right panel and putting it on both PCs as follows.



CLI Commands:

There are various common commands that one needs to be familiar with. These commands will be used all the time.

There are different modes. All modes have their own distinct commands.

Router>**enable** = Router>**enab** = Router>**en**	Entering a shortened form of a command is sufficient as long as there is no confusion about which command you are attempting to enter.
Router#**configure terminal** is the same as Router#**config t**	

All the configuration commands will be written in configuration mode.

Using the tab Key to Complete Commands

When you are entering a command, you can use the tab key to complete the command. Enter the first few characters of a command and press the tab key. If the characters are unique to the command, the rest of the command is entered in for you. This is helpful if you are unsure about the spelling of a command. For example, if we write in enable mode, “sh” and press tab button, “show” command will be written on the CLI mode.

Router#**sh** Tab = Router#**show**	

Router>	User mode
Router#	Privileged mode (also known as EXEC-level mode)
Router(config)#	Global configuration mode
Router(config-if)#	Interface mode

Router(config-subif)#	Subinterface mode
Router(config-line)#	Line mode
Router(config-router)#	Router configuration mode

TIP: There are other modes than these. Not all commands work in all modes. Be

careful. If you type in a command that you know is correct—**show running-config**, for example—and you get an error, make sure that you are in the correct mode.

Entering Global Configuration Mode

`Router>`	Limited viewing of configuration. You cannot make changes in this mode.
`Router#`	You can see the configuration and move to make changes.
`Router#configure terminal` `Router(config)#`	Moves to global configuration mode. This prompt indicates that you can start making changes.

Configuring a Router Name

This command works on both routers and switches.

`Router(config)#hostname Cisco`	The name can be any word you choose.
`Cisco(config)#`	

Configuring Passwords

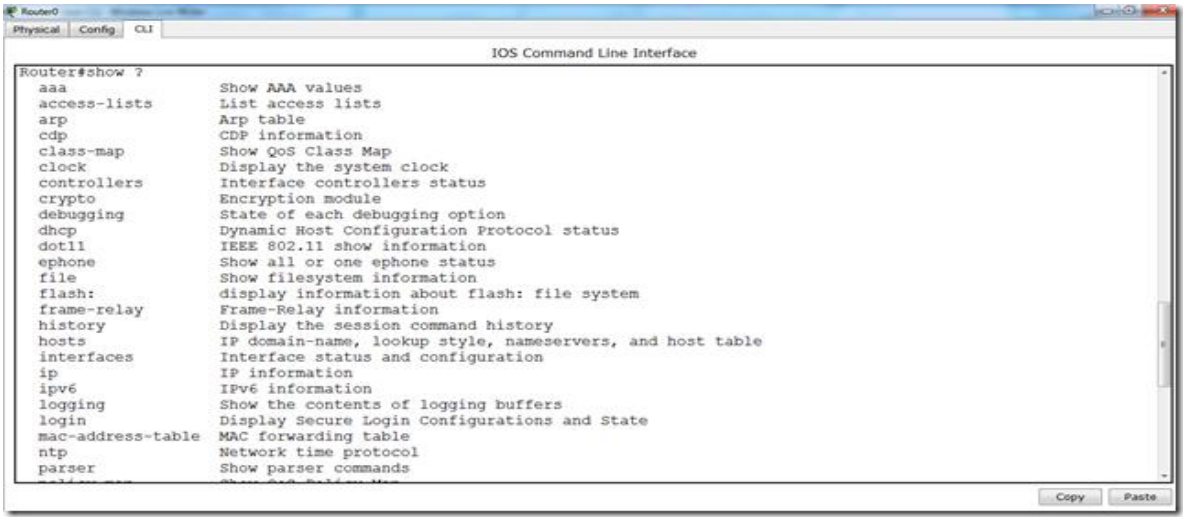
These commands work on both routers and switches.

`Router(config)#enable password cisco`	Sets **enable** password
`Router(config)#enable secret class`	Sets **enable secret** password

Here it is important to know that the enable secret password is encrypted by default. The enable password is not. For this reason, recommended practice is that you *never* use the enable password command. Use only the enable secret password command in a router or switch configuration. You cannot set both enable secret*password* and enable *password* to the same password. By doing so, it defeats the use of encryption.

Show Commands:

There are various show commands. in order to get familiar with these commands just write “show ?” in the enable mode.

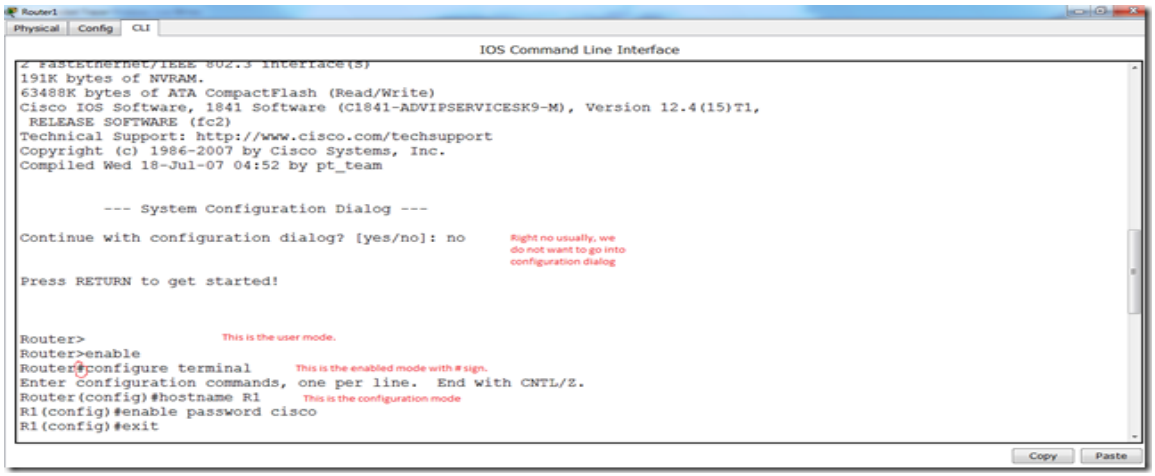


Similarly, there are various commands that shows us the configurations that we have done on our router or any other device depending on the device we are working on. We will talk about later on.

Similarly, there are various commands in the configuration mode which we attain by entering the following command

Router# configure terminal.

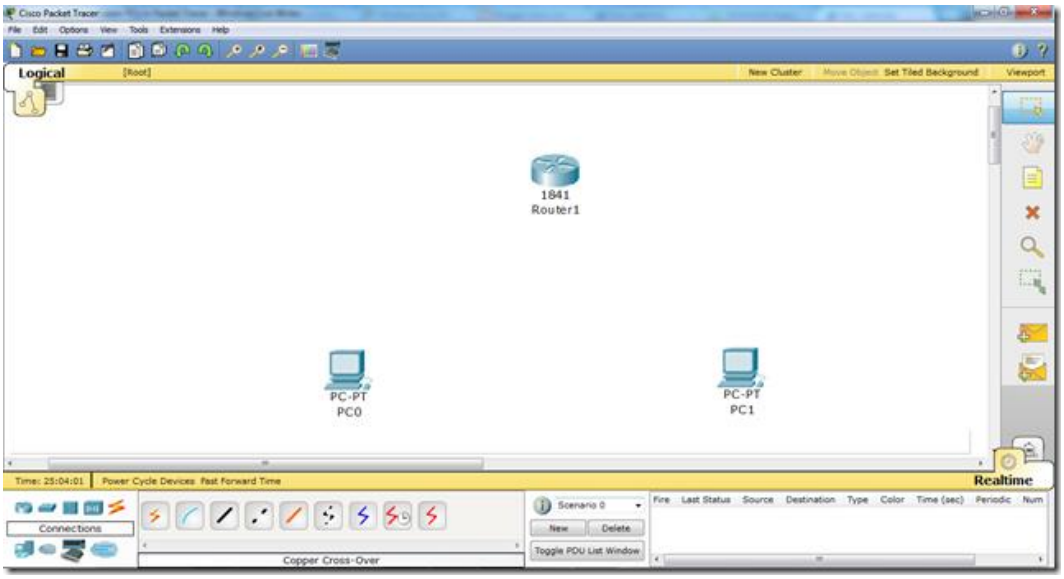
Look at the following diagram.



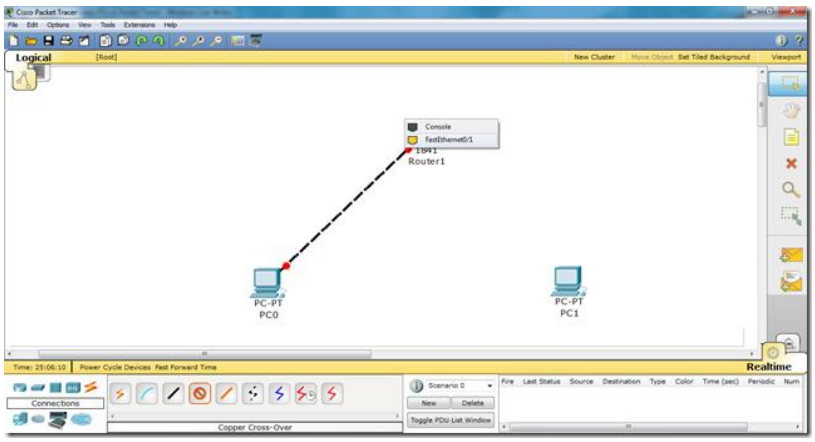
Ok, so now lot of things are happening here. Red markers explain them, also we are setting the hostname of the router and we have successfully applied the password by the following command.
enable password cisco
Here, “cisco” is the password.
Similarly, the command **“router#show ip interface brief”** gives us the information about the interfaces of the router. Now, detail discussion on the interfaces will be done later.

2. Connect different devices and establish the communication

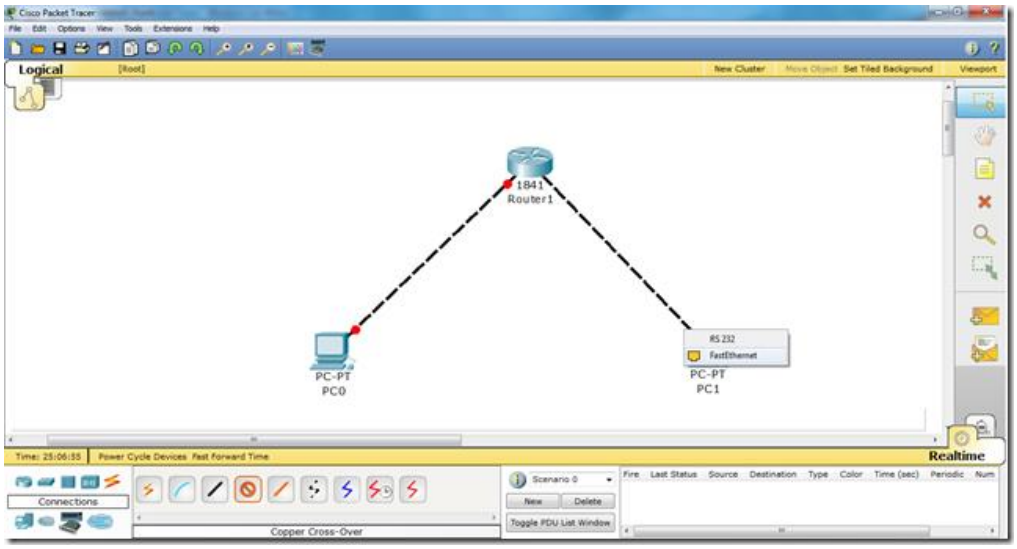
Here, we will see communication enabled between PCs via Router in Packet Tracer. So, for this we need two PCs, a router, and two cross over cables to connect them. Important point is that we use cross over cable to connect PC to a router because they both use the same pins for transmission and receiving of data.



Now, we will connect them by selecting fast ethenet interfaces on both ends.



Similarly, on the PC side we will select fast Ethernet interface.



Now, we have connected the devices. Further, we will go to the router CLI mode and enter the following commands.

Step by step, we will have to do the following things.

i. Access the interfaces one by one

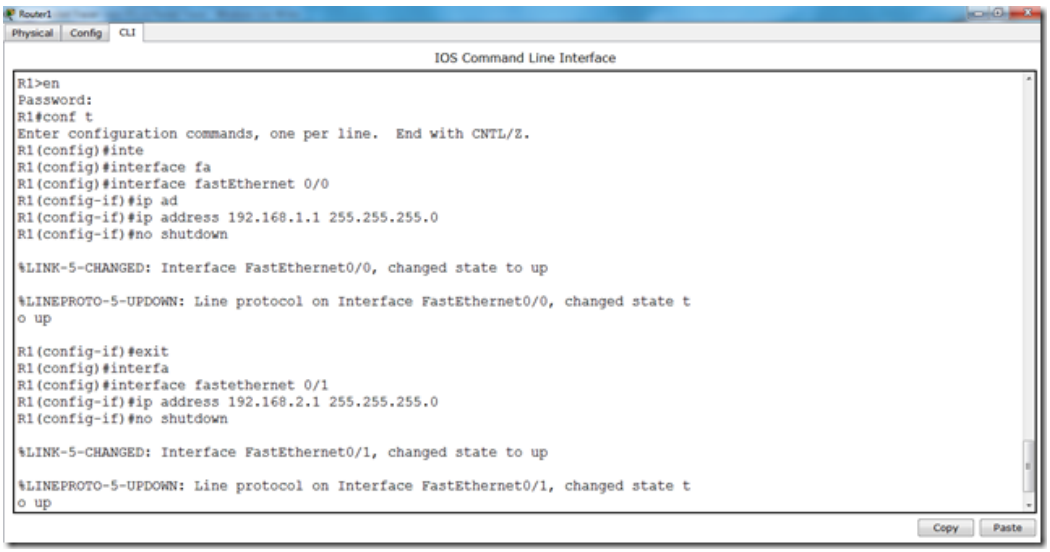
ii. Assign IP addresses to interfaces

iii. Change the status of the interfaces i.e. from Down to Up.

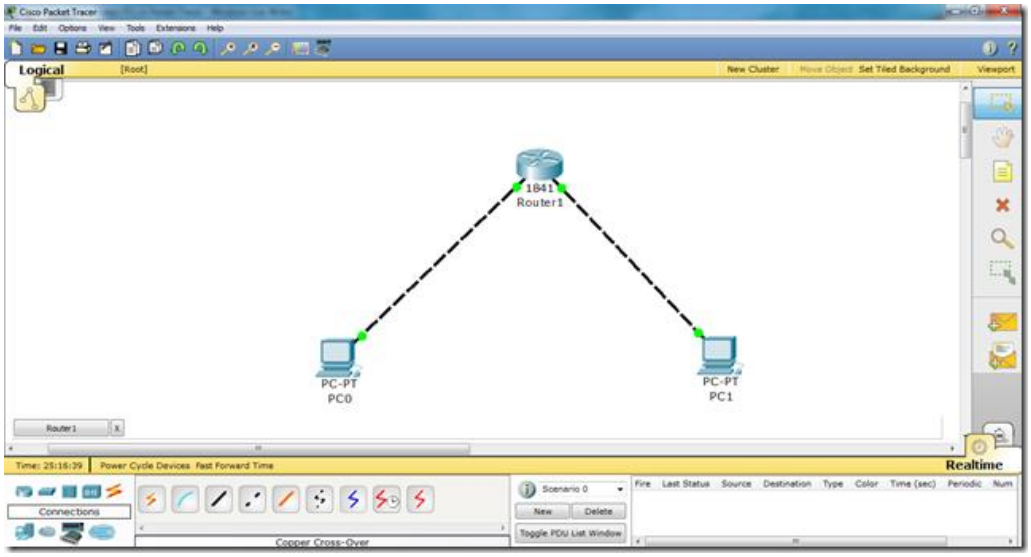
iv. Assign IP addresses to PCs.

v. Assign Default GateWay to PCs. FYI fast ethernet ip address is the gateway address to the PC.

Now, commands of the Router CLI mode are as follows.

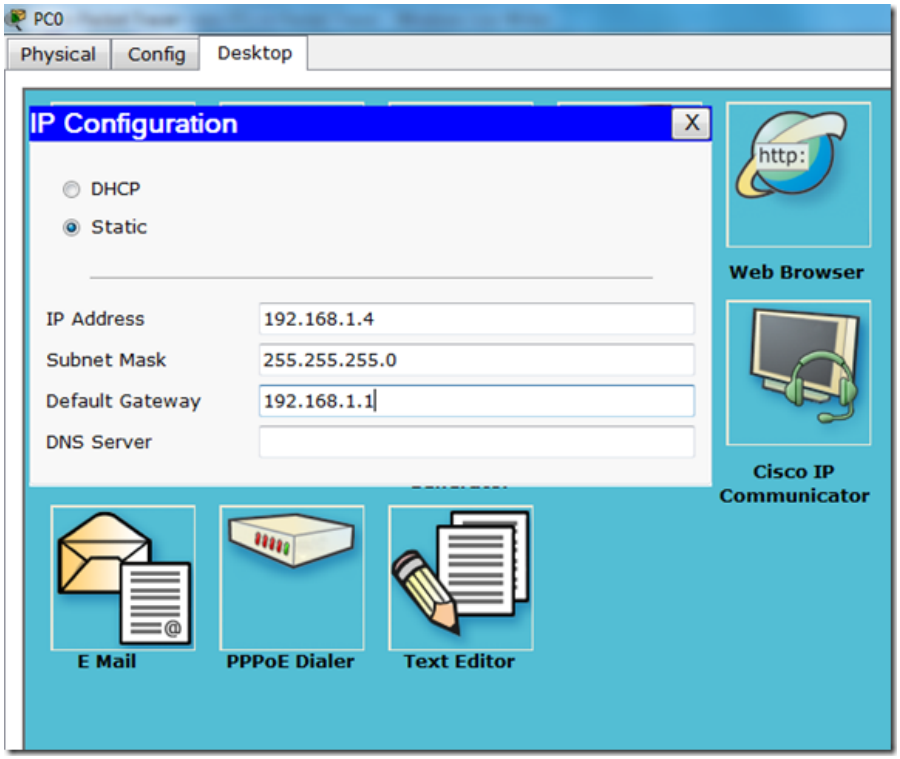


Now, we have accessed both interfaces one by one and we have assigned IP addresses respectively.

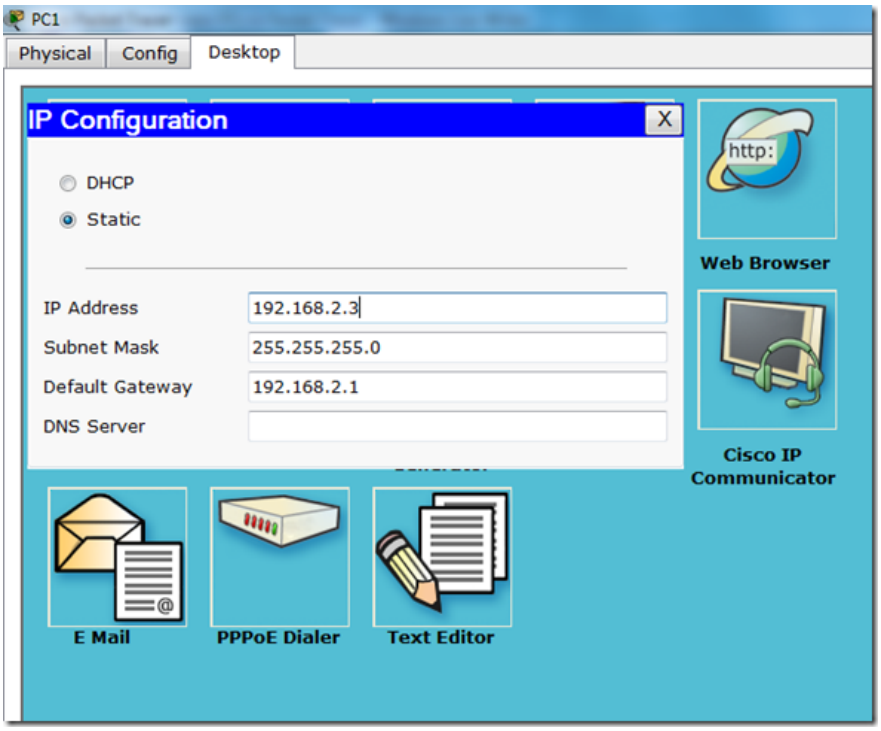


See the difference the lights have changed the color from Red to Green :)
Now, lets assign IP addresses to the PCs.
Click on PC1, go to Desktop, then click IP Configuration.

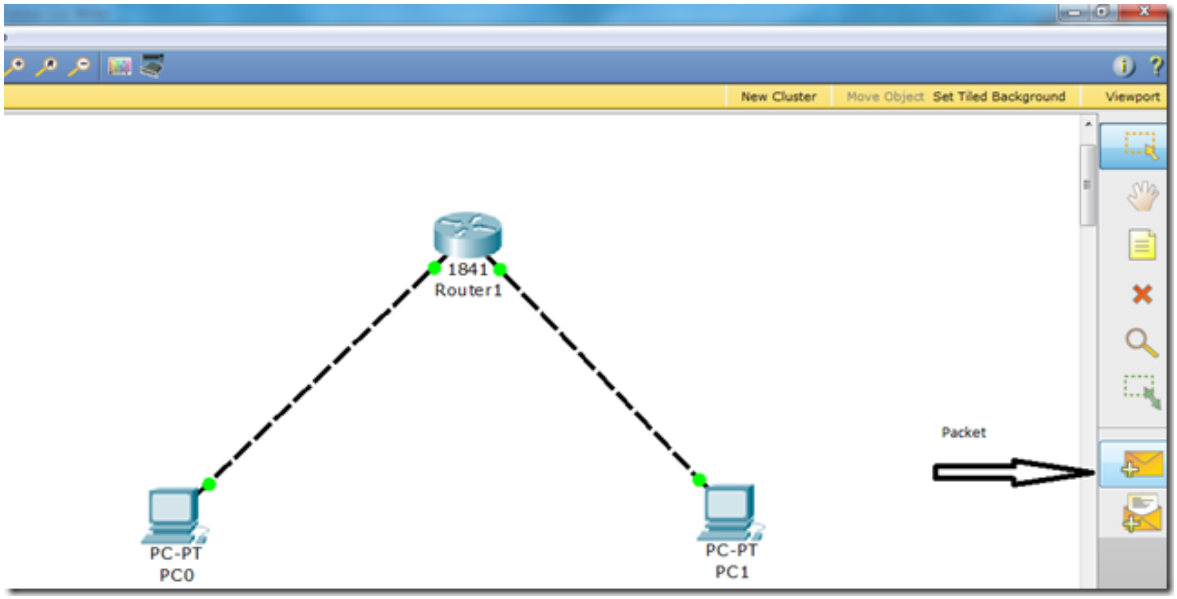
PC1



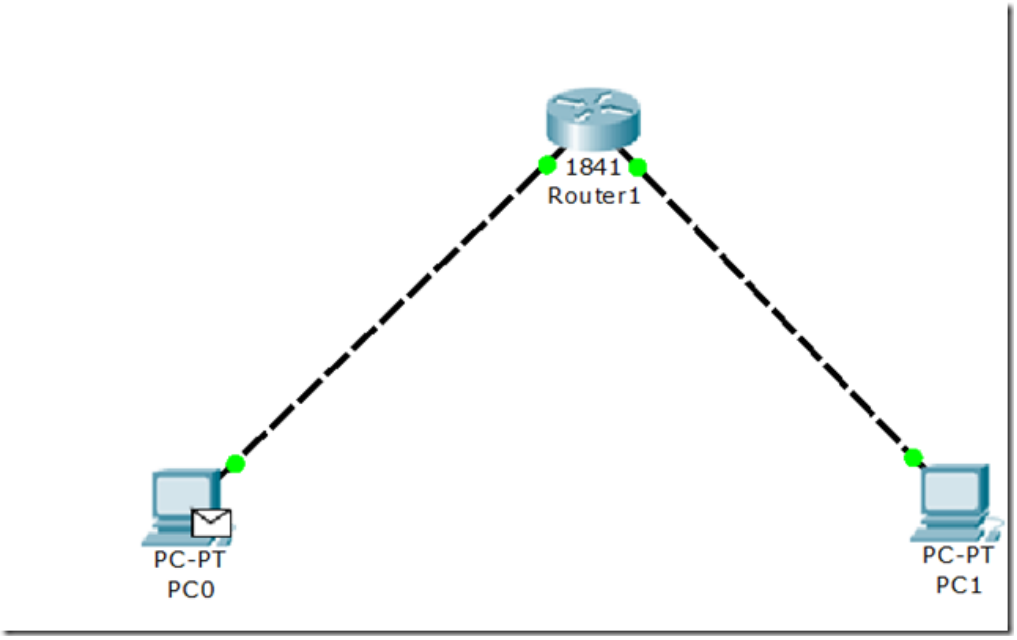
PC2



Now, our communication is enabled, and we are able to communicate from PC1 to PC2 via Router. Click on the packet in the right panel on the packet tracer, then click on PC1 and then click on PC2. You will see the successful packet tracer (status is shown in the bottom right corner)



Select it and click on both PCs.



Bingo, your communication is successful.

Lab 5 & 6 – Routing Protocols implementation on Packet Tracer

Objectives

In this lab, students will learn how to routing protocols using packet tracer.

TASKS

1. Application of RIP (routing information protocol) version 1 and version 2

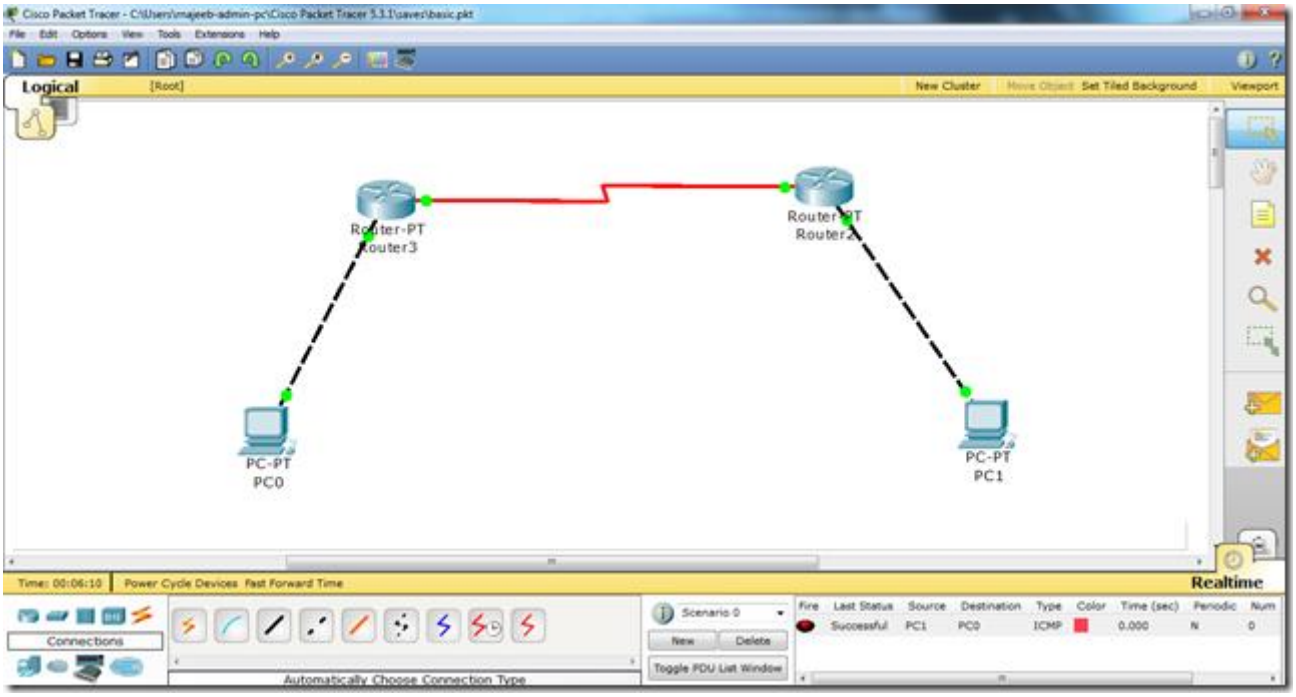
Main Commands

`Router(config)#router rip`	Enables RIP as a routing protocol.
`Router(config-router)#network w.x.y.z`	w.x.y.z is the network number of the *directly connected* network you want to advertise.

You need to advertise only the classful network number, not a subnet.

Explanation:

Lets apply RIP protocol on the following topology.



Now, we will follow the steps as mentioned in detail in the following article. i.e.
i. We will assign IP addresses to all the fast Ethernet and serial interfaces respectively.
ii We will change the state of the interfaces from down to UP.
Then, after we are done with the basic step. We will apply RIP protocol commands on both routers.
As follows.

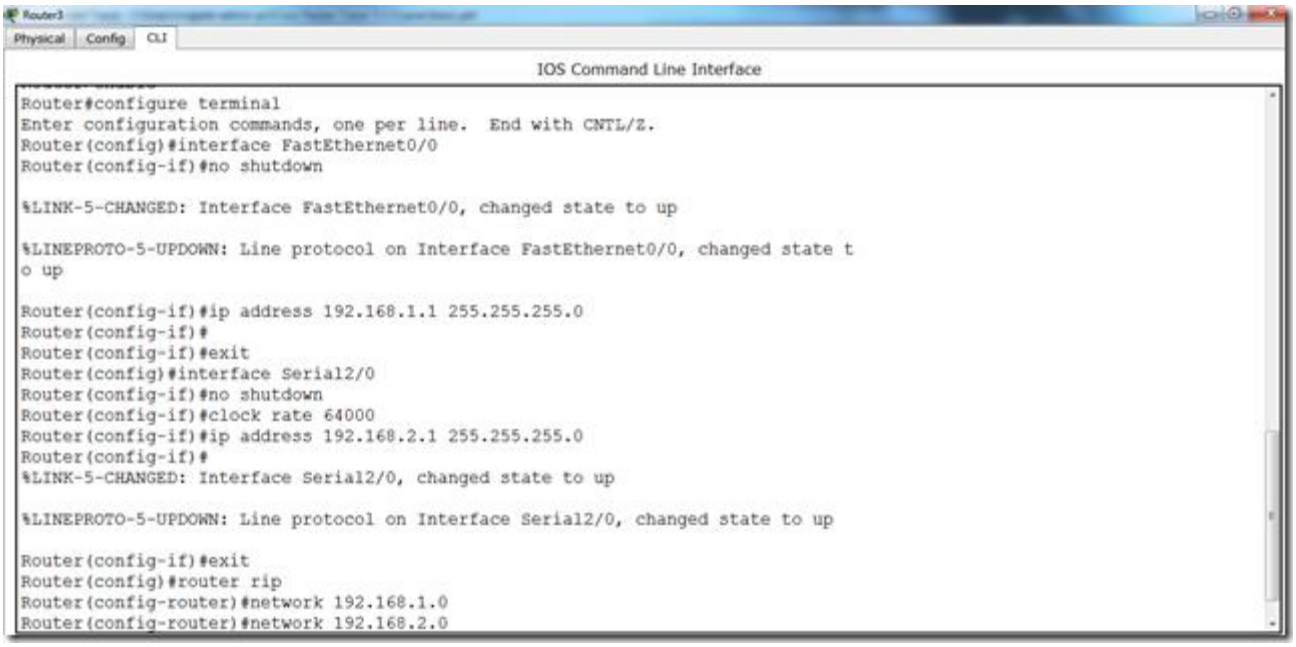
R1
In order to apply protocol RIP, we will write the following set of commands.

Router(config)# router rip

Router(config-router)# network 192.168.1.0

Router(config-router)# network 192.168.2.0

Router(config-router)#exit



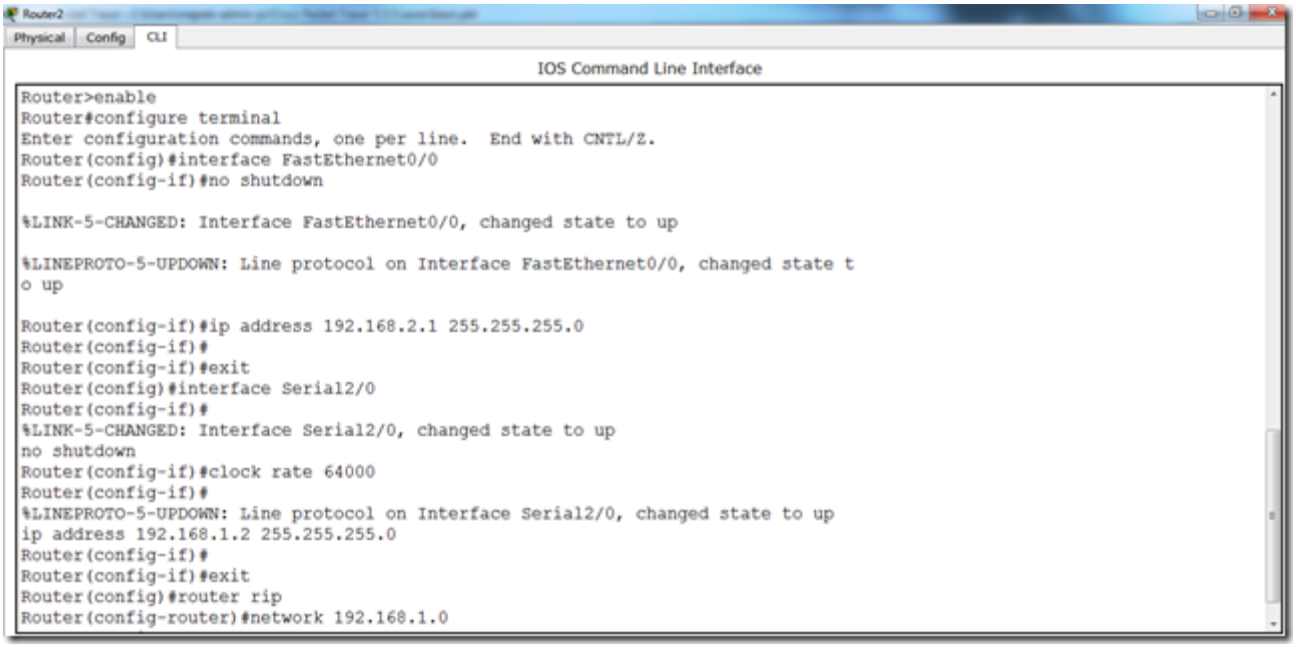
R2:
In order to apply protocol RIP, we will write the following set of commands on R2 as well.

Router(config)# router rip

Router(config-router)# network 192.168.1.0

Router(config-router)# network 192.168.2.0

Router(config-router)#exit

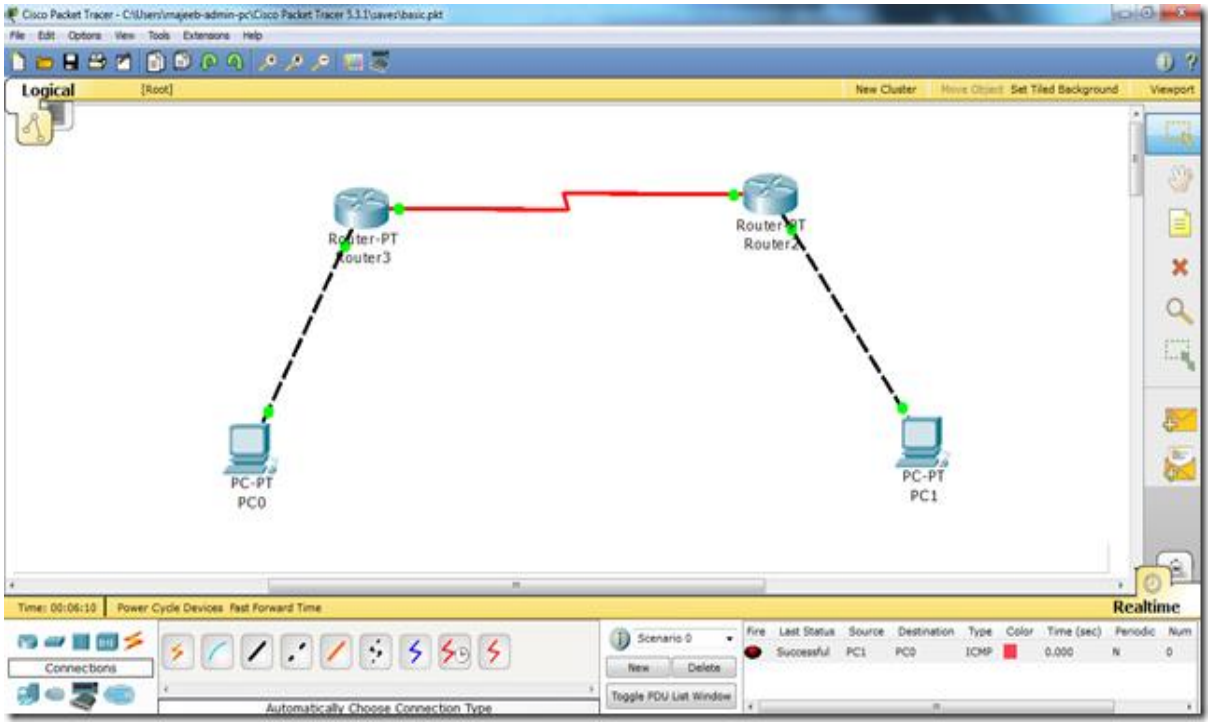


Write all the commands in the same fashion as in the above screen shots and voila, we are done with RIP protocol. Another important thing here is that we will add all the networks that we are using in our topology. Here in this particular example i am just using two networks x.x.1.0 and x.x.2.0 so thats why i have added these two network addresses to the RIP protocol.

Now, you can check it. Traffic is enabled and you can easily send data from PC0 to PC1.

RIP V2

There is no big difference between RIP version 1 and version 2 when we are applying them in packet tracer. In order to apply RIP version 2 on packet tracer. we will just have to add the following command. We will follow the same example that we used in RIP version 1 in this article.



Router(config)# router rip

Router(config-router)# network 192.168.1.0

Router(config-router)# network 192.168.2.0

Router(config-router)# version 2

Router(config-router)#exit

You see there is just the addition of one statement i.e. “version 2”. The rest is the same. We will apply the above set of commands on both routers i.e. Router 3 and Router 2 ,used in the topology above which is also used above , we have applied RIP V2 on packet tracer.

Just make sure that the protocol is applied as an additional step and cannot replace the basic steps i.e. we have to assign IP addresses to the router’s interfaces and PCs and also change the state of the interfaces from down to UP like we did above and then we will go ahead and apply Protocol.

2. Application of EIGRP on Packet Tracer

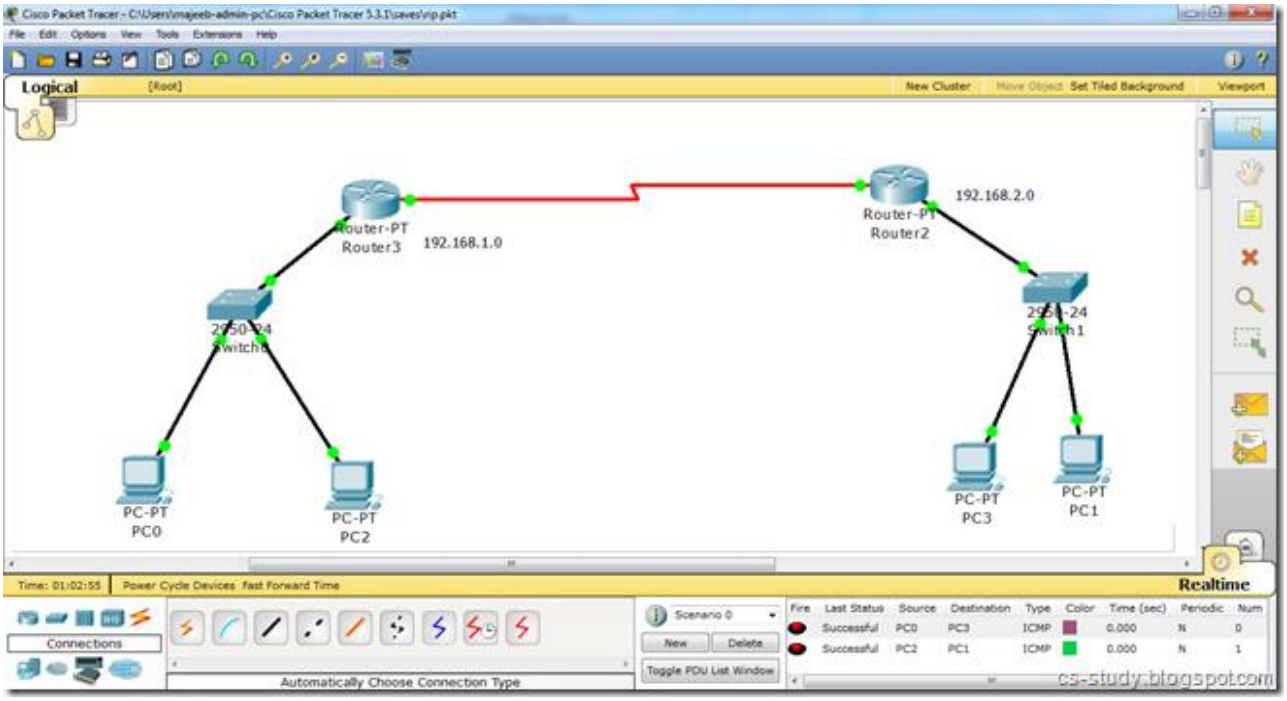
Hi everyone, today we are going to apply Enhanced Interior Gate Way Routing Protocol (EIGRP) on packet tracer. Here are the basic set of commands that we can apply on router CLI mode in order to apply EIGRP on router.

`Router(config)#router eigrp 100`	Turns on the EIGRP process. 100 is the autonomous system number, which can be a number between 1 and 65,535.
	All routers in the same autonomous system must use the same autonomous system number.
`Router(config-router)#network 10.0.0.0`	Specifies which network to advertise in EIGRP.
`Router(config-if)#bandwidth x`	Sets the bandwidth of this interface to *x* kilobits to allow EIGRP to make a better metric calculation.
	TIP: The **bandwidth** command is used for metric calculations only. It does not change interface performance.
`Router(config-router)#no network 10.0.0.0`	Removes the network from the EIGRP process.

Also, look at some additional commands.

`Router#show ip eigrp neighbors`	Displays the neighbor table.
`Router#show ip eigrp neighbors detail`	Displays a detailed neighbor table.
	TIP: The **show ip eigrp neighbors detail** command verifies whether a neighbor is configured as a stub router.
`Router#show ip eigrp interfaces`	Shows information for each interface.
`Router#show ip eigrp interfaces serial 0/0`	Shows information for a specific interface.
`Router#show ip eigrp interfaces 100`	Shows information for interfaces running process 100.
`Router#show ip eigrp topology`	Displays the topology table.
	TIP: The **show ip eigrp topology** command shows you where your feasible successors are.
`Router#show ip eigrp traffic`	Shows the number and type of packets sent and received.
`Router#show ip route eigrp`	Shows a routing table with only EIGRP entries.

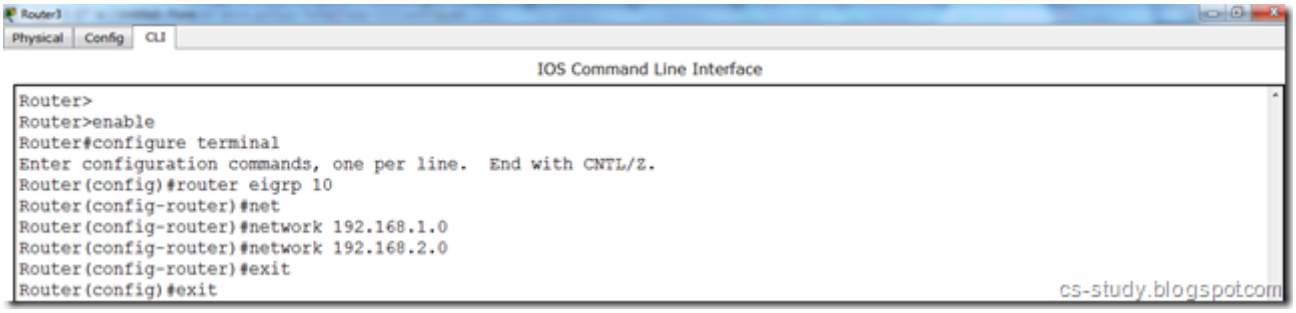
Now, we are going to apply EIGRP on the following topology.



Now, after successfully applying IP addresses like in this topology, we will apply following commands.

Router(config)#router eigrp 10
Router(config-router)#network 192.168.1.0
Router(config-router)#network 192.168.2.0
Router(config-router)#exit

Apply the above set of commands on both routers like this.

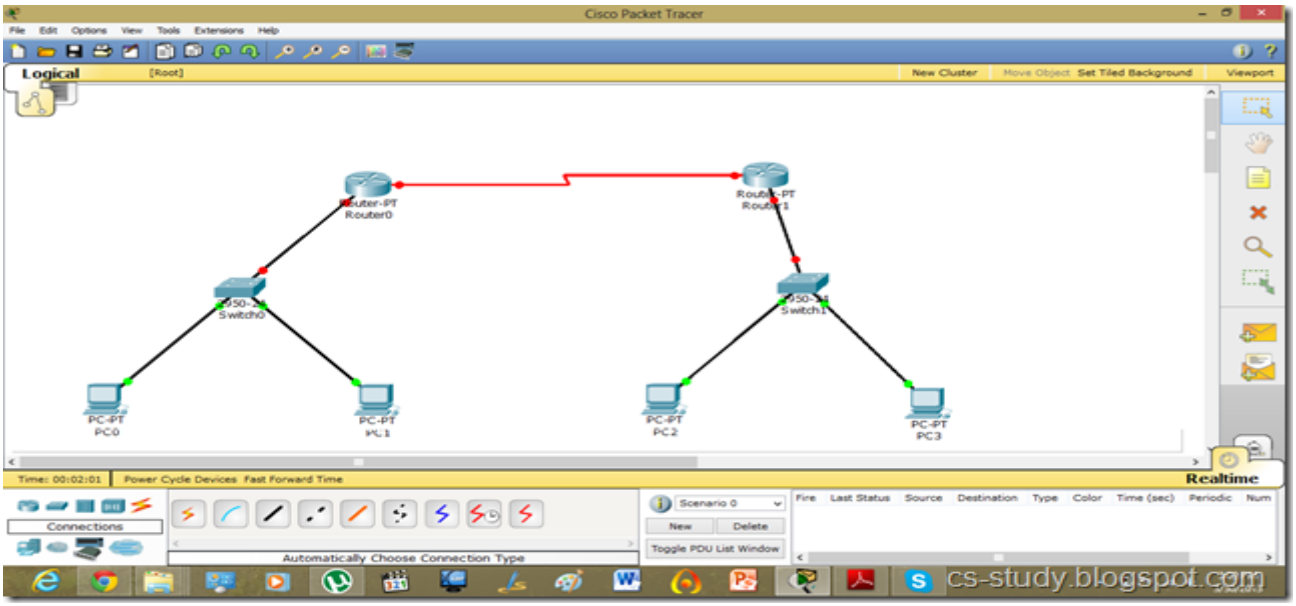


And eigrp protocol has been applied on this topology. Notice the following command.

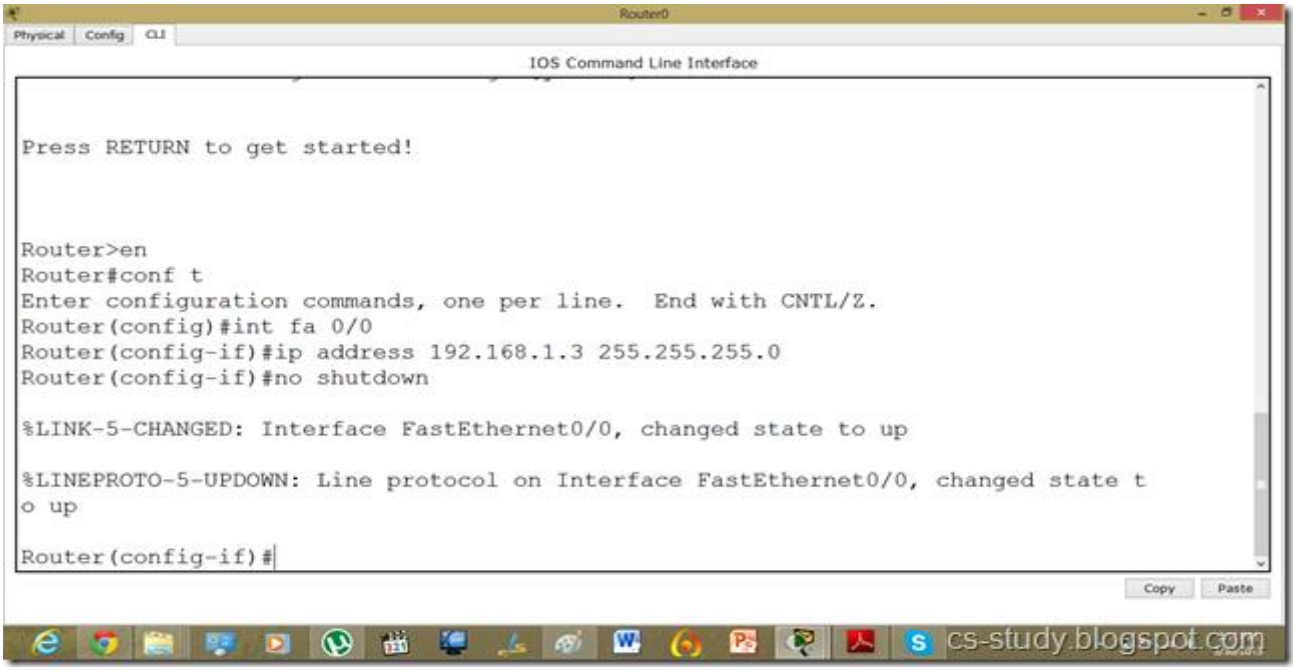
router eigrp 10
This number ‘10' is the process ID.

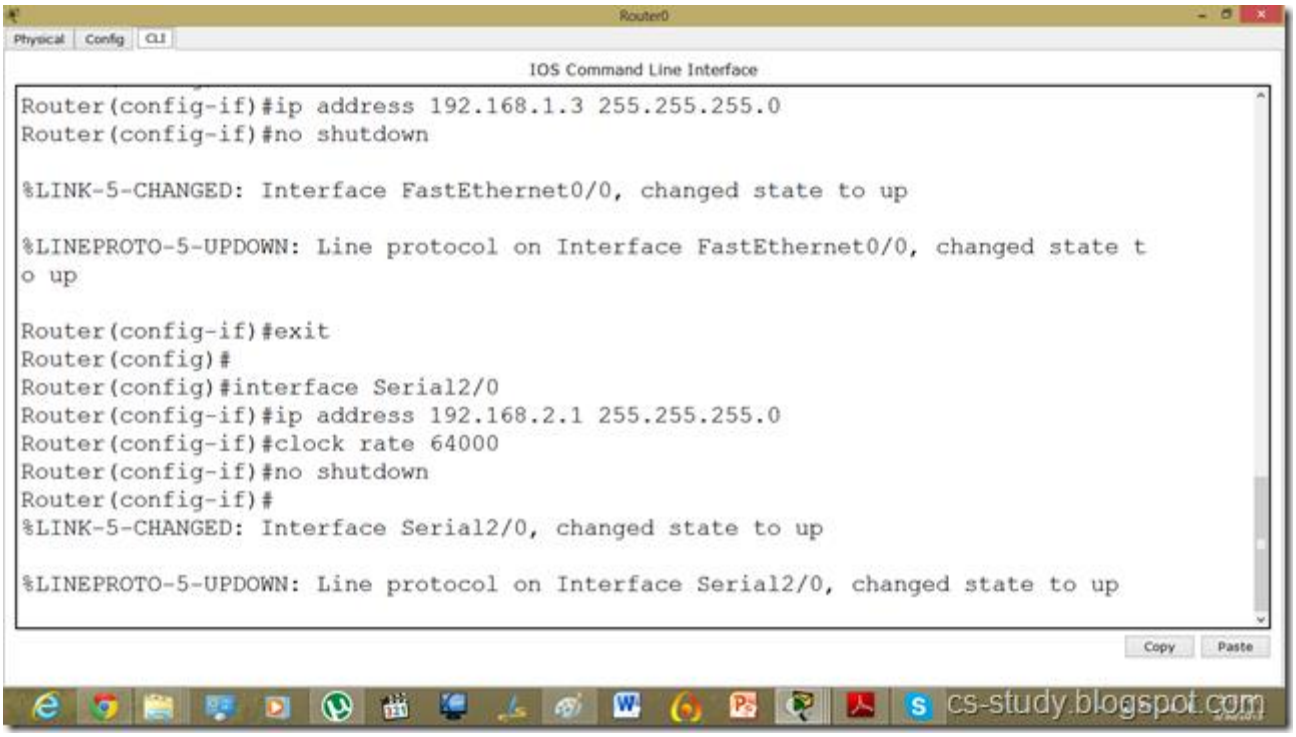
3. OSPF application on Packet Tracer

Now, we are going to apply OSPF(open shortest path first) protocol on packet tracer. Let us take the following simple topology.

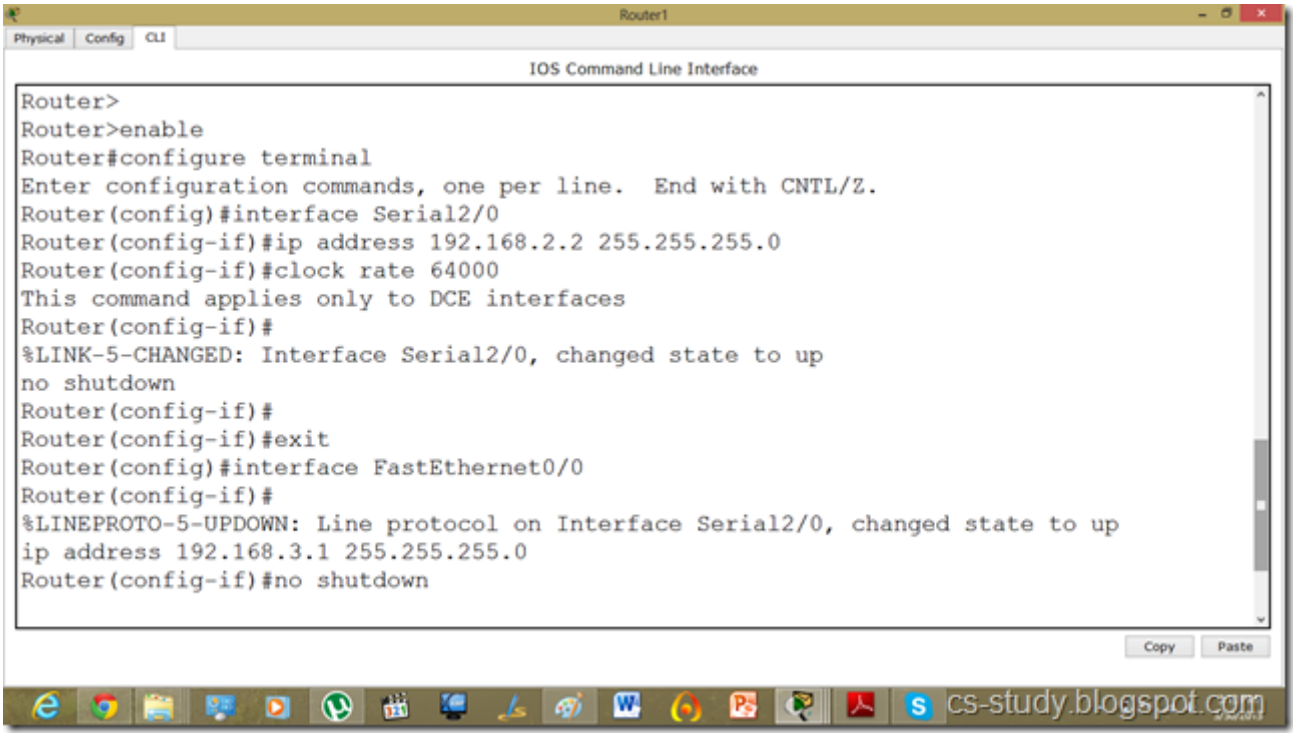


Now, let us apply the ospf on it. But before that, as usual, let us assign IP addresses and change the state of interfaces.

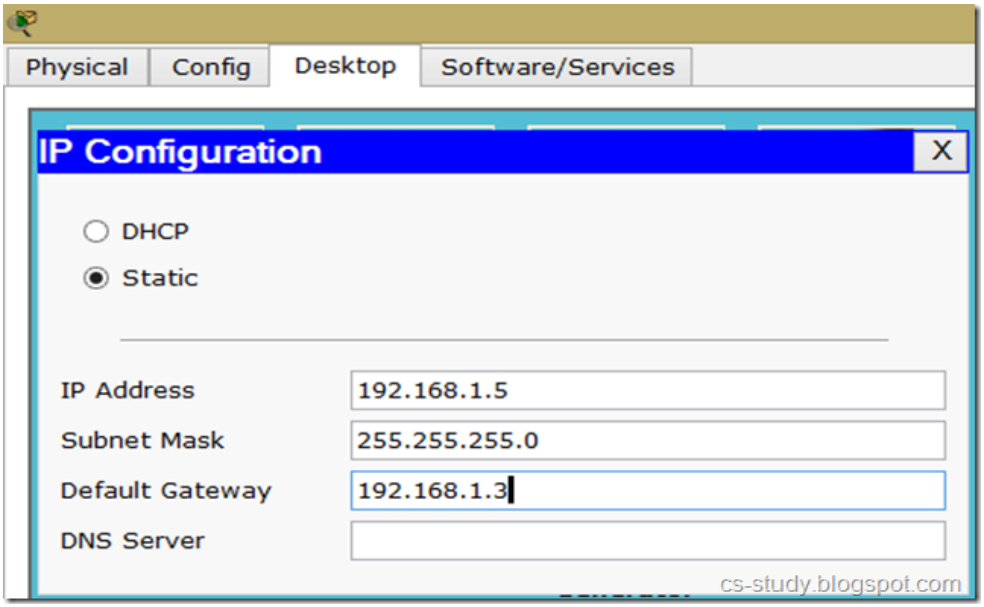


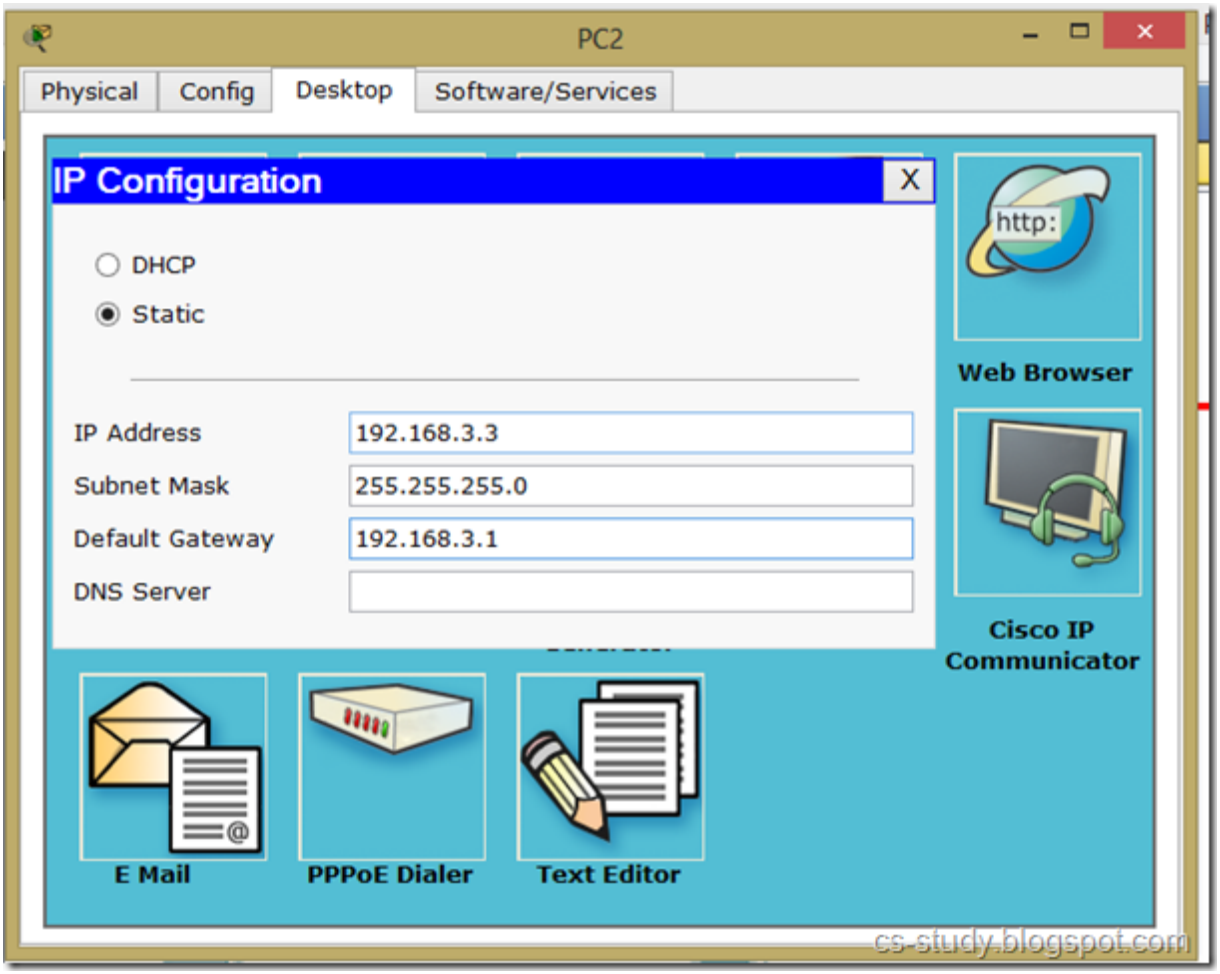


Similarly for the other router.

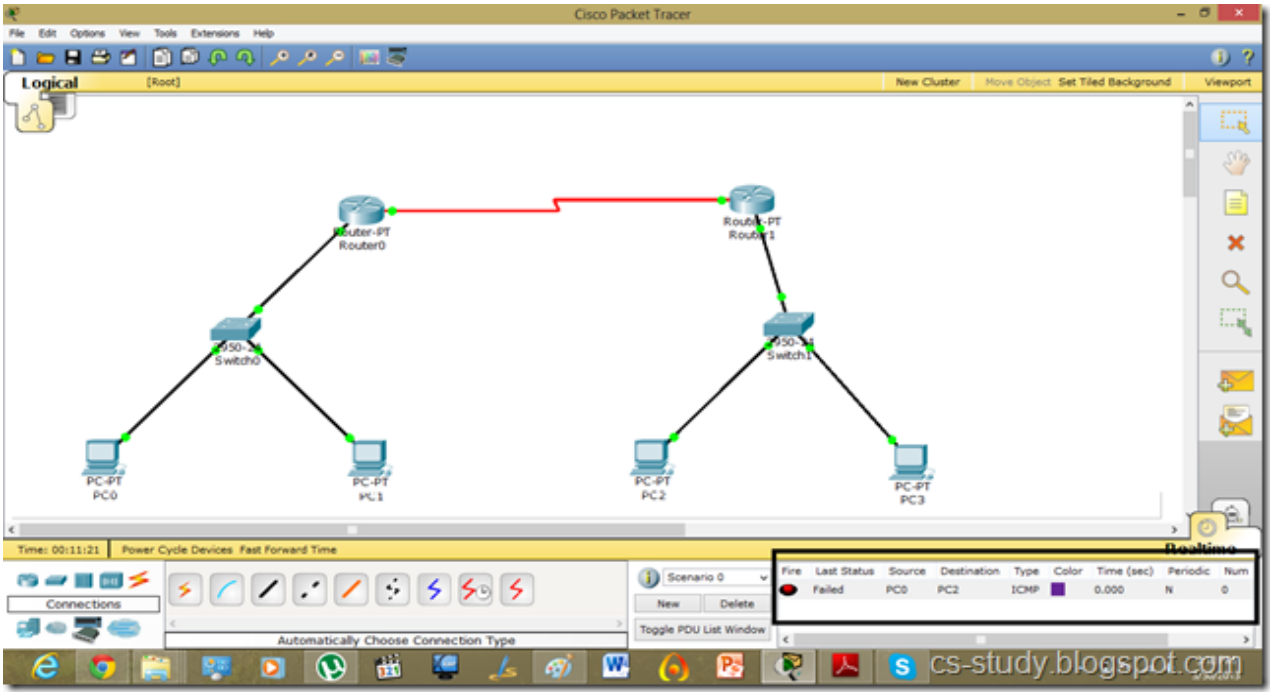


Assigning the IP addresses to PCs as follows.

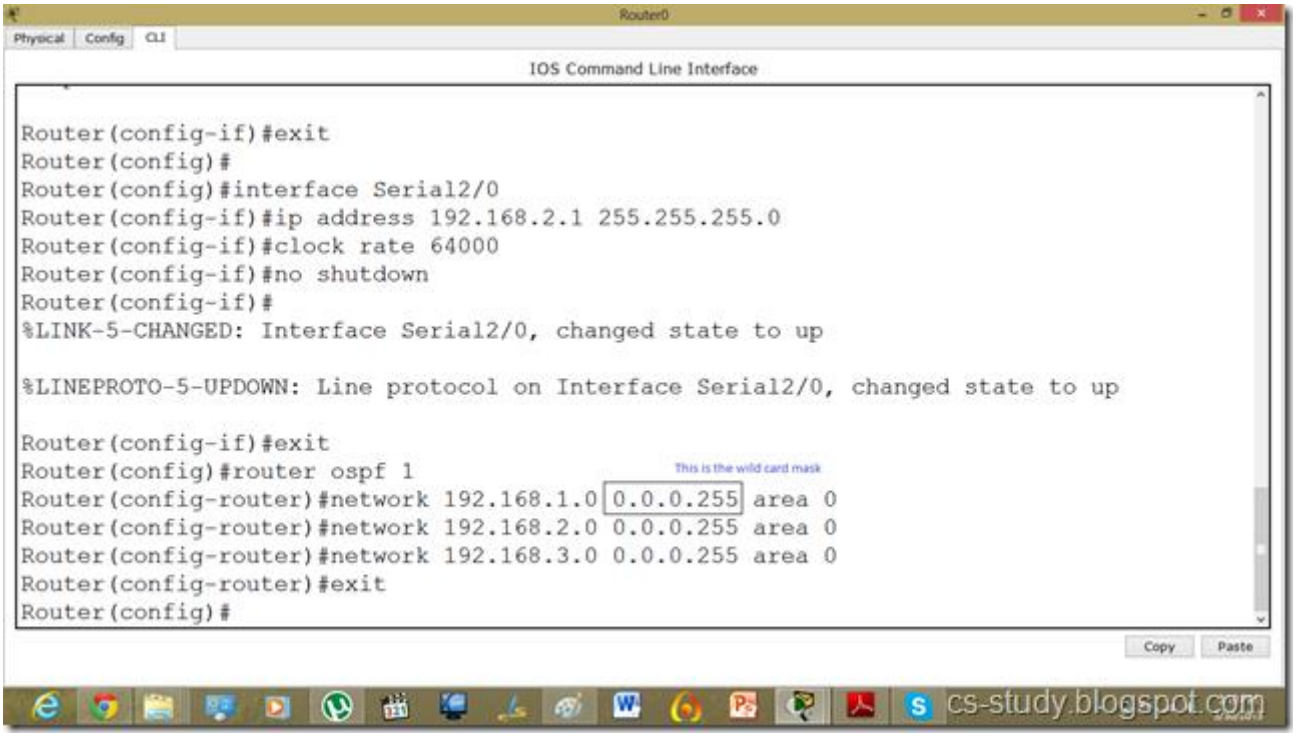

Physical
Config
Desktop
Software/Services
IP Configuration
DHCP
Static
IP Address
192.168.1.5
Subnet Mask
255.255.255.0
Default Gateway
192.168.1.3
DNS Server
cs-study.blogspot.com


PC2
Physical
Config
Desktop
Software/Services
IP Configuration
DHCP
Static
IP Address
192.168.3.3
Subnet Mask
255.255.255.0
Default Gateway
192.168.3.1
DNS Server
Web Browser
Cisco IP Communicator
E Mail
PPPoE Dialer
Text Editor
cs-study.blogspot.com

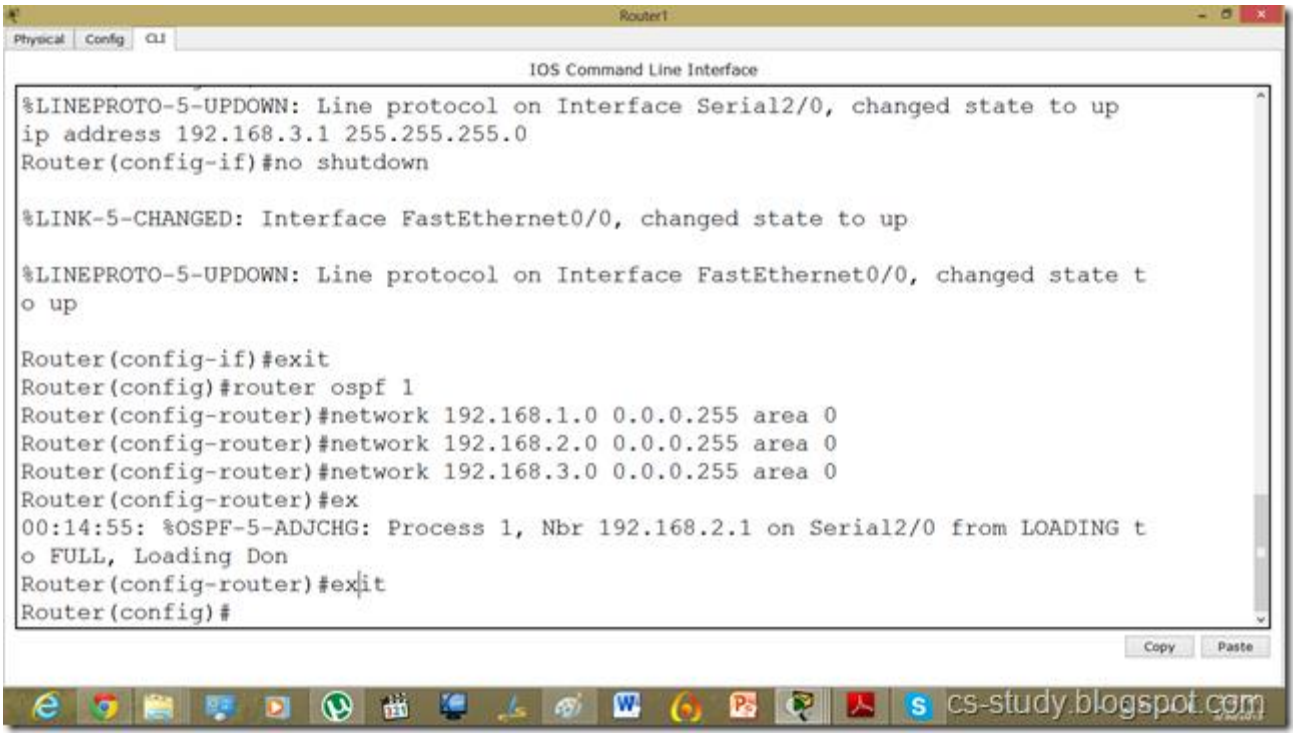
Now, as we can see, interfaces are up but the communication is not enabled because we have not applied the protocol yet.



Lets do it.



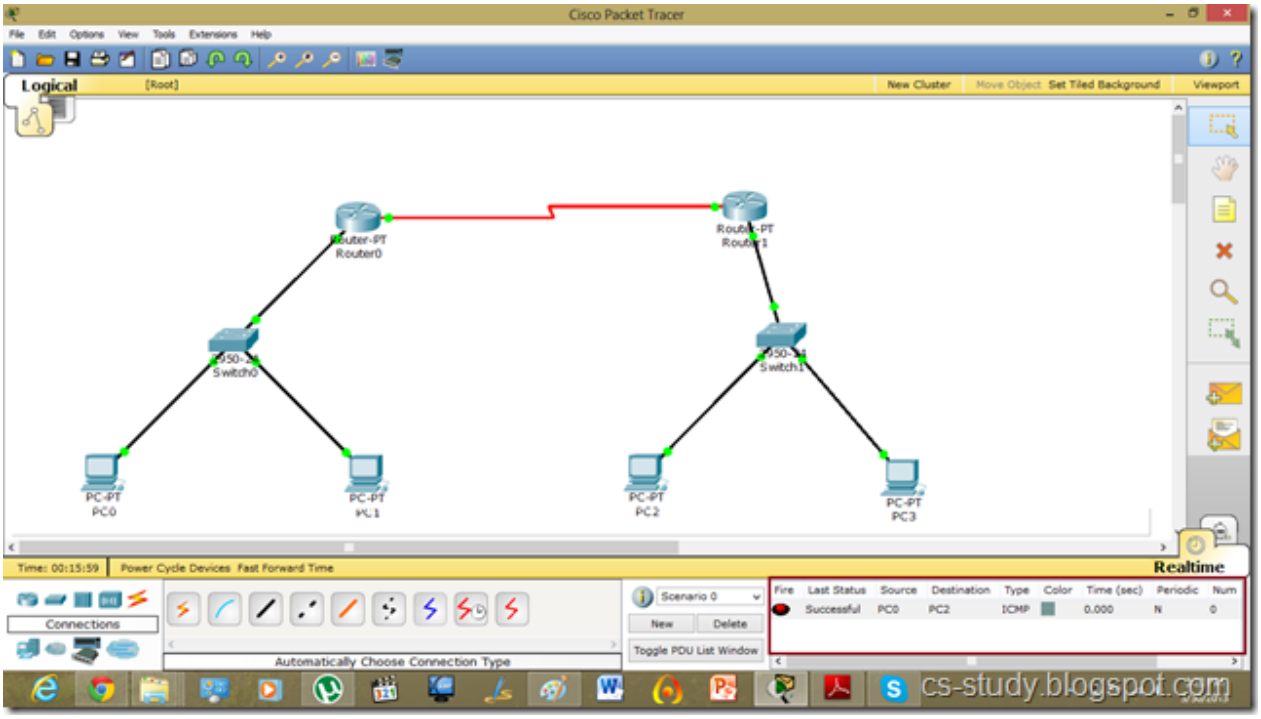
On router 1.



After applying protocol successfully , the traffic is flowing . Couple of things worth discussing

i. you will have to provide area id and process id on ospf protocol.

ii. you will have to provide wildcard mask on ospf.



Lab 7 – DHCP and DNS implementation using Packet Tracer

Objectives

In this lab, students will learn how to implement Dynamic Host Configuration Protocol and Domain Name Services on packet tracer

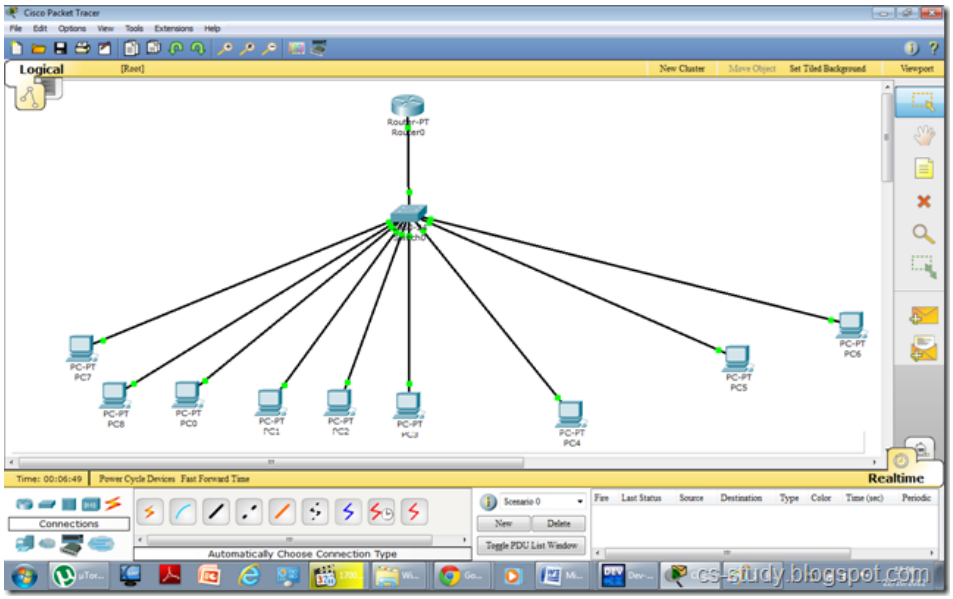
Lab Tasks

Task 1: Apply DHCP on packet tracer

This tutorial is about how to configure dhcp on cisco router in packet tracer. The Dynamic Host Configuration Protocol (DHCP) is a network protocol that is used to configure network devices. DHCP allows a computer to join an IP-based network without having a pre-configured IP address. DHCP is a protocol that assigns unique IP addresses to devices, then releases and renews these addresses as devices leave and re-join the network. Internet Service Providers (ISPs) usually use DHCP to allow customers to join the Internet with minimum effort. The DHCP server maintains a database of available IP addresses and configuration information. When it receives a request from a client, the DHCP server determines the network to which the DHCP client is connected, and then allocates an IP address. DHCP servers typically grant IP addresses to clients only for a limited interval.

Lets apply DHCP on packet tracer.

First, let us make a topology with one router on which we will apply DHCP and several client PCs. More like this one,



Now, we will apply DHCP on the router.

The commands in sequence are as follows.

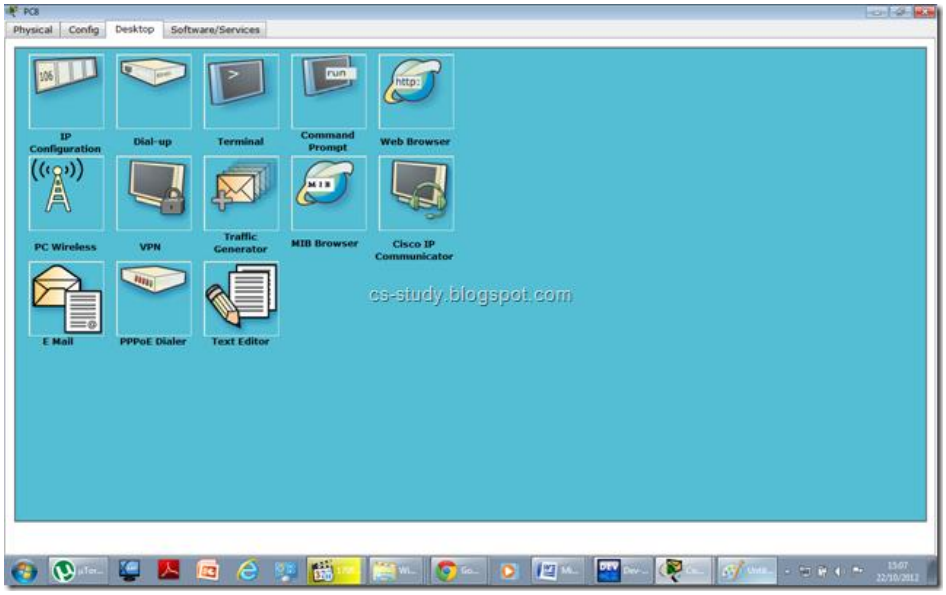
In the following command "ip dhcp pool cisco", we are creating a pool for DHCP called cisco. cisco is the name here and we can name it whatever we want.

Similarly, in the command "default-router" we are telling the DHCP about the default route to follow.

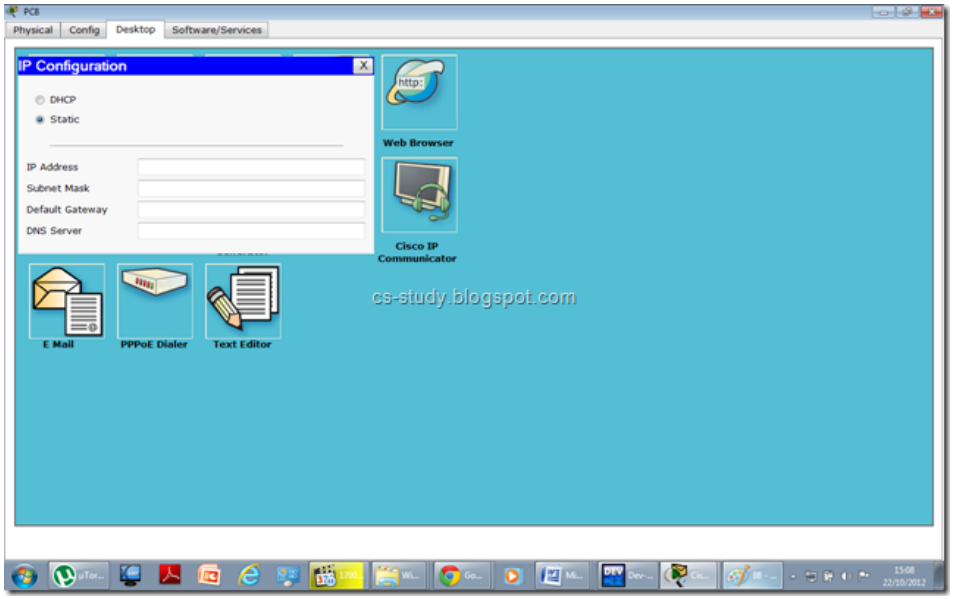
Notice, after we exit from DHCP mode, we are excluding some IP addresses by applying this command "ip dhcp excluded-addresses x-x", where x is the starting and ending IP address respectively. We are basically reserving some IPs for our use. It can be used to attach printers or assign it to some specific users for security purposes. You can also give dns address in dhcp by using the following command.

dns-server 192.168.1.15.

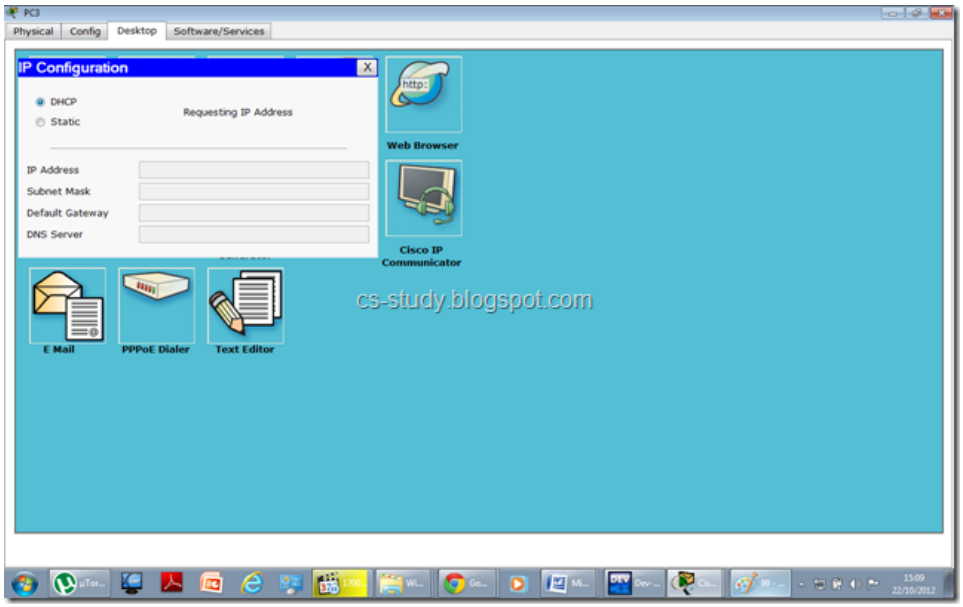
Now, open the PC.



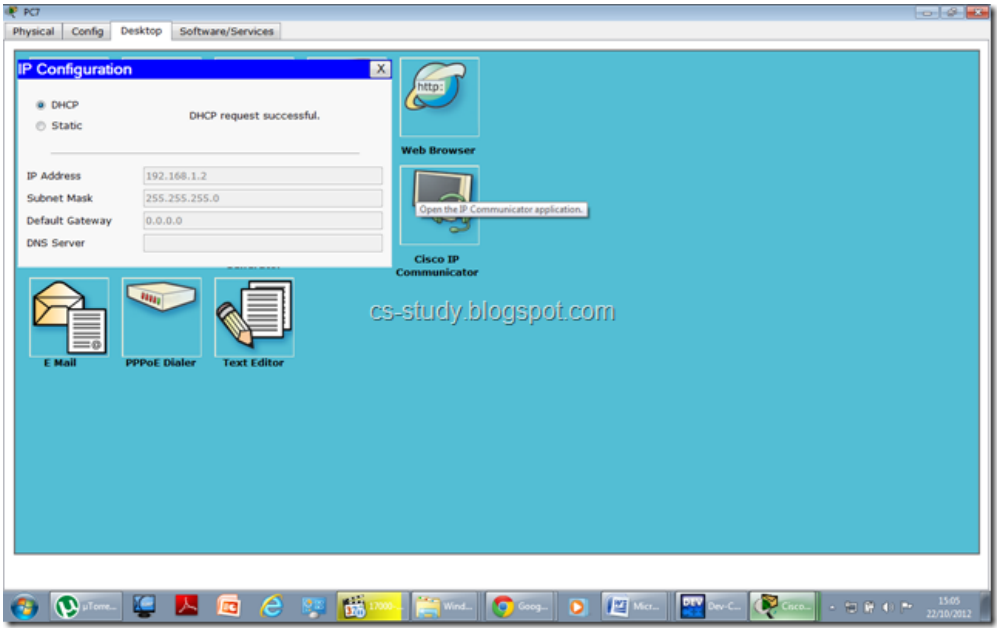
Click on IP Configuration



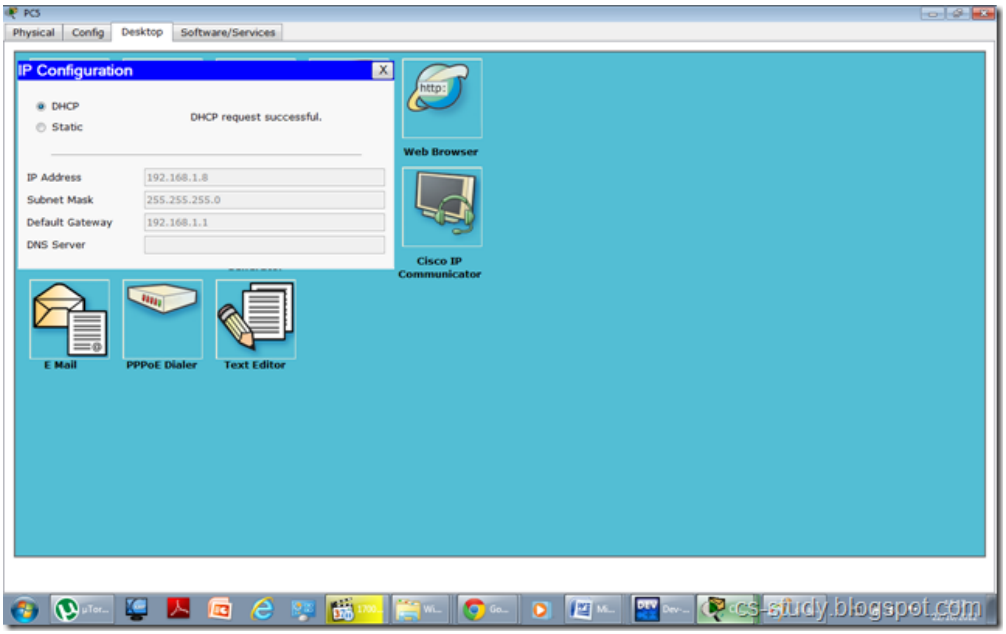
Select from Static to DHCP



And after DHCP request is completed you will see the following screen.



Now, after applying some IPs in sequence, DHCP will skip the IPs that we have excluded from our DHCP pool.



That is all, we have applied DHCP on packet tracer.

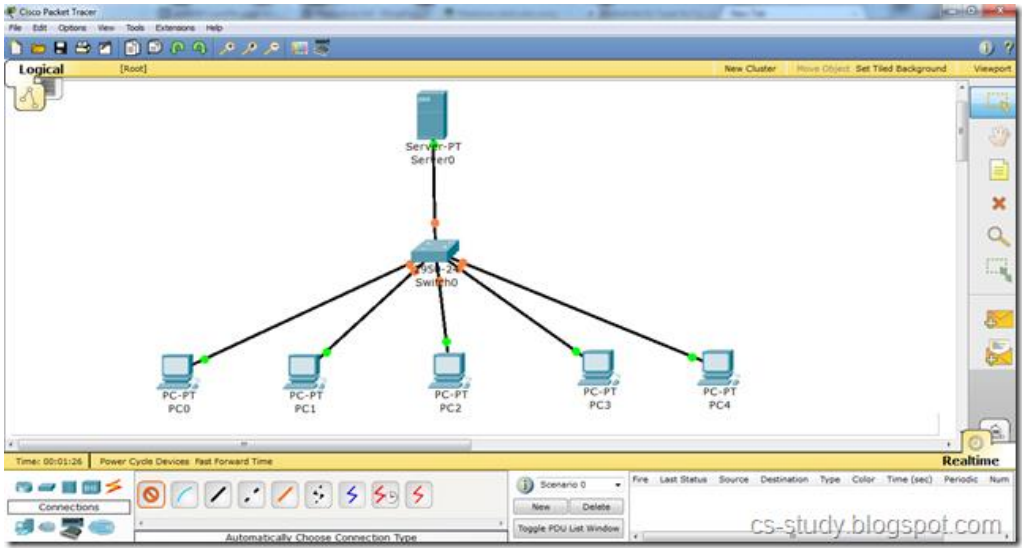
Task 2: DHCP on packet tracer through Server

Objective

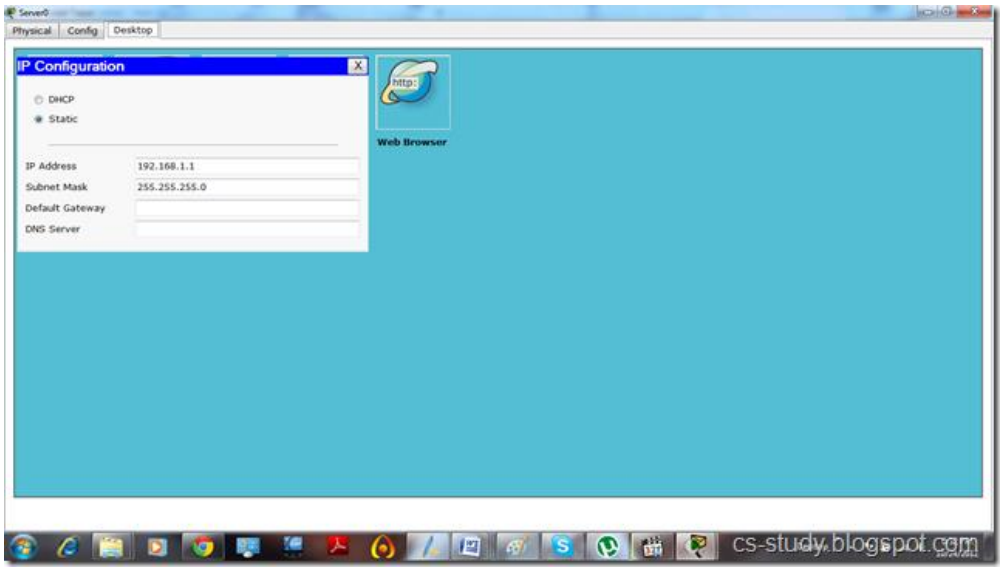
The objective of this lab is to apply DHCP on servers using packet tracer.

Solution

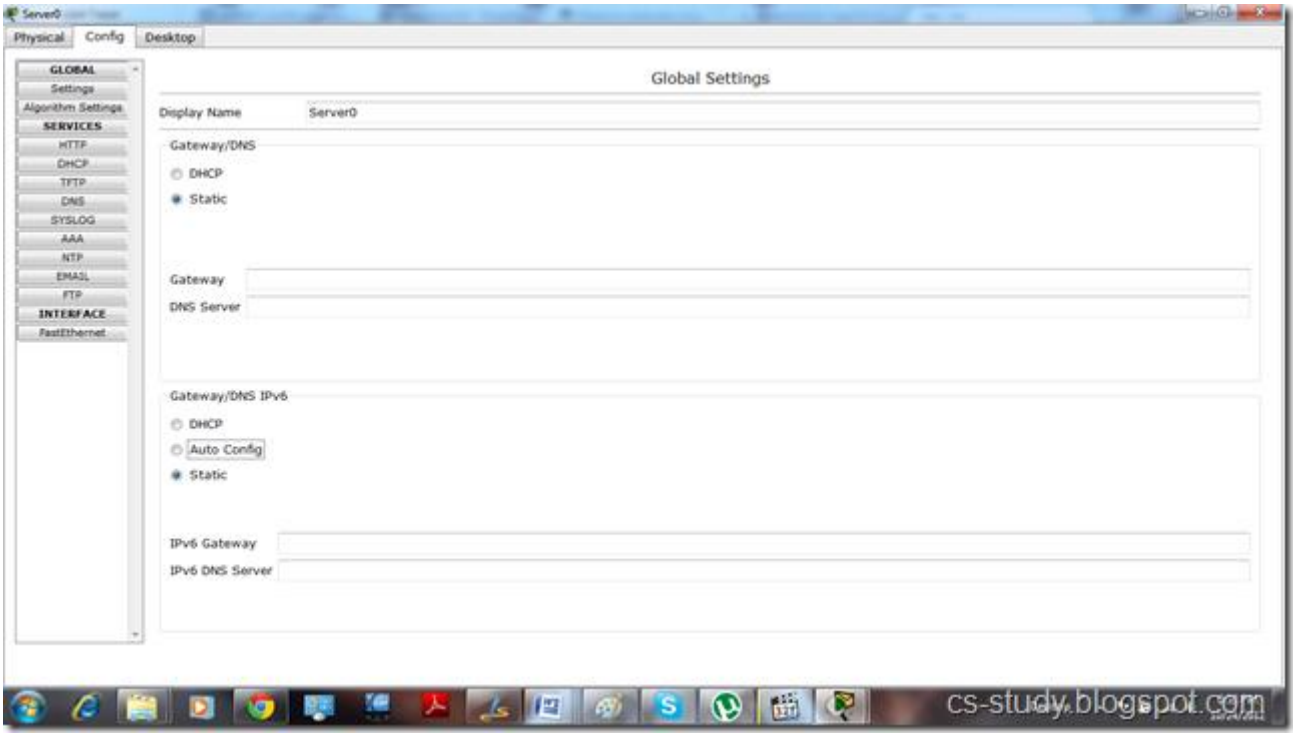
We are going to apply DHCP on server and PCs will be assigned IP addresses through DHCP.



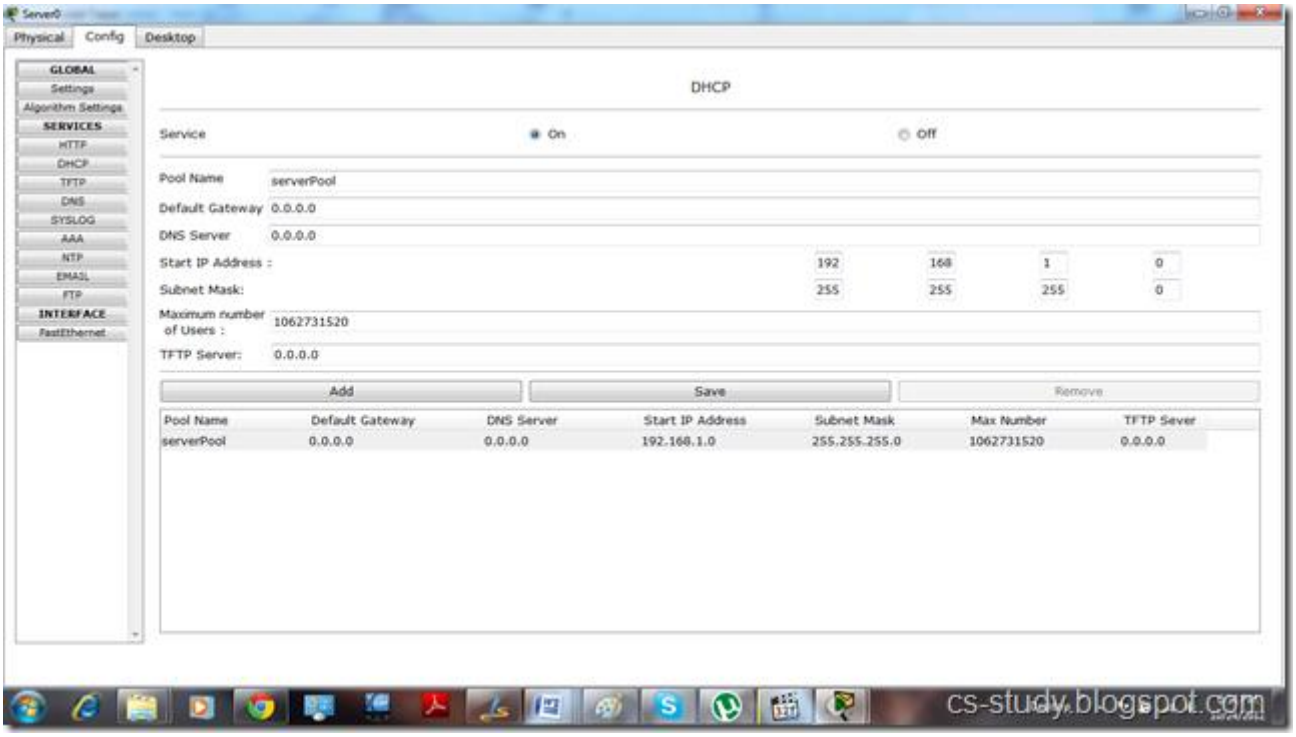
Open the server and go to the Desktop tab, click IP Configuration and enter the IP address.



Now, go to the Config tab.



And go to the DHCP

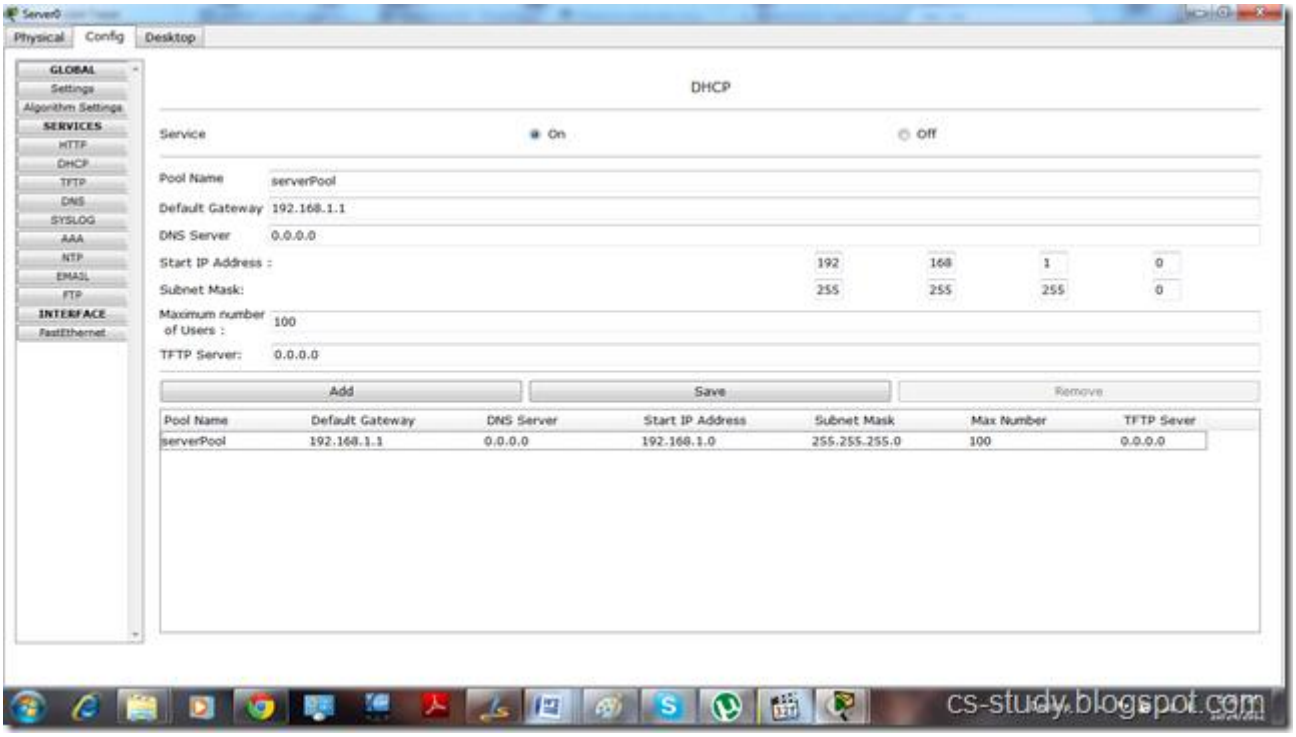


i. Enter IP for default Gateway.

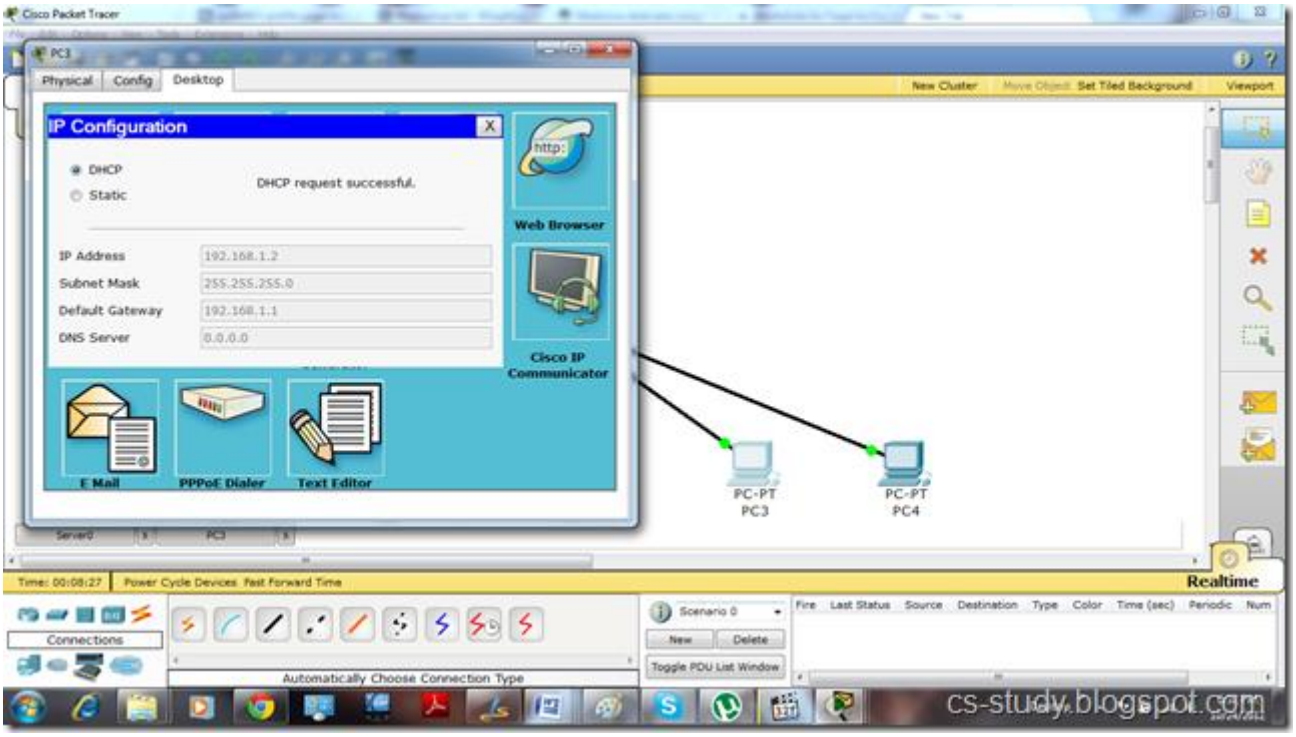
ii. Start IP address

iii.. Maximum number of Users.

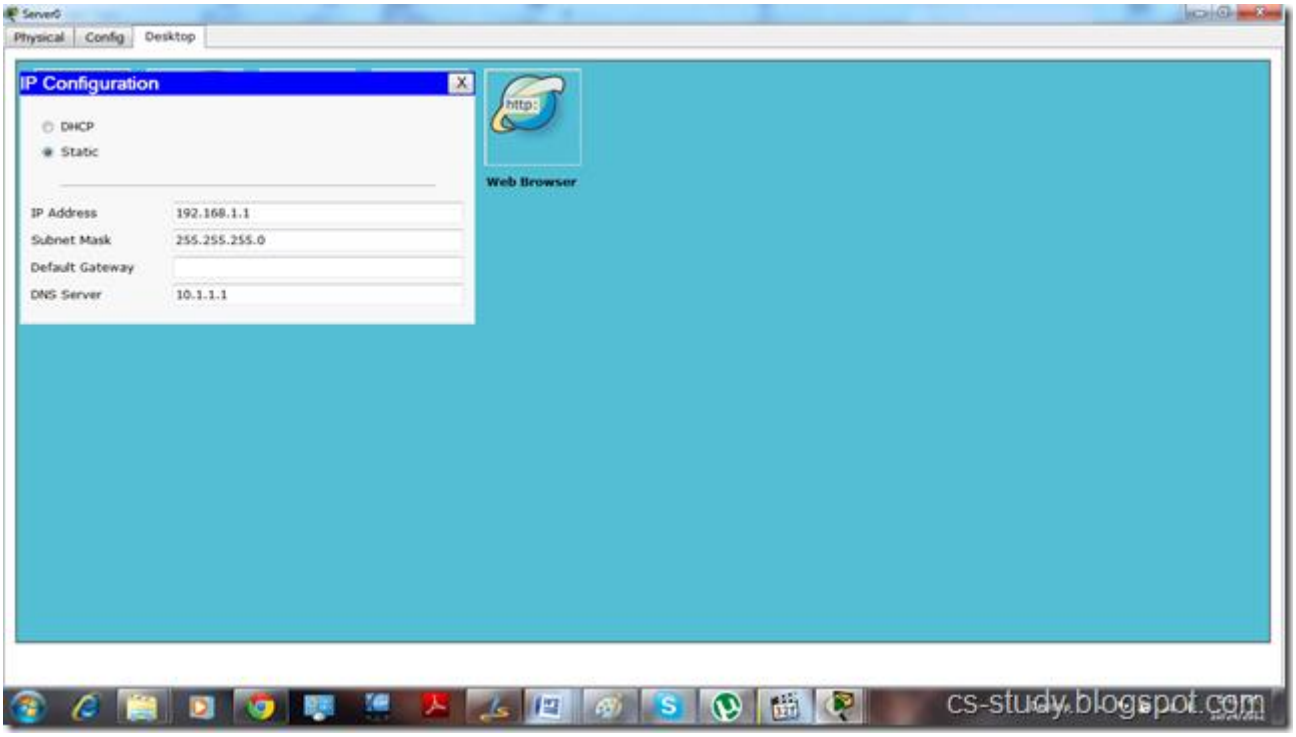
iv. Click Save.



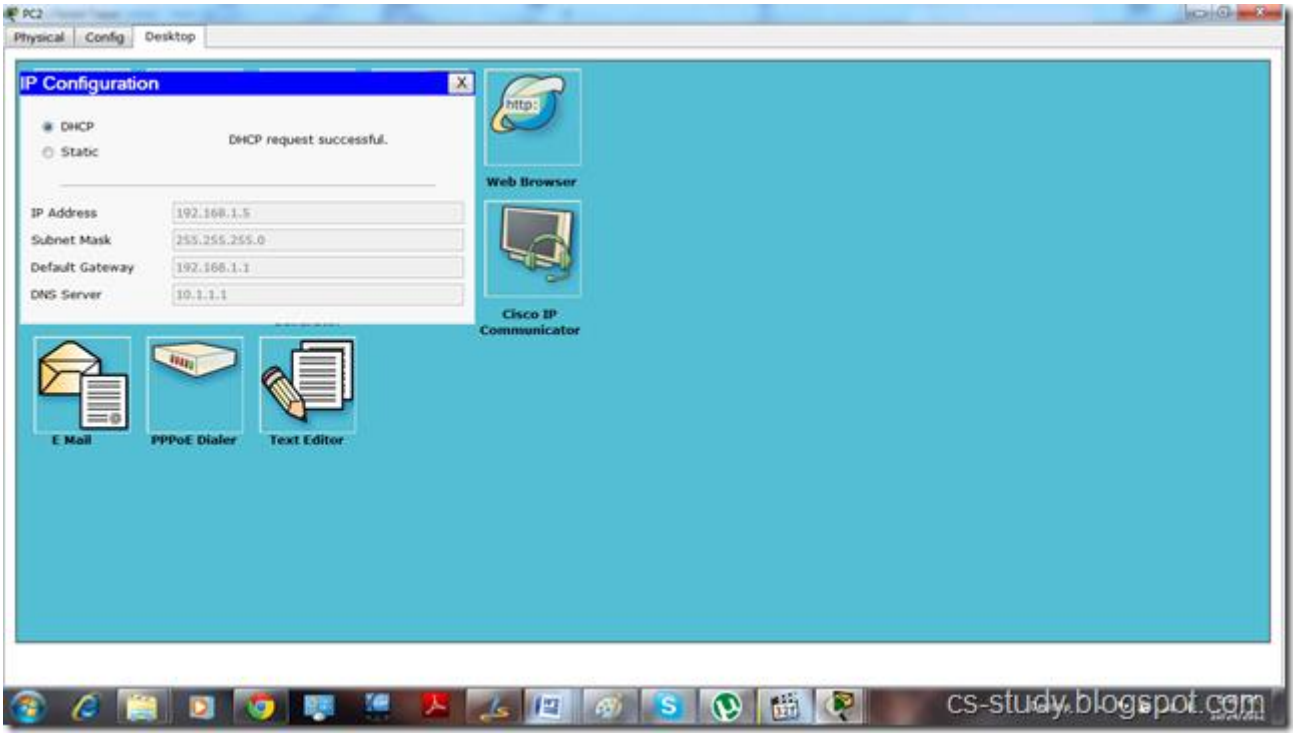
Now, click on any PC that is attached to the server, go to IP configuration and select DHCP. You will see that DHCP will successfully assign IP address to the PC



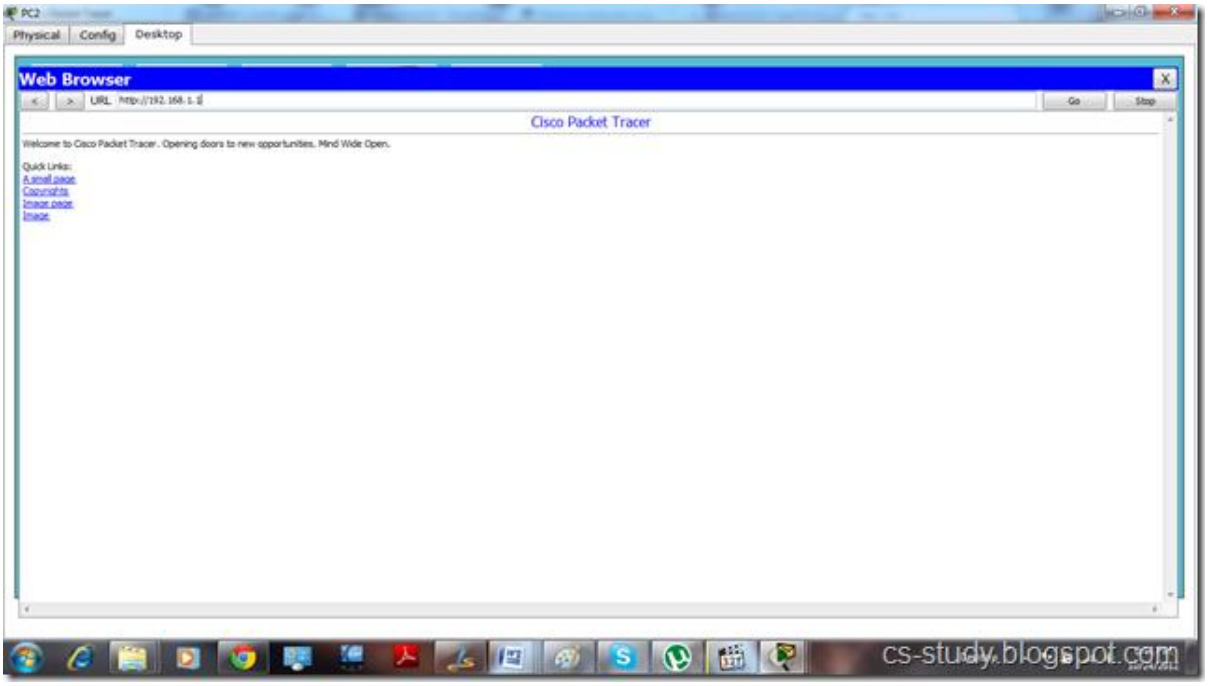
Now, if we go back to server and assign DNS Server address and then go to any PC and select DHCP.



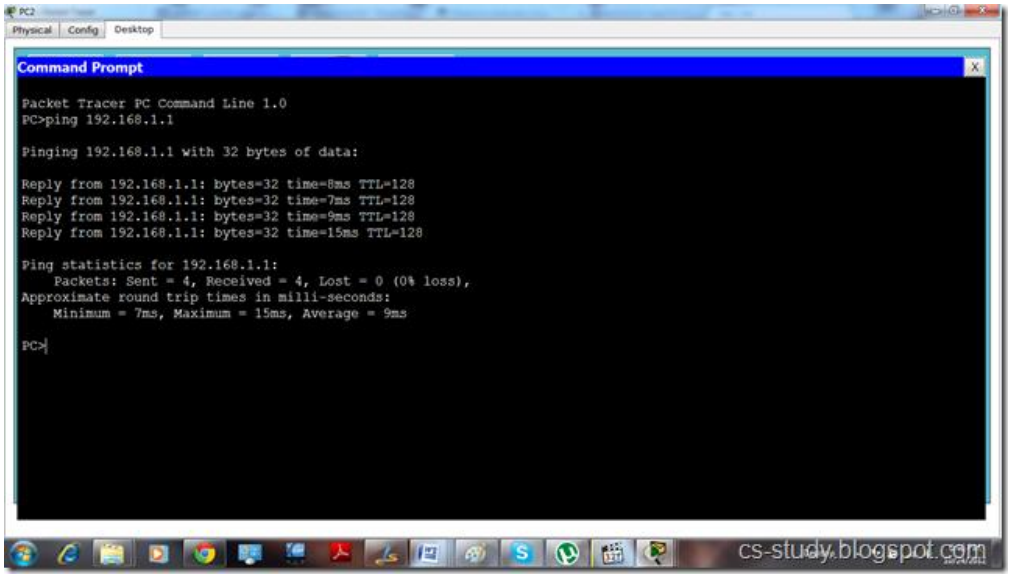
It will also assign DNS to the PC as well.



We can also open the website of the server through any PC by going to the Web Browser option and entering the IP address of the server.

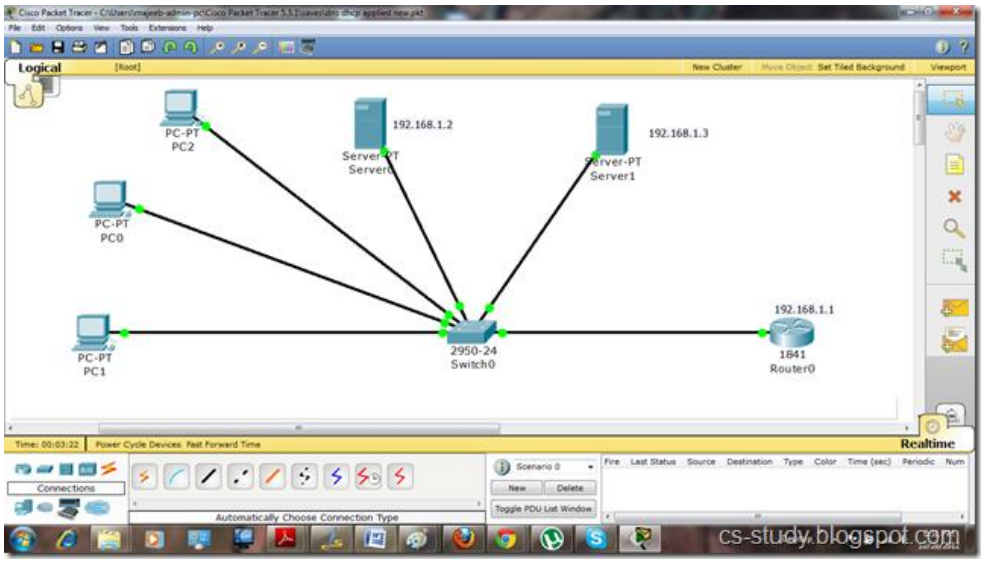


And we can ping the server by going to the PC's command prompt and entering server's IP address.

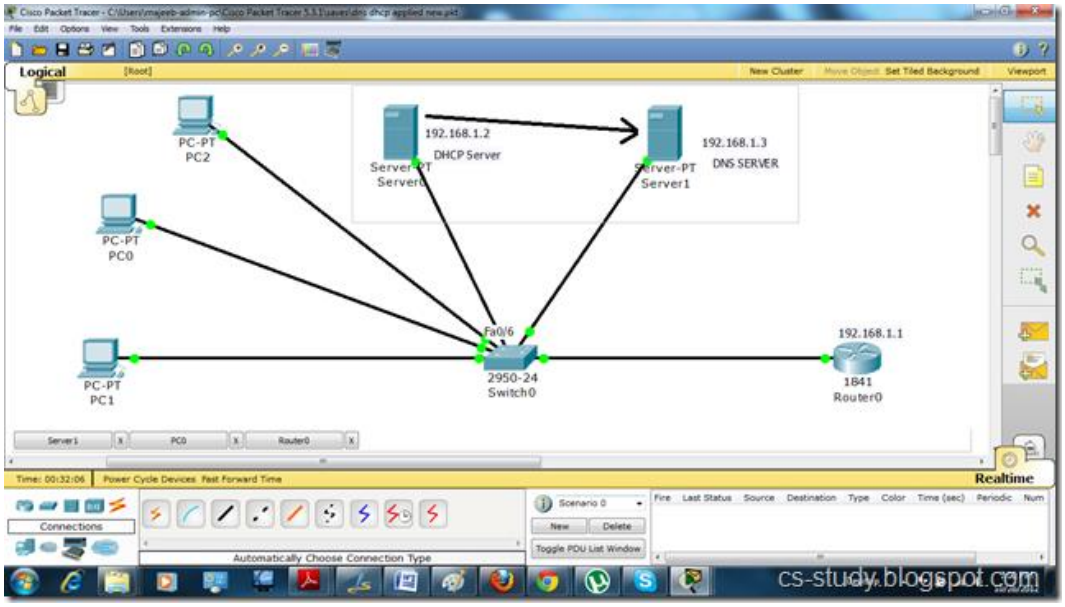


Task 3: Apply DNS on packet tracer

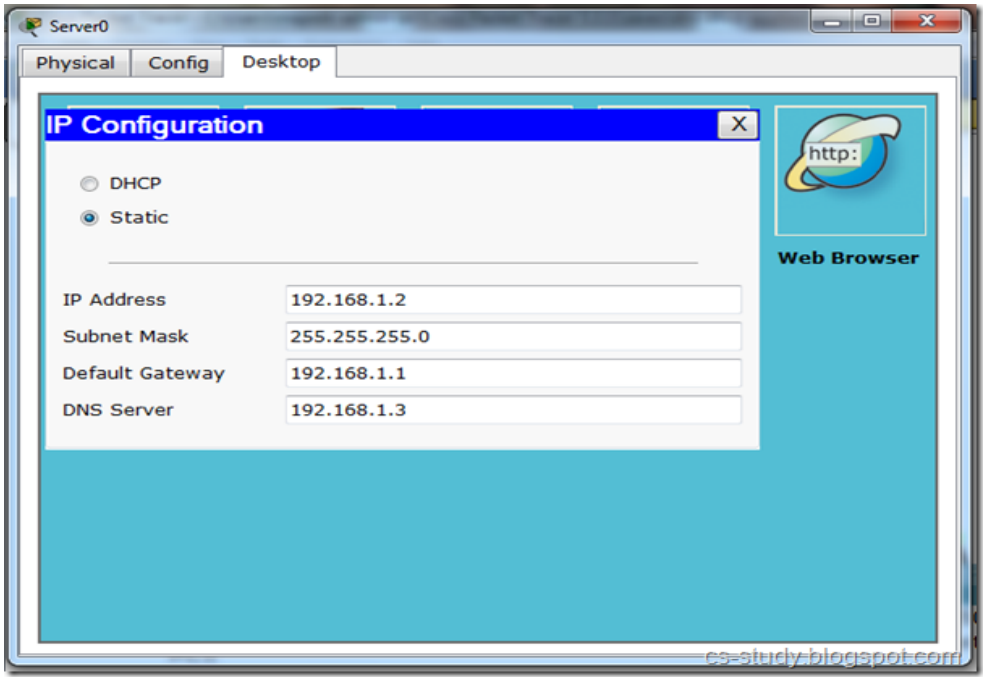
we are going to set a dns (domain name system) server and a dhcp server. And then from our PC we will use dns service.



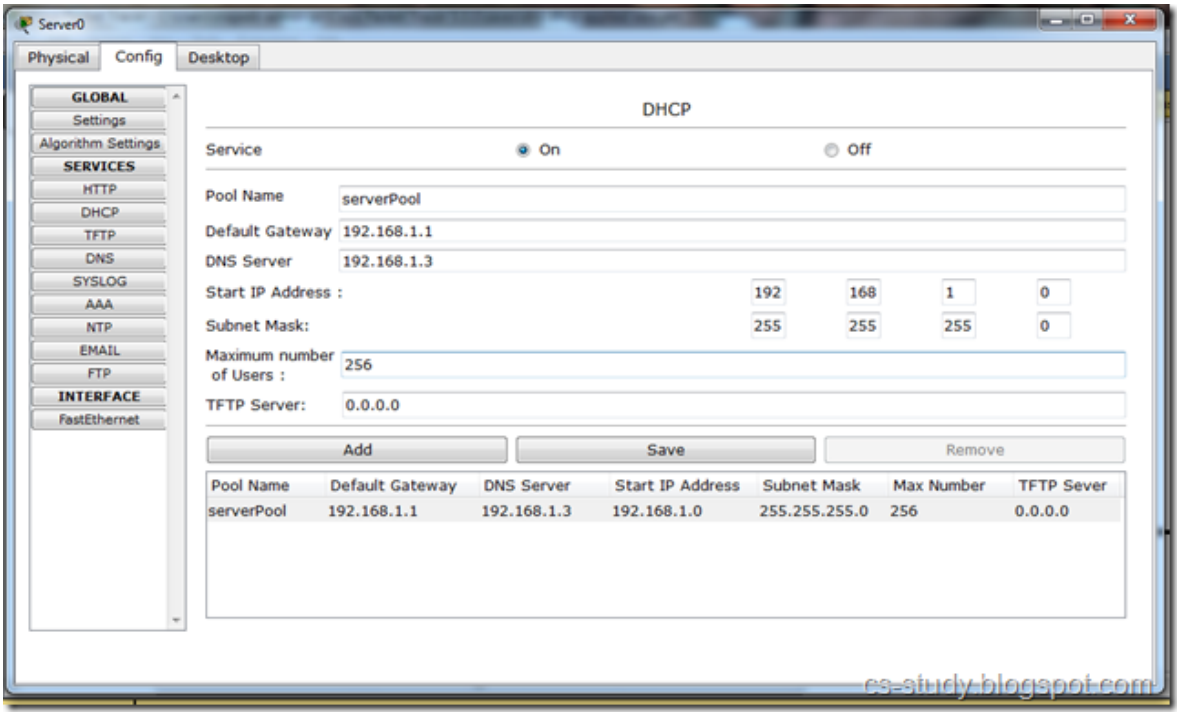
Server 0 in the above topology is our dhcp server and Server 1 is our dns server.



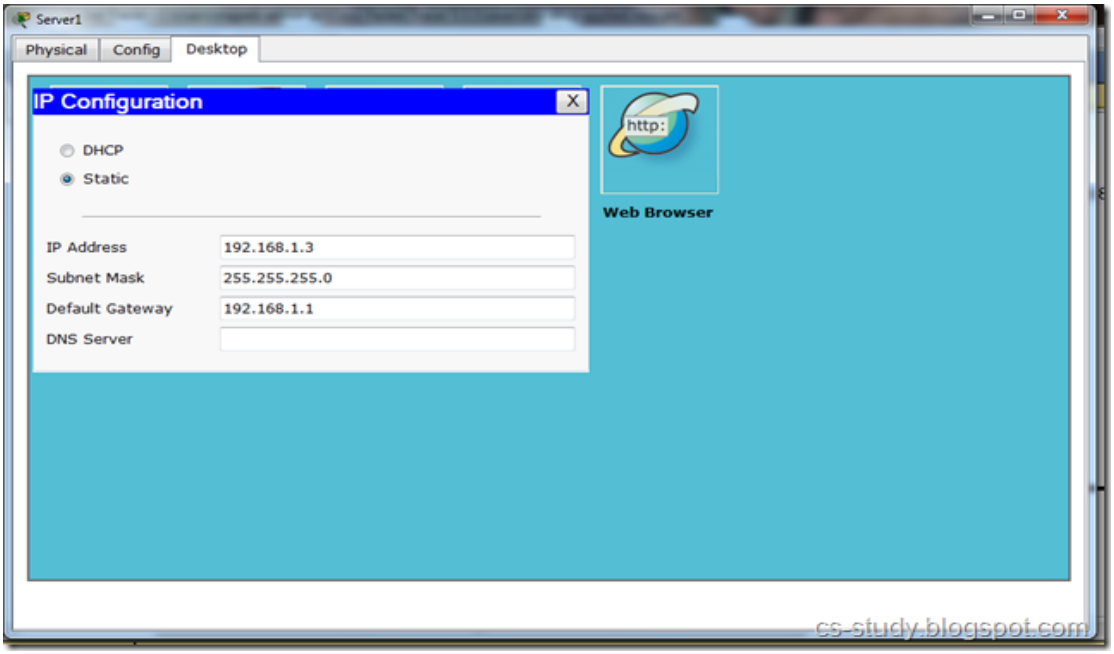
Set up IP on server 0.



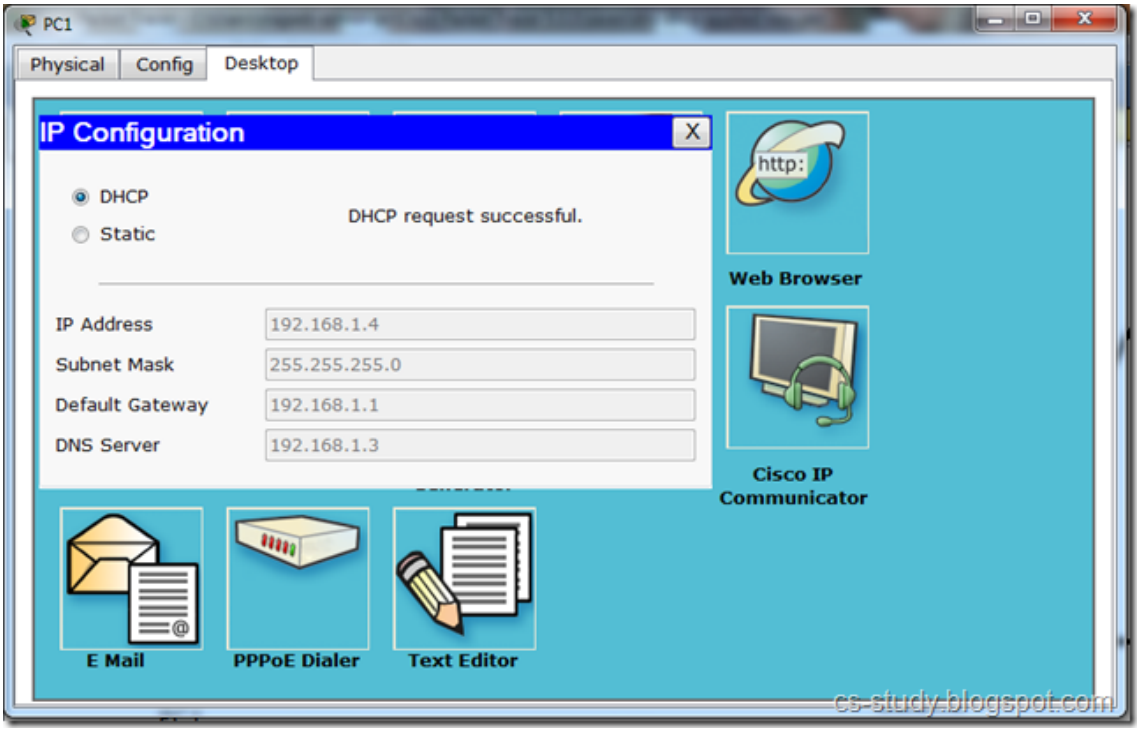
Set up DHCP on server 0.



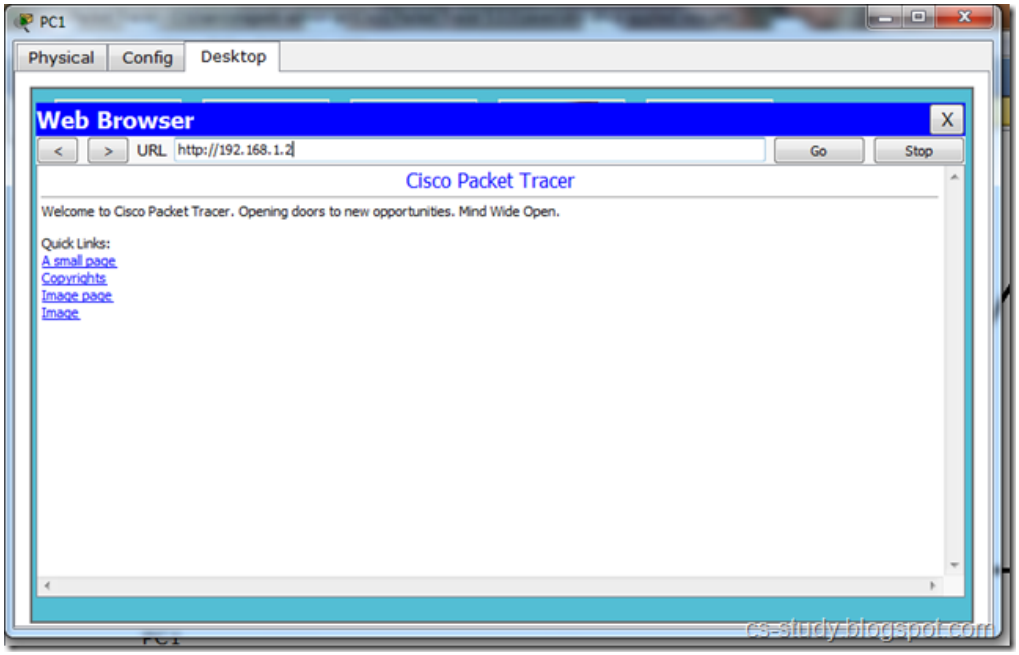
Set up IP on server 1.



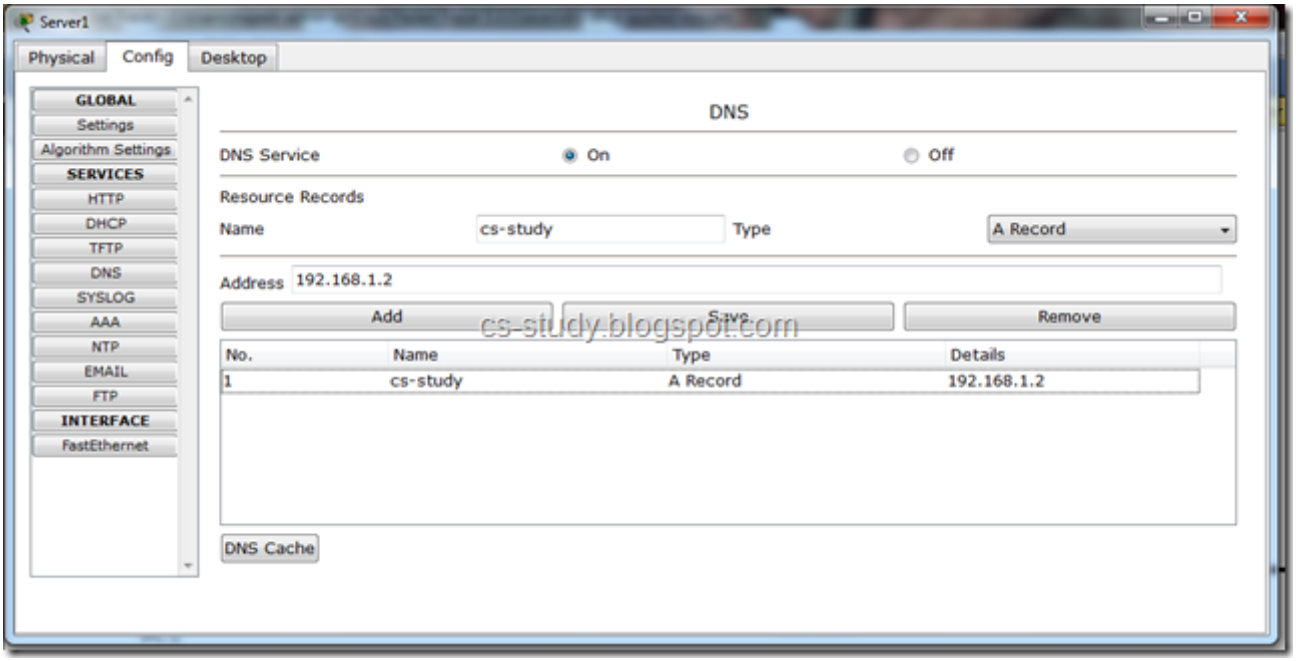
Now, go to PC and select DHCP.



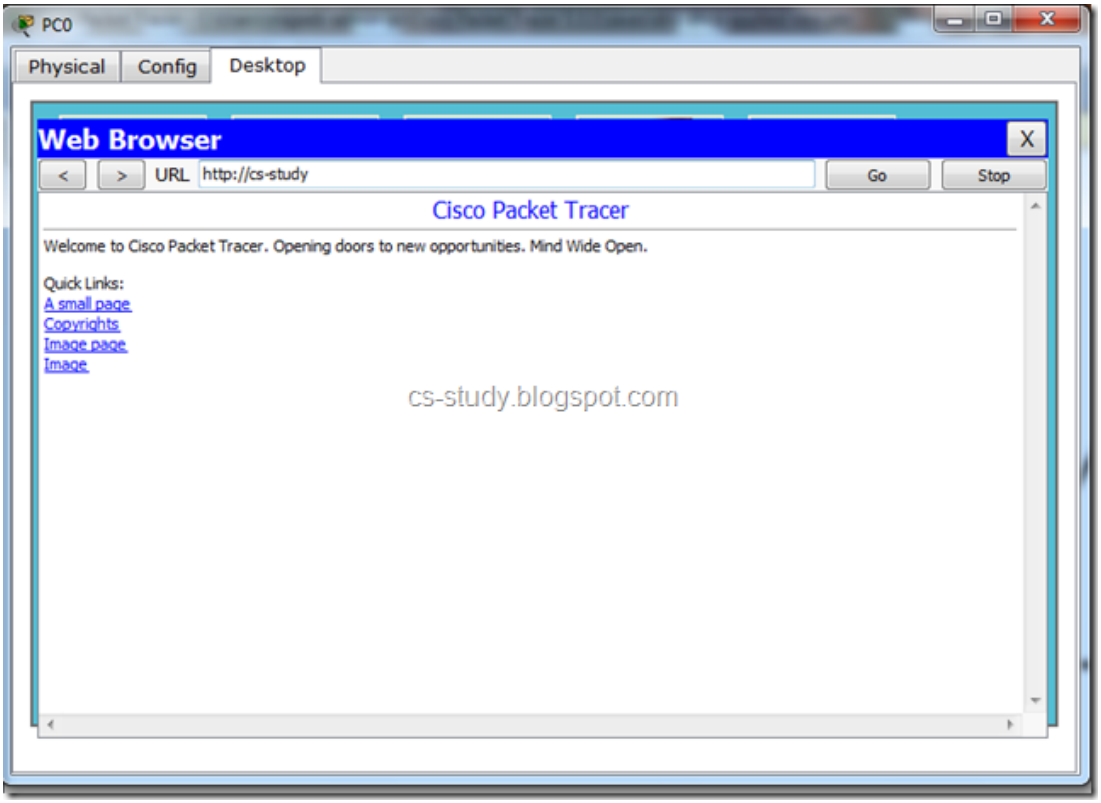
Go to Web Browser and enter Server 0 ip address. You can access the website of the server.



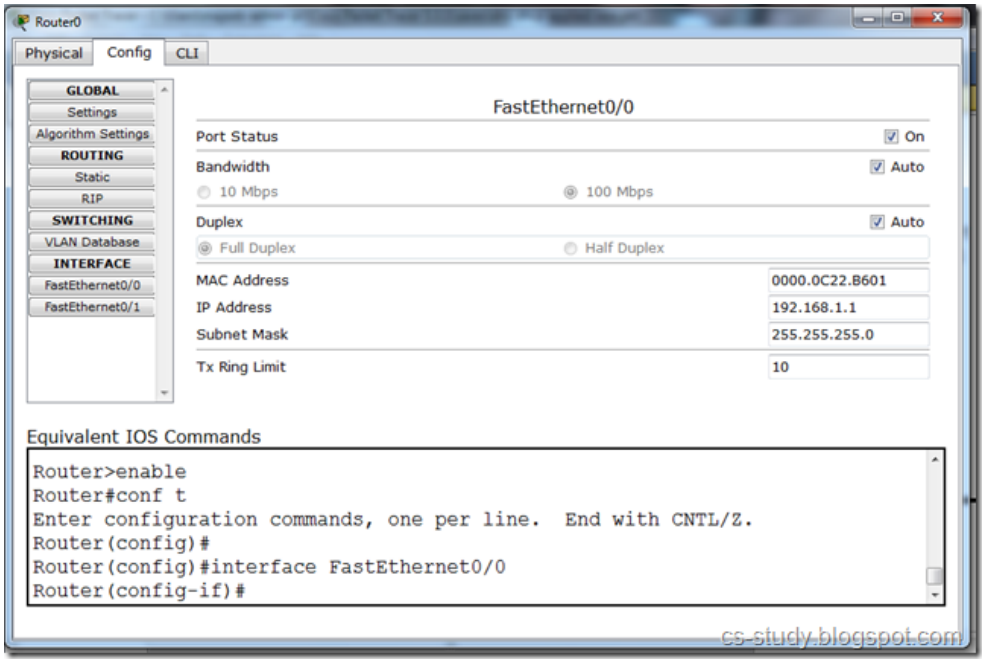
Now, let us set DNS on server 1.



Now, again go to PC and in the web browser enter the name that you set in DNS.



Voila, we have done it. Now, in this tutorial, router is additional and we can use it if required, Set the IP of the interface. Though i do not recommend to use this GUI panel for this.



Lab 8 – VLAN on Packet Tracer

Objectives

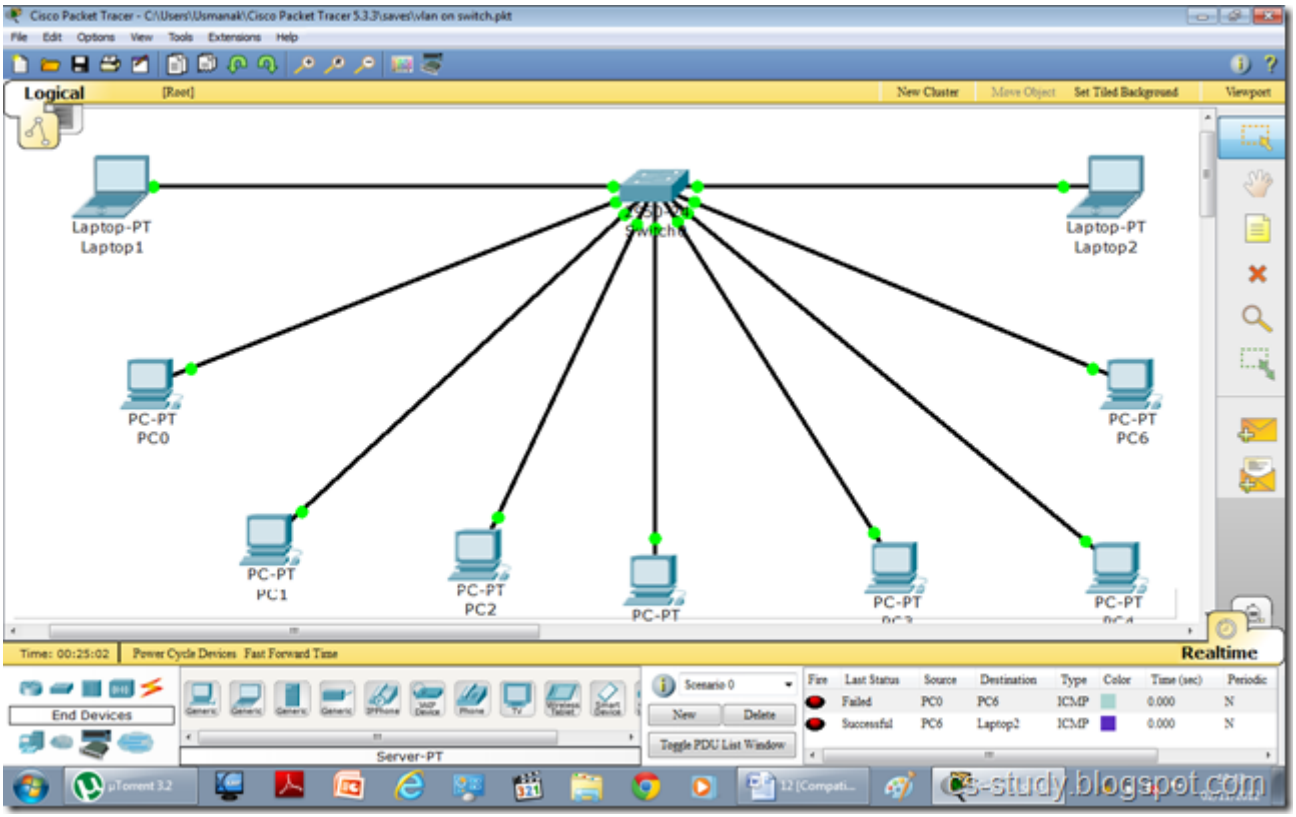
In this lab, students will

- Apply VLANS on packet tracer.
- Assign them to different ports.
- Assign them different names.
- Validate them by showing they aren't accessible from other vlans.

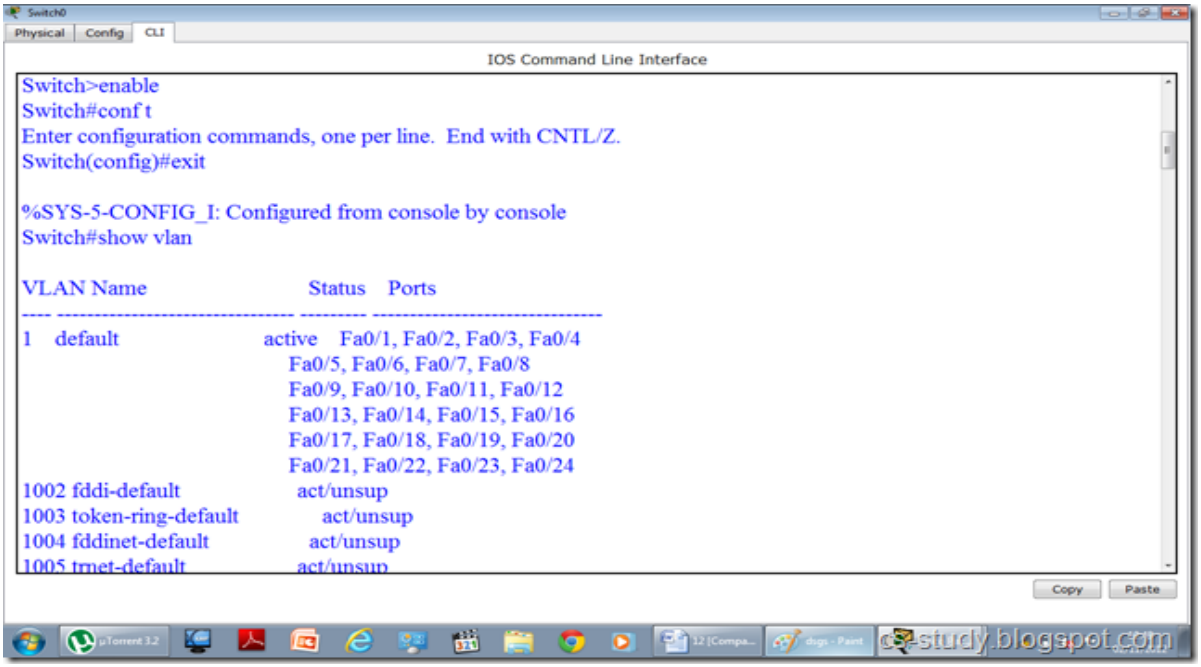
Lab Tasks

Apply VLANS on packet tracer.

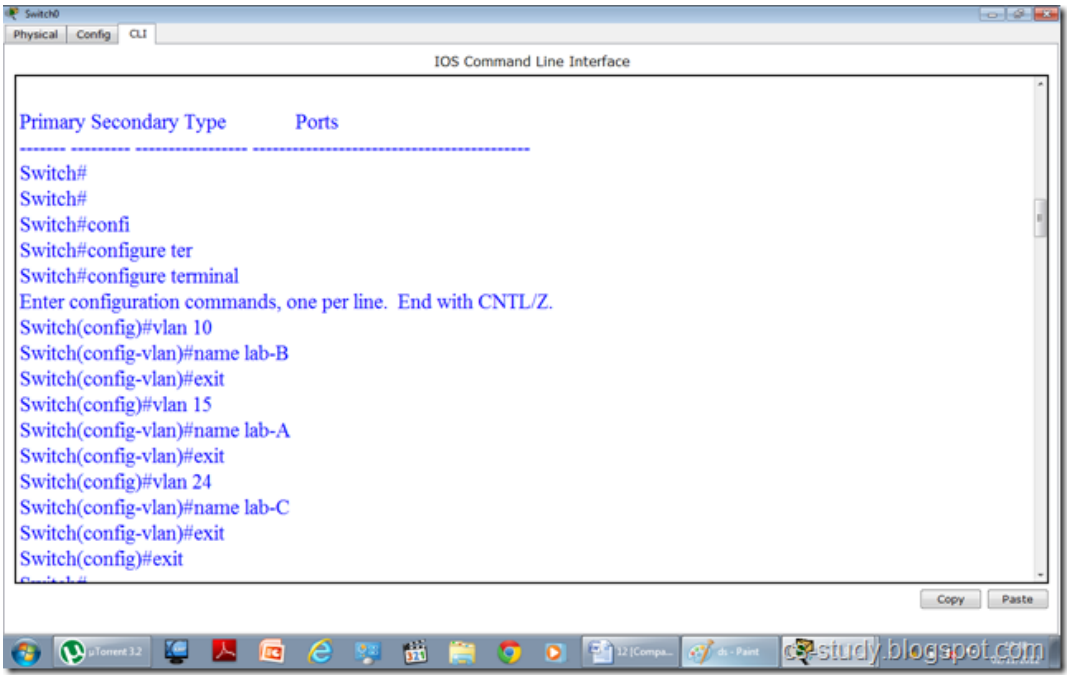
In this tutorial, we are going to apply VLAN on packet tracer. Let us create a topology with one switch and multiple hosts like in the figure below.



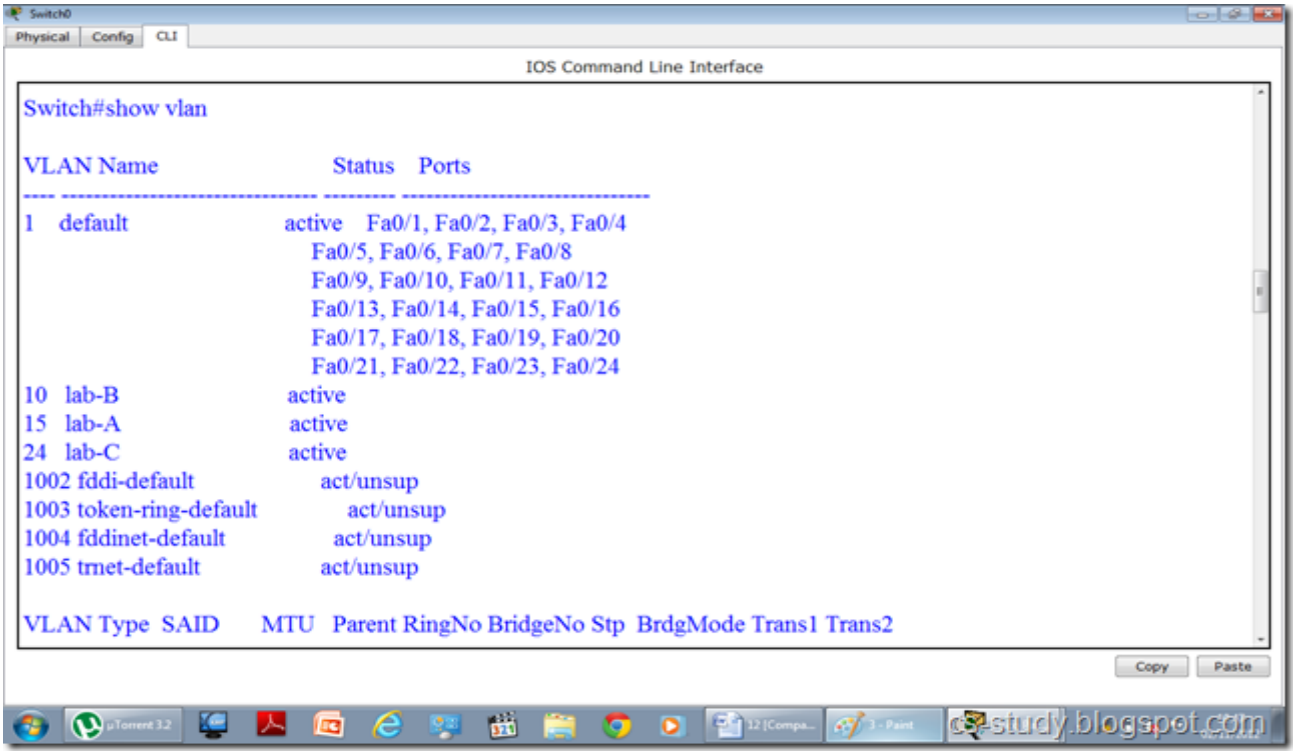
if we go to the switch and enter the command “show vlan ”. It shows the following.



As we can see in the figure above all the interfaces are being displayed and they are all the part of the default vlan 1. Now let us apply vlans on the switch. We are going to create three vlans as follows.



Now, that we have created the vlans. Lets see if they are visible to us.

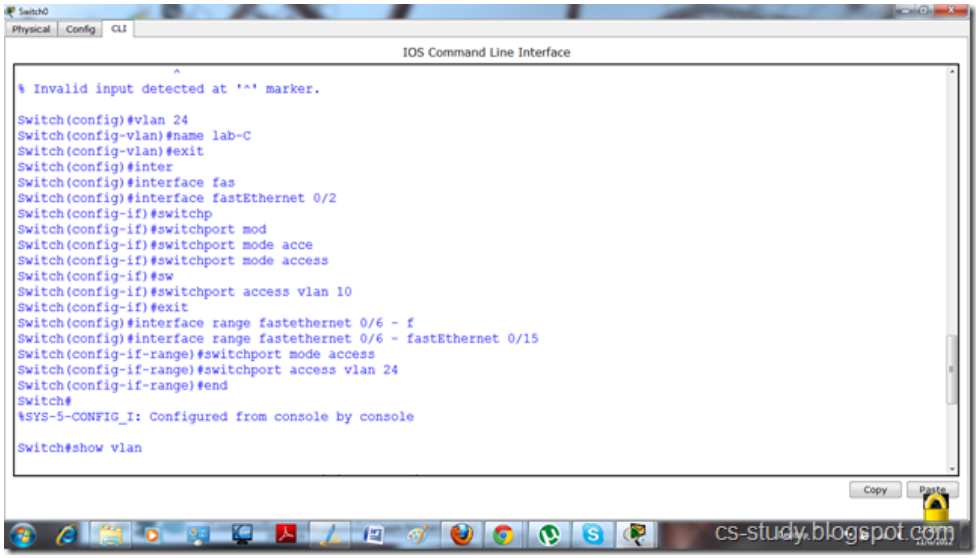


In the above figure, vlans are visible. Now, we are going to assign interfaces to vlans. They are two ways to do this.

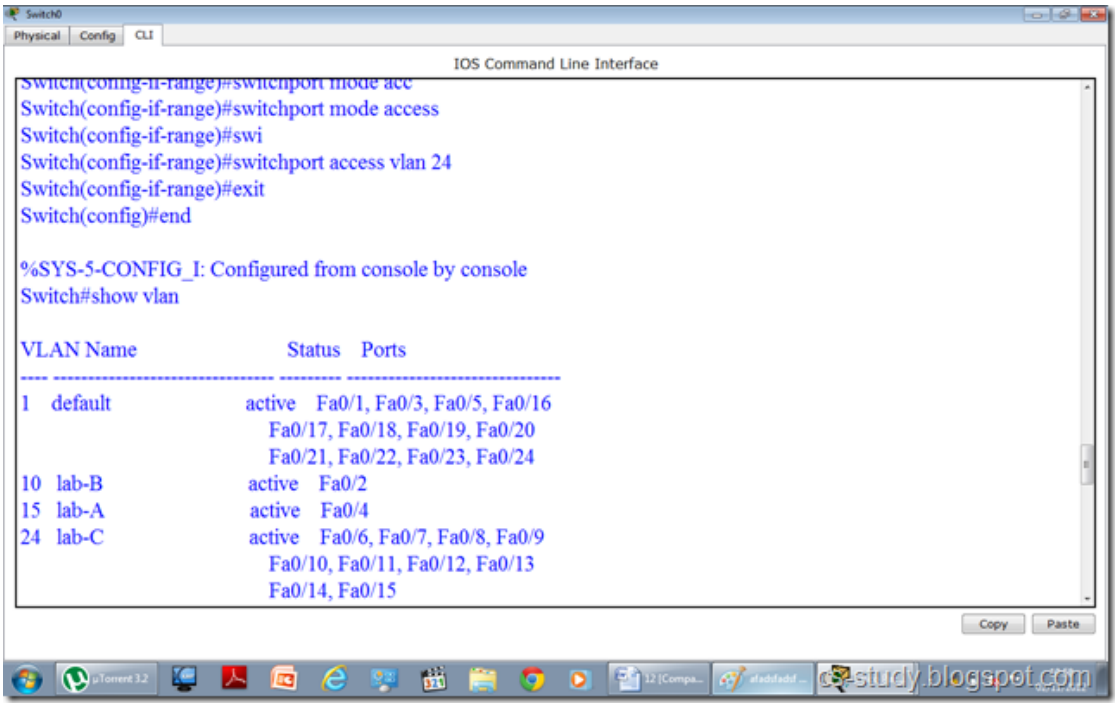
i. We can select an interface and assign that interface to a specific vlan

ii. We can select multiple interfaces (range of interfaces) at once and assign those interfaces to vlan.

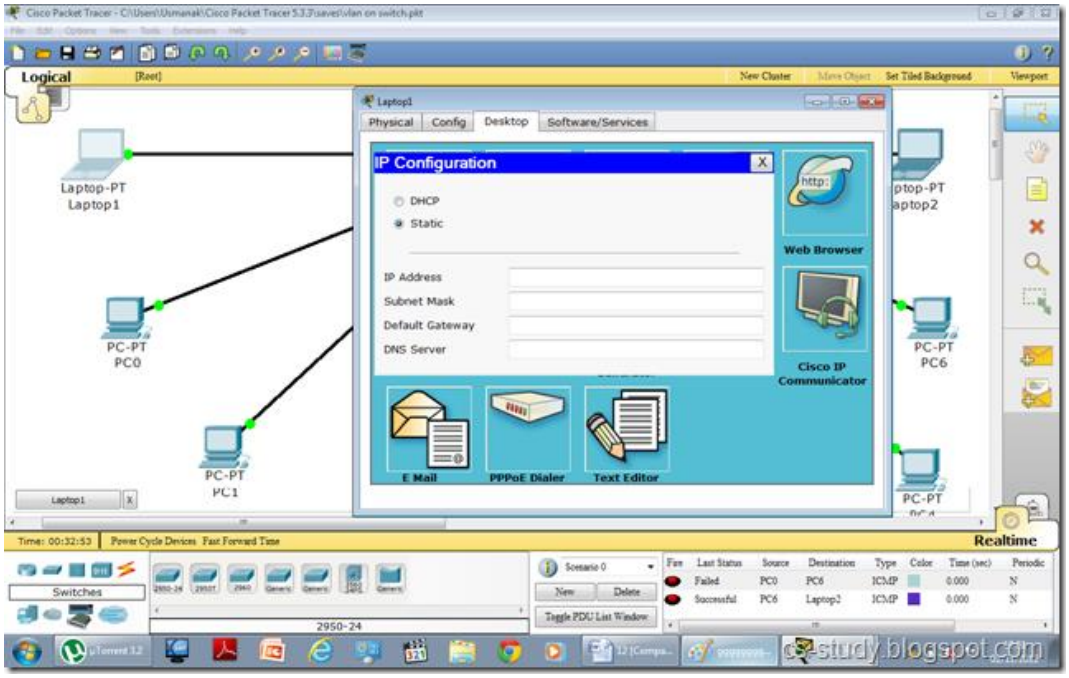
In the figure below, we have done both of these.



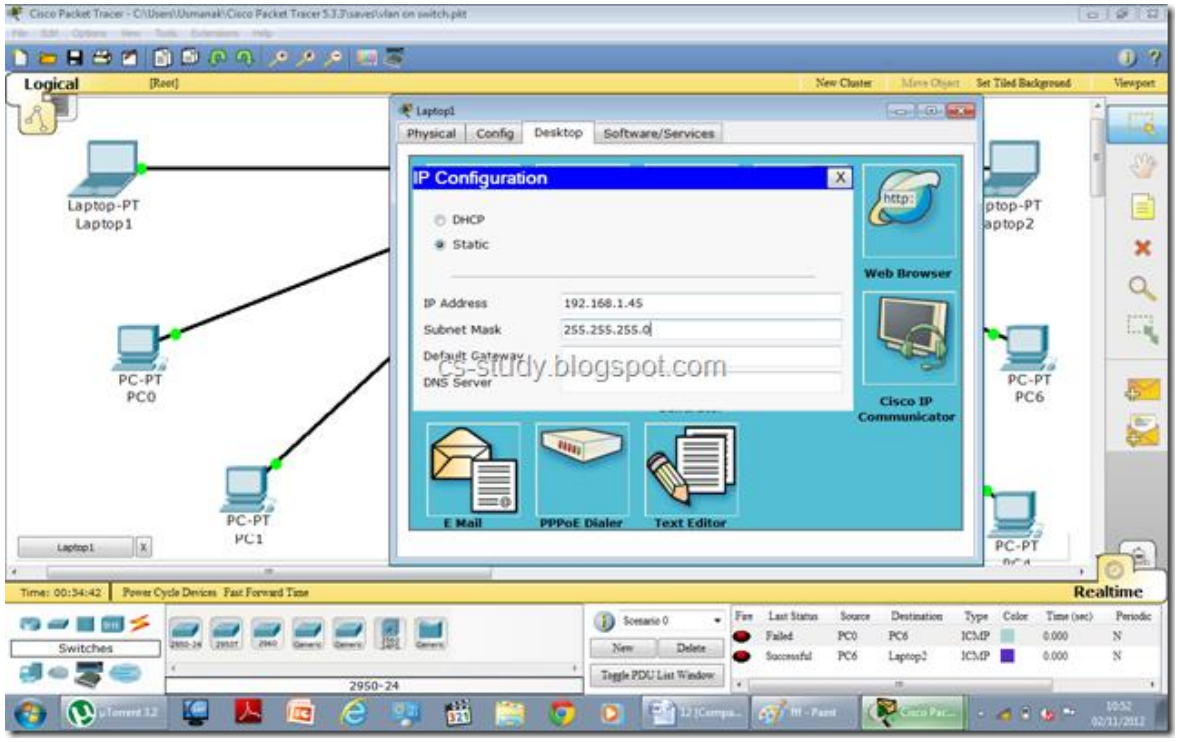
Now, when we write “show vlan “ command and observe it. We will realize that interfaces have been assigned to desired vlans respectively.



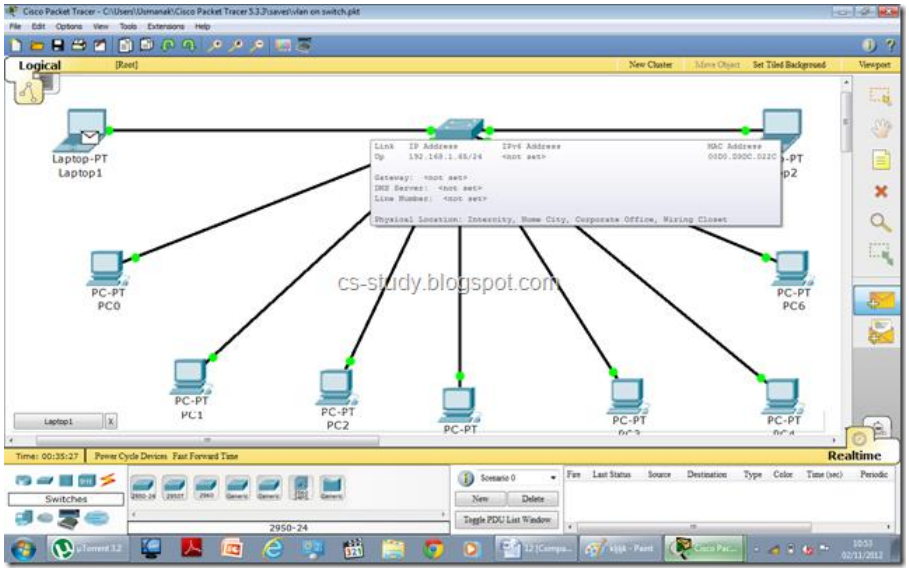
Let us assign IP addresses to PCs. Open the PC.



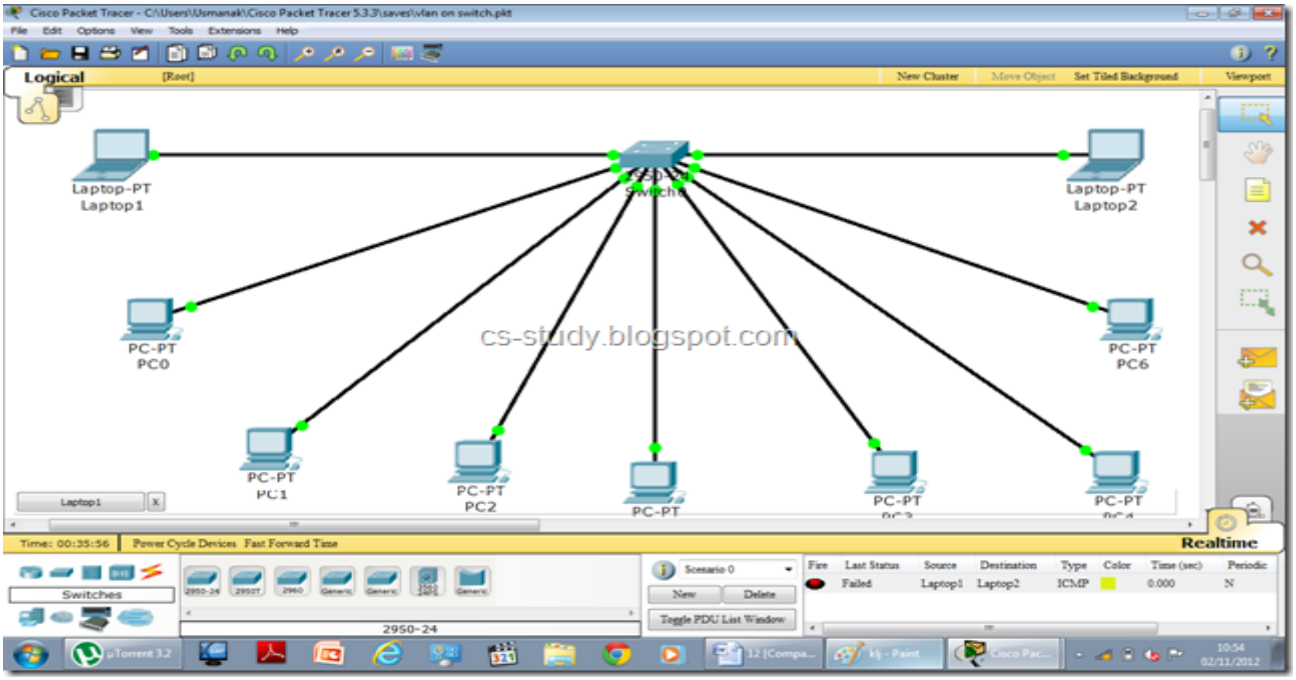
Assign IP address.



After assigning IP addresses, when we try to communicate between two PCs belonging to two different vlans, it will fail. Thus, we have achieved our purpose.

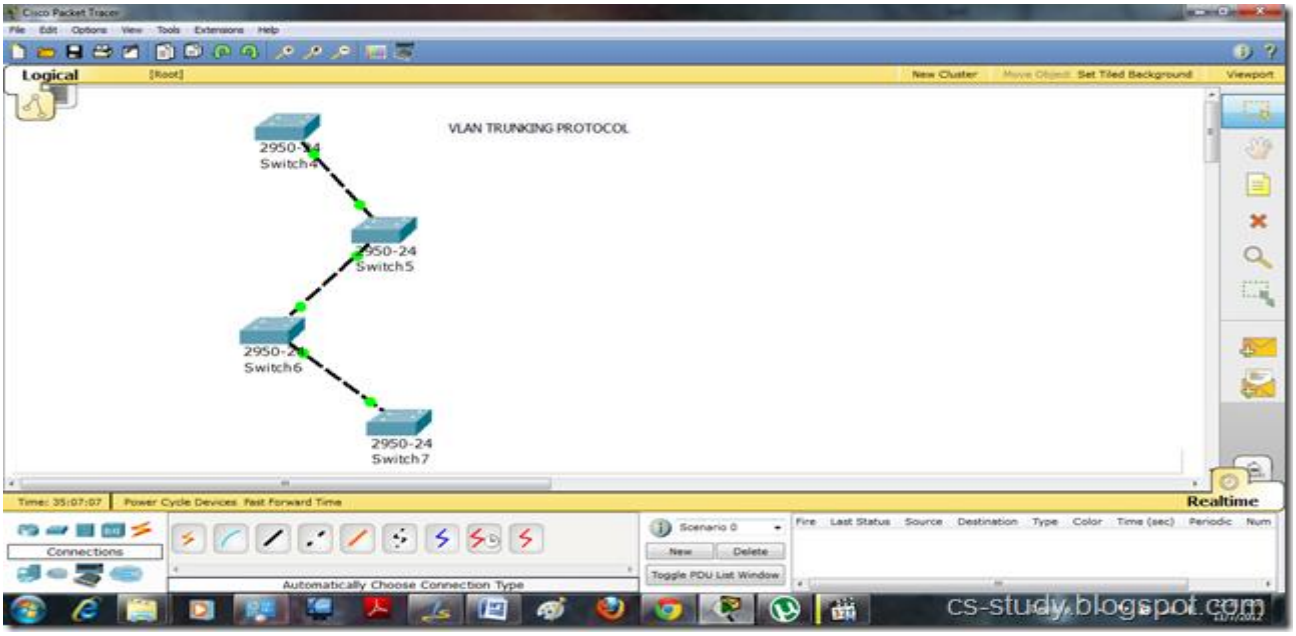


The message sending failure status can be seen in the bottom right corner.

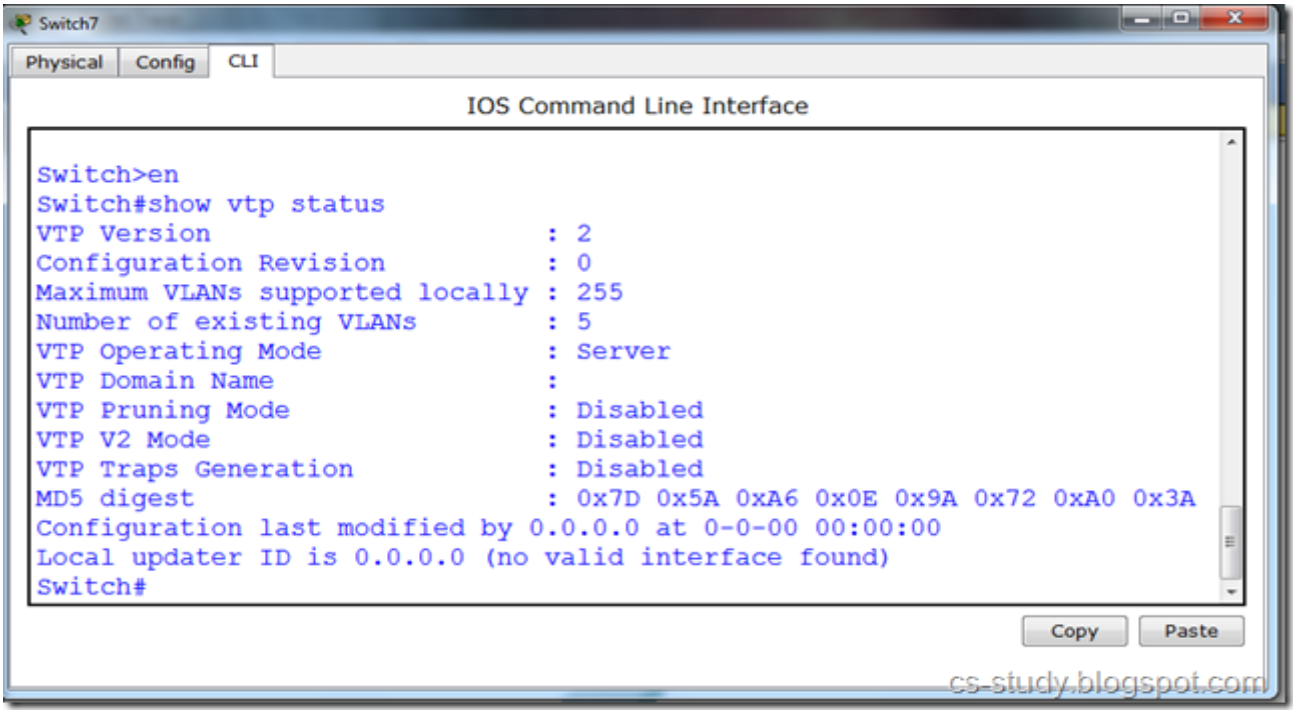


Task 2: VTP on packet tracer

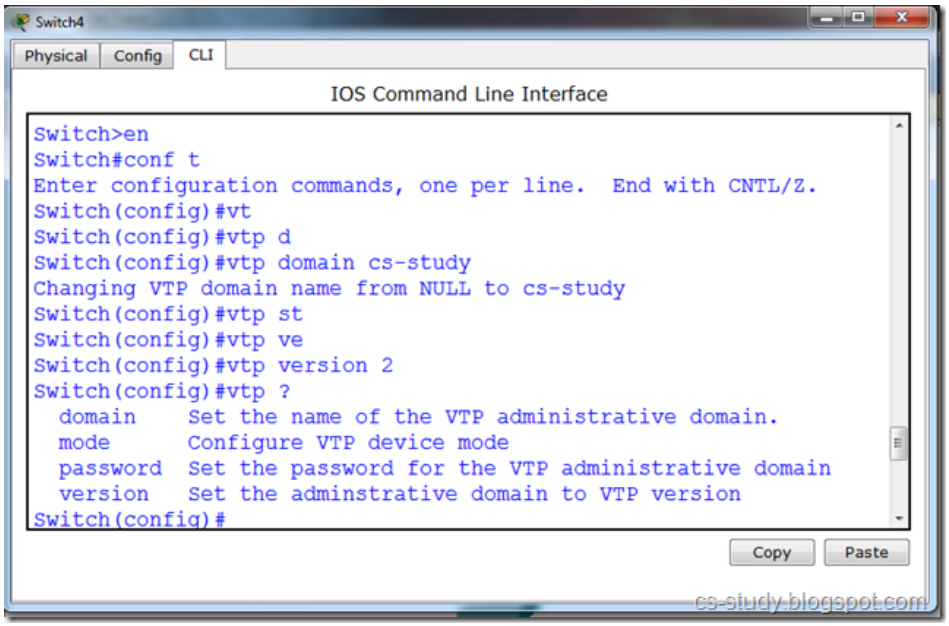
VLAN Trunk Protocol (VTP) reduces administration in a switched network. When you configure a new VLAN on one VTP server, the VLAN is distributed through all switches in the domain. This reduces the need to configure the same VLAN everywhere. Let us apply VTP on packet tracer.



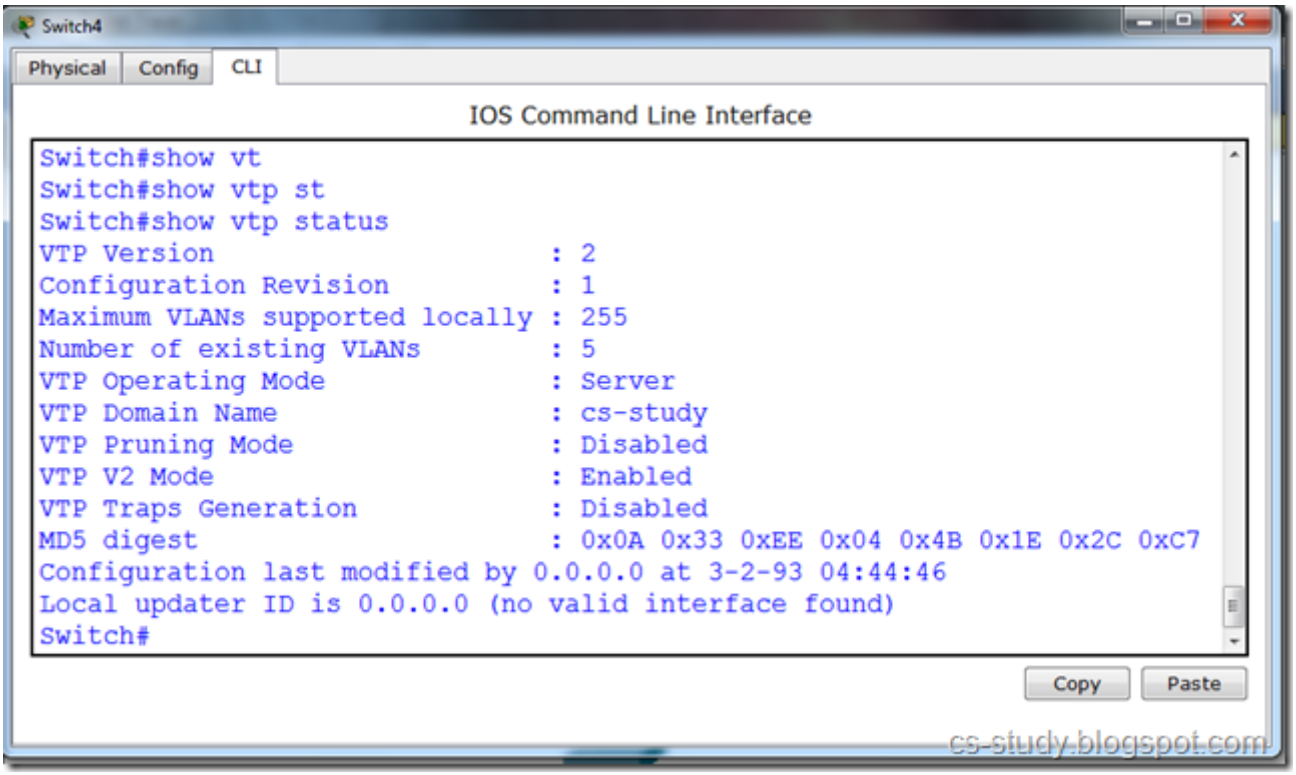
Let us see vtp status by applying the command “show vtp status”.



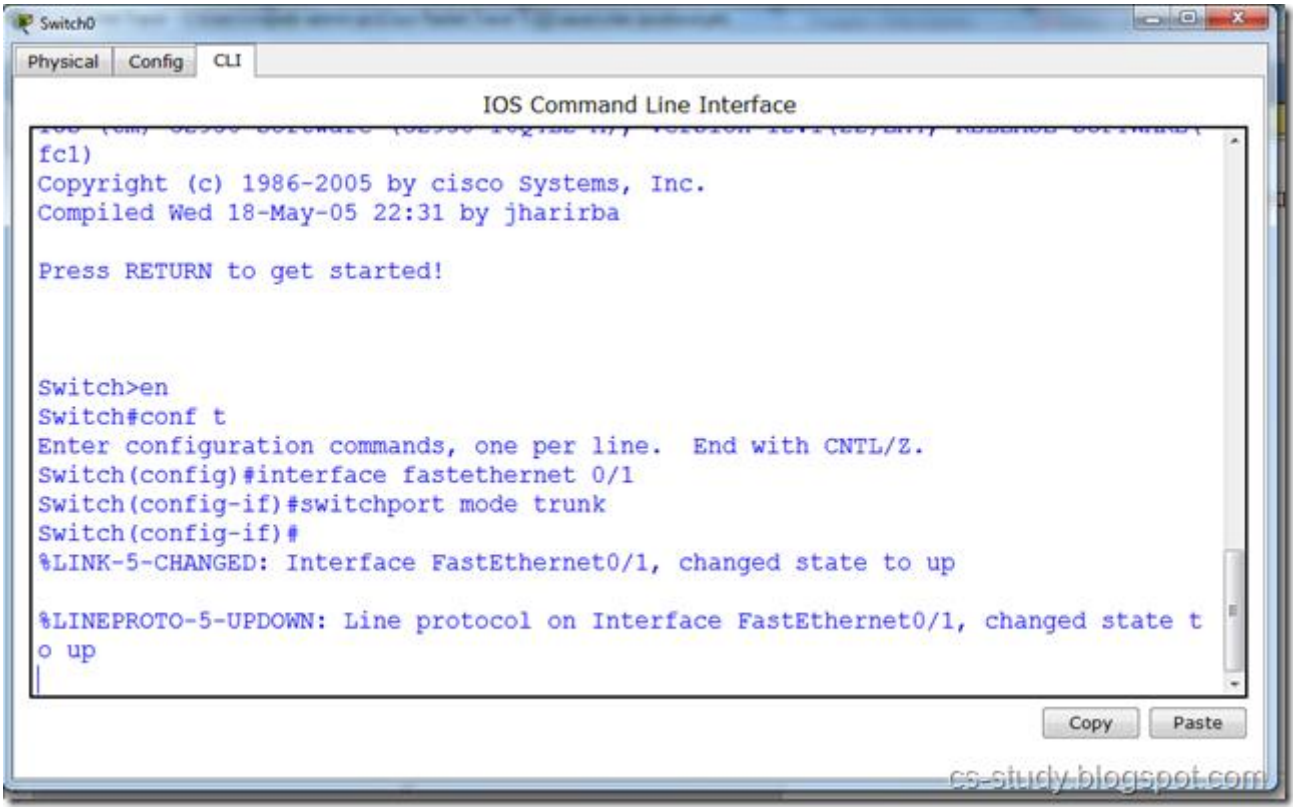
Let us set domain name. In VTP there should be only one domain name through out to synchronize between all the switches.



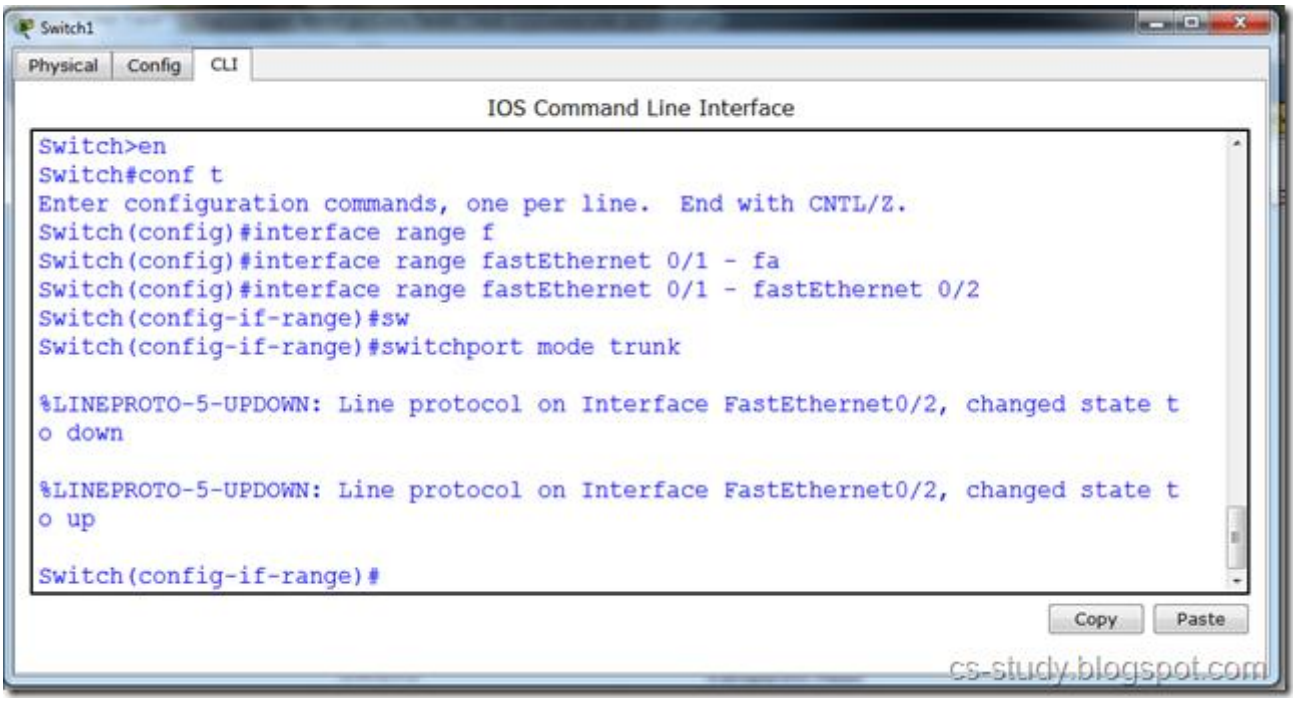
Domain name is set.



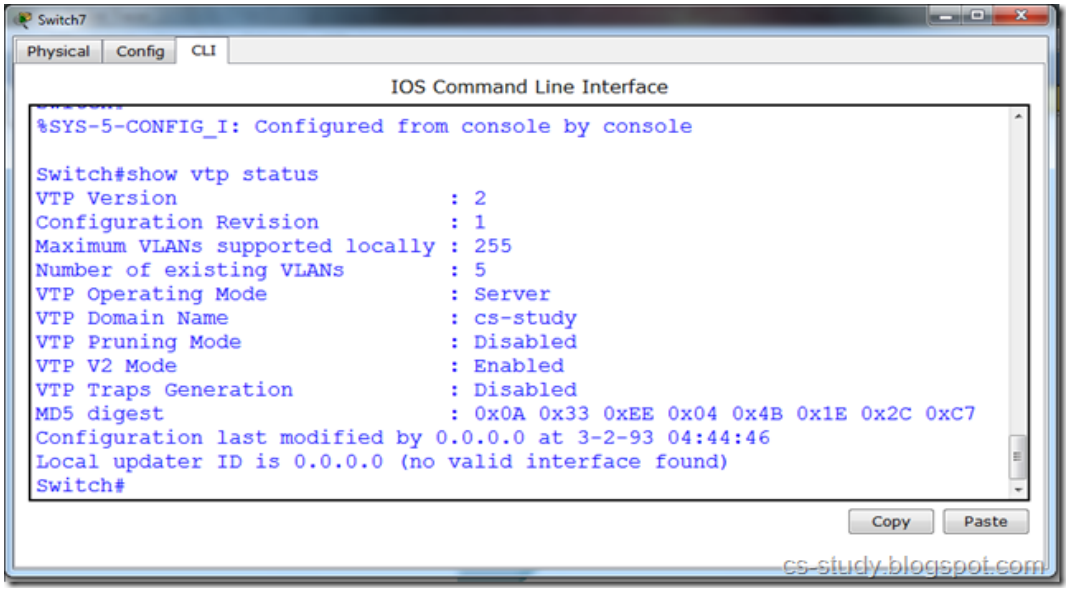
In order for changes made in one switch to take place in other switches as well. we will have to trunk the interfaces. Only those interfaces that are connected.



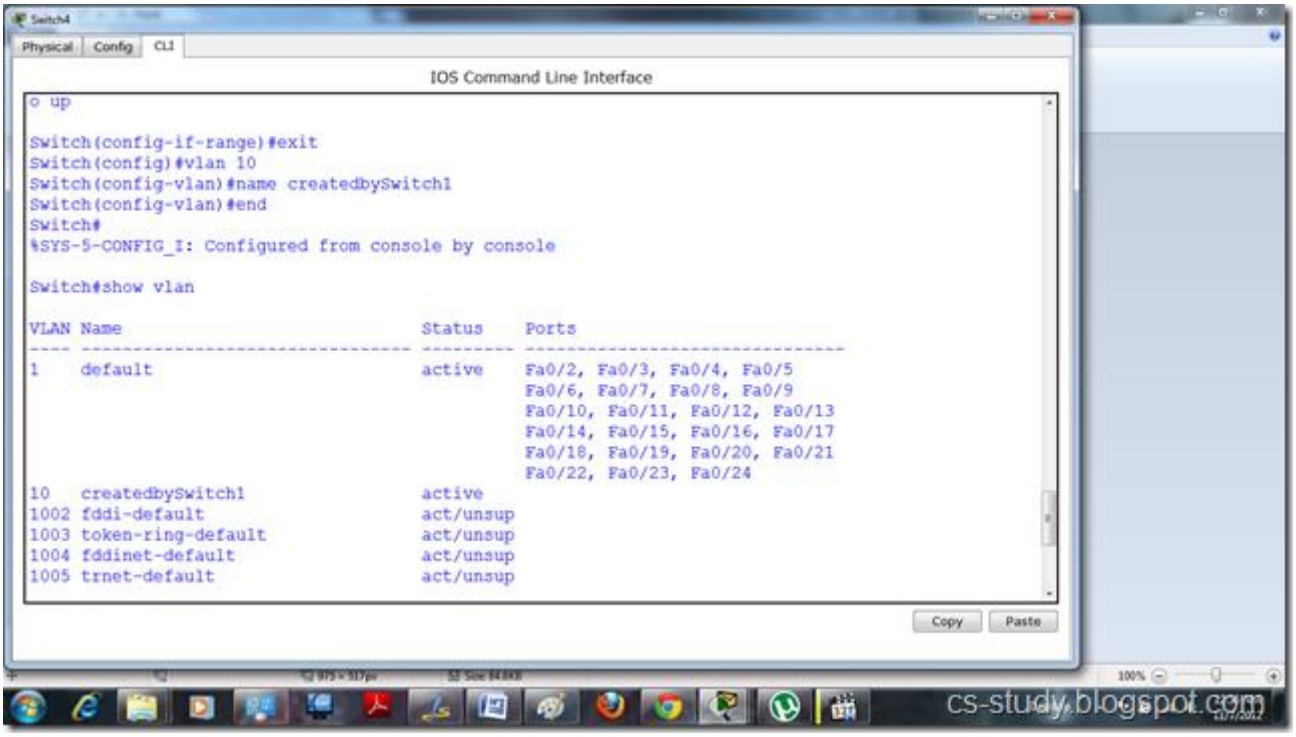
Or we can select a range of interfaces and trunk them.



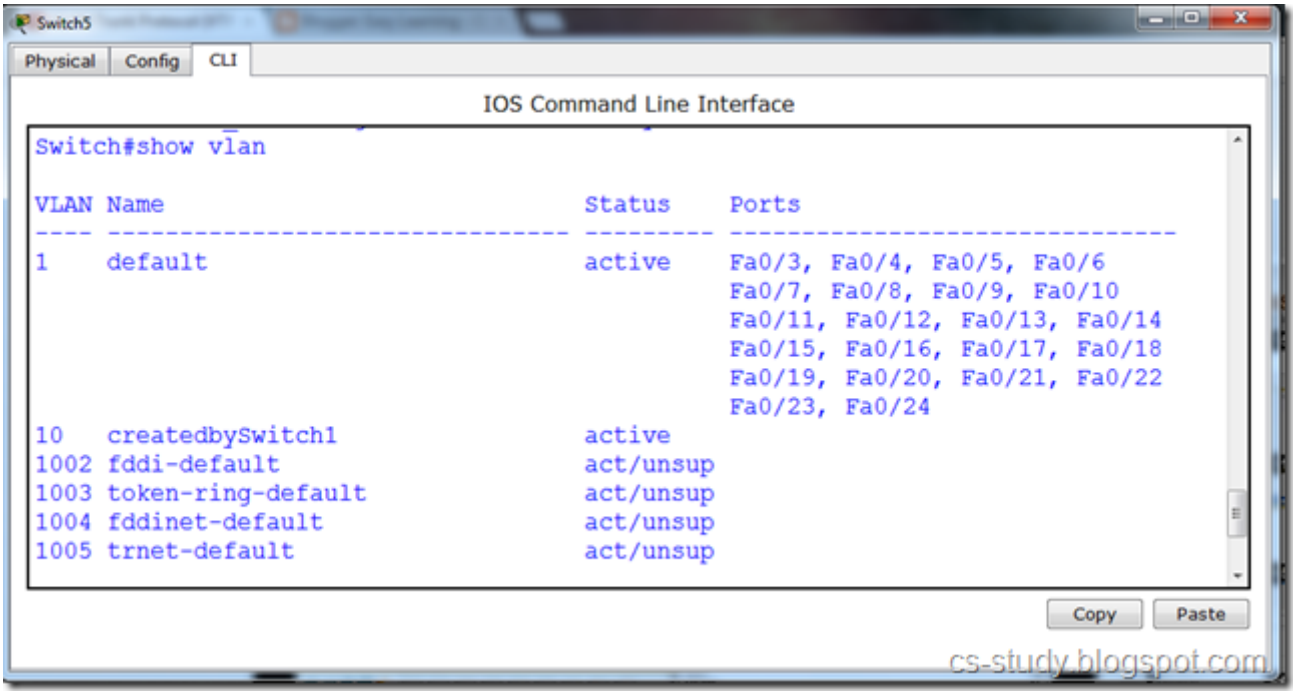
Now, when we check the status on other switches we can see that the domain name has been set on all the other switches as well.



Let us create VLAN.



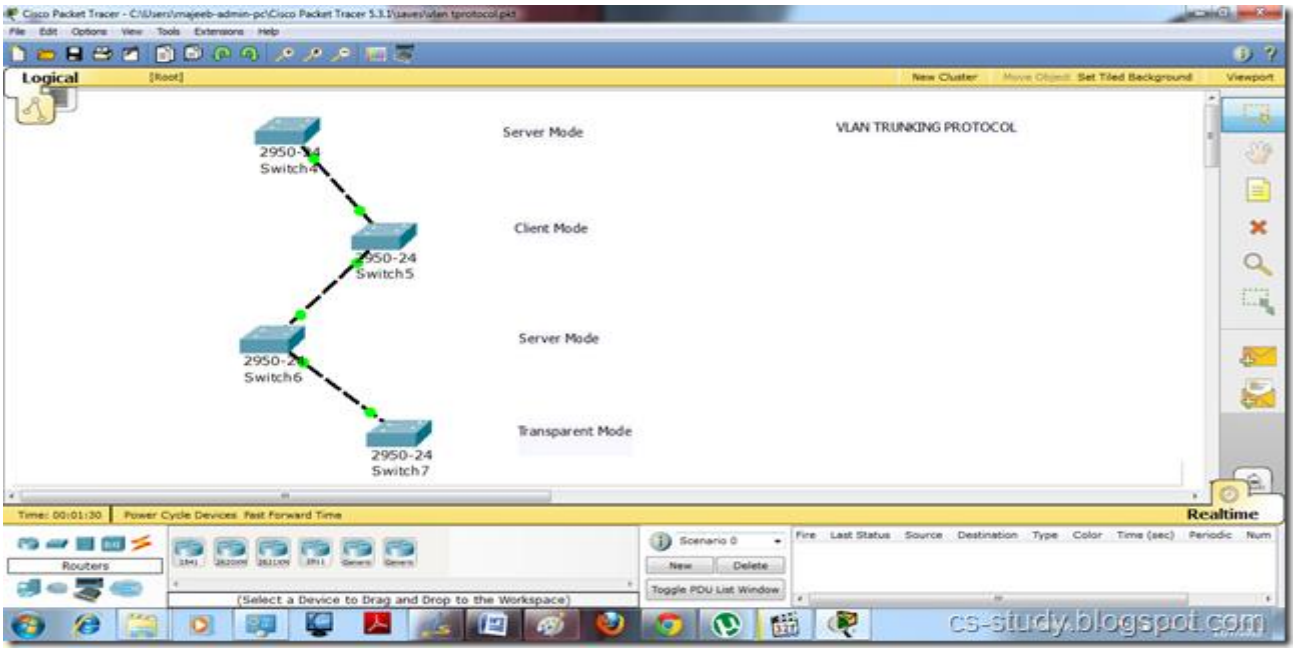
This vlan is shown in other switch due to the trunking .



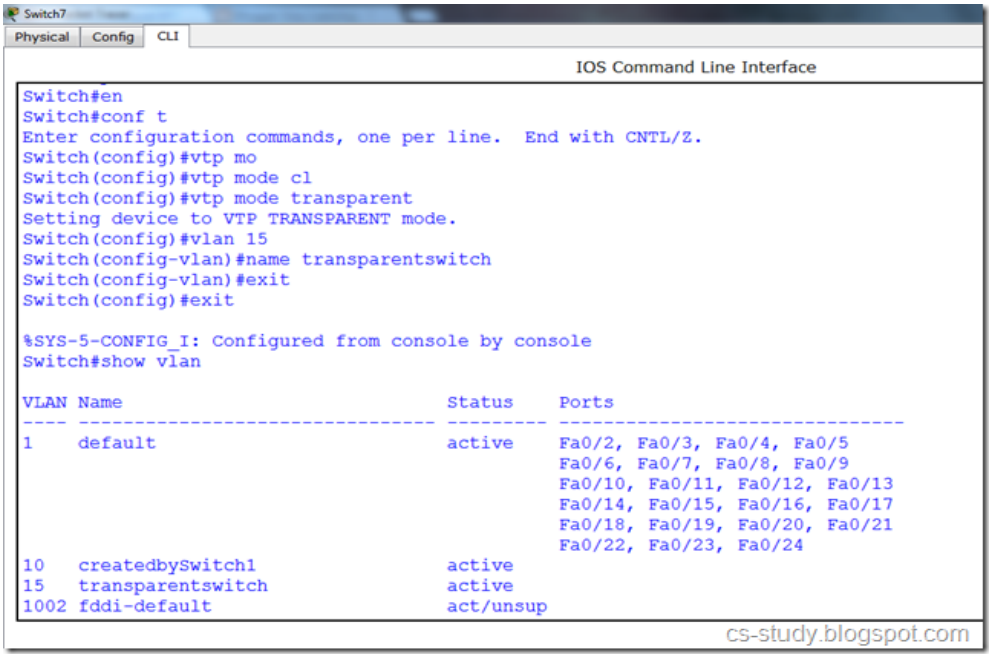
Now, there are three modes in a vtp.

i. Server
ii. Client
iii. Transparent

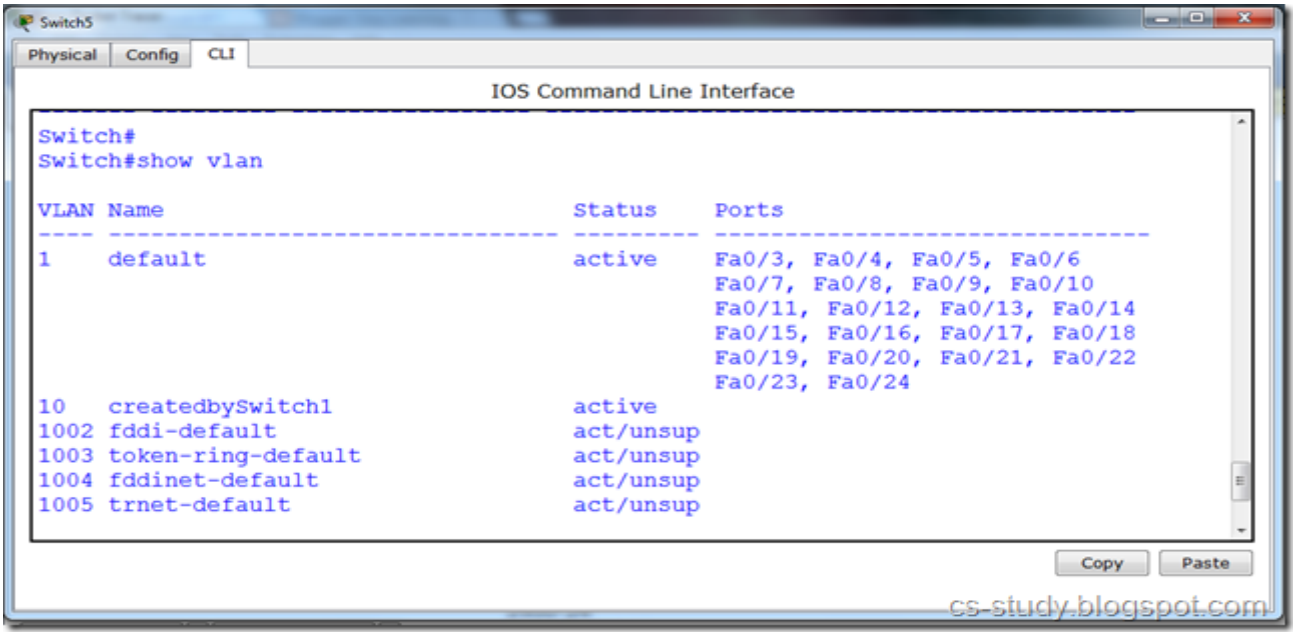
We are going to apply all three modes on different switches. We can create vlan in server mode, only use them in client mode. But the changes made in transparent mode are independent and does not have affect on other modes.



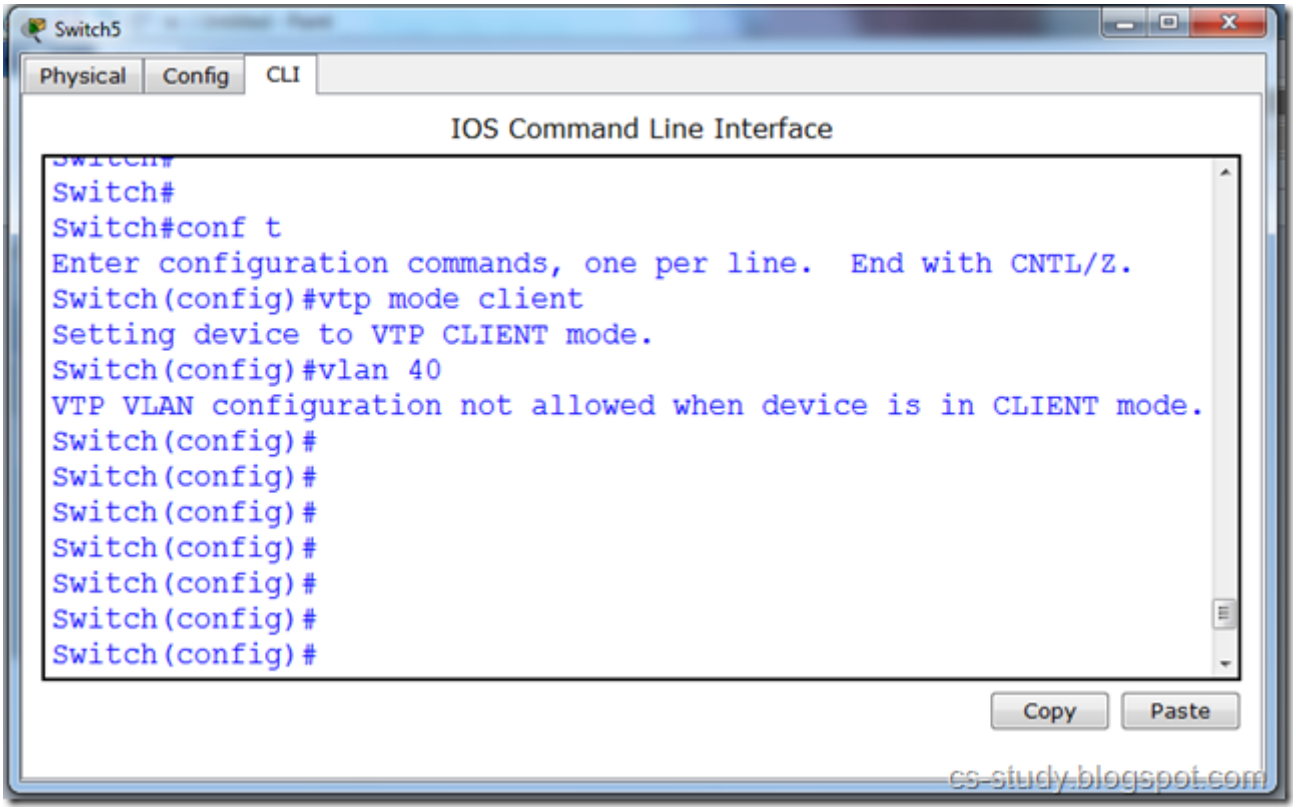
Let us turn the switch 7 to transparent mode and create a vlan in it.



The vlan created in transparent mode is not visible in the other modes.



If we change the mode of vtp from server to client , we are unable to create vlan now in the client mode as shown in the message below.



Lab 9 – STP and Sticky MAC addresses

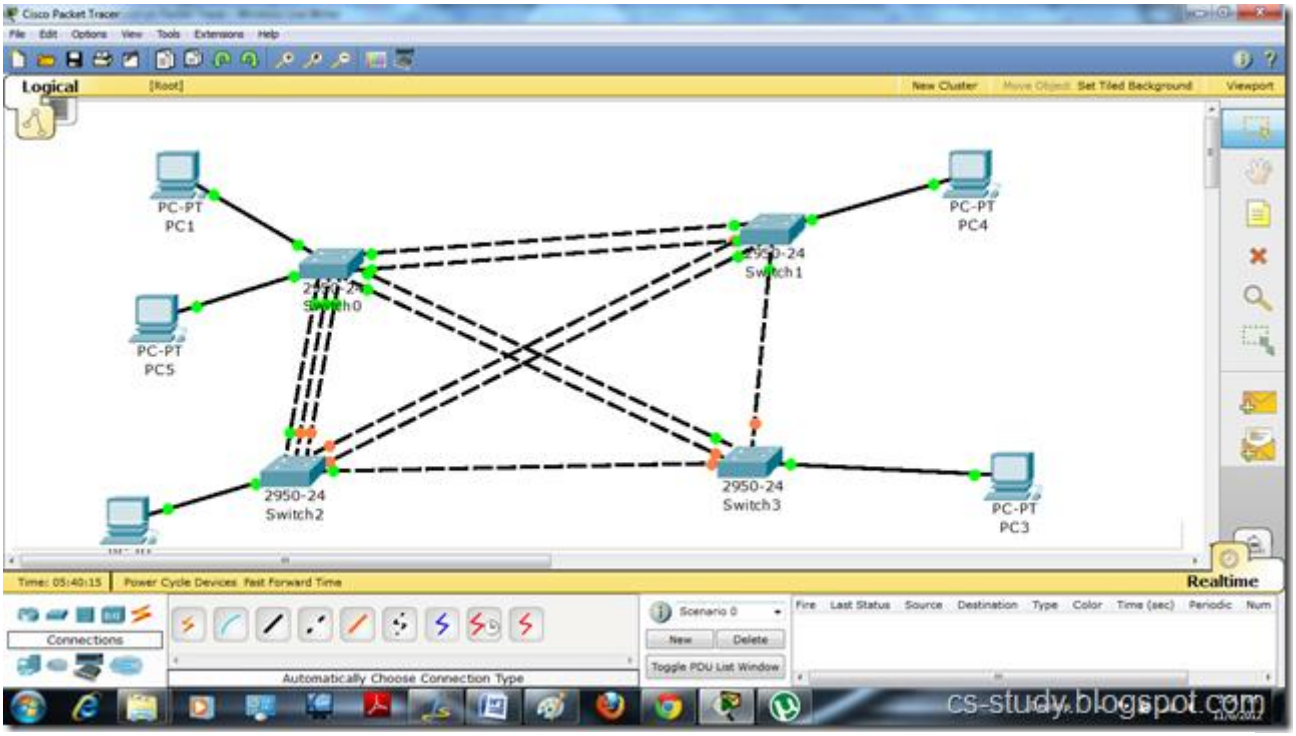
Objectives

In this lab, students will learn to implement spanning tree protocol, sticky MAC addresses etc.

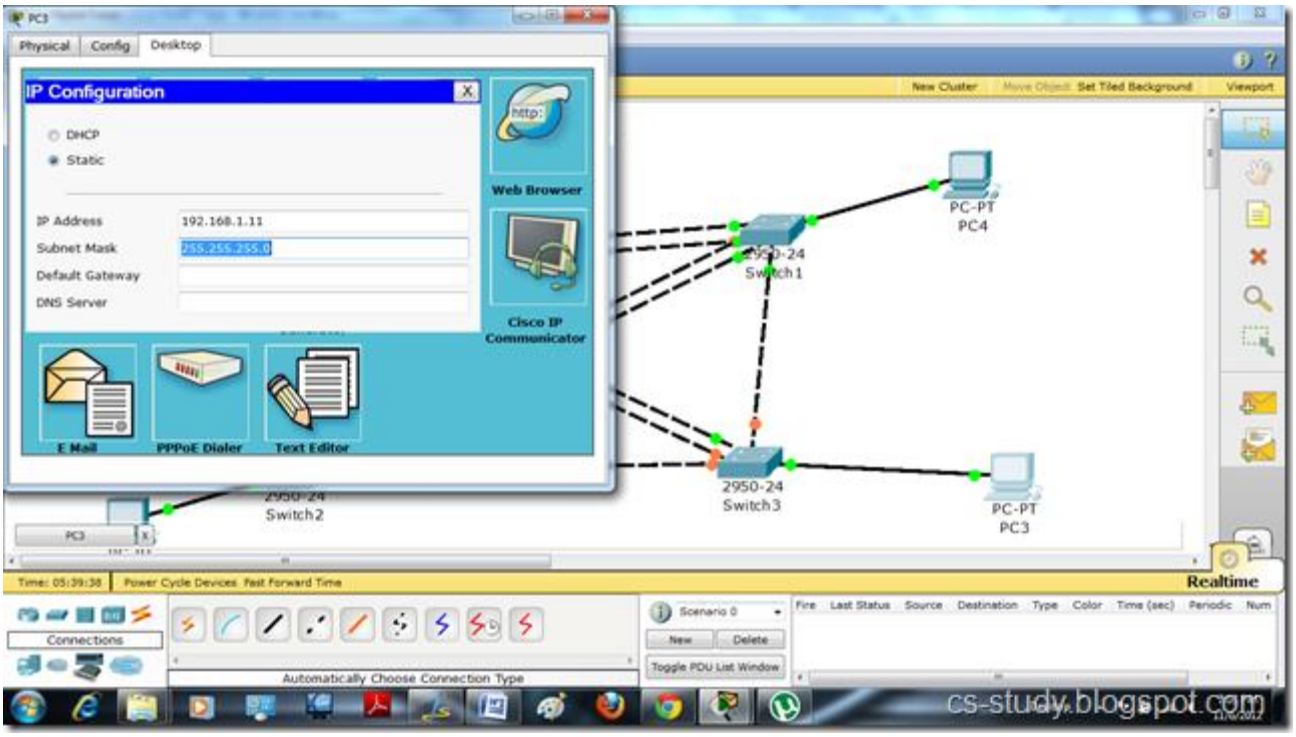
Lab Tasks

1. **Apply Spanning Tree Protocol on packet tracer.**

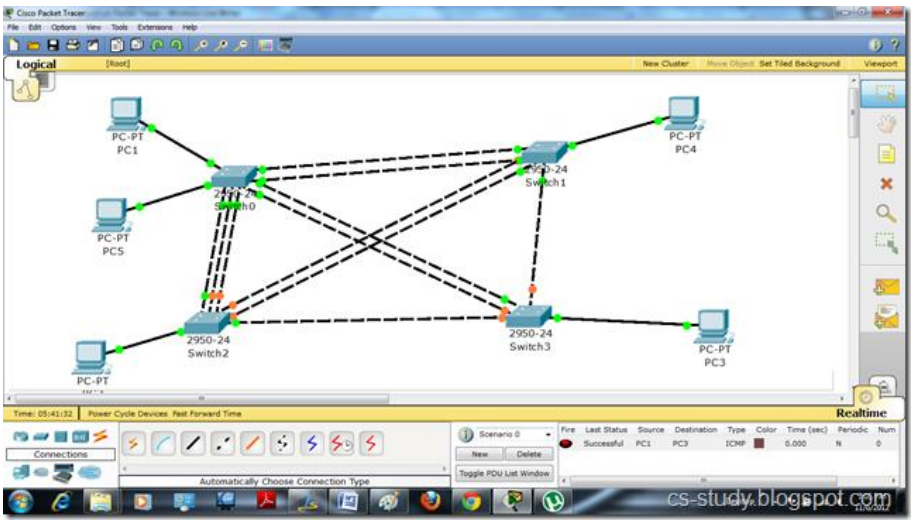
Let us apply STP on packet tracer. Let us develop a basic topology like the one in the following diagram.



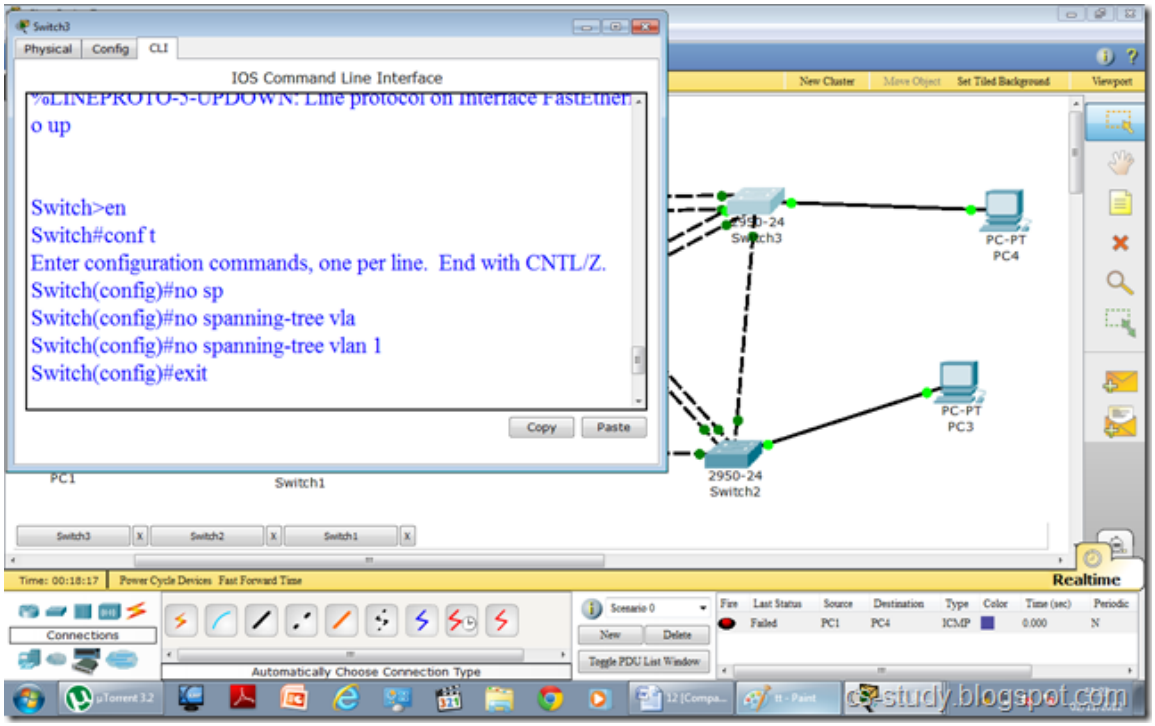
As we can see in the above diagram that some light are green while others are orange. Y is it so ? We will see that in a moment. Let us try to communicate between two Hosts. Assign IP addresses to all hosts



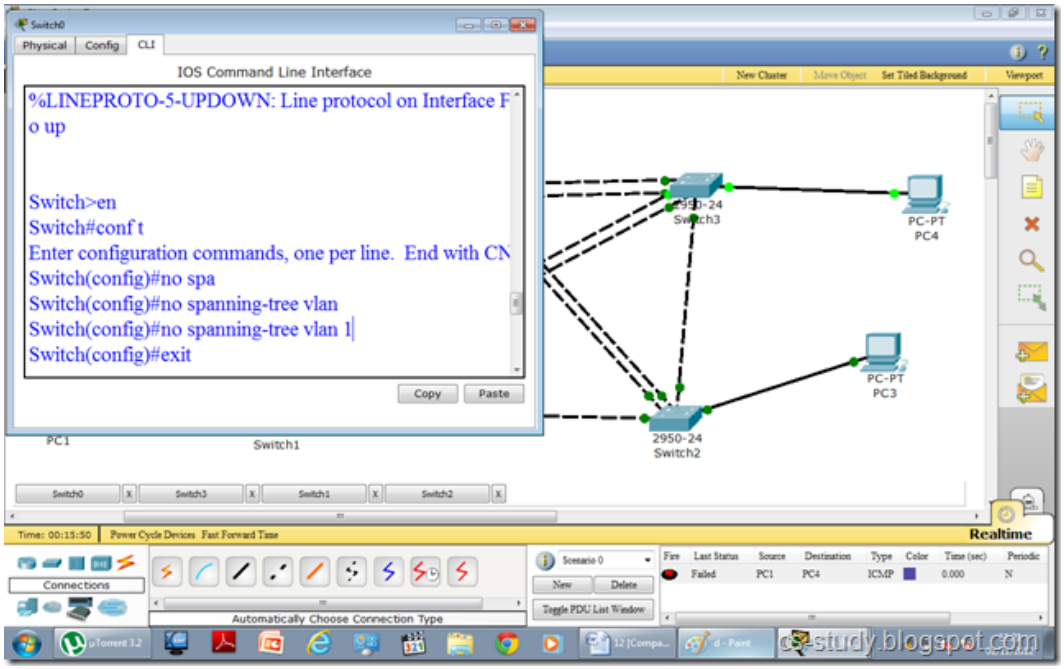
As we can see in the figure below, the communication is successful. It is due to the spanning tree protocol applied on the switch by default. It provides us with the loop free environment. It calculates the cost of each path and provides us with the one that has the minimum cost. That is the reason that some links are up while others are down with the orange light.

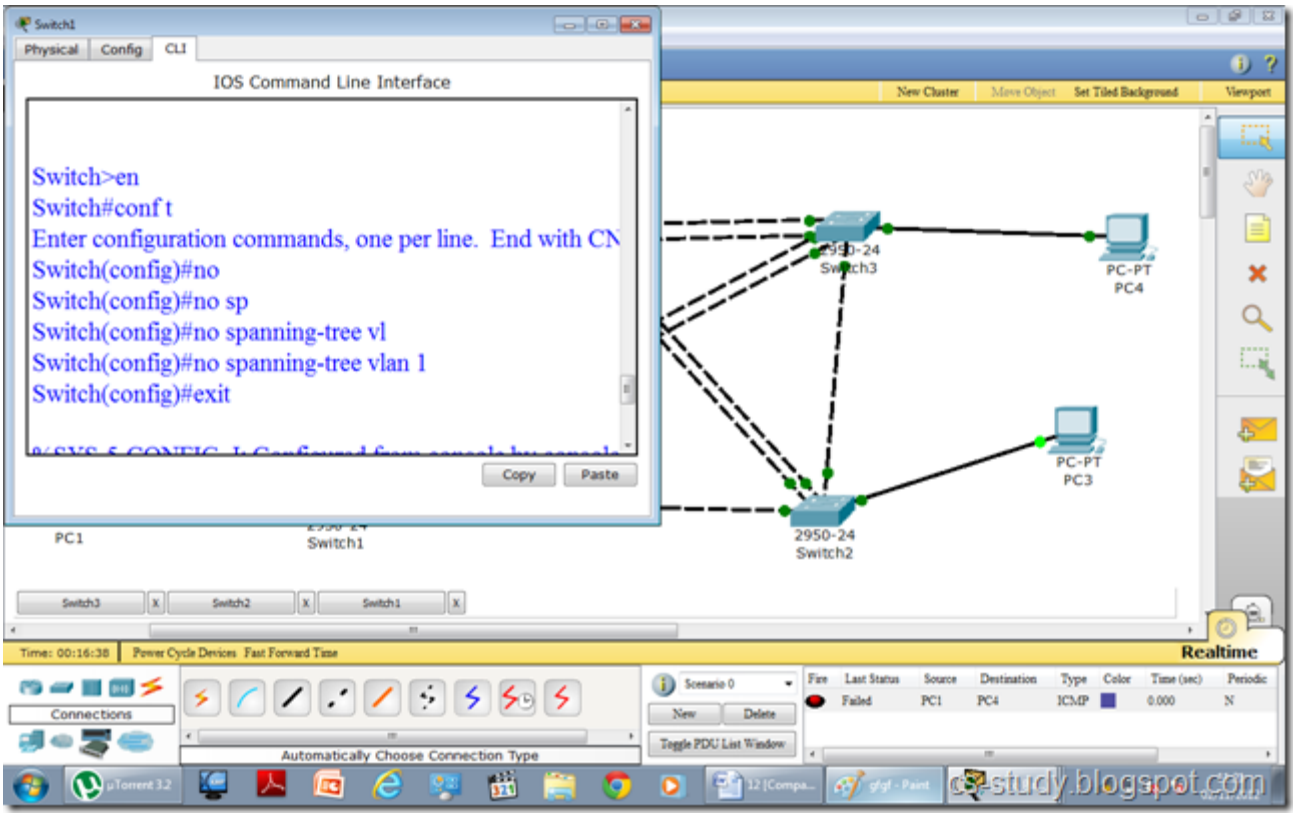


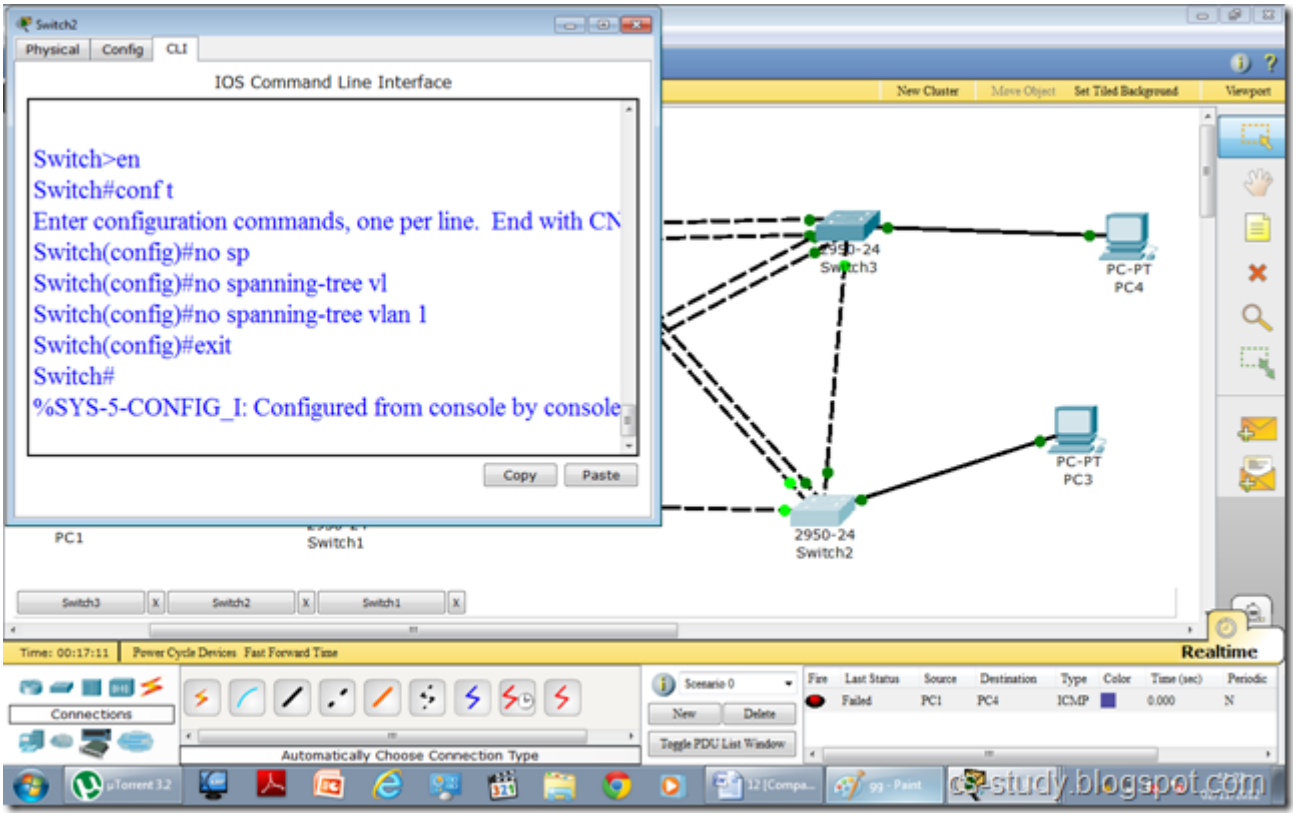
So let us see what happens if we remove the spanning tree protocol from this topology.



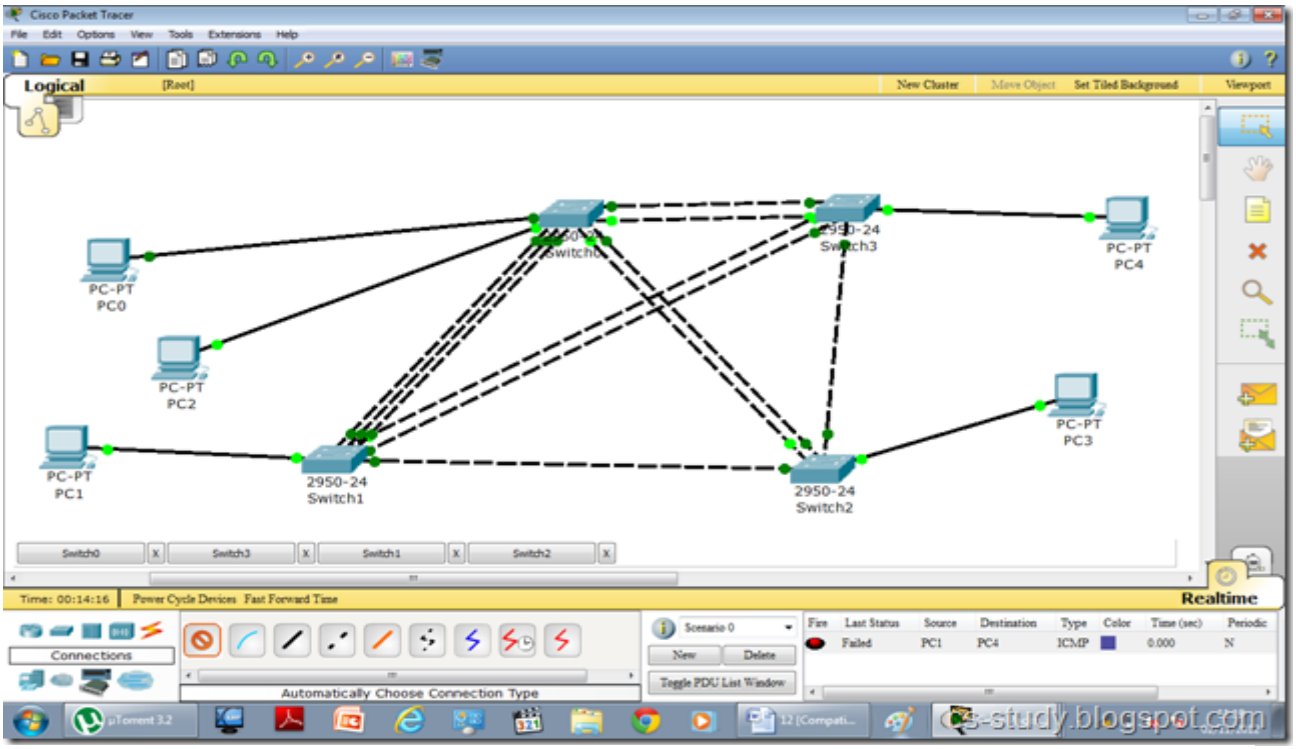
We will remove STP from all the switches one by one.



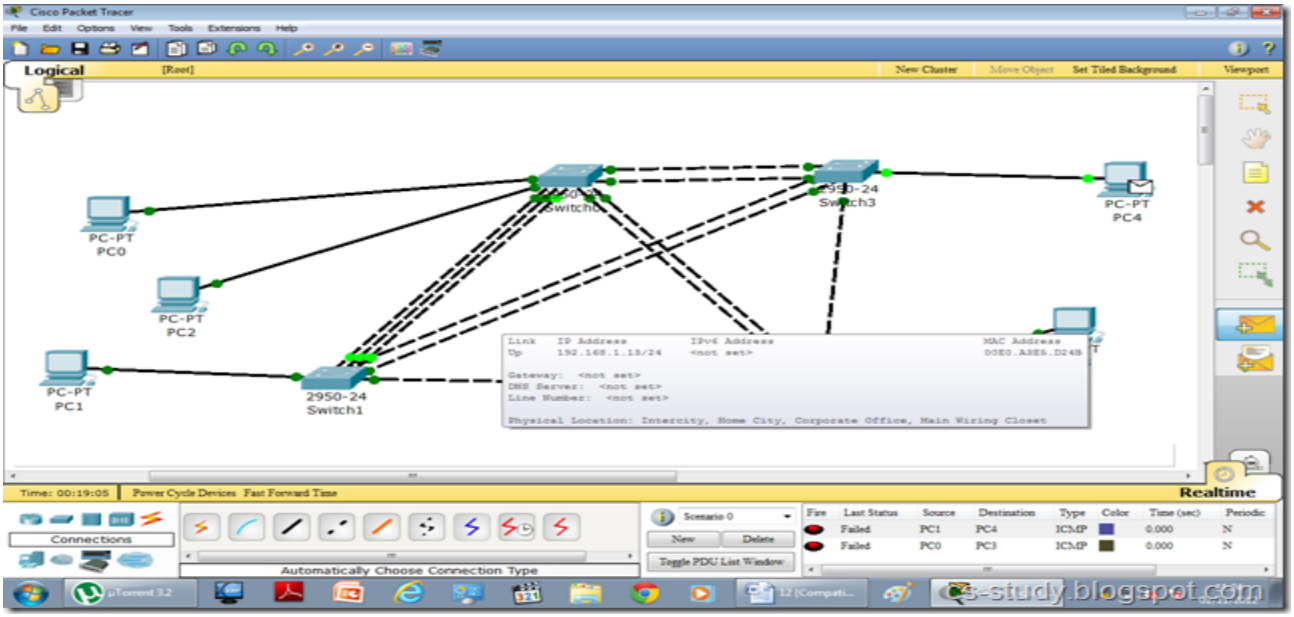

Switch1
Physical
Config
CLI
IOS Command Line Interface
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CN
Switch(config)#no
Switch(config)#no sp
Switch(config)#no spanning-tree vl
Switch(config)#no spanning-tree vlan 1
Switch(config)#exit
Copy
Paste
PC-PT
PC4
PC-PT
PC3
2950-24
Switch2
PC1
Switch1
Realtime
Connections
Automatically Choose Connection Type


Switch2
Physical
Config
CLI
IOS Command Line Interface
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CN
Switch(config)#no sp
Switch(config)#no spanning-tree vl
Switch(config)#no spanning-tree vlan 1
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console
Copy
Paste
PC-PT
PC4
PC-PT
PC3
2950-24
Switch2
PC1
Switch1
Realtime
Connections
Automatically Choose Connection Type

After removing STP, we have observed by the following diagram, a couple of changes. i.e. all the lights are green. In fact, some are dark green. Some lights are blinking, while some are not. This is due to fact that as there is no protocol to decide that which path to choose as we have removed STP.



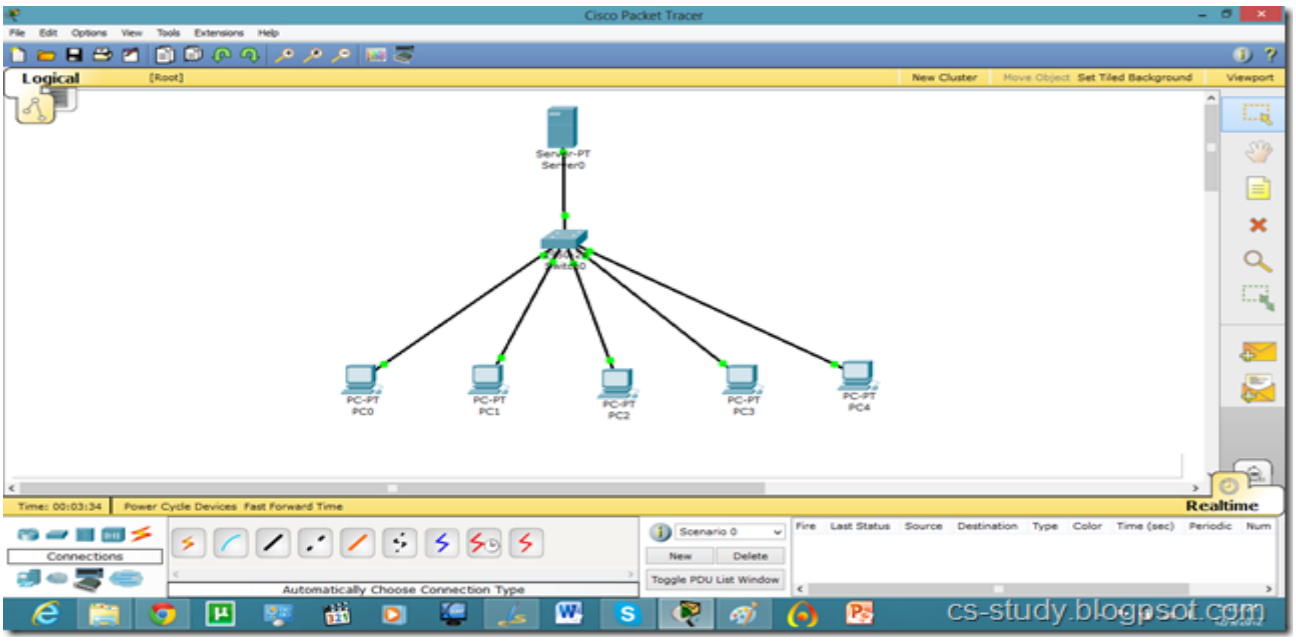
Now that if we try to communicate between any hosts it will fail and communication is disabled.



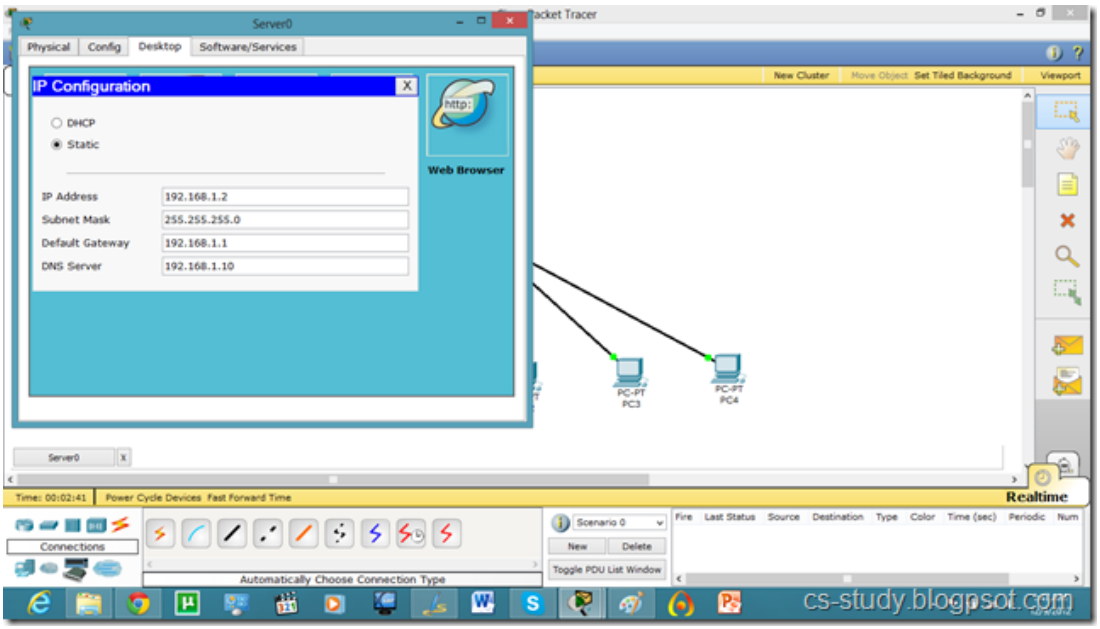
Task 2: Sticky MAC addresses

Sticky MAC addresses are addresses that are dynamically learned once and remain stick to the port, we can adjust the max number of sticky MAC addresses to a single interface. the use of this feature is in large networks usually where we cant afford to waste time doing manual mac addresses to port mapping.

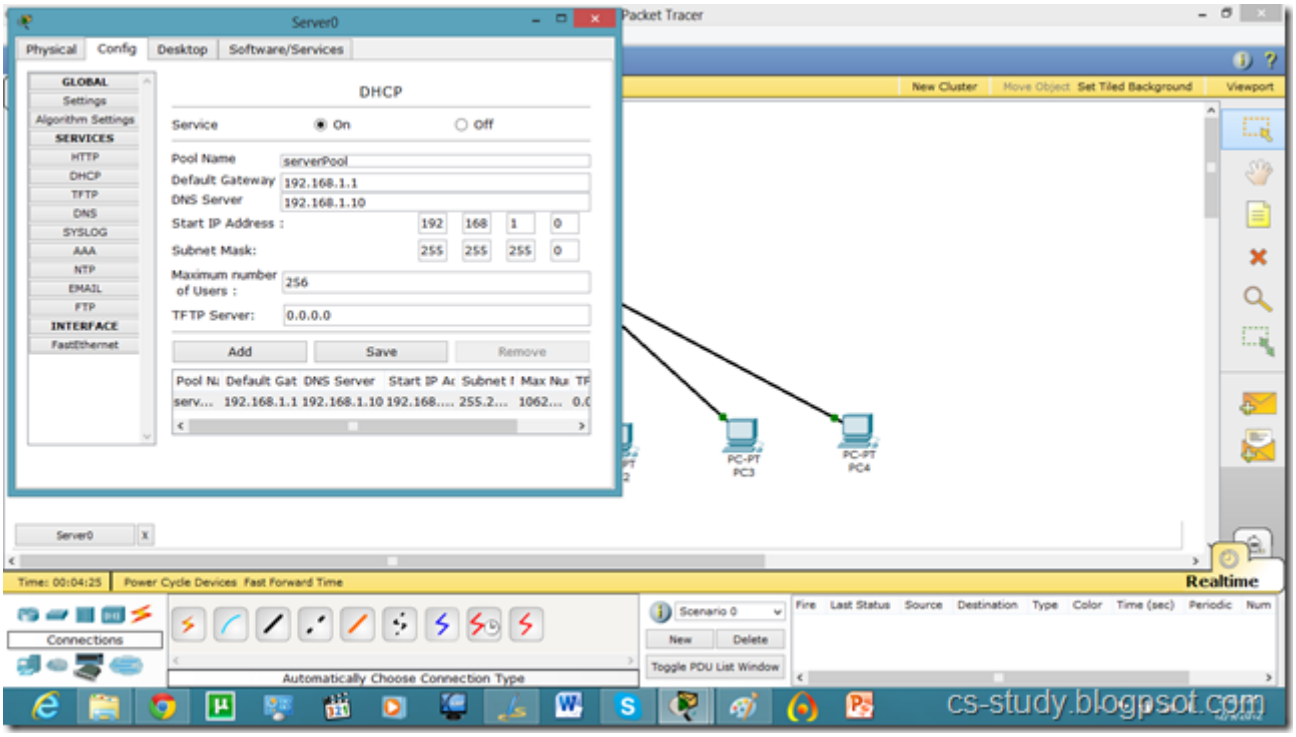
Let us apply this concept on packet tracer.



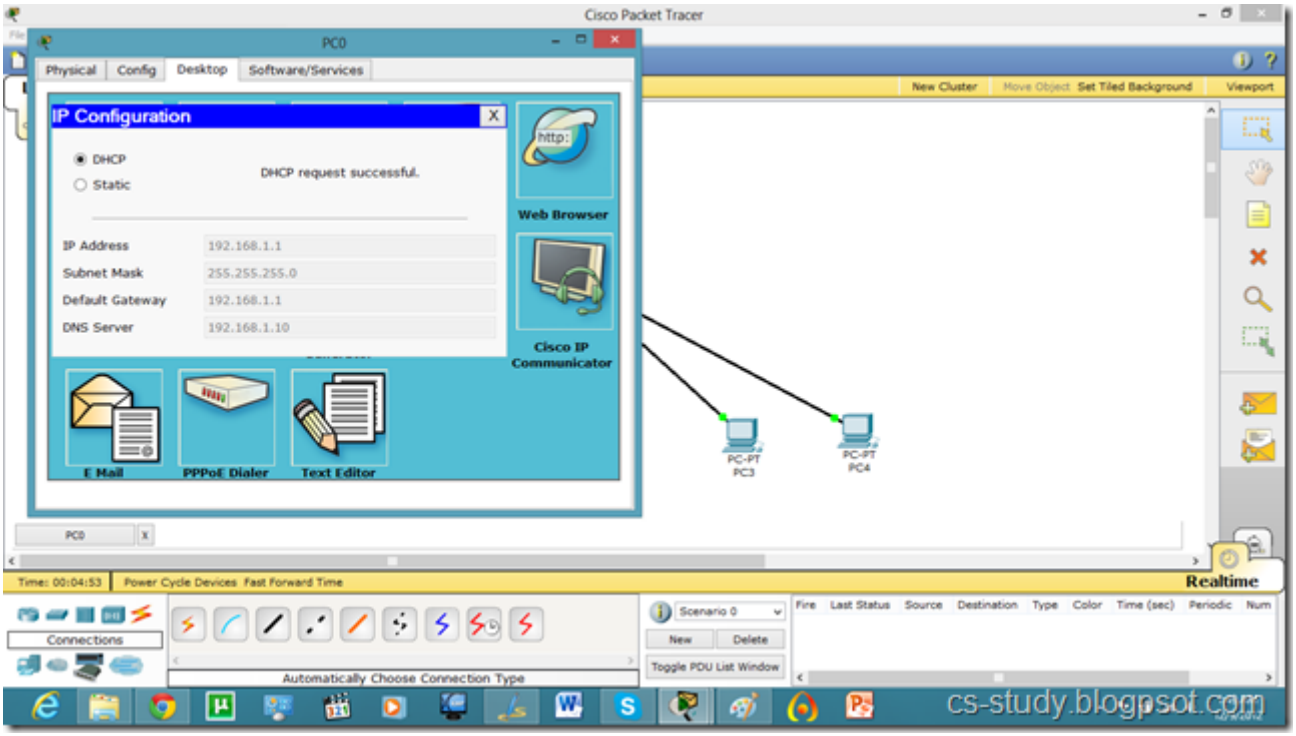
Let us set up a topology. Apply PCs with the IP addresses dynamically with the DHCP server set on the Server.



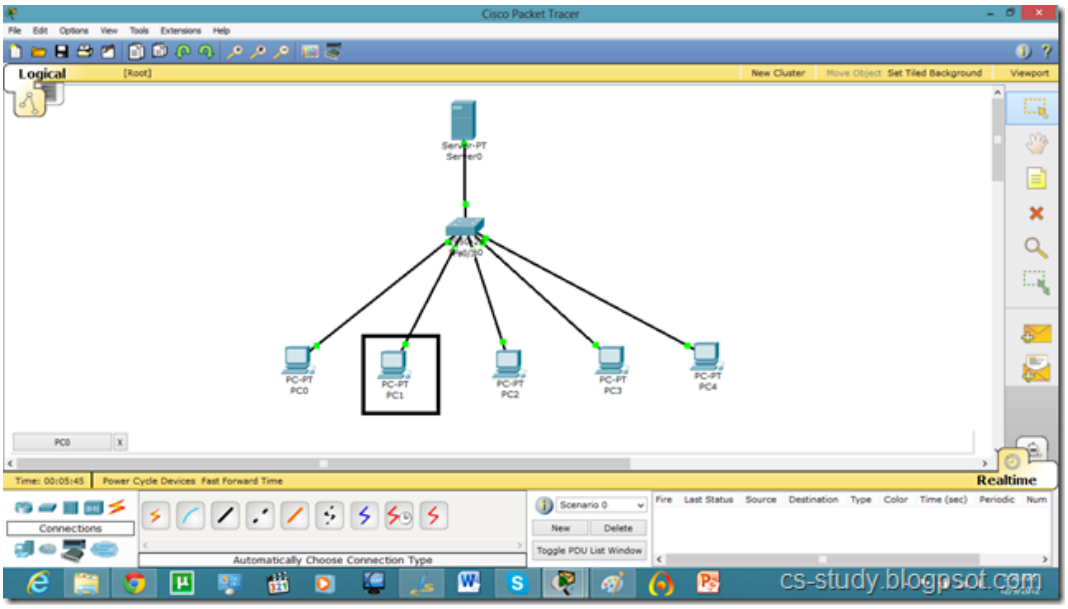
DHCP server setup on server.



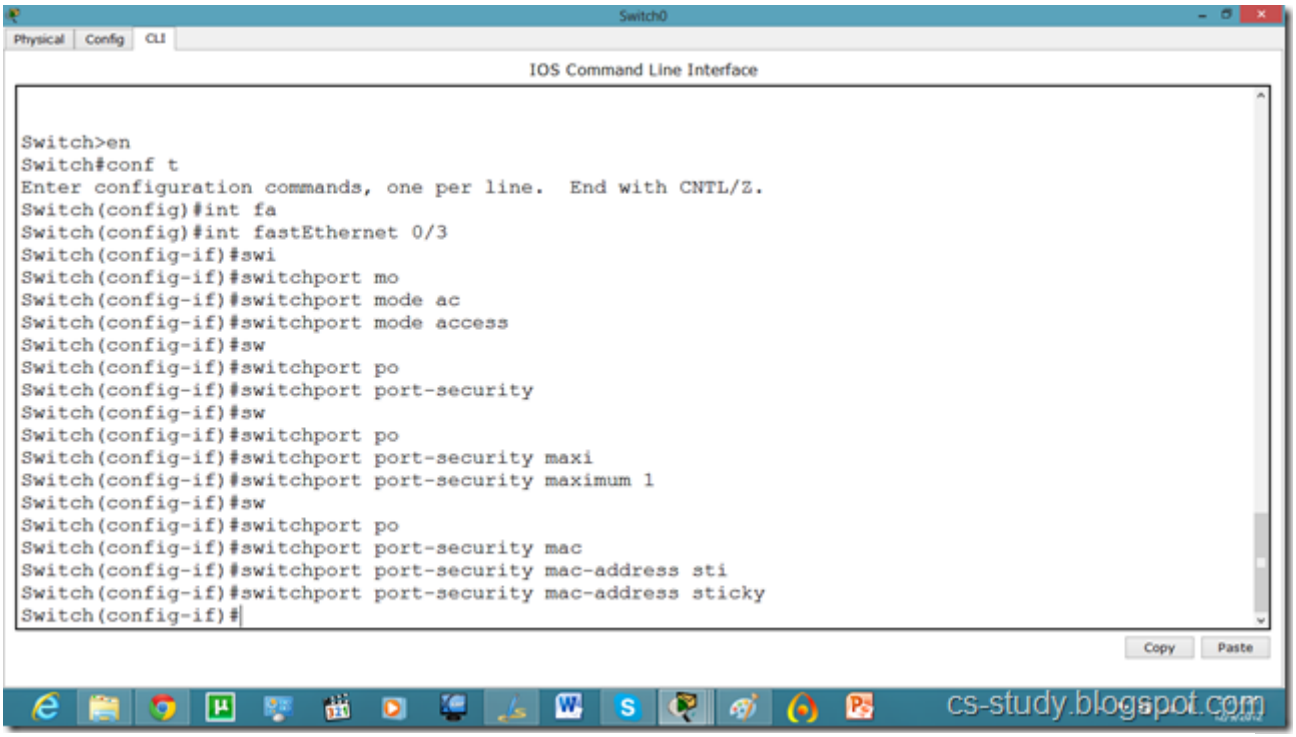
Assigning IP to
PC



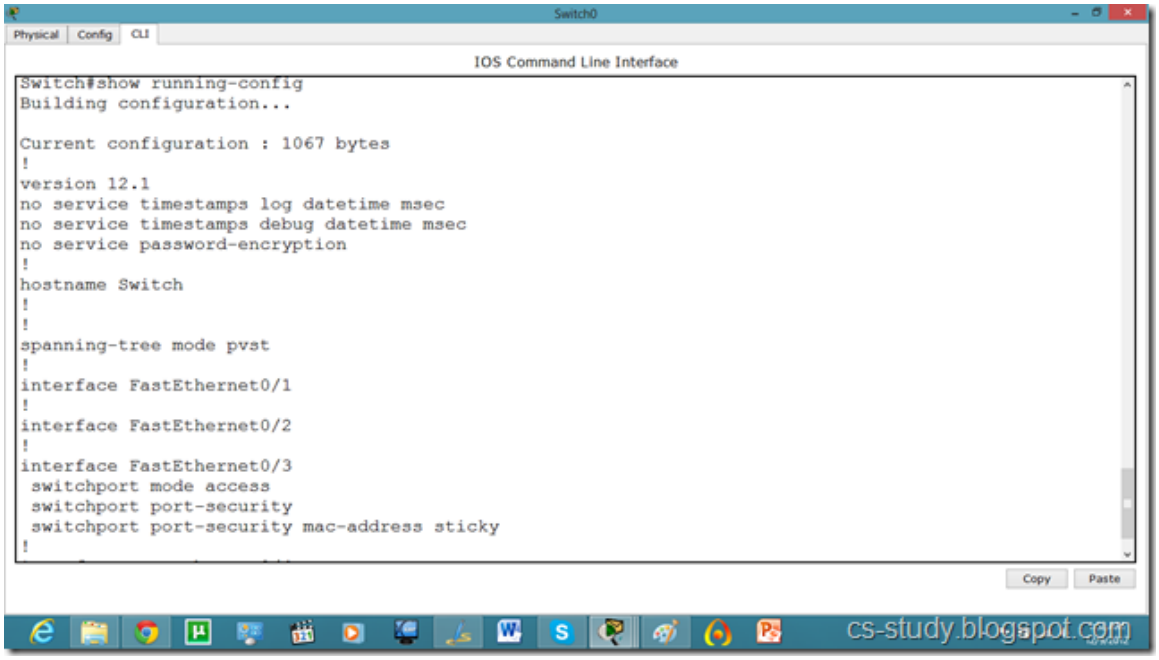
Now, let us apply Sticky Mac address to the following interface of switch which is currently attach to PC1. It is interface fa 0/3.



The following commands will apply this concept.

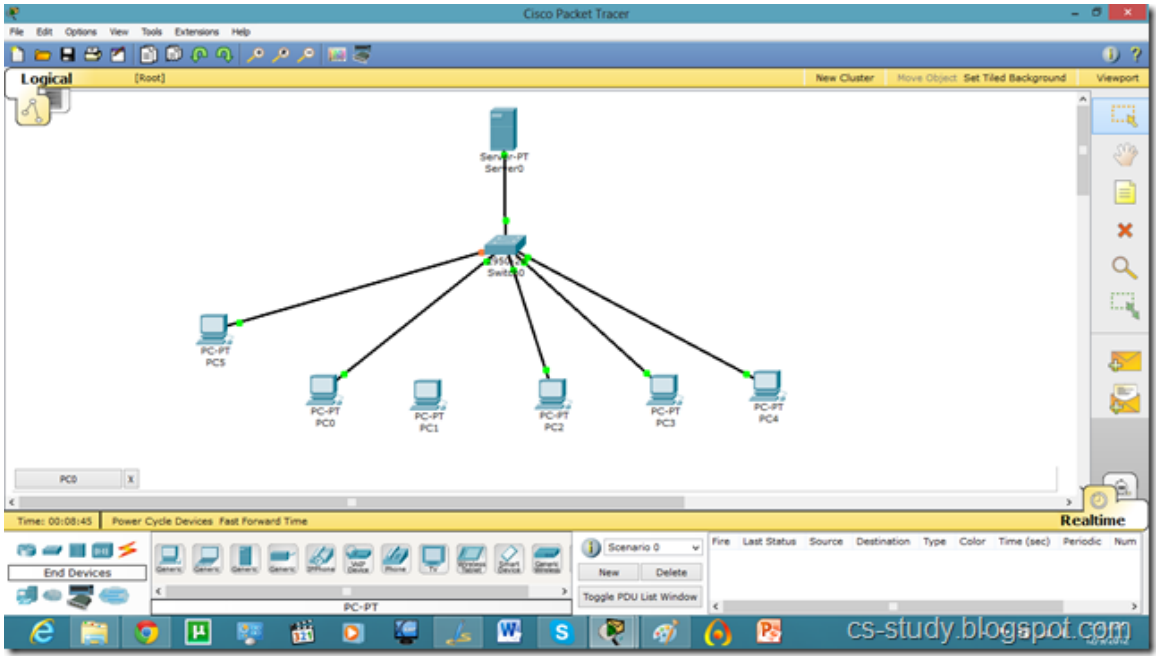


Go to enable mode. Apply command, show running-config.

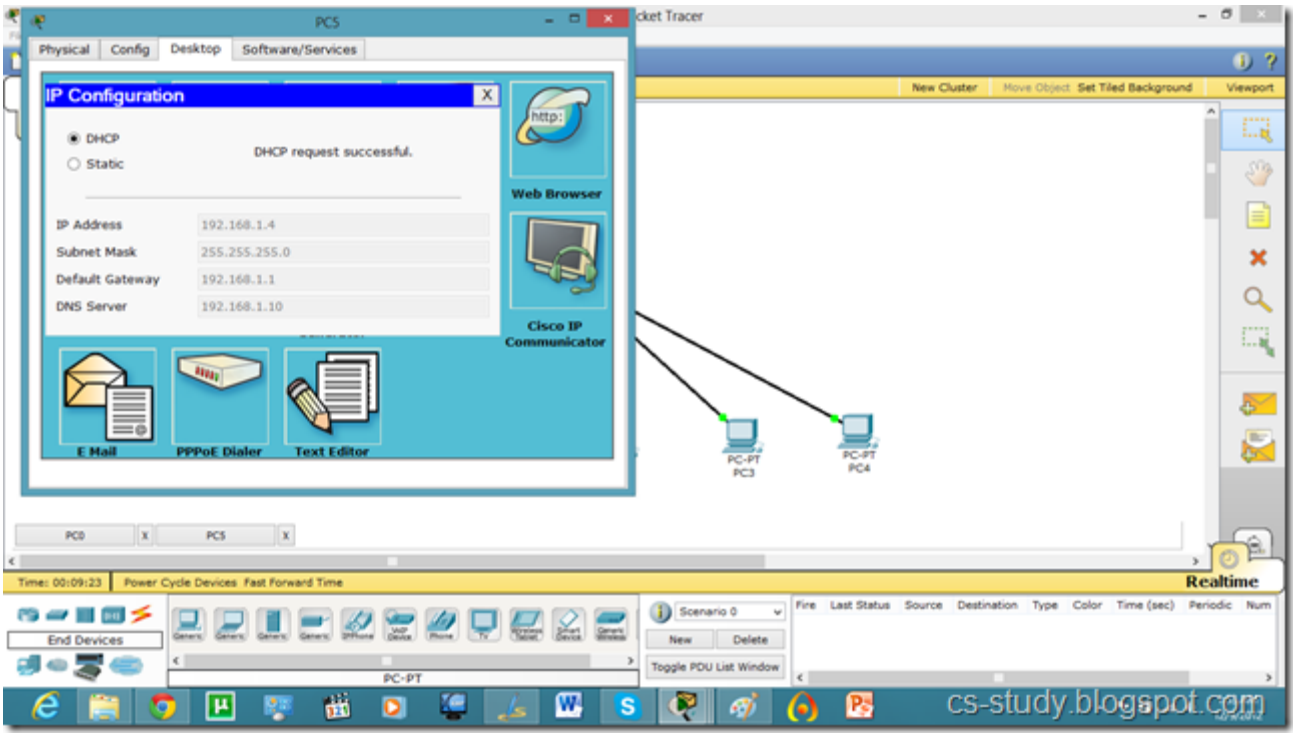


Now, we do not have MAC address of PC here.

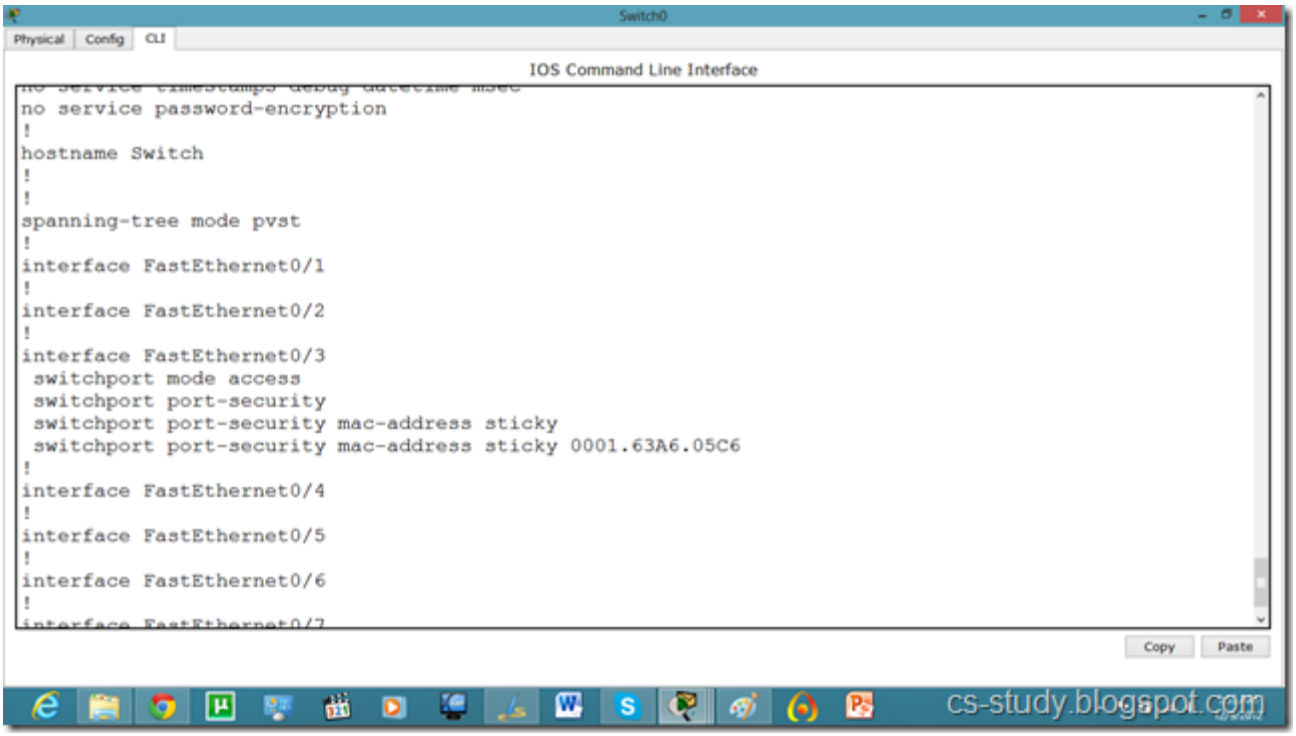
For this, remove the interface from PC1 attach it to another PC as shown in the figure below.



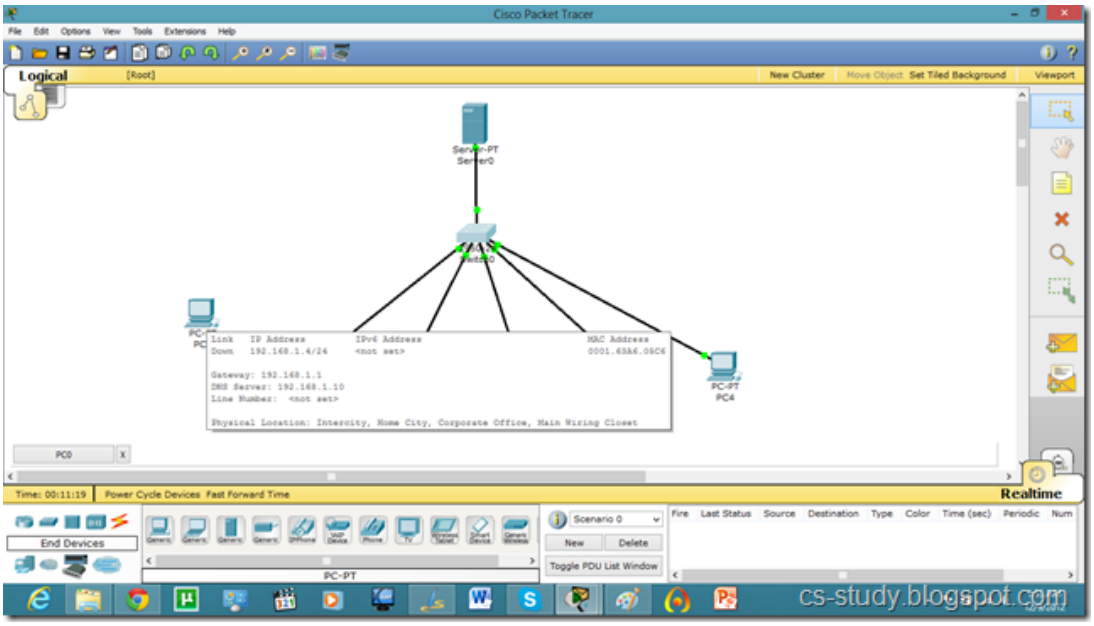
Request for DHCP.



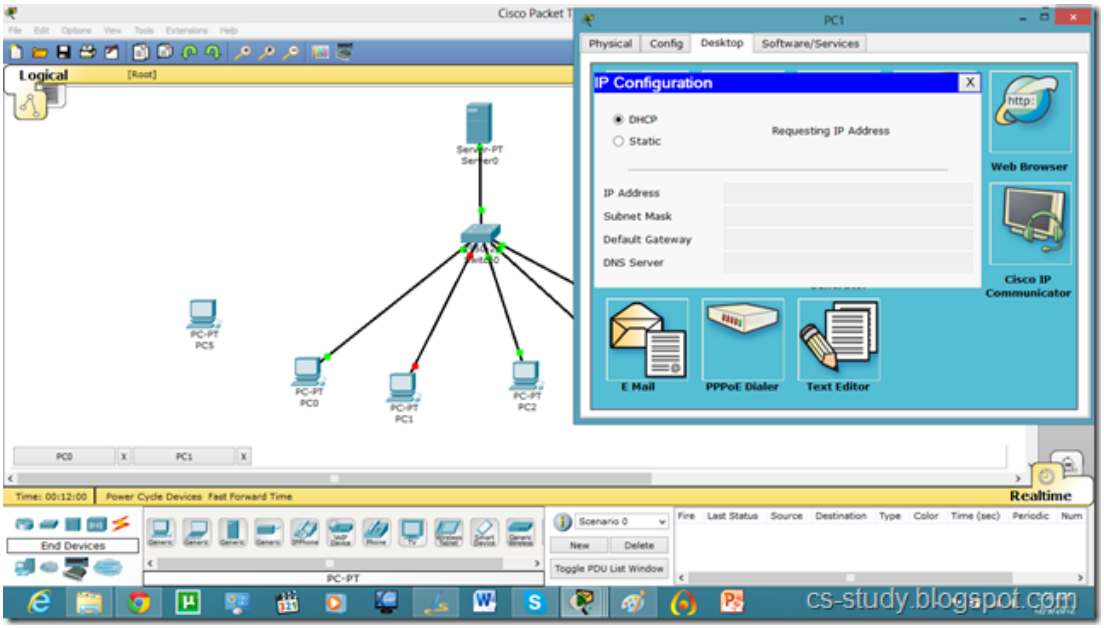
Now. we have mac address and our port is sticky as shown in figure below.



MAC address of PC.

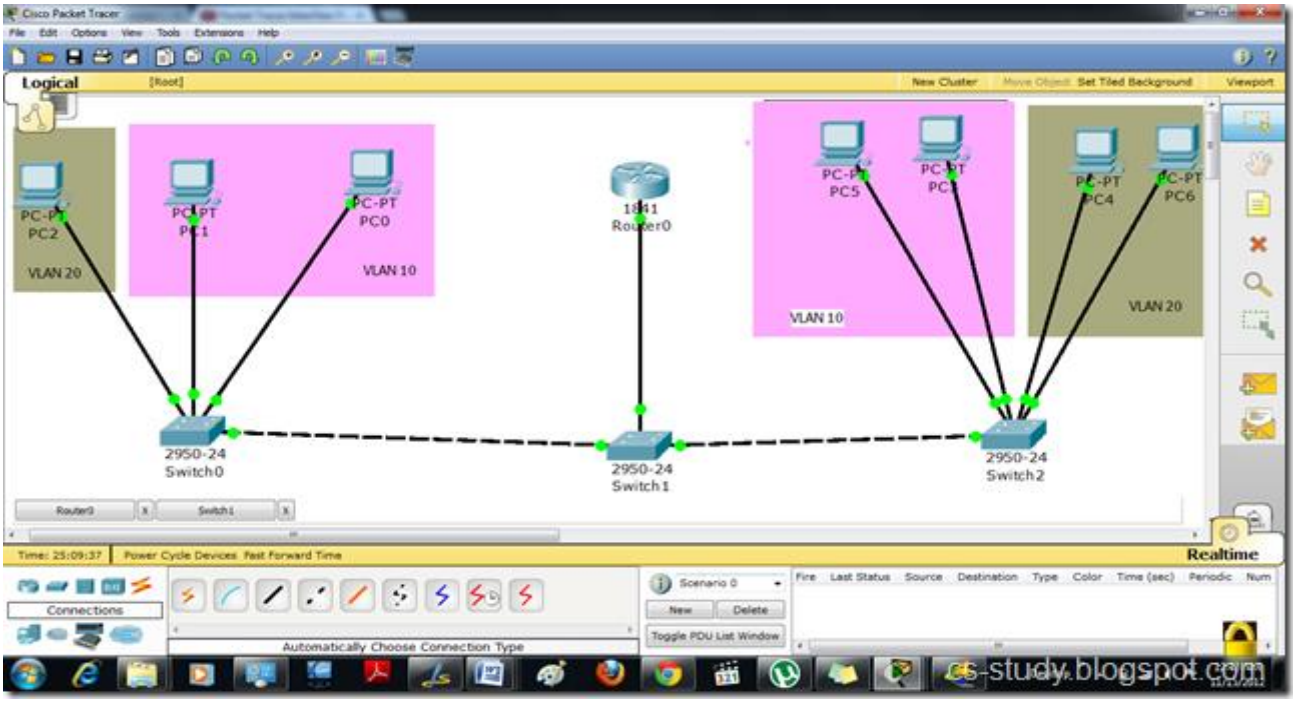


Now, re connect to PC1 and request for IP address. It has failed and it has also shutdown the interface because MAC address does not match with the MAC address of the port.



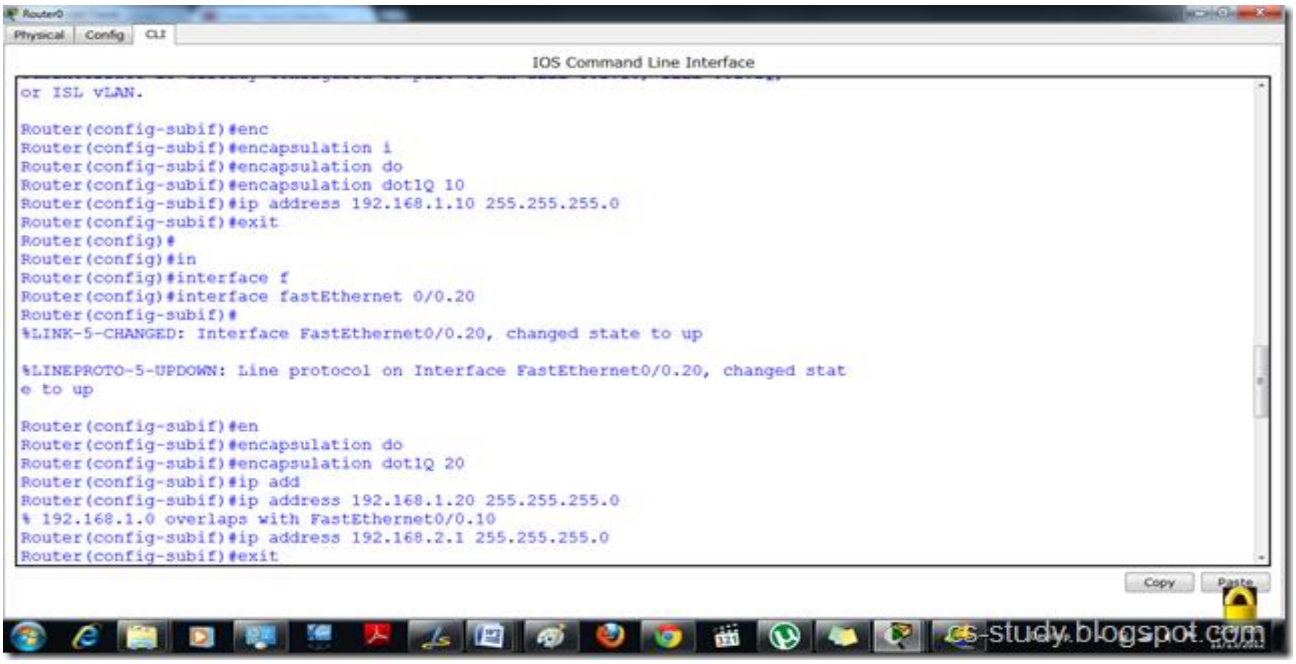
Task 3: Inter VLAN routing on packet tracer

In this tutorial we are going apply inter vlan routing. In order to communicate between different vlans, we use this phenomenon. Let us apply vlan on the following topology, then we will apply vtp (vlan trunking protocol).

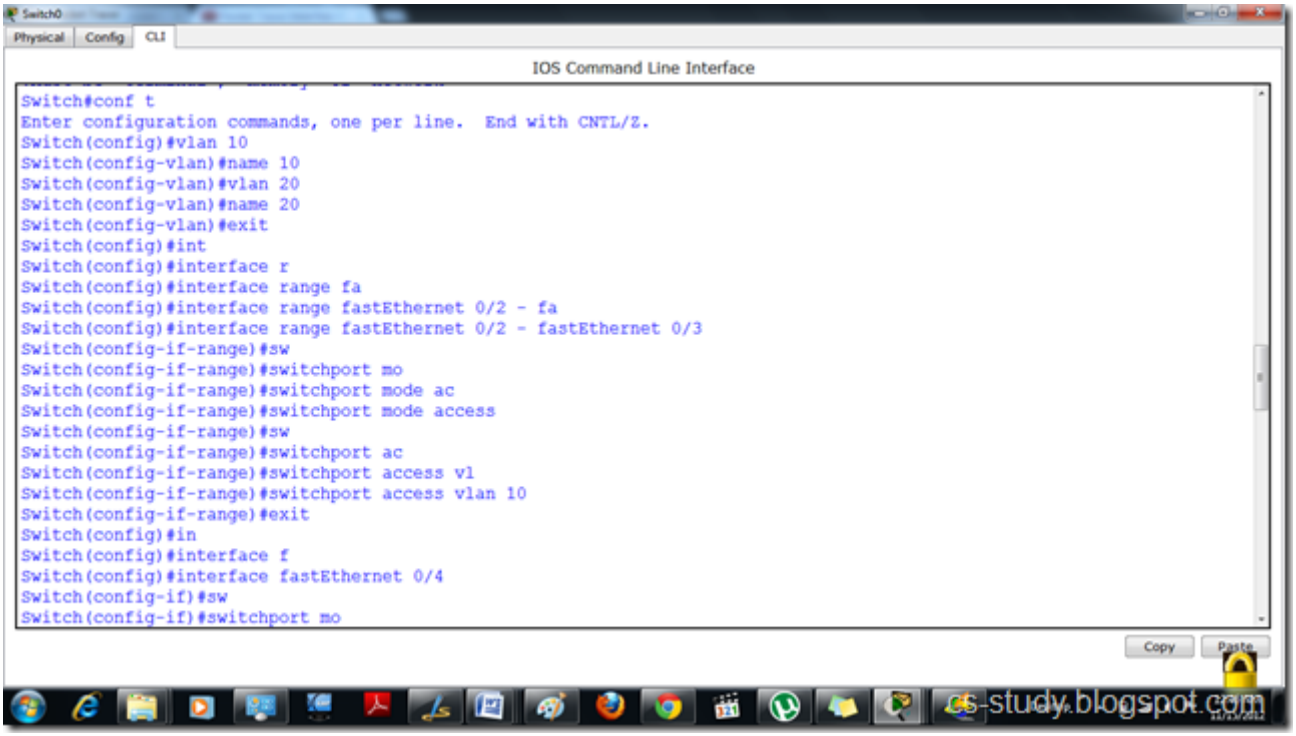


Now, in order to communicate between different vlans we will have to create sub interfaces of fast ethernet interface of router. Let us do that. Note that, we will create as many sub interfaces as many vlans we are using in our topology. In this case we are using 2 vlans, so we will create two sub interfaces for both vlans.

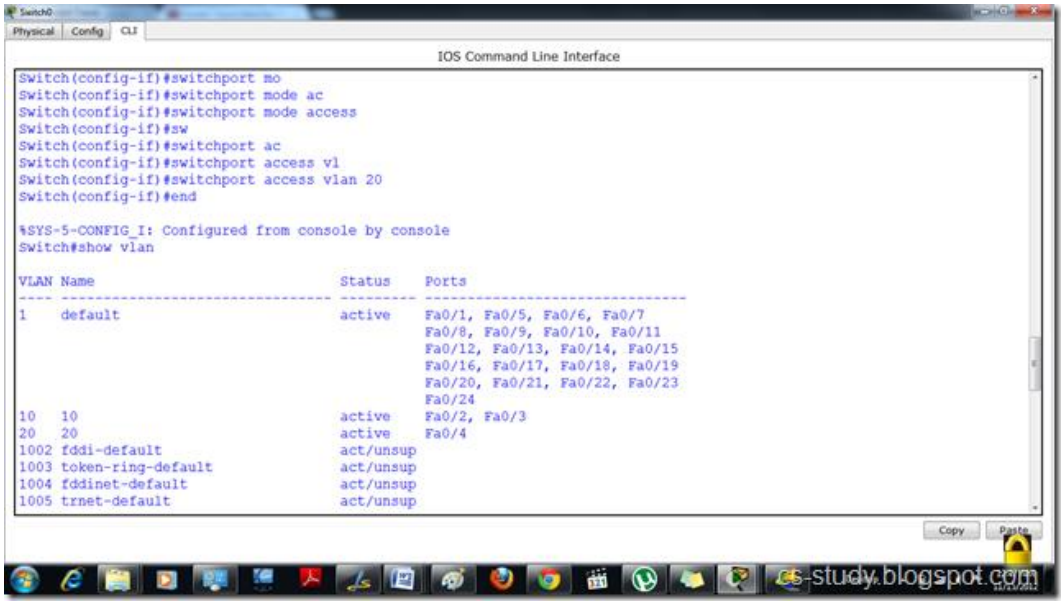
We will also apply encapsulation of sub interface.



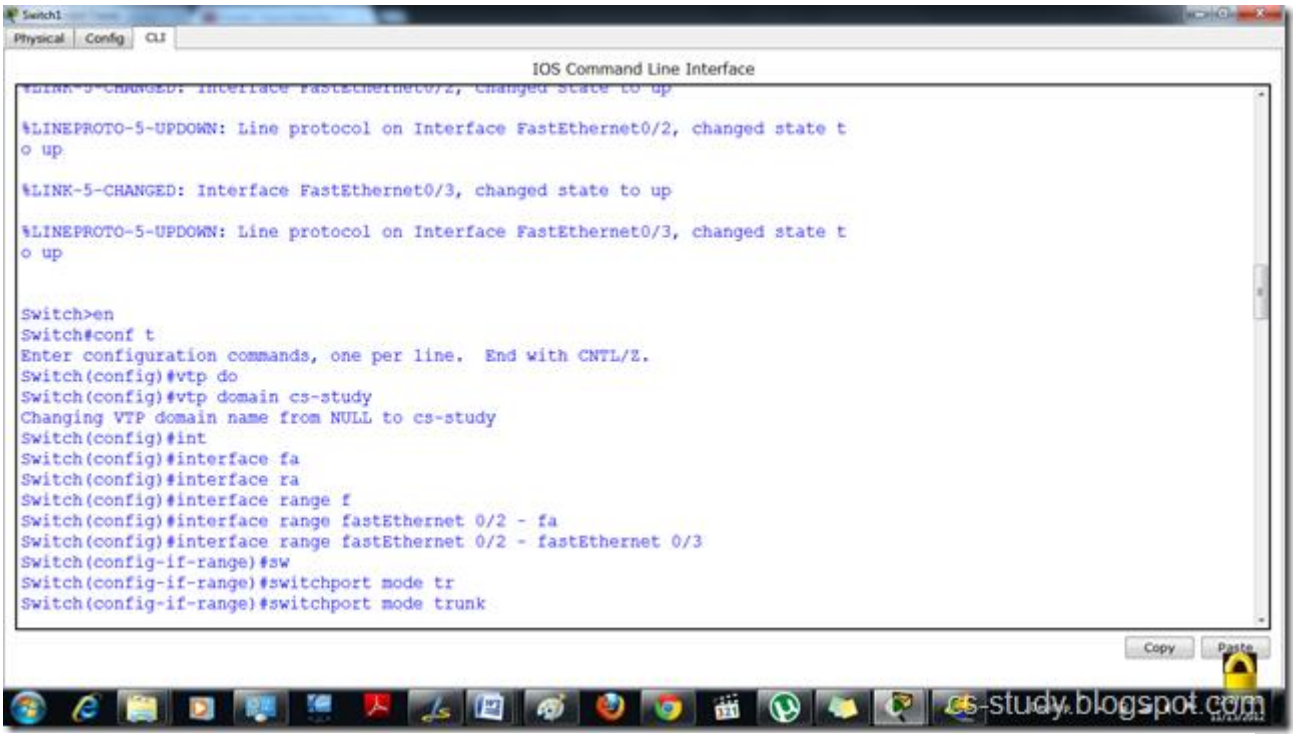
Let us create vlans and assign those vlans to switch interfaces i.e. PCs



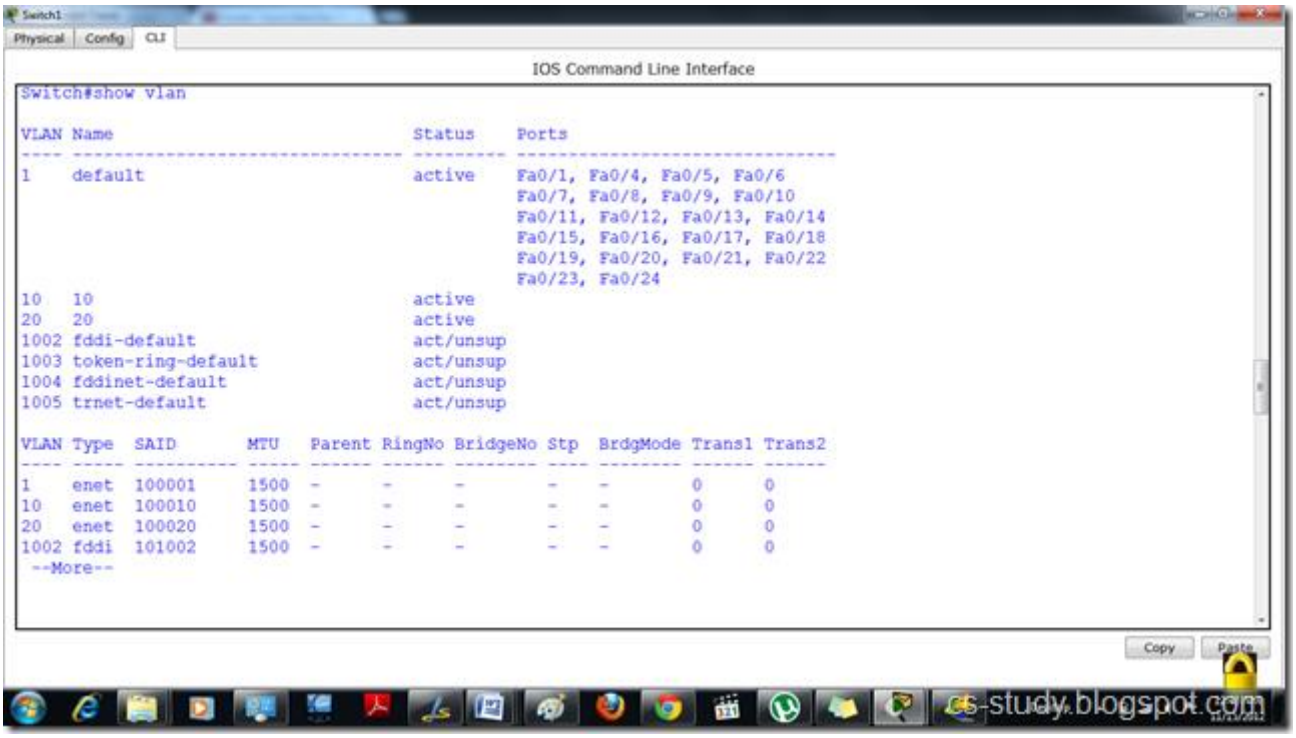
And,



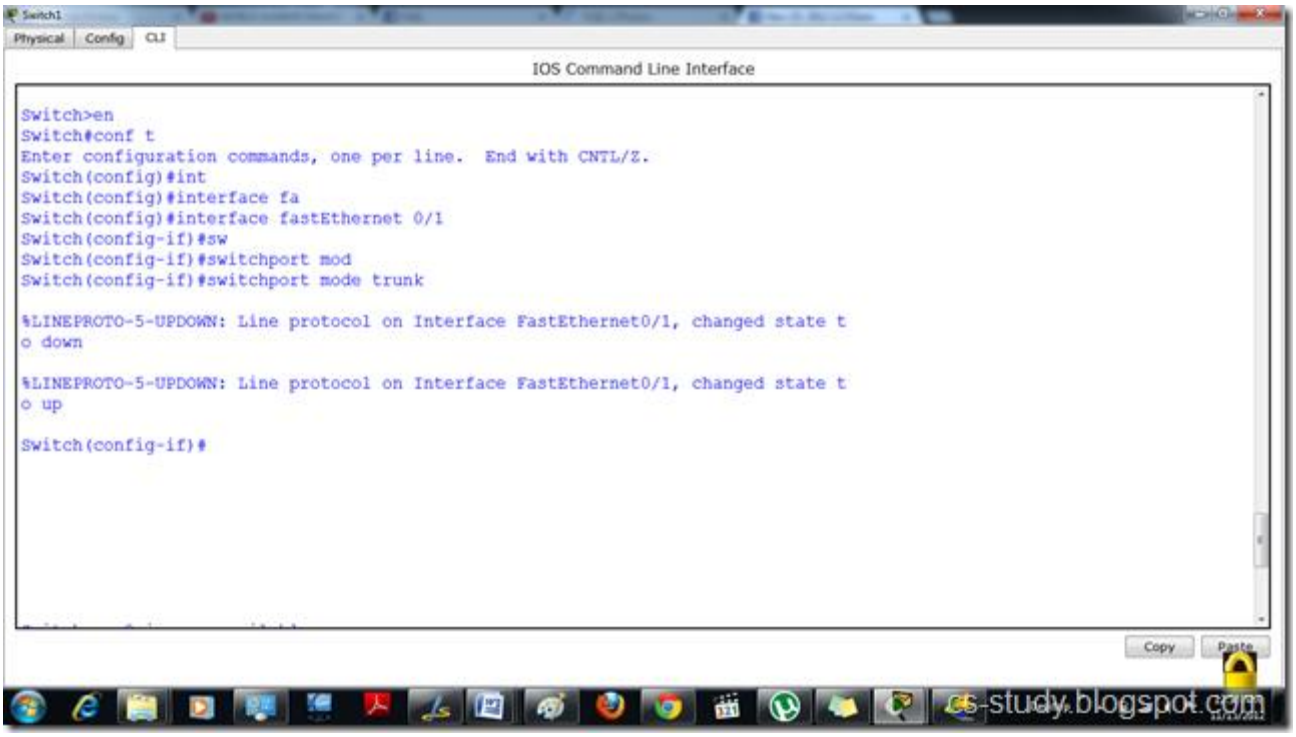
In order to apply VTP we will have write the following commands. Apply the domain name, and trunk the ports of switch. Note, trunk only those ports that are connected to router and switches.



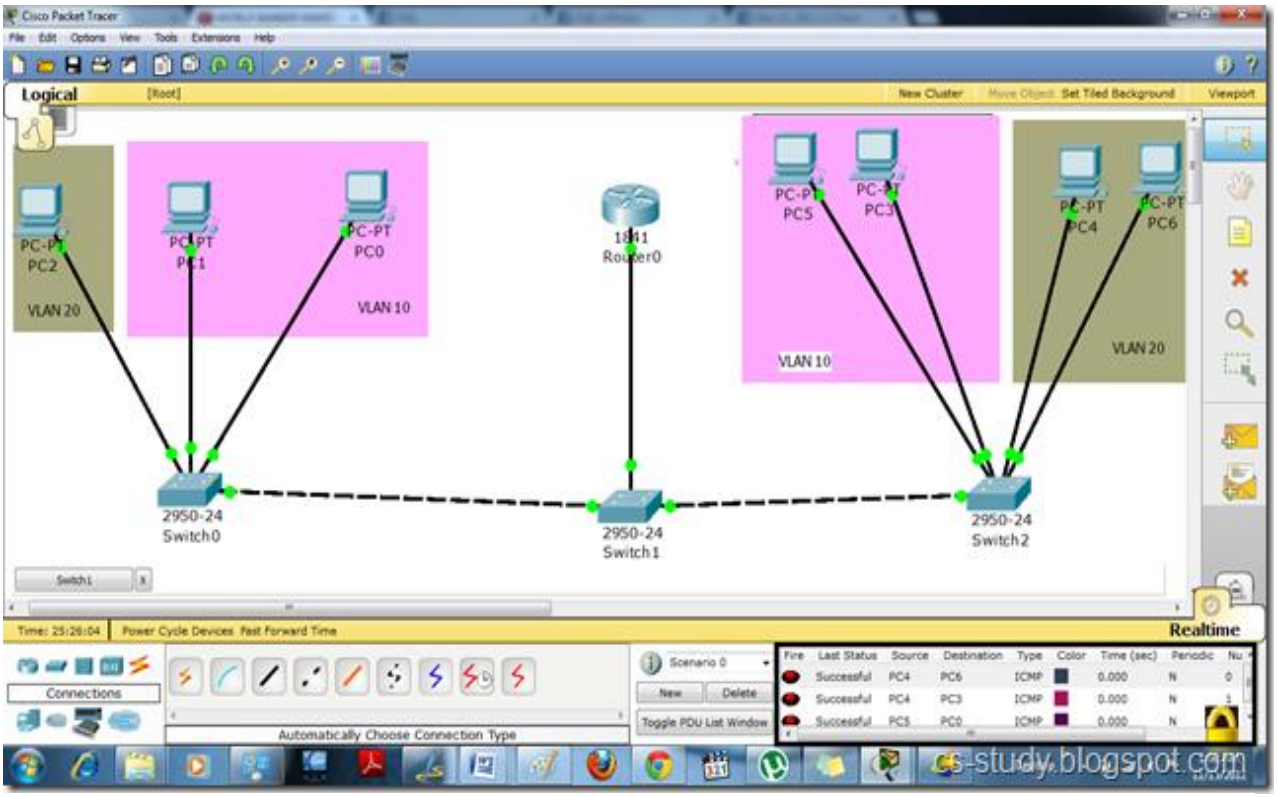
Thus, vlans are visible in other switches too after trunking.



Trunk the port of the switch 1 as well.



Now, we see, in the bottom right corner, we are able to communicate in between vlans and that s what we wanted , isn't it ?



Lab 10 – CDP, Telnet and SSH, PAP and CHAP

Objectives

In this lab, students will learn about:

- Cisco Discovery Protocol
- Telnet and SSH
- PAP and CHAP

Lab Tasks

Task 1: Apply CDP

Cisco Discovery Protocol (CDP) is primarily used to obtain protocol addresses of neighboring devices and discover the platform of those devices. CDP can also be used to show information about the interfaces your router uses. CDP is media- and protocol-independent, and runs on all Cisco-manufactured equipment including routers, bridges, access servers, and switches.

Explanation:

CDP runs on all media that support Subnetwork Access Protocol (SNAP), including local-area network (LAN), Frame Relay, and Asynchronous Transfer Mode (ATM) physical media. CDP runs over the data link layer only. Therefore, two systems that support different network-layer protocols can learn about each other.

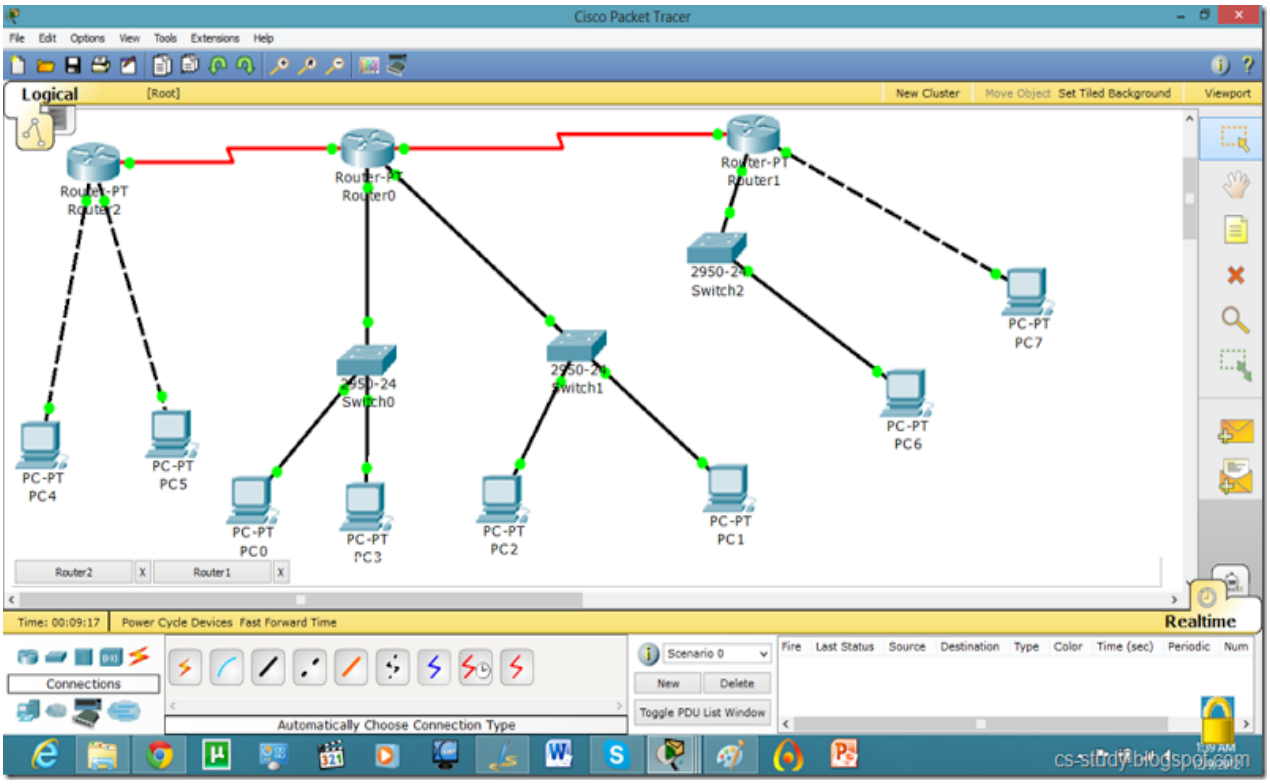
Each device configured for CDP sends periodic messages, known as advertisements, to a multicast address. Each device advertises at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime, information, which indicates the length of time a receiving device should hold CDP information before discarding it. Each device also listens to the periodic CDP messages sent by others in order to learn about neighboring devices and determine when their interfaces to the media go up or down.

CDP is a Cisco proprietary Layer 2 protocol that is media- and protocol-independent, and runs on all Cisco-manufactured equipment that includes:

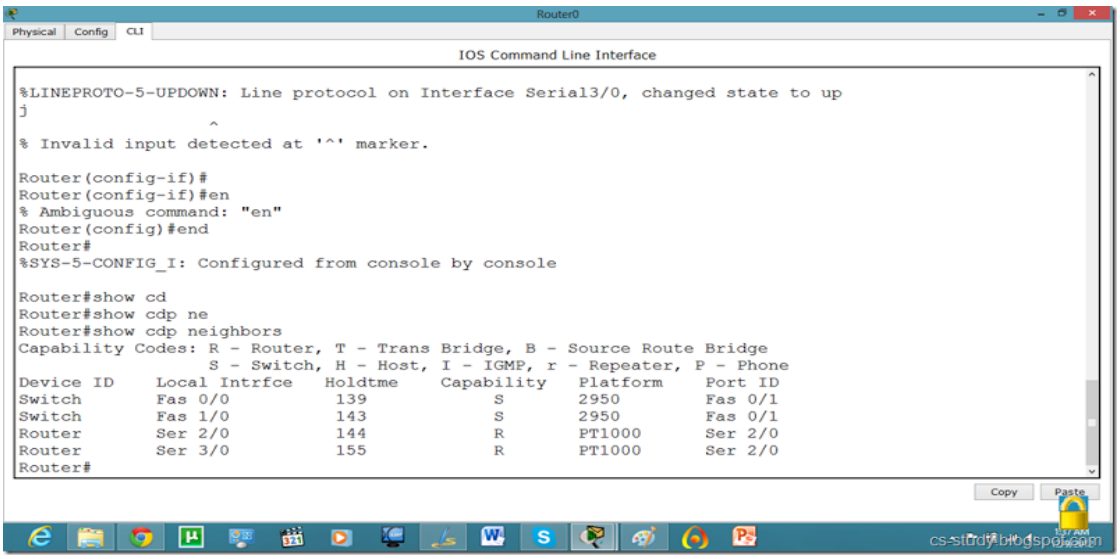
- routers
- bridges
- access servers
- switches

A Cisco device enabled with CDP sends out periodic interface updates to a multicast address in order to make itself known to neighbors. Since it is a layer two protocol, these packets (frames) are not routed. Use of SNMP with the CDP MIB allows network management applications to learn the device type and the SNMP agent address of neighboring devices, and to send SNMP queries to those devices.

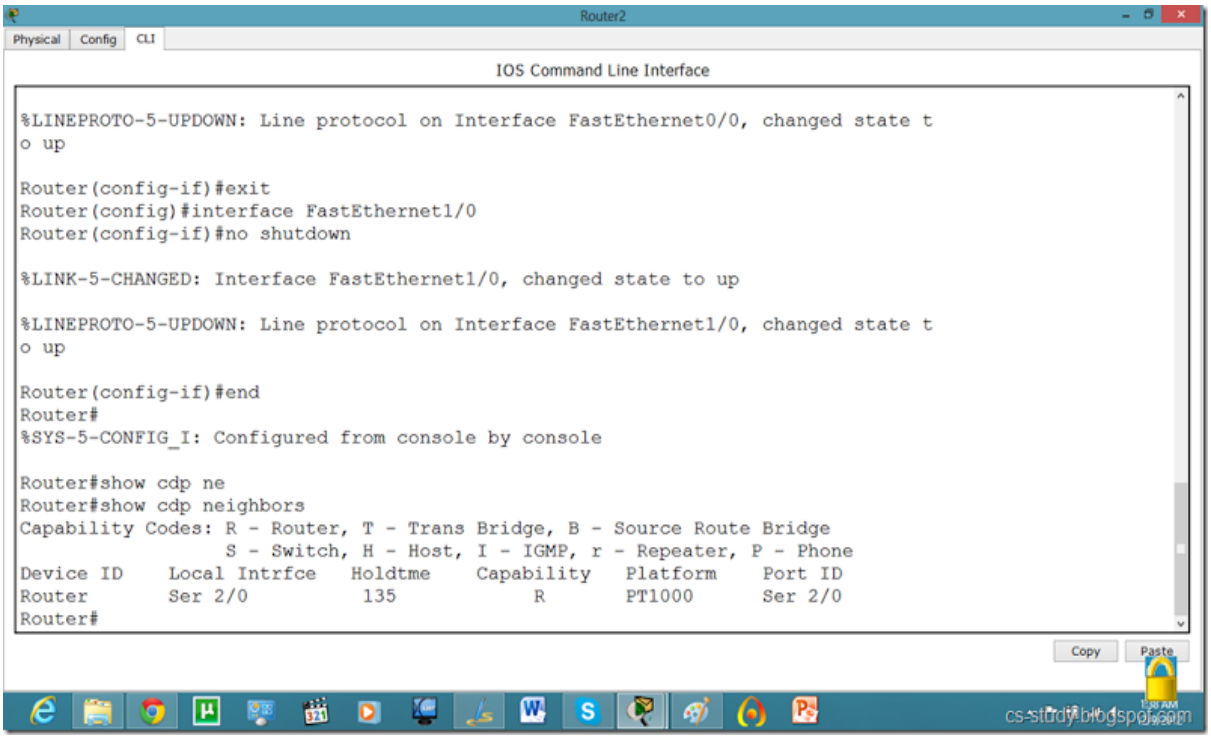
Let us apply CDP on packet tracer.



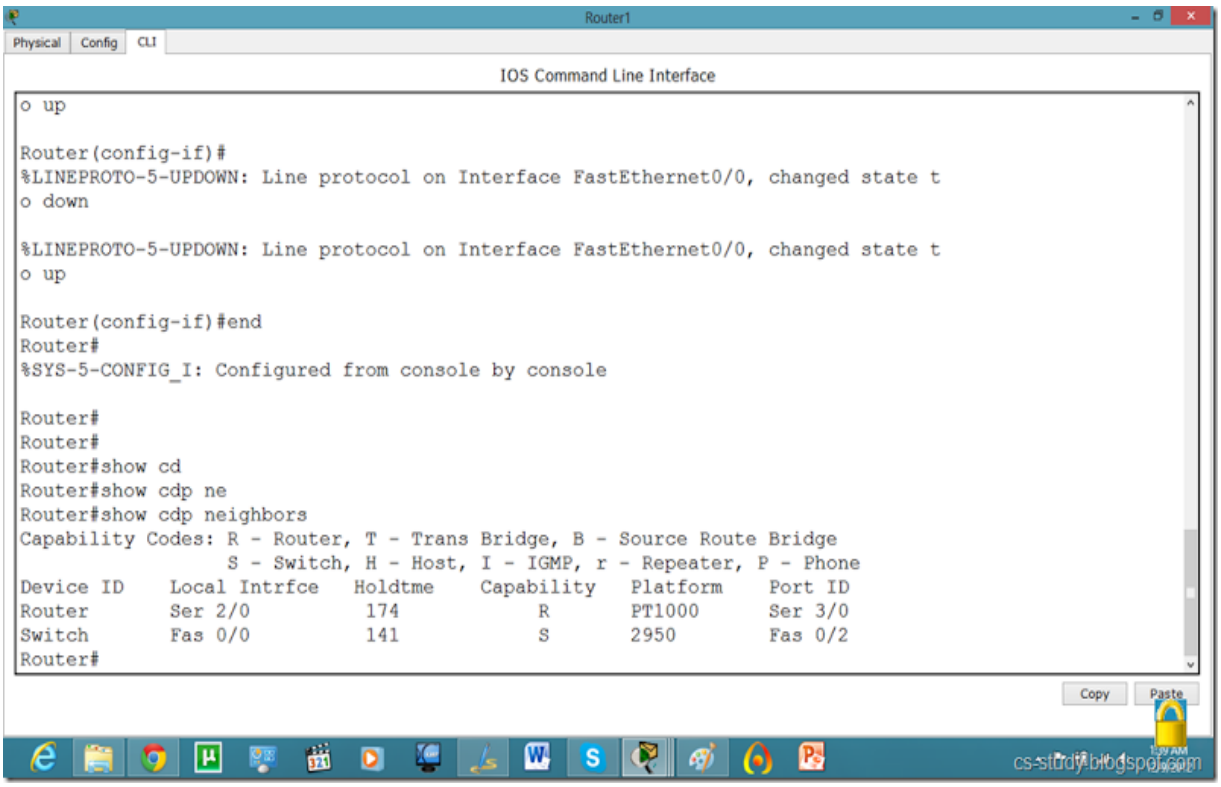
In the above topology, we have different devices attached with each other. So, if we want to look for the information of neighboring devices, we will apply the following command on Router enabled mode, “show cdp neighbors”.



And on Router 2 and 3 as well, as shown in figures below.



It can be seen clearly that it gives us the information of the neighboring devices. Note one thing here, that it gives us the information of the routers and switches that are directly attached to its ports. However, it does not give us information about the hosts that are attached to it directly.



Task 2: Apply Telnet and SSH.

A terminal emulation program for TCP/IP networks such as the Internet. The Telnet program runs on your computer and connects your PC to a server on the network. You can then enter commands through the Telnet program and they will be executed as if you were entering them directly on the server console. This enables you to control the server and communicate with other servers on the network. To start a Telnet session, you must log in to a server by entering a valid username and password. Telnet is a common way to remotely control Web servers. To telnet means to establish a connection with the Telnet protocol, either with command line client or with a programmatic interface.

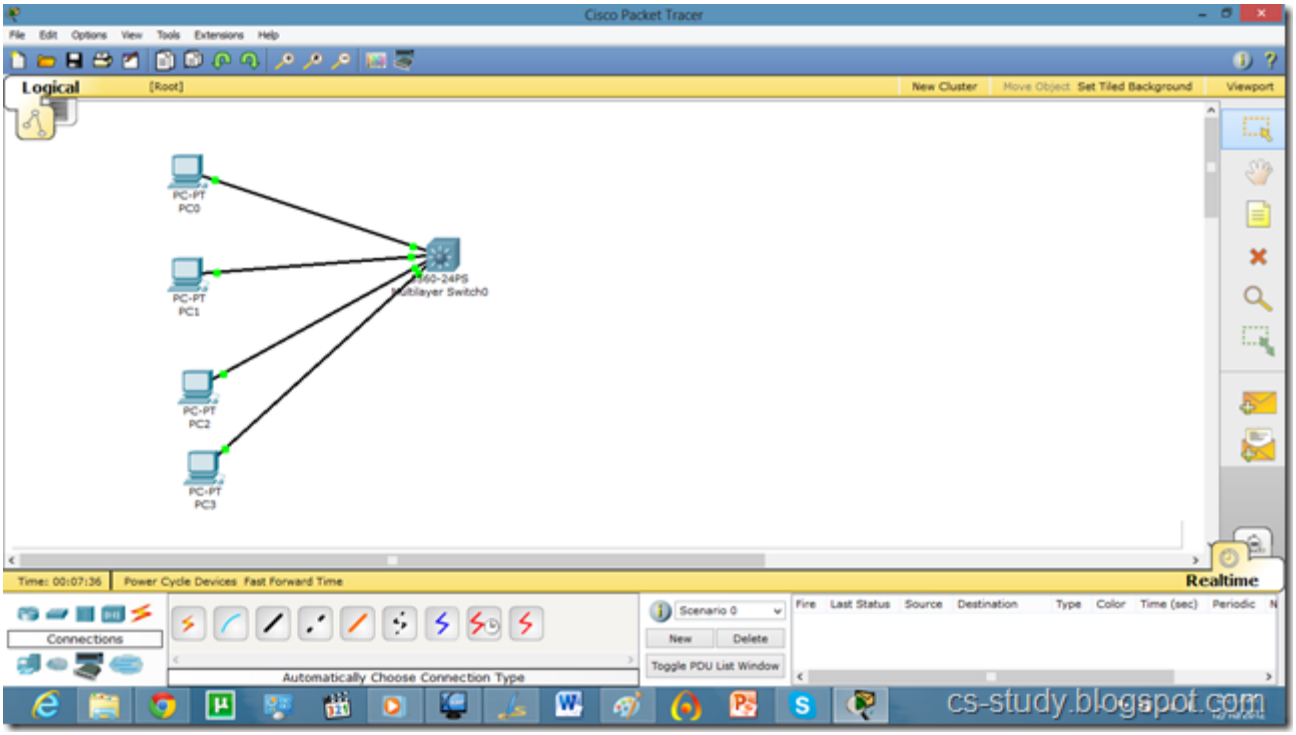
SSH

Secure Shell (SSH) is a cryptographic network protocol for secure data communication, remote shell services or command execution and other secure network services between two networked computers that connects, via a secure channel over an insecure network, a server and a client (running SSH server and SSH client programs, respectively). It was designed as a replacement for Telnet and other insecure remote shell protocols such as the Berkeley rsh and rexec protocols, which send information, notably passwords, in plaintext, rendering them susceptible to interception and disclosure using packet analysis. The encryption used by SSH is intended to provide confidentiality and integrity of data over an unsecured network, such as the Internet.

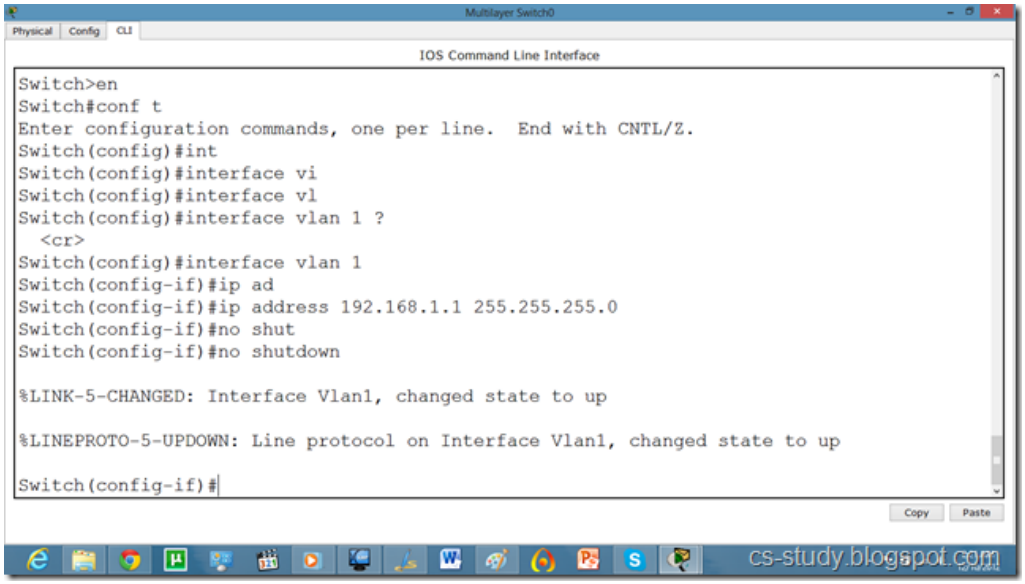
A network protocol that ensures a high-level encryption, allowing for the data transmitted over insecure networks, such as the Internet, to be kept intact and integrate. SSH and SSH Telnet, in particular, work for establishing a secure communication between two network-connected computers as an alternative to remote shells, such as TELNET, that send sensitive information in an insecure environment.

In contrast to other remote access protocols, such as FTP, SSH Telnet ensures higher level of connection security between distant machines but at the same time represents a potential threat to the server stability. Thus, SSH access is considered a special privilege by hosting providers and is often assigned to users only per request.

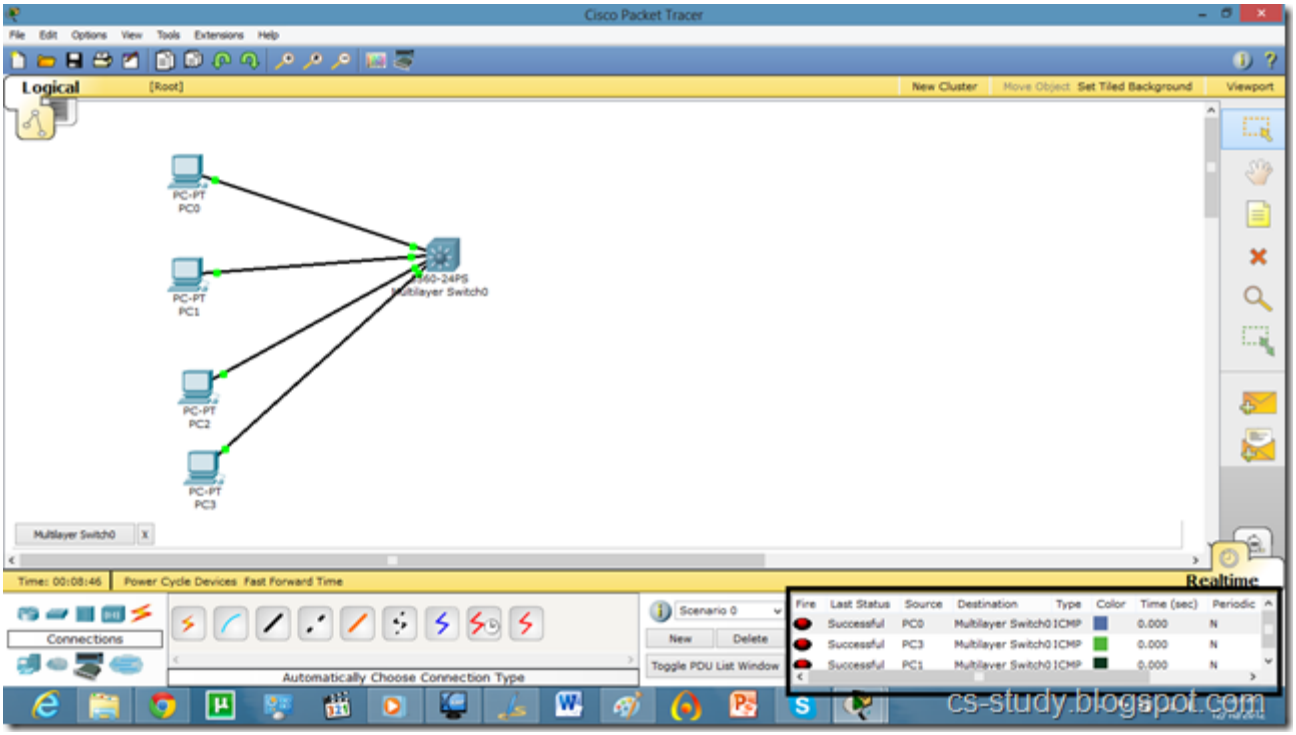
Let us apply Telnet and SSH on packet tracer.



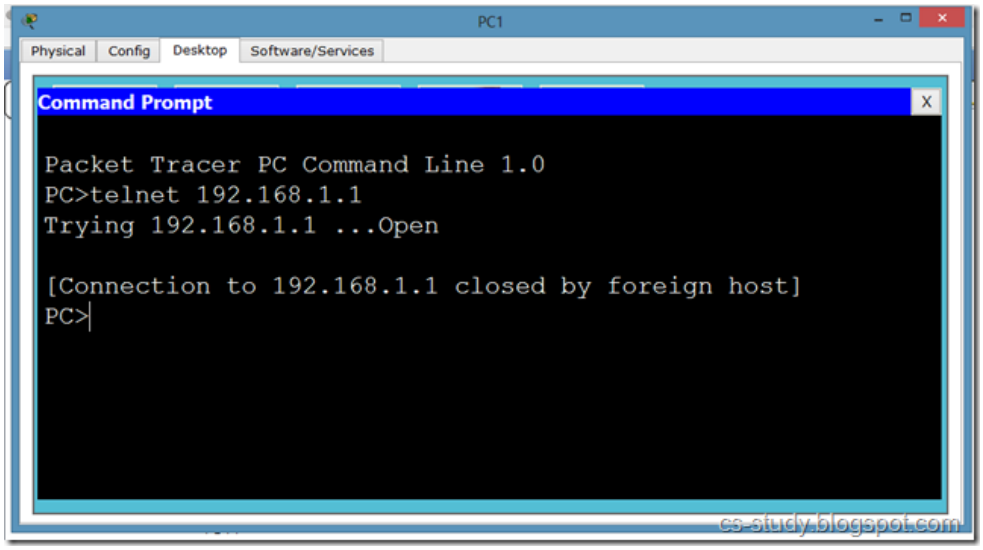
Take the topology as in the above diagram. Set IPs on the PCs. As, by default, all PCs are in vlan 1. We will create a virtual interface on switch with vlan 1 as follows.



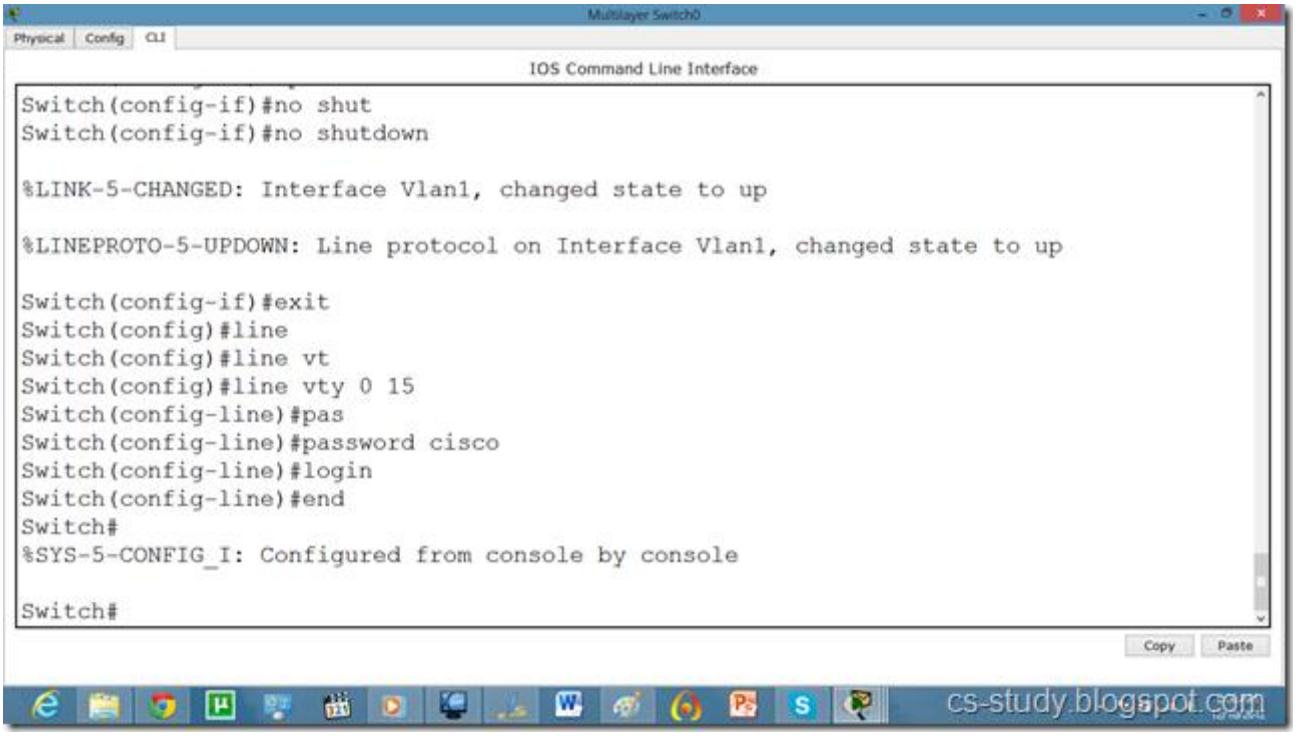
Now, we can ping to switch by our hosts because hosts are in vlan 1 and switch also has a vlan 1 interface.



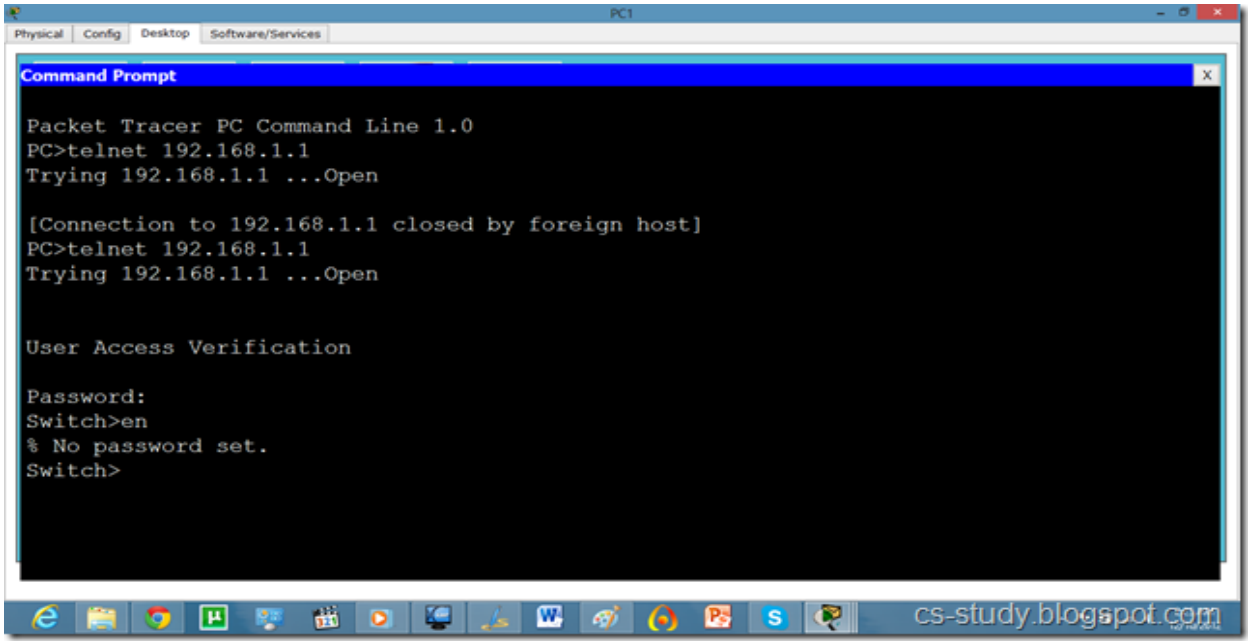
Now, try to telnet the switch from our PC, it refuses because we have not applied authentication on the switch yet.



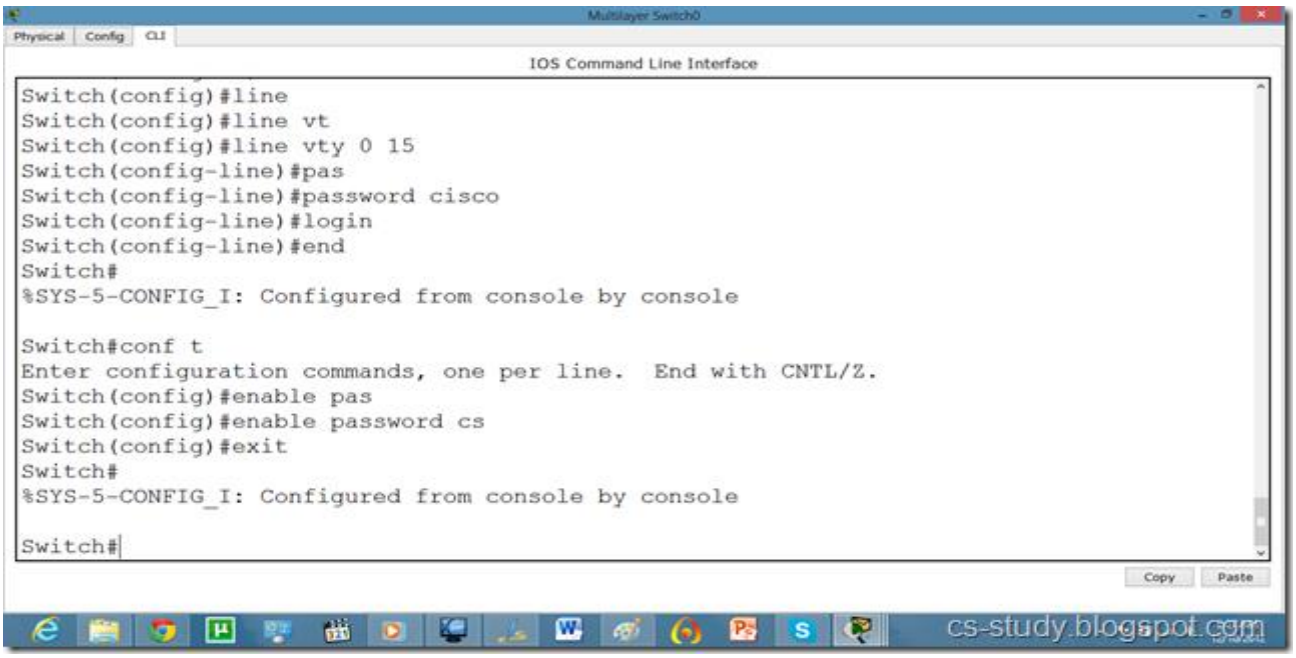
So, lets apply line authentication on the switch. The system supports 20 virtual tty (vty) lines for Telnet, Secure Shell Server (SSH) and FTP services. Each Telnet, SSH, or FTP session requires one vty line. You can add security to your system by configuring the software to validate login requests.



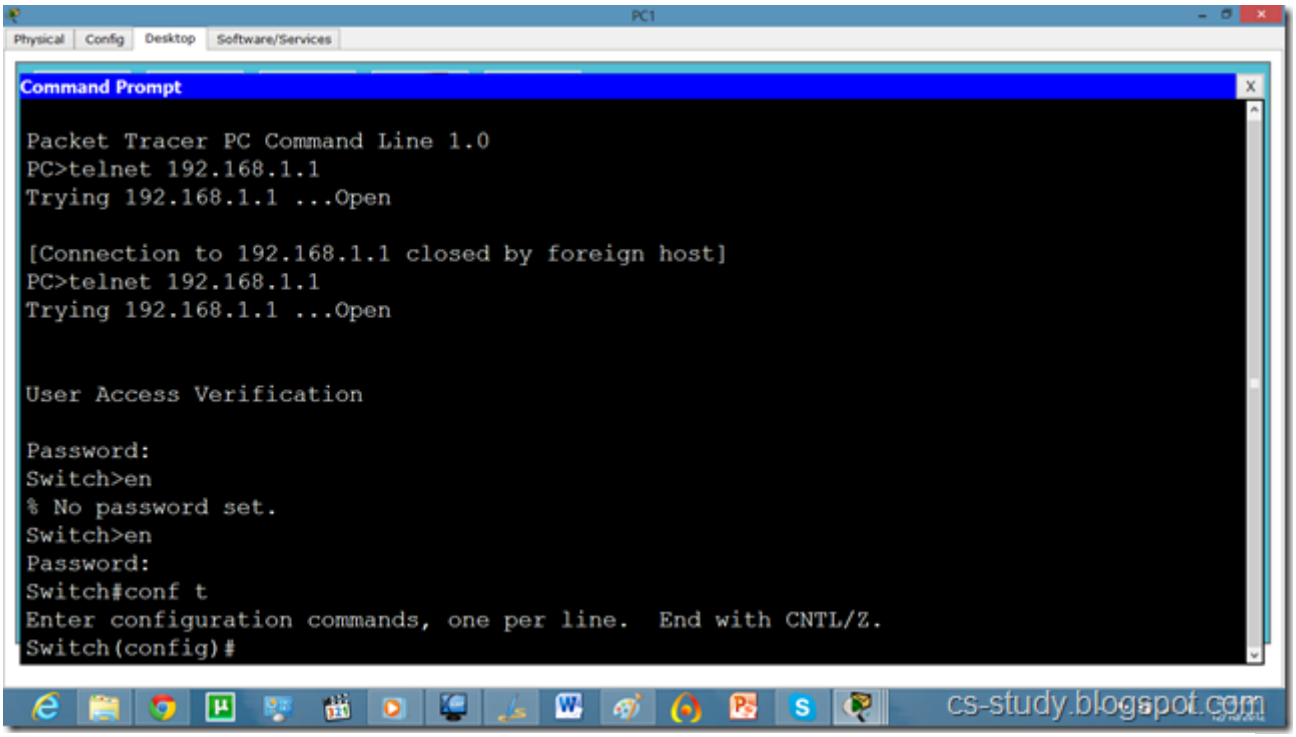
Now, we can easily telnet. But it does not let us go in the switch enabled mode because we have not set the password on the switch yet.



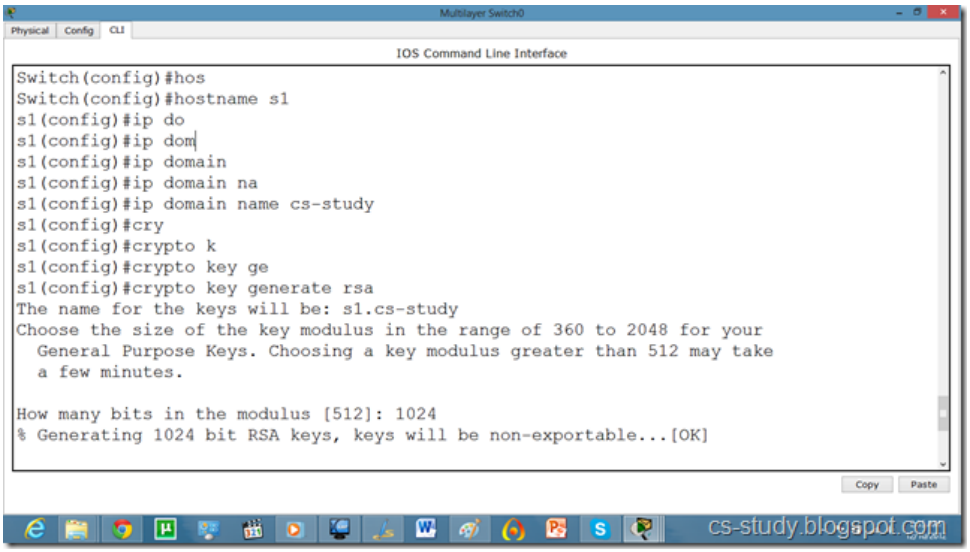
Let's apply password on the switch enabled mode.



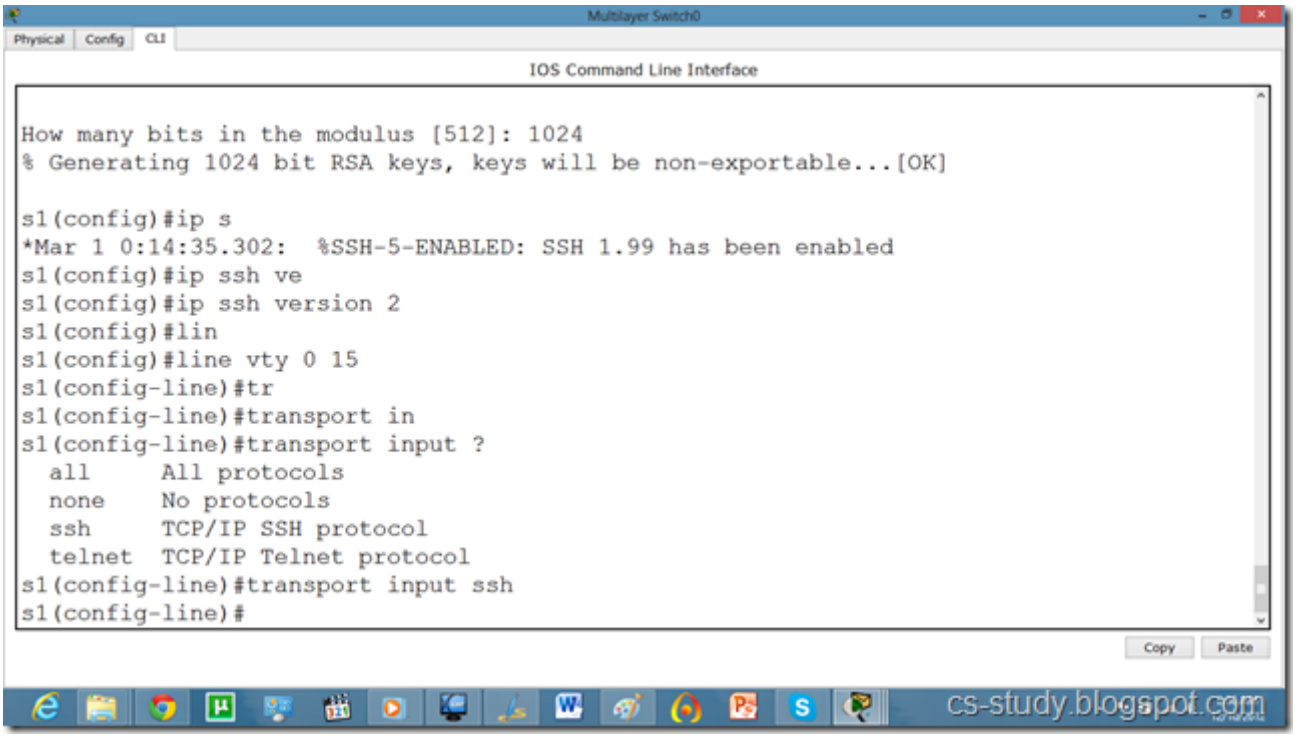
Now, we can go inside Switch configuration mode from our pc.



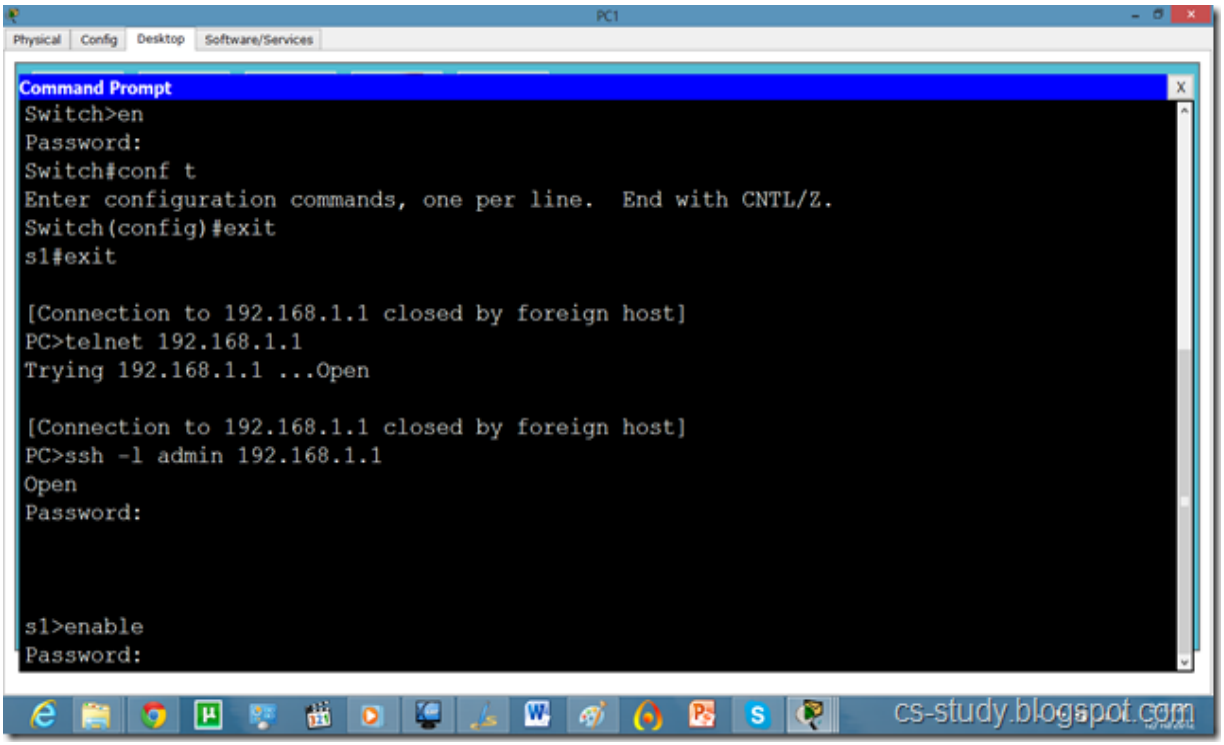
So, now let us apply SSH on the switch.



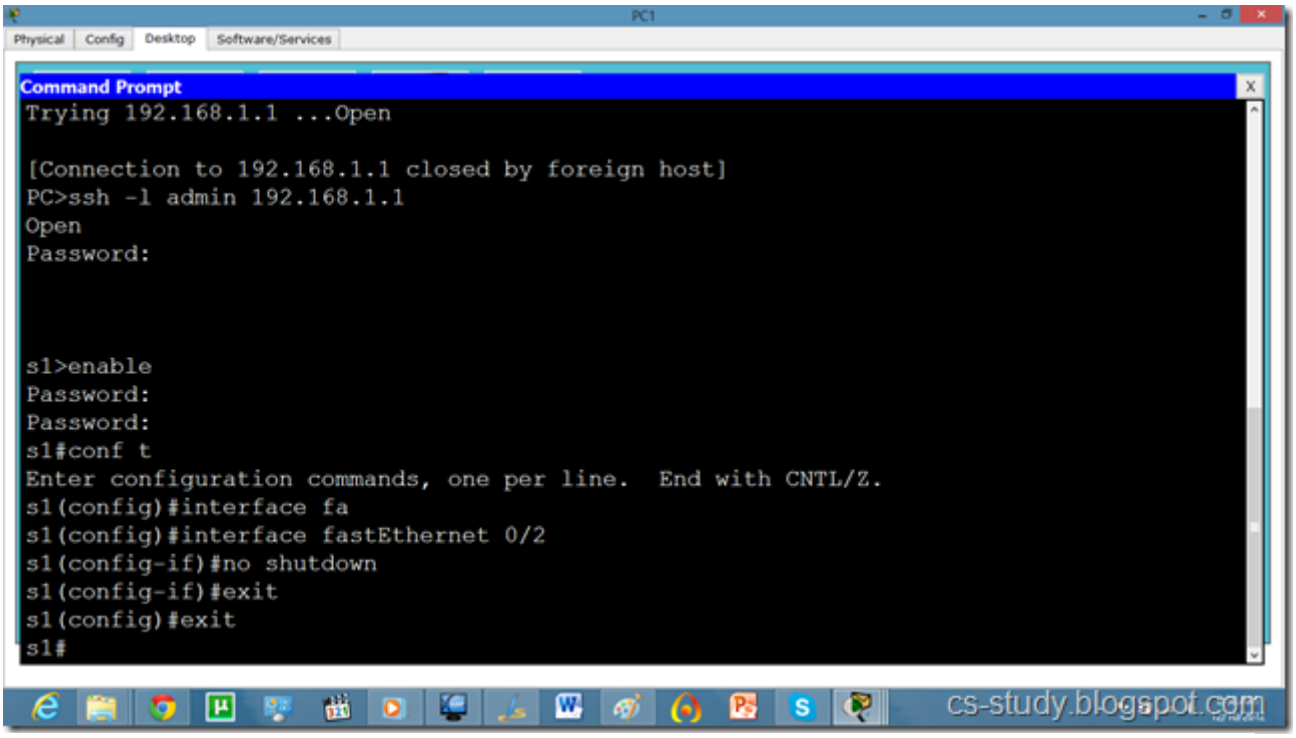
Commands continued.



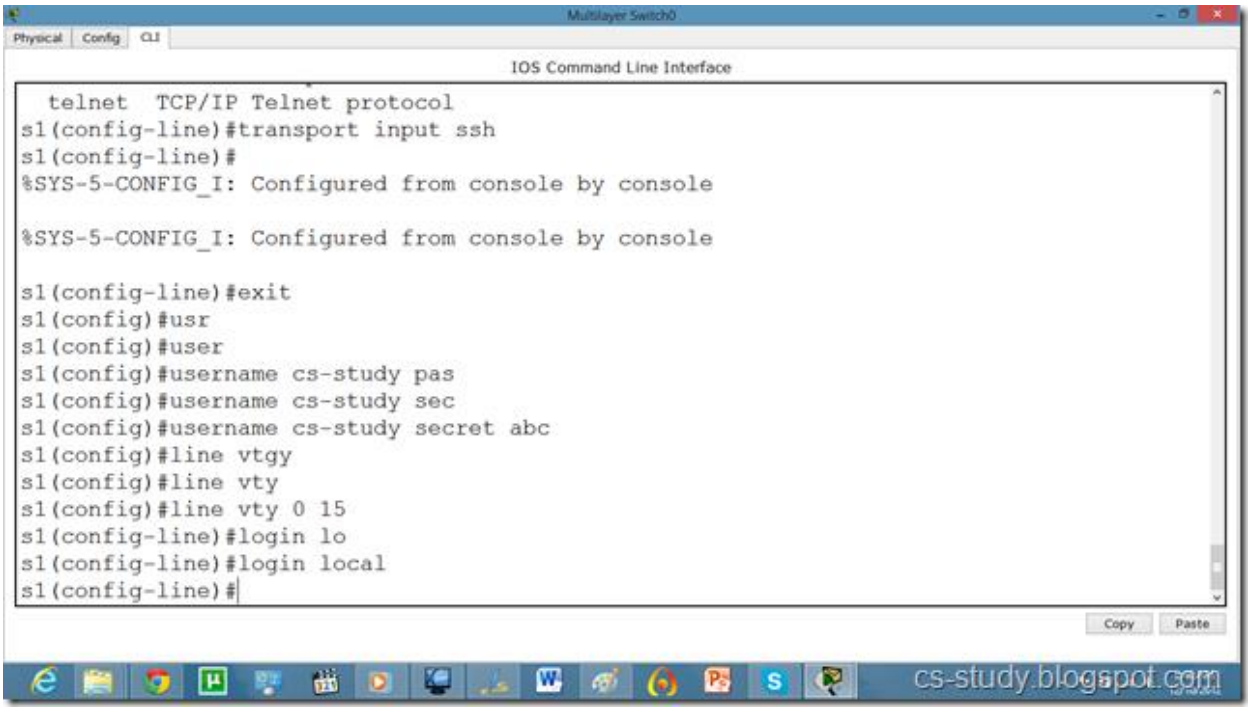
Now, we try to telnet it but it is refused because ssh has over ruled telnet. So, we will use SSH protocol on it. By default, username is admin.



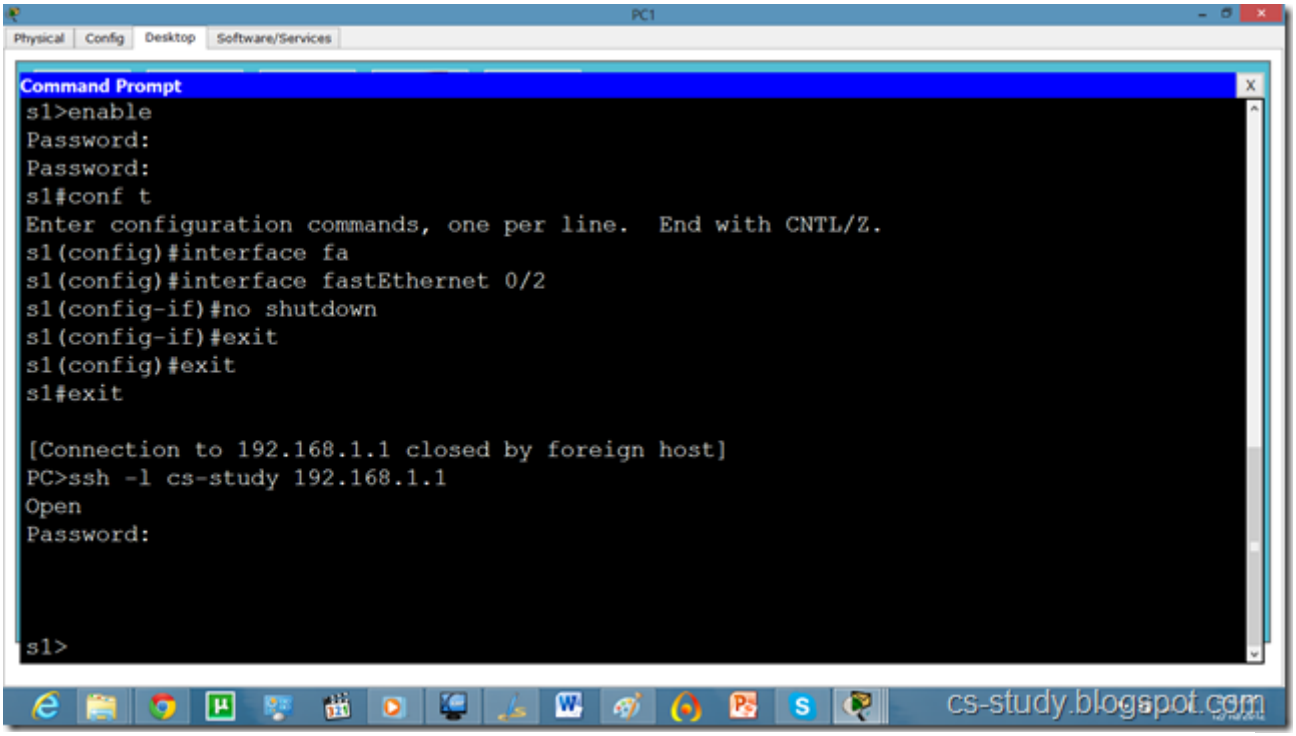
And we can apply any sort of configuration on our switch from out pc.



Now, if we want to change the username from admin to something else, we will do it as follows.



and from our pc as follows.



The SSH commands are as follows.

Switch(config)#ip domain name ?
WORD Default domain nameSwitch(config)#ip domain name abc.comSwitch(config)#crypto key generate rsa
% Please define a hostname other than Switch.
Switch(config)#hostname s1
s1(config)#ip domain name cs-studys1
s1(config)#crypto key generate rsa

Choose the size of the key modulus in the range of 360 to 2048 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.
s1(config)#line vty 0 15
s1(config-line)#transport input ssh

The name for the keys will be: s1.cs-study
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]
s1(config)#ip ssh version 2
s1(config-line)#

Task 3: Apply PAP and CHAP

The Point-to-Point Protocol (PPP) is a data link protocol commonly used in establishing a direct connection between two networking nodes. It can provide connection authentication, transmission encryption and compression. PPP is used over many types of physical networks including serial cable, phone line, trunk line, cellular telephone, specialized radio links, and fiber optic links etc. Internet service providers (ISPs) have used PPP for customer dial-up access to the Internet, since IP packets cannot be transmitted over a modem line on their own, without some data link protocol. PPP is commonly used as a data link layer protocol for connection over synchronous and asynchronous circuits, where it has largely superseded the older Serial Line Internet Protocol (SLIP) and telephone company mandated standards (such as Link Access Protocol, Balanced (LAPB).

PPP was designed somewhat after the original HDLC specifications. The designers of PPP included many additional features that had been seen only in proprietary data-link protocols up to that time.

HDLC

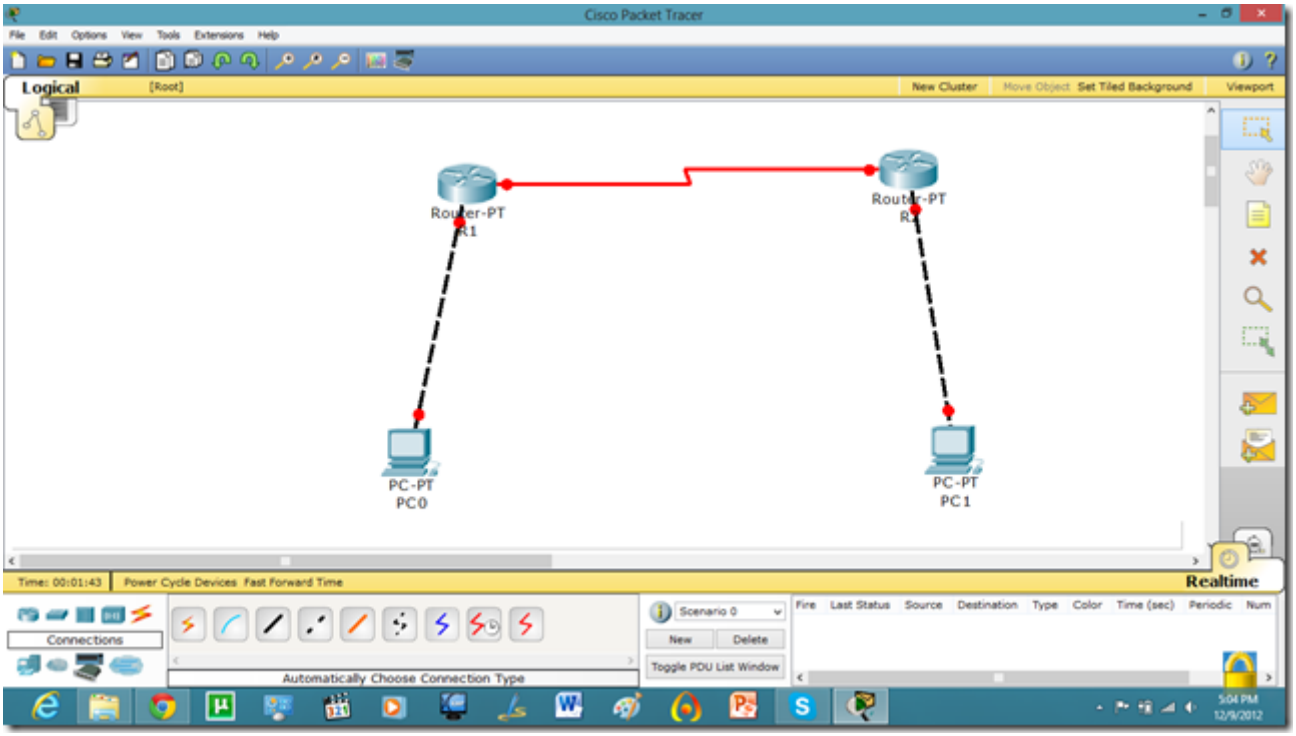
HDLC provides both connection-oriented and connectionless service. HDLC can be used for point to multipoint connections but is now used almost exclusively to connect one device to another, using what is known as Asynchronous Balanced Mode (ABM).

PAP

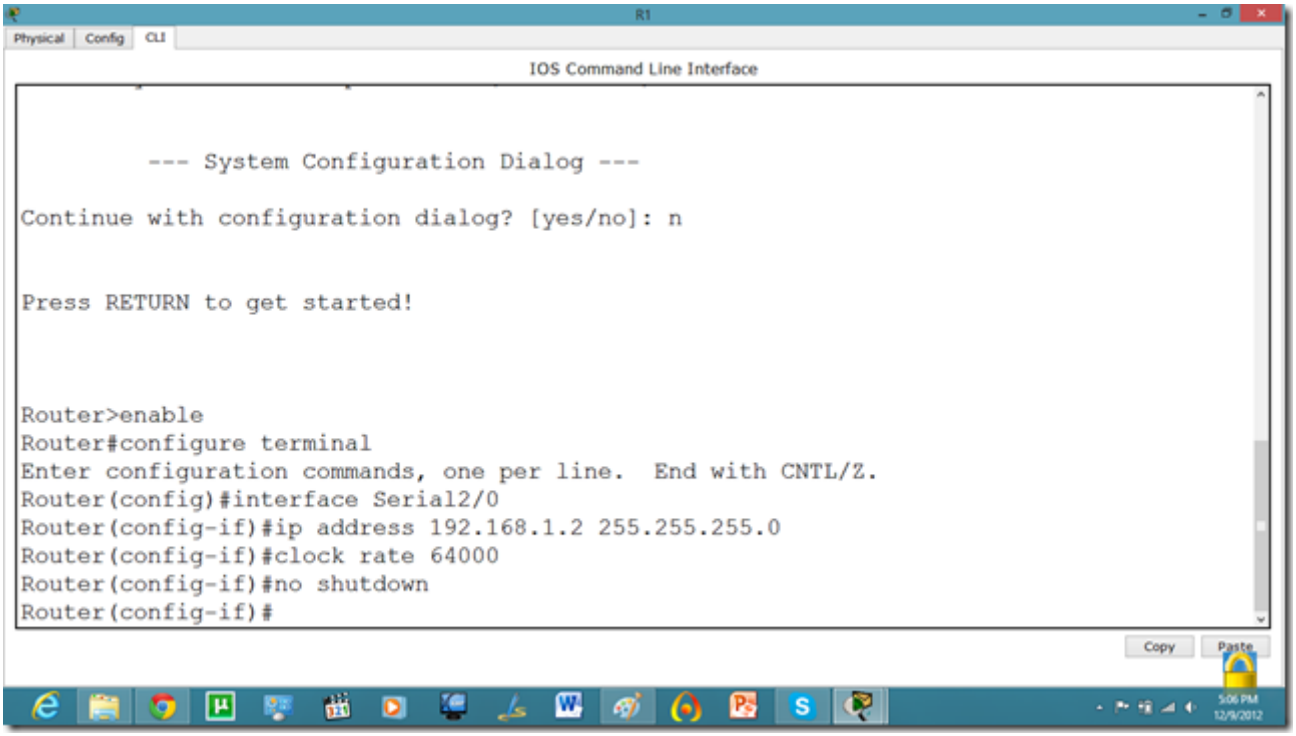
A **password authentication protocol** (**PAP**) is an authentication protocol that uses a password. PAP is used by Point to Point Protocol to validate users before allowing them access to server resources. Almost all network operating system remote servers support PAP. PAP transmits unencrypted ASCII passwords over the network and is therefore considered insecure. It is used as a last resort when the remote server does not support a stronger authentication protocol, like CHAP or EAP (the latter is actually a framework). **Password-based authentication** is the protocol that two entities share a password in advance and use the password as the basis of authentication. Existing password authentication schemes can be categorized into two types: weak-password authentication schemes and strong-password authentication schemes. In general, strong-password authentication protocols have the advantages over the weak-password authentication schemes in that their computational overhead are lighter, designs are simpler, and implementation are easier, and therefore are especially suitable for some constrained environments.

PAP works basically the same way as the normal login procedure. The client authenticates itself by sending a username and an (optionally encrypted) password to the server, which the server compares to its secrets database. This technique is vulnerable to eavesdroppers who may try to obtain the password by listening in on the serial line, and to repeated trial and error attacks.

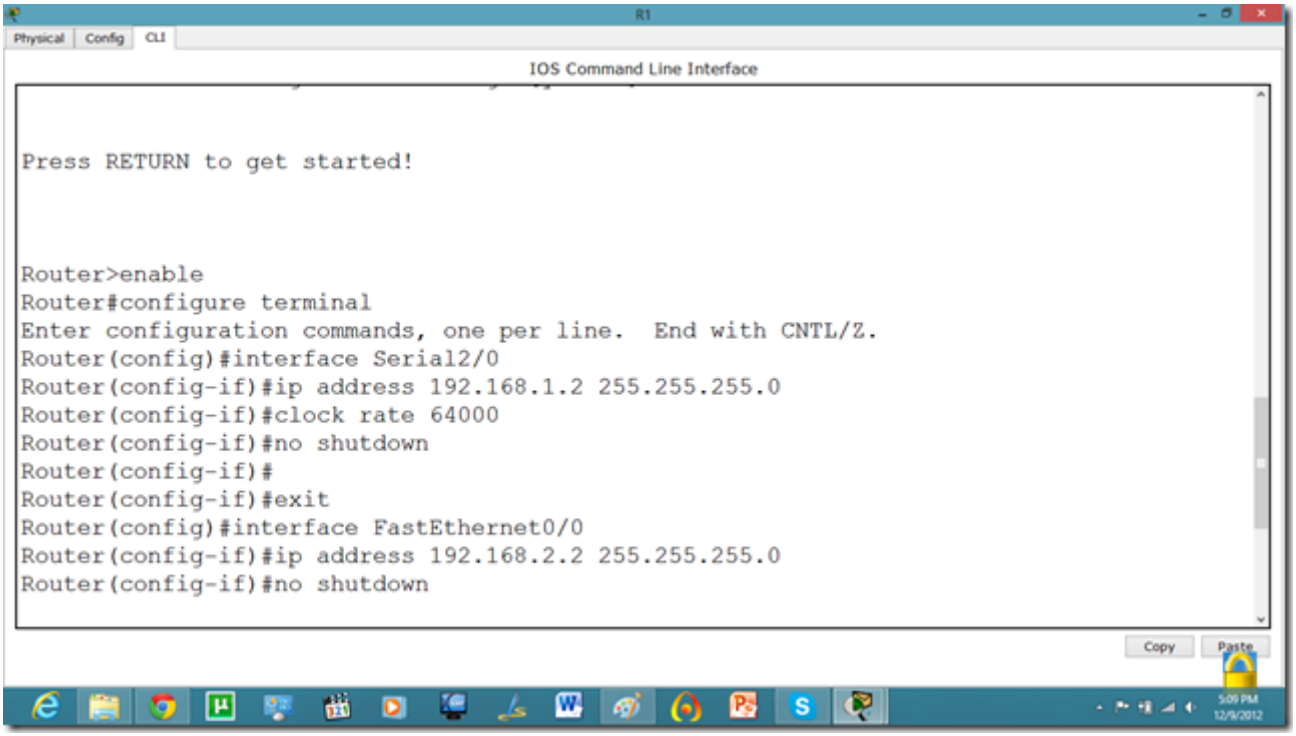
Let us apply PPP on packet tracer. Consider the following simpler topology.



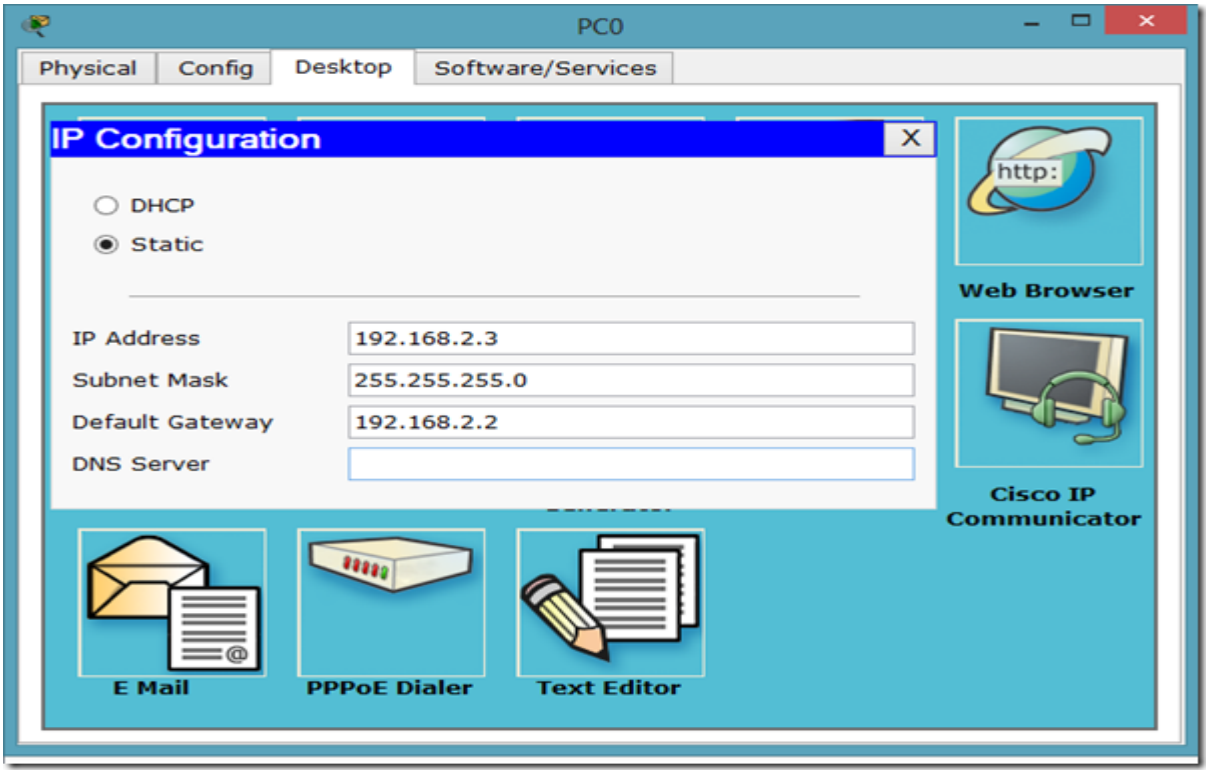
Let us apply IP addresses on the interfaces and change the state of the interface from down to UP. So that they can communicate.



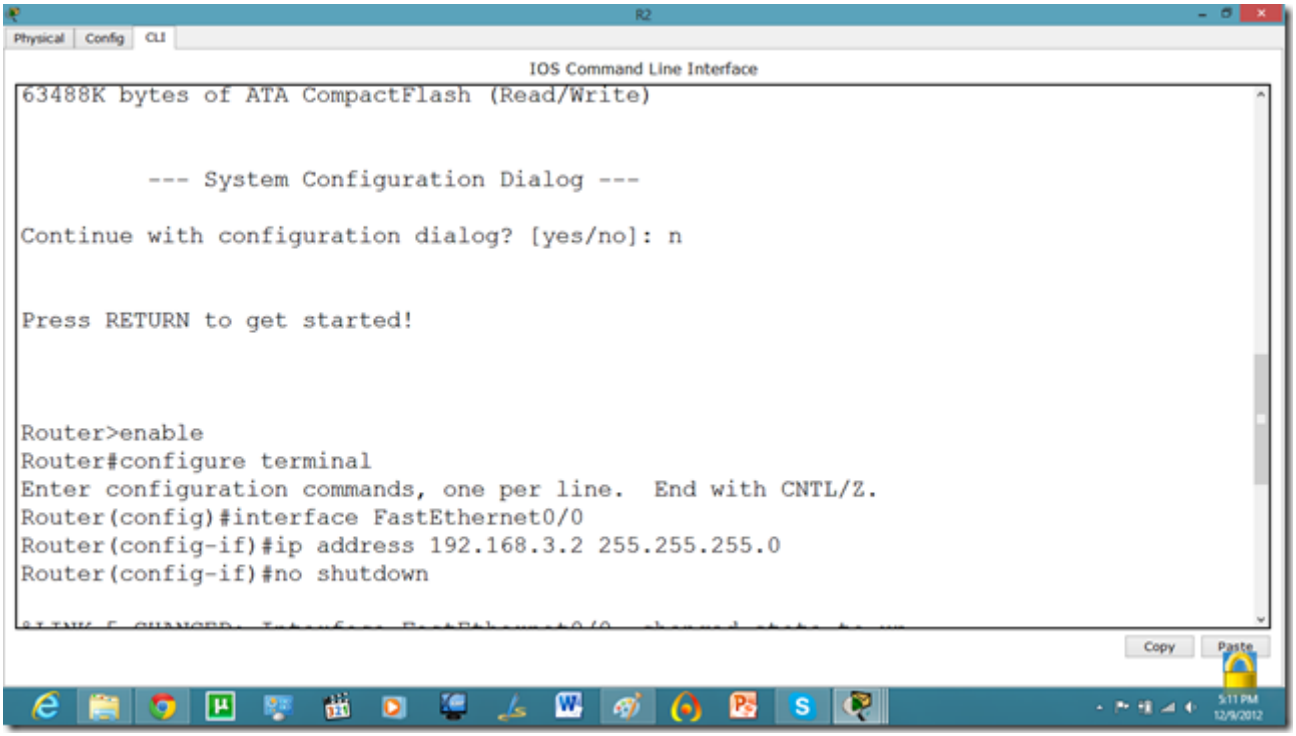
Similarly, for serial interface.



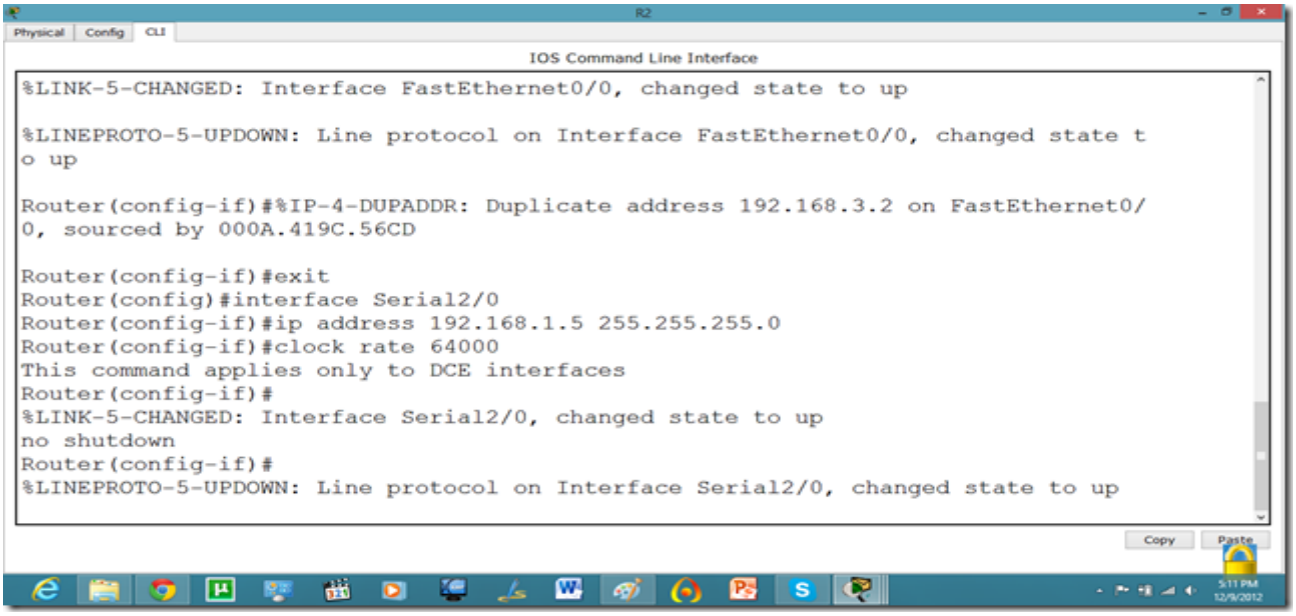
PC IP setup



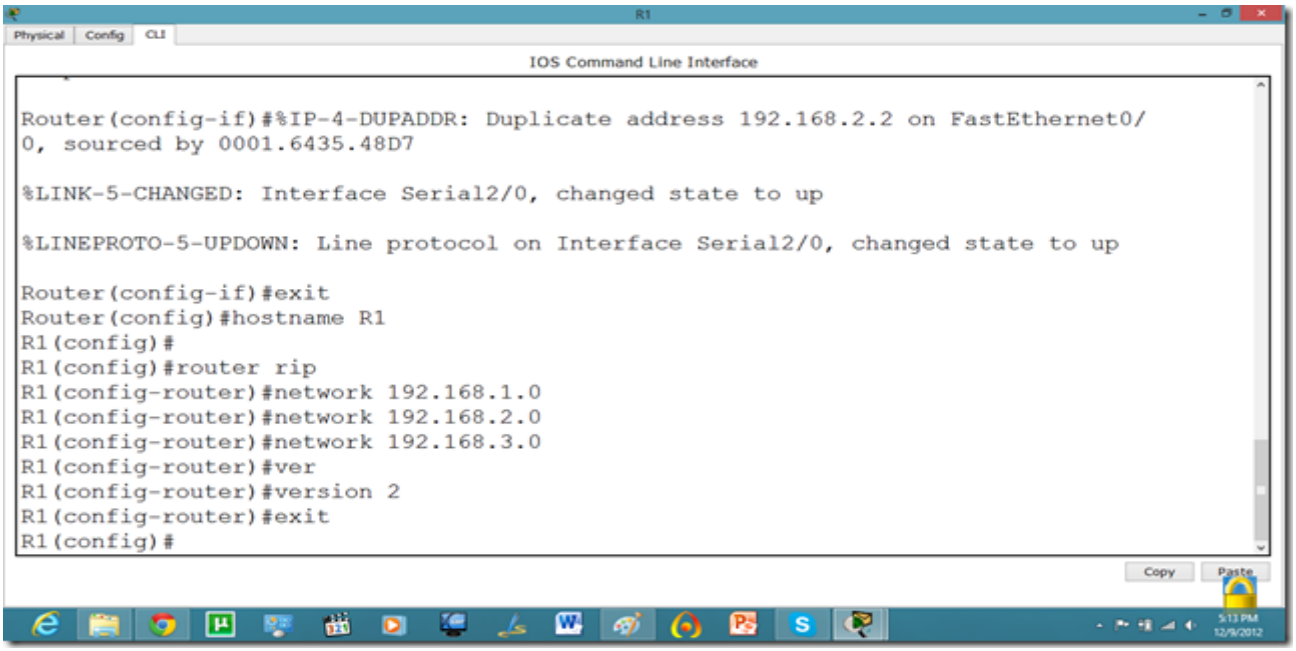
The IP configuration on other router.



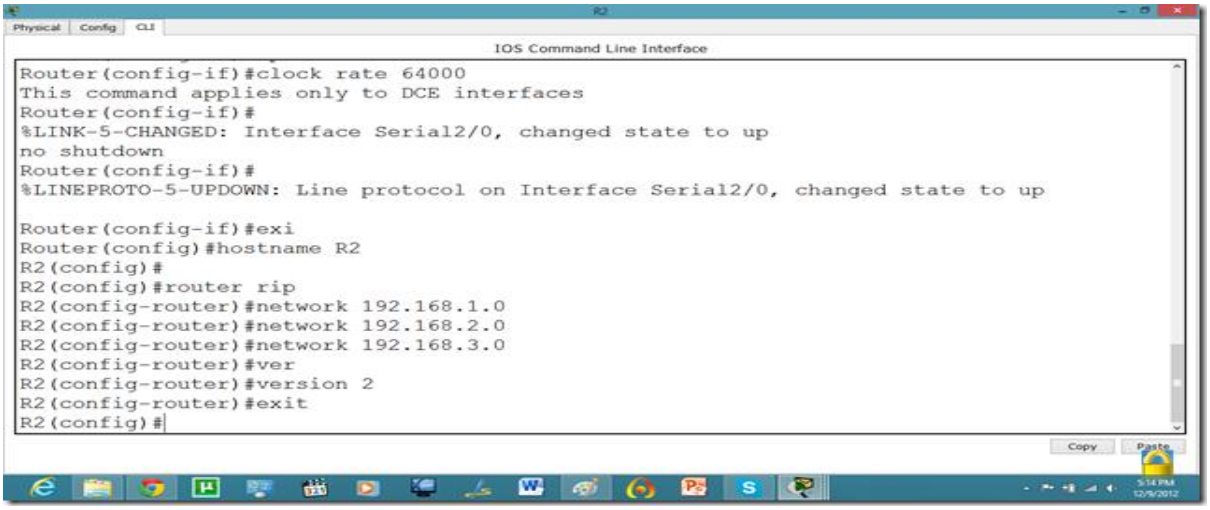
serial int setup.



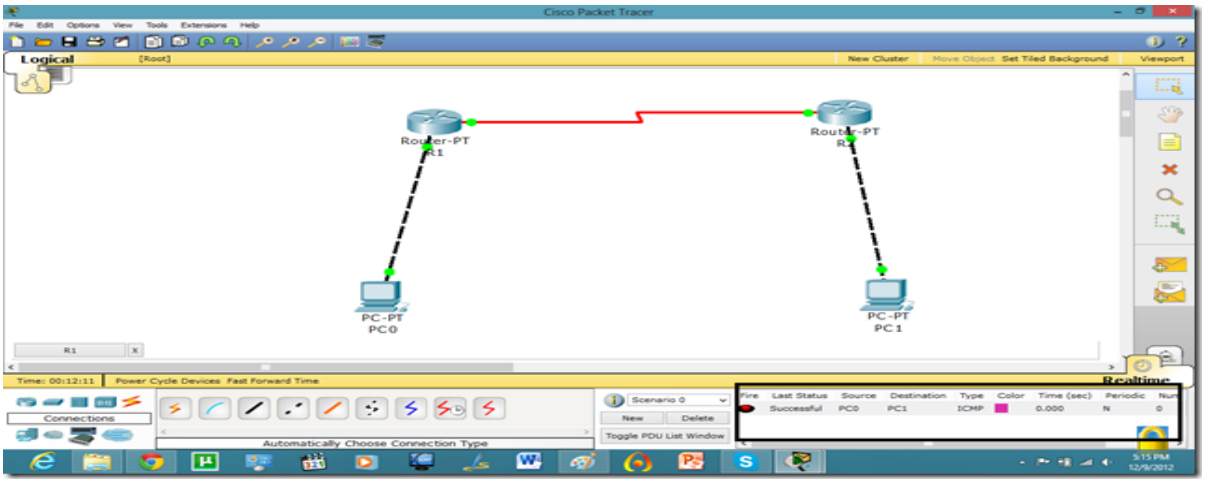
Now, we know that PCs that are attached cannot communicate until we apply a routing mechanism. In this case we are applying the RIP V2 protocol. Apply the following set of commands on both routers. We have also set the hostname of the router which will be useful to us later.



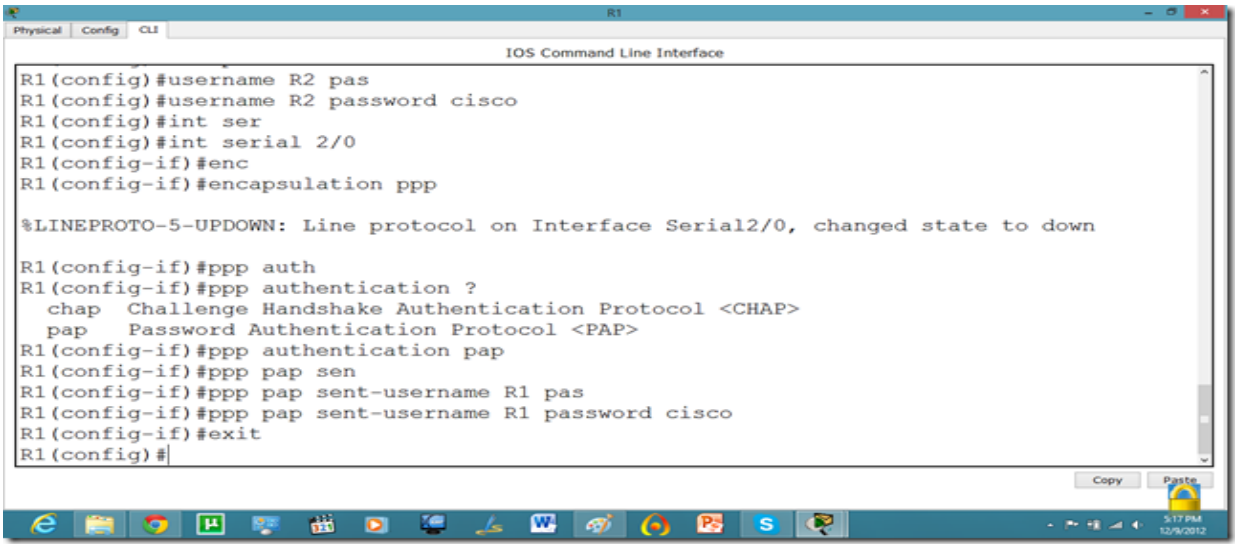
Now, let us set the commands on the second router as well.



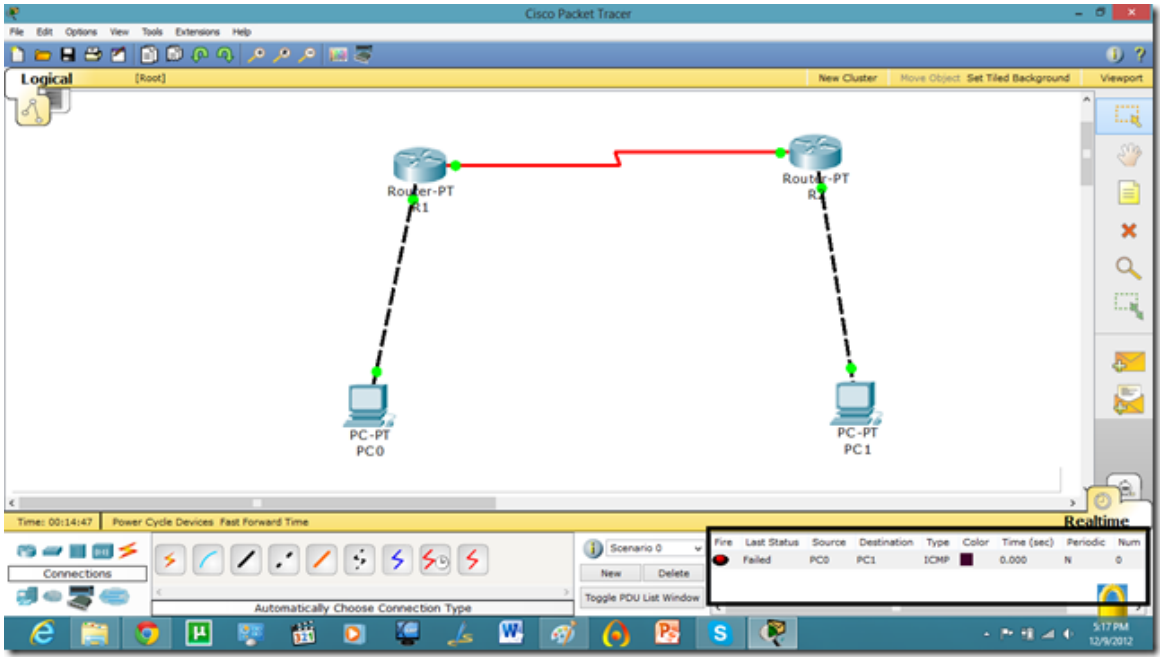
Now, both PCs can communicate.



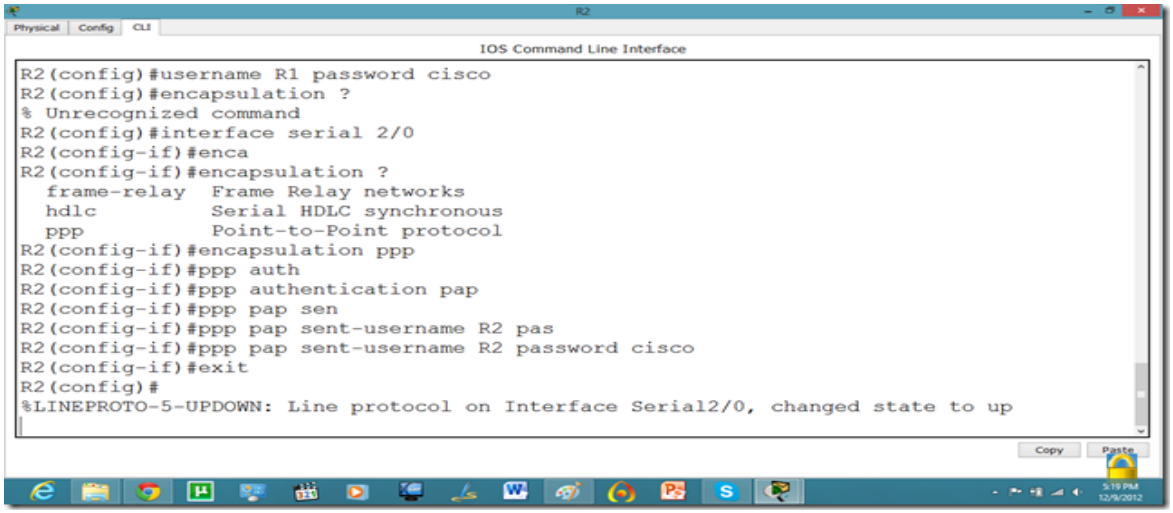
Now, we will set the authentication, In this tutorial we are going to apply PAP.



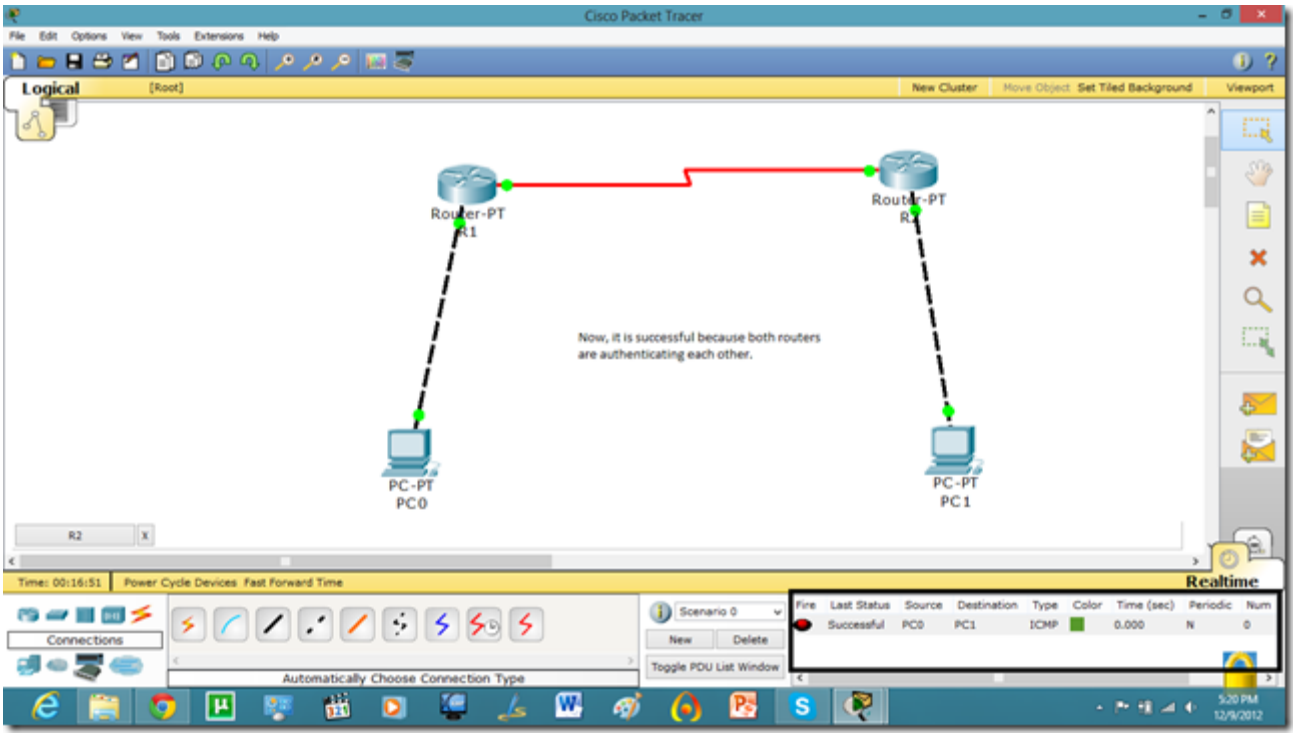
As we set the authentication on one router the communication is disabled.



Let us set it on other router as well.



Now, they can communicate.



Now, if we run show run command in enable mode. We can see the authentication enabled in router.

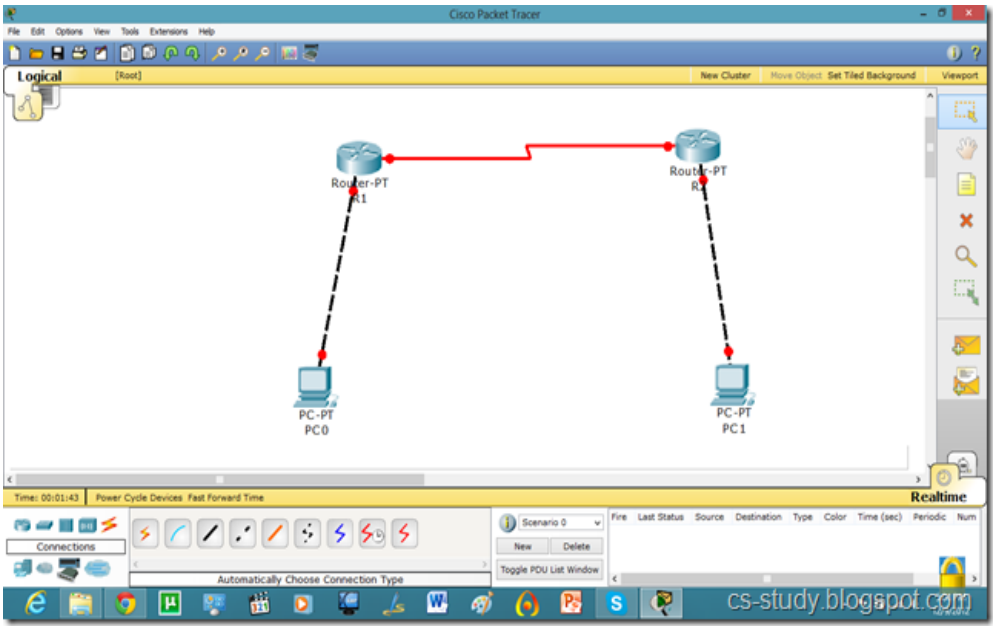
CHAP Working

CHAP is an authentication scheme used by Point to Point Protocol (PPP) servers to validate the identity of remote clients. CHAP periodically verifies the identity of the client by using a three-way handshake. This happens at the time of establishing the initial link (LCP), and may happen again at any time afterwards. The verification is based on a shared secret (such as the client user's password).

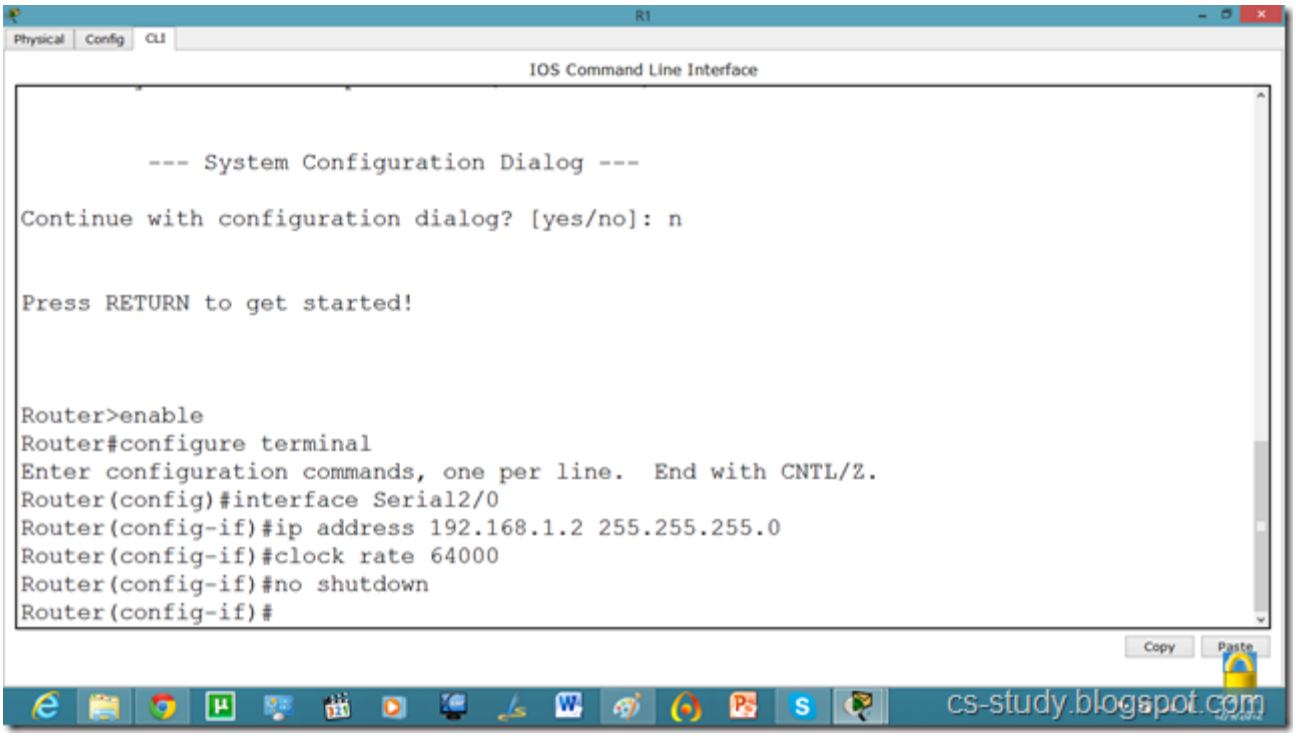
1. After the completion of the link establishment phase, the authenticator sends a "challenge" message to the peer.
2. The peer responds with a value calculated using a one-way hash function on the challenge and the secret combined.
3. The authenticator checks the response against its own calculation of the expected hash value. If the values match, the authenticator acknowledges the authentication; otherwise it should terminate the connection.
4. At random intervals the authenticator sends a new challenge to the peer and repeats steps 1 through 3.

Another feature of CHAP is that it doesn't only require the client to authenticate itself at startup time, but sends challenges at regular intervals to make sure the client hasn't been replaced by an intruder, for instance by just switching phone lines.

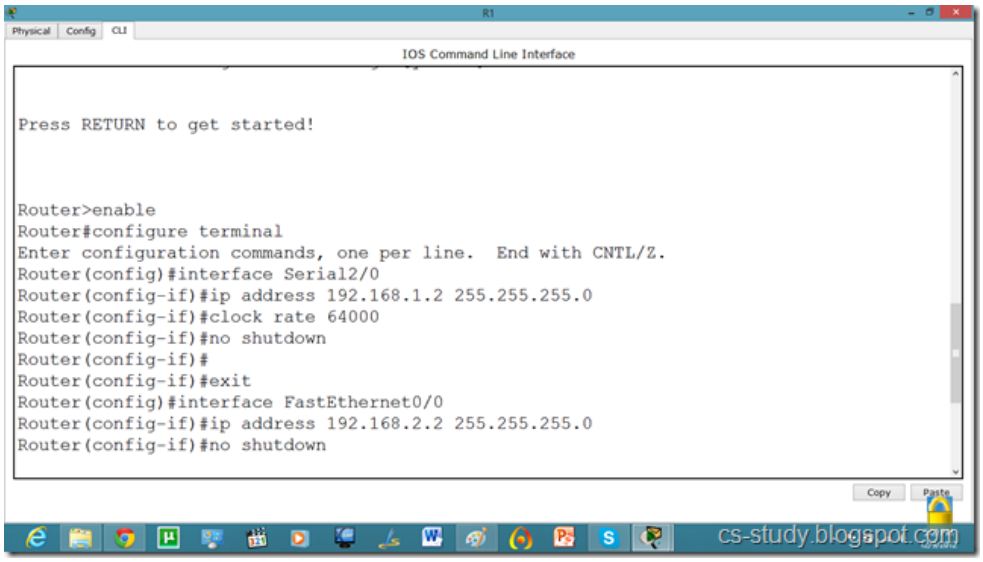
Let us apply PPP on packet tracer. Consider the following simpler topology.



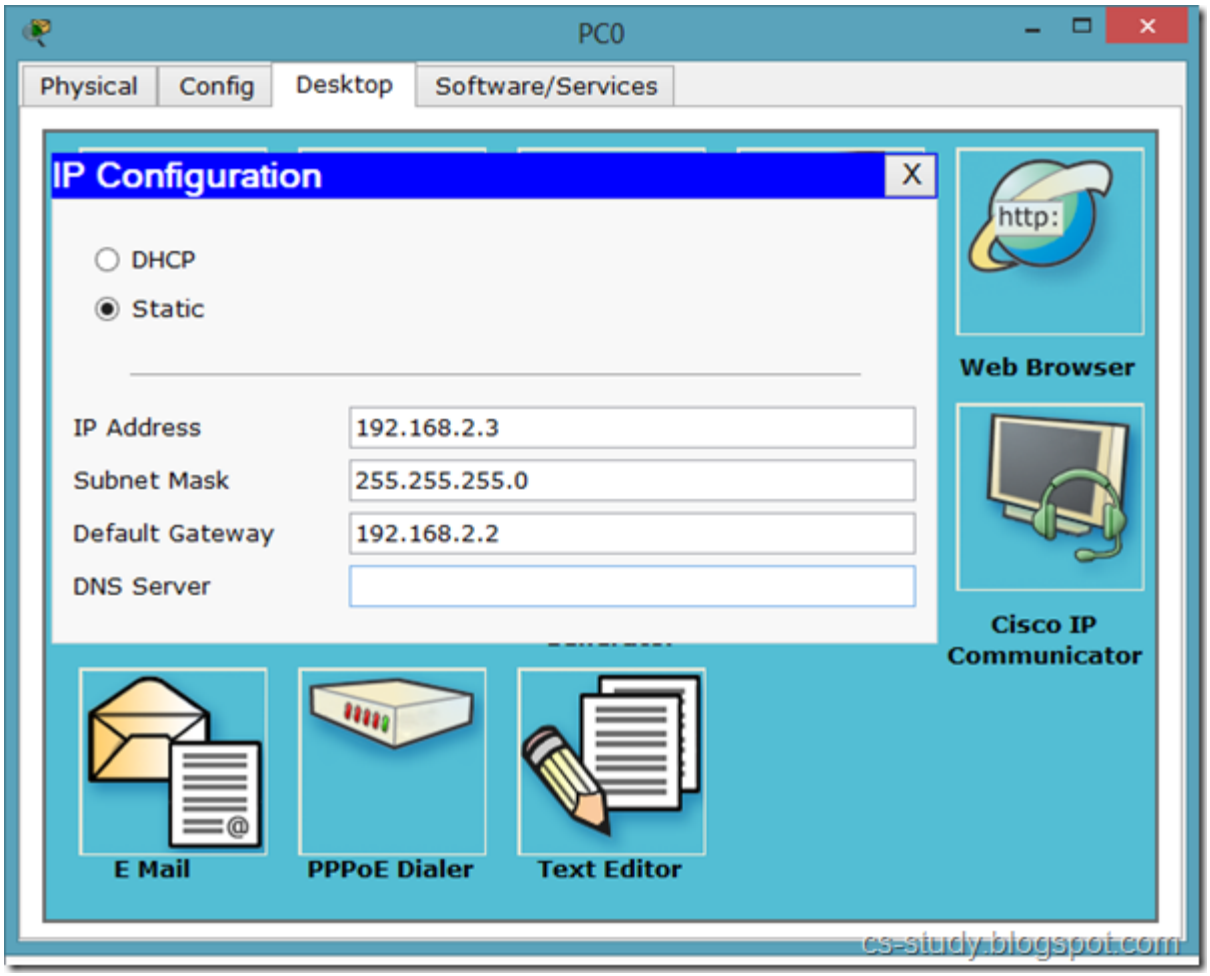
Let us apply IP addresses on the interfaces and change the state of the interface from down to UP. So that they can communicate.



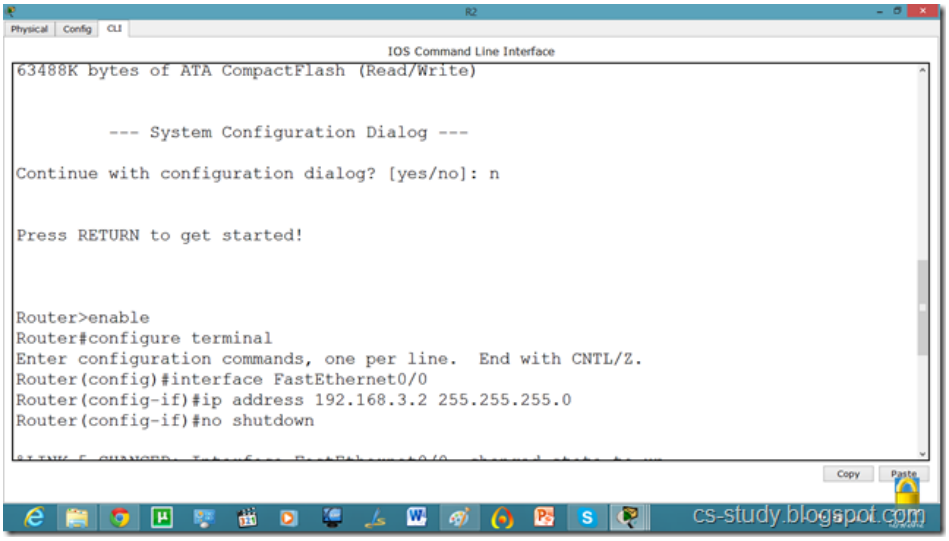
Similarly, for serial interface.



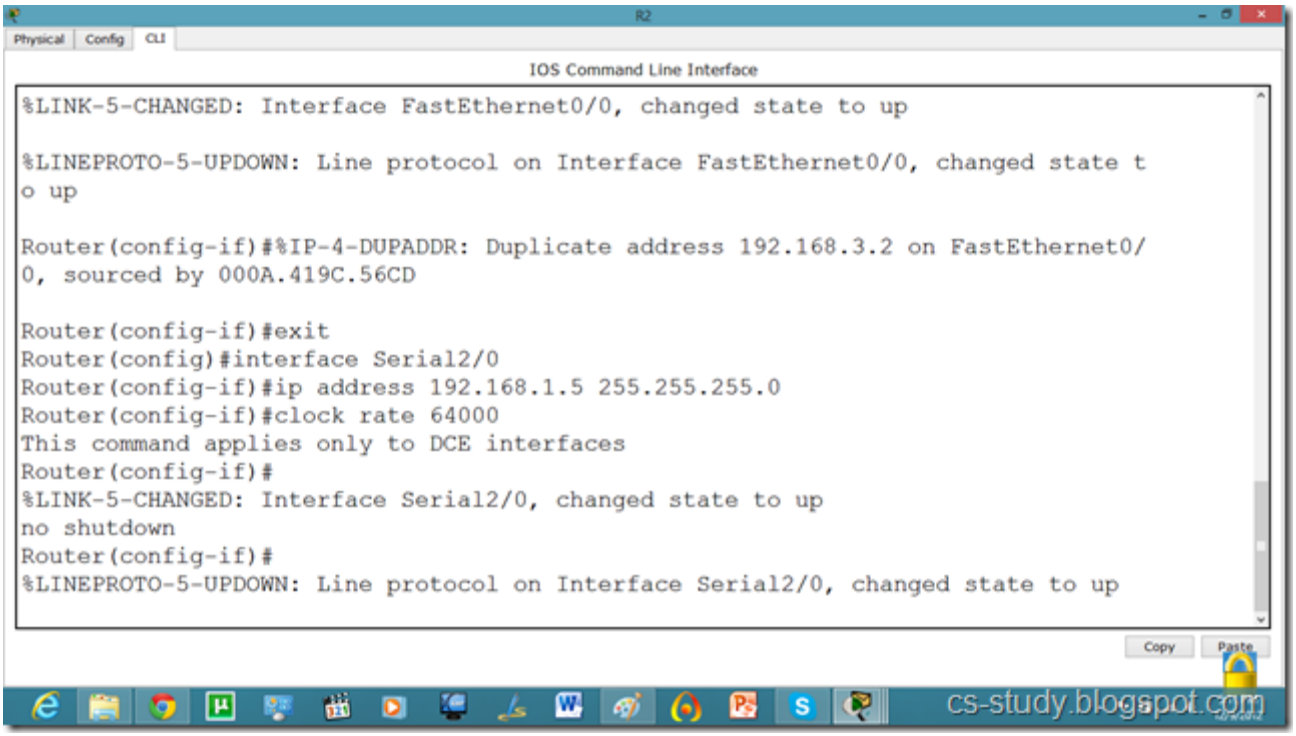
And IP configuration on PC.



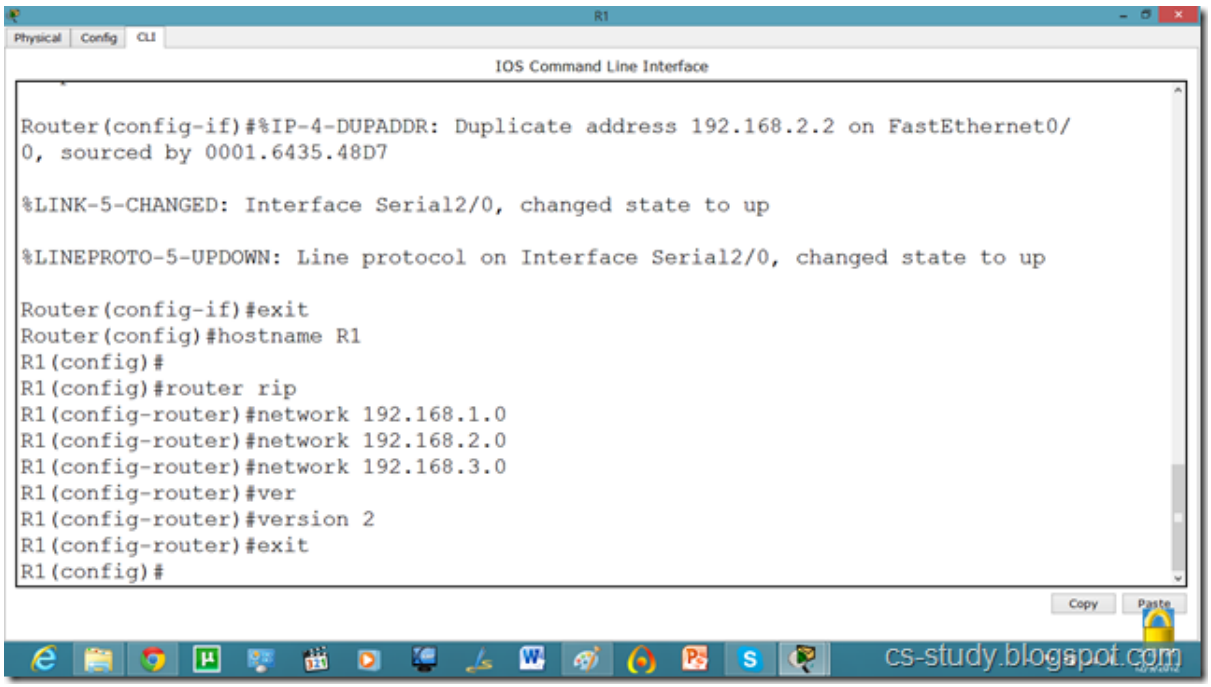
The IP configuration on other router.



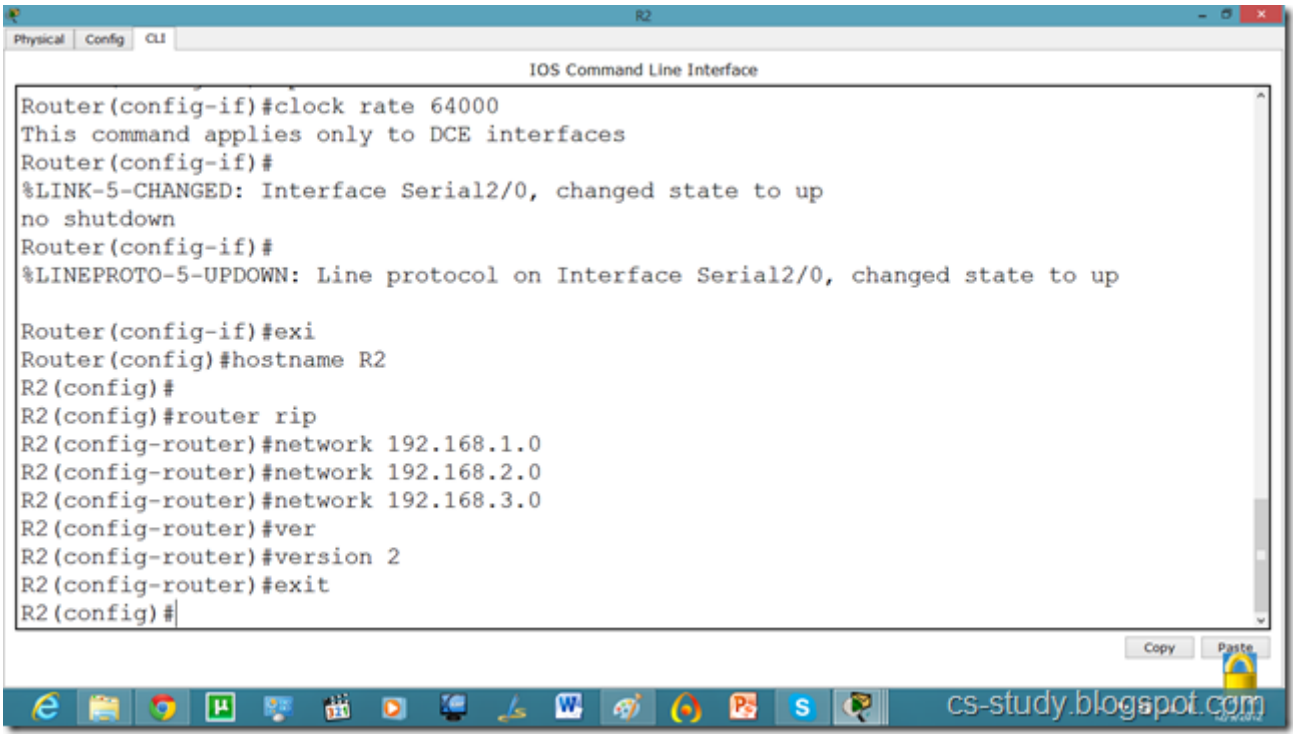
Serial Interface setting.



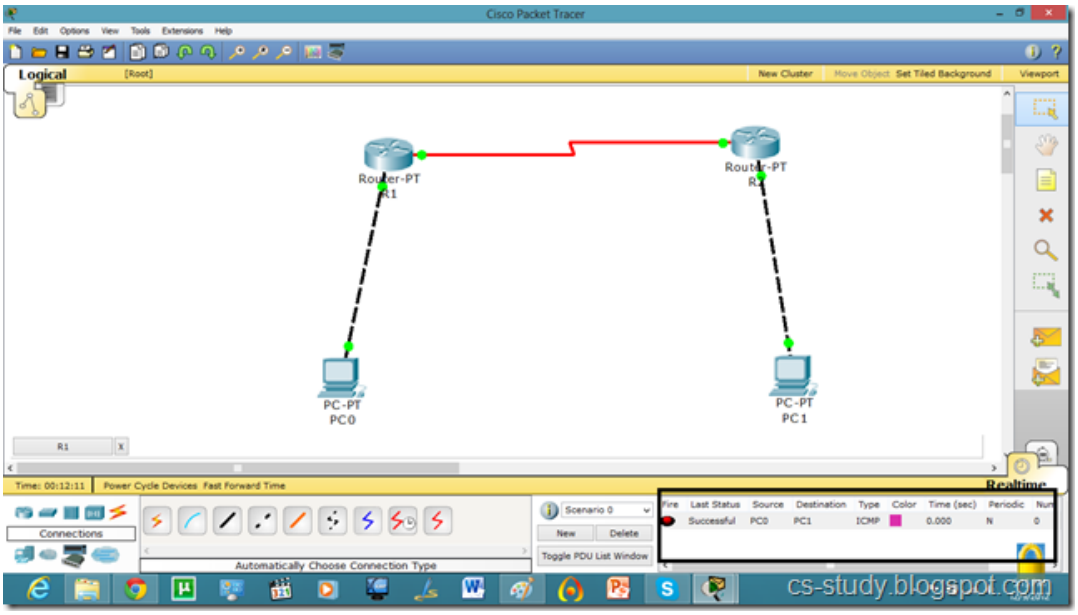
Now, we know that PCs that are attached cannot communicate until we apply a routing mechanism. In this case we are applying the RIP V2 protocol. Apply the following set of commands on both routers. We have also set the hostname of the router which will be useful to us later.



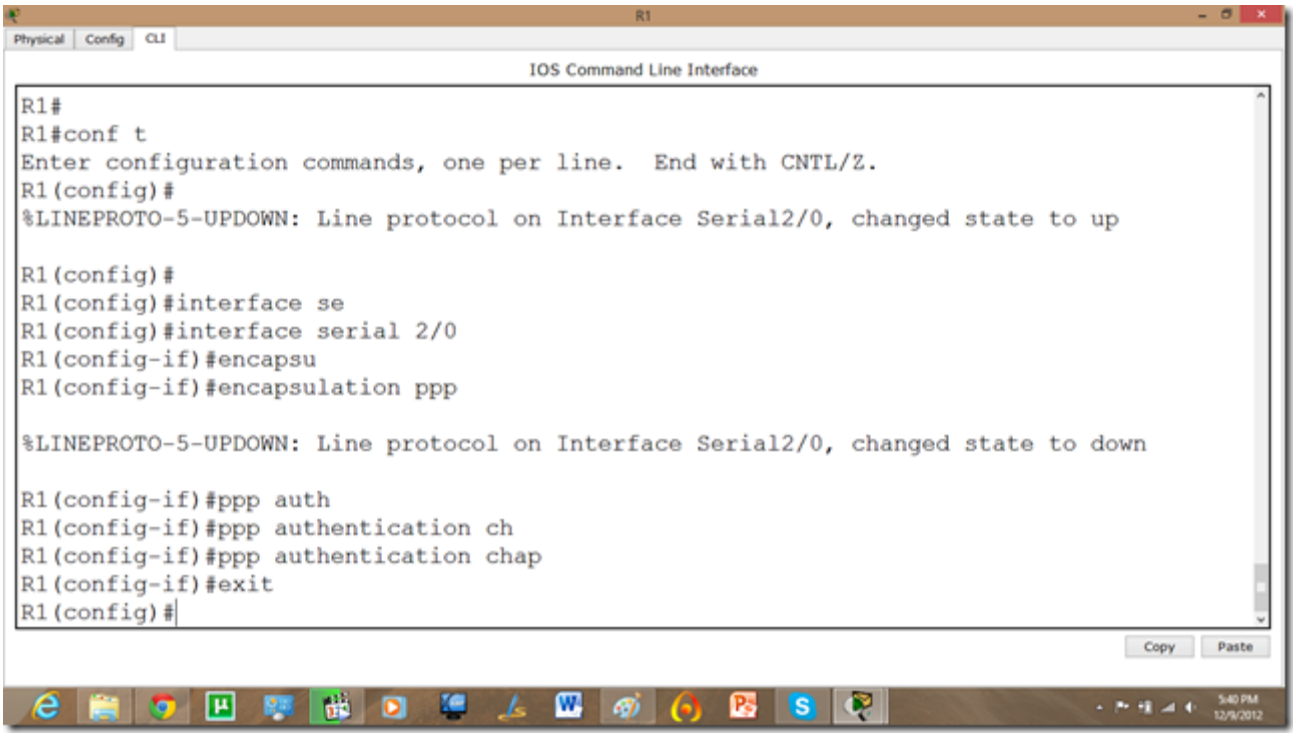
Now, let us set the commands on the second router as well.



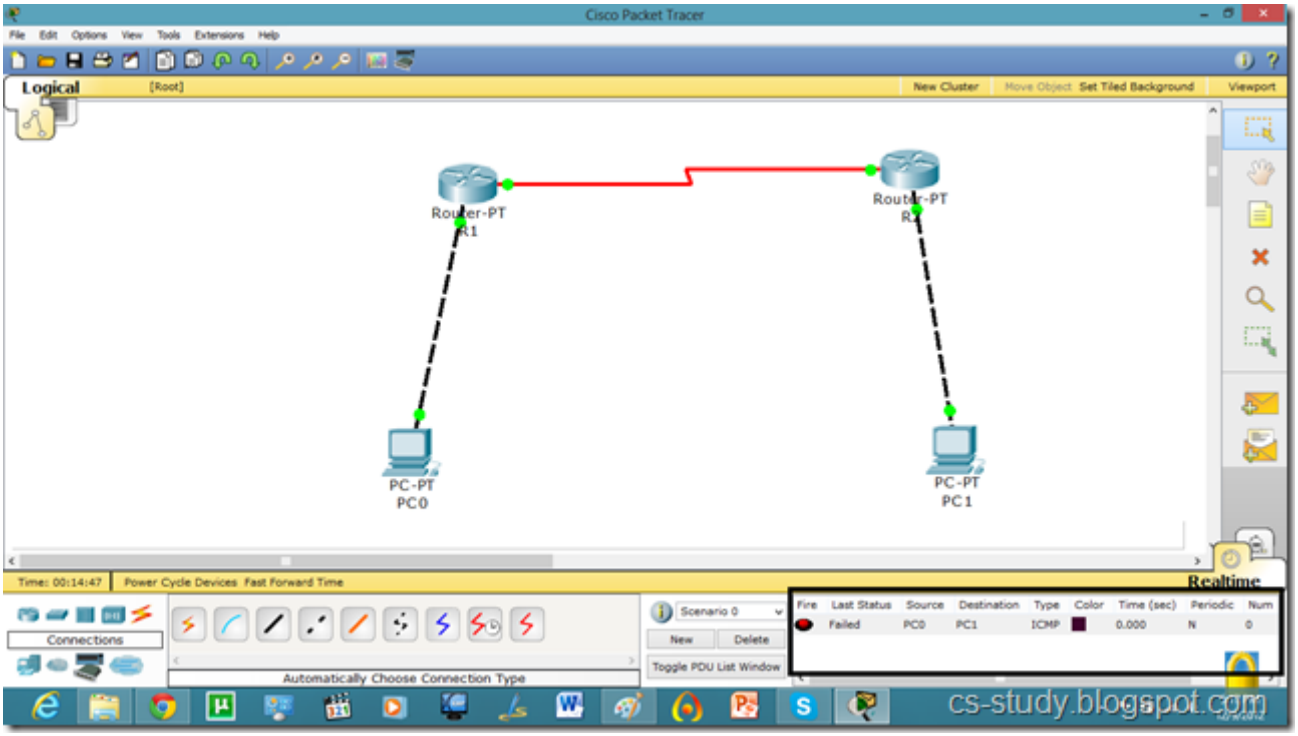
Now, both PCs can communicate.



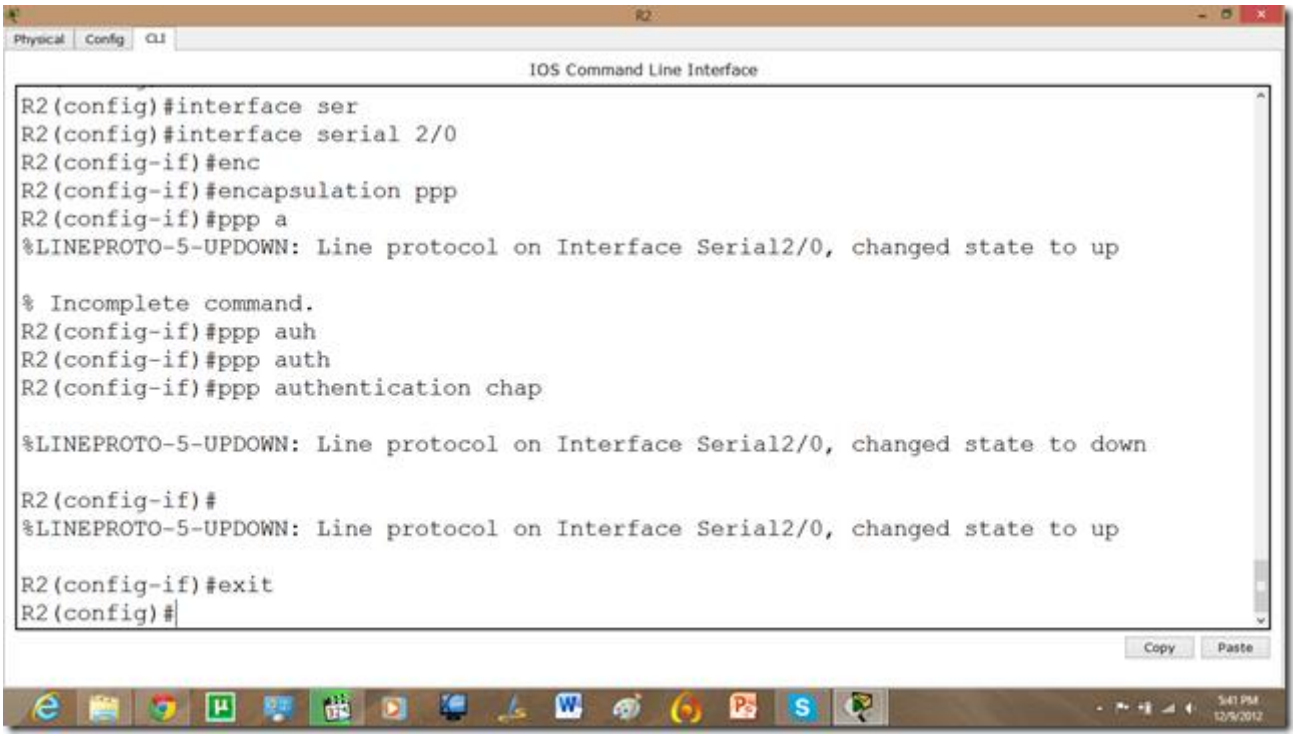
Now, we will set the authentication, In this tutorial we are going to apply CHAP(Challenge Handshake Authentication Protocol).



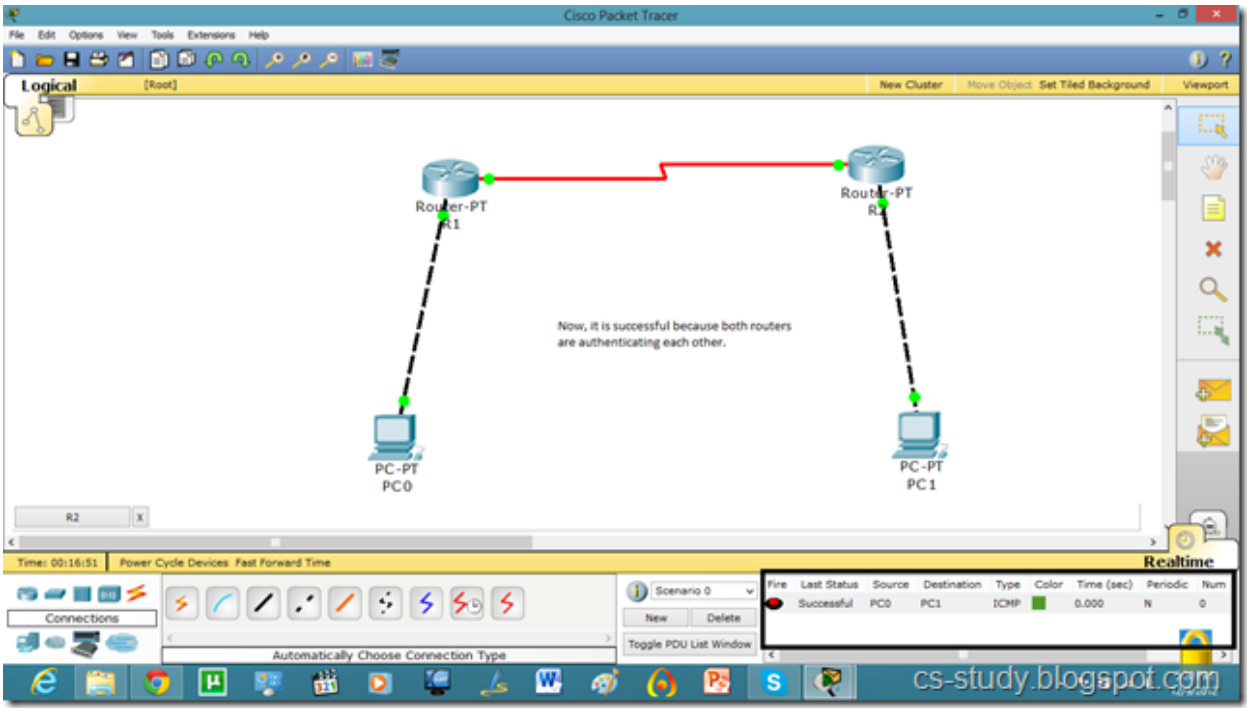
As we set the authentication on one router the communication is disabled.



Let us set it on other router as well.



Now, the communication is enabled.



Lab 11 & 12 – VOIP, Wireless, ACL on Packet Tracer

Objectives

In this lab, students will apply following on packet tracer

- Voice Over IP
- Wireless Communication
- Access Control List

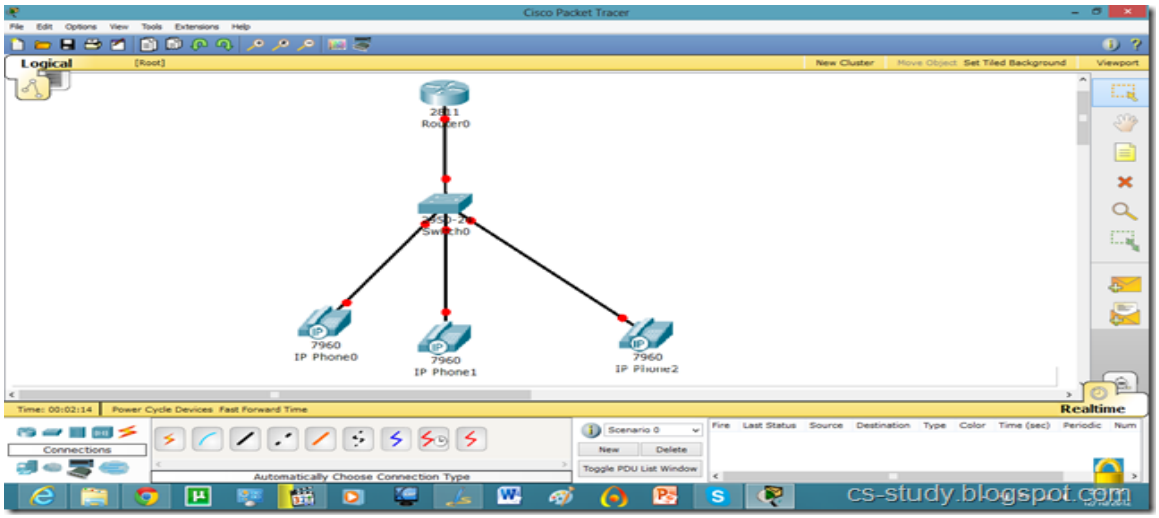
Lab Tasks

Task 1: Voice over IP

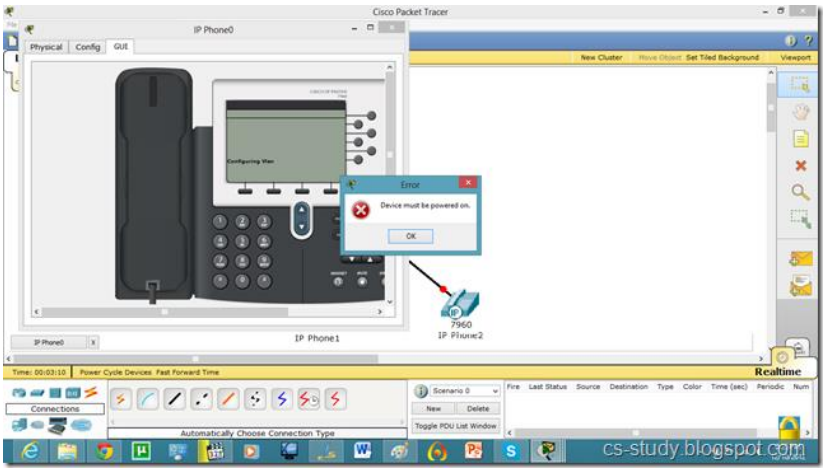
Voice over IP (VoIP, or voice over Internet Protocol) commonly refers to the communication protocols, technologies, methodologies, and transmission techniques involved in the delivery of voice communications and multimedia sessions over Internet Protocol (IP) networks, such as the Internet. Other terms commonly associated with VoIP are IP telephony,Internet telephony, voice over broadband (VoBB), broadband telephony, IP communications, and broadband phone.

Internet telephony refers to communications services —voice, fax, SMS, and/or voice-messaging applications— that are transported via the Internet, rather than the public switched telephone network (PSTN). The steps involved in originating a VoIP telephone call are signaling and media channel setup, digitization of the analog voice signal, encoding, packetization, and transmission as Internet Protocol (IP) packets over a packet-switched network. On the receiving side, similar steps (usually in the reverse order) such as reception of the IP packets, decoding of the packets and digital-to-analog conversion reproduce the original voice stream. Even though IP telephony and VoIP are used interchangeably, IP telephony refers to all use of IP protocols for voice communication by digital telephony systems, while VoIP is one technology used by IP telephony to transport phone calls

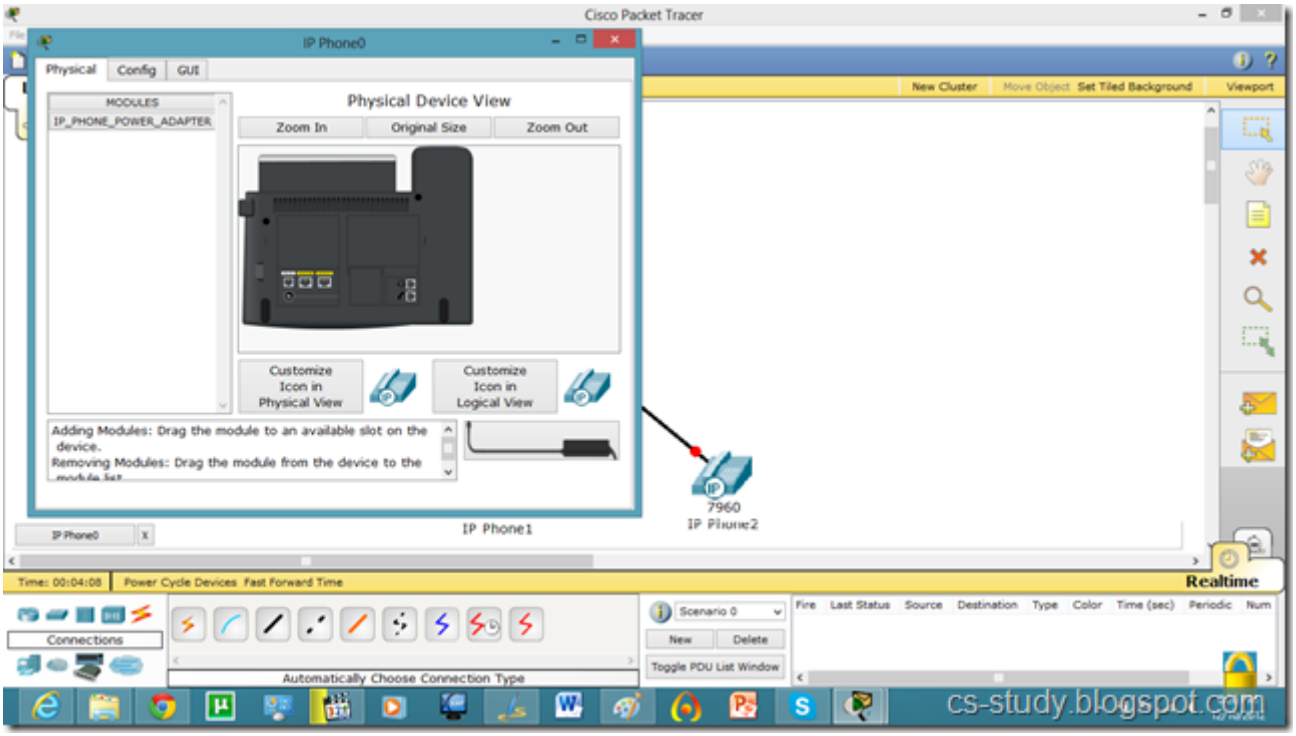
Let us apply it on packet tracer.



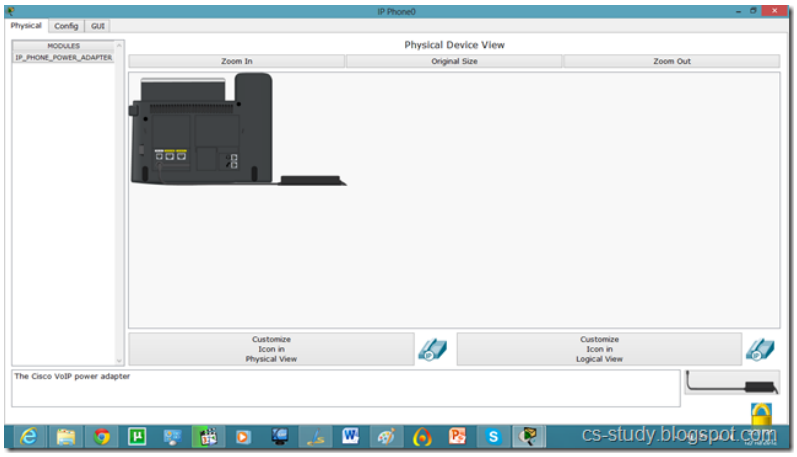
This IP Phone is displayed below. And when we try to go to any other mode its give us error and ask us to switch it on first.



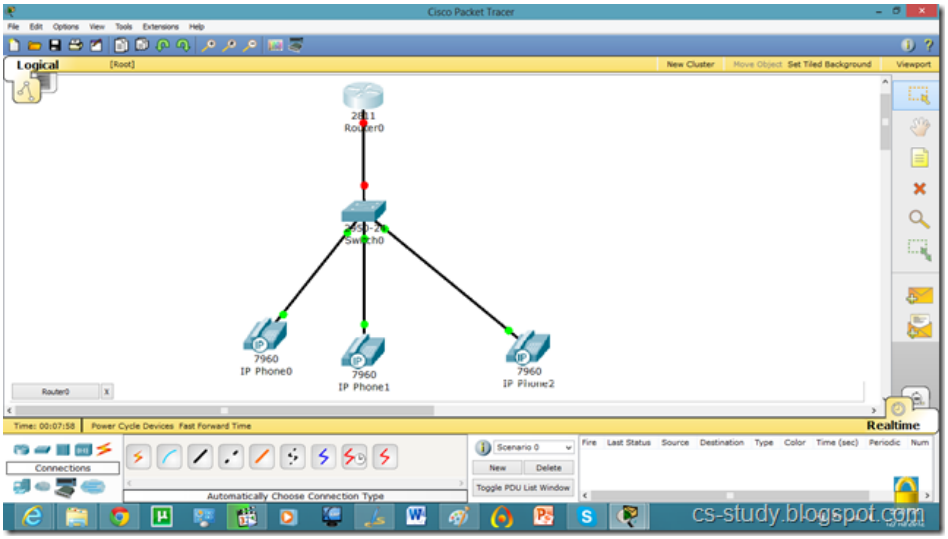
For that, go to Physical mode and put the power adapter (in the bottom right corner) into the phone as shown in figure.



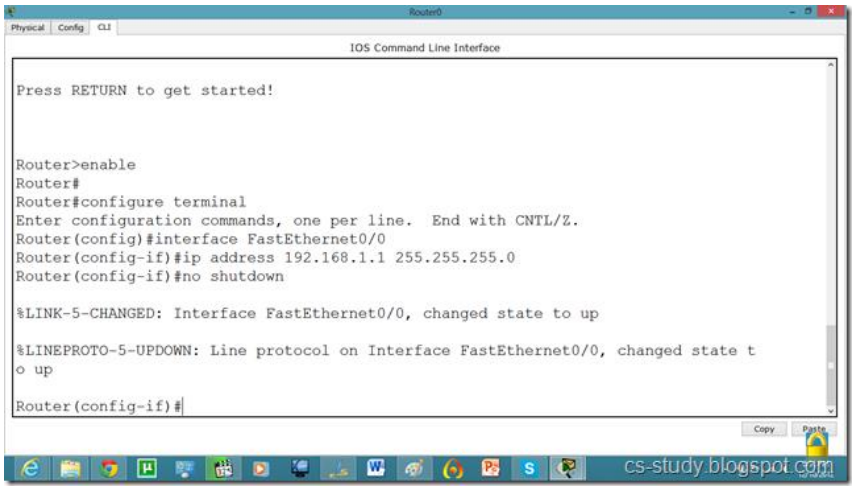
Now, we have inserted the antenna into IP phone. Repeat it for other phones as well.



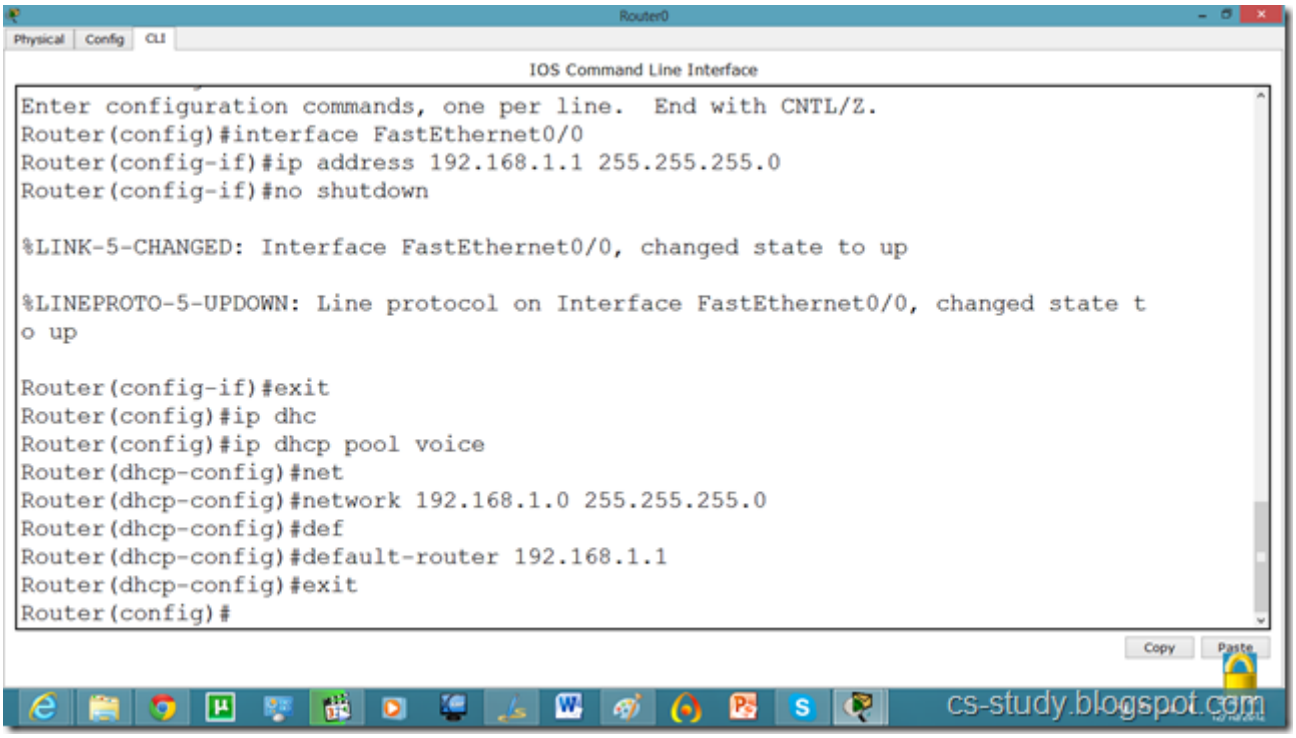
Now, we see that interface of IP phone is UP.



Now, go to router and assign IP address.

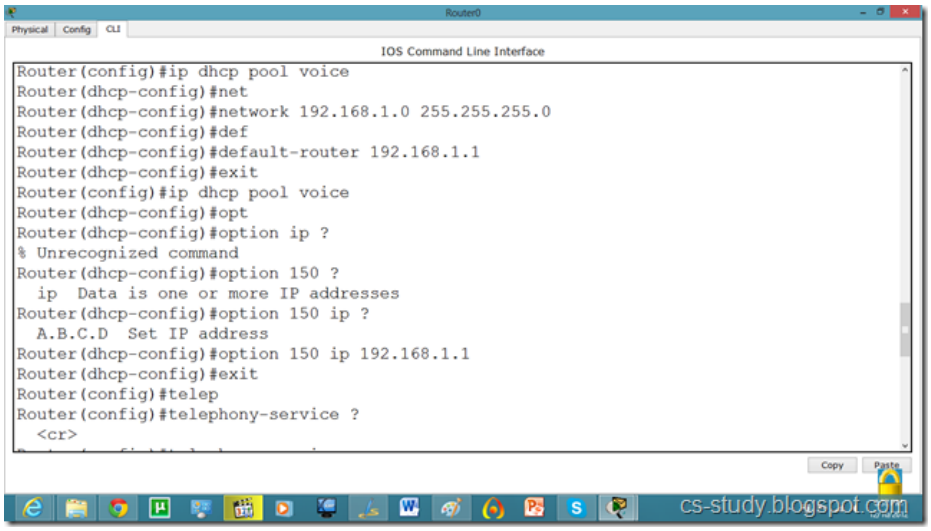


We will have to set DHCP server on router to assign IP addresses to IP phones.



We will have to give an additional command for voip.

Router(dhcp-config)#option 150 ip 192.168.1.1



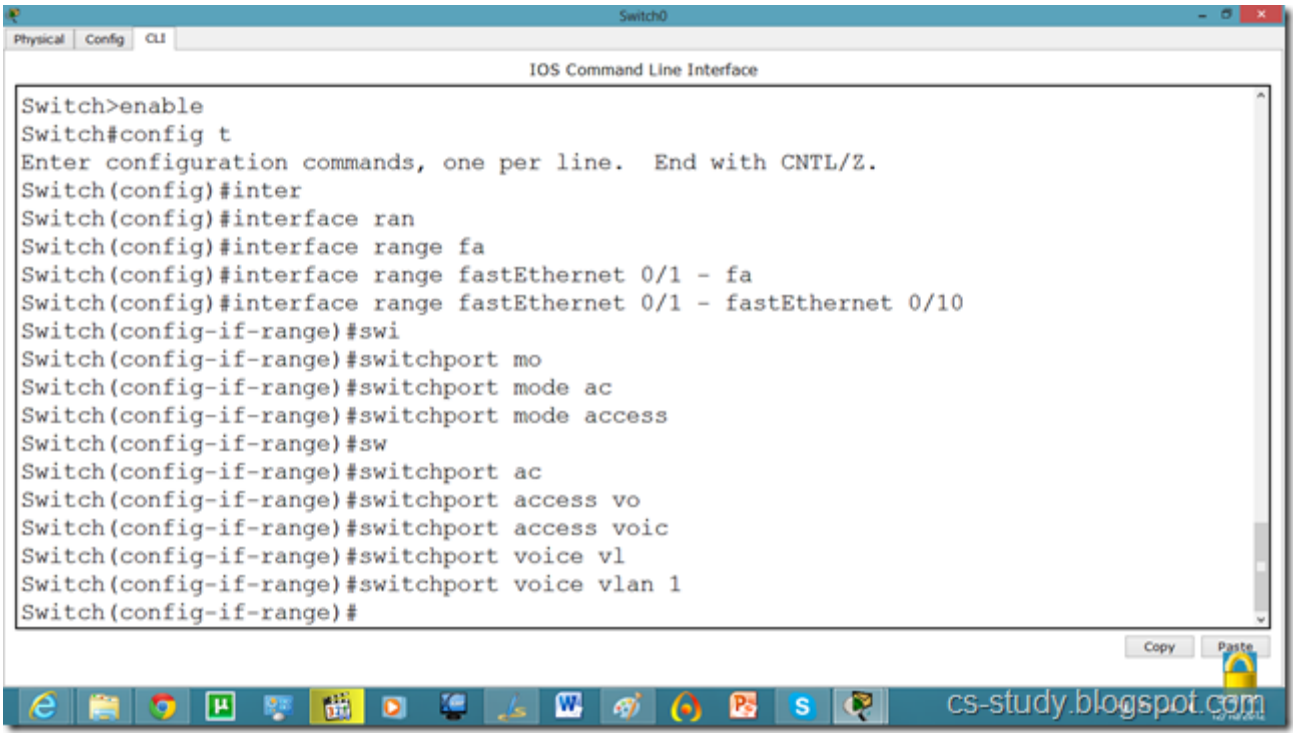
Now, let us apply commands to the router for voip. You might have noticed we took 2811 series router because it facilitates the following commands.

```
Router(config)#telephony-service
Router(config-telephony)#?
  auto              Define dn range for auto assignment
  auto-reg-ephone   Enable Ephone Auto-Registration
  create            create cnf for ethernet phone
  exit              Exit from telephony config mode
  ip                Define IP address and port for Telephony-Service/Fallback
  keepalive         Define keepalive timeout period to unregister IP phones
  max-dn            Maximum directory numbers supported
  max-ephones       Define max number of IP phones
  no                Negate or set default values of a command
Router(config-telephony)#max-dn ?
  <1-144>  Maximum directory numbers supported
Router(config-telephony)#max-dn 5
Router(config-telephony)#max-eph
Router(config-telephony)#max-ephones 5
Router(config-telephony)#ip sour
Router(config-telephony)#ip source-address 192.168.1.1 ?
  port  Define tcp port for Telephony Service/CM FALLBACK
```

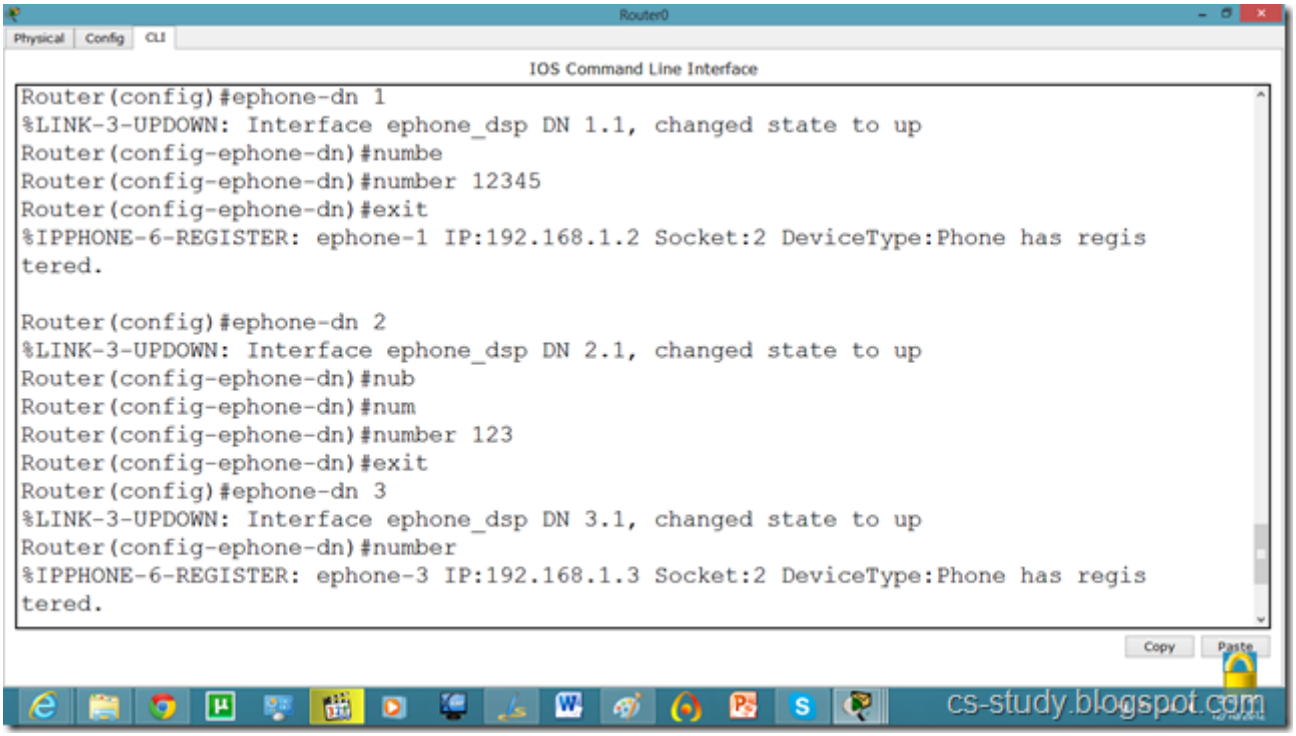
Continued.

```
Router(config-telephony)#max-dn ?
  <1-144>  Maximum directory numbers supported
Router(config-telephony)#max-dn 5
Router(config-telephony)#max-eph
Router(config-telephony)#max-ephones 5
Router(config-telephony)#ip sour
Router(config-telephony)#ip source-address 192.168.1.1 ?
  port  Define tcp port for Telephony Service/CM FALLBACK
Router(config-telephony)#ip source-address 192.168.1.1 port ?
  <2000-9999>  Specify the port: 2000 - 9999
Router(config-telephony)#ip source-address 192.168.1.1 port 2000?
<2000-9999>
Router(config-telephony)#ip source-address 192.168.1.1 port 2000 ?
  <cr>
Router(config-telephony)#ip source-address 192.168.1.1 port 2000
Router(config-telephony)#aut
Router(config-telephony)#auto as
Router(config-telephony)#auto assign 1 to 9
Router(config-telephony)#
```

Now, we will go to the switch and make the interfaces support voip as follows.



After that, we will have to assign phone number to our IP phone by applying following commands.



Continued…

Now, we can see that in GUI mode of Phone, we have a phone number available.

This means that we can call from one phone to the other. Lets do that.

And when we pick up the receiver, it says that we are connected.

Commands on Router for VOIP

```
Router(config)#ip dhcp pool voice
Router(dhcp-config)#option 150 ip 192.168.1.1
Router(dhcp-config)#exit
Router(config)#telephony-service
Router(config-telephony)#max-dn 5
Router(config-telephony)#max-ephones 5
Router(config-telephony)#ip source-address 192.168.1.1 port 2000
Router(config-telephony)#auto assign 1 to 9
Router(config-telephony)#exit
Router(config)#ephone-dn 1
%LINK-3-UPDOWN: Interface ephone_dsp DN 1.1, changed state to up
Router(config-ephone-dn)#number 12345
Router(config-ephone-dn)#exit
```

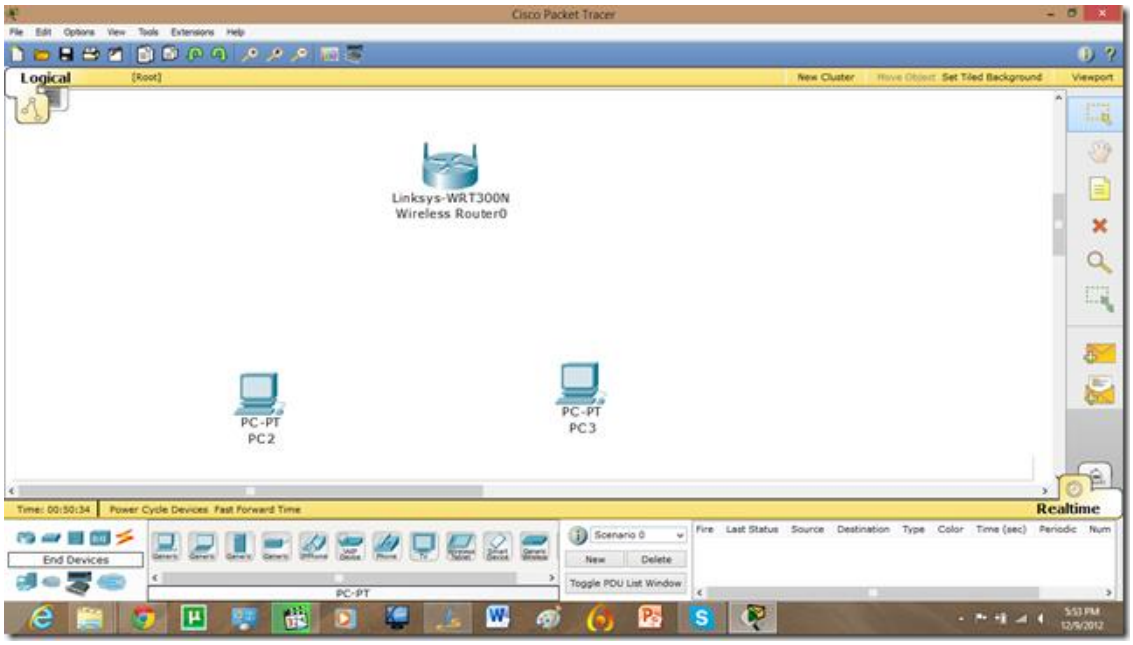
%IPPHONE-6-REGISTER: ephone-1 IP:192.168.1.2 Socket:2 DeviceType:Phone has registered.
Router(config)#ephone-dn 2
%LINK-3-UPDOWN: Interface ephone_dsp DN 2.1, changed state to up
Router(config-ephone-dn)#number 123
Router(config-ephone-dn)#exit
Router(config)#ephone-dn 3
%LINK-3-UPDOWN: Interface ephone_dsp DN 3.1, changed state to up
Router(config-ephone-dn)#number 11111
Router(config-ephone-dn)#exit

Commands on Switch for VOIP

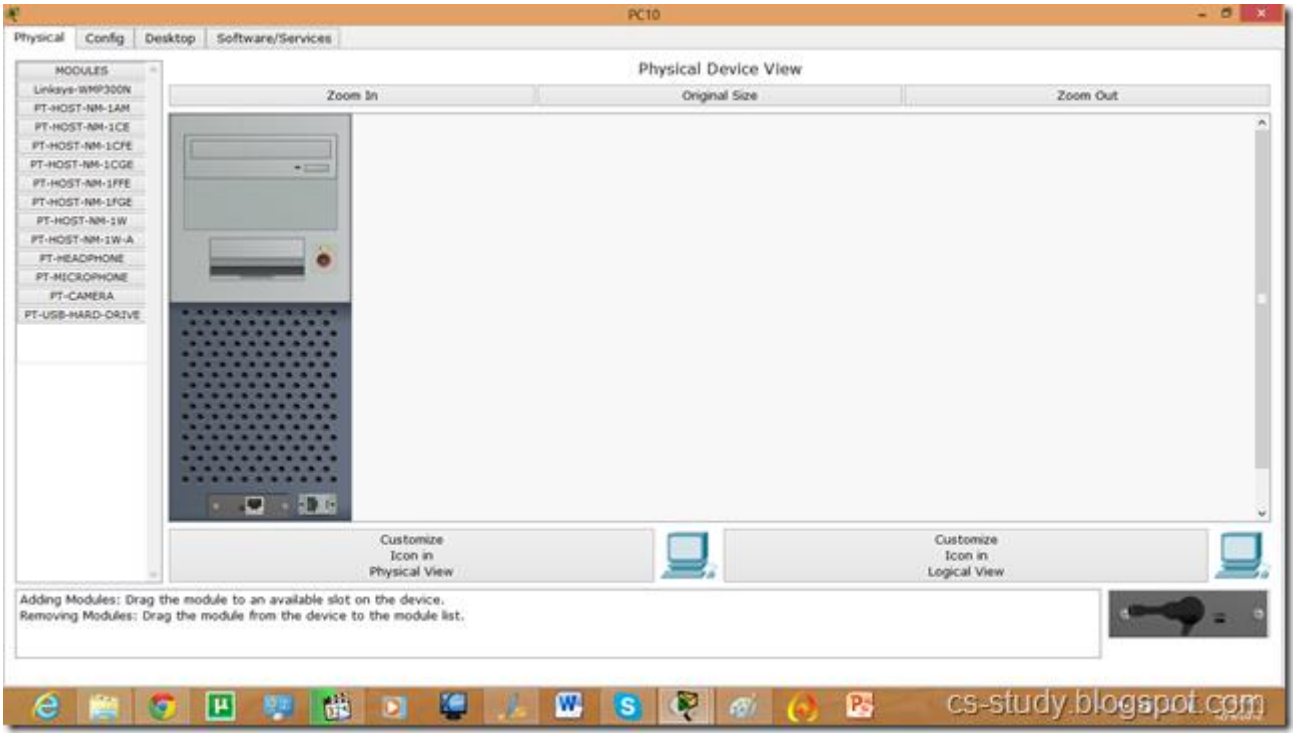
Switch(config)#interface range fastEthernet 0/1 - fastEthernet 0/10
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport voice vlan 1

Task 2: Wireless on packet tracer

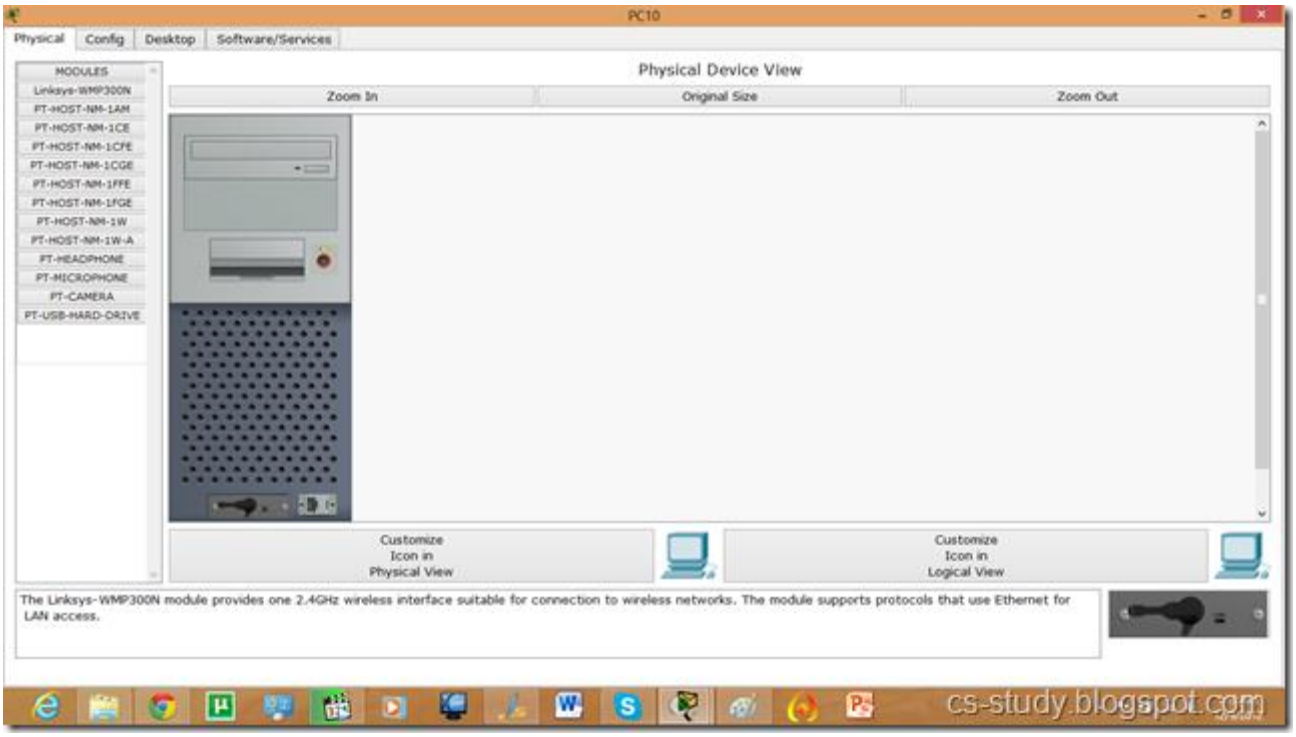
Let us create wireless topology on packet tracer. For this go to the wireless devices and select linksys wireless router, take some PCs and provide them with wireless linksys module so that they can communicate through router wirelessly. For that go to the PC physical mode as shown in the figures below.



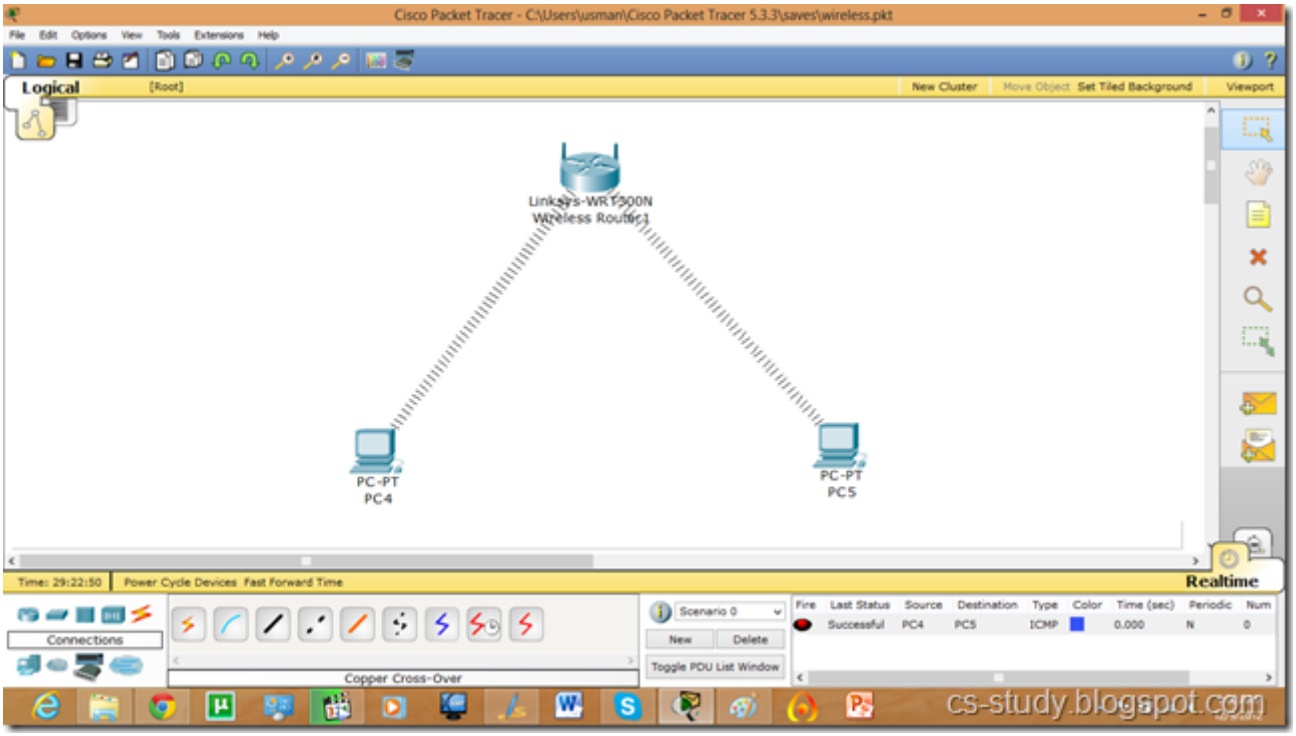
Go to PC, and remove wired LAN and install wireless LAN module.



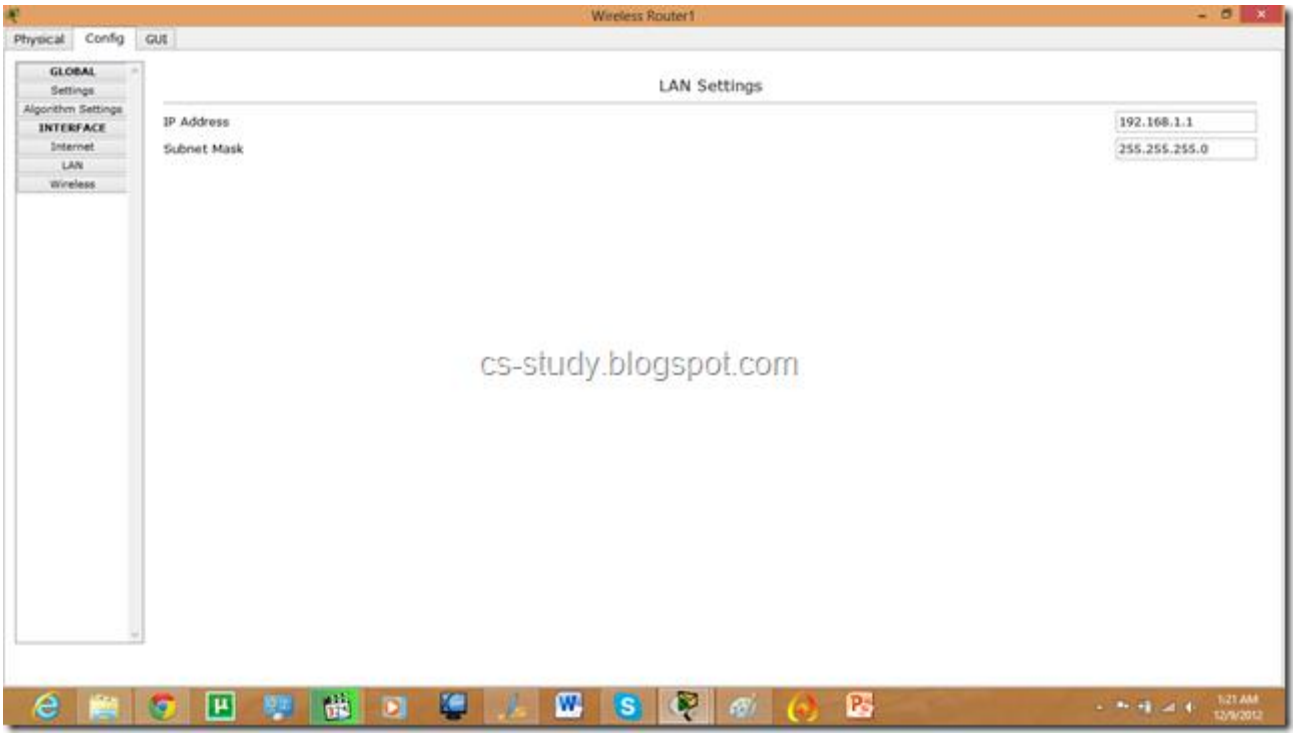
Now, it is removed. Let us add wireless module.



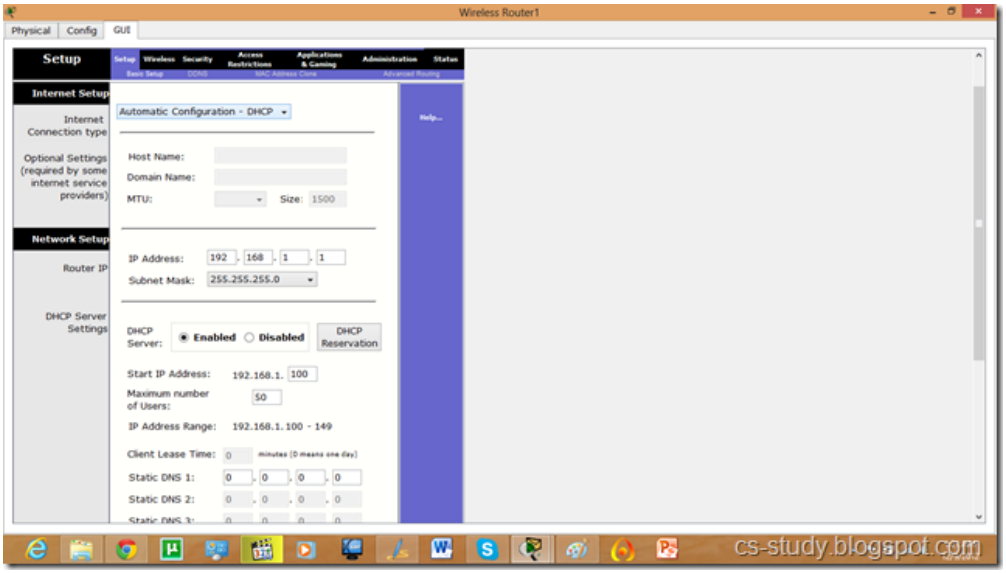
Now, PCs are connected.



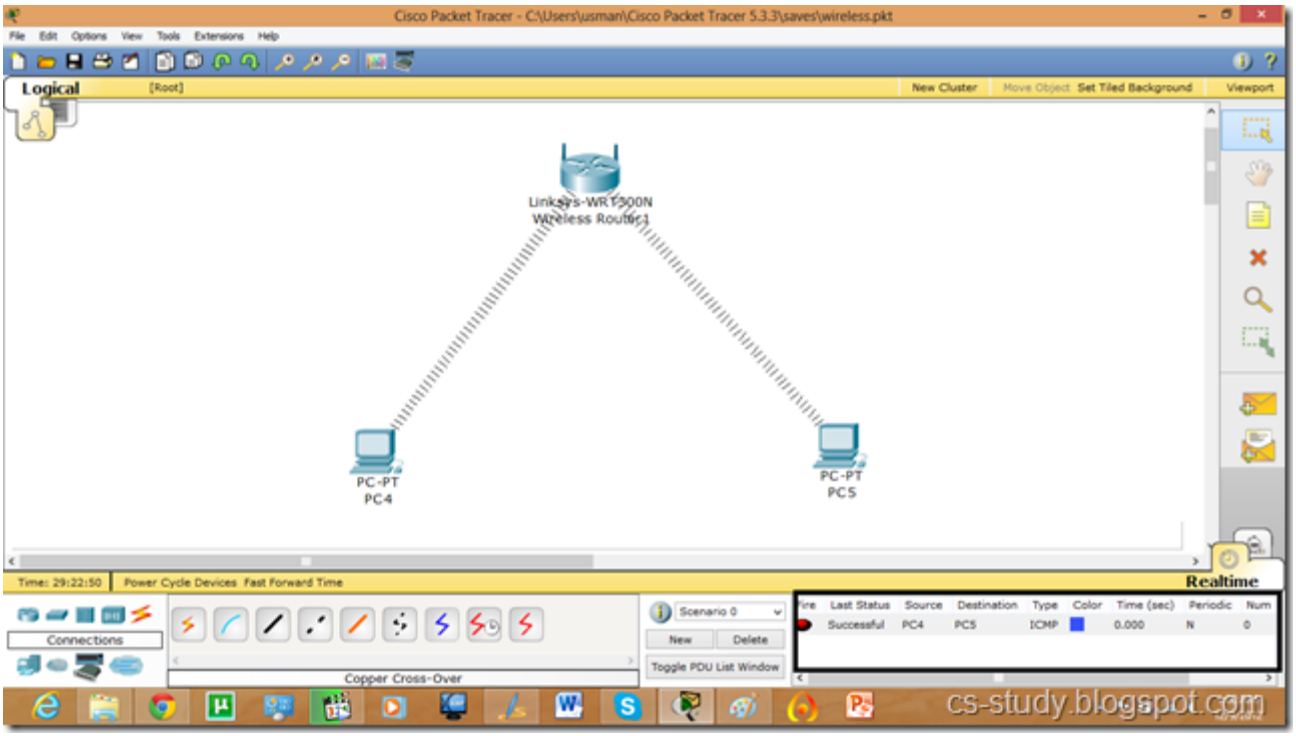
Set the IP address.



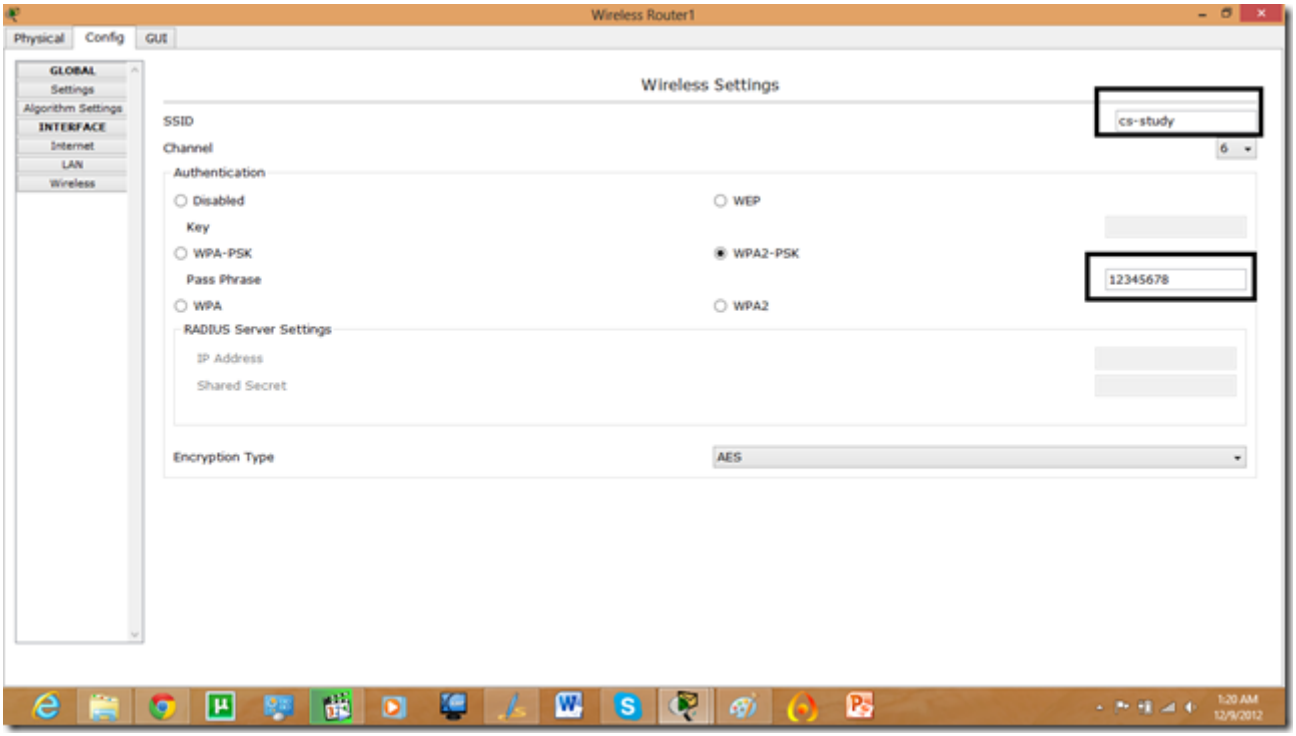
As this wireless router provides us with the DHCP service, so we can obtain IP automatically by using this service for our PCs.



So now our PCs can communicate.

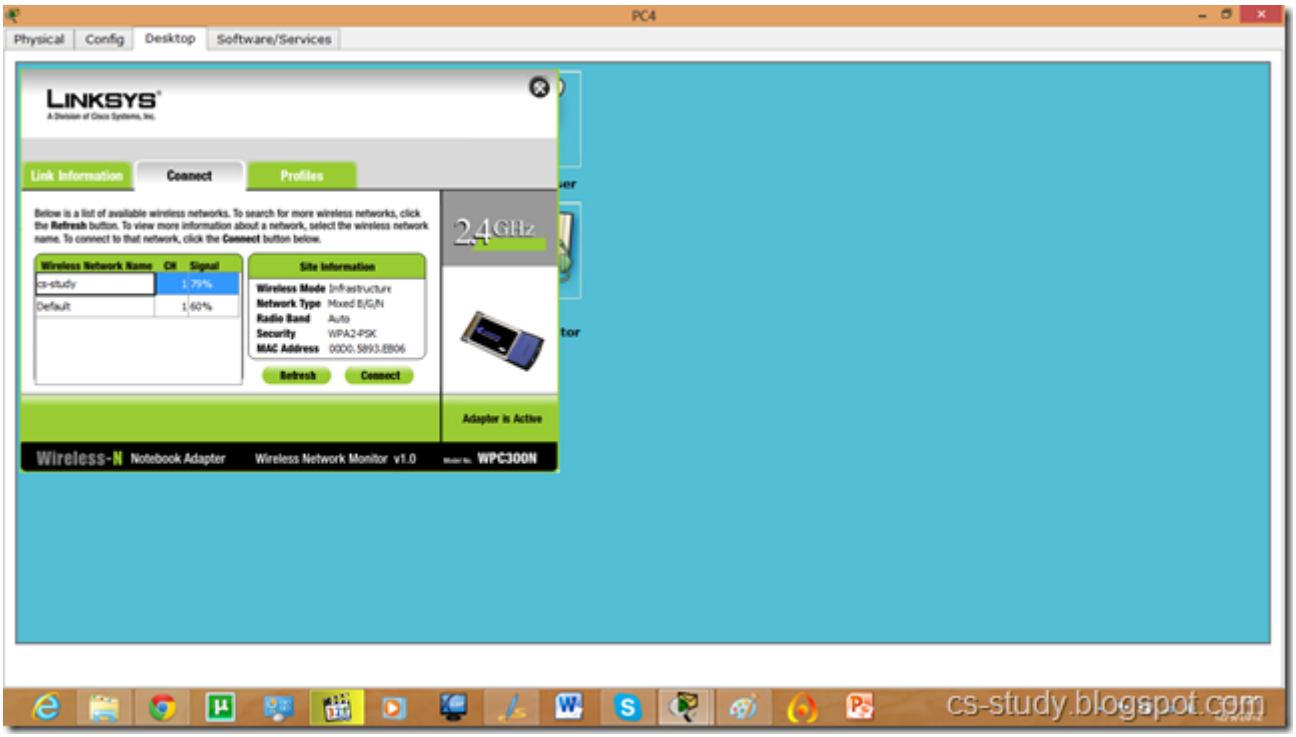


Now, let us apply authentication to our wireless router. For that, go to Config tab, click on Wireless. Provide it with the information as described below.

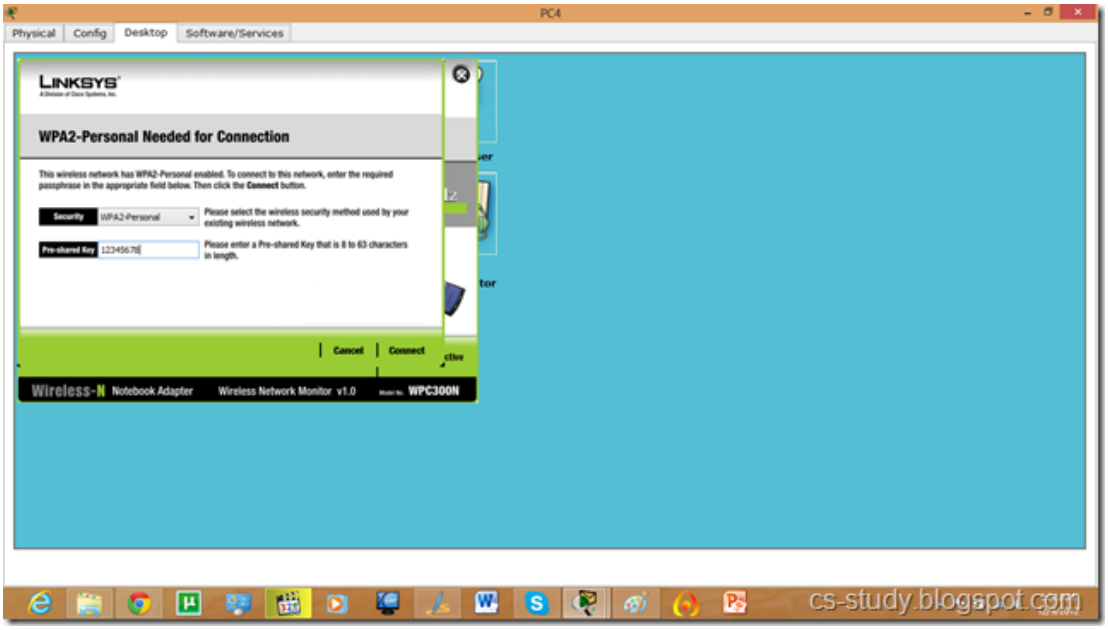


Go to PC desktop mode, Click on PC Wireless.

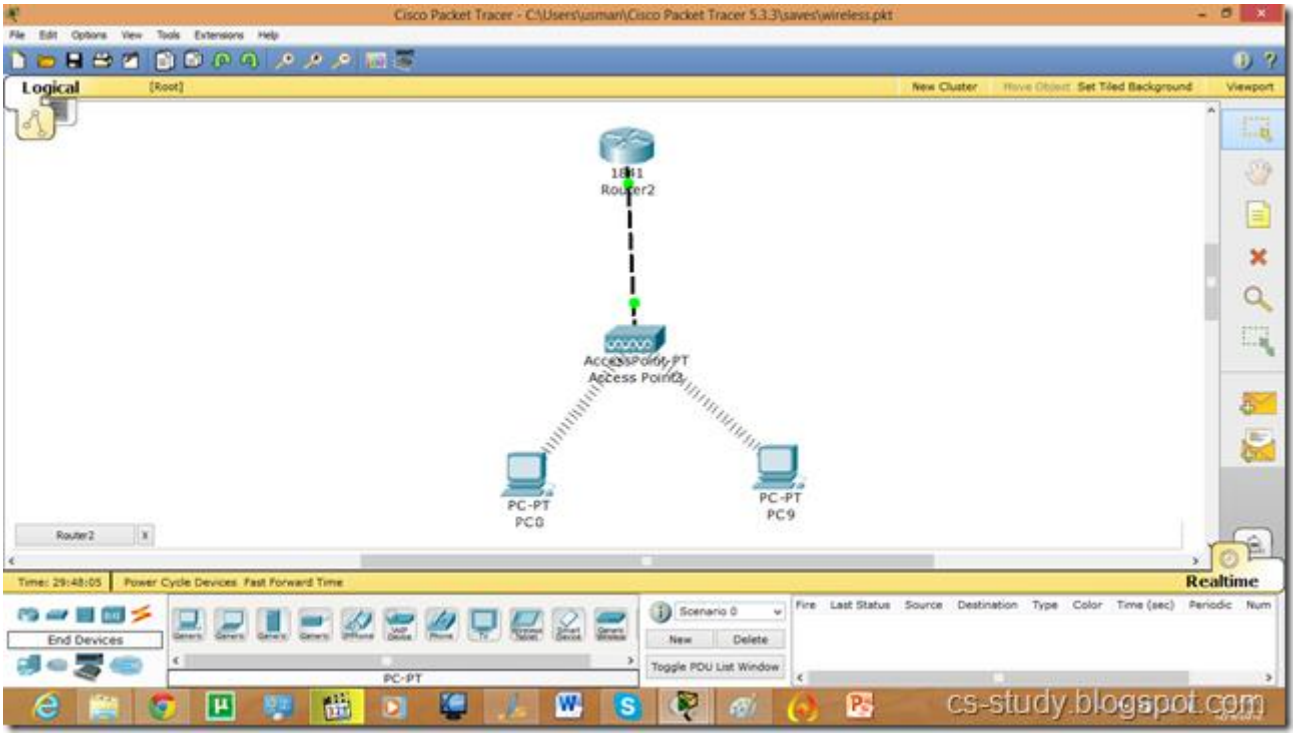
Click on connect. Select the device, you want to connect to, click connect.



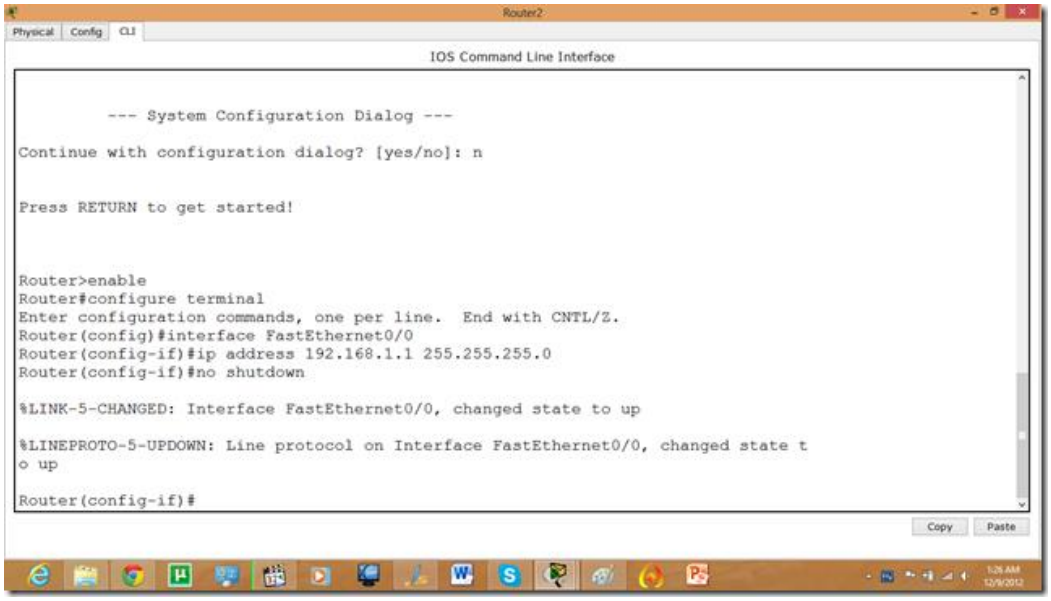
Give correct password.



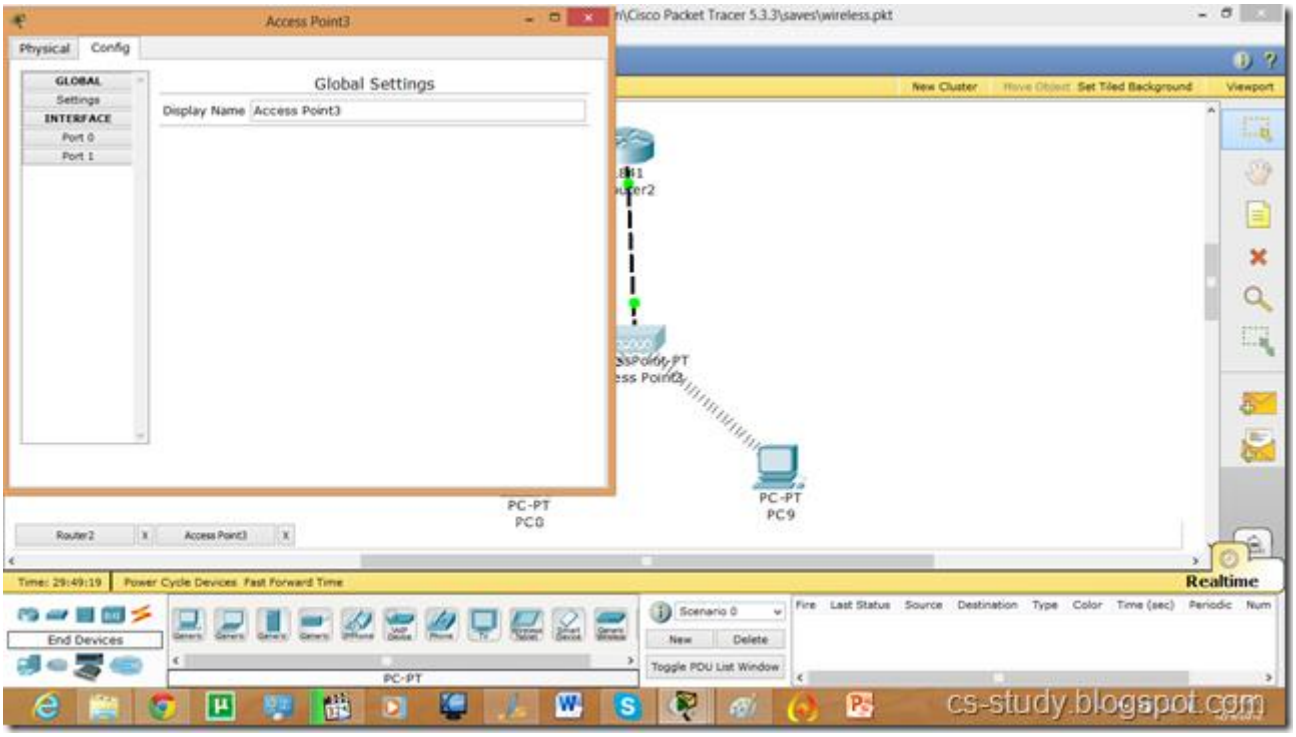
Now, we are done with it. We have successfully applied authentication. Now, let us use Access point to connect to PCs wirelessly. We can also connect wired router to access point in order to make our router wireless.



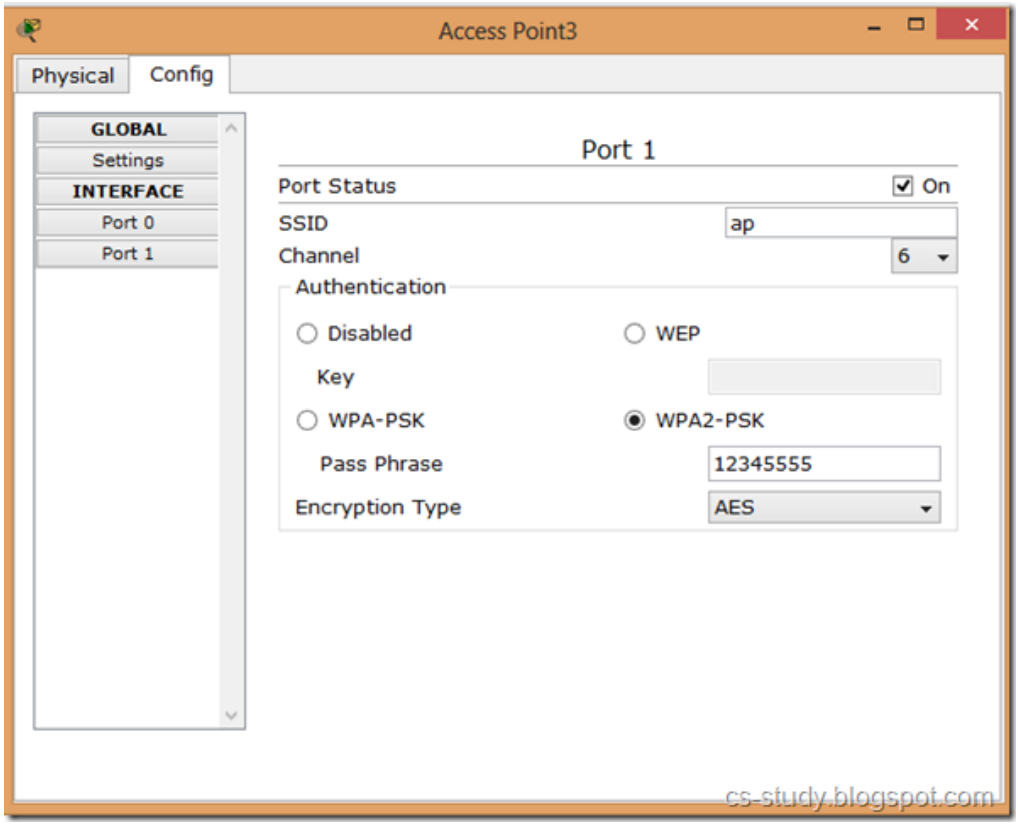
Apply IP addresses and put the status on.



We can also give authentication key to Access Point as well.

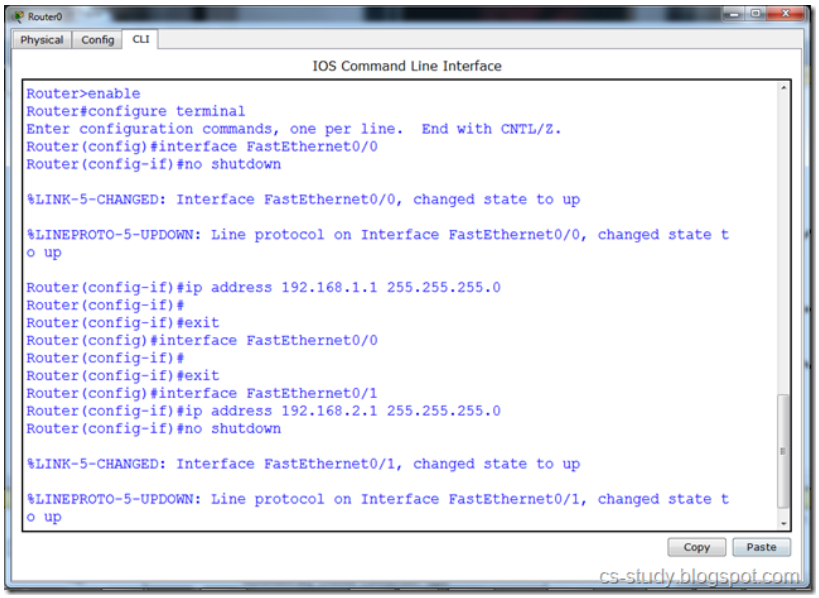


As in this figure below.

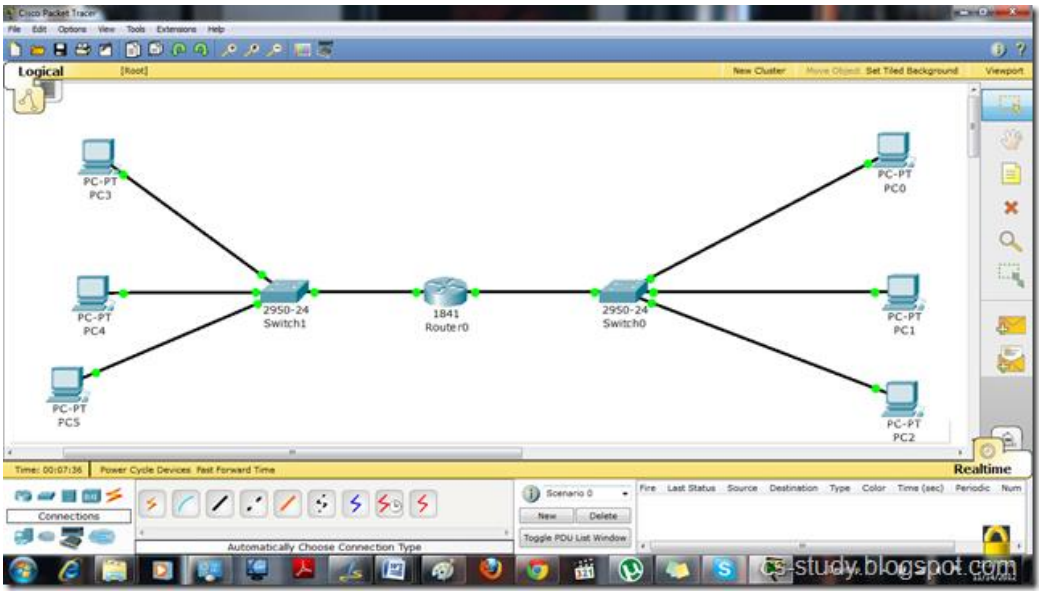


Task 3: ACL

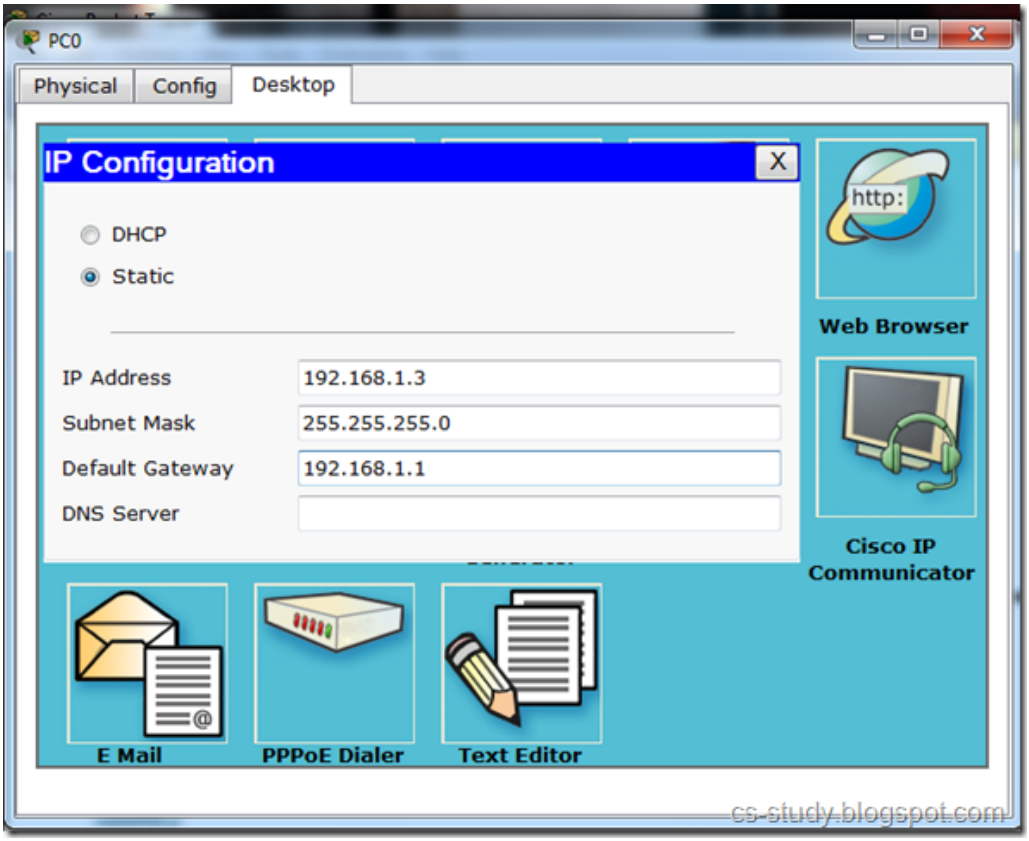
Let us apply ACL (Access Control List) on the topology. First, let us assign IP addresses and change the state of the interfaces.



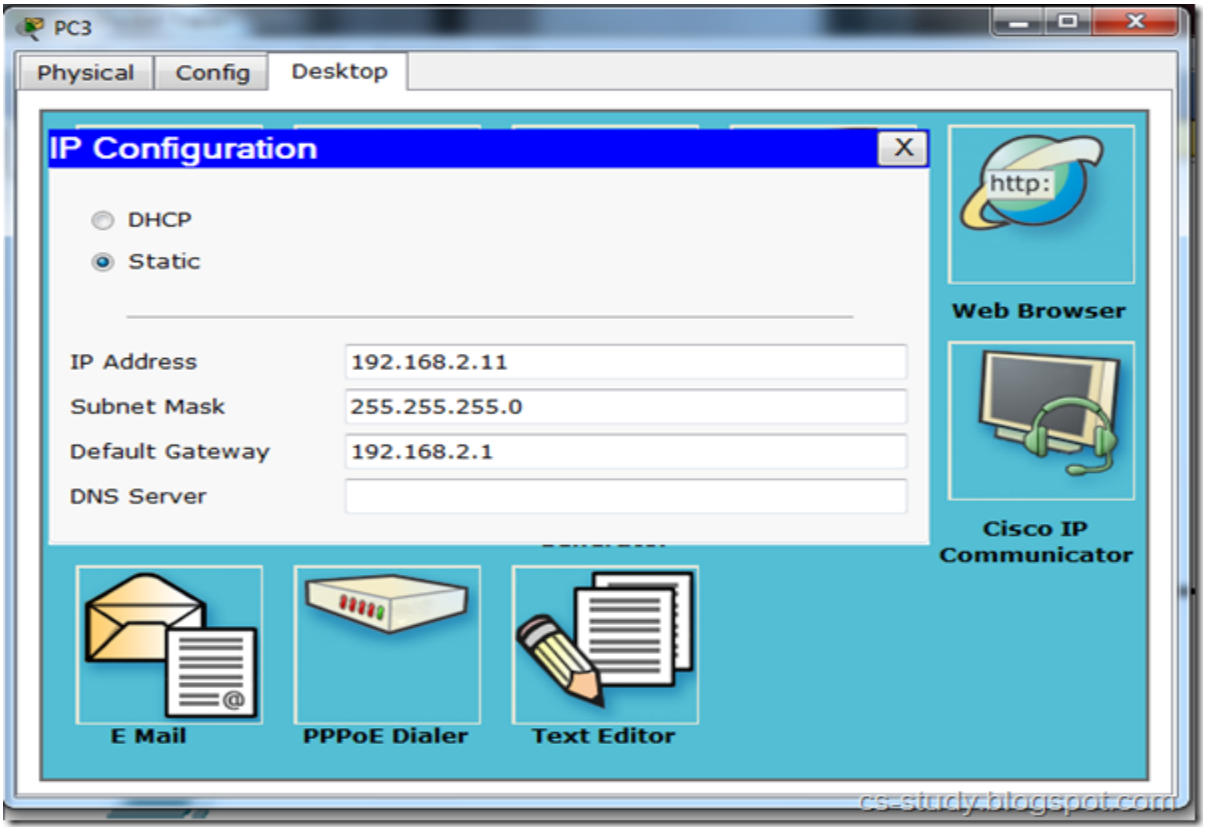
Here, the status is change as shown in the following diagram.



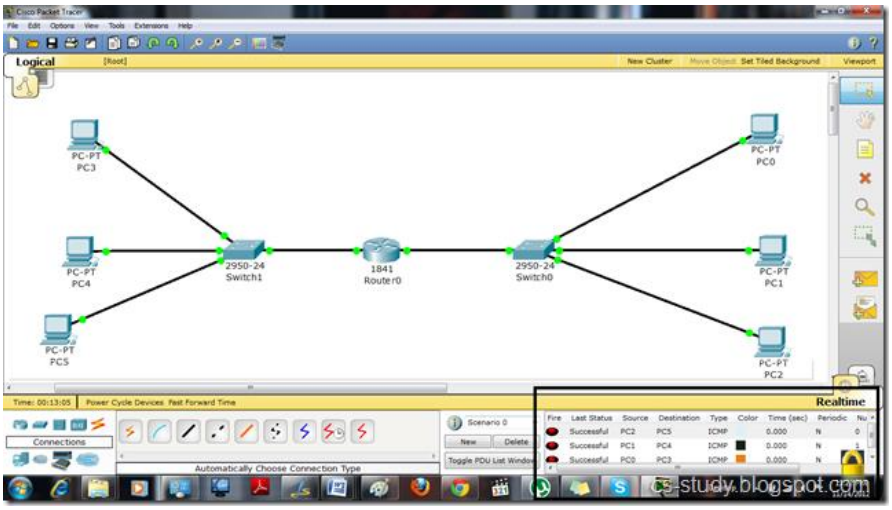
Then, we will have to assign IP addresses to the PCs.



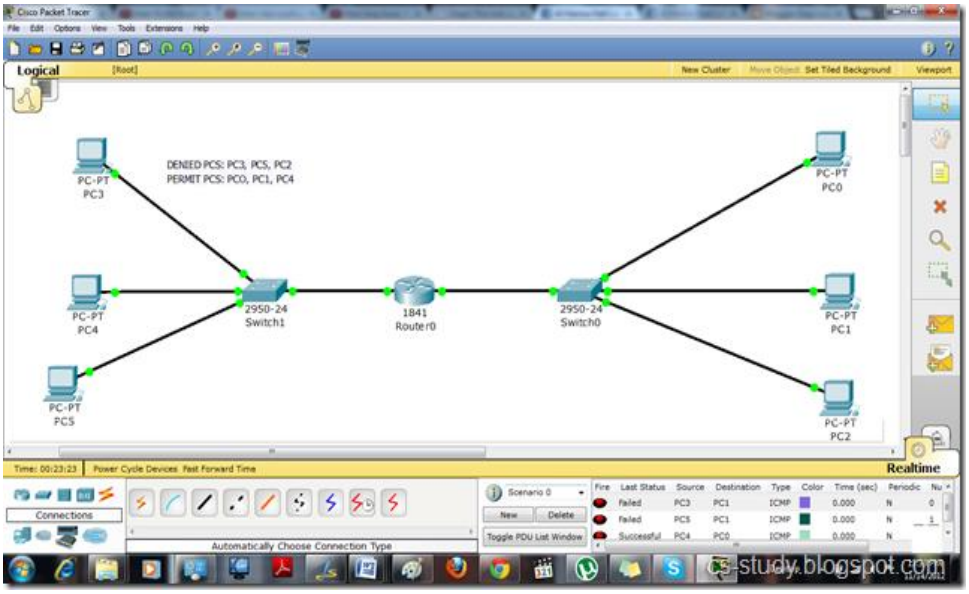
And the PC attach to the other interface. Please note the difference between default gateways. We assign fast Ethernet interface address to default gateway.



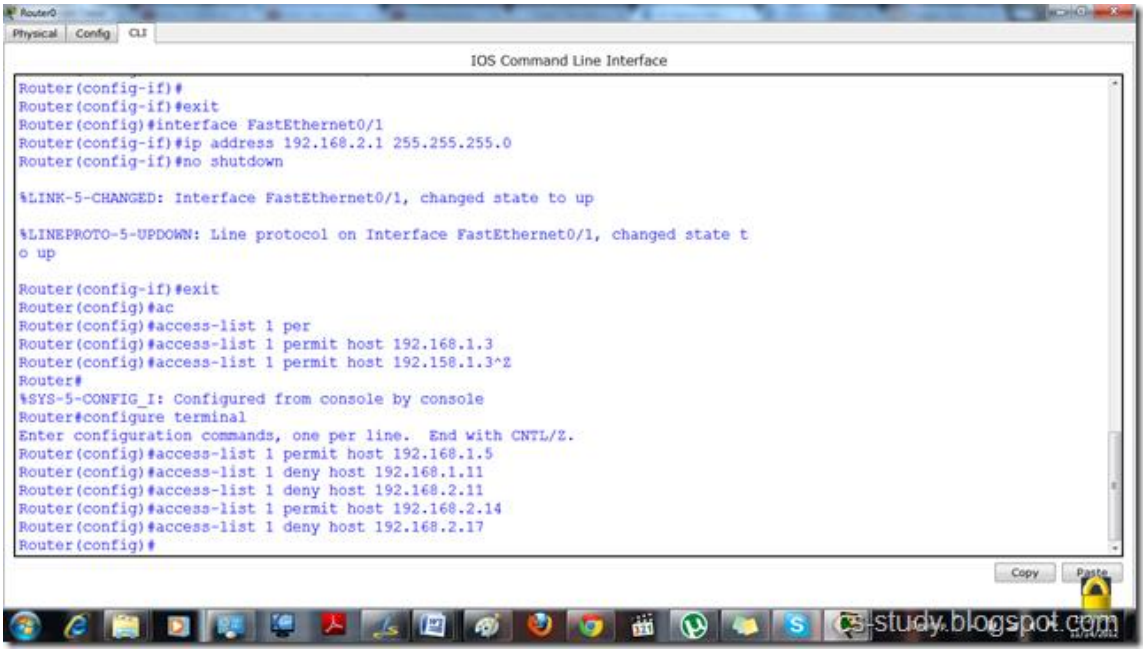
Thus, after applying IP addresses. We see that Packet transfer is successful.



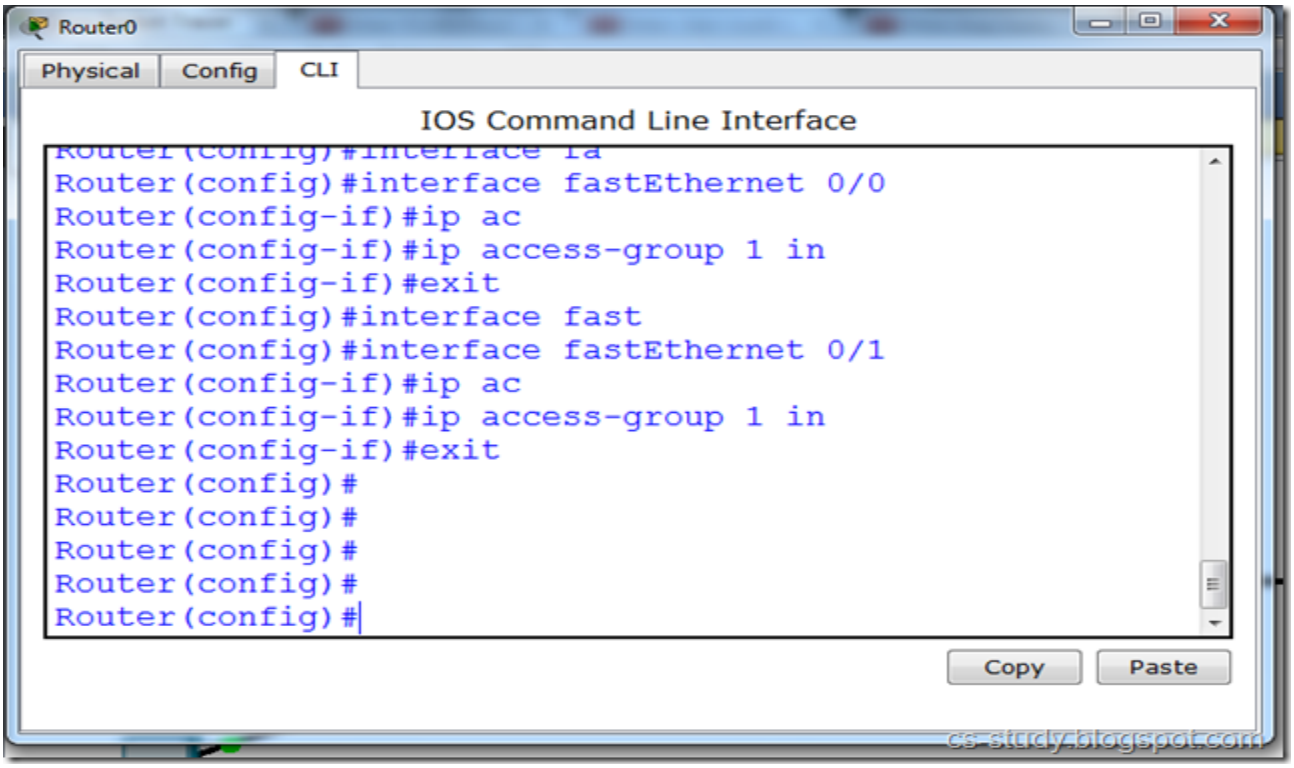
Let us apply ACL and permit and deny Hosts IP's as we want. We are going to deny and permit certain hosts as follows.



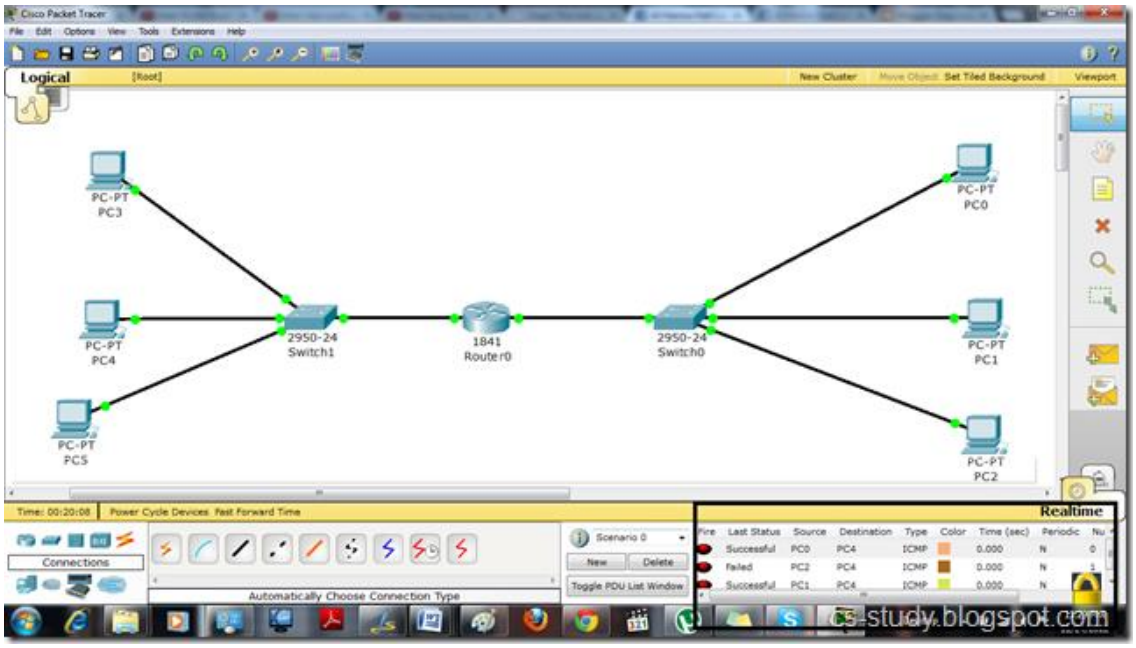
Let us apply ACL and permit and deny Hosts IP's as we want.



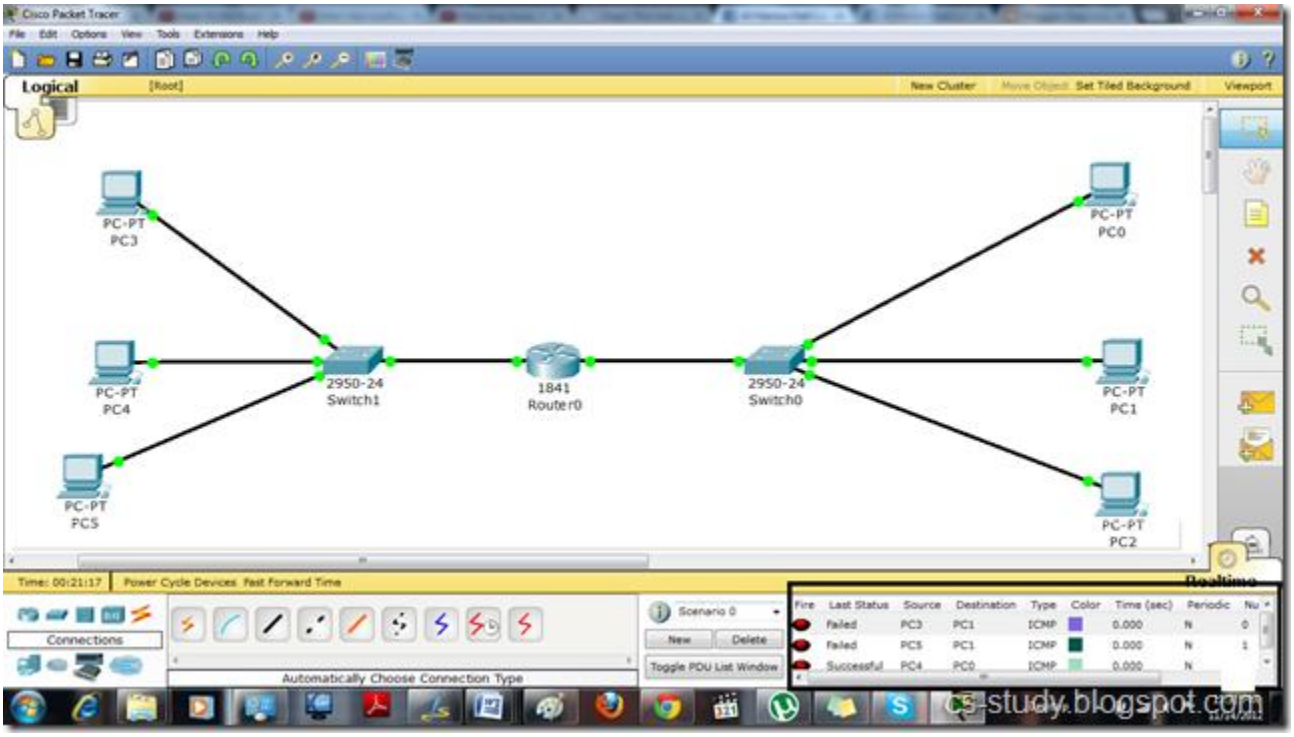
Then, we will have to tell the interface that which ACL to follow. ACL is uniquely identified with the number, in this case 1.



Now, you see that the denied will not be able to send the data while those who we permit can send packet.



Similarly, in the bottom right corner you can see status.


Cisco Packet Tracer
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
PC-PT
PC3
PC-PT
PC4
PC-PT
PC5
2950-24
Switch1
1841
Router0
2950-24
Switch0
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
Time: 00:21:17
Power Cycle Devices
Fast Forward Time
Realtime
Connections
Automatically Choose Connection Type
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time (sec)
Periodic
failed
PC3
PC1
ICMP
0.000
N
failed
PC5
PC1
ICMP
0.000
N
Successful
PC4
PC0
ICMP
0.000
N
cs-study.blogspot.com

Lab 13 – NS2 and Ubuntu Installation, NS2 introduction

Objectives

The main objective of this lab is to install and introduce network simulator.

Lab Tasks

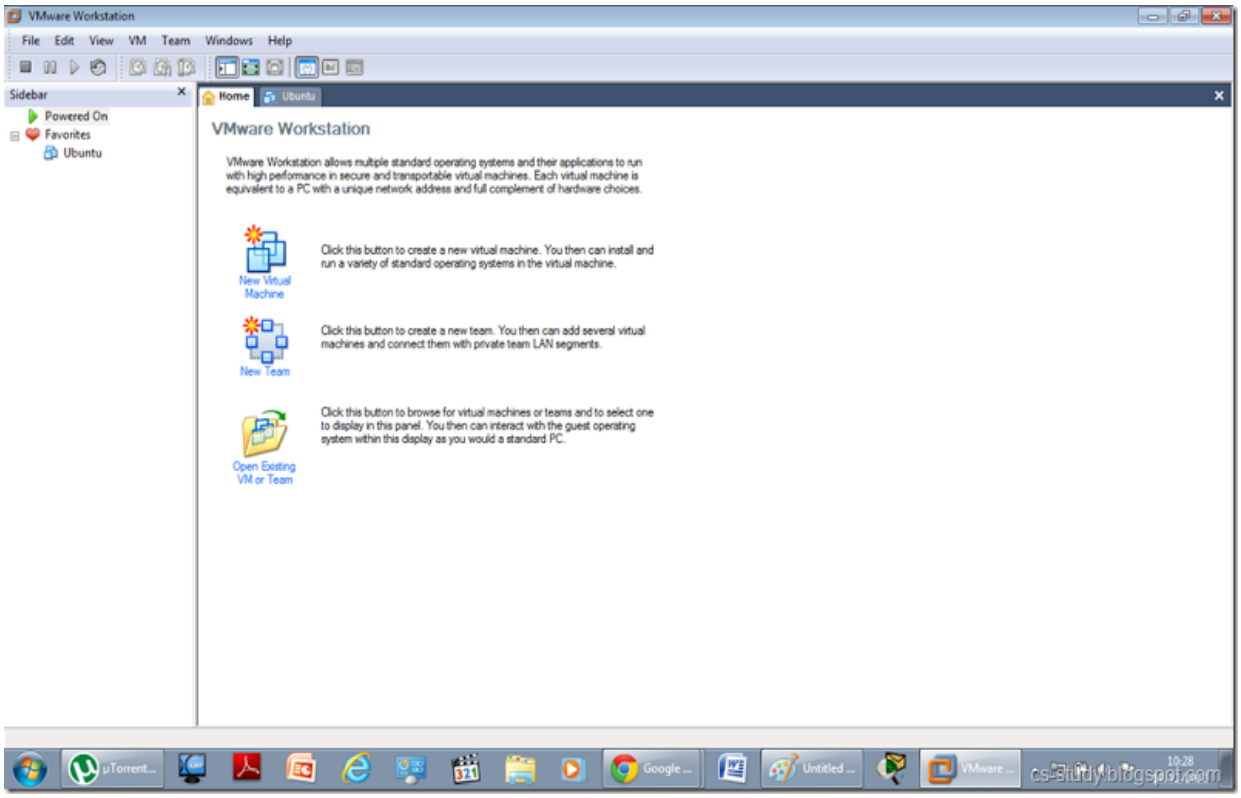
Install Ubuntu and then NS2.

There are multiple ways that you can adopt to install ns2. As ns2 runs on the linux. so first you will have to install linux. You can install any distribution of linux fedora, ubuntu etc.

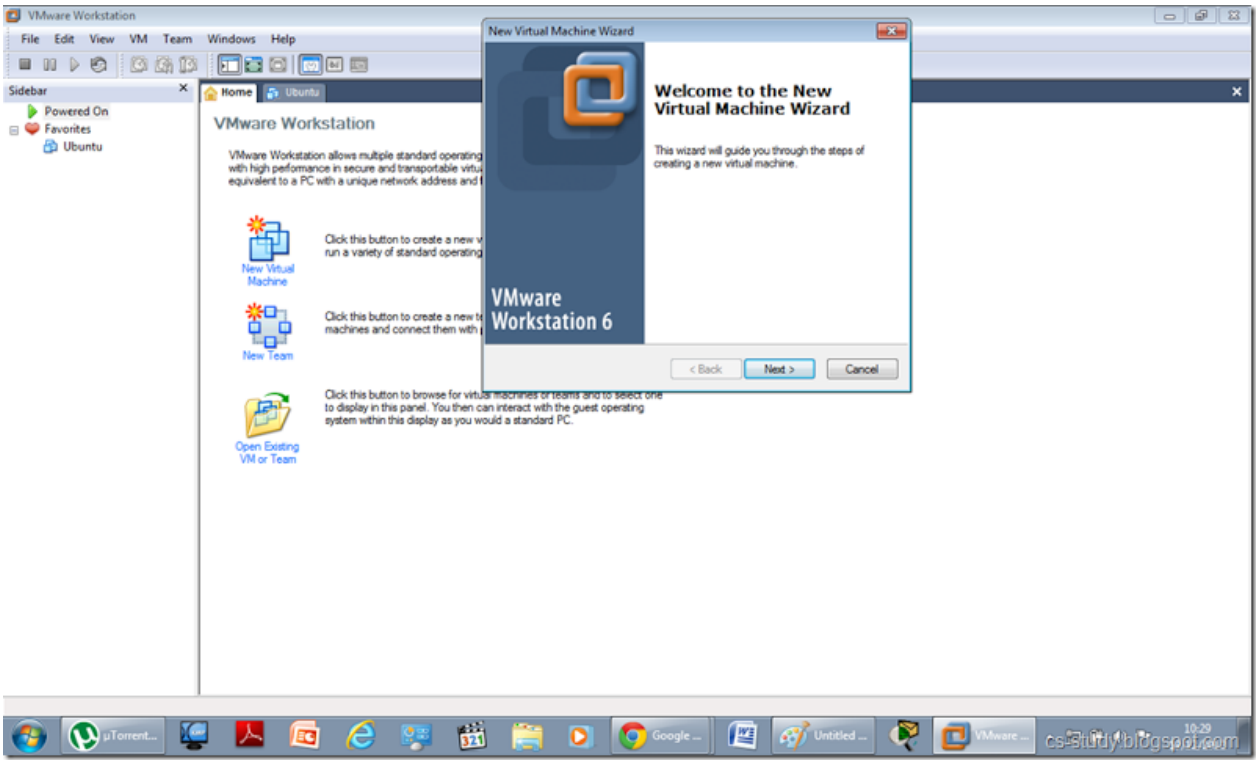
So, one way to do the installation is to first install this software called VMWARE. It creates a virtual machine which allows you to run operating system on another operating system just like any other application. Other ways include by installing Ubuntu direct from website using windows installer. Lets look at the first option.

So,

1. Install Vmware
2. Install fedora or any other Linux distribution on VMware. Select 'New Virtual Machine' option.

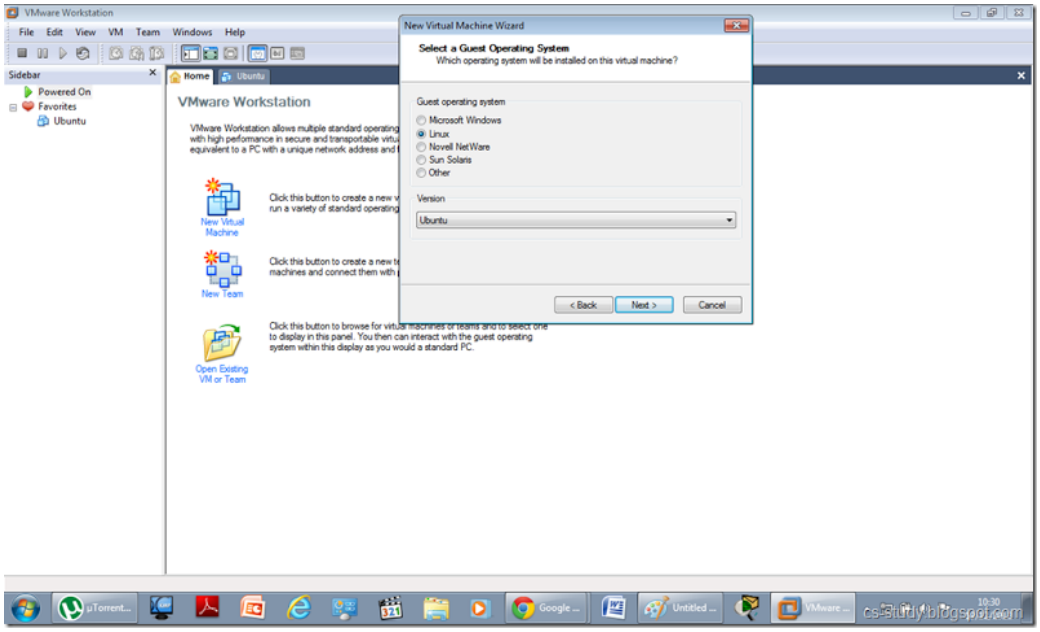


Click Next.

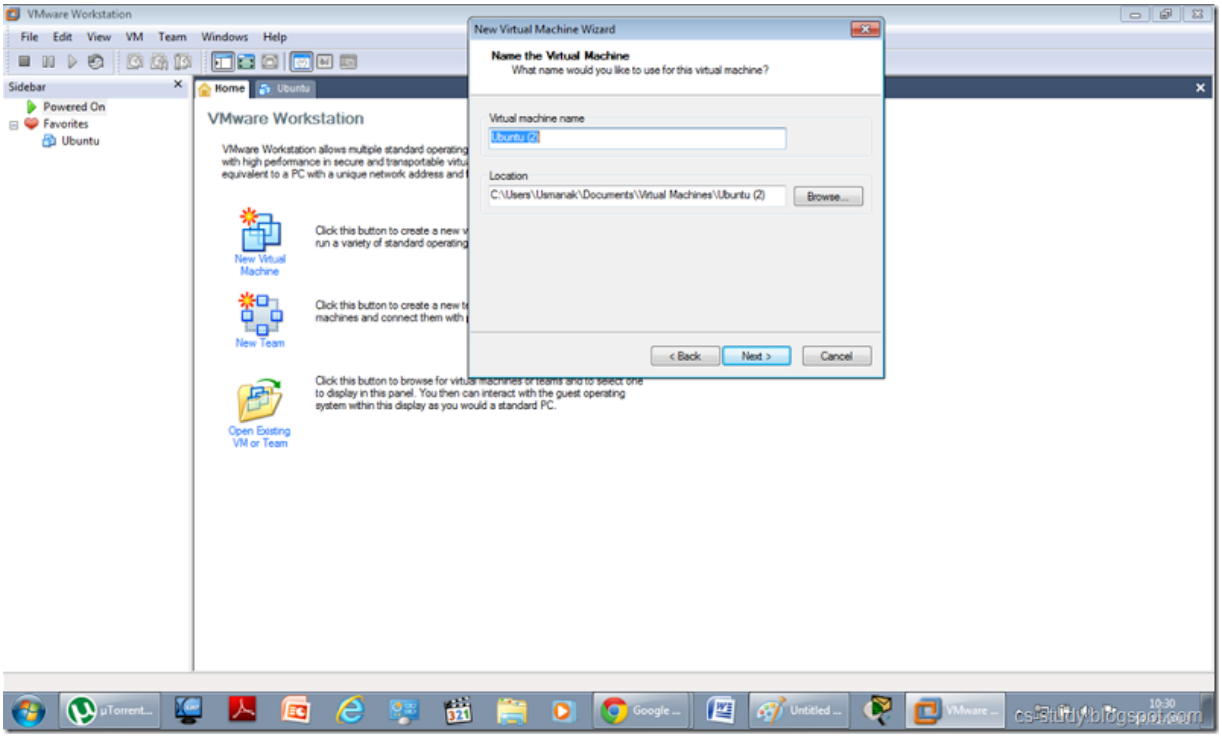


Select Typical.

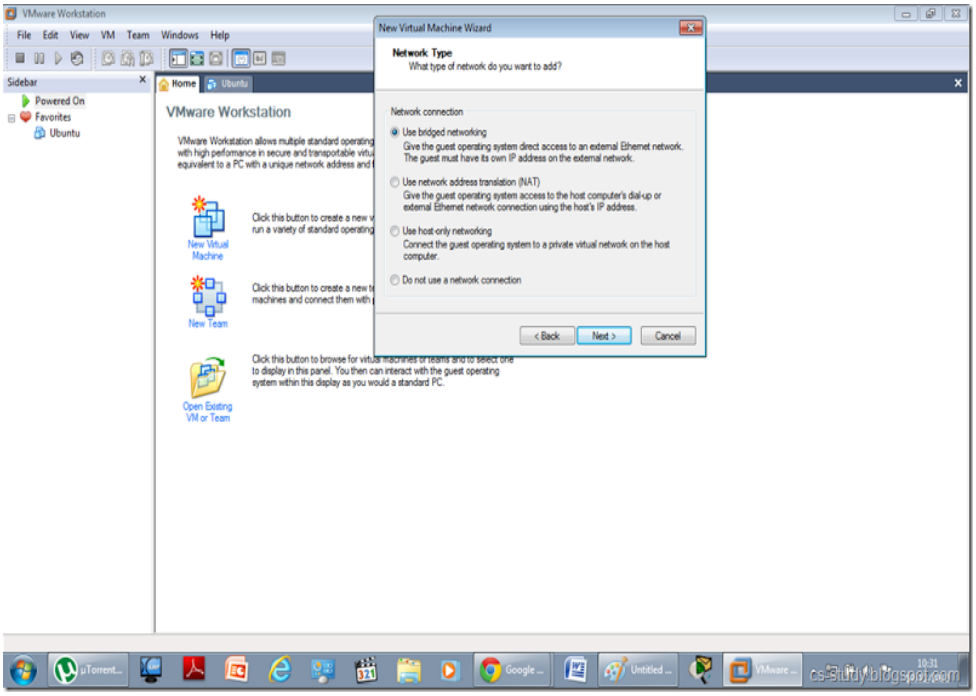
Select Linux and Ubuntu.



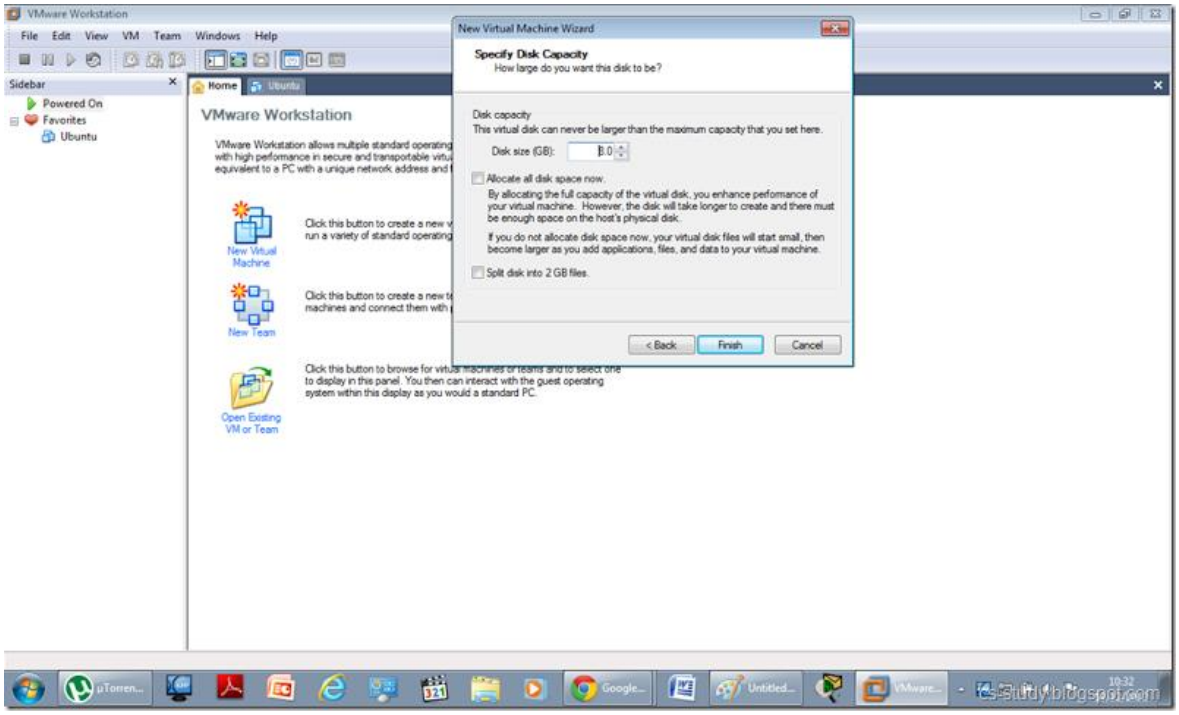
Next …



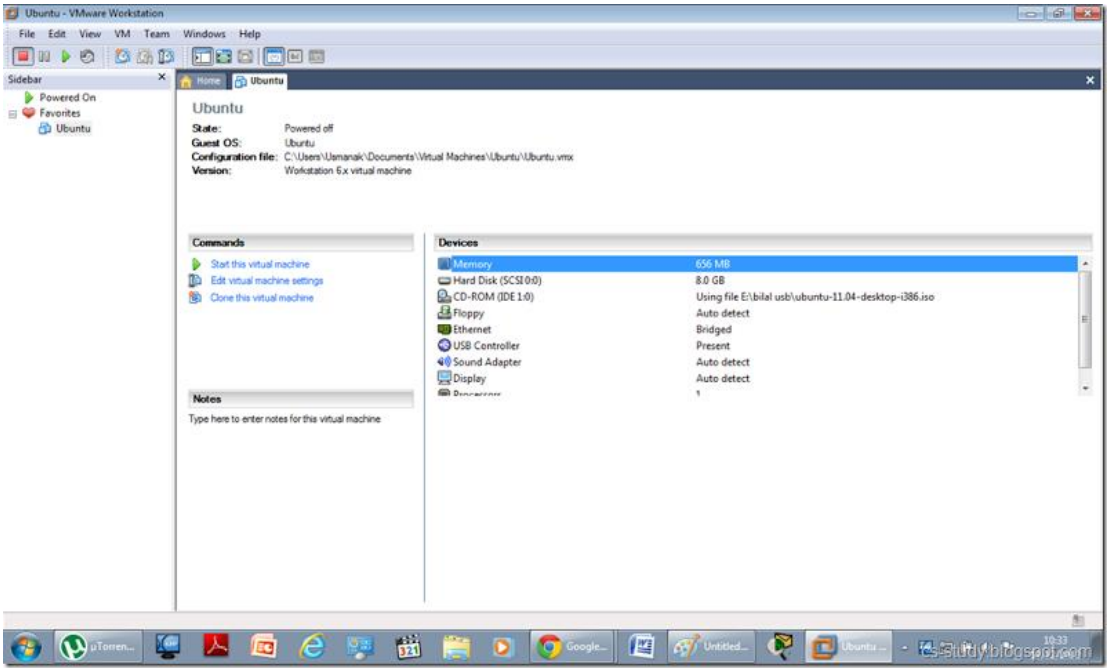
Select Bridged networking.



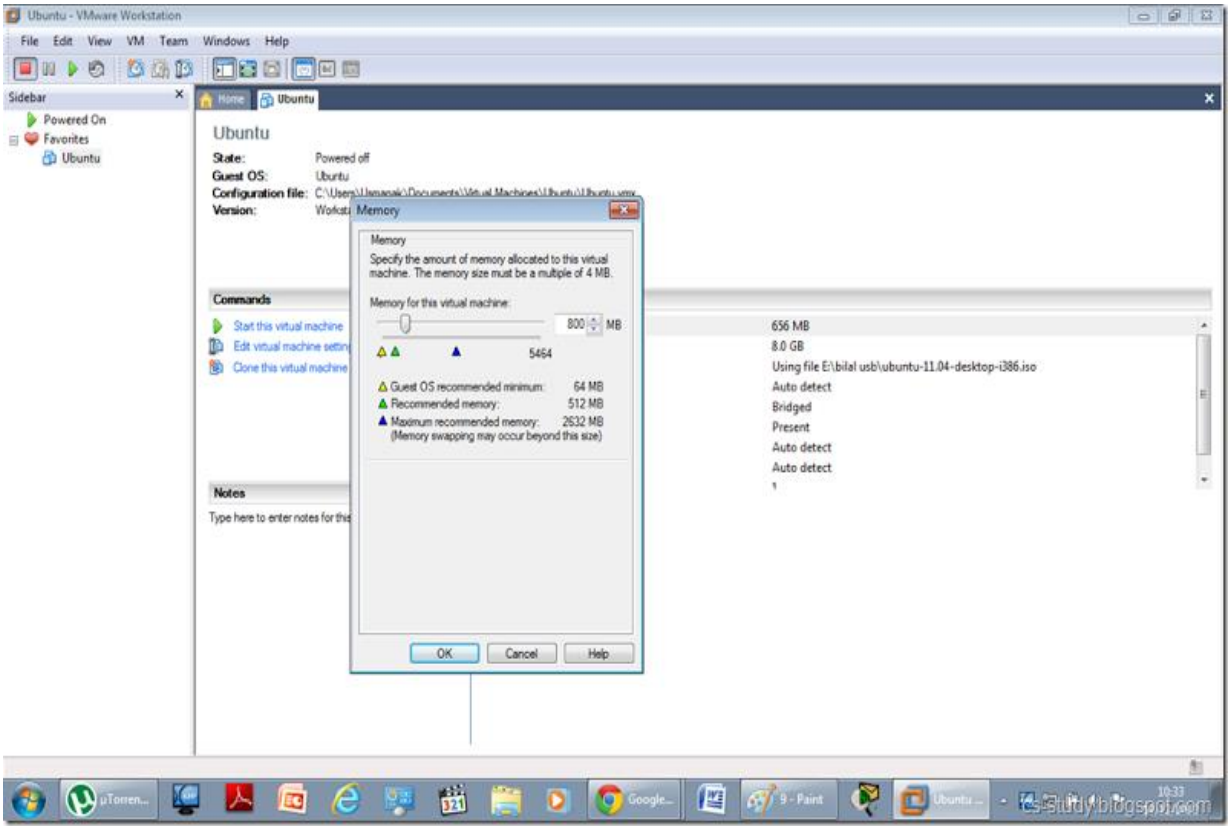
You can change the hard disk space availability for your virtual machine based on your needs and available space. Click Finish.



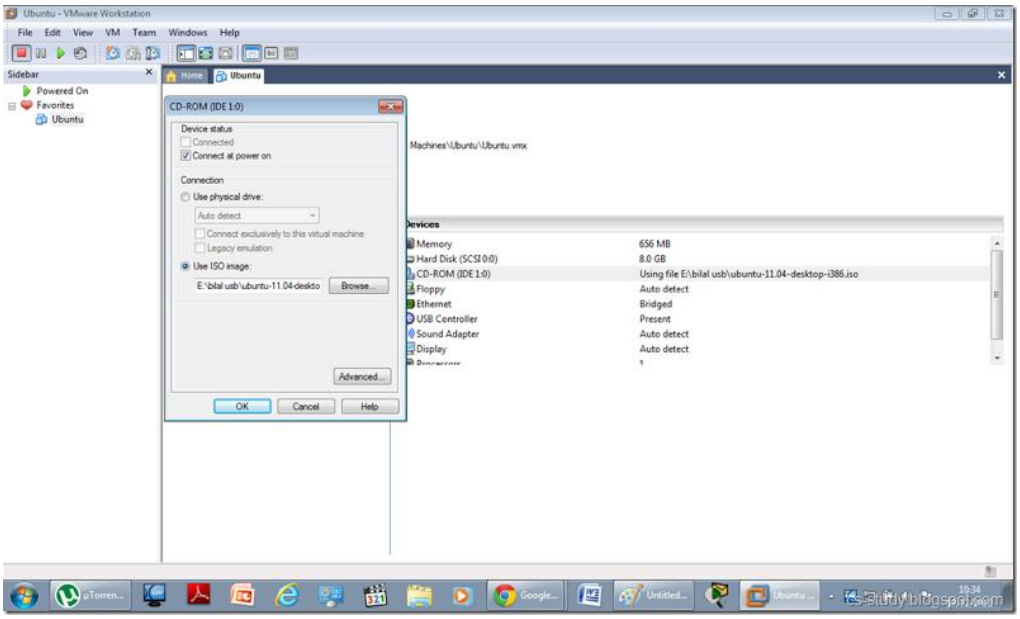
Now, that we have create our virtual machine, let us set some more parameters.



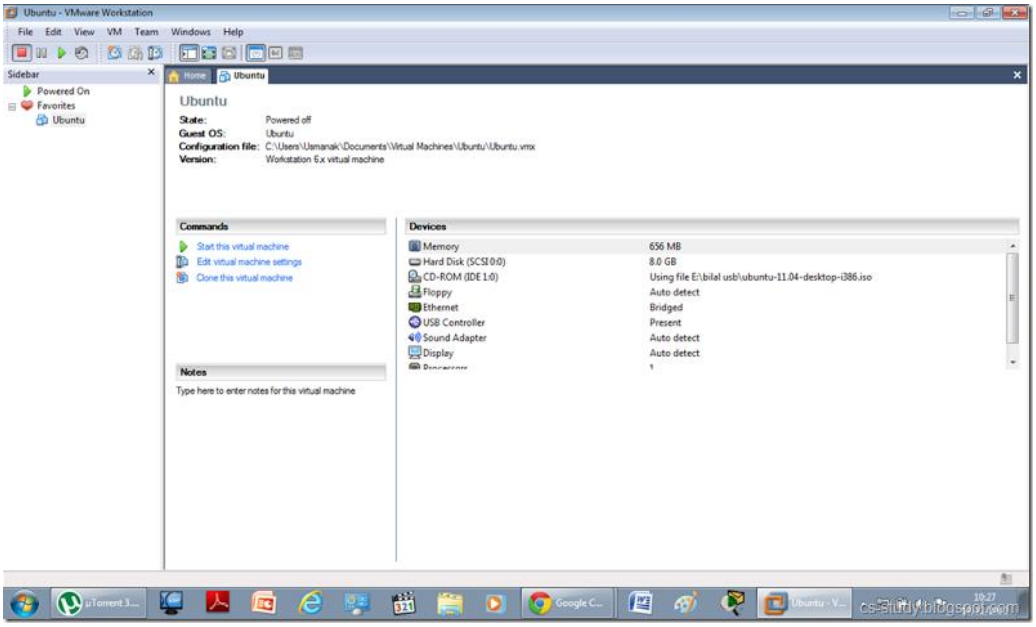
Set RAM that will be available for this virtual machine.



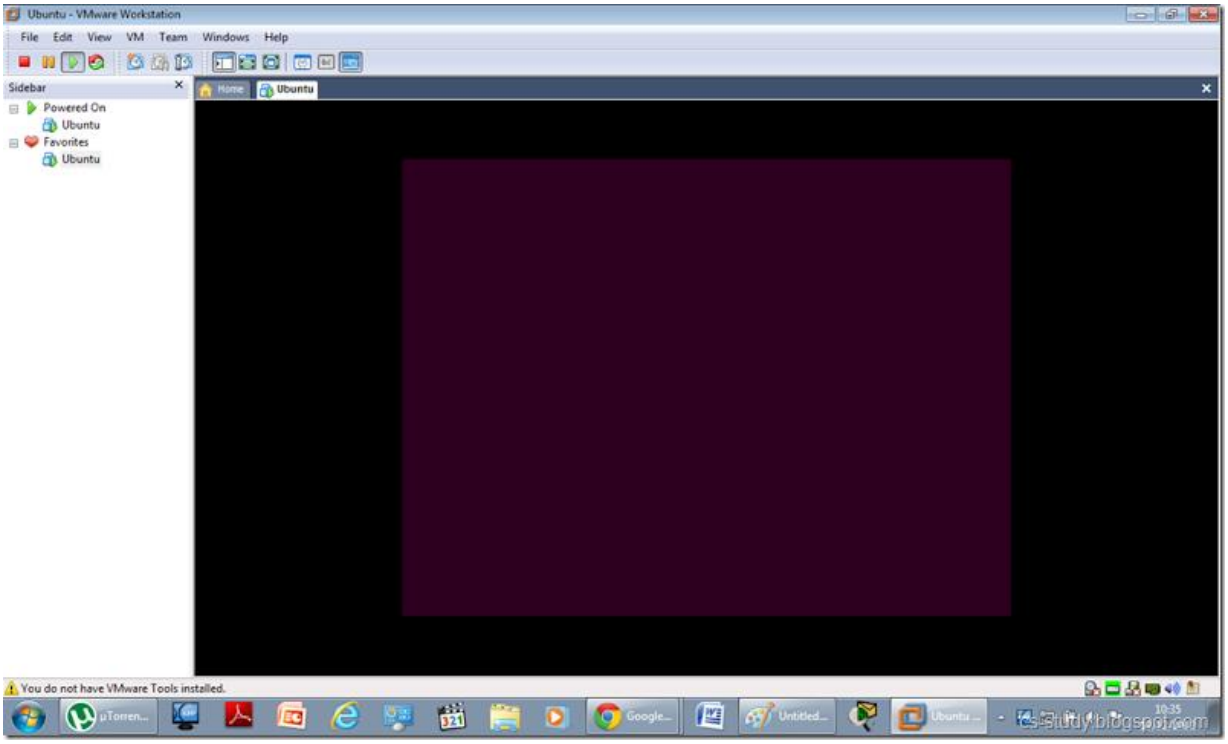
This is the important part. Here you give the path of the ISO image of the ubuntu file by selecting the 'Use ISO image' option and giving the path by browsing it as follows.



Now, when you click on the start this virtual machine .



For the very first time, when it loads it will start installation. I have done that already. So, my ubuntu is loading.



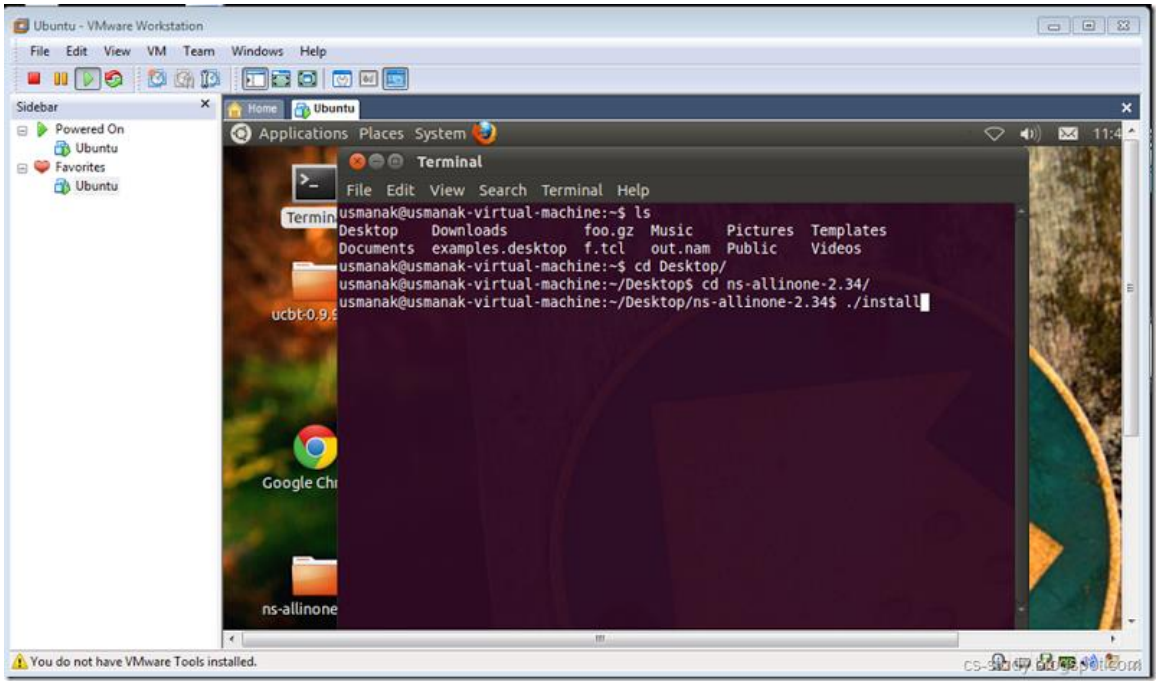
You can also use FULL SCREEN mode of virtual machine.

Now, let us install ns2. Download ns2 from the following link.

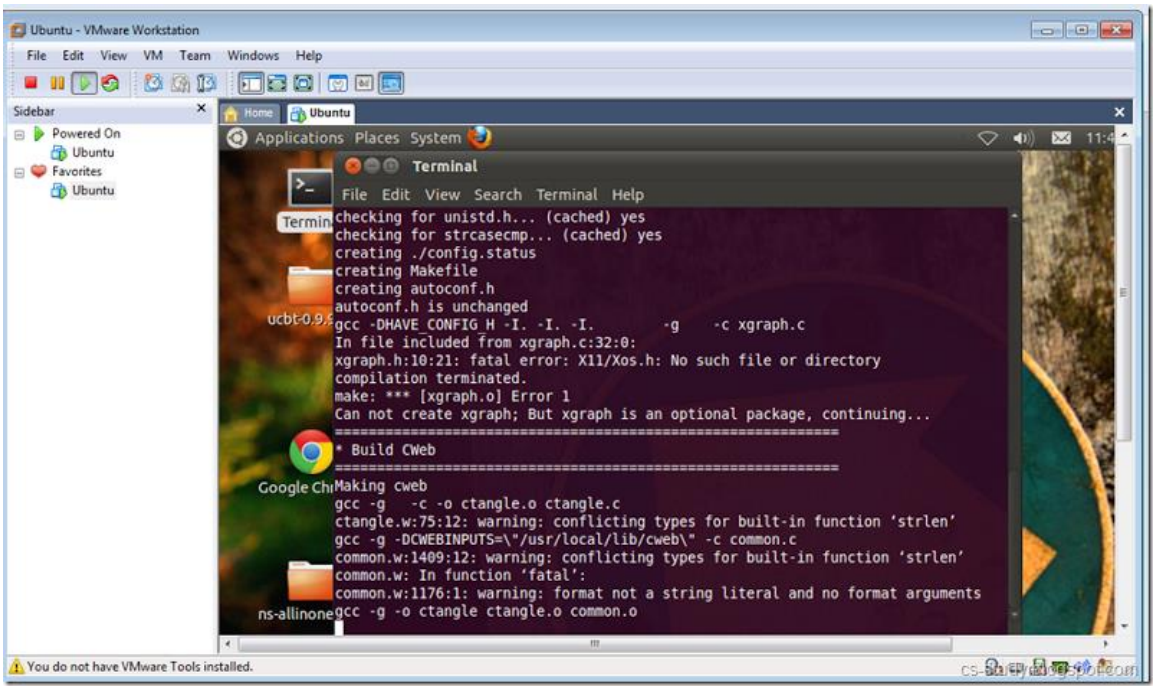
3. **<u>NS2 Download</u>**

It is in zipped folder. You will have to extract it twice. Then,
Go to the ns2 folder and run the install file by applying following command.

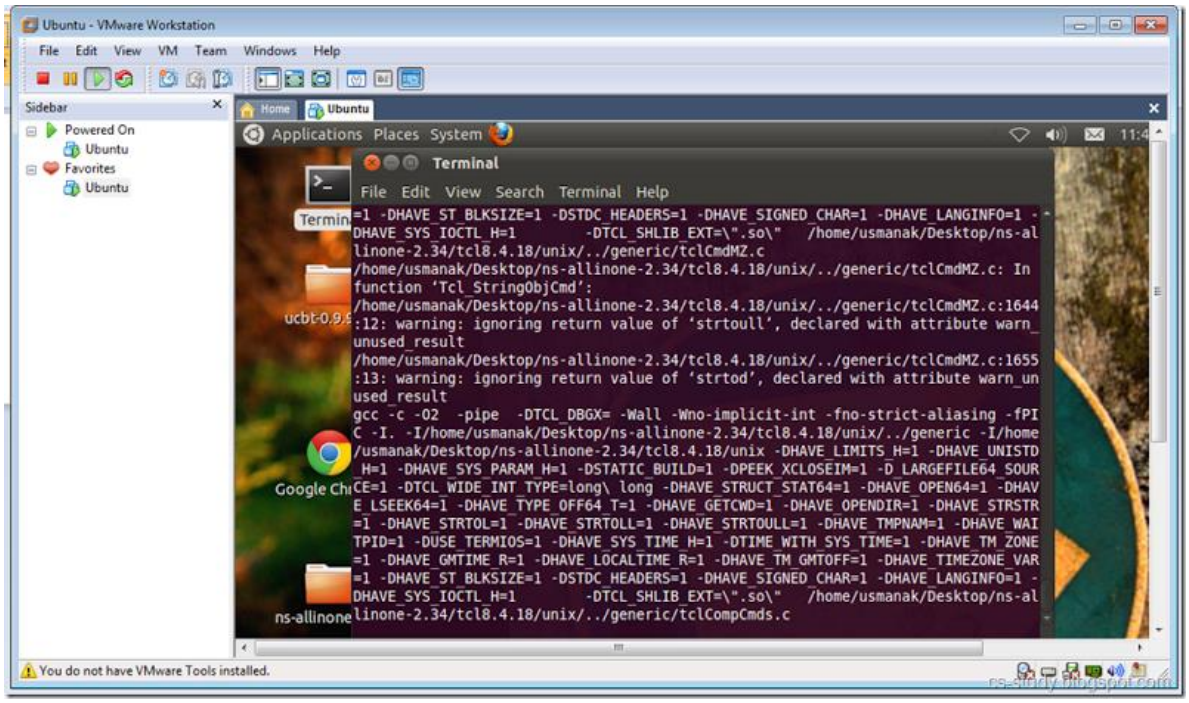
. /install



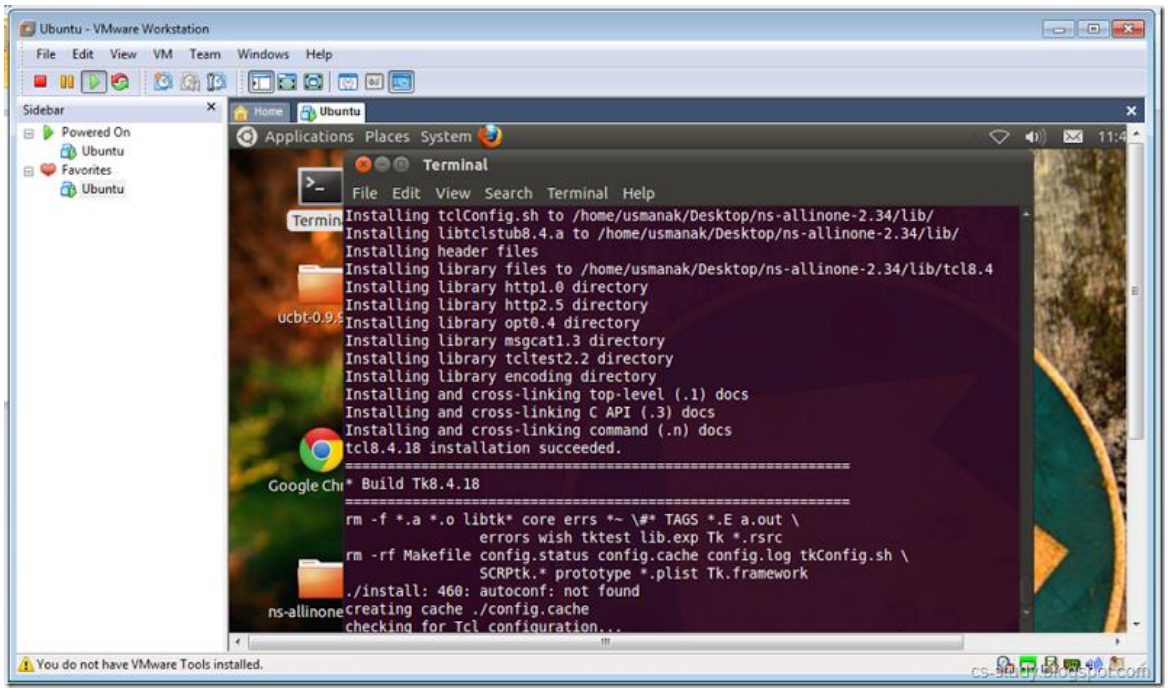
Installation starts. It will take a minute or two.


Ubuntu - VMware Workstation
File Edit View VM Team Windows Help
Sidebar
Powered On
Ubuntu
Favorites
Ubuntu
Home
Ubuntu
Applications Places System
Terminal
File Edit View Search Terminal Help
checking for unistd.h... (cached) yes
checking for strcasecmp... (cached) yes
creating ./config.status
creating Makefile
creating autoconf.h
autoconf.h is unchanged
gcc -DHAVE_CONFIG_H -I. -I. -I. -g -c xgraph.c
In file included from xgraph.c:32:0:
xgraph.h:10:21: fatal error: X11/Xos.h: No such file or directory
compilation terminated.
make: *** [xgraph.o] Error 1
Can not create xgraph; But xgraph is an optional package, continuing...
* Build CWeb
Making cweb
gcc -g -c -o ctangle.o ctangle.c
ctangle.w:75:12: warning: conflicting types for built-in function 'strlen'
gcc -g -DCWEBINPUTS=\"/usr/local/lib/cweb\" -c common.c
common.w:1409:12: warning: conflicting types for built-in function 'strlen'
common.w: In function 'fatal':
common.w:1176:1: warning: format not a string literal and no format arguments
gcc -g -o ctangle ctangle.o common.o
You do not have VMware Tools installed.

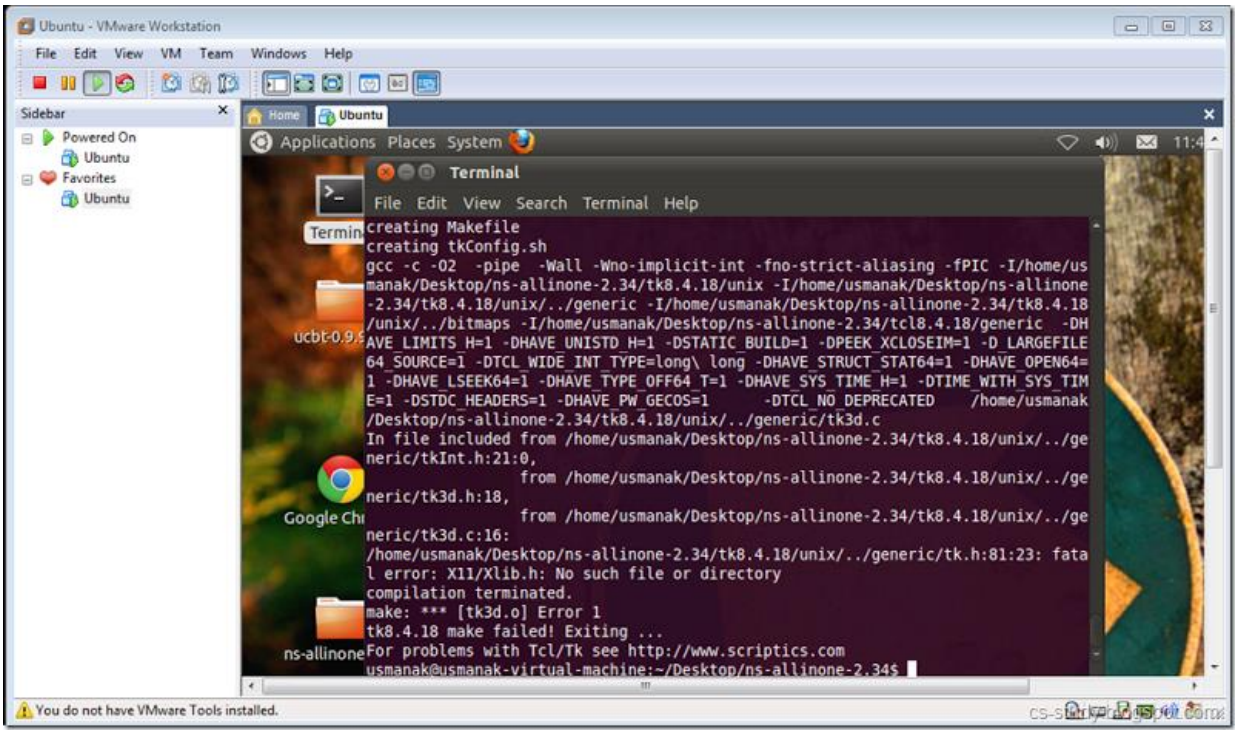
Continued…


Ubuntu - VMware Workstation
File Edit View VM Team Windows Help
Sidebar
Powered On
Ubuntu
Favorites
Ubuntu
Home
Ubuntu
Applications Places System
Terminal
File Edit View Search Terminal Help
=1 -DHAVE_ST_BLKSIZE=1 -DSTDC_HEADERS=1 -DHAVE_SIGNED_CHAR=1 -DHAVE_LANGINFO=1 -
DHAVE_SYS_IOCTL_H=1 -DTCL_SHLIB_EXT=\".so\" /home/usmanak/Desktop/ns-al
linone-2.34/tcl8.4.18/unix/../generic/tclCmdMZ.c
/home/usmanak/Desktop/ns-allinone-2.34/tcl8.4.18/unix/../generic/tclCmdMZ.c: In
function 'Tcl_StringObjCmd':
/home/usmanak/Desktop/ns-allinone-2.34/tcl8.4.18/unix/../generic/tclCmdMZ.c:1644
:12: warning: ignoring return value of 'strtoull', declared with attribute warn_
unused_result
/home/usmanak/Desktop/ns-allinone-2.34/tcl8.4.18/unix/../generic/tclCmdMZ.c:1655
:13: warning: ignoring return value of 'strtod', declared with attribute warn_un
used_result
gcc -c -O2 -pipe -DTCL_DBGX= -Wall -Wno-implicit-int -fno-strict-aliasing -fPI
C -I. -I/home/usmanak/Desktop/ns-allinone-2.34/tcl8.4.18/unix/../generic -I/home
/usmanak/Desktop/ns-allinone-2.34/tcl8.4.18/unix -DHAVE_LIMITS_H=1 -DHAVE_UNISTD
_H=1 -DHAVE_SYS_PARAM_H=1 -DSTATIC_BUILD=1 -DPEEK_XCLOSEIM=1 -D_LARGEFILE64_SOUR
CE=1 -DTCL_WIDE_INT_TYPE=long\ long -DHAVE_STRUCT_STAT64=1 -DHAVE_OPEN64=1 -DHAV
E_LSEEK64=1 -DHAVE_TYPE_OFF64_T=1 -DHAVE_GETCWD=1 -DHAVE_OPENDIR=1 -DHAVE_STRSTR
=1 -DHAVE_STRTOL=1 -DHAVE_STRTOLL=1 -DHAVE_STRTOULL=1 -DHAVE_TMPNAM=1 -DHAVE_WAI
TPID=1 -DUSE_TERMIOS=1 -DHAVE_SYS_TIME_H=1 -DTIME_WITH_SYS_TIME=1 -DHAVE_TM_ZONE
=1 -DHAVE_GMTIME_R=1 -DHAVE_LOCALTIME_R=1 -DHAVE_TM_GMTOFF=1 -DHAVE_TIMEZONE_VAR
=1 -DHAVE_ST_BLKSIZE=1 -DSTDC_HEADERS=1 -DHAVE_SIGNED_CHAR=1 -DHAVE_LANGINFO=1 -
DHAVE_SYS_IOCTL_H=1 -DTCL_SHLIB_EXT=\".so\" /home/usmanak/Desktop/ns-al
linone-2.34/tcl8.4.18/unix/../generic/tclCompCmds.c
You do not have VMware Tools installed.

……

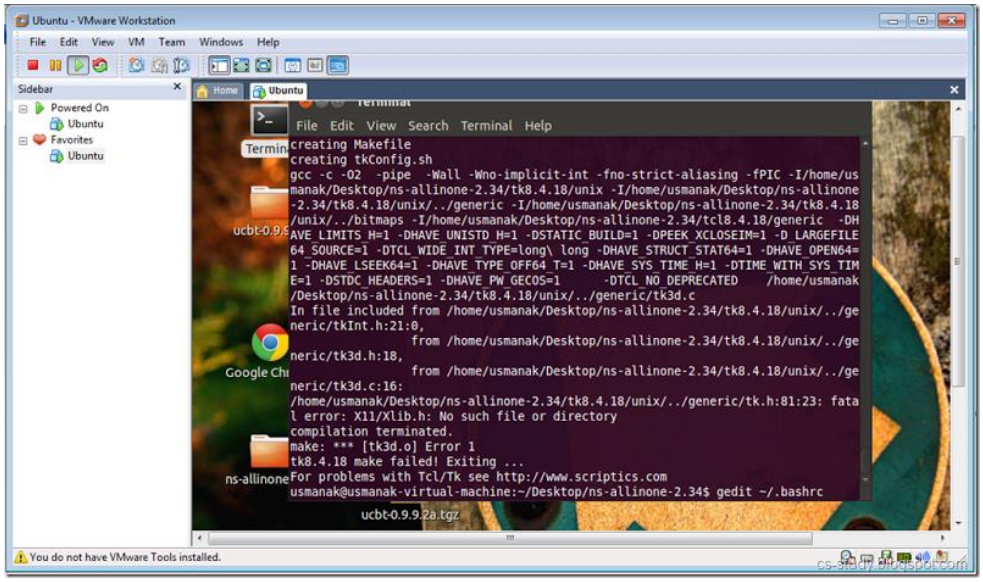


And it ends. Do not worry about the error messages. It will be fine.

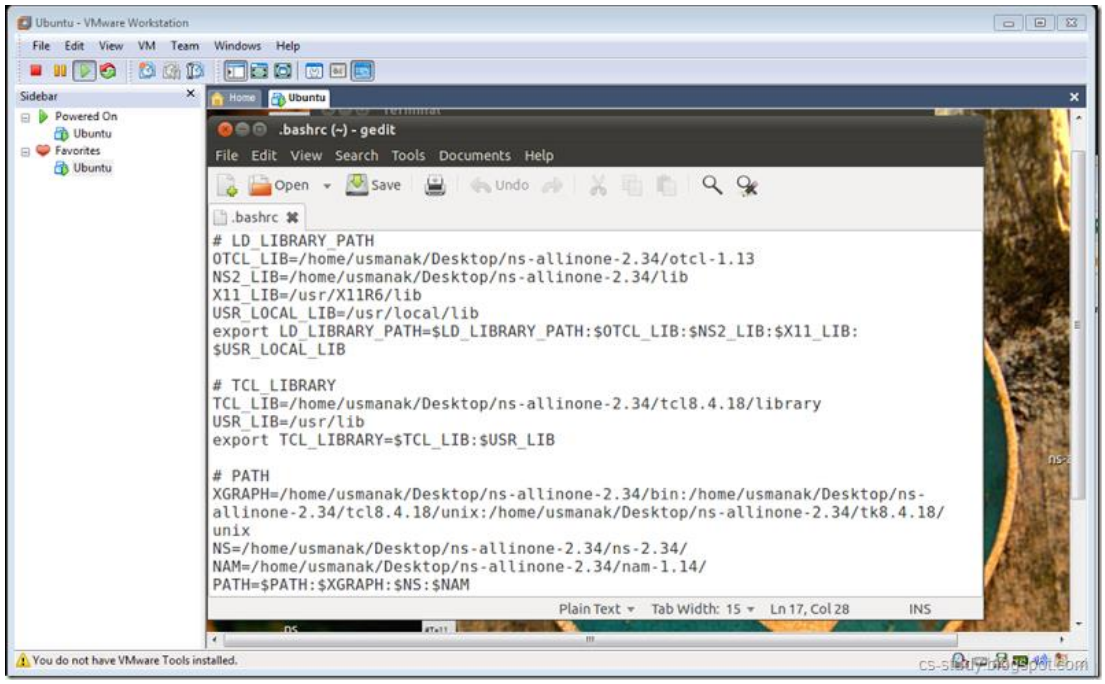


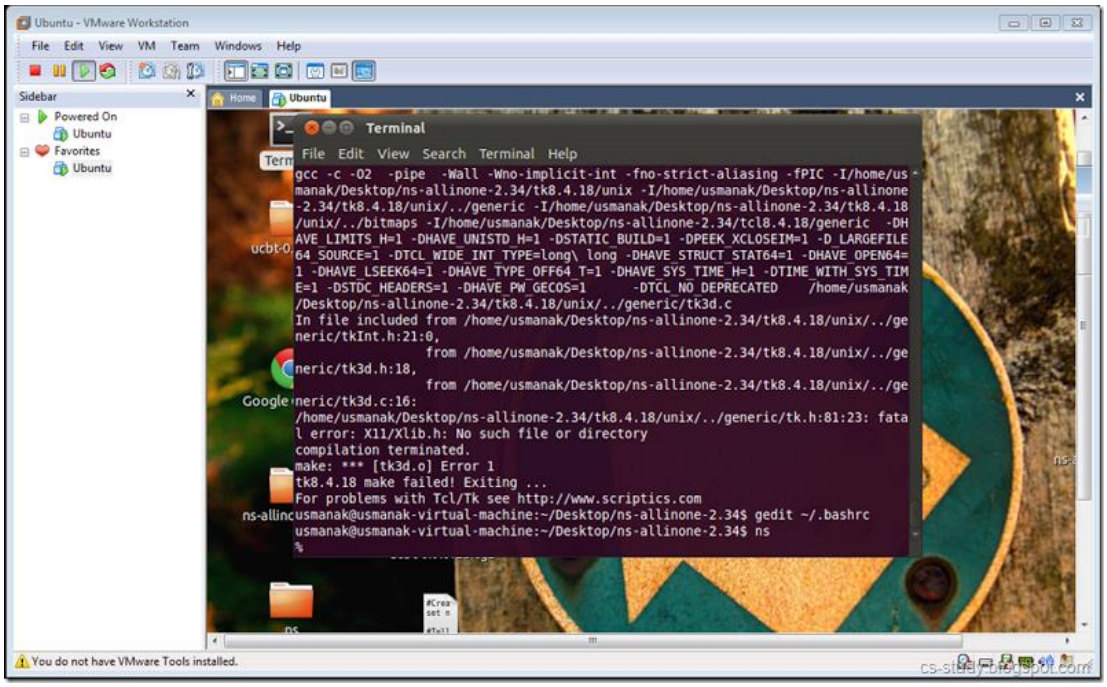
Now, let us set environment variables and set the paths.

Run, command 'gedit ~/.bashrc'.



Set the following paths. Set the right folder hierarchy according to your own like I have set mine as follows.



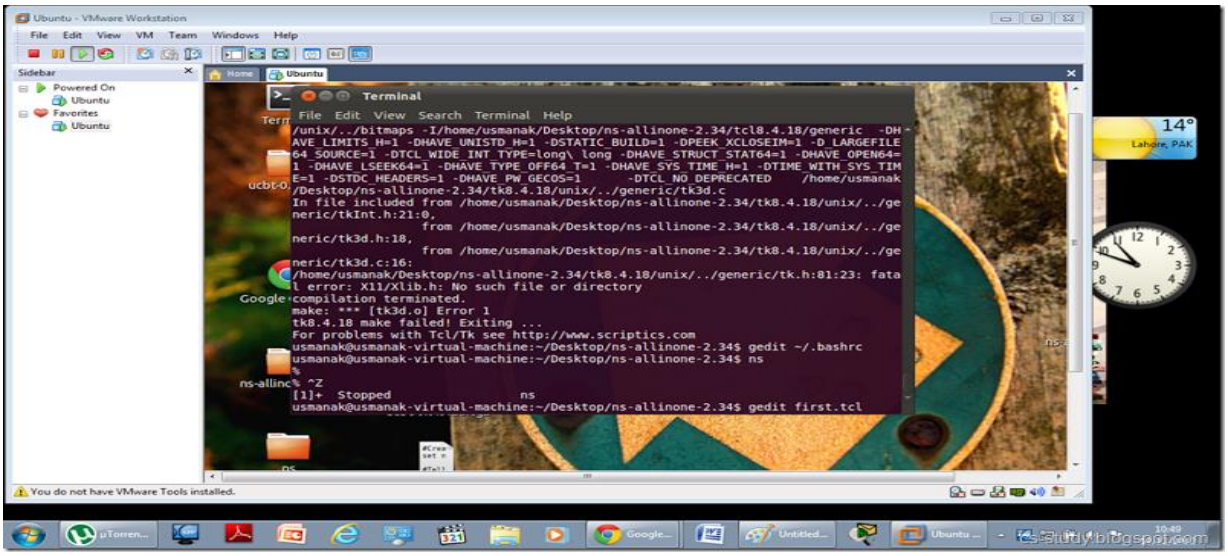


Now, if we write ns command and press enter. If we see this % sign, this means that our ns2 installation is successful.

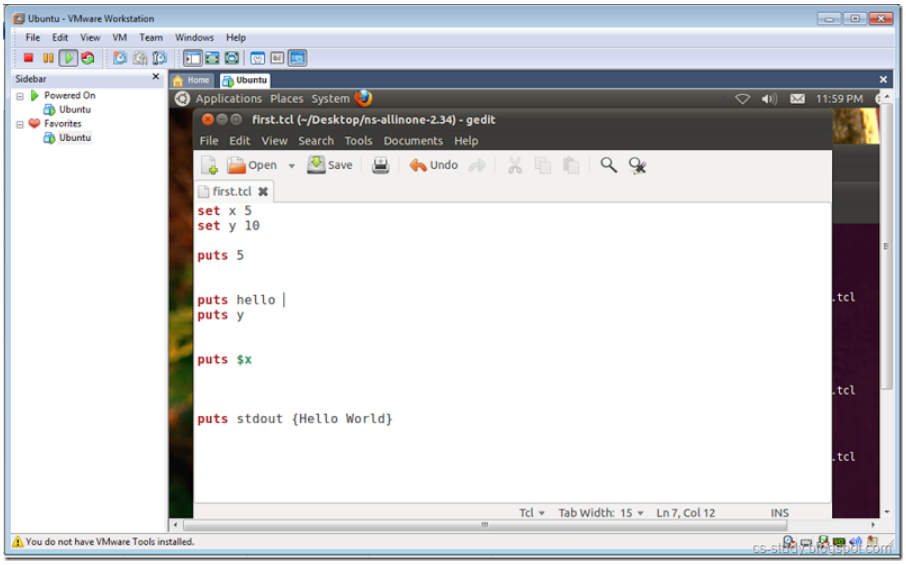
Hurrah. FYI, if you dont see the % sign and instead see some errors, run the following command.

sudo apt-get install ns2 nam xgraph

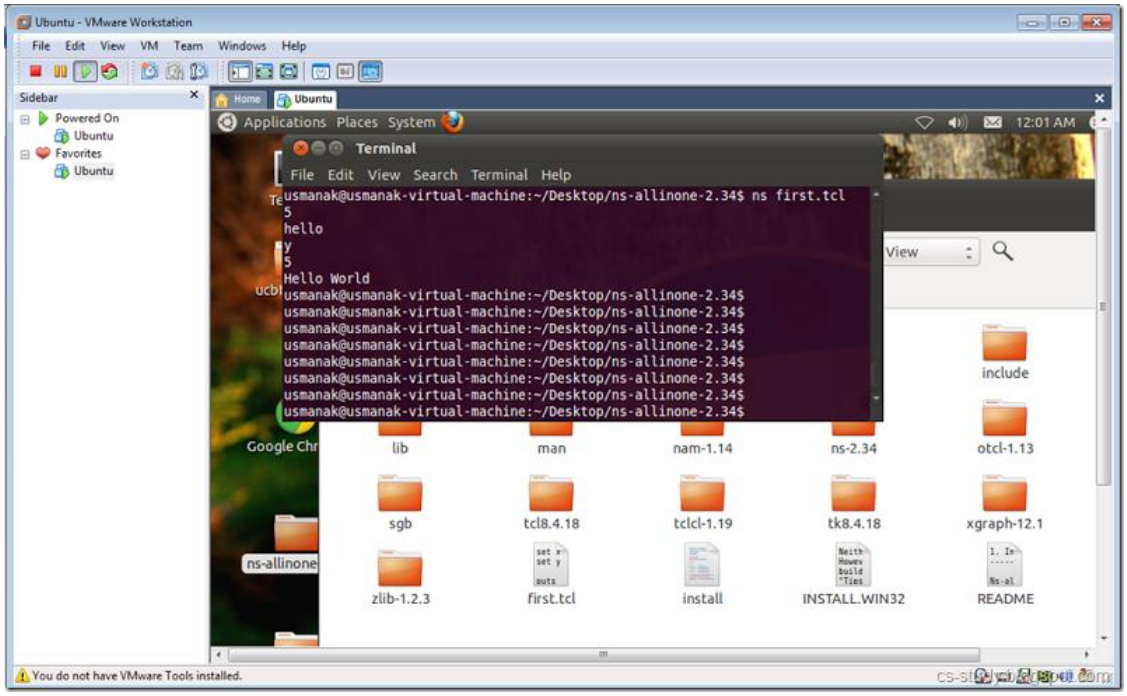
Provide the password. It will download some packaged online and install ns2. Now, let us create a file. Write some TCL code and run it. Just for fun :)



Let us write some code.



Now, let us run it as follows.



Lab 14 & 15 – Traffic Flow on Nodes in NS2

Objectives

The objective of this lab is to create traffic flow in NS2.

Lab Tasks

Task 1: Create Nodes and show traffic flow.

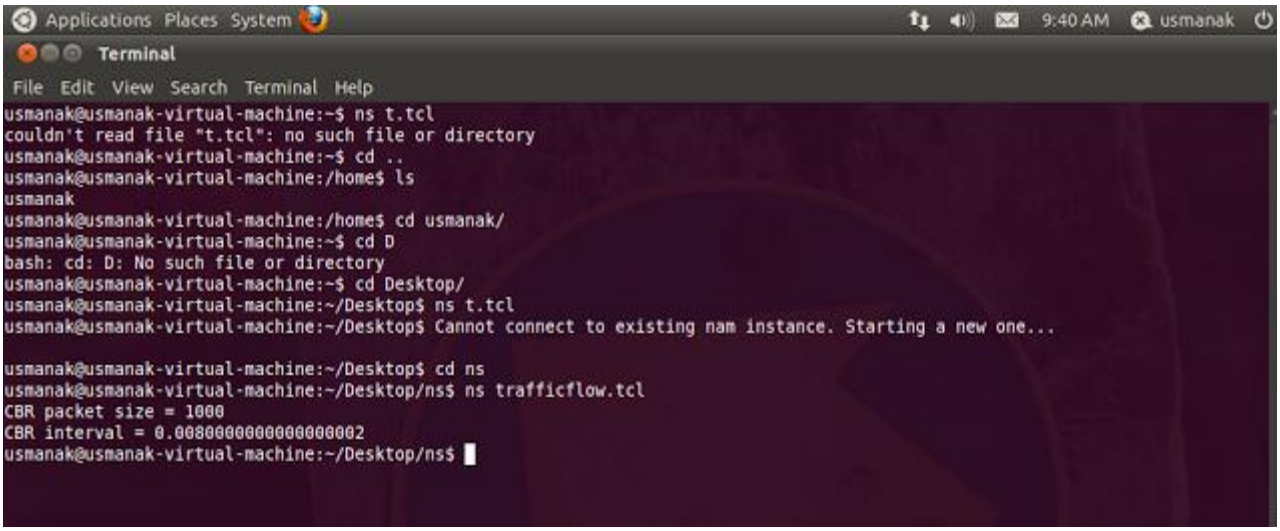
In this tutorial, we are going to create some nodes and then enable the flow of traffic on those nodes. There are two ways to flow the traffic i.e. through TCP and UDP. We will use both in this tutorial.

```
#Create a simulator object
set ns [new Simulator]

#Define different colors for data flows (for NAM)
$ns color 1 Blue
$ns color 2 Red
$ns color 3 Black

#Open the NAM trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
proc finish {} {
	global ns nf
	$ns flush-trace
	#Close the NAM trace file
	close $nf
	#Execute NAM on the trace file
	exec nam out.nam &
	exit 0
}

#Create four nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
```

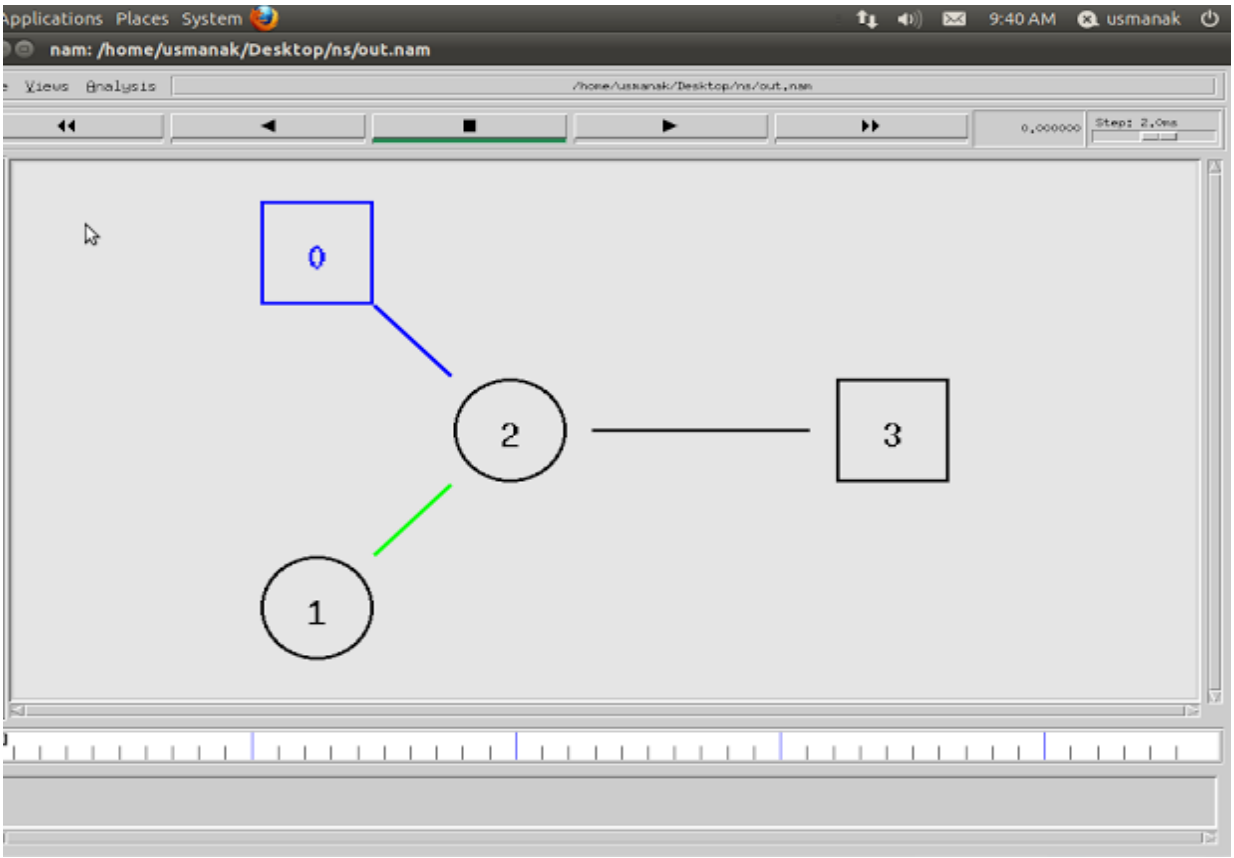
```
$n3 shape "square"
$n3 color "black"
$n0 shape "square"
$n0 color "blue"

#Create links between the nodes
$ns duplex-link $n0 $n2 2Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms RED
$ns duplex-link $n2 $n3 1.7Mb 20ms RED

$ns duplex-link-op $n0 $n2 color "blue"
$ns duplex-link-op $n0 $n2 label "cs-study"

$ns duplex-link-op $n1 $n2 color "green"
$ns duplex-link-op $n1 $n2 label "easylearning"

#Set Queue Size of link (n2-n3) to 10
$ns queue-limit $n2 $n3 10

#Give node position (for NAM)
$ns duplex-link-op $n0 $n2 orient right-down
$ns duplex-link-op $n1 $n2 orient right-up
$ns duplex-link-op $n2 $n3 orient right

#Monitor the queue for link (n2-n3). (for NAM)
$ns duplex-link-op $n2 $n3 queuePos 0.5

#Setup a TCP connection
set tcp [new Agent/TCP]
$tcp set class_ 3
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink]
$ns attach-agent $n3 $sink
$ns connect $tcp $sink
$tcp set fid_ 1

#Setup a FTP over TCP connection
```

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set type_ FTP

#Setup a UDP connection
set udp [new Agent/UDP]
$ns attach-agent $n1 $udp
set null [new Agent/Null]
$ns attach-agent $n3 $null
$ns connect $udp $null
$udp set fid_ 2

#Setup a CBR over UDP connection
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_ CBR
$cbr set packet_size_ 1000
$cbr set rate_ 1mb
$cbr set random_ false

#Schedule events for the CBR and FTP agents
$ns at 0.1 "$cbr start"
$ns at 1.0 "$ftp start"
$ns at 4.0 "$ftp stop"
$ns at 4.5 "$cbr stop"

#Detach tcp and sink agents (not really necessary)
$ns at 4.5 "$ns detach-agent $n0 $tcp ; $ns detach-agent $n3 $sink"

#Call the finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"

#Print CBR packet size and interval
puts "CBR packet size = [$cbr set packet_size_]"
puts "CBR interval = [$cbr set interval_]"
#Run the simulation
$ns run
```

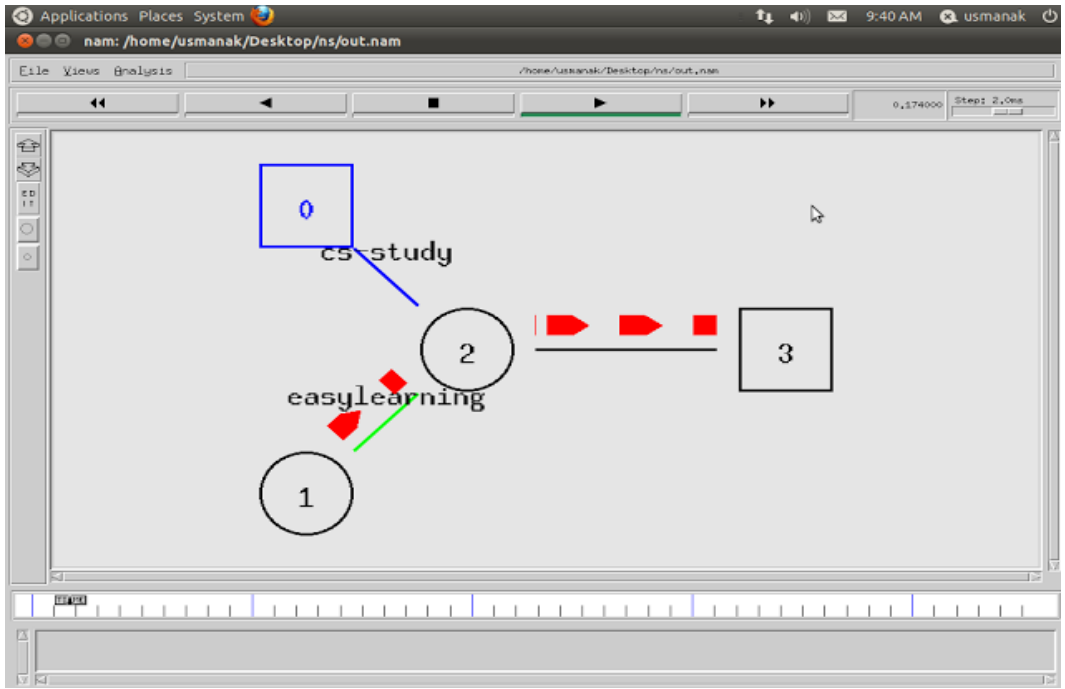
The output of the above code is as follows. Run the program by typing ‘ns trafficflow.tcl’ in terminal.



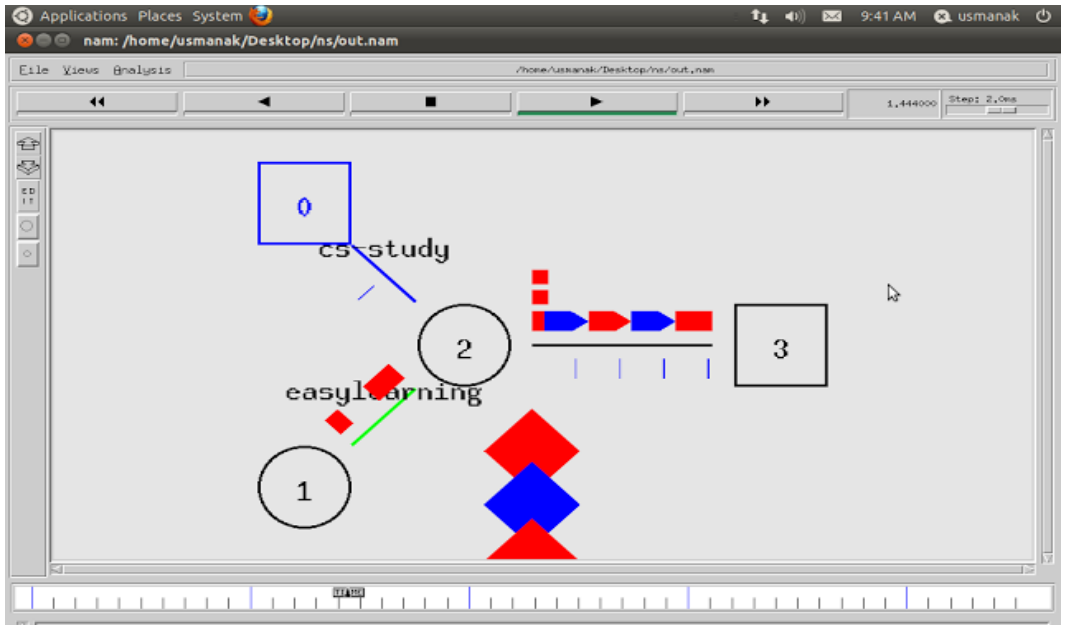
After it gets executed, the following output is displayed in nam.

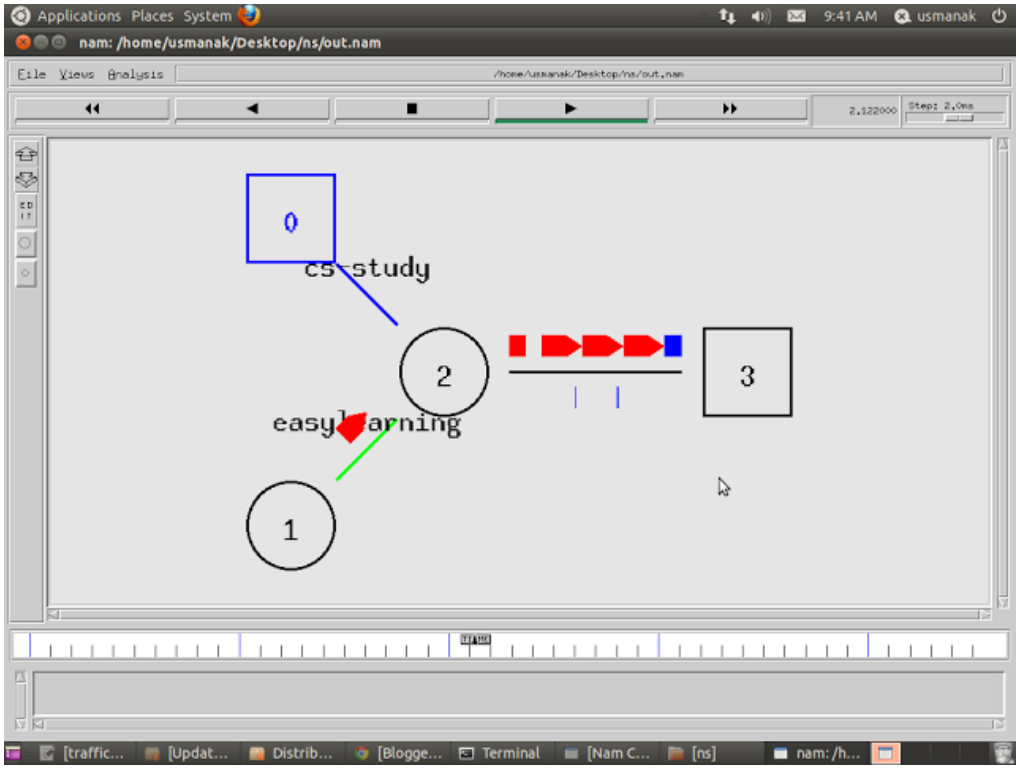


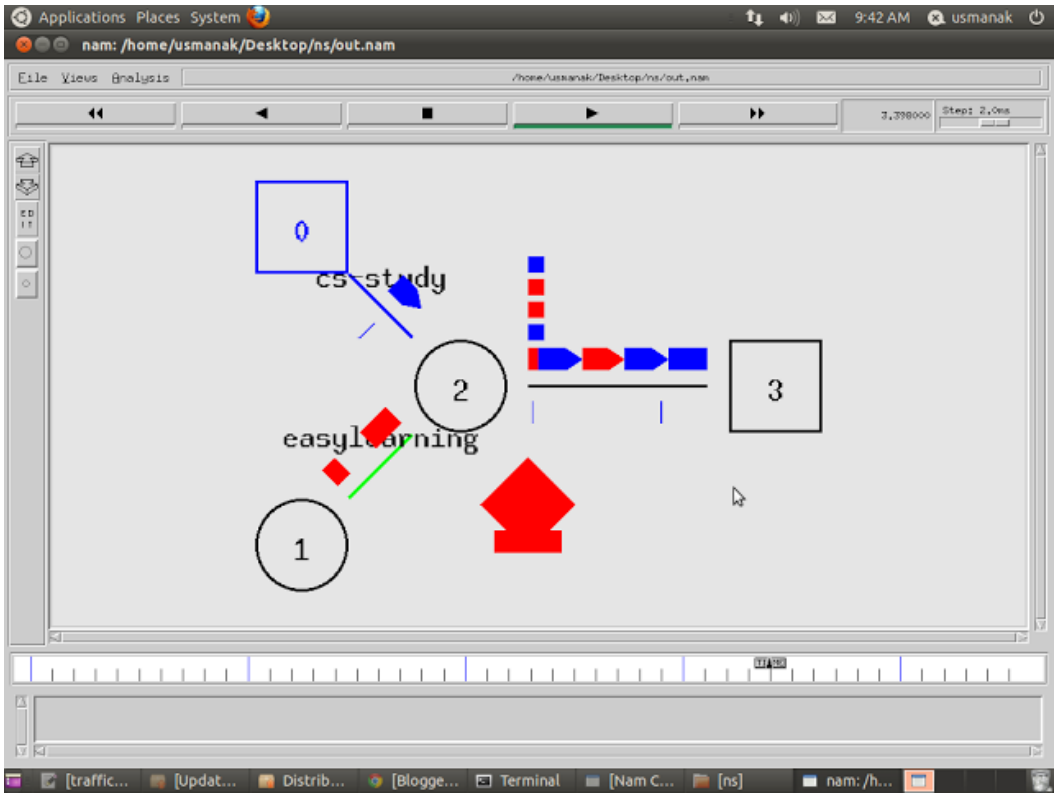
Now, click on the Play button which is underlined green.



We can see the traffic flow as well as the traffic which are being dropped on node 2 because the limits of the node to hold the packets is limited. Thats why when the queue is full and there is no more capacity available, the packet drops.




Applications Places System
9:41 AM
usmanak
nam: /home/usmanak/Desktop/ns/out.nam
File Views Analysis
0
cs-study
2
3
easylearning
1


Applications Places System
9:42 AM
usmanak
nam: /home/usmanak/Desktop/ns/out.nam
File Views Analysis
0
cs-study
2
3
easylearning
1

The little bars below the link between node 2 and node 3 indicates the acknowledgement that is being sent from node 3 to node 0 because it is using TCP.

Code Explained:

set *ns* [new Simulator]: generates an NS simulator object instance, and assigns it to variable *ns.*

The "Simulator" object has member functions that do the following:

Create compound objects such as nodes and links (described later)

Connect network component objects created (ex. attach-agent)

Set network component parameters (mostly for compound objects)

Create connections between agents (ex. make connection between a "tcp" and "sink")

Specify NAM display options

Most of member functions are for simulation setup (referred to as plumbing functions in the Overview section) and scheduling, however some of them are for the NAM display. The "Simulator" object member function implementations are located in the "ns-2/tcl/lib/ns-lib.tcl" file.

- ***$ns* color *fid color***: is to set color of the packets for a flow specified by the flow id (fid). This member function of "Simulator" object is for the NAM display, and has no effect on the actual simulation.

- ***$ns* namtrace-all *file-descriptor*:** This member function tells the simulator to record simulation traces in NAM input format. It also gives the file name that the trace will be written to later by the command *$ns* flush-trace. Similarly, the member function trace-all is for recording the simulation trace in a general format.

- **proc *finish* {}:** is called after this simulation is over by the command *$ns* at 5.0 "*finish*". In this function, post-simulation processes are specified.

- **set *n0* [*$ns* node]:** The member function node creates a node. A node in NS is compound object made of address and port classifiers (described in a later section). Users can create a node by separately creating an address and a port classifier objects and connecting them together. However, this member function of Simulator object makes the job easier. To see how a node is created, look at the files: "ns-2/tcl/libs/ns-lib.tcl" and "ns-2/tcl/libs/ns-node.tcl".

- ***$ns* duplex-link *node1 node2 bandwidth delay queue-type*:** creates two simplex links of specified bandwidth and delay, and connects the two specified nodes. In NS, the output queue of a node is implemented as a part of a link, therefore users should specify the queue-type when creating links. In the above simulation script, DropTail queue is used. If the reader wants to use a RED queue, simply replace the word DropTail with RED. The NS implementation of a link is shown in a later section. Like a node, a link is a compound object, and users can create its sub-objects and connect them and the nodes. Link source codes can be found in "ns-2/tcl/libs/ns-lib.tcl" and "ns-2/tcl/libs/ns-link.tcl" files. One thing to note is that you can insert error modules in a link component to simulate a lossy link (actually users can make and insert any network objects). Refer to the NS documentation to find out how to do this.

- ***$ns* queue-limit *node1 node2 number*:** This line sets the queue limit of the two simplex links that connect node1 and node2 to the number specified. At this point, the authors do not know how many of these kinds of member functions of Simulator objects are available and what they are. Please take a look at "ns-2/tcl/libs/ns-lib.tcl" and "ns-2/tcl/libs/ns-link.tcl", or NS documentation for more information.

- ***$ns* duplex-link-op *node1 node2 ..*:** The next couple of lines are used for the NAM display. To see the effects of these lines, users can comment these lines out and try the simulation.

  Now that the basic network setup is done, the next thing to do is to setup traffic agents such as TCP and UDP, traffic sources such as FTP and CBR, and attach them to nodes and agents respectively.

- **set *tcp* [new *Agent/TCP*]:** This line shows how to create a TCP agent. But in general, users can create any agent or traffic sources in this way. Agents and traffic sources are in fact basic objects (not compound objects), mostly implemented in C++ and linked to OTcl. Therefore, there are no specific Simulator object member functions that create these object instances. To create agents or traffic sources, a user should know the class names these objects (Agent/TCP, Agnet/TCPSink, Application/FTP and so on). This information can be found in the NS documentation or partly in this documentation. But one shortcut is to look at the "ns-2/tcl/libs/ns-default.tcl" file. This file contains the default configurable parameter value settings for available network objects. Therefore, it works as a good indicator of what kind of network objects are available in NS and what are the configurable parameters.

- ***$ns* attach-agent *node agent*:** The attach-agent member function attaches an agent object created to a node object. Actually, what this function does is call the attach member function of specified node, which attaches the given agent to itself. Therefore, a user can do the same thing by, for example, $n0 attach $tcp. Similarly, each agent object has a member function attach-agent that attaches a traffic source object to itself.

- ***$ns* connect *agent1 agent2*:** After two agents that will communicate with each other are created, the next thing is to establish a logical network connection between them. This line establishes a network connection by setting the destination address to each others' network and port address pair.

  Assuming that all the network configuration is done, the next thing to do is write a simulation scenario (i.e. simulation scheduling). The Simulator object has many scheduling member functions. However, the one that is mostly used is the following:

- ***$ns* at *time "string"*:** This member function of a Simulator object makes the scheduler (scheduler_ is the variable that points the scheduler object created by [new Scheduler] command at the beginning of the script) to schedule the execution of the specified string at given simulation time. For example, *$ns* at *0.1 "$cbr start"* will make the scheduler call a start member function of the CBR traffic source object, which starts the CBR to transmit data. In NS, usually a traffic source does not transmit actual data, but it notifies the underlying agent that it has some amount of data to transmit, and the agent, just knowing how much of the data to transfer, creates packets and sends them.

  After all network configuration, scheduling and post-simulation procedure specifications are done, the only thing left is to run the simulation. This is done by ***$ns*run.**

Lab 16 & 17 – Dynamic Routing in NS2

Objectives

The objective of this lab is to differentiate static routing from dynamic routing using DV protocol.

Lab Tasks

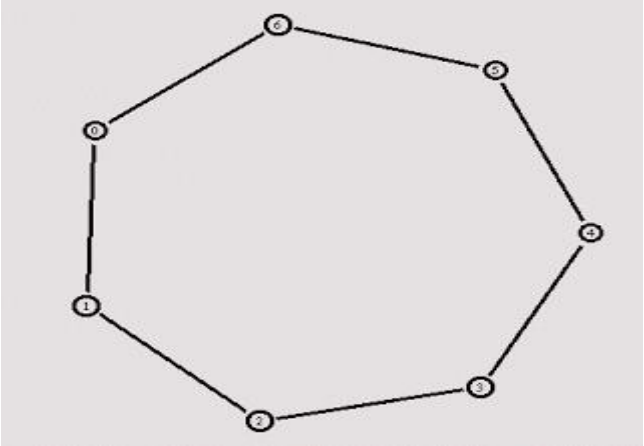
Create dynamic routing.

Here is the code for dynamically creating nodes and create traffic flow dynamically using protocol.

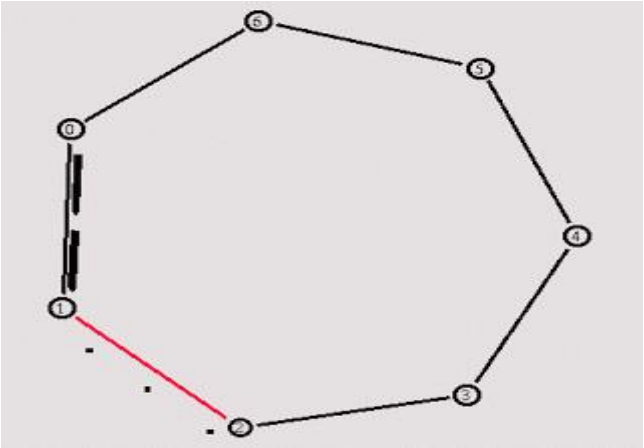
```
#Create a simulator object
set ns [new Simulator]

#Tell the simulator to use dynamic routing
$ns rtproto DV

#Open the nam trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
proc finish {} {
	global ns nf
	$ns flush-trace
 #Close the trace file
	close $nf
 #Execute nam on the trace file
	exec nam out.nam &
	exit 0
}

#Create seven nodes
for {set i 0} {$i < 7} {incr i} {
	set n($i) [$ns node]
}

#Create links between the nodes
for {set i 0} {$i < 7} {incr i} {
```

```
        $ns duplex-link $n($i) $n([expr ($i+1)%7]) 1Mb 10ms DropTail
}

#Create a UDP agent and attach it to node n(0)
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0

# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

#Create a Null agent (a traffic sink) and attach it to node n(3)
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0

#Connect the traffic source with the traffic sink
$ns connect $udp0 $null0

#Schedule events for the CBR agent and the network dynamics
$ns at 0.5 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 4.5 "$cbr0 stop"

#Call the finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"

#Run the simulation
$ns run
```

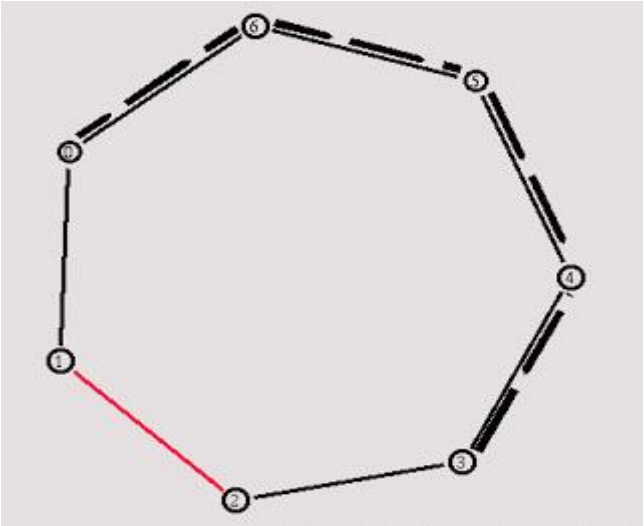
So, here is the output sequence.
First, nodes are created



Then, down the links and see how dynamically traffic flow by following the other path



Then, here is the final output.



Lab 18 – X-Graph in NS2

Objectives

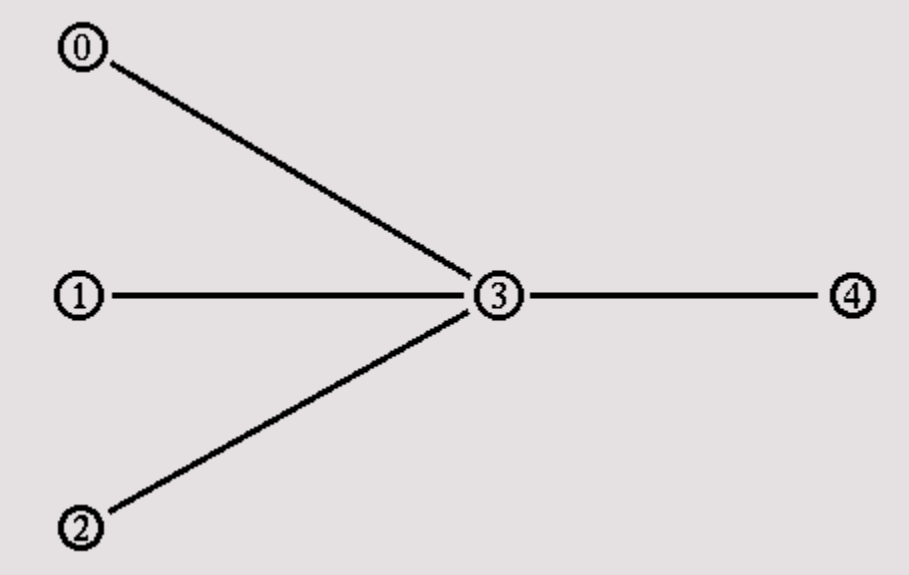
The objective of this lab is to produce X-graph in NS2.

Lab Tasks

Produce X-graph in NS2.

One part of the ns-allinone package is 'xgraph', a plotting program which can be used to create graphic representations of simulation results. In this section, I will show you a simple way how you can create output files in your Tcl scripts which can be used as data sets for xgraph. On the way there, I will also show you how to use traffic generators.
A note: The technique I present here is one of many possible ways to create output files suitable for xgraph. If you think there is a technique which is superior in terms of understandability (which is what I aim for in this tutorial), please let me know.

First of all, we create the following topology:



The following piece of code should look familiar to you by now if you read the first sections of this tutorial.

```
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
```

```
set n4 [$ns node]

$ns duplex-link $n0 $n3 1Mb 100ms DropTail
$ns duplex-link $n1 $n3 1Mb 100ms DropTail
$ns duplex-link $n2 $n3 1Mb 100ms DropTail
$ns duplex-link $n3 $n4 1Mb 100ms DropTail
```

We are going to attach traffic sources to the nodes n0, n1 and n2, but first we write a procedure that will make it easier for us to add the traffic sources and generators to the nodes:

```
proc attach-expoo-traffic { node sink size burst idle rate } {
#Get an instance of the simulator
set ns [Simulator instance]
#Create a UDP agent and attach it to the node
set source [new Agent/UDP]
$ns attach-agent $node $source
#Create an Expoo traffic agent and set its configuration parameters
set traffic [new Application/Traffic/Exponential]
$traffic set packet-size $size
$traffic set burst-time $burst
$traffic set idle-time $idle
$traffic set rate $rate
# Attach traffic source to the traffic generator
$traffic attach-agent $source
#Connect the source and the sink
$ns connect $source $sink
return $traffic
}
```

This procedure looks more complicated than it really is. It takes six arguments: A node, a previously created traffic sink, the packet size for the traffic source, the burst and idle times (for the exponential distribution) and the peak rate. For details about the Expoo traffic sources, please refer to the documentation.

First, the procedure creates a traffic source and attaches it to the node, then it creates a Traffic/Expoo object, sets its configuration parameters and attaches it to the traffic source, before eventually the source and the sink are connected. Finally, the procedure returns a handle for the traffic source. This procedure is a good example how reoccurring tasks like attaching a traffic source to several nodes can be handled. Now we use the procedure to attach traffic sources with

different peak rates to n0, n1 and n2 and to connect them to three traffic sinks on n4 which have to be created first:

```
set sink0 [new Agent/LossMonitor]
set sink1 [new Agent/LossMonitor]
set sink2 [new Agent/LossMonitor]
$ns attach-agent $n4 $sink0
$ns attach-agent $n4 $sink1
$ns attach-agent $n4 $sink2

set source0 [attach-expoo-traffic $n0 $sink0 200 2s 1s 100k]
set source1 [attach-expoo-traffic $n1 $sink1 200 2s 1s 200k]
set source2 [attach-expoo-traffic $n2 $sink2 200 2s 1s 300k]
```

In this example we use Agent/LossMonitor objects as traffic sinks, since they store the number of bytes received, which can be used to calculate the bandwidth.

Now we have to open three output files. The following lines have to appear 'early' in the Tcl script.

```
set f0 [open out0.tr w]
set f1 [open out1.tr w]
set f2 [open out2.tr w]
```

These files have to be closed at some point. We use a modified 'finish' procedure to do that.

```
proc finish {} {
    global f0 f1 f2
    #Close the output files
    close $f0
    close $f1
    close $f2
    #Call xgraph to display the results
    exec xgraph out0.tr out1.tr out2.tr -geometry 800x400 &
    exit 0
}
```

It not only closes the output files, but also calls xgraph to display the results. You may want to adapt the window size (800x400) to your screen size.

Now we can write the procedure which actually writes the data to the output files.

```
proc record {} {
    global sink0 sink1 sink2 f0 f1 f2
    #Get an instance of the simulator
    set ns [Simulator instance]
    #Set the time after which the procedure should be called again
    set time 0.5
    #How many bytes have been received by the traffic sinks?
    set bw0 [$sink0 set bytes_]
    set bw1 [$sink1 set bytes_]
    set bw2 [$sink2 set bytes_]
    #Get the current time
    set now [$ns now]
    #Calculate the bandwidth (in MBit/s) and write it to the files
    puts $f0 "$now [expr $bw0/$time*8/1000000]"
    puts $f1 "$now [expr $bw1/$time*8/1000000]"
    puts $f2 "$now [expr $bw2/$time*8/1000000]"
    #Reset the bytes_ values on the traffic sinks
    $sink0 set bytes_ 0
    $sink1 set bytes_ 0
    $sink2 set bytes_ 0
    #Re-schedule the procedure
    $ns at [expr $now+$time] "record"
}
```

This procedure reads the number of bytes received from the three traffic sinks. Then it calculates the bandwidth (in MBit/s) and writes it to the three output files together with the current time before it resets the bytes_ values on the traffic sinks. Then it re-schedules itself.

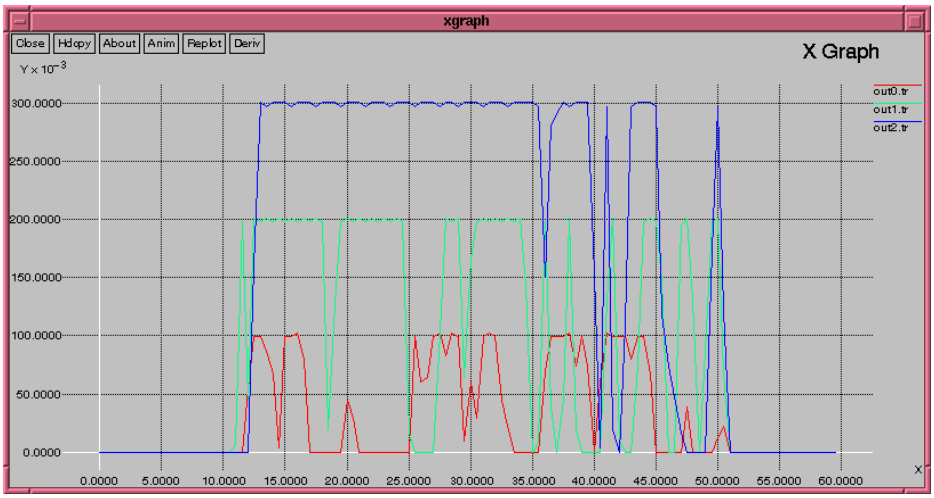
Time Scheduling of Simulation

We can now schedule the following events:

```
$ns at 0.0 "record"
$ns at 10.0 "$source0 start"
$ns at 10.0 "$source1 start"
$ns at 10.0 "$source2 start"
$ns at 50.0 "$source0 stop"
$ns at 50.0 "$source1 stop"
$ns at 50.0 "$source2 stop"
$ns at 60.0 "finish"
```

$ns run

First, the 'record' procedure is called, and afterwards it will re-schedule itself periodically every 0.5 seconds. Then the three traffic sources are started at 10 seconds and stopped at 50 seconds. At 60 seconds, the 'finish' procedure is called. You can find the full example script here.
When you run the simulation, an xgraph window should open after some time which should look similar to this one:



As you can see, the bursts of the first flow peak at 0.1Mbit/s, the second at 0.2Mbit/s and the third at 0.3Mbit/s. Now you can try to modify the 'time' value in the 'record' procedure. Set it to '0.1' and see what happens, and then try '1.0'. It is very important to find a good 'time' value for each simulation scenario.

Lab 19 – Installation of IT Guru

Objectives

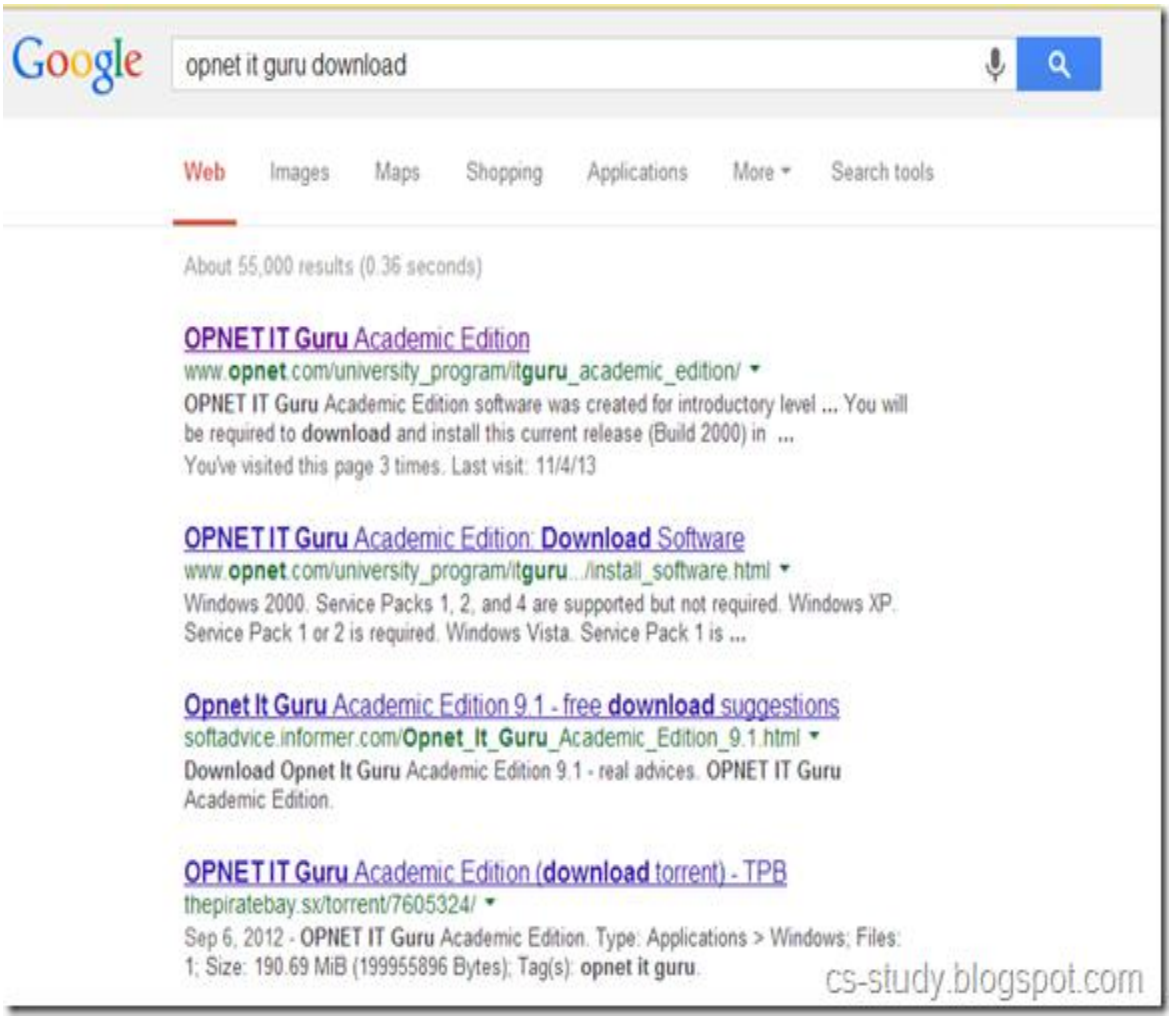
The objective of this lab is to install IT Guru software.

Lab Tasks

Task 1: Install IT Guru

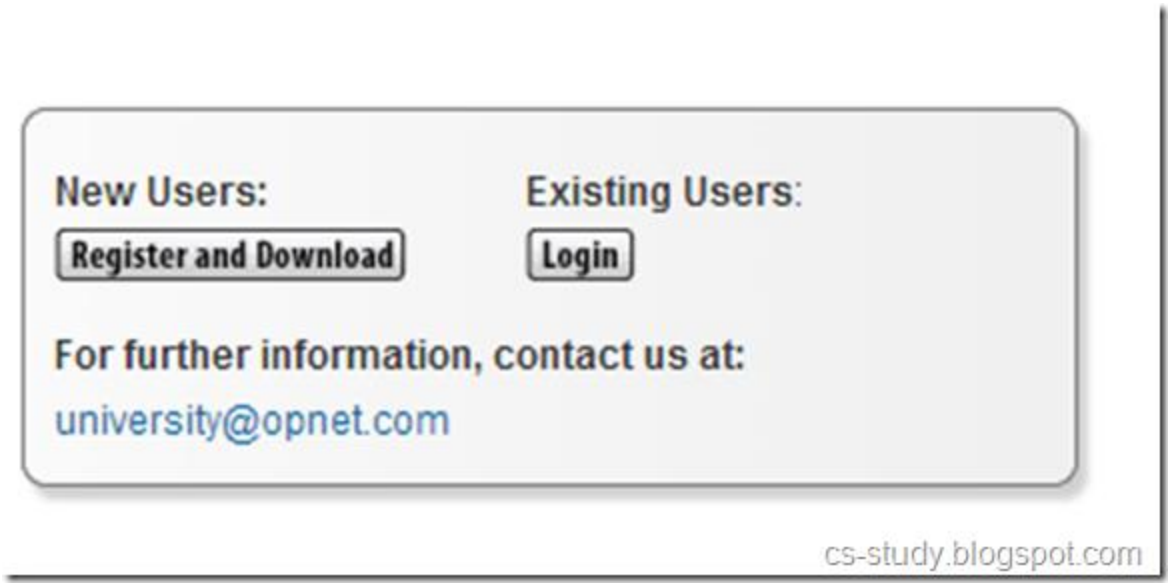
The installation of IT Guru can be a bit challenging. The first part is to do the registration online. Only then you will recieve the link to download opnet IT Guru. So, lets walk through it.

Type in Google as shown below. Click the first link.

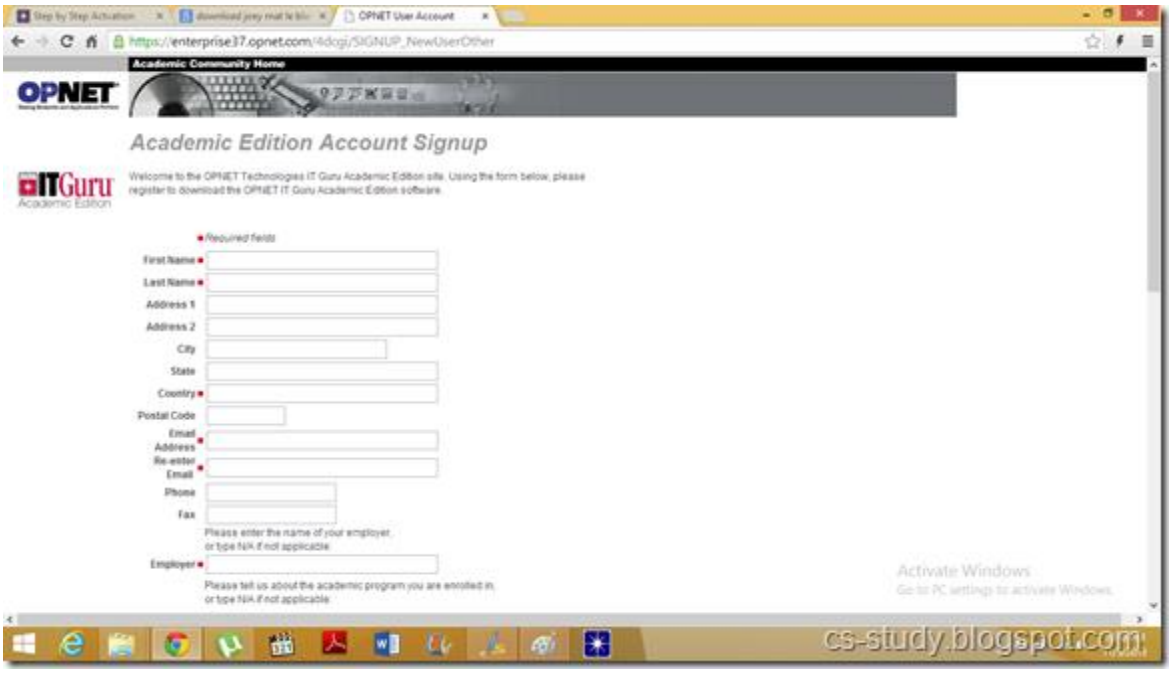


The following main IT Guru page will open.

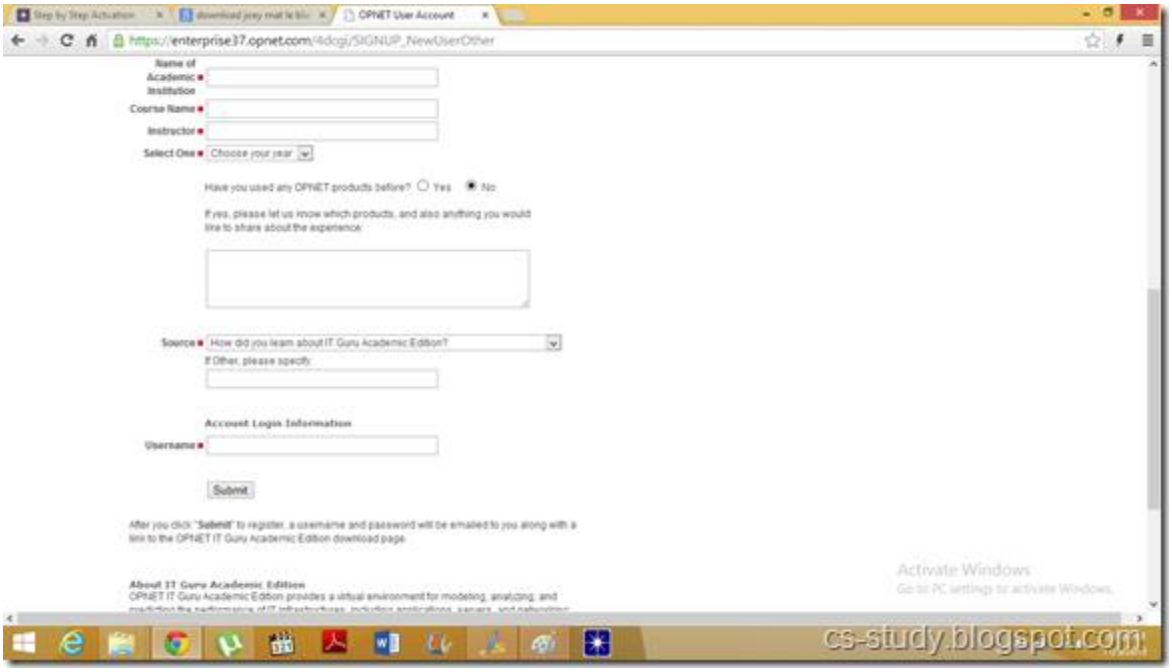
At the bottom, there is a button “Register and Download”. Click it.



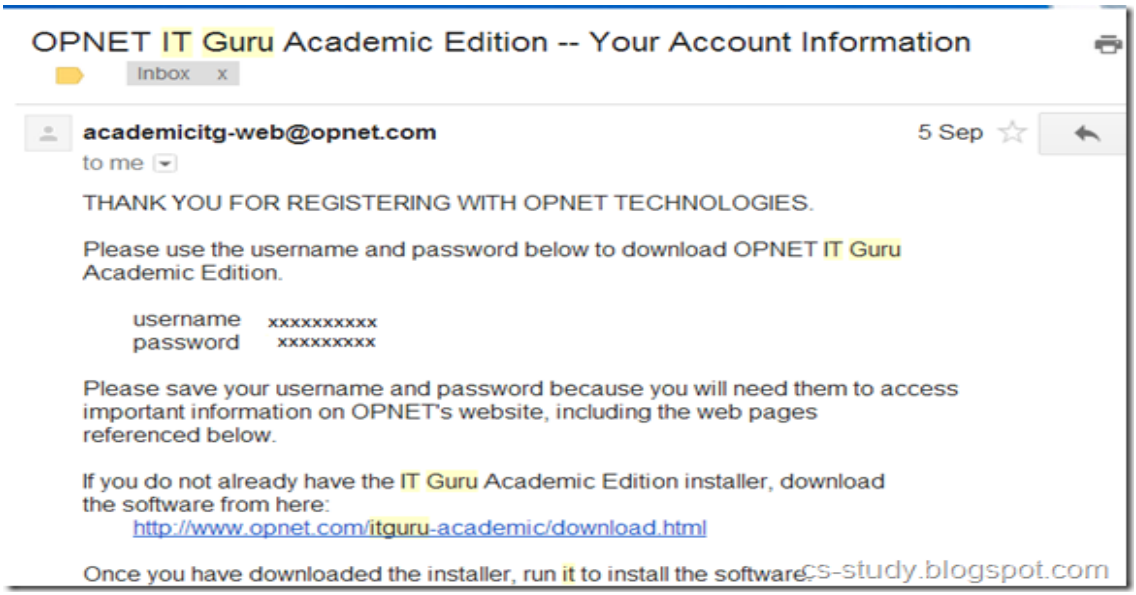
The following registration page will open. Fill it with correct credentials. Provide valid email id.



Registration continued. …



After registration, You will recieve an email from the OPNET IT Guru. In which you will recieve your password. And you will recieve your download link as shown below.



OPNET IT Guru Academic Edition -- Your Account Information

Inbox x

academicitg-web@opnet.com 5 Sep

to me

THANK YOU FOR REGISTERING WITH OPNET TECHNOLOGIES.

Please use the username and password below to download OPNET IT Guru Academic Edition.

username xxxxxxxxxx
password xxxxxxxxx

Please save your username and password because you will need them to access important information on OPNET's website, including the web pages referenced below.

If you do not already have the IT Guru Academic Edition installer, download the software from here:
http://www.opnet.com/itguru-academic/download.html

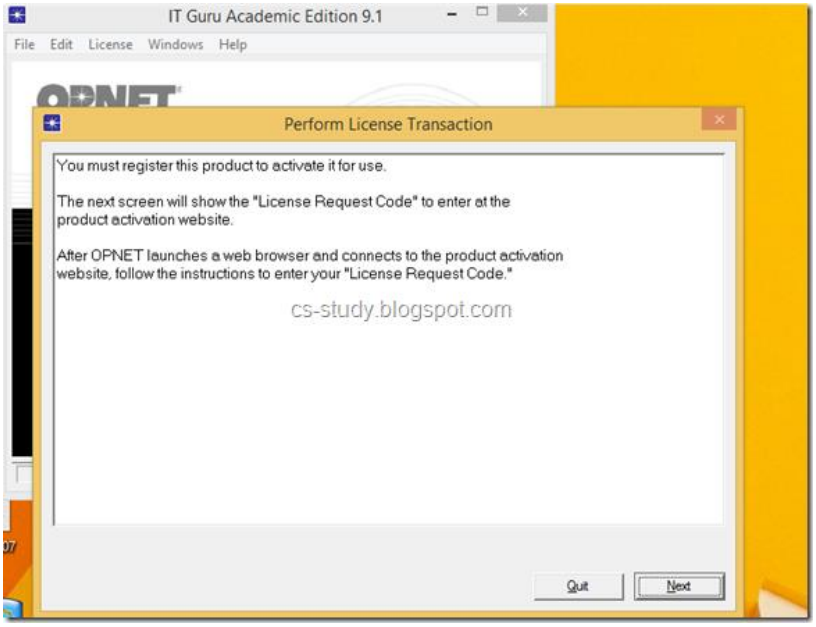
Once you have downloaded the installer, run it to install the software.

cs-study.blogspot.com

In the above diagram, instead of "xxxx….." , you will recieve your actual username and password.

After click on the link above. It will start downloading the setup. Its about 200 Mb. Download it and then install the IT Guru. When you are done with IT Guru installation, run it.

It will show you the screen to either quit or go in the license management. Click license management you will see the following screen.



IT Guru Academic Edition 9.1

File Edit License Windows Help

Perform License Transaction

You must register this product to activate it for use.

The next screen will show the "License Request Code" to enter at the product activation website.

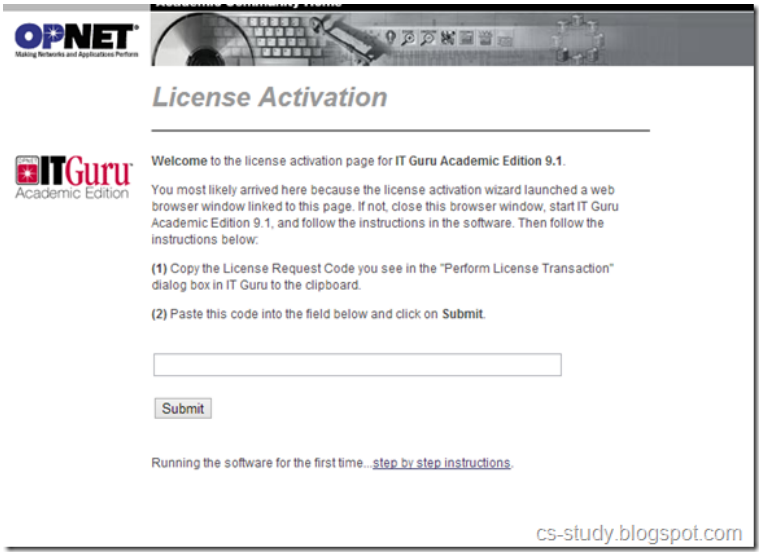
After OPNET launches a web browser and connects to the product activation website, follow the instructions to enter your "License Request Code."

cs-study.blogspot.com

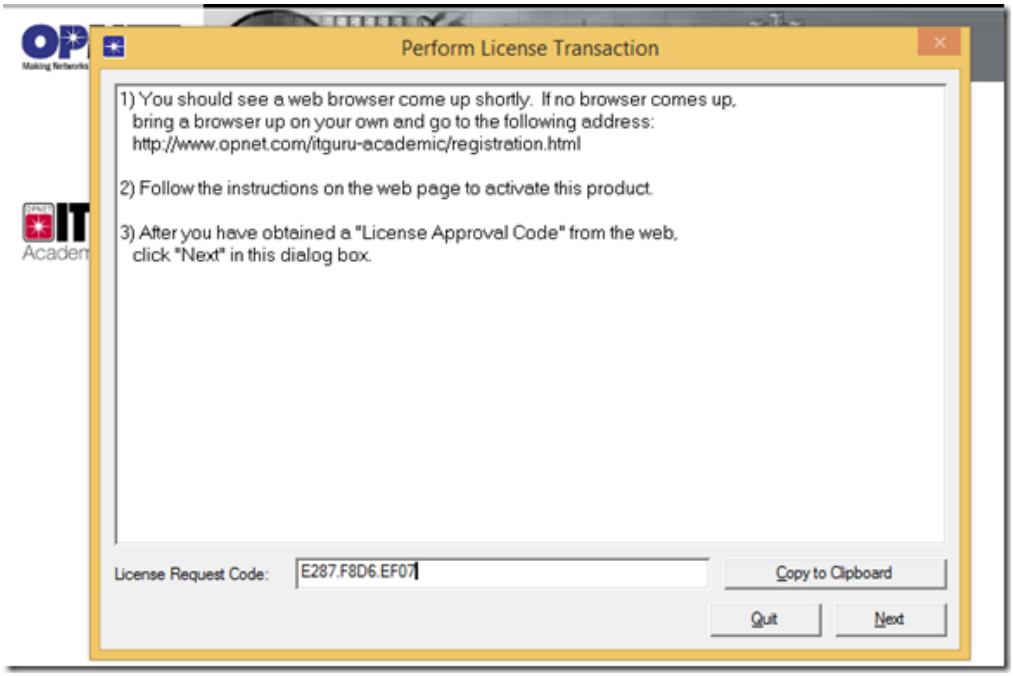
Quit Next

Click next, in the above screen. It will also open the web browser for you and ask for authentication. Here you will provide the username and password that you recieved in your email.

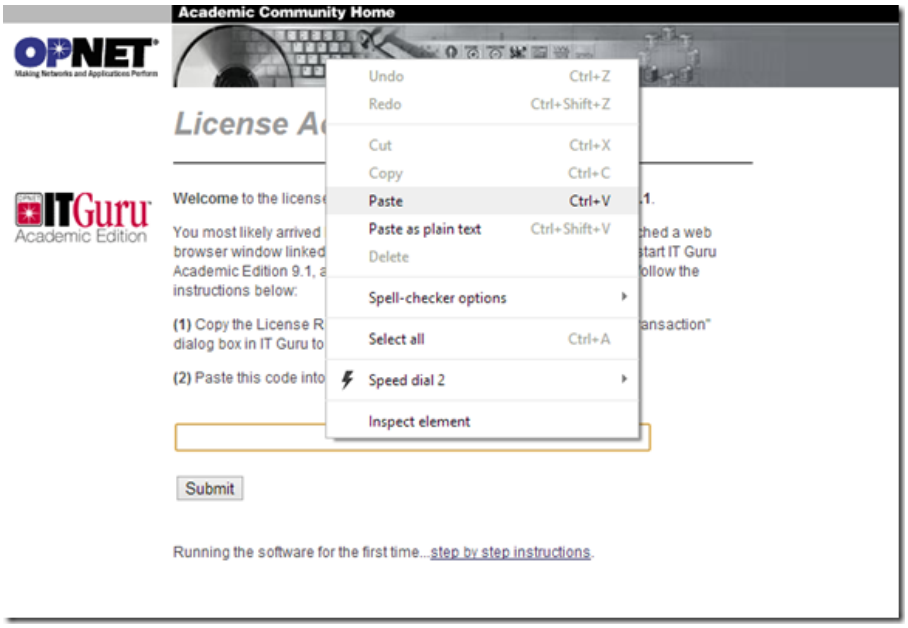
In the below window, you will enter the key which will be displayed on the IT Guru.



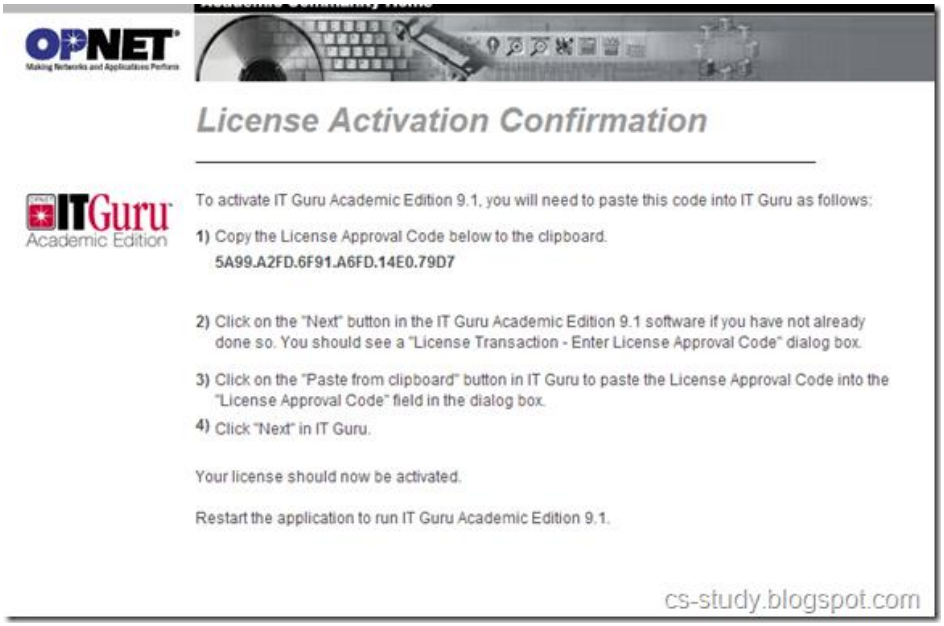
Provide the following key in the browser. As shown in the above window.



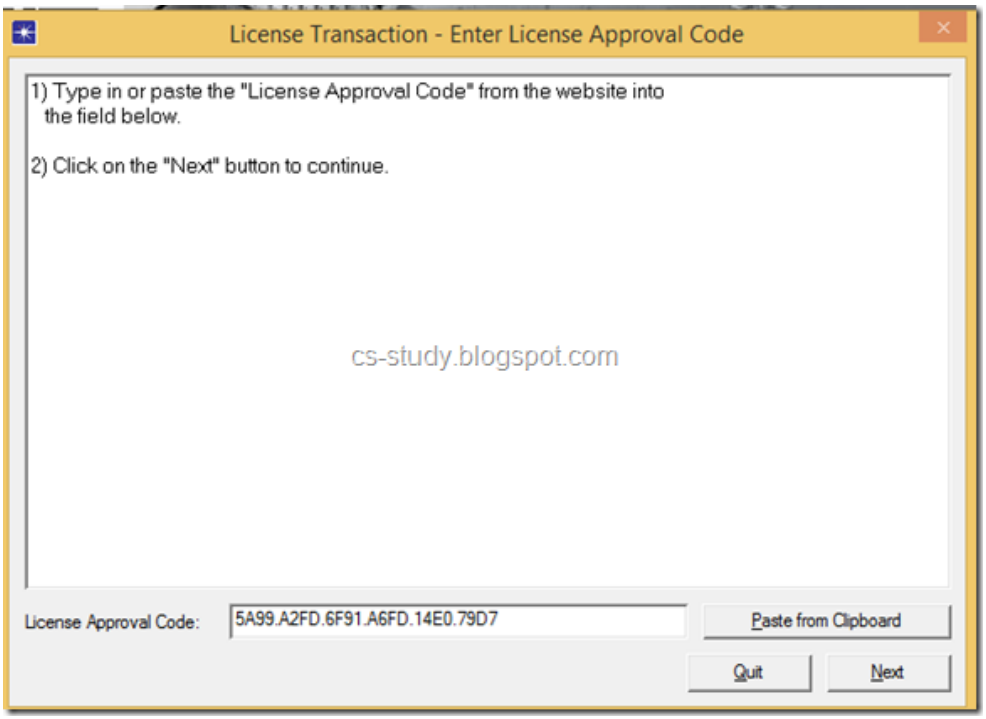
Paste it there.



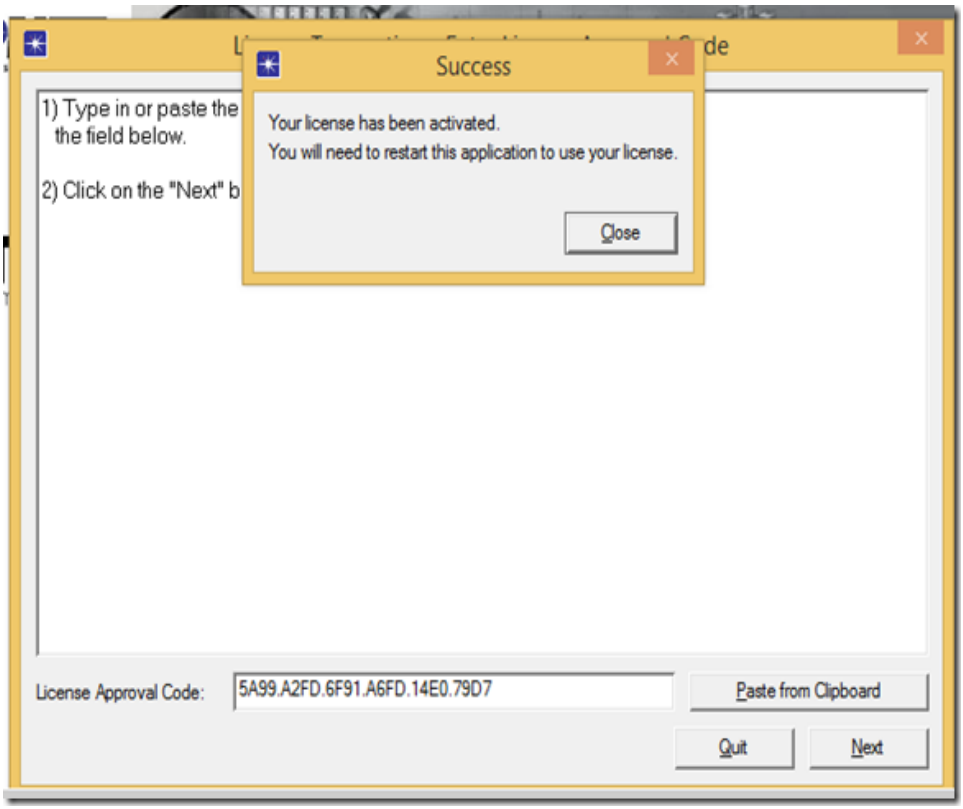
And click submit.



You will be provided with the key. You need to copy that key and paste it in the IT Guru as shown below and click next.



There, you go , IT Guru has been activated with key. Now, you can use it. Cheers.



Lab 20 – Simulation of a Small Office Network on IT Guru

Objectives

The objective of this lab is to implement and simulate small office network on IT Guru.

Lab Tasks

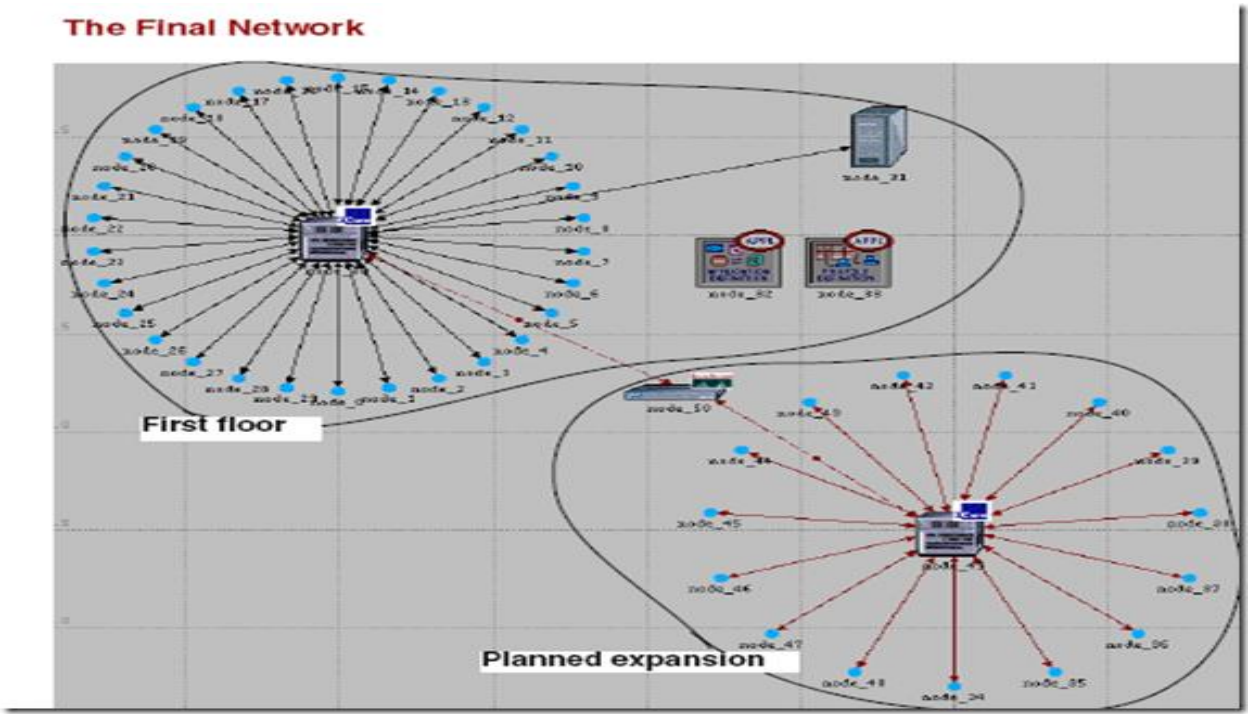
Task 1: Simulate office network on IT Guru.

In this lesson, you will learn how IT Guru can model organizational scaling by using the tool to model a real-world "what if" problem. You will learn how to use

IT Guru features to build and analyze network models.
In this lesson, you will

- Build a network quickly
- Collect statistics about network performance
- Analyze these statistics

In this lesson, you use the Project Editor to build a topology of a small internetwork, choose statistics to collect, run a simulation, and analyze the results.
In this lesson, you plan for the expansion of a small company's intranet. Currently, the company has a star topology network on the first floor of its office building and plans to add an additional star topology network on another floor. You will build and test this "what-if" scenario to ensure that the load added by the second network will not cause the network to fail.



Getting Started

When creating a new network model, you must first create a new **project** and **scenario**. A project is a group of related scenarios that each explore a different aspect of the network. Projects can contain multiple scenarios.
After you create a new project, you use the Startup Wizard to set up a new scenario. The options in the
Wizard let you
• Define the initial topology of the network
• Define the scale and size of the network
• Select a background map for the network.
• Associate an object palette with the scenario
Startup Wizard automatically appears each time you create a new project. The Startup Wizard allows you to define certain aspects of the network environment.
To use the Startup Wizard to set up a new scenario, do the following:
1 If IT Guru is not already running, start it.
2 Select **File** > **New...**.
3 Select **Project** from the pull-down menu and click
OK.
4 Name the project and scenario, as follows:
4.1 Name the project **<initials>_Sm_Int**
Include your initials in the project name to distinguish it from other versions of this project.
4.2 Name the scenario **first_floor**.
4.3 Click **OK**.
➥ The Startup Wizard opens.
Enter the values shown in the following table in the dialog boxes of the Startup Wizard:

Values to Enter in the Startup Wizard

Dialog Box Name	Value
1. Initial Topology	Select the default value: **Create Empty Scenario.**
2. Choose Network Scale	Select **Office.** Check the **Use Metric Units** box.
3. Specify Size	Select the default size: **100 m x 100 m**
4. Select Technologies	Include the **Sm_Int_Model_List** model family.
5. Review	Check values, then click **OK.**

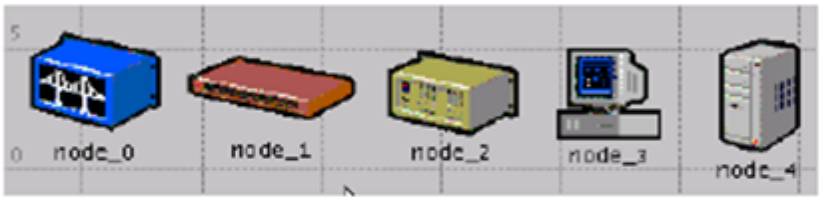
➥ A workspace of the size you specified is created. The object palette you specified opens in a separate window.

Creating the Network

Network models are created in the Project Editor using **nodes** and **links** from the **object palette**.

Node - A representation of a real-world network object that can transmit and receive information.

Nodes



Link - A communication medium that connects nodes to one another. Links can represent electrical or fiber optic cables.

A Link

These objects are found in the **object palette**, a dialog box that contains graphical representations of node and link models.

You can use any of three methods to create a network topology, or a combination of all three. One method is to import the topology (discussed in a later lesson).

Another is to place individual nodes from the object palette into the workspace. The third method is to use

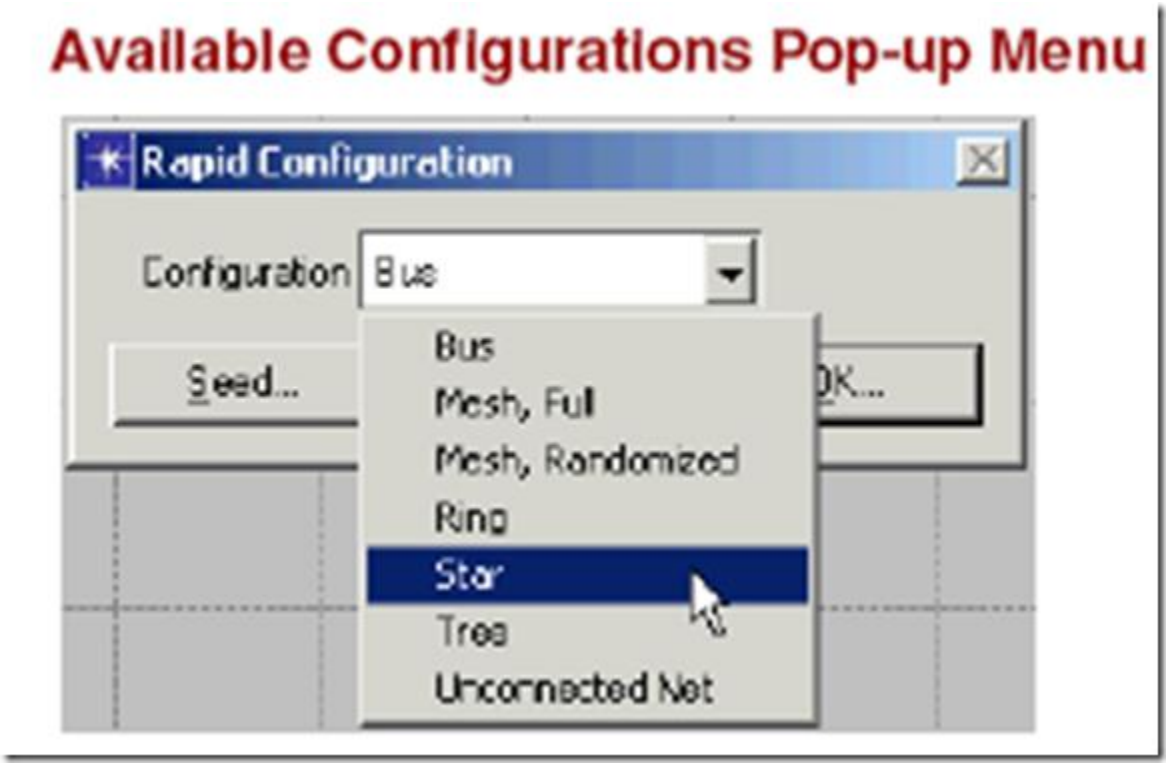
Rapid Configuration.

Rapid Configuration creates a network in one action after you select a network configuration, the types of nodes within the network, and the types of links that connect the nodes.

To create the first-floor network using Rapid Configuration:

1 Select **Topology** > **Rapid Configuration**.

Select **Star** from the drop-down menu of available configurations, then click **OK...**.



Specify the node models and link models in the network. Models follow this naming scheme:

<protocol1>_..._<protocol*n*>_<function>_<mod>

where:

• **<protocol>** specifies the specific protocol(s) supported by the model

• **<function>** is an abbreviation of the general function of the model

• **<mod>** indicates the level of derivation of the model

For example:

ethernet2_bridge_int

specifies the intermediate (**int**) derivation of a 2-port

Ethernet (**ethernet2**) bridge (**bridge**).

Vendor models have an additional prefix that specifies the vendor and the vendor product number for that particular network object.

For example, the 3Com switch used in this lesson is named:

3C_SSII_1100_3300_4s_ae52_e48_ge3

This node is a stack of two 3Com SuperStack II 1100 and two Superstack II 3300 chassis

(**3C_SSII_1100_3300**) with four slots (**4s**), 52 auto-sensing Ethernet ports (**ae52**), 48 Ethernet ports

(**e48**), and 3 Gigabit Ethernet ports (**ge3**).

To specify the nodes and links to use to build the network:

1 Set the **Center Node Model** to**3C_SSII_1100_3300_4s_ae52_e48_ge3**. This is a 3Com switch.

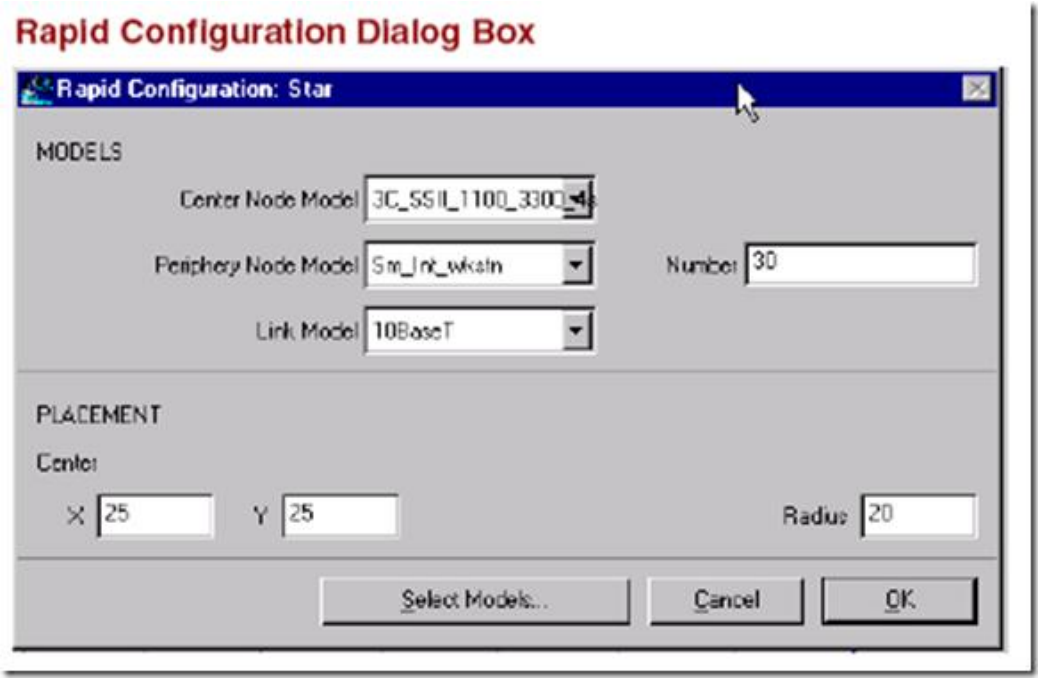
2 Set the **Periphery Node Model** to **Sm_Int_wkstn**, and change the **Number** of periphery nodes to **30**. This provides 30 Ethernet workstations as the peripheral nodes.

Set the **Link Model** to **10BaseT**.

Specify where the new network will be placed:

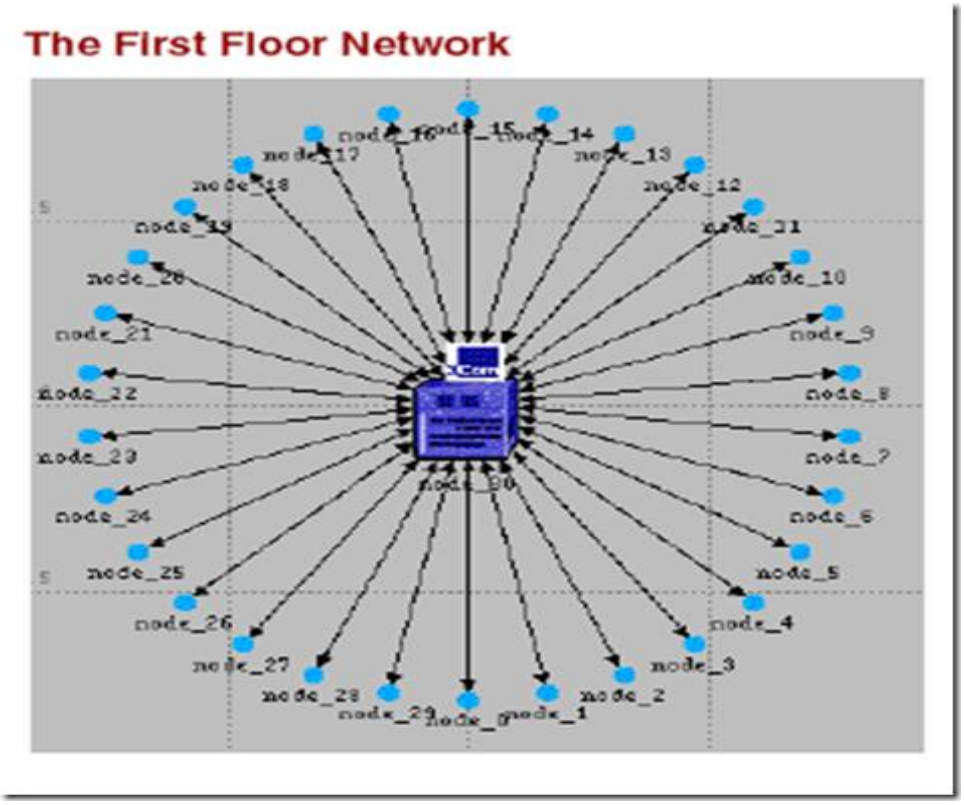
1 Set the **X center** and **Y center** to **25**.

2 Set the **Radius** to **20**.



3 Click **OK**.

➥ The network is drawn in the Project Editor:



Now that the general network topology has been built, you need to add a server. You will use the second method of creating network objects: dragging them from the object palette into the workspace.

1 If it is not already open, open the object palette by clicking on the

Object Palette action button.

2 Find the **Sm_Int_server** object in the palette and drag it into the workspace.

You will not find this exact server model on other object palettes because we created it with the correct configuration for this tutorial.

By default, you can create additional instances of the same object by left-clicking after the initial "drag-and-drop" from the palette.

3 Because you do not need additional copies of this model, right-click to turn off node creation.

You also need to connect the server to the star network.

1 Find the **10BaseT** link object in the palette and click on it.

2 Click on the server object, then click on the switch object in the center of the star.

➥ A link is drawn, connecting the two objects.

3 Right-click to turn off link creation.

Finally, you need to add configuration objects to specify the application traffic that will exist on the network. Configuring the application definition and profile definition objects can be complicated, so you do not have to do these tasks right now. For this tutorial, we included, on the object palette:

• an application definition object with the default configurations of the standard applications, and

• a profile definition object with a profile that models light database access

You need only drag the objects into your network. Doing so means that the traffic caused by workstations accessing a database at a low rate will be modeled.

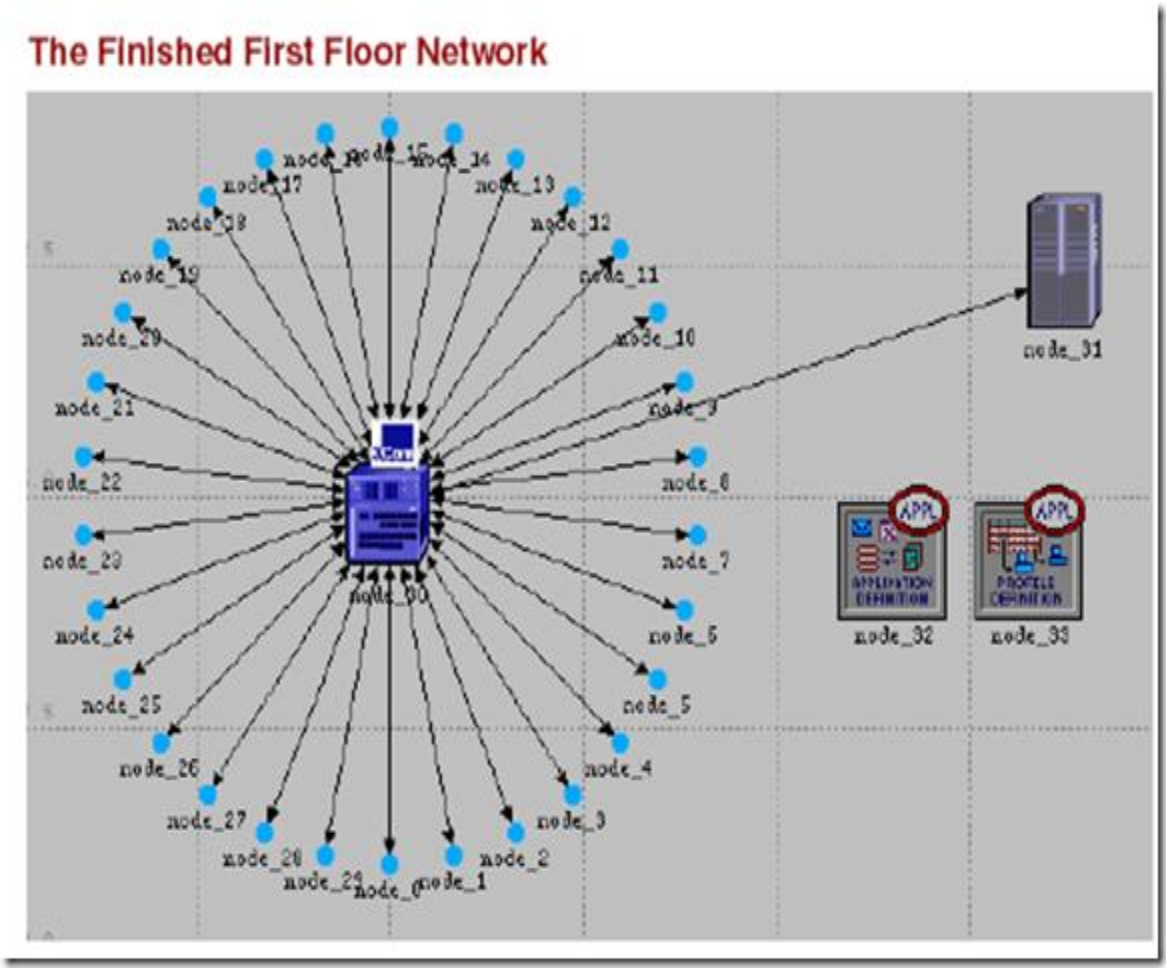
1 Find the **Sm_Application_Config** object in the palette and drag it into the workspace

2 Right-click to turn off object creation.

3 Find the **Sm_Profile_Config** object in the palette, drag it into the workspace, and right-click.

4 Close the object palette.

The network is now built and should look similar to the following figure.



You are now ready to begin collecting statistics.

Collecting Statistics

You can collect statistics from individual nodes in your network (**object statistics**) or from the entire network (**global statistics**).

Now that you have created the network, you should decide which statistics you need to collect to answer the questions presented earlier in this lesson:

• Will the server be able to handle the additional load of the second network?

• Will the total delay across the network be acceptable once the second network is installed?

To answer these questions, you need a snapshot of current performance for comparison. To get this baseline, you will collect one object statistic, **Server Load**, and one global statistic, **Ethernet Delay**.

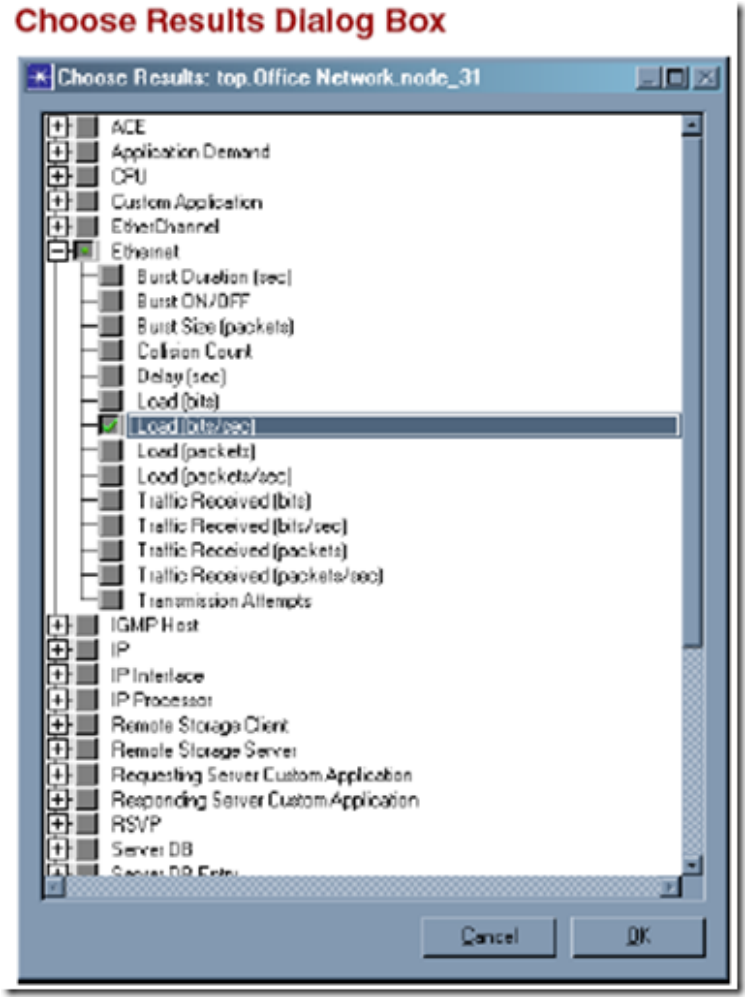
Server load is a key statistic that reflects the performance of the entire network. To collect statistics related to the server's load, do the following steps:

1 Right-click on the server node (**node_31**) and select **Choose Individual Statistics** from the server's Object pop-up menu.

➥ The Choose Results dialog box for node_31 appears.

The Choose Results dialog box hierarchically organizes the statistics you may collect. To collect the Ethernet load on the server:

2 Click the plus sign next to **Ethernet** in the Choose Results dialog box to expand the Ethernet statistic hierarchy.



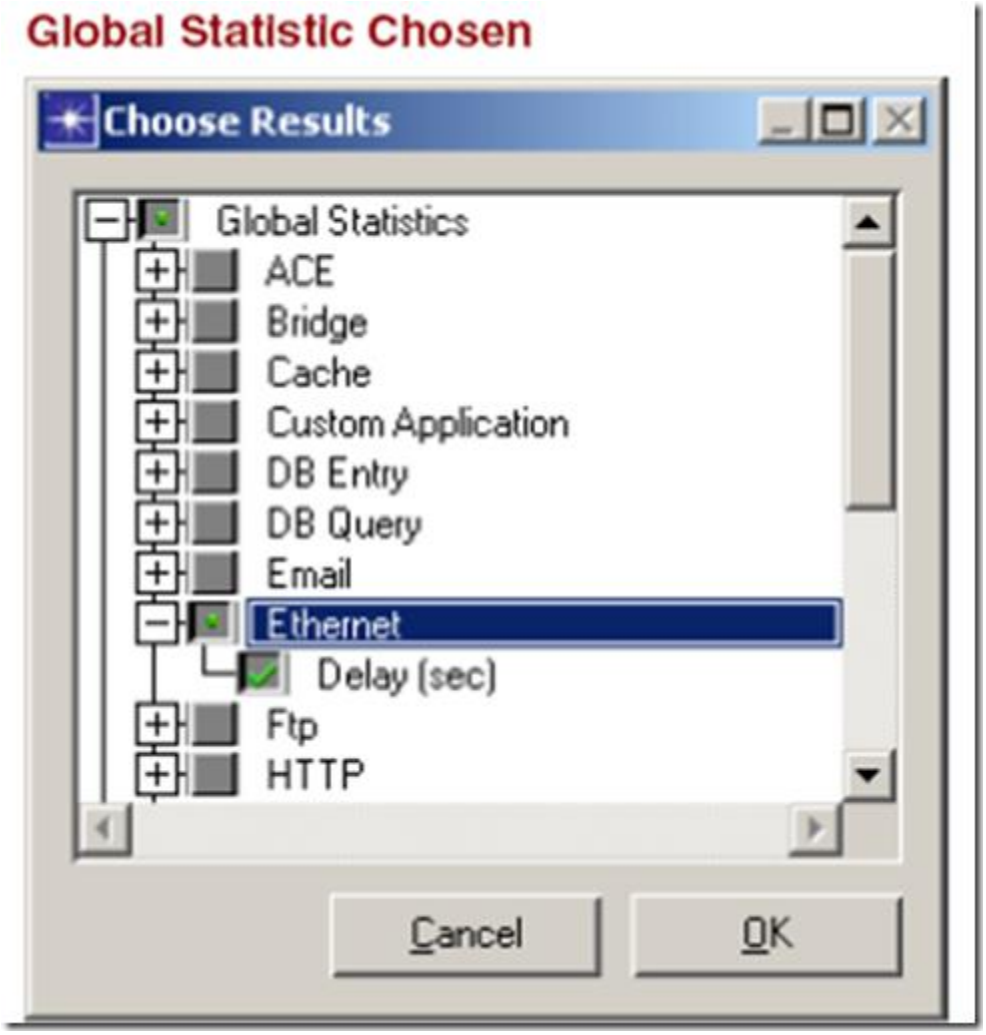
3 Click the checkbox next to **Load (bits/sec)** to enable collection for that statistic.

4 Click **OK** to close the dialog box.

Global statistics can be used to gather information about the network as a whole. For example, you can find out the delay for the entire network by collecting the global **Delay** statistic:

1 Right-click in the workspace (but not on an object) and select **Choose Individual Statistics** from the

Workspace pop-up menu.



2 Expand the Global Statistics hierarchy.

3 Expand the Ethernet hierarchy.

4 Click the checkbox next to **Delay (sec)** to enable data collection.

5 Click **OK** to close the Choose Results dialog box.

It is good to get into the habit of saving your project every so often. To save the project:

1 Choose **File > Save**, then click **OK** (the project already has a name, so you don't need to rename it).

Now that you have specified which statistics to collect and saved the project, you are almost ready to run your simulation.

First, though, verify that your **repositories** preference is set. Repositories contain user-defined components such as process models and pipeline stages that are saved so that simulations will take less time to begin execution.

1 Choose **Edit > Preferences.**

2 Type **repositories** in the **Find** field and click on the **Find**button.

3 If the value for **repositories** is not **stdmod**, click on the field and enter **stdmod** in the dialog box.

4 Click **OK** to close the **repositories** and

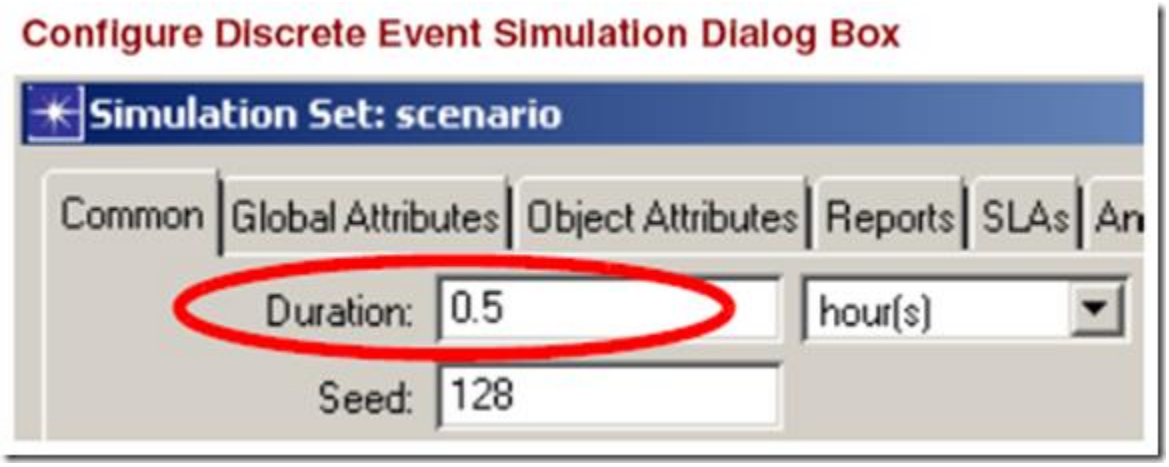
Preferences dialog boxes

To run a simulation:

1 Select **Simulation > Configure Discrete EventSimulation….**

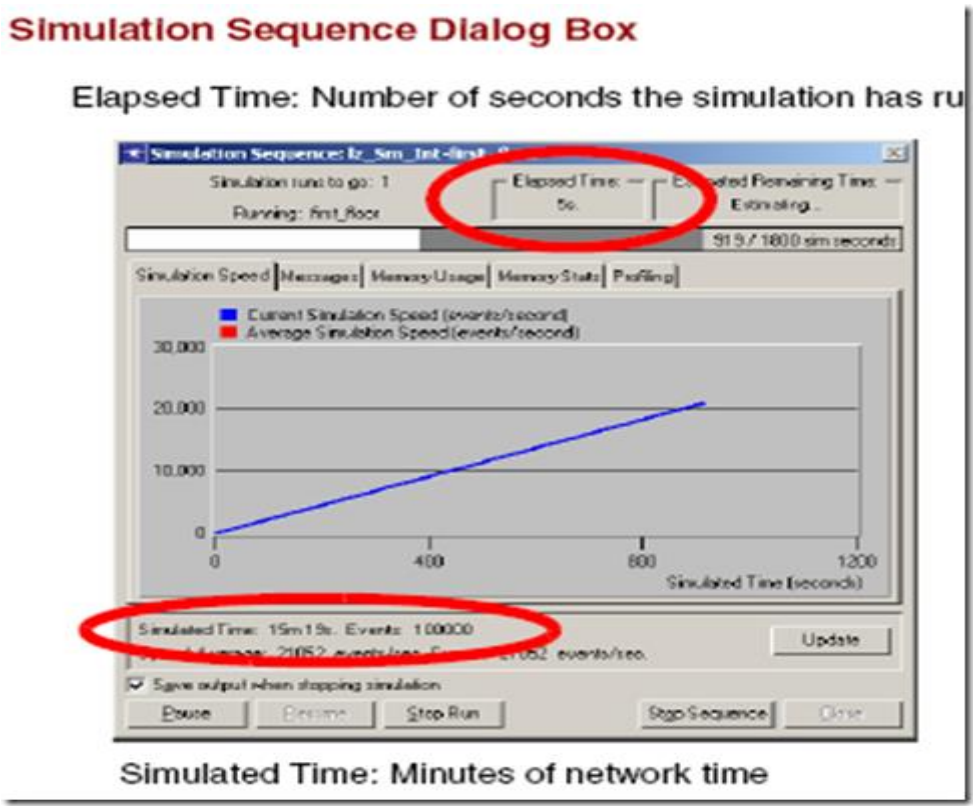
You can also open the Configure Discrete Event Simulation dialog box by clicking on the **configure/run simulation** action button.

2 Type **0.5** in the **Duration:** field to simulate one-half hour of network activity.



Click the **Run** button to begin the simulation.

While the simulation runs, a dialog box appears showing the simulation's progress.



The dialog box above shows that, in 5 seconds of elapsed (actual) time, IT Guru has simulated 15 minutes and 19 seconds of network time. The entire simulation should take less than one minute to complete—the elapsed time varies according to the speed of your computer.

4 When the simulation finishes, the contents of the

Messages tab appears. Click the **Close** button in the Simulation Sequence dialog box.

5 If your simulation does not complete, if no results were collected, or if the results vary significantly from those shown, you will have to troubleshoot your simulation.

Viewing Results

You can view results graphically in the Project Editor by selecting **View Results** from the Workspace pop-up menu.

After your simulation has executed, you will want to see the information collected for each statistic. There are several ways to view results; in this lesson you will use the View Results option in the Workspace pop-up menu.

You will learn different ways to view results in later lessons.

To view the server Ethernet load for the simulation:

1 Right-click on the server node (**node_31**) choose

View Results from the server's Object pop-up menu.

➥ The node's View Results dialog box opens.

2 Expand the **Office network.node_31 > Ethernet** hierarchy.

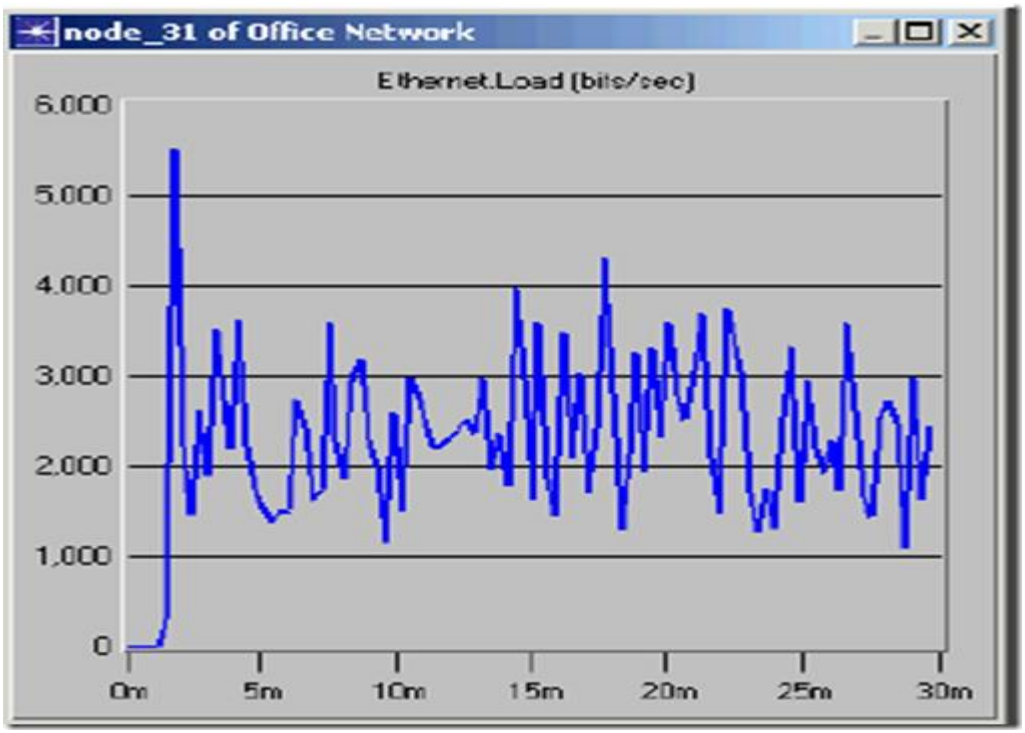
Click on the checkbox next to **Load (bits/sec)** to indicate that you want to view that result.

4 Click the **Show** button in the View Results dialog box.

➥ The graph of the server load appears in the Project Editor, as shown in the following figure.

The graph of the server load should resemble the following graph. Your results may differ slightly due to differences in node placement and link length, but the general trends should be consistent.

Server Load Graph



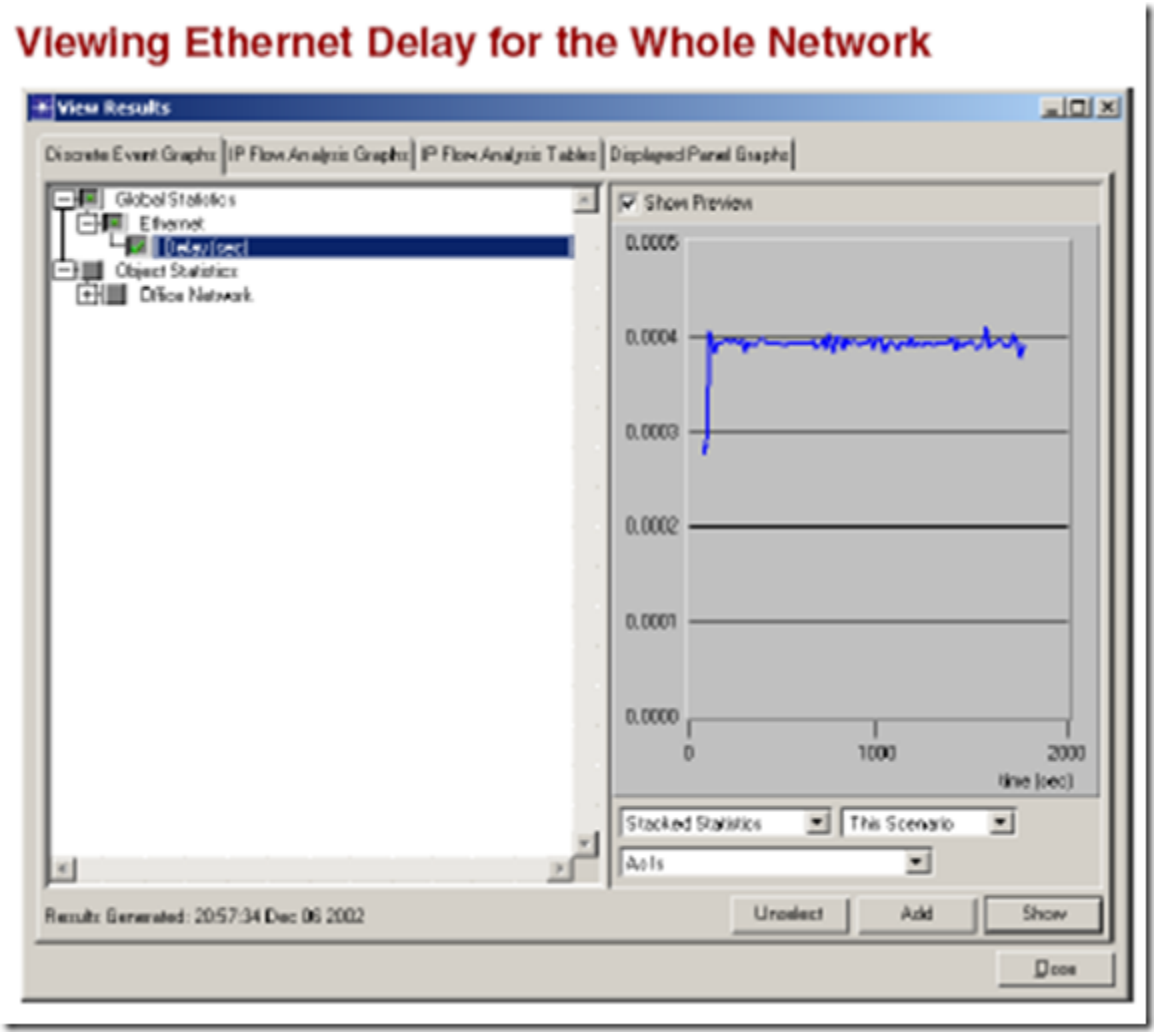
Note that at its peak, the load on the server is well below 6,000 bits/second. You will need this baseline for comparison after you add the second network.

When you finish viewing the server load graph, close this dialog box and the View Results dialog box. (If the system prompts you, choose to delete the graph panel.)

You also should look at the Global Ethernet Delay on the network. To view this statistic:

1 Right-click in the workspace, then select View Results from the pop-up menu.

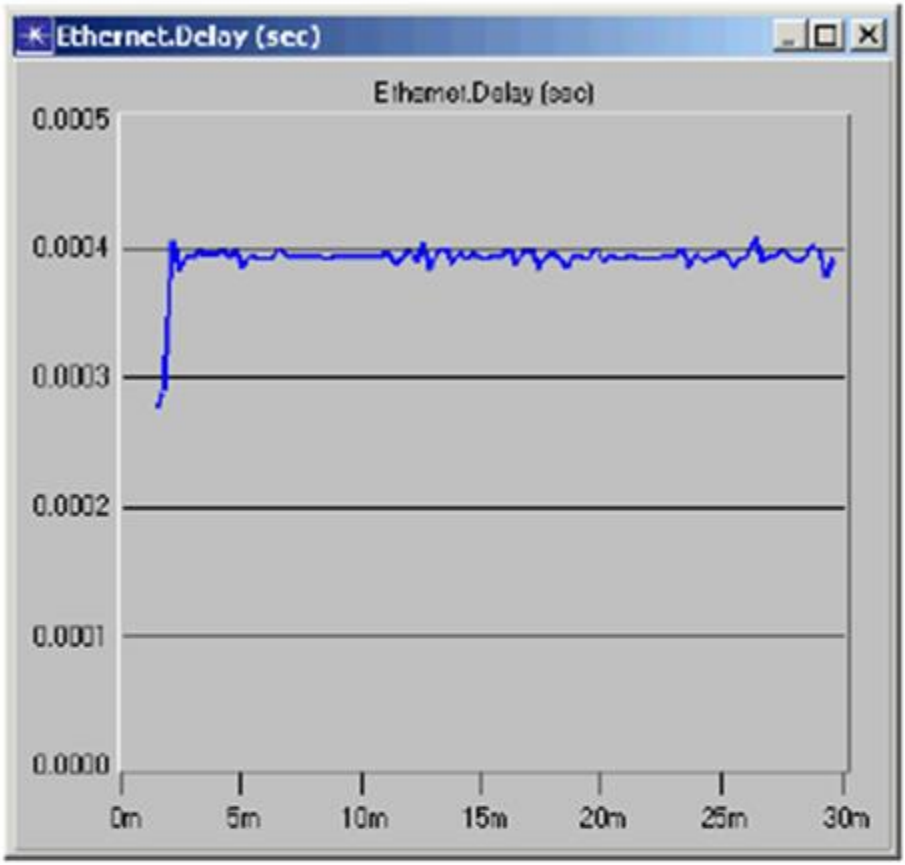
2 Check the box next to Global Statistics > Ethernet > Delay, then click the Show button to view the Ethernet delay for the whole network.



➥ The Ethernet delay graph appears in the Project Editor.The graph should resemble the following figure.

Ethernet Delay Graph

Seconds



Simulation time, in minutes

Note that after the network reaches steady state the maximum delay is around 0.4 milliseconds.

When you are finished viewing the graph, close it and the View Results dialog box.

Lab 21 – A Direct Link Network with Media Access Control

Objectives

The objectives of this lab is to apply a direct link network with MAC

Lab Tasks

Task: Apply Direct Link Network with MAC

The Ethernet is a working example of the more general Carrier Sense, Multiple Access with Collision Detect (CSMA/CD) local area network technology. The Ethernet is a multiple-access network, meaning that a set of nodes sends and receives frames over a shared link. The "carrier sense" in CSMA/CD means that all the nodes can distinguish between an idle and a busy link. The "collision detect" means that a node listens as it transmits and can therefore detect when a frame it is transmitting has interfered (collided) with a frame transmitted by another node. The Ethernet is said to be a 1-persistent protocol because an adaptor with a frame to send transmits with probability 1 whenever a busy line goes idle.

In this lab you will set up an Ethernet with 14 nodes connected via a coaxial link in a bus topology. The coaxial link is operating at a data rate of 10 Mbps. You will study how the throughput of the network is affected by the network load as well as the size of the packets.

Procedure

Create a New Project

To create a new project for the Ethernet network:

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the **File**menu.

2. Select **Project** ⇒ Click **OK** ⇒ Name the project <**your initials**>**_Ethernet**, and the scenario **Coax** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that**Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Choose **Office** from the *Network Scale* list ⇒ Click **Next** ⇒ Assign **200** to **X Span** and keep **Y Span** as **100** ⇒ Click **Next** twice ⇒ Click **OK**.

4. Close the *Object Palette* dialog box.

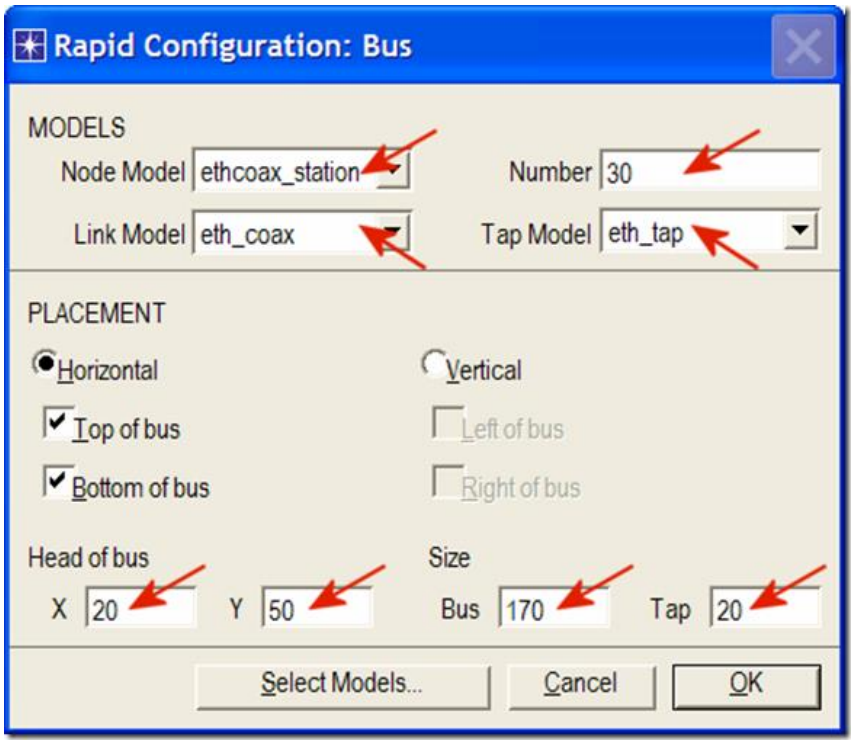
Create the Network

To create our coaxial Ethernet network:

1. To create the network configuration, select **Topology** ⇒ **Rapid Configuration**. From the drop-down menu choose **Bus** and click **OK**.

2. Click the **Select Models** button in the *Rapid Configuration* dialog box. From the *Model List* drop-down menu choose **ethcoax** and click **OK**.

3. In the *Rapid Configuration* dialog box, set the following eight values and click **OK**.



4. To configure the coaxial bus, right-click on the horizontal link ⇒ Select**Advanced Edit Attributes** from the menu:

a. Click on the value of the **model** attribute ⇒ Select **Edit** from the drop-down menu ⇒ Choose the **eth_coax_adv** model.

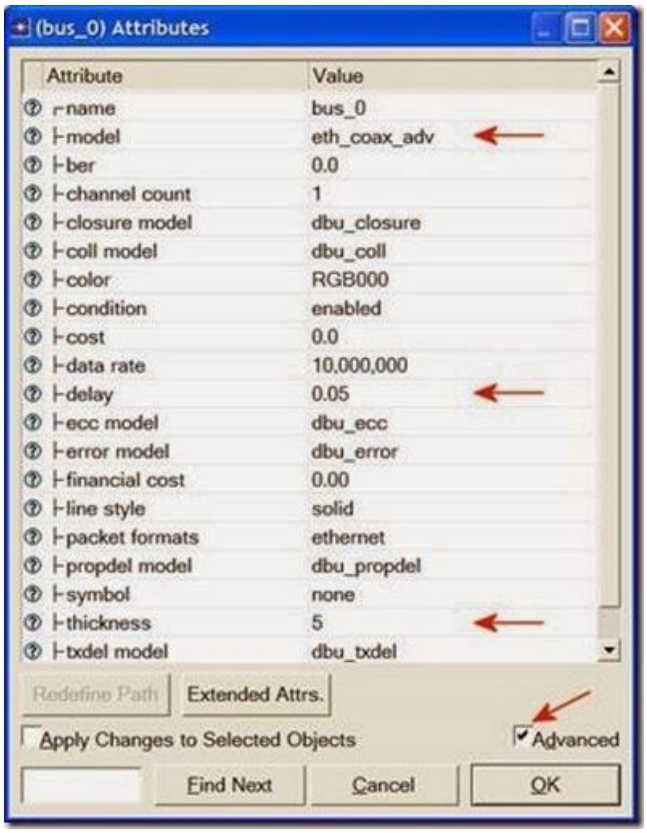
b. Assign the value **0.05** to the **delay** attribute (propagation delay in sec/m).

c. Assign **5** to the **thickness** attribute.

d. Click **OK**.

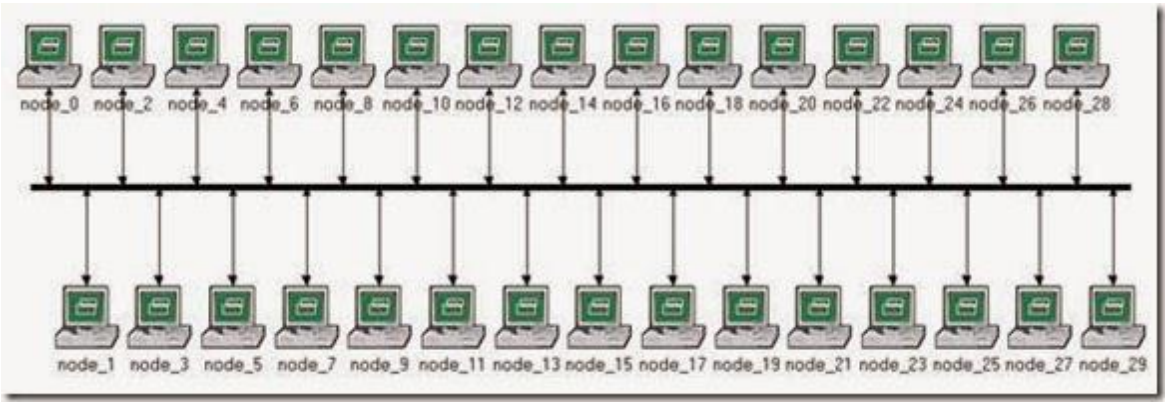
Note: The eth_tap is an Ethernet bus tap that connects a node with the bus. The eth_coax is an Ethernet bus that can connect nodes with bus receivers and transmitters via taps.
A higher delay is used here as an alternative to generating higher traffic which would require much longer simulation time.

Thickness specifies the thickness of the line used to "draw" the bus link.



5. Now you have created the network. It should look like the illustration below.

6. Make sure to save your project.



Configure the Network Nodes

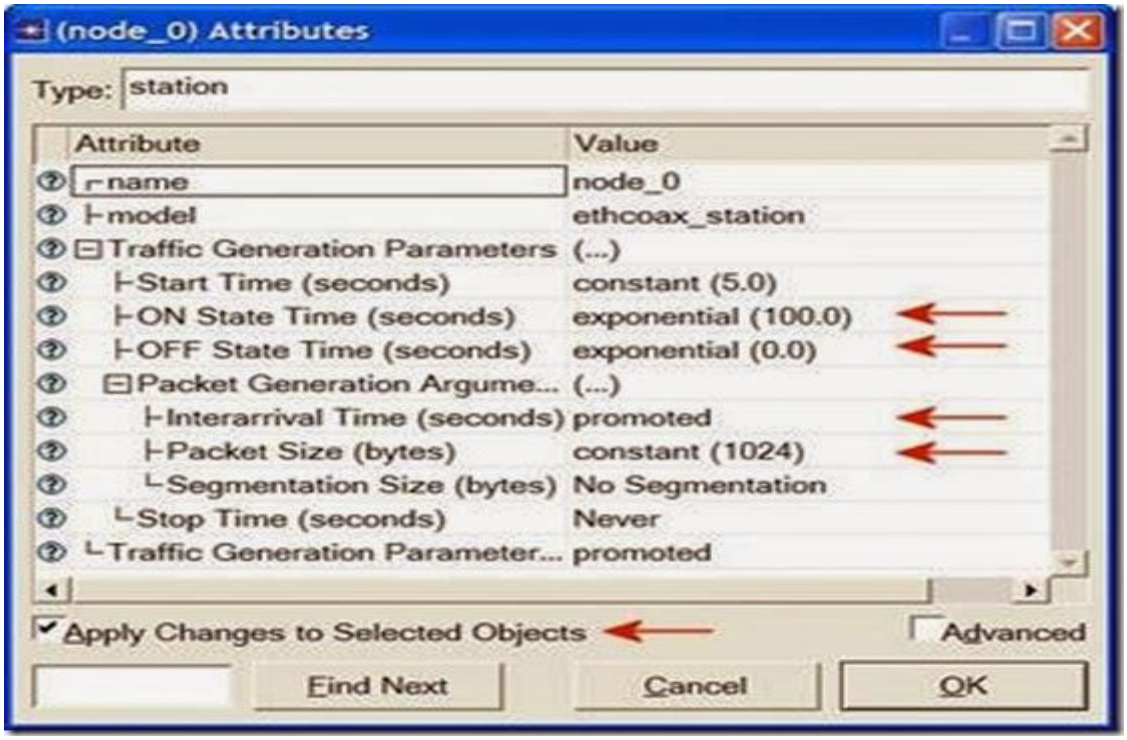
The argument of the **exponential** distribution is the mean of the interval between successive events. In the exponential distribution the probability of occurrence of the next event by a given time is not at all dependent upon the time of occurrence of the last event or the elapsed time since that event.

The **interarrival time** is the time between successive packet generations in the "ON"

state.

To configure the traffic generated by the nodes:

- Right-click on any of the 30 nodes ⇒ Select Similar Nodes. Now all nodes in the network are selected.
- Right-click on any of the 30 nodes ⇒ Edit Attributes.
- Check the Apply Changes to Selected Objects check box. This is important to avoid reconfiguring each node individually.
- Expand the Traffic Generation Parameters hierarchy:
- Change the value of the ON State Time to exponential(100) ⇒ Change the value of the OFF State Time to exponential(0). (Note: Packets are generated only in the "ON" state.)
- Expand the Packet Generation Arguments hierarchy:
- Change the value of the Packet Size attribute toconstant(1024).
- Right-click on the Interarrival Time attribute and choosePromote Attribute to Higher Level. This allows us to assign multiple values to the Interarrival Time attribute and hence to test the network performance under different loads.



6. Click **OK** to return back to the *Project Editor*.

7. Make sure to save your project.

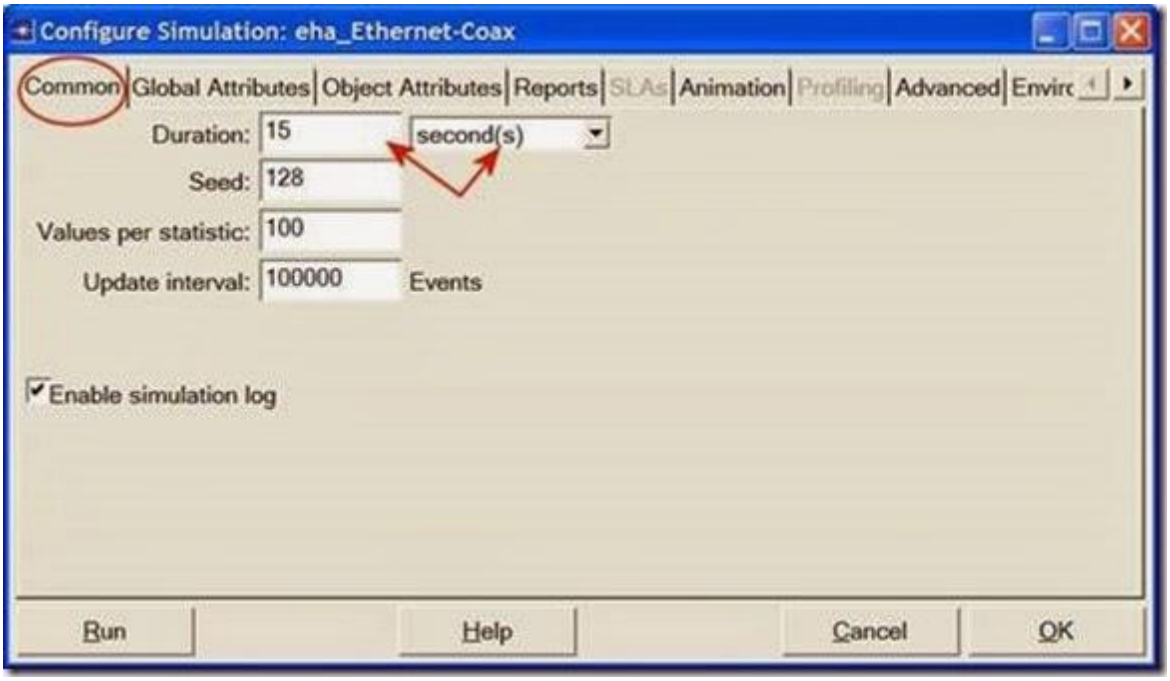
Configure the Simulation

To examine the network performance under different loads, you need to run the simulation several times by changing the load into the network. There is an easy way to do that. Recall that we promoted the **Interarrival Time**attribute for package generation. Here we will assign different values to that attribute:

1. Click on the **Configure/Run Simulation** button:

2. Make sure that the **Common** tab is chosen ⇒ Assign **15** seconds to the

Duration.



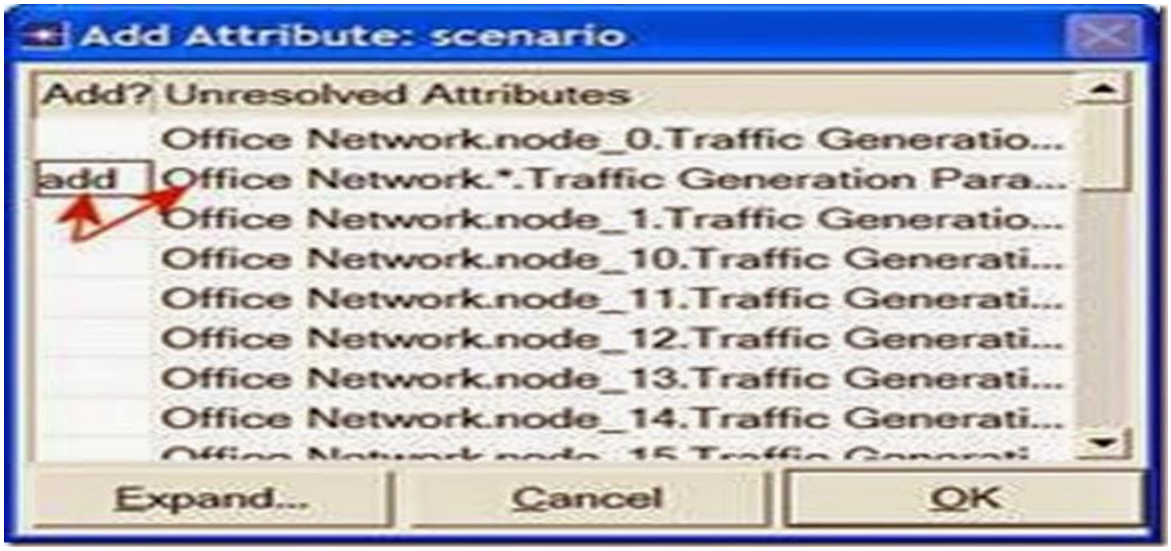
3. Click on the **Object Attributes** tab.

4. Click on the **Add** button. The *Add Attribute* dialog box should appear filled with the promoted attributes of all nodes in the network (if you do not see the attributes in the list, close the whole project and reopen it). You need to add the **Interarrival Time** attribute for all nodes. To do that:

a. Click on the first attribute in the list (**Office Network.node_0.TrafficGeneration** ….) ⇒ Click the **Wildcard** button ⇒ Click on **node_0** andchoose the asterisk (*****) from the drop-down menu ⇒ Click **OK**.

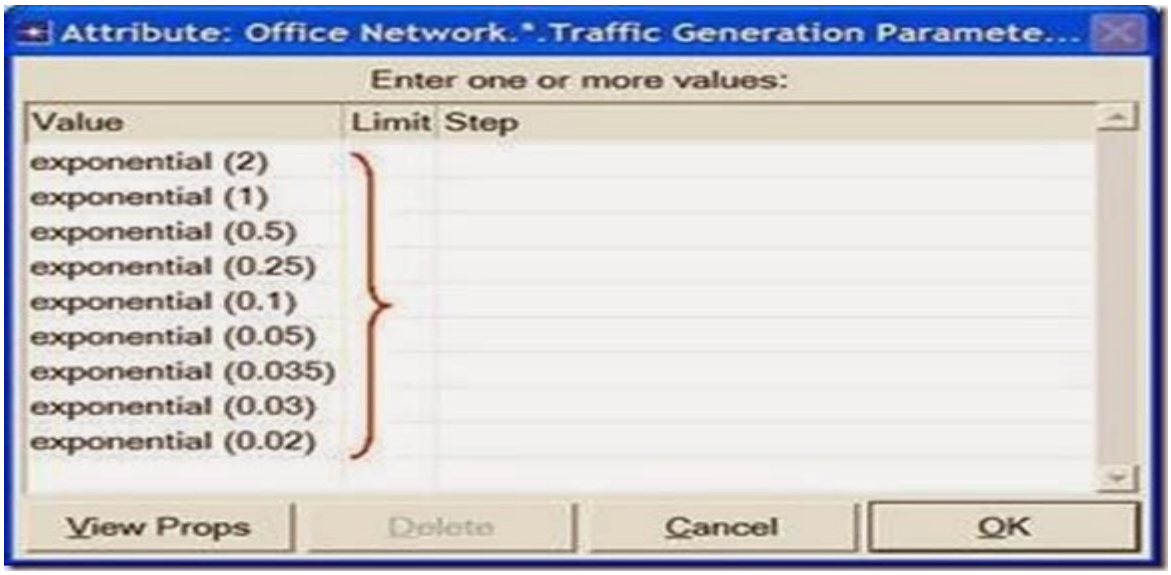
b. A new attribute is now generated containing the asterisk (the second one in the list), and you need to add it by clicking on the corresponding cell under the **Add?** column.

c. The *Add Attribute* dialog box should look like the following. Click **OK**.

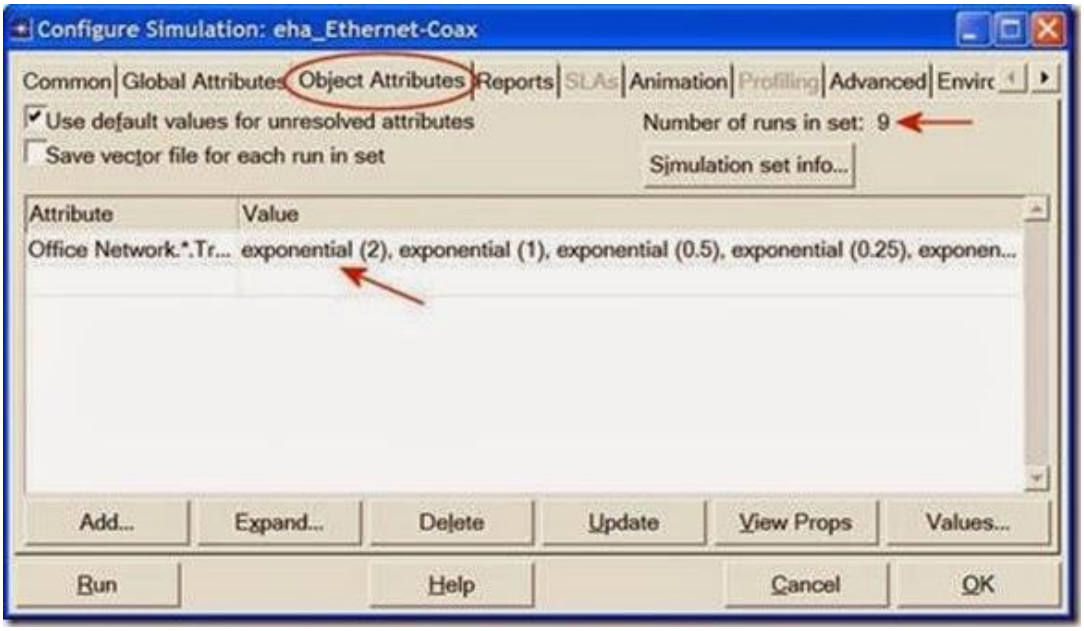


5. Now you should see the **Office Network.*.Traffic Generation Parameter …** in the list of simulation object attributes. Click on that attribute to select it ⇒ Click the **Values** button of the dialog box.

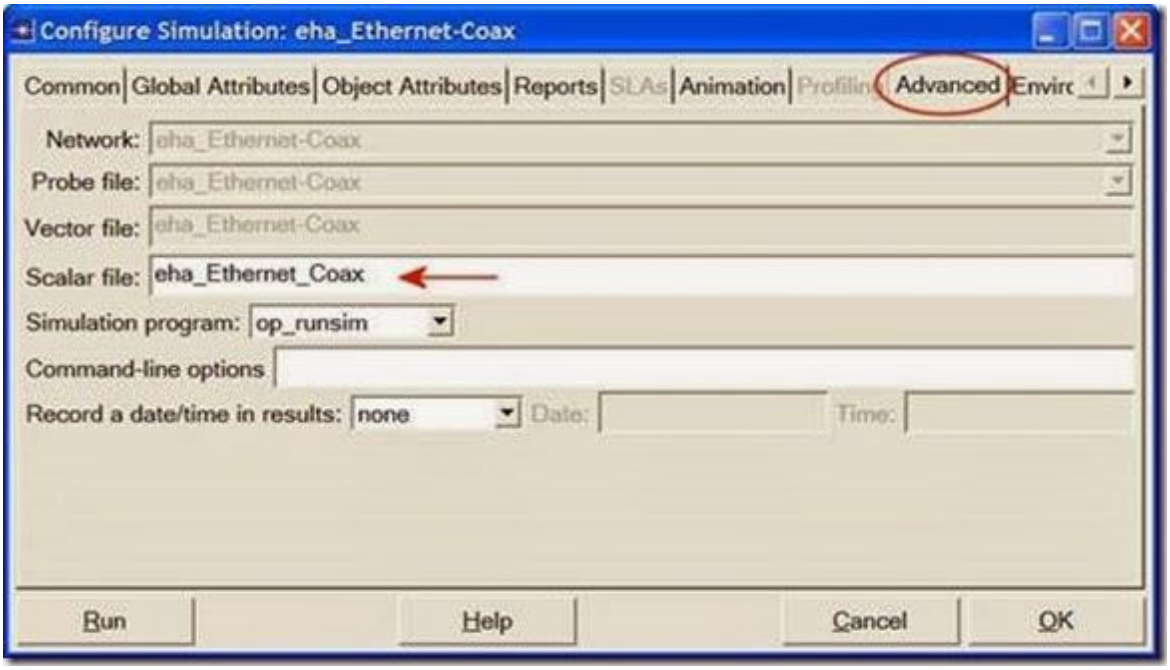
6. Add the following nine values. (*Note:* To add the first value, double-click on the first cell in the **Value** column ⇒ Type "exponential (2)" into the textbox and hit enter. Repeat this for all nine values.)



7. Click **OK**. Now look at the upper-right corner of the *Simulation Configuration* dialog box and make sure that the *Number of runs in set* is **9**.



8. For each simulation of the nine runs, we need the simulator to save a "scalar" value that represents the "average" load in the network and to save another scalar value that represents the average throughput of the network. To save these scalars we need to configure the simulator to save them in a file. Click on the **Advanced** tab in the *Configure Simulation* dialog box.
9. Assign <**your initials**>**_Ethernet_Coax** to the *Scalar file* text field.



10. Click **OK** and then save your project.

Choose the Statistics

a. To choose the statistics to be collected during the simulation:
   - Right-click anywhere in the project workspace (but not on one of the nodes or links) and select Choose Individual Statistics from the pop-up menu ⇒ Expand the Global Statistics hierarchy.
   - Expand the Traffic Sink hierarchy ⇒ Click the check box next to Traffic Received (packets/sec) (make sure you select the statistic with units of packets/sec),
   - Expand the Traffic Source hierarchy ⇒ Click the check box next to
   - Traffic Sent (packets/sec).
   - Click OK.
   - Now to collect the average of the above statistics as a scalar value by the end of each simulation run:
   - Select Choose Statistics (Advanced) from the Simulationmenu.

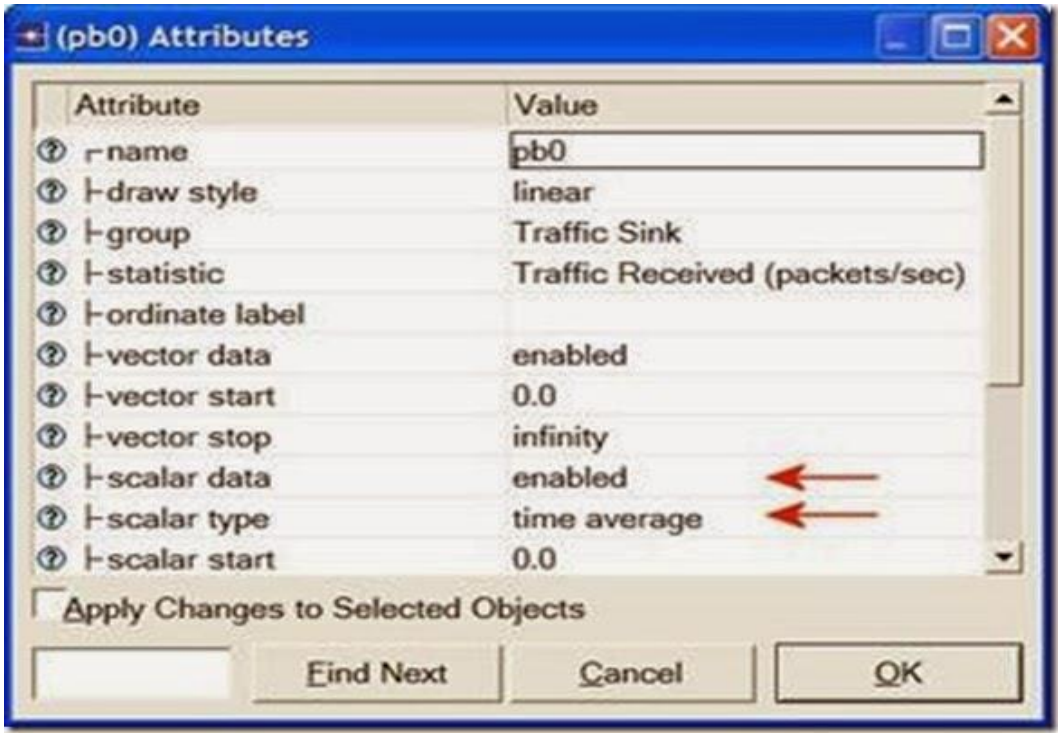
b. The Traffic Sent and Traffic Received probes should appear under theGlobal Statistic Probes.

c. Right-click on Traffic Received probe ⇒ Edit Attributes. Set the scalardata attribute to enabled ⇒ Set the scalar type attribute to time average ⇒Compare to the following figure and click OK.

d. Repeat the previous step with the Traffic Sent probe.

e. Select save from the File menu in the Probe Model window and then close that window.

f. Now you are back to the Project Editor. Make sure to save your project.



Run the Simulation

To run the simulation:

1. Click on the **Configure/Run Simulation** button: ⇒ Make sure that **15second(s)** (not hours) is assigned to the **Duration** ⇒ Click **Run**. Depending on the speed of your processor, this may take several minutes to complete.

2. Now the simulator is completing nine runs, one for each traffic generation interarrival time (representing the load into the network). Notice that each successive run takes longer to complete because the traffic intensity is increasing.

3. After the nine simulation runs complete, click **Close**.

4. Save your project.

View the Results: To view and analyze the results:

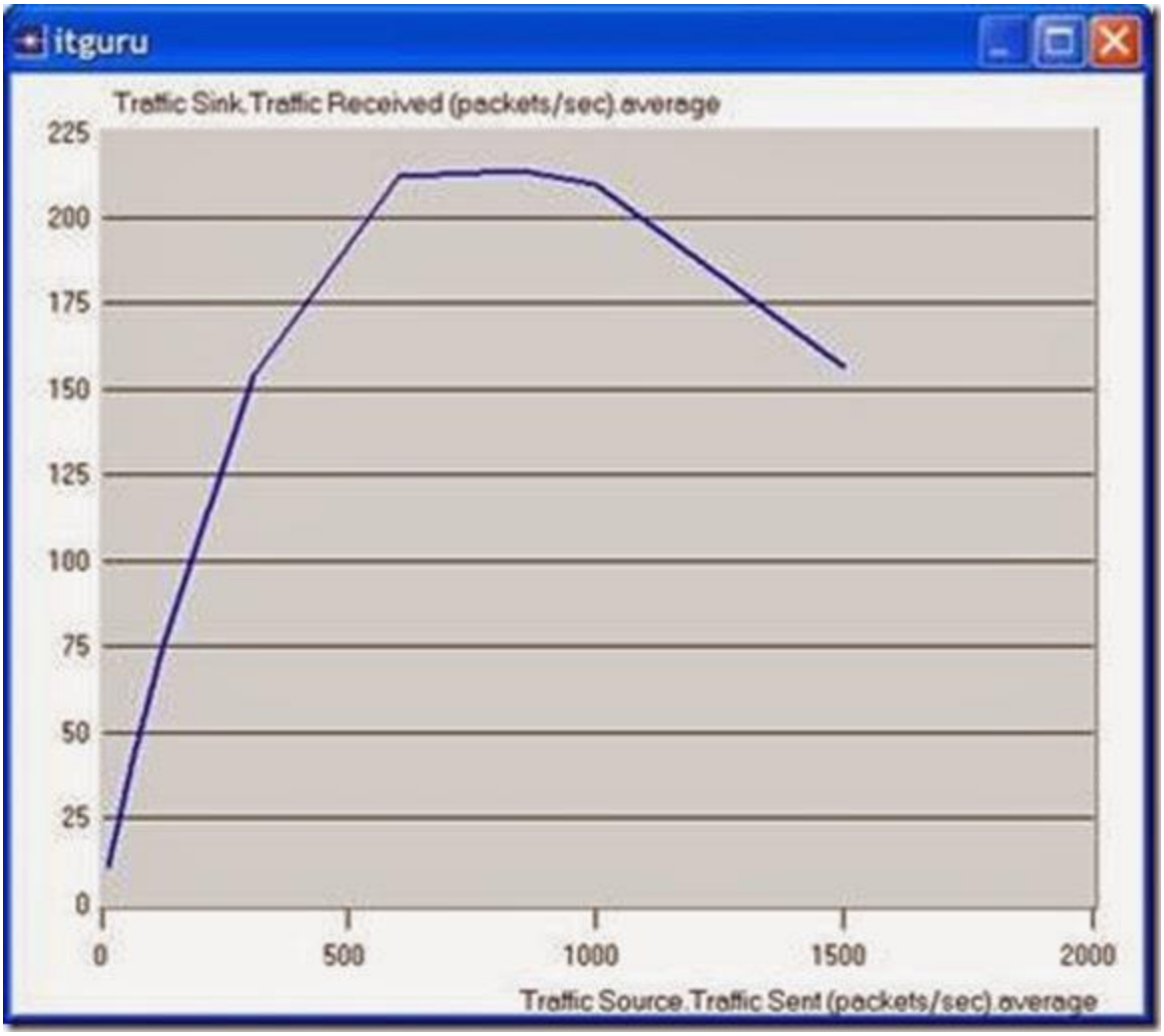
1. Select View Results (Advanced) from the Results menu. Now theAnalysis Configuration tool is open.

2. Recall that we saved the average results in a scalar file. To load this file, select

Load Output Scalar File from the **File** menu ⇒ Select <**your initials**>**_Ethernet-Coax** from the pop-up menu.

3. Select **Create Scalar Panel** from the **Panels** menu ⇒ Assign **TrafficSource.Traffic Sent (packets/sec).average** to **Horizontal** ⇒ Assign**Traffic Sink.Traffic Received (packets/sec).average** to **Vertical** ⇒ Click**OK**.

4. The resulting graph should resemble the one below:



Note: When you rerun the simulation, OPNET IT Guru will "append" the new results to the results already in the scalar file. To avoid that, delete the scalar file *before* you start a new run. (*Note:* Deleting the scalar file *after* a run will result in losing the collected results from that run.)

• Go to the **File** menu ⇒ Select **Model Files** ⇒ **Delete Model Files** ⇒

Select **(.os): Output Scalars** ⇒ Select the scalar file to be deleted; in this lab it is <**your initials**>**_Ethernet_Coax_Scalar** ⇒ Confirm the deletion by clicking **OK** ⇒ Click **Close.**

Lab 22 – A shared Media Network with Media Access Control

Objectives

The objectives of this lab is to apply a shared Media Network with MAC

Lab Tasks

Task: Apply a shared Media Network with MAC.

A token ring network consists of a set of nodes connected in a ring. The ring is a single shared medium. The token ring technology involves a distributed algorithm that controls when each node is allowed to transmit. All nodes see all frames, and the node identified as the destination in the frame header saves a copy of the frame as it flows past. With a ring topology, any link or node failure would render the whole network useless. This problem can be solved by using a star topology where nodes are connected to a token ring hub. The hub acts as a relay, known as a multi station access unit (MSAU). MSAUs are almost always used because of the need for robustness and ease of node addition and removal.

The "token," which is just a special sequence of bits, circulates around the ring; each node receives and then forwards the token. When a node that has a frame to transmit sees the token, it takes the token off the ring and instead inserts its frame into the ring. When the frame makes its way back around to the sender, this node strips its frame off the ring and reinserts the token. The*token holding time* (THT) is the time a given node is allowed to hold the token. From its definition, the THT has an effect on the utilization and fairness of the network, where utilization is the measure of the bandwidth used versus that available on the given ring.

In this lab, you will set up a token ring network with 14 nodes connected in a star topology. The links you will use operate at a data rate of 4 Mbps. You will study how the utilization and delay of the network are affected by the network load as well as the THT.

Procedure

Create a New Project

To create a new project for the token ring network:

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the **File**menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials>_Token**, and the scenario **Balanced** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that**Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Choose **Office** for the Network scale ⇒ Click **Next** three times ⇒ Click **OK**.

4. Close the *Object Palette* and then save your project.

Create the Network

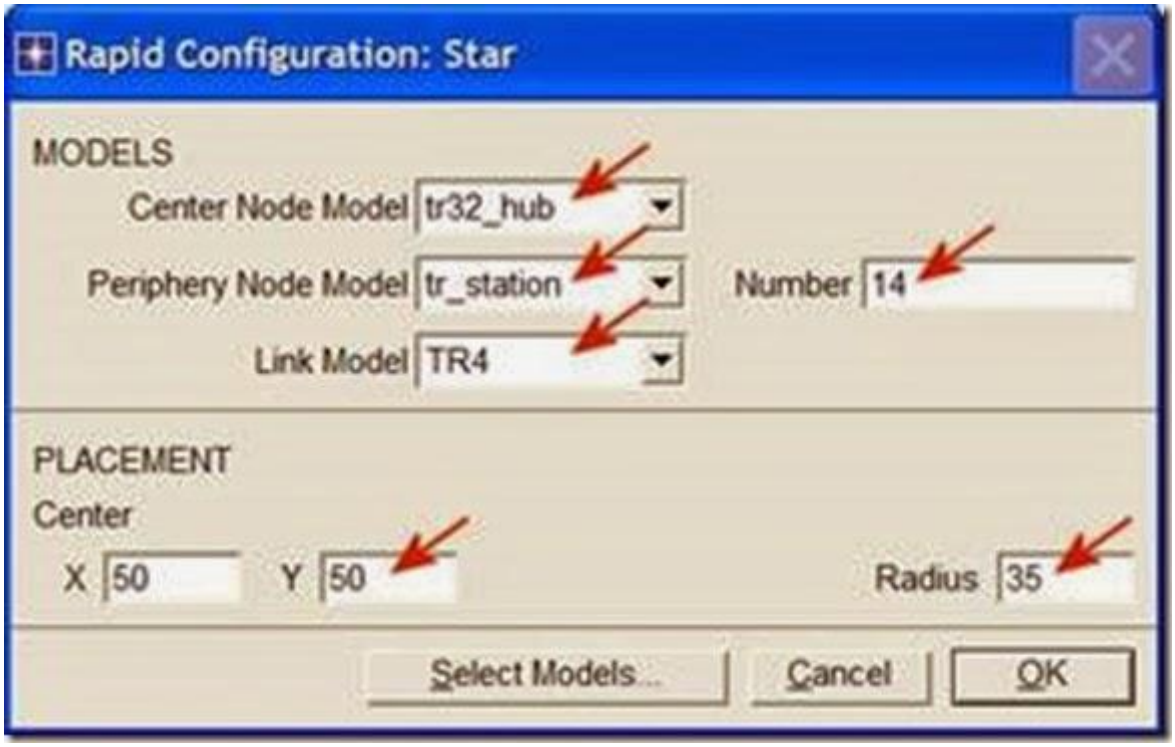
To create our token ring network:

1. Select **Topology ⇒ Rapid Configuration**. From the drop-down menu choose **Star** and click **OK**.

2. Click the **Select Models** button in the *Rapid Configuration* dialog box. From the *Model List* drop-down menu choose **token_ring** and click **OK**.

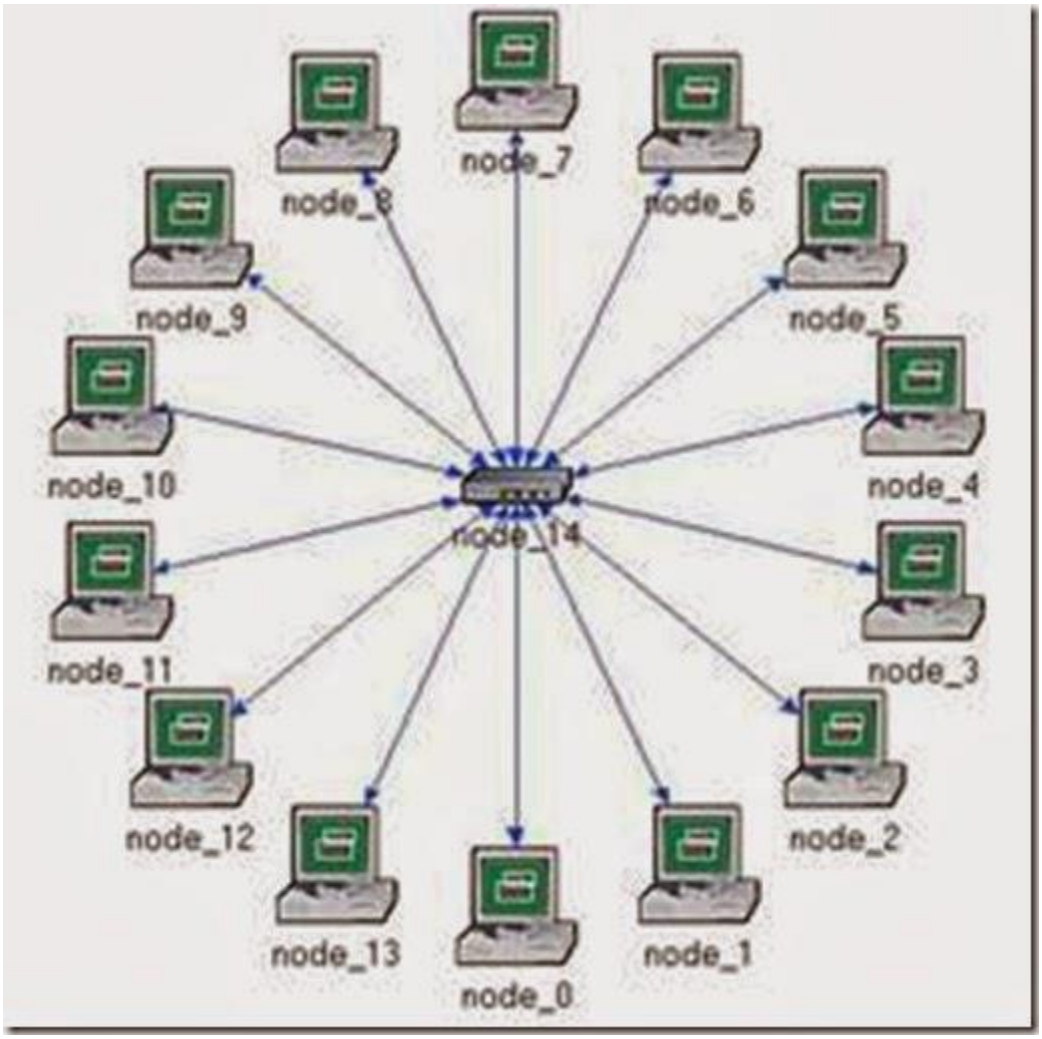
3. In the *Rapid Configuration* dialog box, set the following six values and click **OK**.

The **tr32_hub** node model is a token ring hub supporting up to 32 connections at 4 or 16 Mbps. The hub forwards an arriving packet to the next output port. There is no queuing of packets in the hub itself as the processing time is considered to be zero.



The **TR4 link** connects two token ring devices to form a ring at 4 Mbps.

4. You have now created the network, and it should look like the following:



5. Make sure to save your project.

Configure the Network Nodes

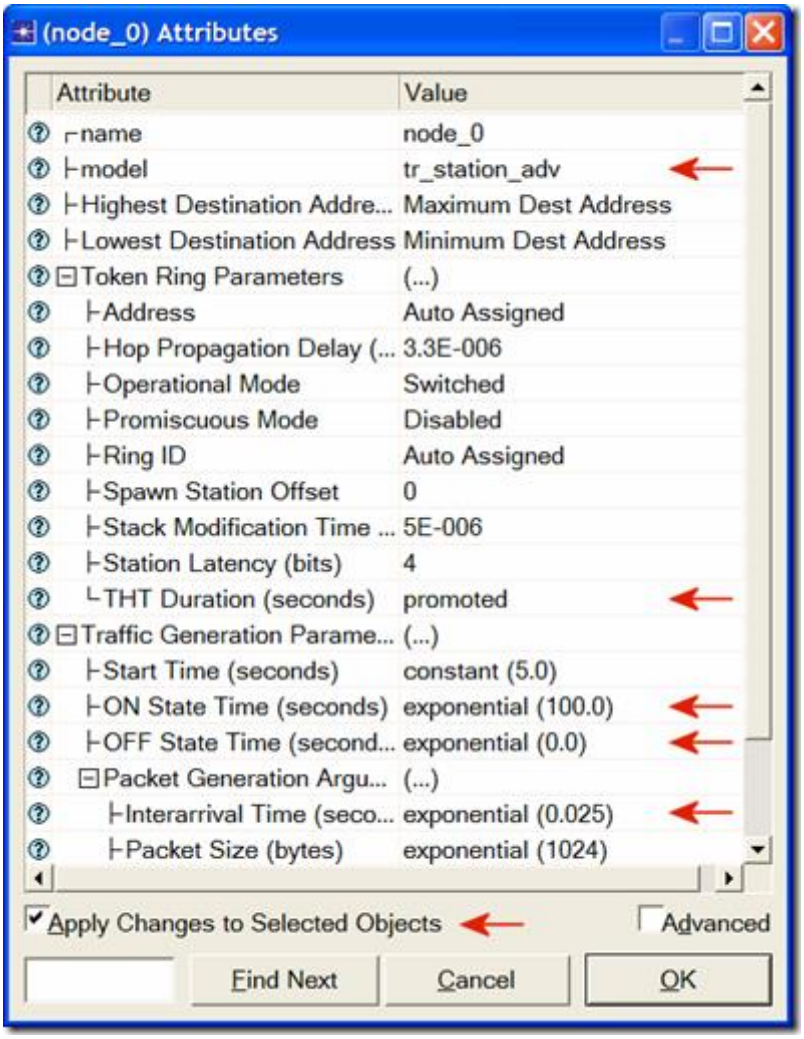
Here you will configure the THT of the nodes as well as the traffic generated by them. To configure the THT of the nodes, you need to use the**tr_station_adv** model for the nodes instead of the current one, **tr_station**.

1. Right-click on any of the 14 nodes ⇒ **Select Similar Nodes**. Now all nodes in the network are selected.

2. Right-click on any of the 14 nodes ⇒ **Edit Attributes.**

a. Check the **Apply Changes to Selected Objects** check box. This is important to avoid reconfiguring each node individually.

The following figure shows the attributes we will change in steps 3 to 6:



3. Click on the model value: **tr_station** and select **Edit** from the drop-down menu. Now select **tr_station_adv** from the extended drop-down menu.

4. To test the network under different THT values, you need to "promote" the THT parameter. This allows us to assign multiple values to the THT attribute.

a. Expand the **Token Ring Parameters** hierarchy.

b. Right-click on the **THT Duration** attribute ⇒ Choose **Promote Attributeto Higher Level**.

5. Expand the **Traffic Generation Parameters** hierarchy ⇒ Assign**exponential(100)** to the **ON State Time** attribute ⇒ Assign **exponential(0)**to the

OFF State Time attribute. (*Note:* Packets are generated only in the "ON" state.)

6. Expand the **Packet Generation Arguments** hierarchy ⇒ Assign**exponential(0.025)** to the **Interarrival Time** attribute**.**

7. Click **OK** to return back to the *Project Editor*.

8. Make sure to save your project.

Note:

The **THT** (token holding time) specifies the maximum amount of time a token ring MAC (media access control) may use the token before releasing it.

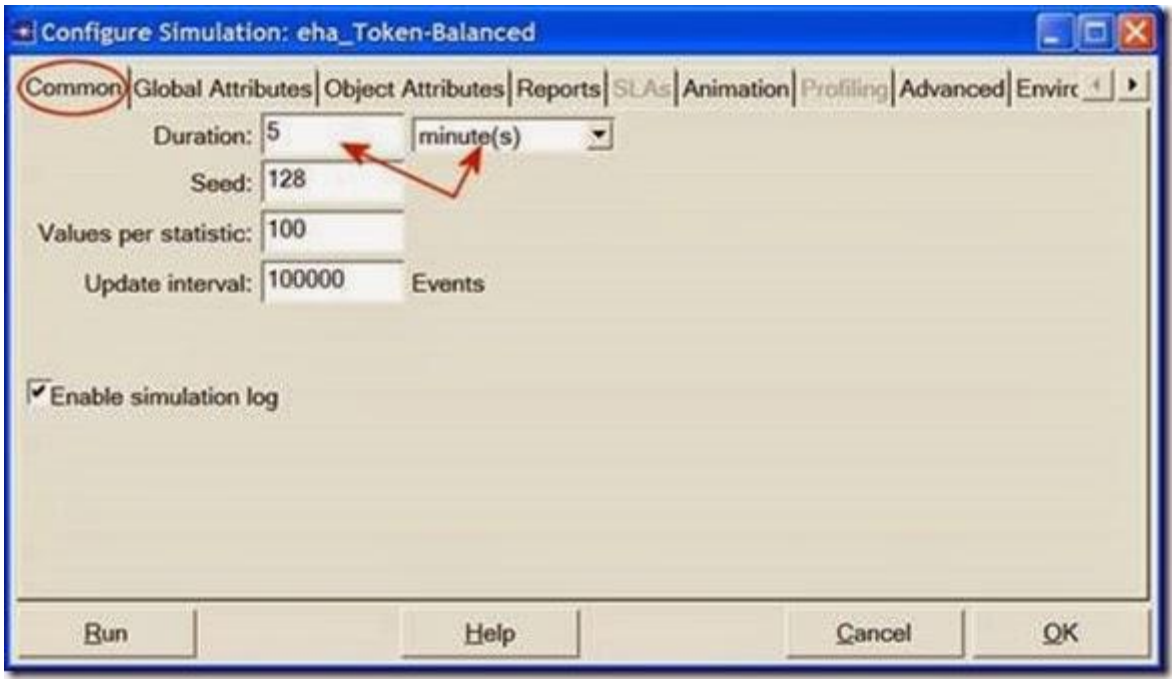
The **interarrival time** is the time between successive packet generations in the "ON" state.

Configure the Simulation

To examine the network performance under different THTs, you need to run the simulation several times by changing THT with every run of the simulation. There is an easy way to do that. Recall that we promoted the**THT Duration** attribute. Here we will assign different values to that attribute:

1. Click on the **Configure/Run Simulation** button:

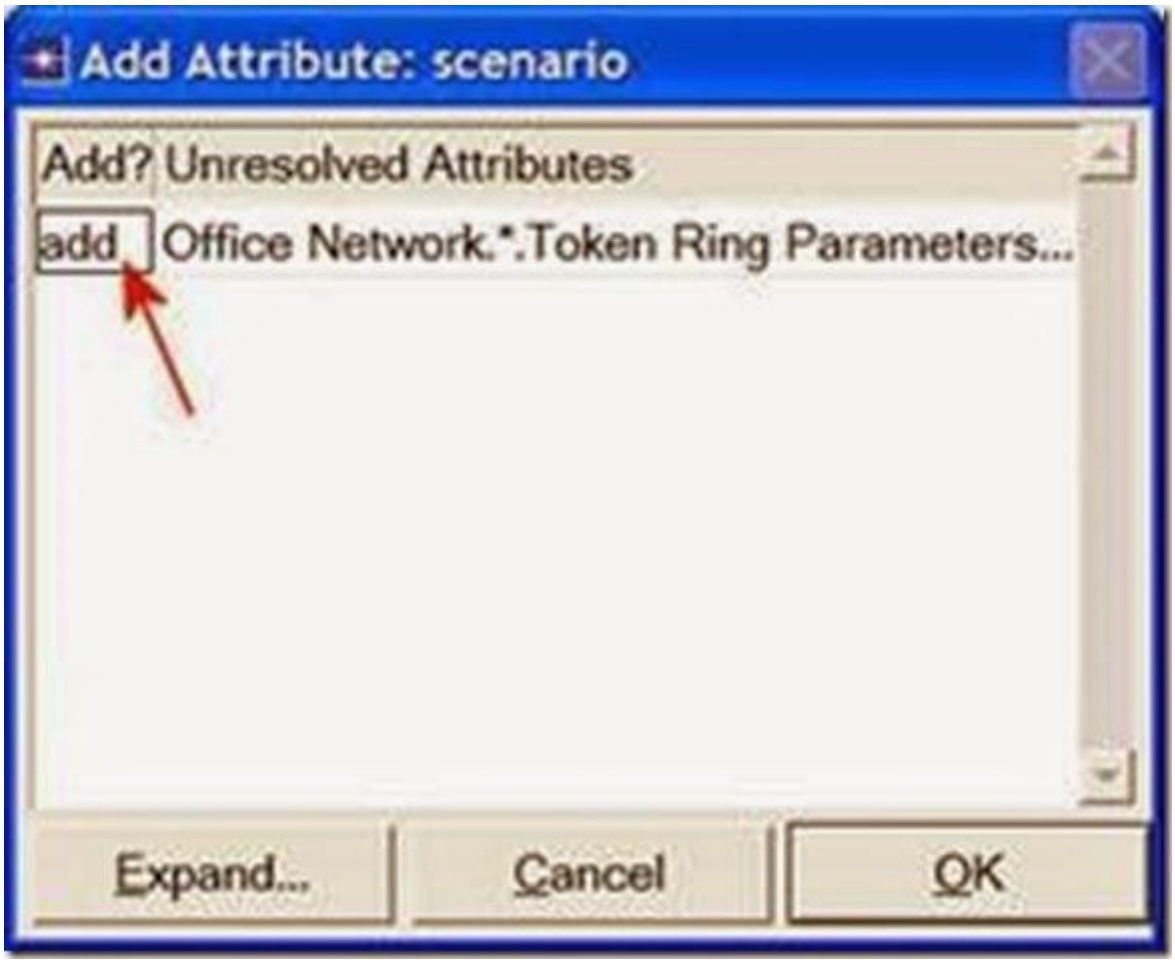
2. Make sure that the **Common** tab is chosen ⇒ Assign **5** minutes to the**Duration.**



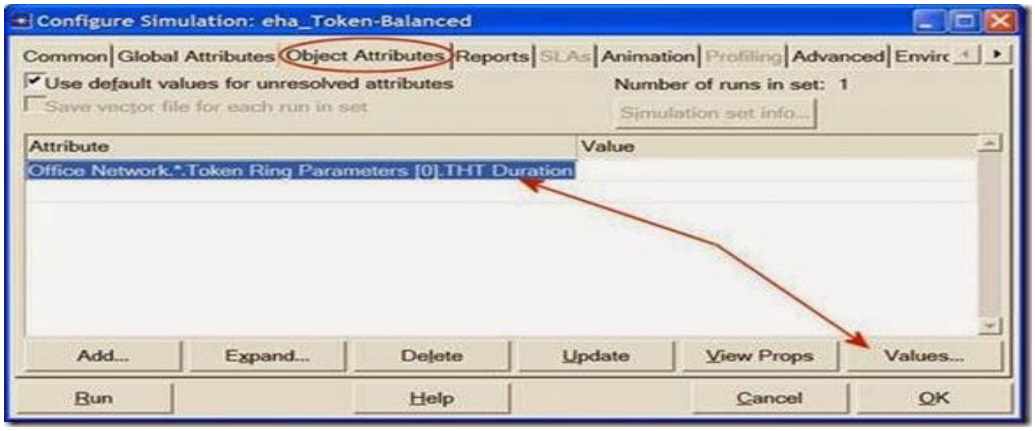
3. Click on the **Object Attributes** tab ⇒ Click the **Add** button.

4. As shown in the following *Add Attribute* dialog box, you need to add the**THT Duration** attribute for all nodes. To do that:

a. Add the unresolved attribute: **Office Network.*.Token RingParameters[0].THT Duration** by clicking on the corresponding cell under the **Add?** column ⇒ Click **OK**

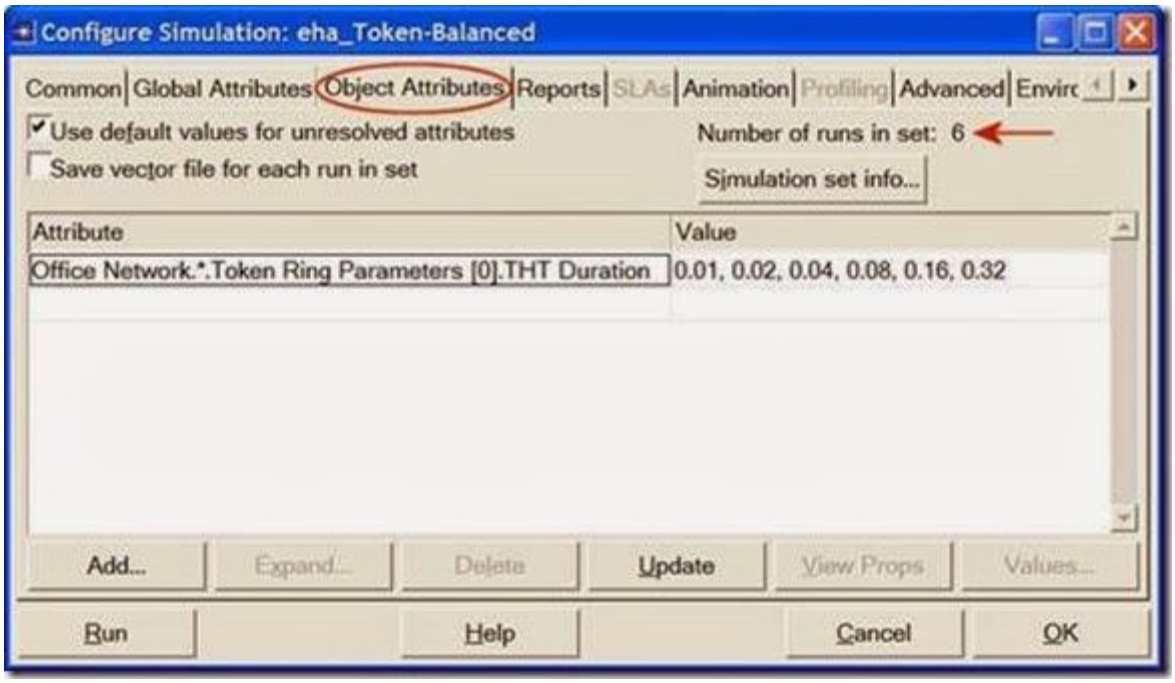


5. Now you should see the **Office Network.*.Token Ring Parameters[0].THT Duration** in the list of simulation object attributes (widen the "Attribute" column to see the full name of the attribute). Click on that attribute ⇒ Click the **Values** button, as shown below.



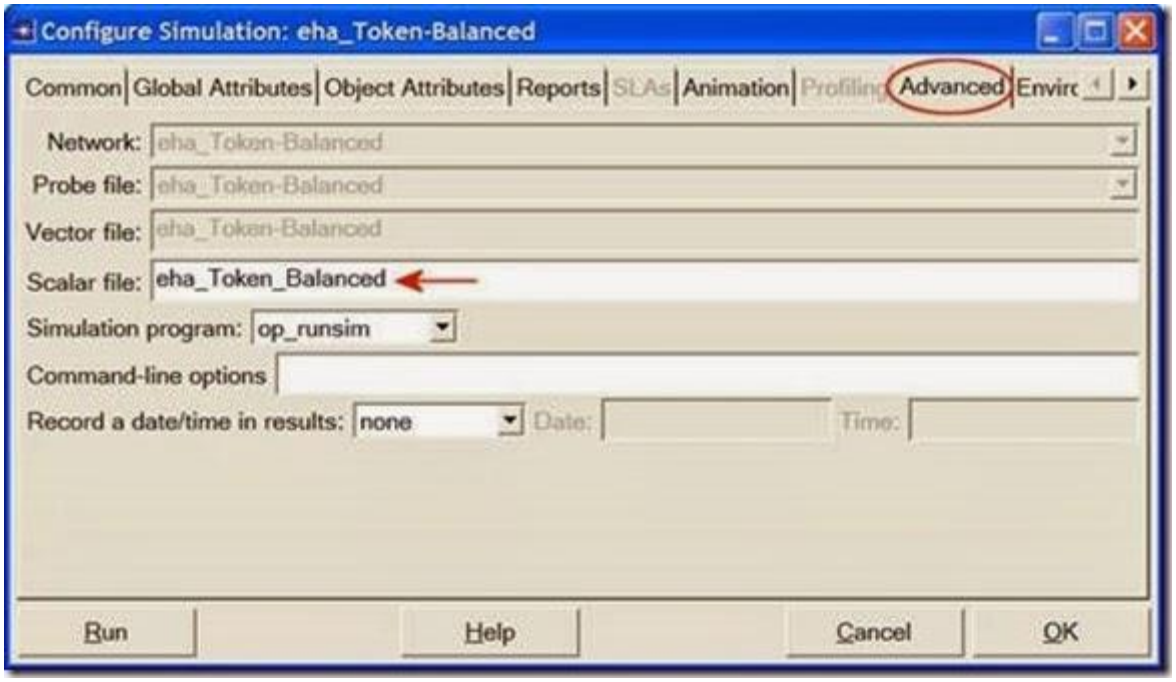
6. Add the following six values. (*Note:* To add the first value, double-click on the first cell in the **Value** column ⇒ Type "0.01" into the textbox and hit enter. Repeat this for all six values.)

7. Click **OK**. Now look at the upper-right corner of the *Simulation Configuration* dialog box and make sure that the *Number of runs in set* is **6**.



8. For each of the six simulation runs we need the simulator to save "scalar" values that represent the "average" values of the statistics to be collected from the simulation. To save these scalars we need to configure the simulator to save them in a file. Click on the **Advanced** tab in the *Configure Simulation* dialog box.

9. Assign <**your initials**>**_Token_Balanced** to the *Scalar file* text field.



10. Click **OK** and then save your project.

Choose the Statistics

The **utilization** is a measure of the bandwidth used versus that available on the given ring.

To choose the statistics to be collected during the simulation:

- Right-click anywhere in the project workspace (but not on a node or link) and select Choose Individual Statistics from the pop-up menu.
- Expand the Global Statistics hierarchy:

− Expand the **Traffic Sink** hierarchy ⇒ Click the check box next to **Traffic Received (packets/sec).**

− Expand the **Traffic Source** hierarchy ⇒ Click the check box next to**Traffic Sent (packets/sec).**

b. Expand the **Node Statistics** hierarchy:

− Expand the **Token Ring** hierarchy ⇒ Click the check box next to**Utilization.**

c. Click **OK**.

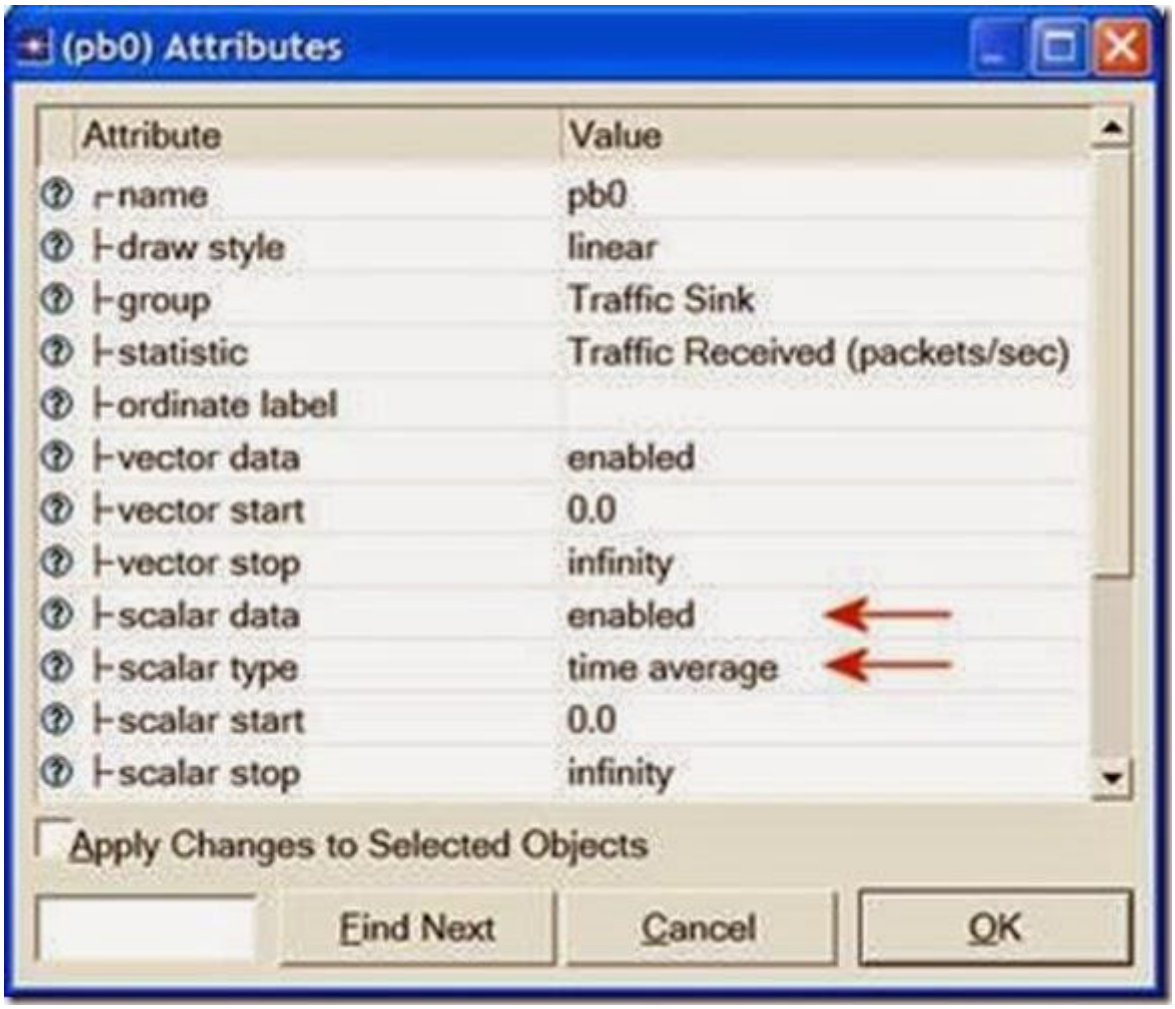
2. Now we want to collect the average of the above statistics as a scalar value by the end of each simulation run.

1. Select **Choose Statistics (Advanced)** from the *Simulation*menu.

b. The **Traffic Sent** and **Traffic Received** probes should appear under the

Global Statistic Probes. The **Utilization** probe should appear under the**Node Statistics Probes**.

c. Right-click on **Traffic Received** probe ⇒ **Edit Attributes**. Set the **scalardata** attribute to **enabled** ⇒ Set the **scalar type** attribute to **time average** ⇒Compare to the following figure and click **OK**.

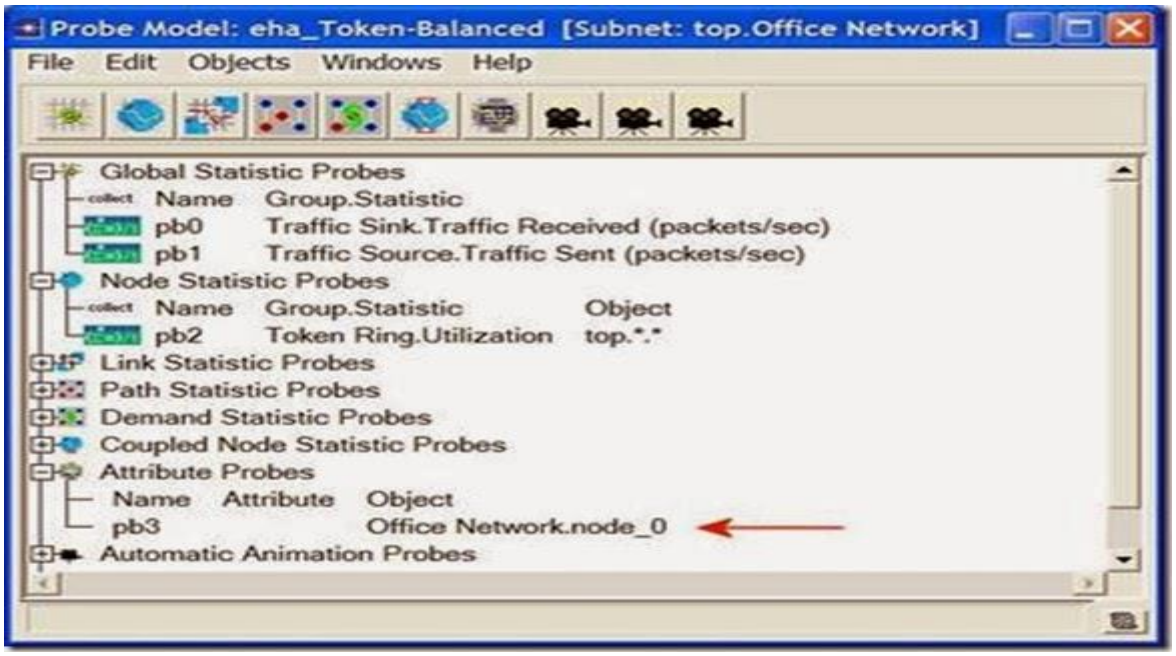


d. Repeat the previous step with the **Traffic Sent** and **Utilization** probes.

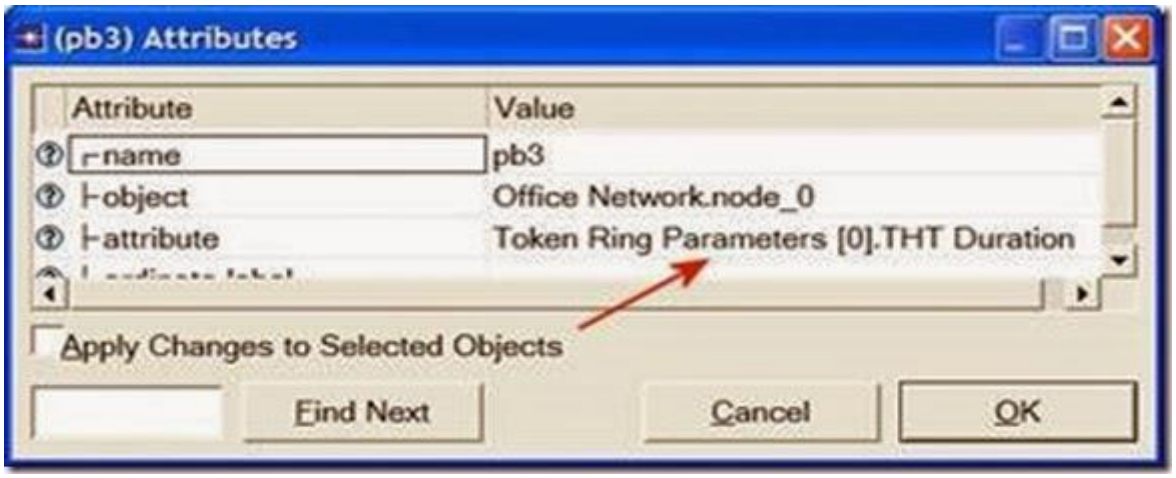
3. Since we need to analyze the effect of THT on the network performance, THT must be added as an "input" statistic to be recorded by the simulation. To do that:

a. Select **Create Attribute Probe** from the **Objects** menu. Now a new attribute is created under the **Attribute Probes** hierarchy as shown.

b. Right-click on the new attribute probe and select **Choose AttributedObject** from the pop-up menu ⇒ Expand the **Office Network** hierarchy ⇒ Click on **node_0** (actually you can pick any other node) ⇒ Click **OK**.



c. Right-click again on the new attribute probe and select **Edit Attributes**from the pop-up menu ⇒ Assign the **Token Ring Parameter[0].THTDuration** value to the "**attribute**" Attribute, as shown in the figure ⇒ Click**OK**.



4. Select save from the **File** menu in the *Probe Model* window and then**Close** the window.

5. Now you are back to the *Project Editor*. Make sure to save your project.

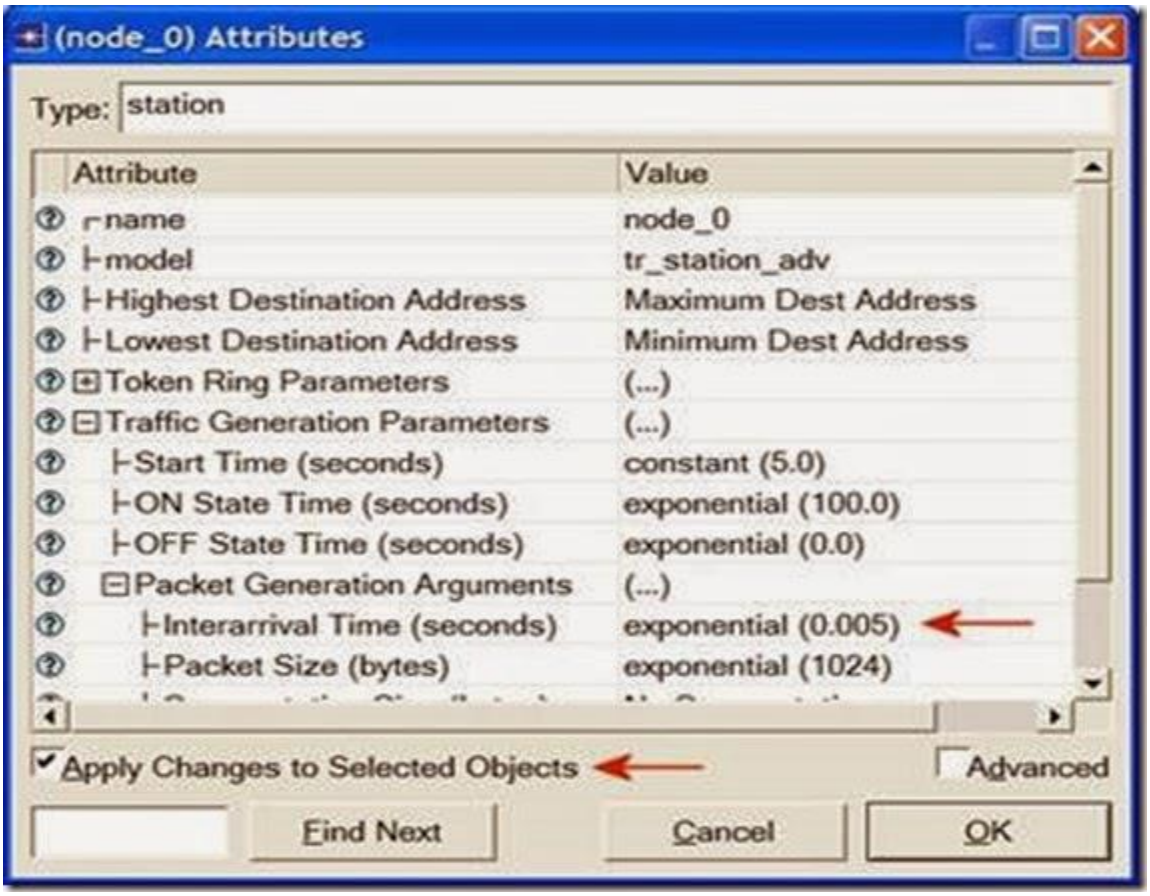
Duplicate the Scenario

The token ring network scenario we just implemented is *balanced:* the distribution of the generated traffic in all nodes is the same. To compare performance, you will create an "unbalanced" scenario as follows:

1. Select **Duplicate Scenario** from the **Scenarios** menu and give it the name

Unbalanced ⇒ Click **OK**.

2. Select **node_0** and **node_7** by shift-clicking on both nodes ⇒ Right-click on one of these two selected nodes and select **Edit Attributes** ⇒ Expand the**Traffic Generation Parameters** hierarchy ⇒ Expand the **Packet Generation Arguments** hierarchy ⇒ Change the value of the **Interarrival Time** attribute to **exponential(0.005)** as shown. Make sure to check the**Apply Changes to Selected Objects** box before you click **OK**.



3. Select all nodes except **node_0** and **node_7** ⇒ Right-click on one of the selected nodes and select **Edit Attributes** ⇒ Change the value of the**Interarrival Time** attribute to **exponential(0.075)** as in the previous step. Make sure to check the **Apply Changes to Selected Objects** box before you click **OK**.

4. Click anywhere in the workspace to unselect objects.

5. Click on the **Configure/Run Simulation** button: ⇒ Click on the**Advanced** tab in the Configure Simulation dialog box ⇒

Assign <**your initials**>**_Token_Unbalanced** to the *Scalar file* text field.

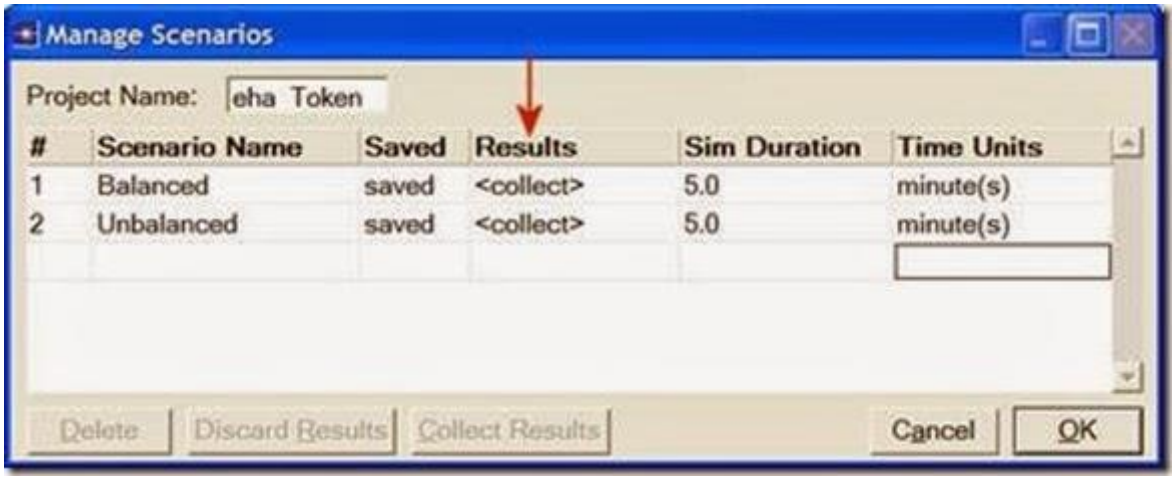
6. Click **OK** and then save your project.

Run the Simulation

To run the simulation for both scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Change the values under the **Results** column to **<collect>** (or**<recollect>**) for both scenarios. Compare to the following figure.



3. Click **OK** to run the simulations. Depending on the speed of your processor, this may take several minutes to complete.

4. After the simulation completes the 12 runs, 6 for each scenario, click**Close**.

5. Save your project.

When you rerun the simulation, OPNET IT Guru will "append" the new results to the results already in the scalar file. To avoid that, delete the scalar file before you start a new run.

• Go to the **File** menu ⇒ Select **Model Files** ⇒ **Delete Model Files** ⇒ From the list, choose **other model types** ⇒ Select **(.os): Output Scalars** ⇒ Select the scalar file to be deleted; in this lab they are<**your initials**>**_Token_Balanced_Scalar** and <**your initials**>**_Token_Unbalanced_Scalar** ⇒ Click **Close**.

View the Results

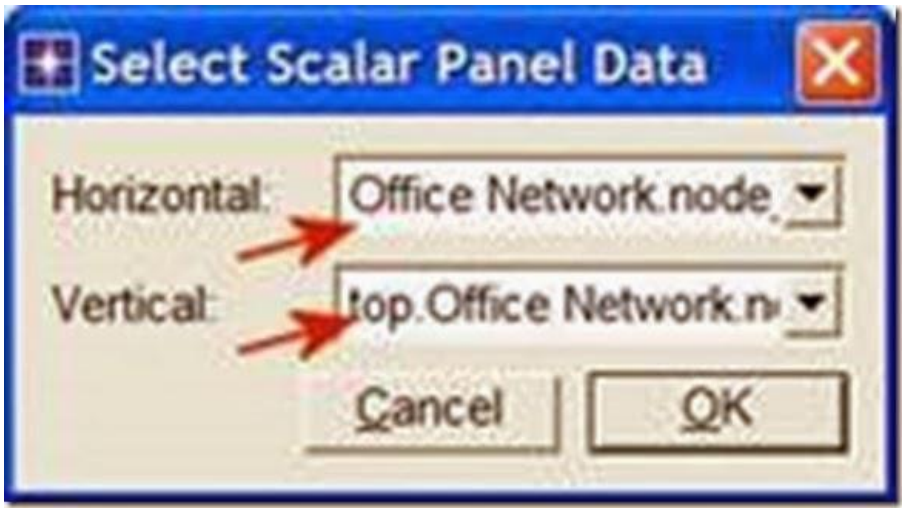
To view and analyze the results:

1. Select View Results (Advanced) from the Results menu. Now theAnalysis Configuration tool is open.

2. Recall that we saved the average results in two scalar files, one for each scenario. To load the scalar file for the **Balanced** scenario, select **Load Output**

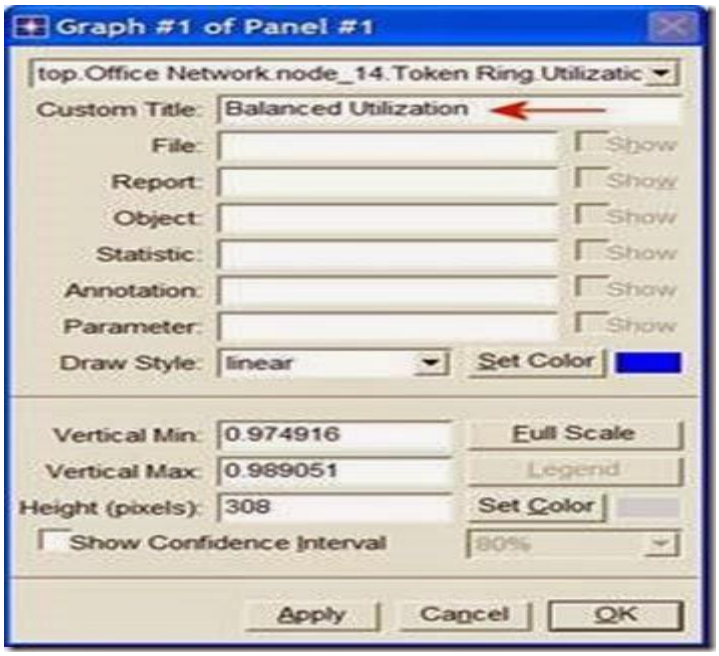
Scalar File from the **File** menu ⇒ Select <**your initials**>**_Token_Balanced**from the pop-up menu.

3. Select **Create Scalar Panel** from the **Panels** menu ⇒ Select the scalar panel data as shown in the following dialog box: **THT** for **Horizontal** and**Utilization** for **Vertical**. (*Note:* If any of the data is missing, make sure that you carried out steps 2.c and 2.d in the *Choose the Statistics* section.)

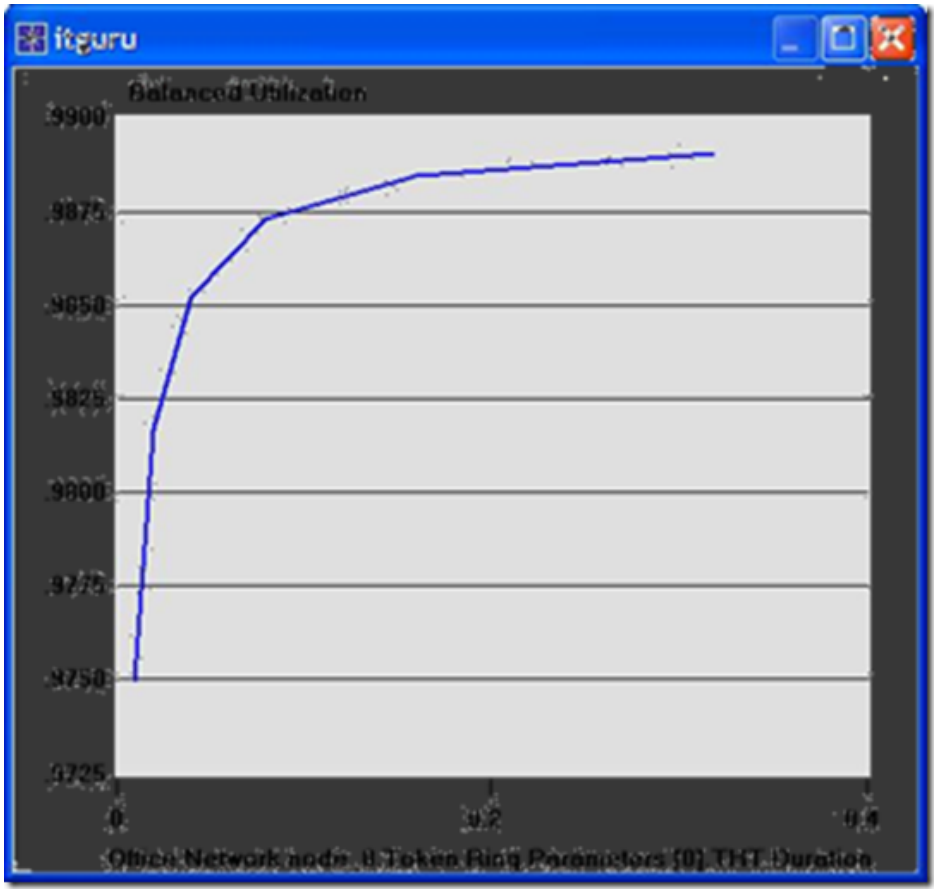


4. Click **OK**.

5. To change the title of the graph, right-click on the graph area and choose**Edit Graph Properties** ⇒ Change the **Custom Title** to **Balanced Utilization** as shown.



6. Click **OK**. The resulting graph should resemble the one shown below. Do not close the graph and continue with the following step.

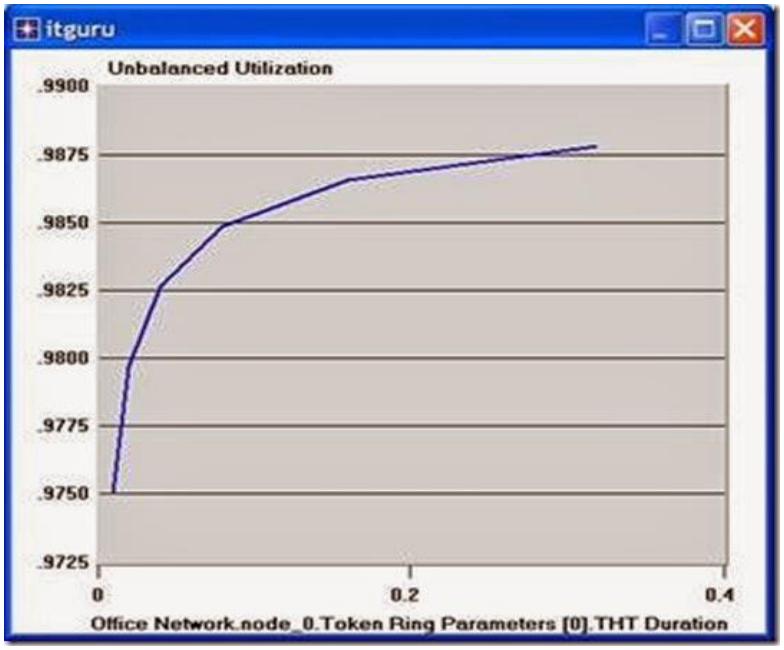


7. To compare with the **Unbalanced** scenario, load its scalar file, select**Load Output Scalar File** from the **File** menu ⇒

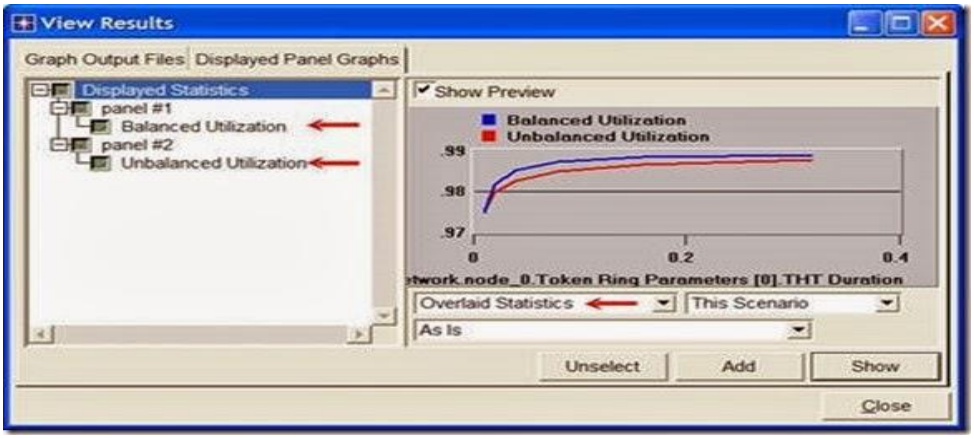
Select <**your initials**>**_Token_Unbalanced** from the pop-up menu.

8. Select **Create Scalar Panel** from the **Panels** menu ⇒ Select the scalar panel data as in step 3.

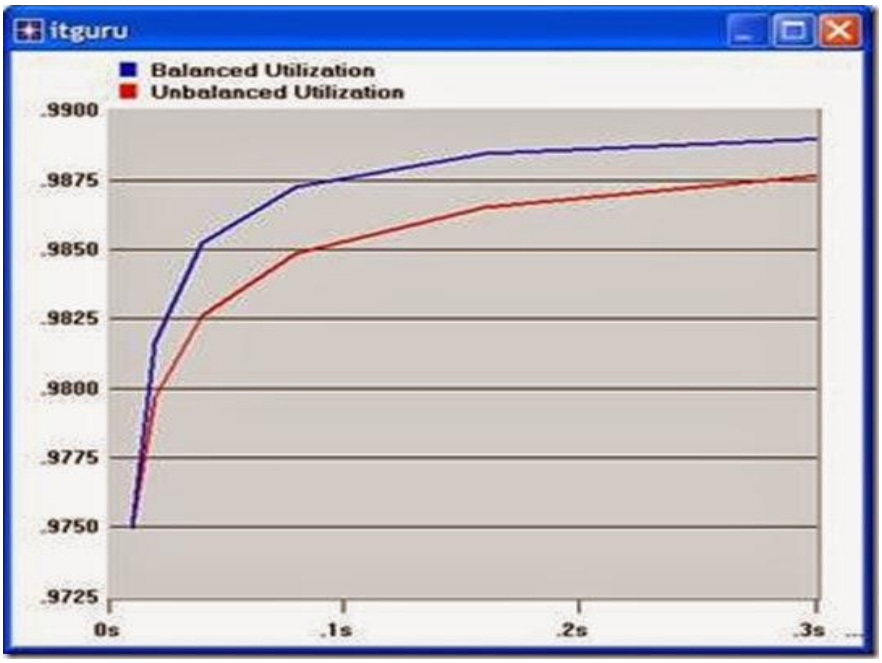
9. Click **OK** ⇒ Change the graph title to **Unbalanced** as in step 5 ⇒ Click**OK**. The resulting graph should resemble the one shown below. Do not close this graph or the previous one and continue with the following step.



10. To combine the above two graphs on a single graph, select **Create Vector Panel** from the **Panels** menu ⇒ Click on the **Display Panel Graphs**tab ⇒ Select both **Balanced** and **Unbalanced** statistics ⇒ Choose **Overlaid Statistics** from the drop-down menu in the right-bottom area of the dialog box as shown.

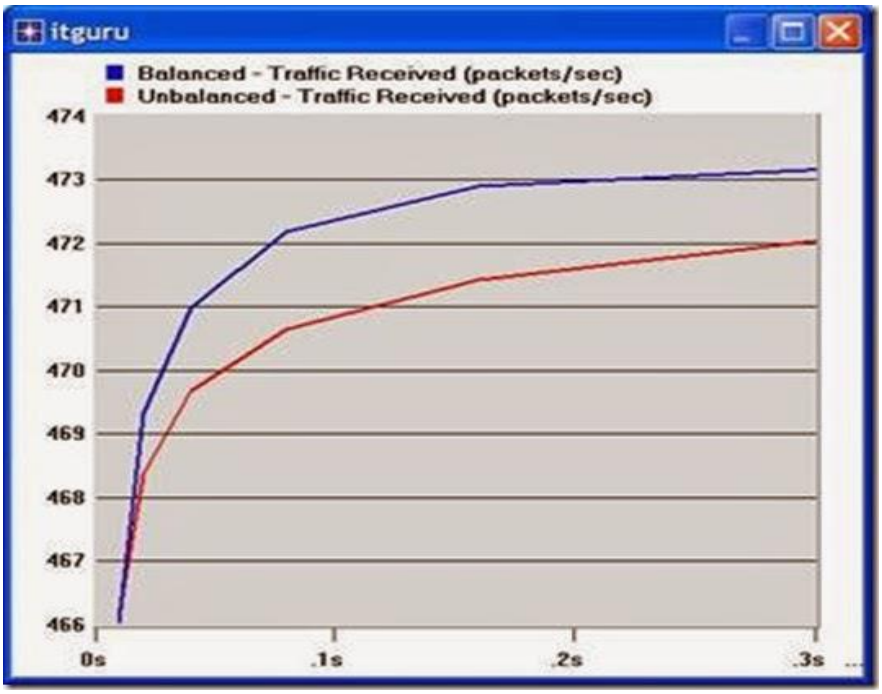


11. Click **Show** and the resulting graph should resemble the one shown below.



12. Repeat the same process to check the effect of the THT on Traffic Received for both scenarios. Assign the appropriate titles to the graphs.

13. The resulting graph, which combines the Traffic Received statistic for both the Balanced and Unbalanced scenarios, should resemble the following one:



Lab 23 – A set of Local Area Network on IT Guru

Objectives

The objectives of this lab is to apply LAN on IT Guru

Lab Tasks

Task: Apply LAN on IT Guru.

There is a limit to how many hosts can be attached to a single network and to the size of a geographic area that a single network can serve. Computer networks use switches to enable the communication between one host and another, even when no direct connection exists between those hosts. A switch is a device with several inputs and outputs leading to and from the hosts that the switch interconnects. The core job of a switch is to take packets that arrive on an input and forward (or switch) them to the right output so that they will reach their appropriate destination.

A key problem that a switch must deal with is the finite bandwidth of its outputs. If packets destined for a certain output arrive at a switch and their arrival rate exceeds the capacity of that output, then we have a problem of contention. In this case, the switch will queue, or buffer, packets until the contention subsides. If it lasts too long, however, the switch will run out of buffer space and be forced to discard packets. When packets are discarded too frequently, the switch is said to be congested.

In this lab you will set up switched LANs using two different switching devices: hubs and switches. A hub forwards the packet that arrives on any of its inputs on all the outputs regardless of the destination of the packet. On the other hand, a switch forwards incoming packets to one or more outputs depending on the destination(s) of the packets. You will study how the throughput and collision of packets in a switched network are affected by the configuration of the network and the types of switching devices that are used.

Procedure

Create a New Project

1. Start the **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the**File** menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_SwitchedLAN**, and the scenario **OnlyHub** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that**Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Choose **Office** from the *Network Scale* list ⇒ Click **Next** three times ⇒ Click **OK**.

4. **Close** the *Object Palette* dialog box.

Create the Network

The prefix **ethernet16_** indicates that the device supports up to 16 Ethernet connections.

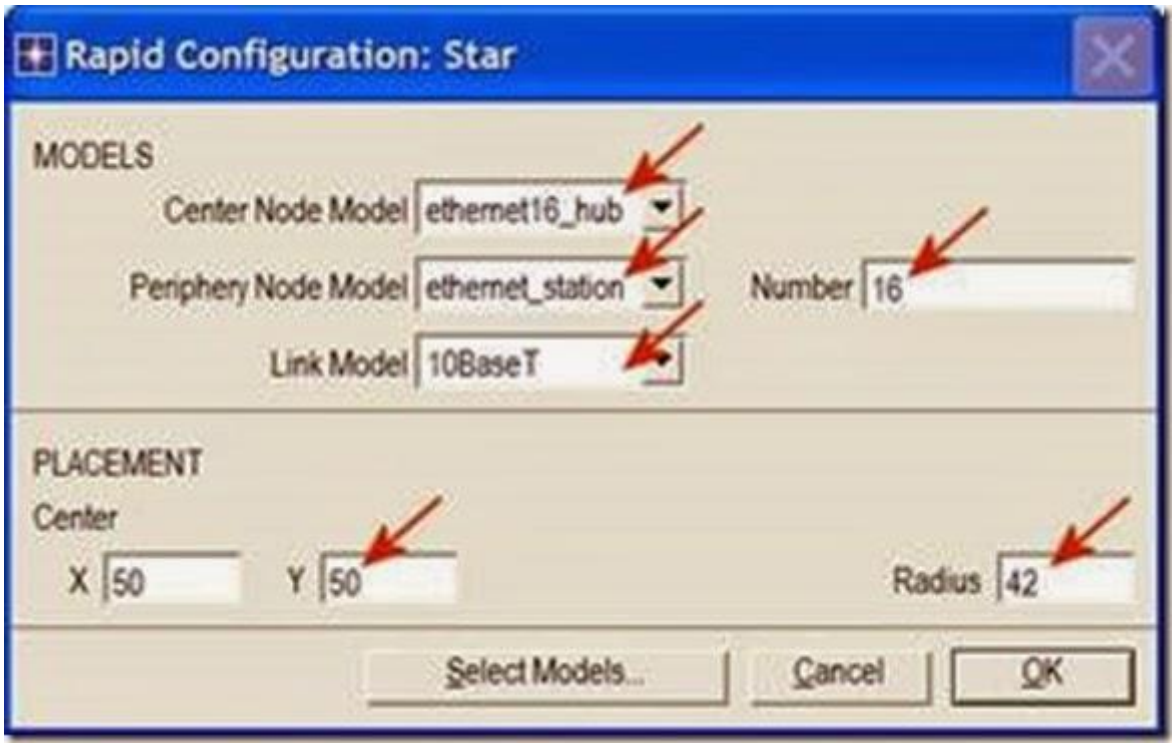
The **10BaseT** link represents an Ethernet connection operating at 10 Mbps.

To create our switched LAN:

1. Select **Topology** ⇒ **Rapid Configuration**. From the drop-down menu choose **Star** and click **OK**.

2. Click the **Select Models** button in the *Rapid Configuration* dialog box. From the *Model List* drop-down menu choose **ethernet** and click **OK**.

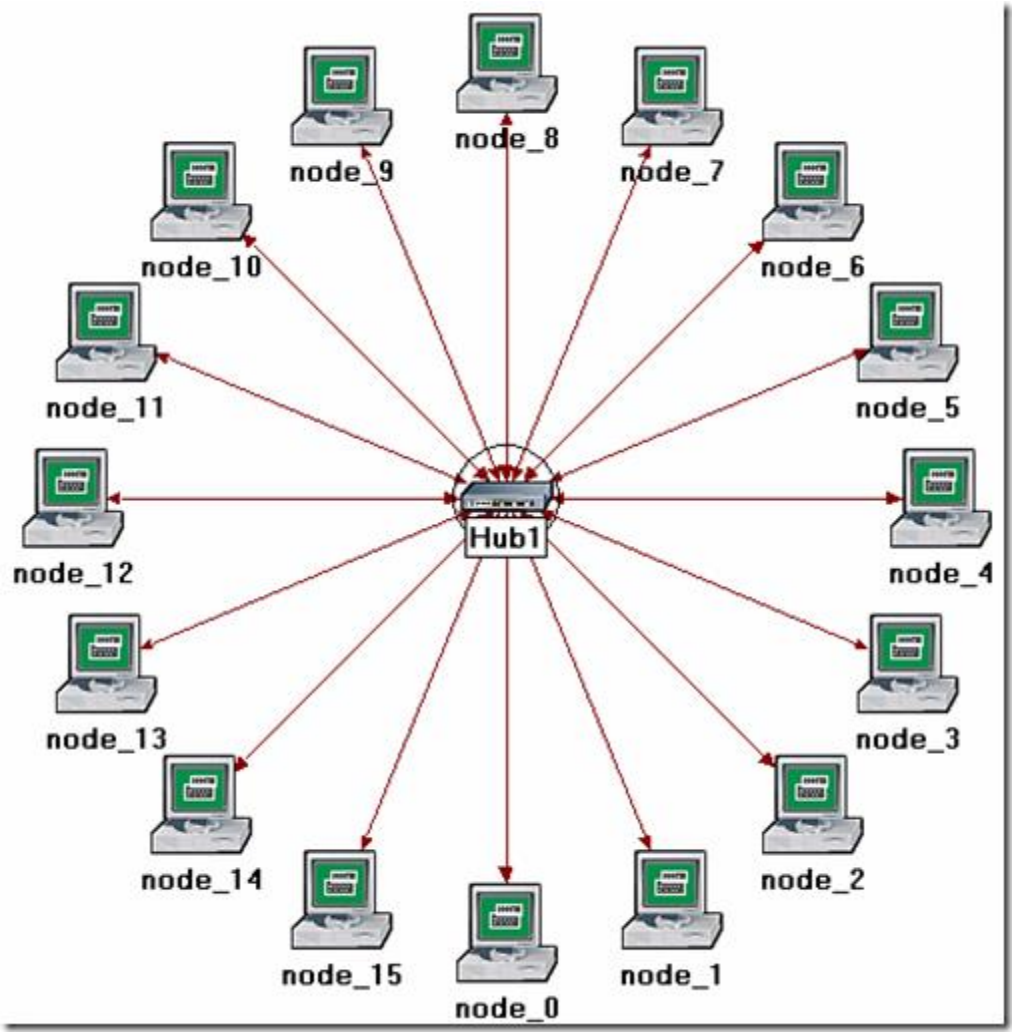
3. In the *Rapid Configuration* dialog box, set the following five values:**Center Node Model = ethernet16_hub**, **Periphery Node Model = ethernet_station**, **Link Model = 10BaseT**, **Number=16**, **Y=50**, and **Radius = 42** ⇒ Click **OK**.



4. Right-click on **node_16**, which is the hub ⇒ **Edit Attributes** ⇒ Change the **name** attribute to **Hub1** and click **OK**.

5. Now that you have created the network, it should look like the following one.

6. Make sure to save your project.



Configure the Network Nodes

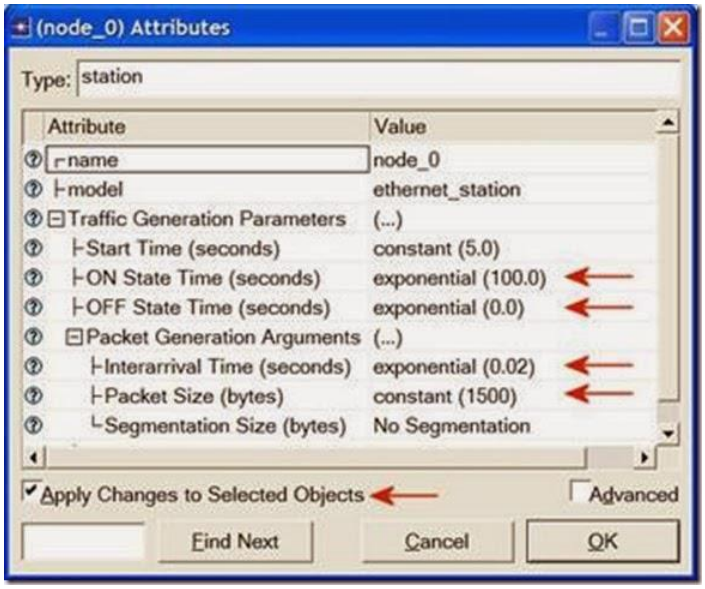
Here you will configure the traffic generated by the stations.

1. Right-click on any of the 16 stations (node_0 to node_15) ⇒ **Select Similar Nodes**. Now all stations in the network are selected.

2. Right-click on any of the 16 stations ⇒ **Edit Attributes.**

a. Check the **Apply Changes to Selected Objects** check box. This is important to avoid reconfiguring each node individually.

3. Expand the hierarchies of the **Traffic Generation Parameters** attribute and the **Packet Generation Arguments** attribute ⇒ Set the following four values:



4. Click **OK** to close the attribute editing window(s). Save your project.

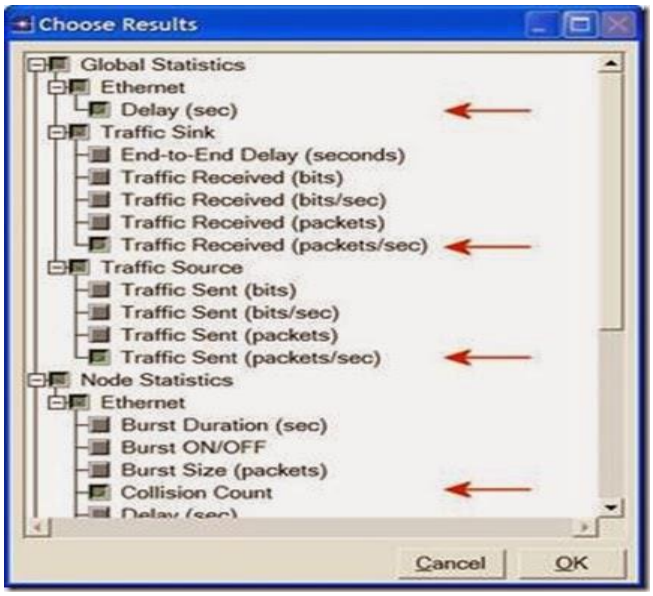
Choose Statistics

To choose the statistics to be collected during the simulation:

1. Right-click anywhere in the project workspace and select **Choose Individual Statistics** from the pop-up menu.

2. In the *Choose Results* dialog box, choose the following four statistics:

The **Ethernet Delay** represents the end to end delay of all packets received by all the stations.



Traffic Received (in packets/sec) by the traffic sinks across all nodes.

Traffic Sent (in packets/sec) by the traffic sources across all nodes.

Collision Count is the total number of collisions encountered by the hub during packet transmissions.

3. Click **OK**.

Configure the Simulation

Here we need to configure the duration of the simulation:

1. Click on the **Configure/Run Simulation** button:

2. Set the duration to be **2.0 minutes**.

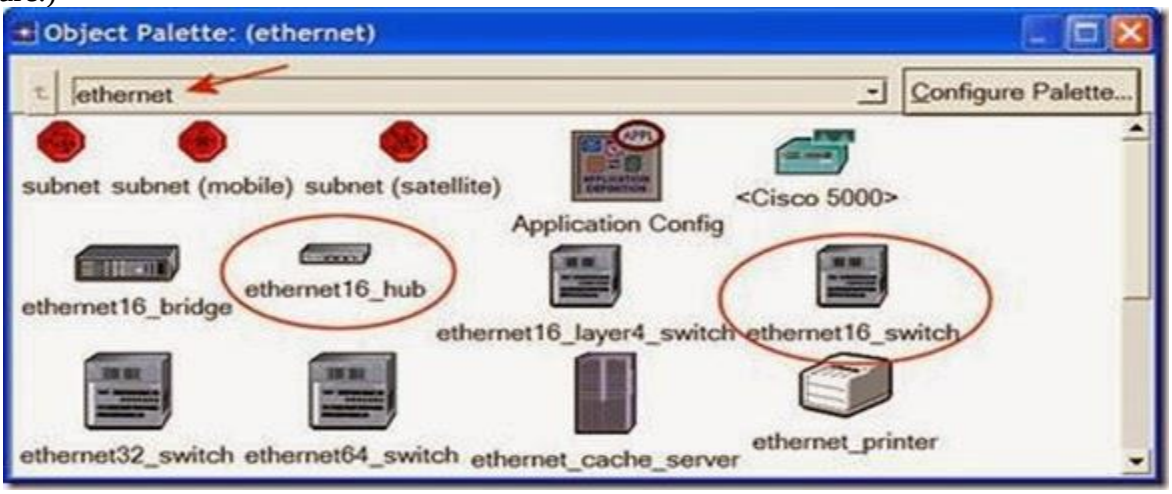
3. Click **OK**.

Duplicate the Scenario

The network we just created utilizes only one hub to connect the 16 stations. We need to create another network that utilizes a switch and see how this will affect the performance of the network. To do that we will create a duplicate of the current network:

1.Select **Duplicate Scenario** from the **Scenarios** menu and give it the name**HubAndSwitch** ⇒ Click **OK**.

2. Open the **Object Palette** by clicking on . Make sure that **Ethernet** is selected in the pull-down menu on the object palette.
3. We need to place a hub and a switch in the new scenario. (They are circled in the following figure.)



4. To add the **Hub**, click its icon in the object palette ⇒ Move your mouse to the workspace ⇒ Click to drop the hub at a location you select. Right-click to indicate you are done deploying hub objects.

5. Similarly, add the **Switch**

6. Close the **Object Palette**.

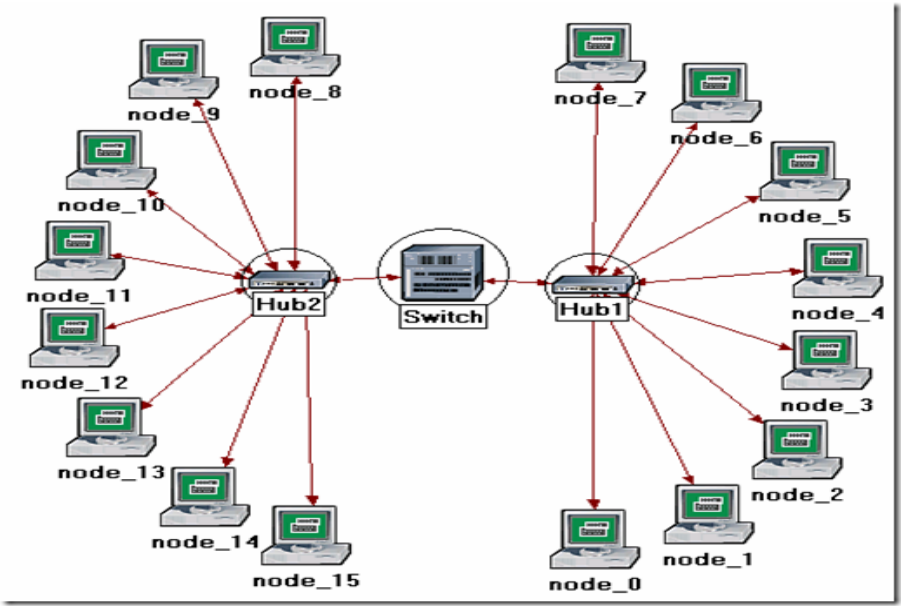
7. Right-click on the new hub ⇒ **Edit Attributes** ⇒ Change the **name**attribute to **Hub2** and click **OK**.

8. Right-click on the switch ⇒ **Edit Attributes** ⇒ Change the **name**attribute to **Switch** and click **OK**.

9. Reconfigure the network of the **HubAndSwitch** scenario so that it looks like the following one.

Hints: a. To remove a link, select it and choose **Cut** from the **Edit** menu (or simply hit the **Delete** key). You can select multiple links and delete all of them at once.

b. To add a new link, use the **10BaseT** link available in the **Object Palette**.



10. Save your project.

Run the Simulation

To run the simulation for both scenarios simultaneously:

1. Select **Manage Scenarios** from the **Scenarios** menu.

2. Change the values under the **Results** column to **<collect>** (or**<recollect>**) for both scenarios. Compare to the following figure.

3. Click **OK** to run the two simulations. Depending on the speed of your processor, this may take several minutes to complete.

4. After the two simulation runs complete, one for each scenario, click**Close**.

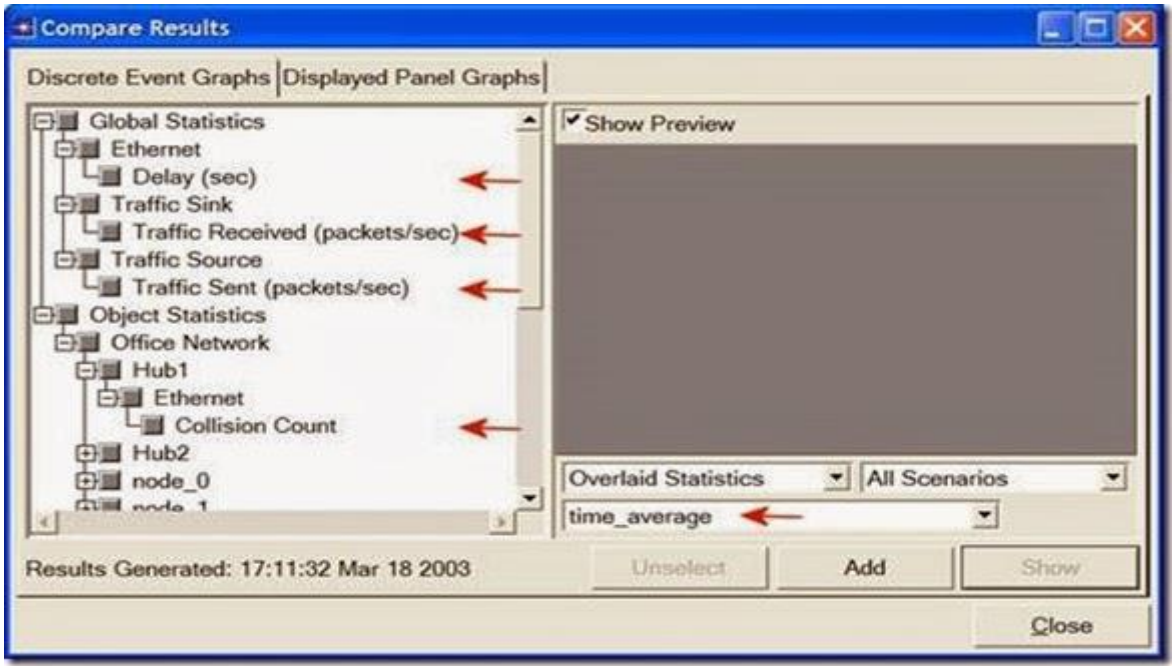
5. Save your project.

View the Results

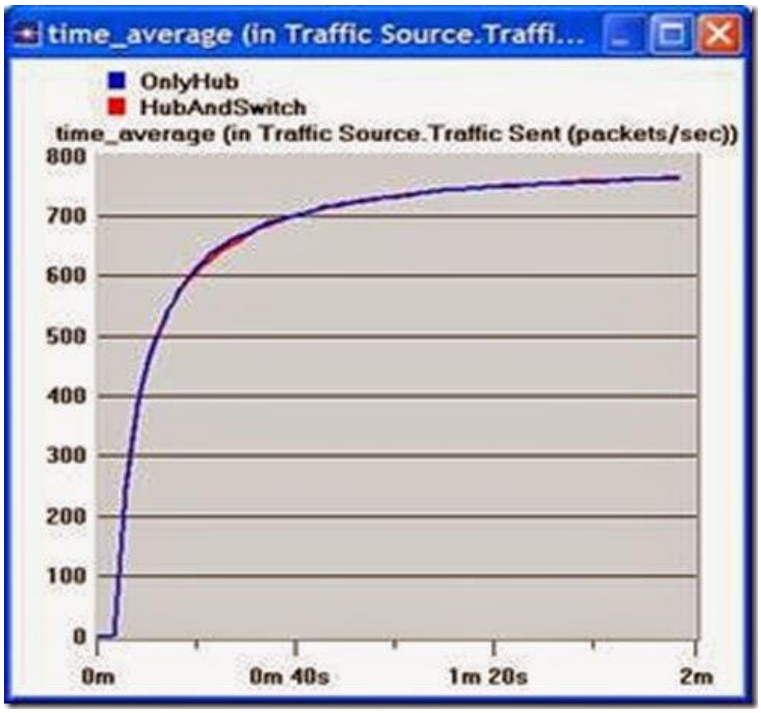
To view and analyze the results:

1. Select **Compare Results** from the **Results** menu.

2. Change the drop-down menu in the lower-right part of the *Compare Results* dialog box from **As Is** to **time_average,** as shown.

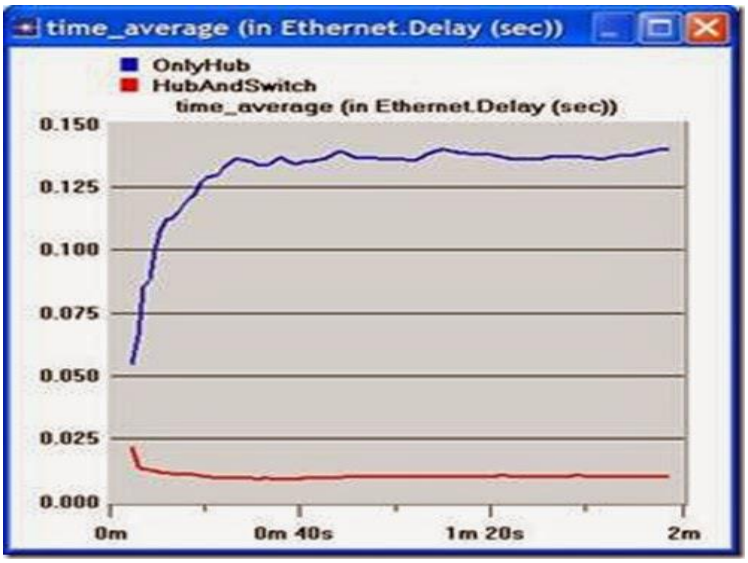


3. Select the **Traffic Sent (packets/sec)** statistic and click **Show**. The resulting graph should resemble the one below. As you can see, the traffic sent in both scenarios is almost identical.



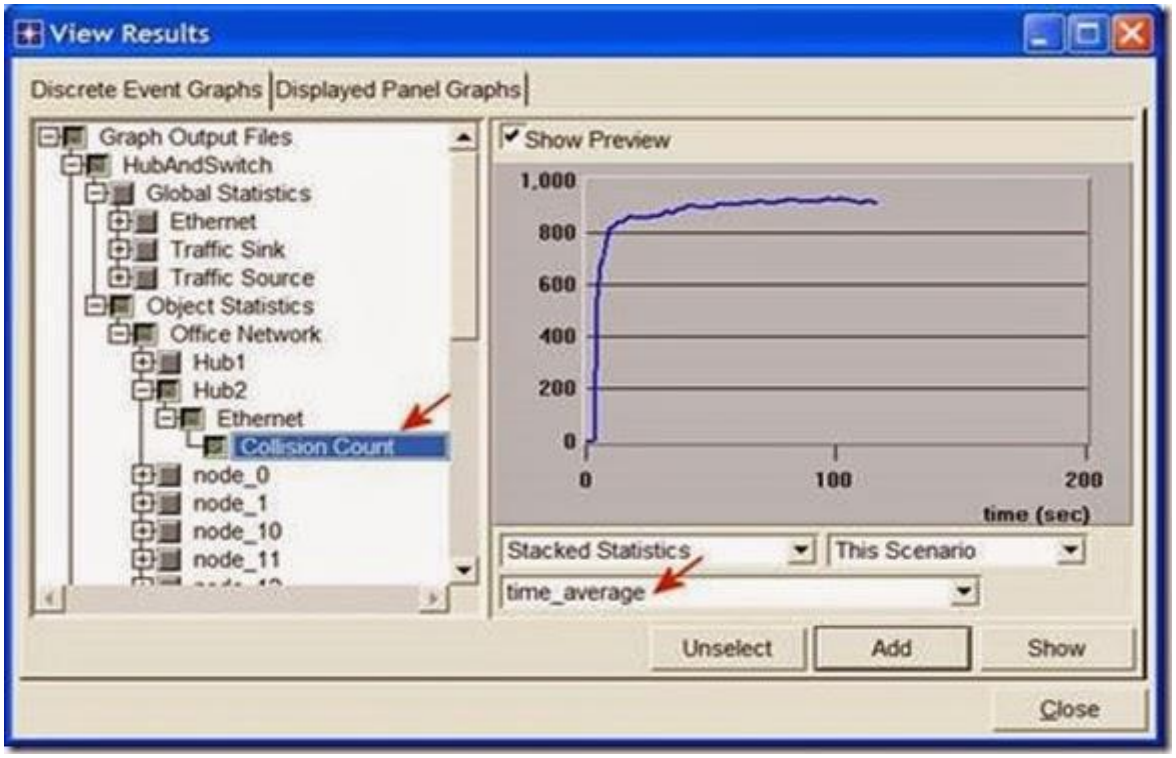
4. Select the **Traffic Received (packets/sec)** statistic and click **Show**. The resulting graph should resemble the one below. As you see, the traffic received with the second scenario, **HubAndSwitch**, is higher than that of the**OnlyHub** scenario.

5. Select the **Delay (sec)** statistic and click **Show**. The resulting graph should resemble the one below. (*Note:* Result may vary slightly due to different node placement.)

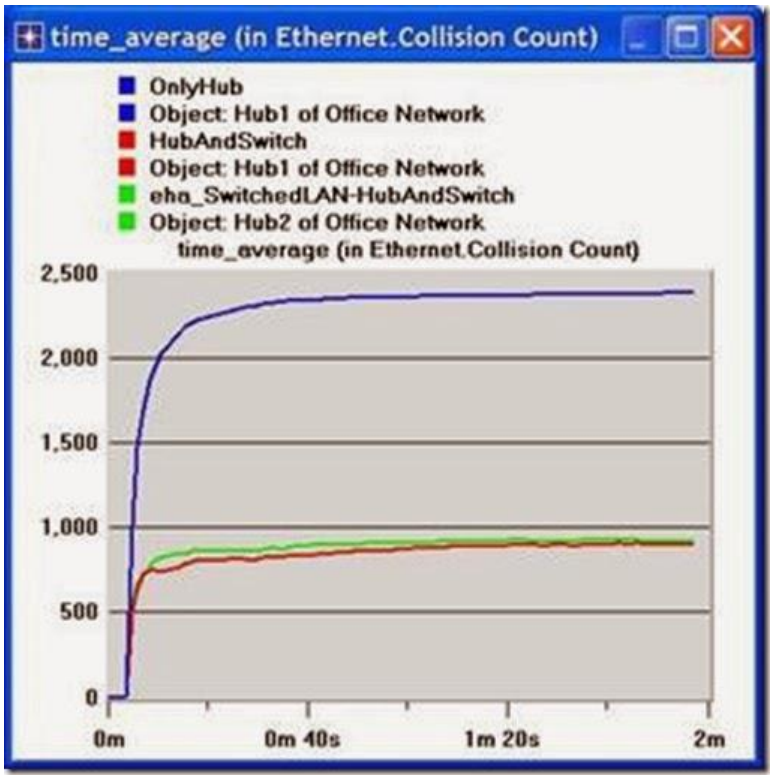


6. Select the **Collision Count** statistic for **Hub1** and click **Show**.

7. On the resulting graph right-click anywhere on the *graph area* ⇒ Choose**Add Statistic** ⇒ Expand the hierarchies as shown below ⇒ Select the**Collision Count** statistic for **Hub2** ⇒ Change **As Is** to **time_average** ⇒Click **Add.**



8. The resulting graph should resemble the one below.



9. Save your project.

Note: time_average is the average value over time of the values generated during the collection window. This average is performed assuming a "sample-and-hold" behavior of the data set (i.e., each value is weighted by the amount of time separating it from the following update and the sum of all the weighted values is divided by the width of the collection window). For example, suppose you have a 1-second bucket in which 10 values have been generated. The first 7 values were generated between 0 and 0.3 seconds, the 8th value at 0.4 seconds, the 9th value at 0.6 seconds, and the 10th at 0.99 seconds. Because the last 3 values have higher durations, they are weighted more heavily in calculating the time average.

Lab 24 – Planning a Network with different Users, Hosts and Services

Objectives

The objective of this lab is to demonstrate the basics of designing a network, taking into consideration the users, services, and locations of the hosts.

Lab Tasks

Task: Design a network with different users, hosts and services.

Optimizing the design of a network is a major issue. Simulations are usually used to analyze the conceptual design of the network. The initial conceptual design is usually refined several times until a final decision is made to implement the design. The objective is to have a design that maximizes the network performance, taking into consideration the cost constraints and the required services to be offered to different types of users. After the network has been implemented, network optimization should be performed periodically throughout the lifetime of the network to ensure maximum performance of the network and to monitor the utilization of the network resources.

In this lab you will design a network for a company that has four departments: Research, Engineering, E-Commerce, and Sales. You will utilize a LAN model that allows you to simulate multiple clients and servers in one simulation object. This model dramatically reduces both the amount of configuration work you need to perform and the amount of memory needed to execute the simulation. You will be able to define a profile that specifies the pattern of applications employed by the users of each department in the company. By the end of this lab, you will be able to study how different design decisions can affect the performance of the network.

Procedure

Create a New Project

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the **File**menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_NetDesign**, and the scenario **SimpleNetwork** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that**Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Choose **Campus**from the *Network Scale* list ⇒ Click **Next** ⇒ Choose **Miles** from the **Size**drop-down menu and assign **1** for both **X Span** and **Y Span** ⇒ Click **Next**twice ⇒ Click **OK**.

Create and Configure the Network

Application Config is used to specify applications that will be used to configure users profiles.

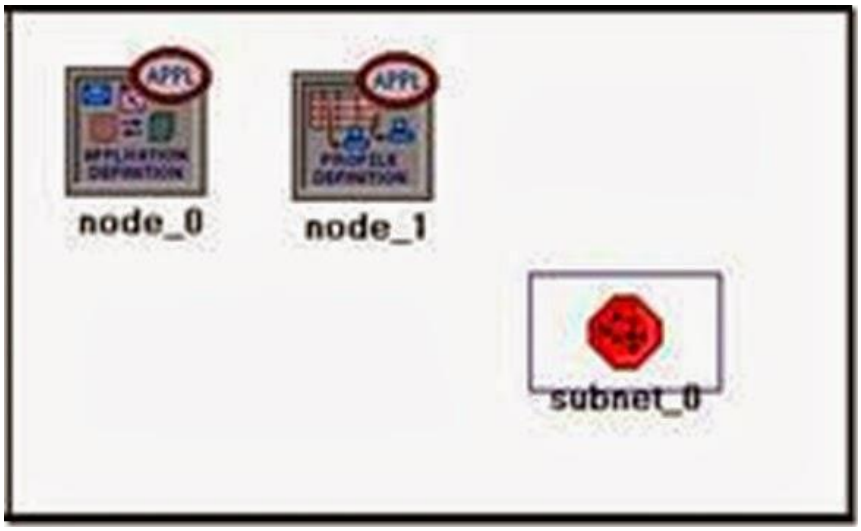
Profile Config describes the activity patterns of a user or group of users in terms of the applications used over a period of time. You must define the applications using the **Application Config** object before using this object.

Initialize the Network:

1. The Object Palette dialog box should be now on the top of your project space. If it is not there, open it by clicking . Make sure that the internet_toolbox is selected from the pull-down menu on the object palette.

2. Add to the project workspace the following objects from the palette: Application Config, Profile Config and a subnet.

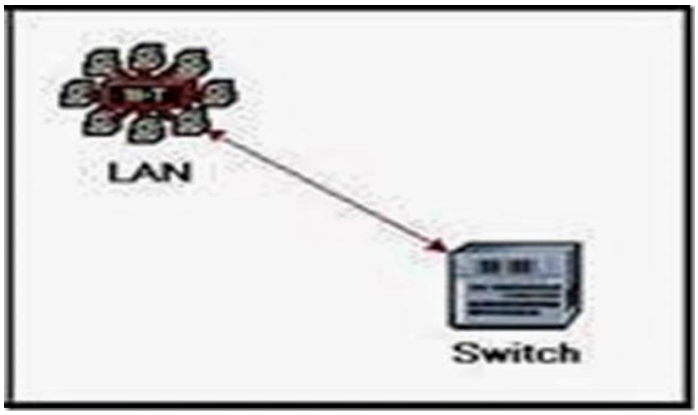
a. To add an object from a palette, click its icon in the object palette ⇒ Move your mouse to the workspace ⇒ Left-click to place the object. Right-click when finished. The workspace should contain the following three objects:



3. Close the *Object Palette* dialog box and save your project.

Configure the Services:

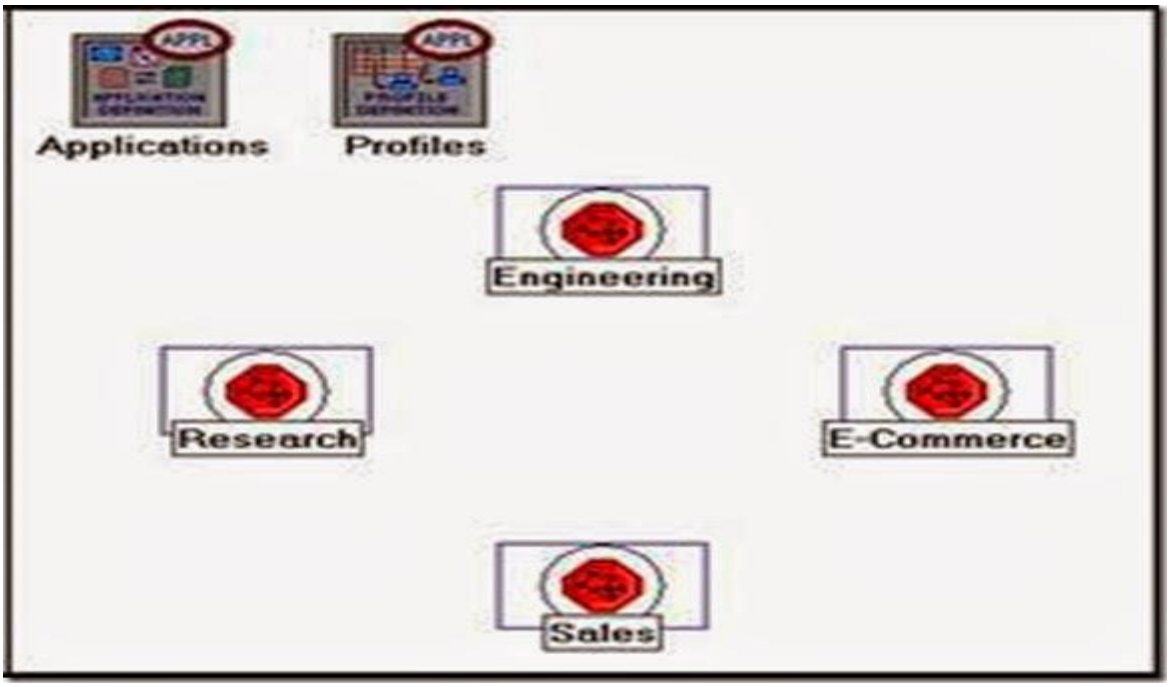
- Right-click on the Application Config node ⇒ Edit Attributes ⇒ Change the name attribute to Applications ⇒ Change theApplication Definitions attribute to Default ⇒ Click OK.
- Right-click on the Profile Config node ⇒ Edit Attributes ⇒ Change the name attribute to Profiles ⇒ Change the Profile Configuration attribute to Sample Profiles ⇒ Click OK.
- Sample Profiles provides patterns of applications employed by users such as engineers, researchers, salespeople, and multimedia users.
- Configure a Subnet:
- Right-click on the subnet node ⇒ Edit Attributes ⇒ Change thename attribute to Engineering and click OK.
- Double-click on the Engineering node. You get an empty workspace, indicating that the subnet contains no objects.
- Open the object palette and make sure it is still set tointernet_toolbox.
- Add the following items to the subnet workspace: 10BaseT LAN,ethernet16 Switch, and a 10BaseT link to connect the LAN with the Switch ⇒ Close the palette.
- Right-click on the 10BaseT LAN node ⇒ Edit Attributes ⇒ Change the name attribute to LAN ⇒ Observe that the Number of Workstations attribute has a value of 10. Click in the Value column for the Application: Supported Profiles attribute, and select Edit. You should get a table in which you should do the following:
- Set the number of rows to 1.
- Set the Profile Name to Engineer. Note: Engineer is one of the "sample" profiles provided within the Profile Configobject.
- Click OK twice.
- The object we just created is equivalent to a 10-workstation star topology LAN. The traffic generated from the users of this LAN resembles that generated by "engineers."
- Rename the ethernet16 Switch to Switch.
- The subnet should look like the shown one.
- Save your project.



Configure All Departments:

- Now you have completed the configuration of the Engineering department subnet. To go back to the main project space, click the**Go to the higher level** button.
- The subnets of the other departments in the company should be similar to the engineering one except for the supported profiles.
- Make three copies of the **Engineering** subnet we just created: Click on the **Engineering** node ⇒ From the **Edit** menu, select **Copy** ⇒From the **Edit** menu, select **Paste** three times, placing the subnet in the workspace after each, to create the new subnets.

3. Rename (right-click on the subnet and select **Set Name**) and arrange the subnets as shown below:



- Double-click the **Research** node ⇒ **Edit** the attributes of its **LAN** ⇒**Edit** the value of the **Application: Supported Profiles** attribute ⇒ Change the value of the **Profile Name** from **Engineer** to **Researcher**⇒ Click **OK** twice ⇒ Go to the higher level by clicking the button.
- Repeat step 4 with the **Sales** node and assign to its **Profile Name** the profile
- **Sales Person**.
- Repeat step 4 with the **E-Commerce** node and assign to its **Profile Name**the profile **E-commerce Customer**.

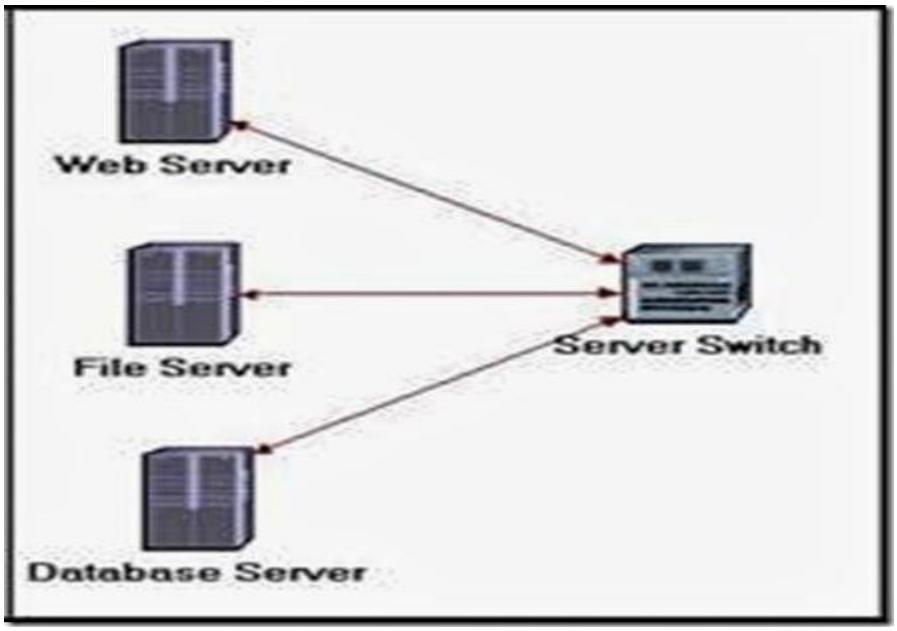
- Save your project.

Configure the Servers:

Now we need to implement a subnet that contains the servers. The servers have to support the applications defined in the profiles we deployed. You can double-check those applications by editing the attributes of our **Profile**node. Inspect each row under the **Applications** hierarchy, which in turn, is under the **Profile Configuration** hierarchy. You will see that we need servers that support the following applications: Web browsing, Email, Telnet, File Transfer, Database, and File Print.

- Open the **Object Palette** and add a new **subnet** ⇒ Rename the new subnet to **Servers** ⇒ Double-click the **Servers** node to enter its workspace.
- From the **Object Palette**, add three **ethernet_servers**, one**ethernet16_switch**, and three **10BaseT** links to connect the servers with the switch.

- Close the **Object Palette**.
- Rename the servers and the switch as follows:



- Right-click on each one of the above servers and **Edit** the value of the

Application: Supported Services attribute.

i. For the *Web Server* add four rows to support the following services: **WebBrowsing (Light HTTP1.1)**, **Web Browsing (Heavy HTTP1.1)**, **Email (Light)**, and **Telnet Session (Light)**.

ii. For the *File Server* add two rows to support the following services: **FileTransfer (Light)** and **File Print (Light)**.

iii. For the *Database Server* add one row to support the following service:

Database Access (Light).

6. Go back to the project space by clicking the **Go to the higher level** button.

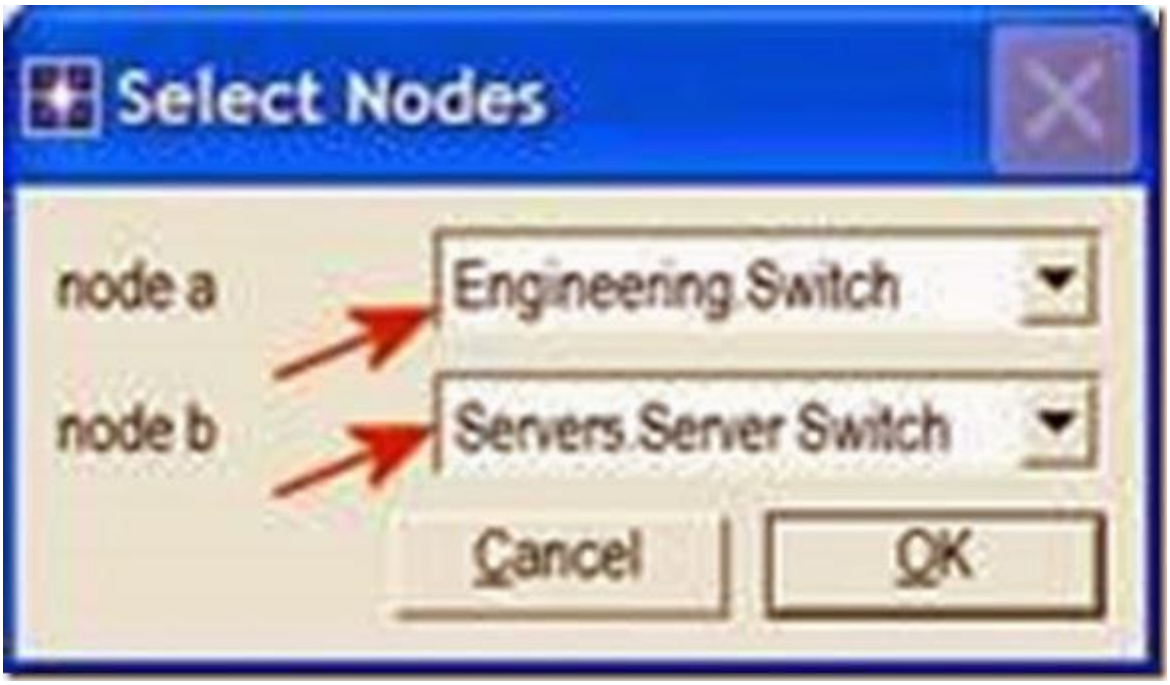
- Save your project.

Connect the Subnets:

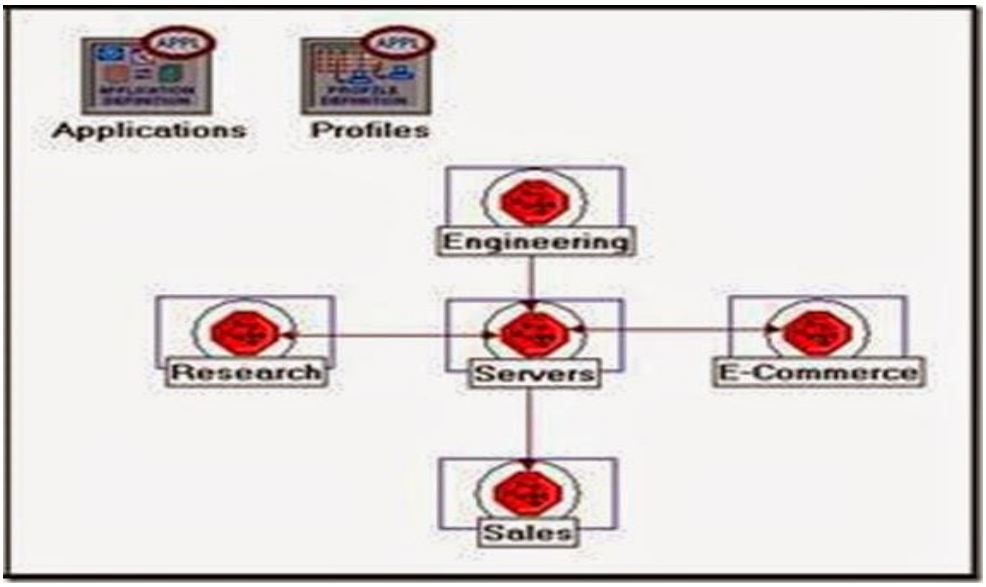
Now all subnets are ready to be connected together.

- **Open** the **Object Palette** and add four **100BaseT** links to connect the subnets of the departments to the **Servers** subnet.

As you create each link, make sure that it is configured to connect the "switches" in both subnets to each other. Do this by choosing them from the drop-down menus as follows:



- Close the **Object Palette**.
- Now your network should resemble the following one:



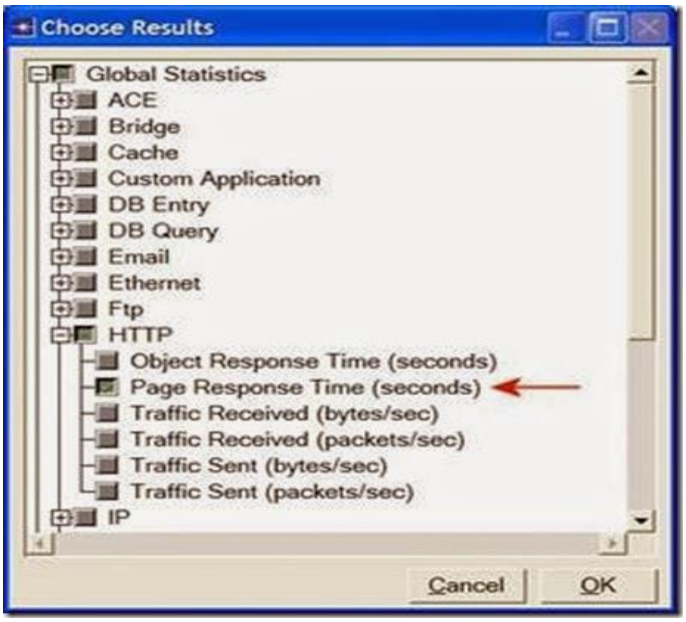
4. Save your project.

Choose the Statistics

To test the performance of our network we will collect one of the many available statistics as follows:

1. Right-click anywhere in the project workspace and select **Choose Individual Statistics** from the pop-up menu.

2. In the *Choose Results* dialog box, choose the following statistic:



Page Response Time is the required time to retrieve the entire page.

3. Click **OK**.

Configure the Simulation
Here we need to configure the duration of the simulation:

1. Click on the **Configure/Run Simulation** button.
2. Set the duration to be **30.0 minutes**.
3. Press **OK**.

Duplicate the Scenario

Link utilization is the percentage of the used link bandwidth.

In the network we just created we assumed that there is no background traffic already in the links. In real networks, the links usually have some existing background traffic. We will create a duplicate of the**SimpleNetwork** scenario but with background utilization in the 100BaseT links.

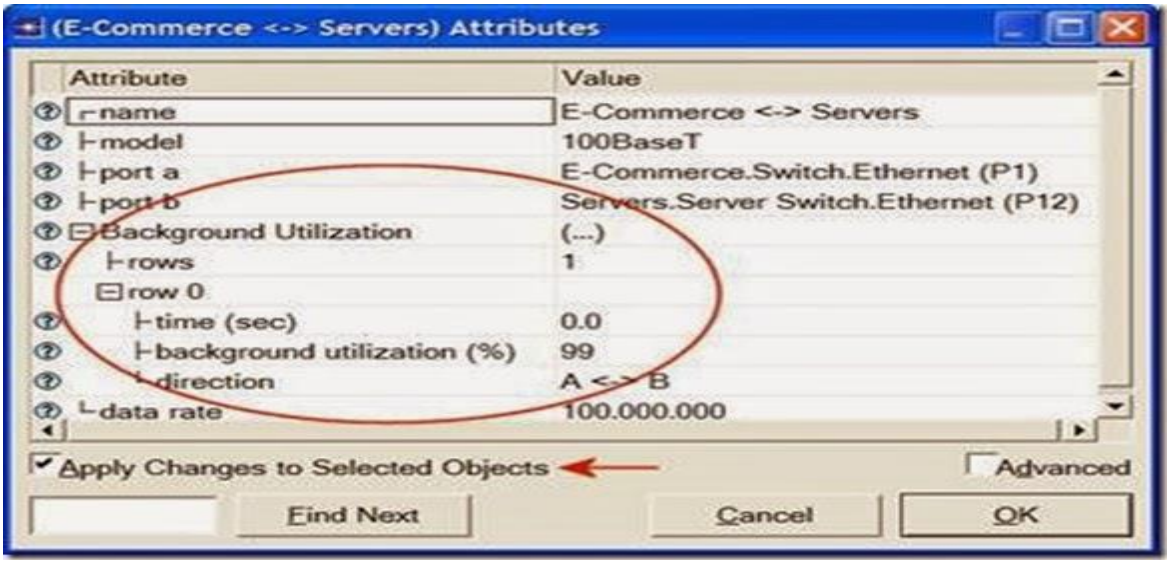
- Select **Duplicate Scenario** from the **Scenarios** menu and give it the name

BusyNetwork ⇒ Click **OK**.

2. Select all the 100BaseT links simultaneously (click on all of them while holding the **Shift** key) ⇒ Right-click on anyone of them ⇒ **Edit Attributes**⇒ Check the

Apply Changes to Selected Objects check box.

3. Expand the hierarchy of the **Background Utilization** attribute ⇒ Expand the **row 0** hierarchy ⇒ Assign **99** to the **background utilization (%)** as shown below.



4. Click **OK**.

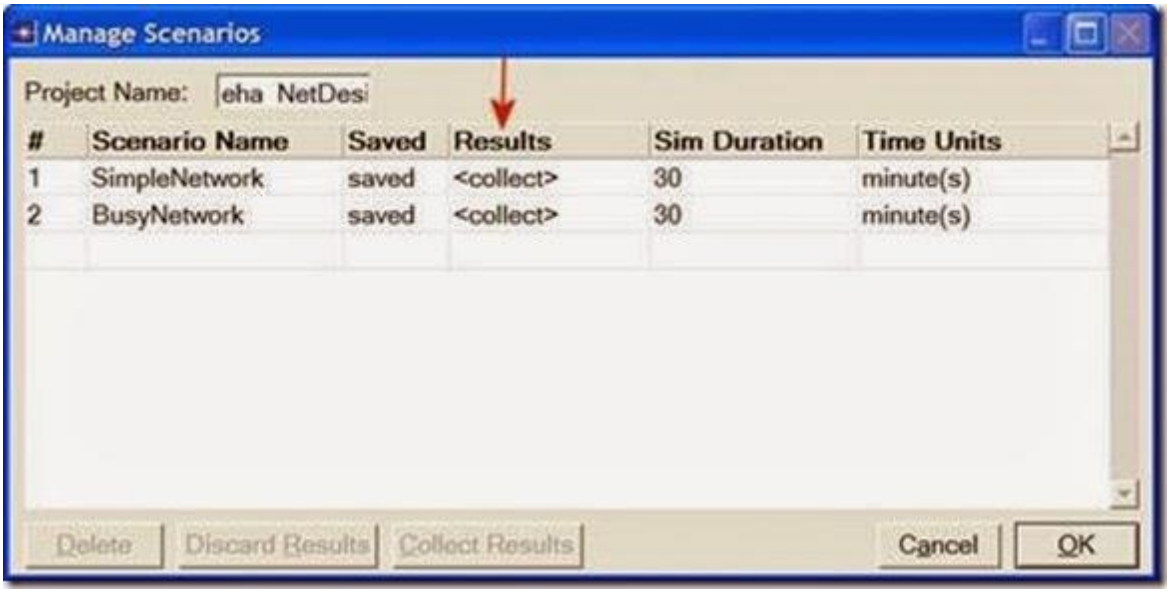
5. Save your project.

Run the Simulation

To run the simulation for both scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Change the values under the **Results** column to **<collect>** (or**<recollect>**) for both scenarios. Compare to the following figure.



3. Click **OK** to run the two simulations. Depending on the speed of your processor, this may take several seconds to complete.

4. After the two simulation runs complete (one for each scenario), click**Close**.

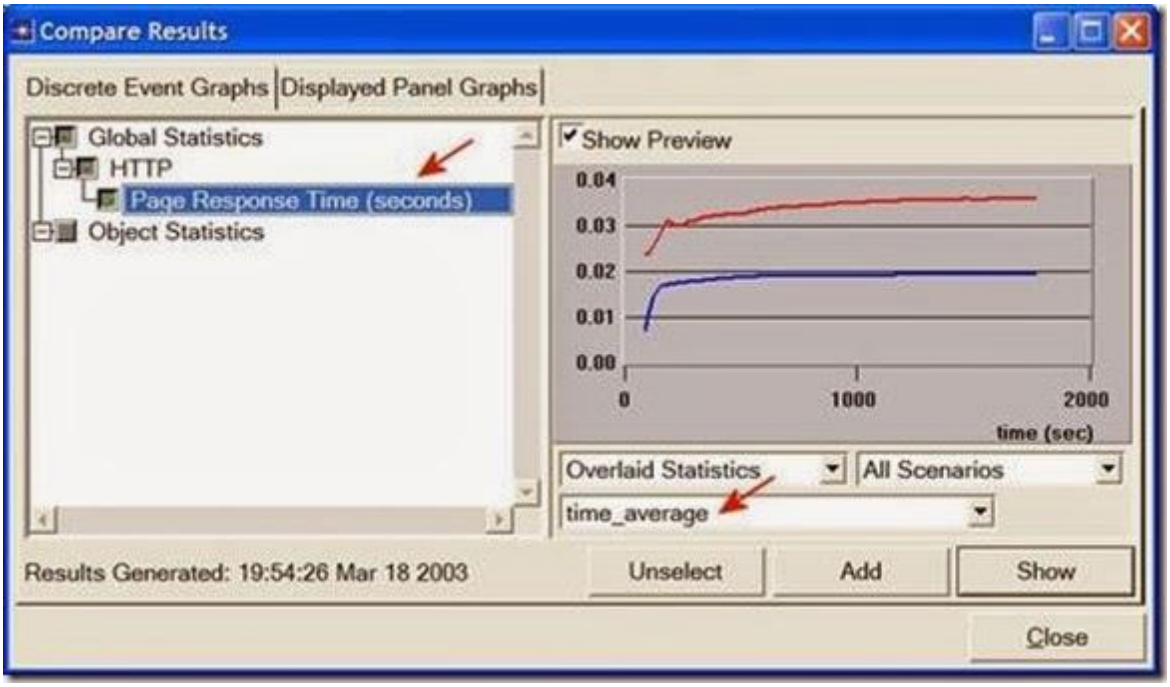
5. Save your project.

View the Results

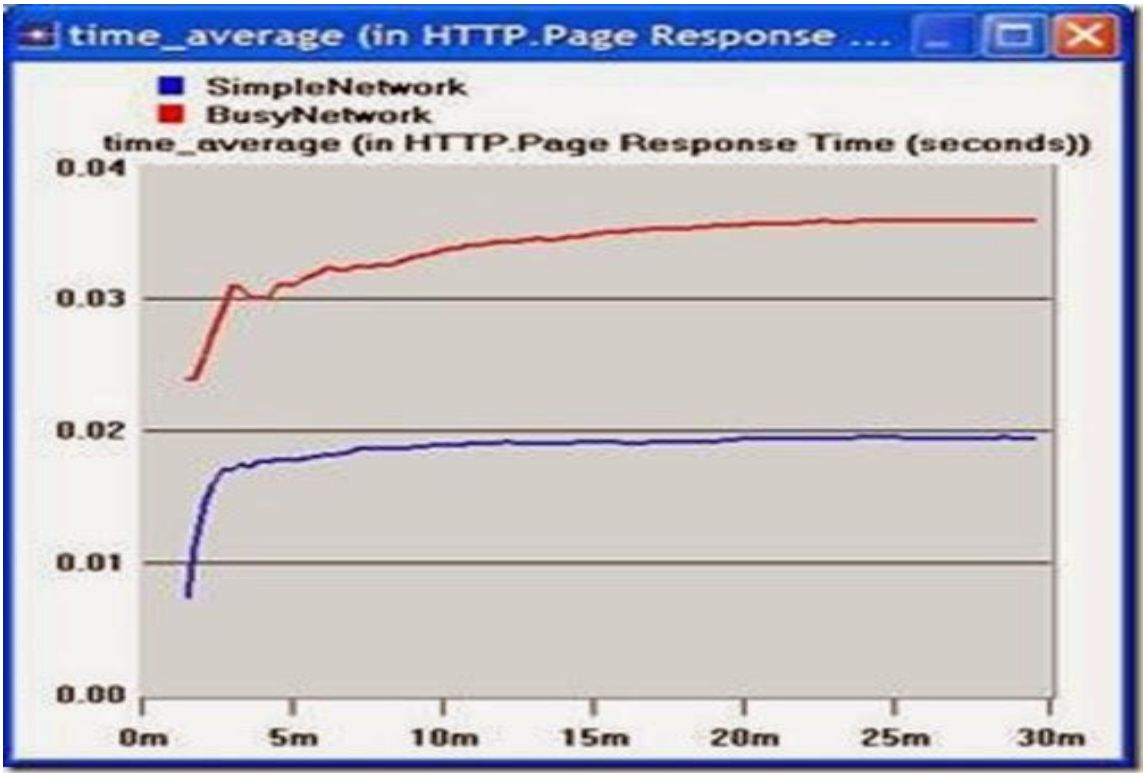
To view and analyze the results:

1. Select **Compare Results** from the **Results** menu.

2. Change the drop-down menu in the lower-right part of the *Compare Results* dialog box from **As Is** to **time_average** as shown.



3. Select the **Page Response Time (seconds)** statistic and click **Show**. The resulting graph should resemble the one below. (*Note:* Results may vary slightly due to different node placement.)



Questions

1) Analyze the result we obtained regarding the HTTP page response time. Collect four other statistics, of your choice, and rerun the simulation of the*Simple* and the *Busy* network scenarios. Get the graphs that compare the collected statistics. Comment on these results.

2) In the **BusyNetwork** scenario, study the **utilization%** of the CPUs in the servers (Right-click on each server and select **Choose Individual Statistics⇒ CPU ⇒ Utilization**).

3) Create a new scenario as a duplicate of the **BusyNetwork** scenario. Name the new scenario **Q3_OneServer**. Replace the three servers with only one server that supports all required services. Study the **utilization%**of that server's CPU. Compare this utilization with the three CPU utilizations you obtained in the previous question.

4) Create a new scenario as a duplicate of the **BusyNetwork** scenario. Name the new scenario **Q4_FasterNetwork.** In the **Q4_FasterNetwork**scenario, replace all **100BaseT** links in the network with **10Gbps Ethernet**links and replace all **10BaseT** links with **100BaseT** links. Study how increasing the bandwidth of the links affects the performance of the network in the new scenario (e.g., compare the HTTP page response time in the new scenario with that of the **BusyNetwork**).

Lab 25 – RIP on IT Guru

Objectives

The objective of this lab is to configure and analyze the performance of the Routing Information Protocol (RIP) model.

Lab Tasks

Task: Apply RIP on IT Guru.

A router in the network needs to be able to look at a packet's destination address and then determine which of the output ports is the best choice to get the packet to that address. The router makes this decision by consulting a forwarding table. The fundamental problem of routing is: How do routers acquire the information in their forwarding tables?

Routing algorithms are required to build the routing tables and hence forwarding tables. The basic problem of routing is to find the lowest-cost path between any two nodes, where the cost of a path equals the sum of the costs of all the edges that make up the path. Routing is achieved in most practical networks by running routing protocols among the nodes. The protocols provide a distributed, dynamic way to solve the problem of finding the lowest-cost path in the presence of link and node failures and changing edge costs.

One of the main classes of routing algorithms is the distance-vector algorithm. Each node constructs a vector containing the distances (costs) to all other nodes and distributes that vector to its immediate neighbors. RIP is the canonical example of a routing protocol built on the distance-vector algorithm. Routers running RIP send their advertisements regularly (e.g., every 30 seconds). A router also sends an update message whenever a triggered update from another router causes it to change its routing table.

In this lab you will set up a network that utilizes RIP as its routing protocol. You will analyze the routing tables generated in the routers, and you will observe how RIP is affected by link failures.

Procedure

Create a New Project

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the **File** menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_RIP**, and the scenario **NO_Failure** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that **Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Select **Campus** from the *Network Scale* list ⇒ Click **Next** three times ⇒ Click **OK**.

Create and Configure the Network

The **ethernet4_slip8_ gtwy** node model represents an IP-based gateway supporting four Ethernet hub interfaces and eight serial line interfaces. IP packets arriving on any interface are routed to the appropriate output interface based on their destination IP address. The Routing Information Protocol (RIP) or the Open Shortest Path First (OSPF) protocol may be used to dynamically and automatically create the gateway's routing tables and select routes in an adaptive manner.

Initialize the Network:

1. The *Object Palette* dialog box should now be on top of your project workspace. If it is not there, open it by clicking . Make sure that the **internet_toolbox** is selected from the pull-down menu on the object palette.

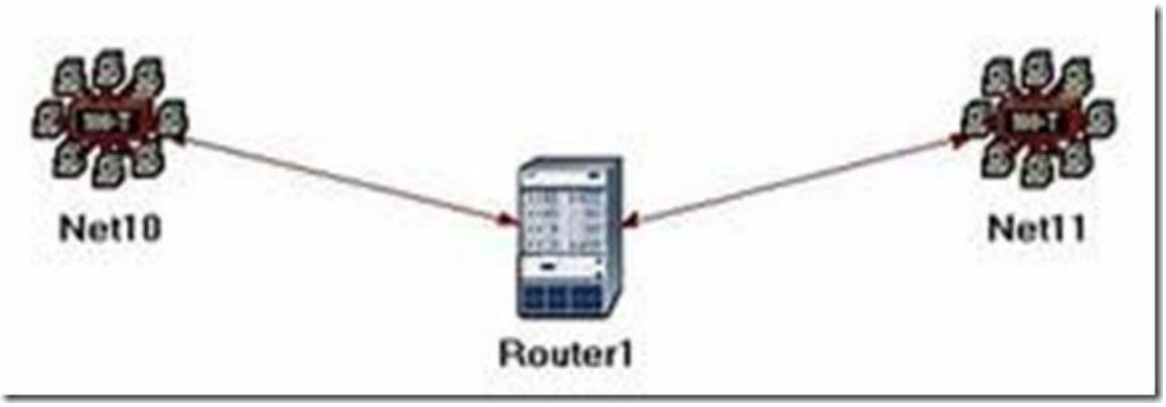
1. Add to the project workspace the following objects from the palette: one **ethernet4_slip8_gtwy** router and two **100BaseT_LAN** objects.

a. To add an object from a palette, click its icon in the object palette ⇒ Move your mouse to the workspace ⇒ Click to place the object ⇒ Right-click to stop creating objects of that type.

1. Use bidirectional **100BaseT** links to connect the objects you just added as in the following figure. Also, rename the objects as shown (right-click on the node ⇒ **Set Name**).

1. Close the *Object Palette* dialog box.

1. Save your project.



Configure the Router:

1. Right-click on **Router1** ⇒ **Edit Attributes** ⇒ Expand the **IP Routing Parameters** hierarchy and set the following:

i. **Routing Table Export** = **Once at End of Simulation**. This asks the router to export its routing table at the end of the simulation to the *simulation log*.

1. Click **OK** and then save your project.

Add the Remaining LANs:

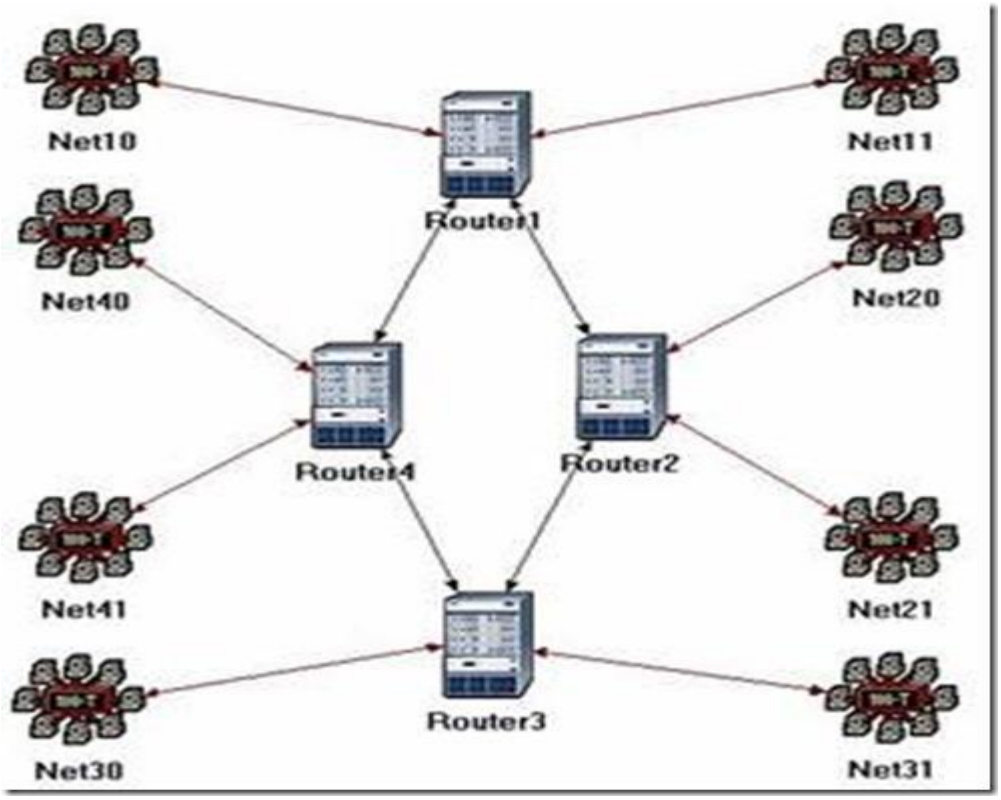
The **PPP_DS3** link has a data rate of 44.736 Mbps.

1. Highlight or select simultaneously (using shift and left-click) all five objects that you currently have in the project workspace (one router, two LANs, and two links). You can click-and-drag a box around the objects to do this.

2. Press **Ctrl+C** to copy the selected objects and then press **Ctrl+V** to paste them.

3. Repeat step 2 three times to generate three new copies of the objects and arrange them in a way similar to the following figure. Rename all objects as shown.

4. Connect routers, as shown, using **PPP_DS3** links.



Choose the Statistics

RIP traffic is the total amount of RIP update traffic (in bits) sent/received per second by all the nodes using RIP as the routing protocol in the IP interfaces in the node.

Total Number of Updates is the number of times the routing table at this node gets updated (e.g., due to a new route addition, an existing route deletion, and/or a next hop update).

To test the performance of the RIP protocol, we will collect the following statistics:

1. Right-click anywhere in the project workspace and select **Choose Individual Statistics** from the pop-up menu.

1. In the *Choose Results* dialog box, check the following statistics:

    1. **Global Statistics ⇒ RIP ⇒ Traffic Sent (bits/sec).**

    1. **Global Statistics ⇒ RIP ⇒ Traffic Received (bits/sec).**

    1. **Nodes Statistics ⇒ Route Table ⇒ Total Number of Updates**.

1. Click **OK** and then save your project.

Configure the Simulation

Auto Addressed means that all IP interfaces are assigned IP addresses automatically during simulation. The class of address (e.g., A, B, or C) is determined based on the number of hosts in the designed network. Subnet masks assigned to these interfaces are the default subnet masks for that class.

Export causes the auto-assigned IP interface to be exported to a file (name of the file is <net_name>-ip_addresses.gdf and gets saved in the primary model directory).

Here we need to configure some of the simulation parameters:

1. Click on and the *Configure Simulation* window should appear.

2. Set the duration to be **10.0 minutes**.

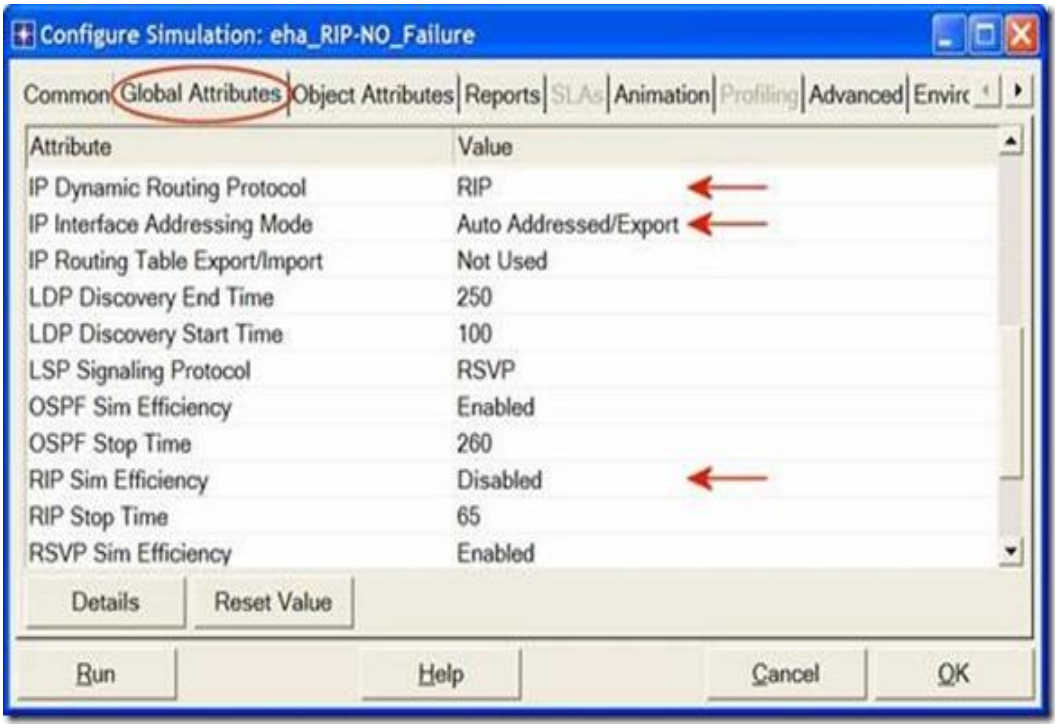
3. Click on the **Global Attributes** tab and change the following attributes:

a. **IP Dynamic Routing Protocol** = **RIP**. This sets the RIP protocol to be the routing protocol of all routers in the network.

b. **IP Interface Addressing Mode** = **Auto Addressed/Export**.

c. **RIP Sim Efficiency** = **Disabled**. If this attribute is enabled, RIP will stop after the "RIP Stop Time." But we need the RIP to keep updating the routing table in case there is any change in the network (as we will see in the second scenario).

4. Click **OK** and then save the project.



Duplicate the Scenario

In the network we just created, the routers will build their routing tables, and then they will not need to update them further because we didn't simulate any node or link failures. In this scenario we will simulate failures so that we can compare the behavior of the routers in both cases.

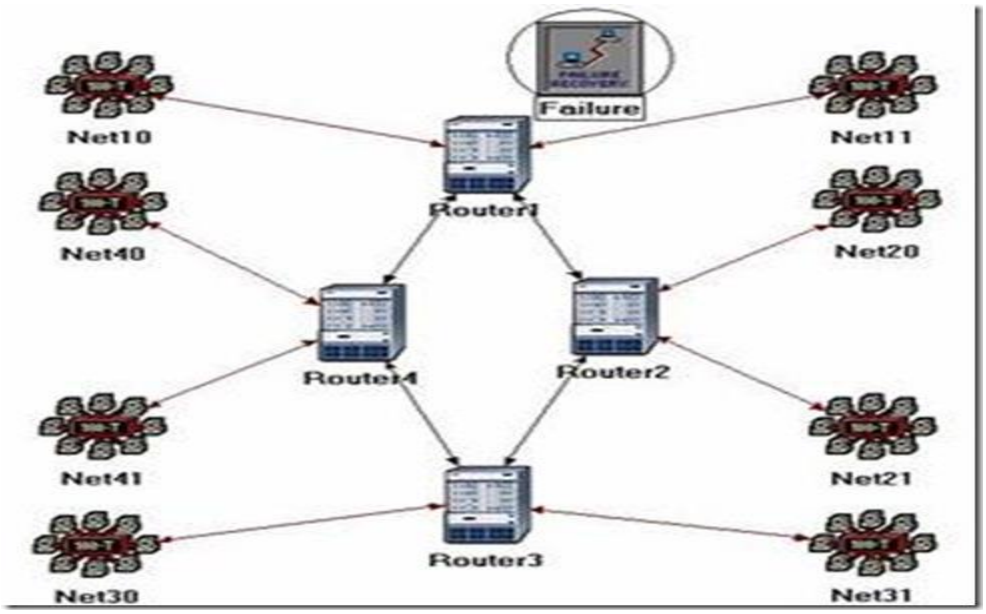
1. Select **Duplicate Scenario** from the **Scenarios** menu and name it **Failure** ⇒

Click **OK**.

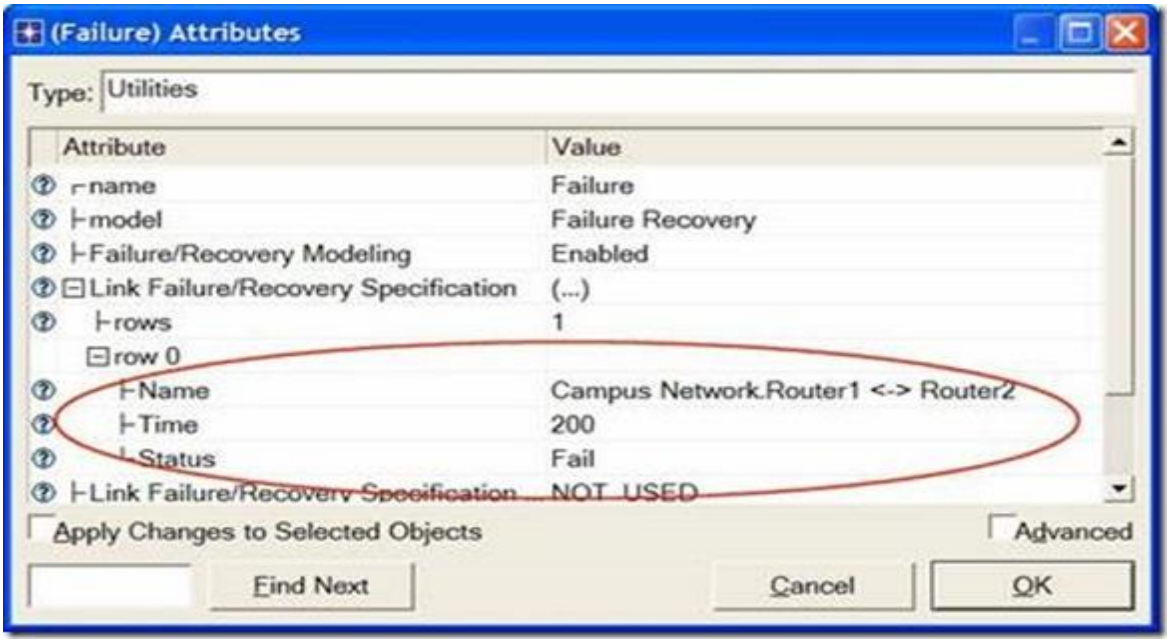
2. Open *Object Palette* by clicking . Select the **Utilities** palette from the drop-down menu.

4

3. Add a **Failure Recovery** object to your workspace and name it **Failure** as shown ⇒ Close the *Object Palette* dialog box.



4. Right-click on the **Failure** object ⇒ **Edit Attributes** ⇒ Expand the **Link Failure/Recovery Specification** hierarchy ⇒ Set **rows** to 1 ⇒ Set the attributes of the added row, **row 0**, as follows:



This will "fail" the link between **Router1** and **Router2** 200 seconds into the simulation.

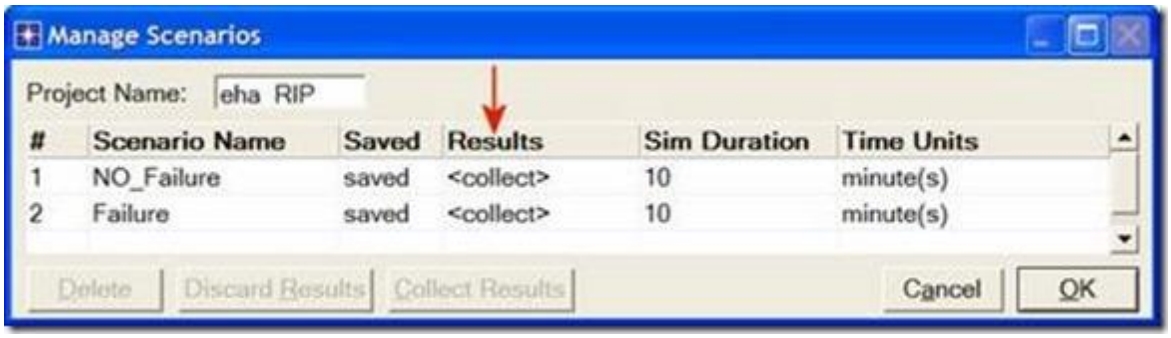
5. Click **OK** and then save the project.

Run the Simulation

To run the simulation for both scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Change the values under the **Results** column to **<collect>** (or **<recollect>**) for both scenarios. Compare to the following figure.



3. Click **OK** to run the two simulations. Depending on the speed of your processor, this may take several seconds to complete.

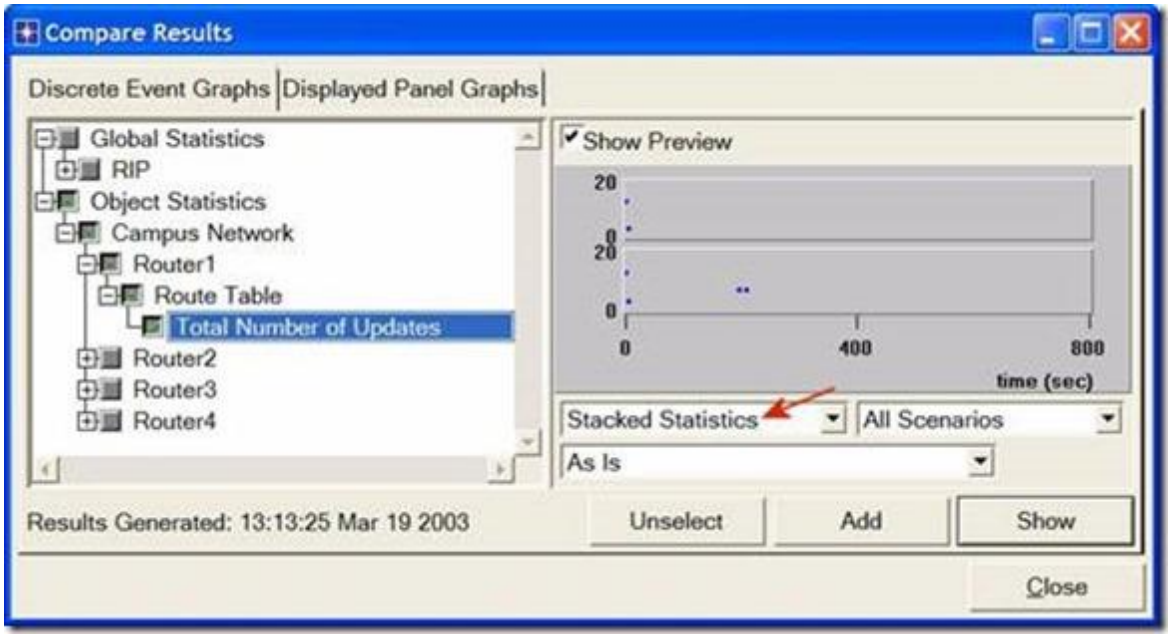
4. After the two simulation runs complete, one for each scenario, click **Close** ⇒ Save your project.

View the Results

Compare the Number of Updates:

1. Select **Compare Results** from the **Results** menu.

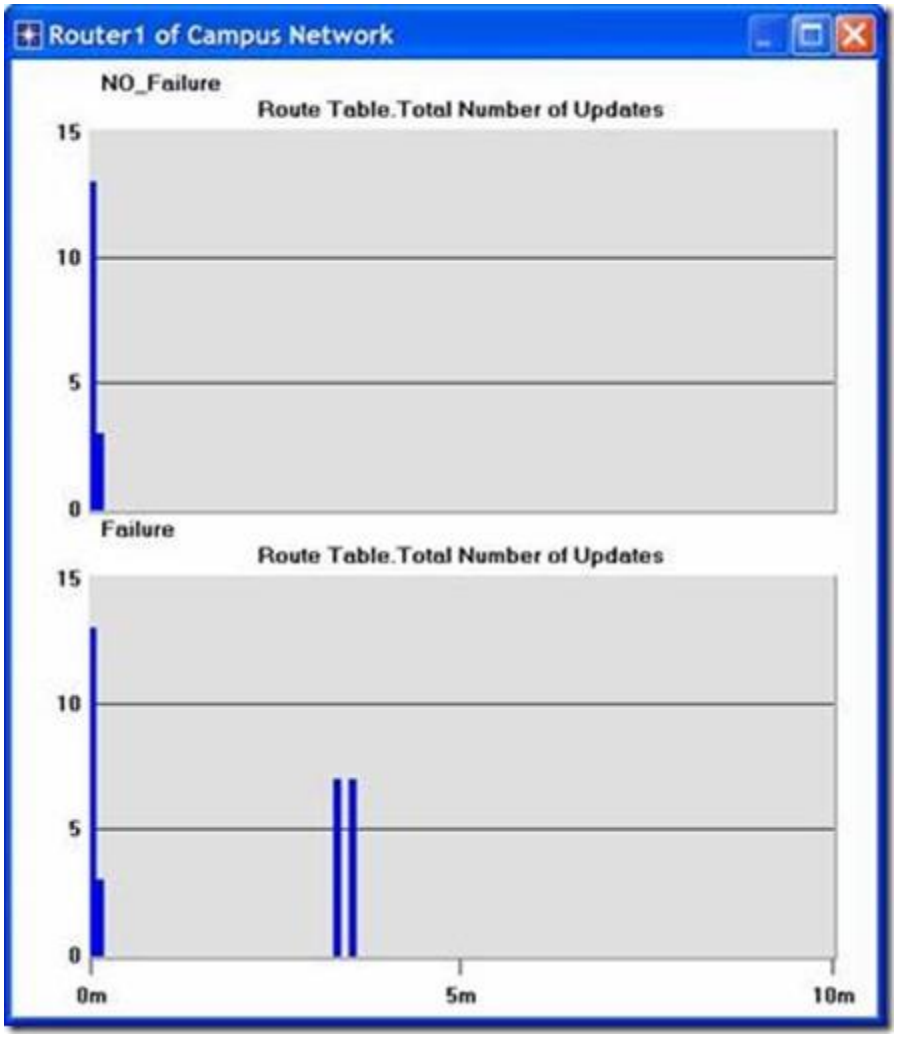
2. Change the drop-down menu in the right-lower part of the *Compare Results* dialog box to **Stacked Statistics** as shown.



3. Select the **Total Number of Updates** statistic for **Router1** and click **Show**.

4. You should get two graphs, one for each scenario. Right-click on each graph and select **Draw Style ⇒ Bar**.

5. The resulting graphs should resemble the following (you can zoom in on the graphs by clicking-and-dragging a box over the region of interest):

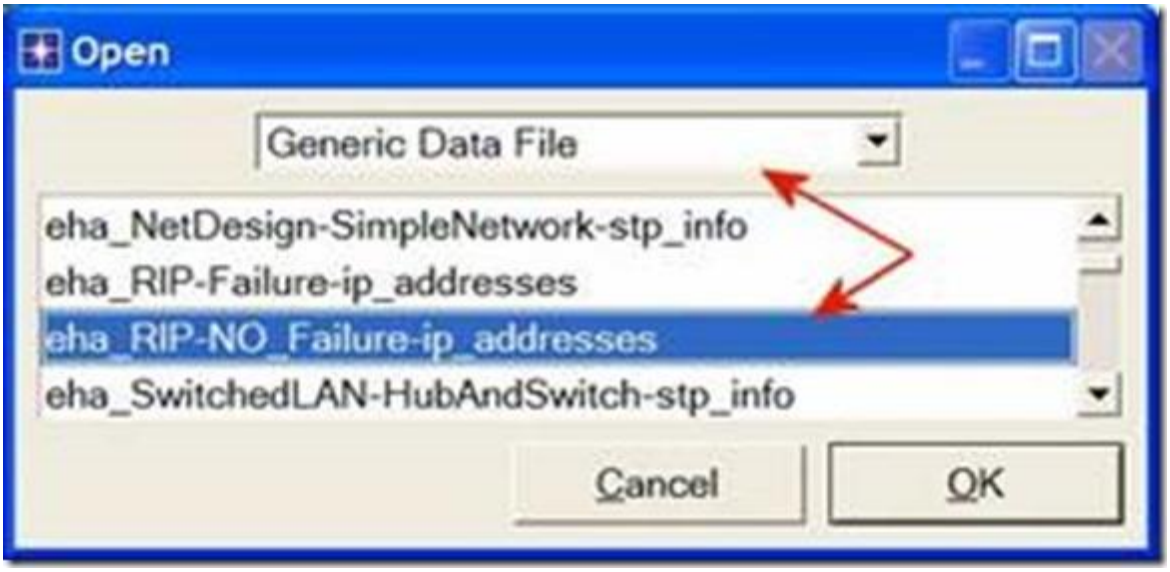


Obtain the IP Addresses of the Interface:

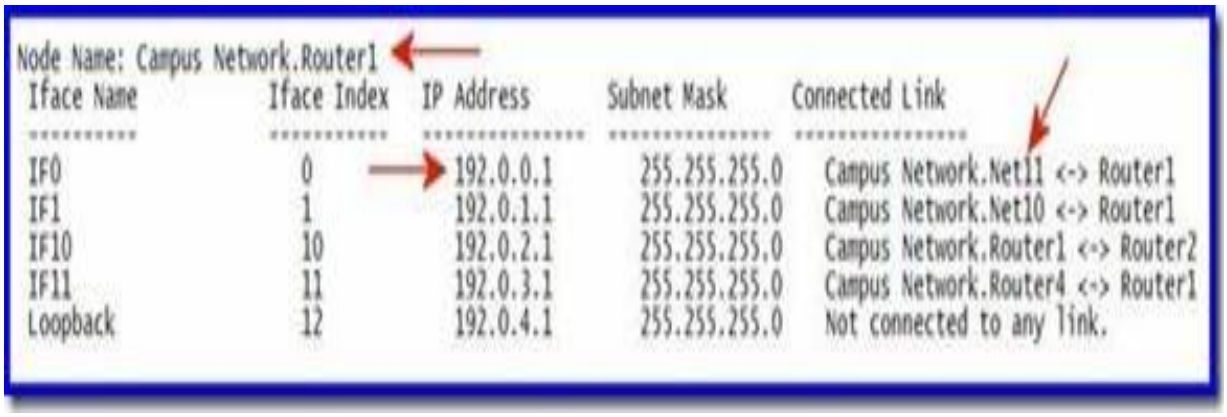
Before checking the contents of the routing tables, we need to determine the IP address information for all interfaces in the current network. Recall that these IP addresses are assigned automatically during simulation, and we set the global attribute **IP Interface Addressing Mode** to export this information to a file.

1. From the **File** menu choose **Model Files ⇒ Refresh Model Directories**. This causes OPNET IT Guru to search the model directories and update its list of files.

1. From the **File** menu choose **Open** ⇒ From the drop-down menu choose **Generic Data File** ⇒ Select the <**your initials**>**_RIP-NO_Failure-ip_addresses** file (the other file created from the **Failure** scenario should contain the same information) ⇒ Click **OK**.



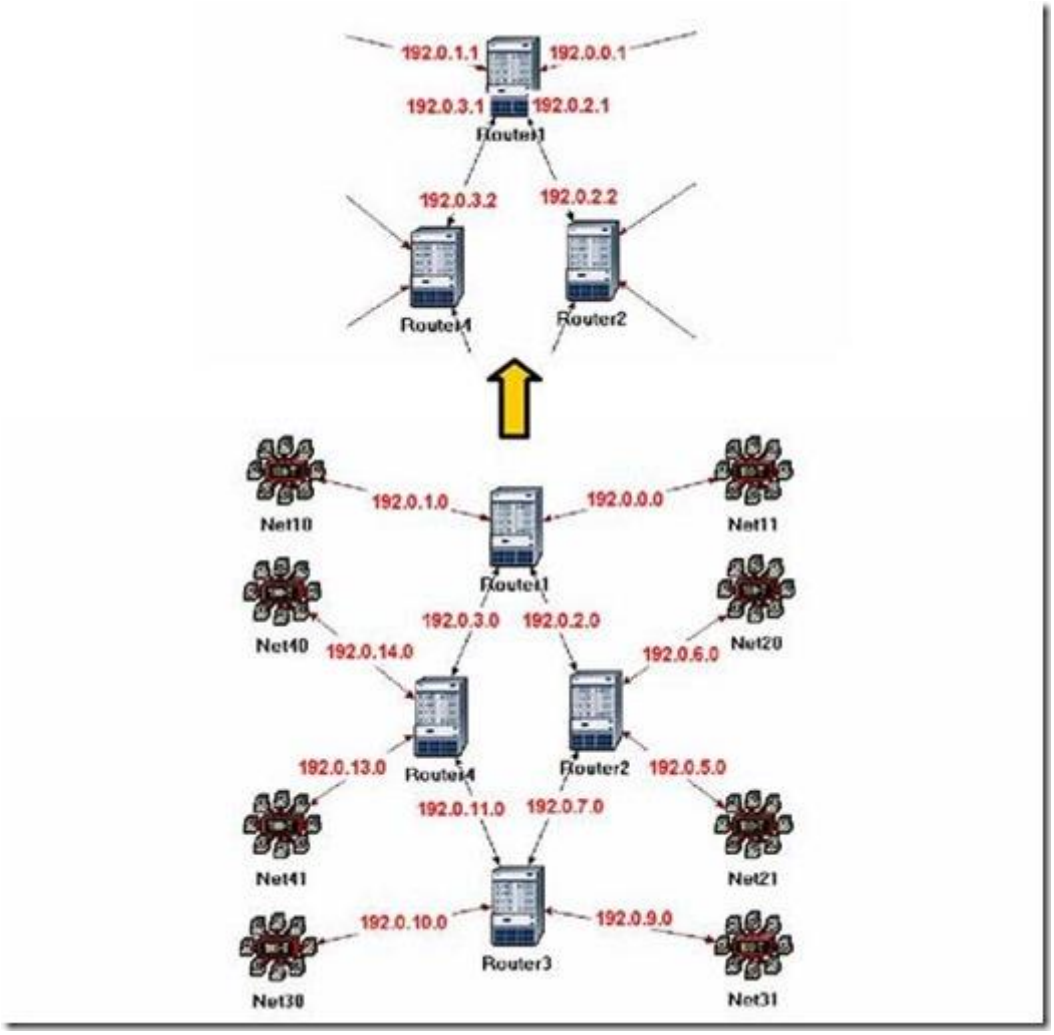
1. The following is a part of the **gdf** file content. It shows the IP addresses assigned to the interfaces of **Router1** in our network. For example the interface of **Router1** that is connected to **Net11** has the IP address **192.0.0.1** (*Note:* Your result may vary due to different nodes placement.) The **Subnet Mask** associated with that interface indicates that the address of the subnetwork, to which the interface is connected, is **192.0.0.0** (i.e., the logical AND of the interface IP address and the subnet mask).



Node Name: Campus Network.Router1

Iface Name	Iface Index	IP Address	Subnet Mask	Connected Link
IF0	0	192.0.0.1	255.255.255.0	Campus Network.Net11 <-> Router1
IF1	1	192.0.1.1	255.255.255.0	Campus Network.Net10 <-> Router1
IF10	10	192.0.2.1	255.255.255.0	Campus Network.Router1 <-> Router2
IF11	11	192.0.3.1	255.255.255.0	Campus Network.Router4 <-> Router1
Loopback	12	192.0.4.1	255.255.255.0	Not connected to any link.

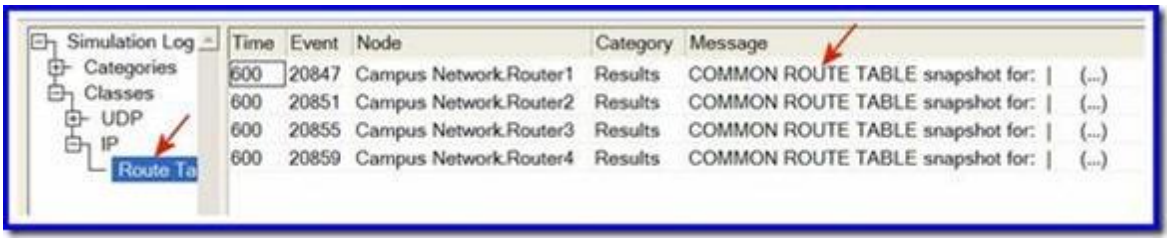
1. Print out the layout of the network you implemented in this lab. On this layout, from the information included in the **gdf** file, write down the IP addresses associated with **Router1** as well as the addresses assigned to each subnetwork as shown in the following two figures (*Note:* Your IP addresses may vary due to different nodes placement.)



Compare the Routing Tables Content:

1. To check the content of the routing tables in **Router1** for both scenarios:

i. Go to the **Results** menu ⇒ **Open Simulation Log** ⇒ Expand the hierarchy on the left as shown below ⇒ Click on the field **COMMON ROUTE TABLE.**



2. Carry out the previous step for both scenarios. The following are partial contents of Router1's routing table for both scenarios (*Note:* Your results may vary due to different nodes placement):

Routing table of Router1 (**NO_Failure** scenario):

Router name: Campus Network.Router1
at time: 600.00 seconds

ROUTE TABLE contents:

Dest. Address	Subnet Mask	Next Hop	Interface Name	Metric	Protocol
192.0.0.0	255.255.255.0	192.0.0.1	IF0	0	Direct
192.0.1.0	255.255.255.0	192.0.1.1	IF1	0	Direct
192.0.2.0	255.255.255.0	192.0.2.1	IF10	0	Direct
192.0.3.0	255.255.255.0	192.0.3.1	IF11	0	Direct
192.0.4.0	255.255.255.0	192.0.4.1	Loopback	0	Direct
192.0.5.0	255.255.255.0	192.0.2.2	IF10	1	RIP
192.0.6.0	255.255.255.0	192.0.2.2	IF10	1	RIP
192.0.7.0	255.255.255.0	192.0.2.2	IF10	1	RIP
192.0.8.0	255.255.255.0	192.0.2.2	IF10	1	RIP
192.0.11.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.13.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.14.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.15.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.9.0	255.255.255.0	192.0.2.2	IF10	2	RIP
192.0.10.0	255.255.255.0	192.0.2.2	IF10	2	RIP
192.0.12.0	255.255.255.0	192.0.2.2	IF10	2	RIP

Loopback interface allows a client and a server on the same host to communicate with each other using

TCP/IP.

Routing table of Router1 (**Failure** scenario):

Router name: Campus Network.Router1
at time: 600.00 seconds

ROUTE TABLE contents:

Dest. Address	Subnet Mask	Next Hop	Interface Name	Metric	Protocol
192.0.0.0	255.255.255.0	192.0.0.1	IF0	0	Direct
192.0.1.0	255.255.255.0	192.0.1.1	IF1	0	Direct
192.0.2.0	255.255.255.0	192.0.2.1	IF10	0	Direct
192.0.3.0	255.255.255.0	192.0.3.1	IF11	0	Direct
192.0.4.0	255.255.255.0	192.0.4.1	Loopback	0	Direct
192.0.11.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.13.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.14.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.15.0	255.255.255.0	192.0.3.2	IF11	1	RIP
192.0.5.0	255.255.255.0	192.0.3.2	IF11	3	RIP
192.0.6.0	255.255.255.0	192.0.3.2	IF11	3	RIP
192.0.7.0	255.255.255.0	192.0.3.2	IF11	2	RIP
192.0.8.0	255.255.255.0	192.0.3.2	IF11	3	RIP
192.0.9.0	255.255.255.0	192.0.3.2	IF11	2	RIP
192.0.10.0	255.255.255.0	192.0.3.2	IF11	2	RIP
192.0.12.0	255.255.255.0	192.0.3.2	IF11	2	RIP

Lab 26 – OSPF on IT Guru

Objectives

The objective of this lab is to apply OSPF on IT Guru.

Lab Tasks

Task: Apply OSPF on IT Guru.

A Routing Protocol Based on the Link-State Algorithm

Objective

The objective of this lab is to configure and analyze the performance of the Open Shortest Path First (OSPF) routing protocol.

Overview

In Lab 6 we discussed RIP, which is the canonical example of a routing protocol built on the distance-vector algorithm. Each node constructs a vector containing the distances (costs) to all other nodes and distributes that vector to its immediate neighbors. Link-state routing is the second major class of intra-domain routing protocol. The basic idea behind link-state protocols is very simple: Every node knows how to reach its directly connected neighbors, and if we make sure that the totality of this knowledge is disseminated to every node, then every node will have enough knowledge of the network to build a complete map of the network.

Once a given node has a complete map for the topology of the network, it is able to decide the best route to each destination. Calculating those routes is based on a well-known algorithm from graph theory—Dijkstra's shortest-path algorithm.

OSPF introduces another layer of hierarchy into routing by allowing a domain to be partitioned into areas. This means that a router within a domain does not necessarily need to know how to reach every network within that domain—it may be sufficient for it to know how to get to the right area. Thus, there is a reduction in the amount of information that must be transmitted to and stored in each node. In addition, OSPF allows multiple routes to the same destination to be assigned the same cost and will cause traffic to be distributed evenly over those routers.

In this lab, you will set up a network that utilizes OSPF as its routing protocol. You will analyze the routing tables generated in the routers and will observe how the resulting routes are affected by assigning areas and enabling load balancing.

Procedure

Create a New Project

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New** from the **File**menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_OSPF**, and the scenario **No_Areas** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that**Create Empty Scenario** is selected ⇒ Click **Next** ⇒ Select **Campus** from the *Network Scale* list ⇒ Click **Next** three times ⇒ Click **OK**.

Create and Configure the Network

The **slip8_gtwy** node model represents an IP-based gateway supporting up to eight serial line interfaces at a selectable data rate. The RIP or OSPF protocols may be used to automatically and dynamically create the gateway's routing tables and select routes in an adaptive manner.

The **PPP_DS3** link has a data rate of 44.736 Mbps.

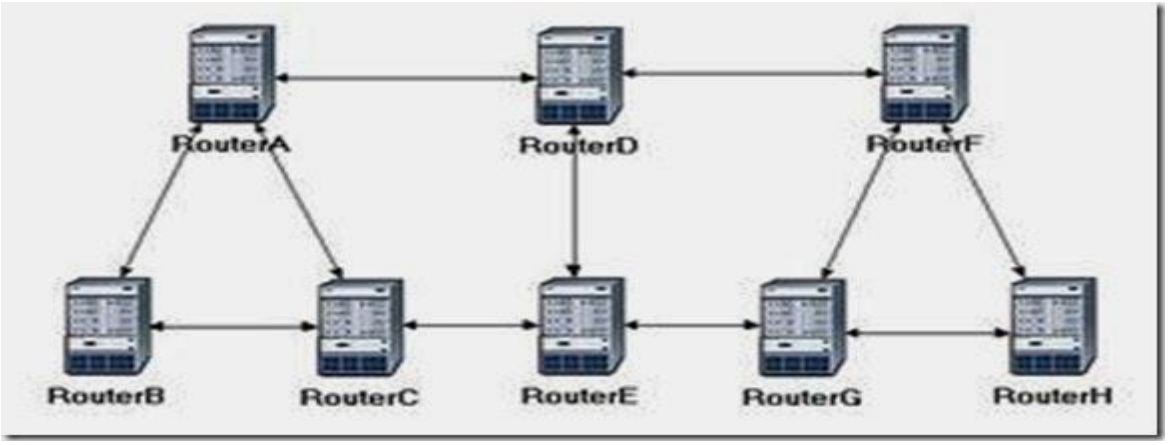
Initialize the Network:

1. The *Object Palette* dialog box should now be on top of your project workspace. If it is not there, open it by clicking . Select the**routers** item from the pull-down menu on the object palette.

   a. Add to the project workspace eight routers of type **slip8_gtwy**. To add an object from a palette, click its icon in the object palette ⇒ Move your mouse to the workspace and click to place the object ⇒ You can keep on left-clicking to create additional objects. Right-click when you are finished placing the last object.

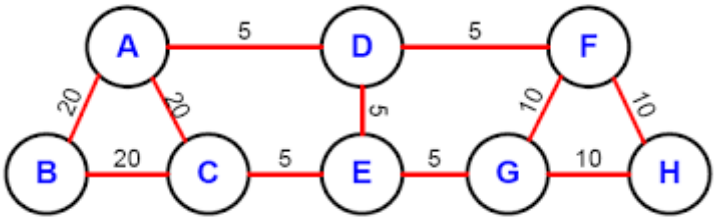
1. Switch the palette configuration so it contains the **internet_toolbox**. Use bidirectional **PPP_DS3** links to connect the routers. Rename the routers as shown below.

1. Close the *Object Palette* and then save your project.



Configure the Link Costs:

1. We need to assign link costs to match the following graph:



2. Like many popular commercial routers, OPNET router models support a parameter called a *reference bandwidth* to calculate the actual cost, as follows:

Cost = (Reference bandwidth) / (Link bandwidth)

where the default value of the *reference bandwidth* is 1,000,000 Kbps.

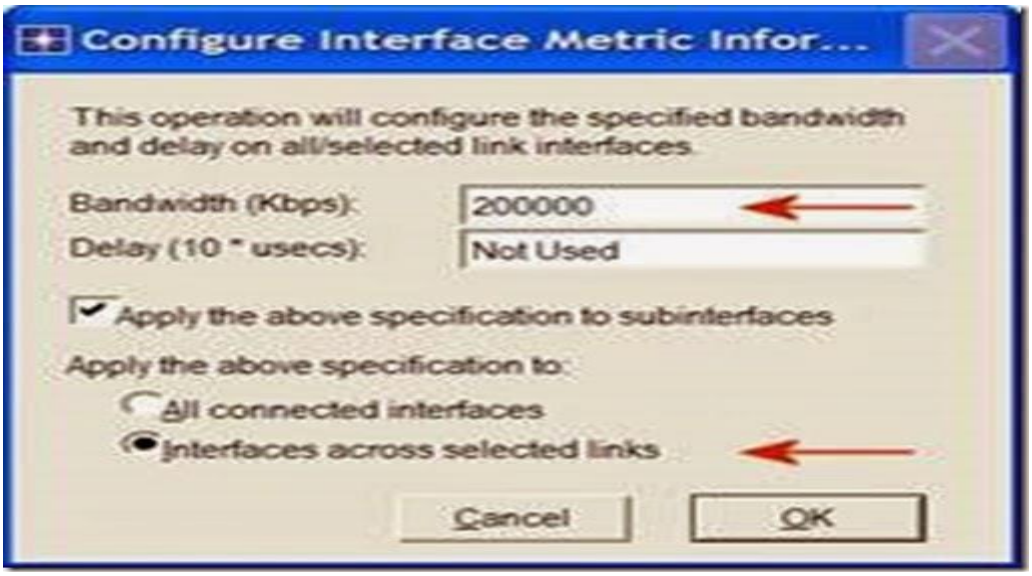
3. For example, to assign a cost of 5 to a link, assign a bandwidth of 200,000 Kbps to that link. Note that this is not the actual bandwidth of the link in the sense of transmission speed, but merely a parameter used to configure link costs.

1. To assign the costs to the links of our network, do the following:

i. Select all links in your network that correspond to the links with a cost of 5 in the above graph by shift-clicking on them.

ii. Select the **Protocols** menu ⇒ **IP** ⇒ **Routing** ⇒ **Configure Interface Metric Information**.

iii. Assign **200000** to the **Bandwidth (Kbps)** field ⇒ Check the **Interfacesacross selected links** radio button, as shown ⇒ Click **OK**.



1. Repeat step 4 for all links with a cost of 10 but assign 100,000 Kbps to the

   Bandwidth (Kbps) field.

1. Repeat step 4 for all links with a cost of 20 but assign 50,000 Kbps to the

   Bandwidth (Kbps) field.

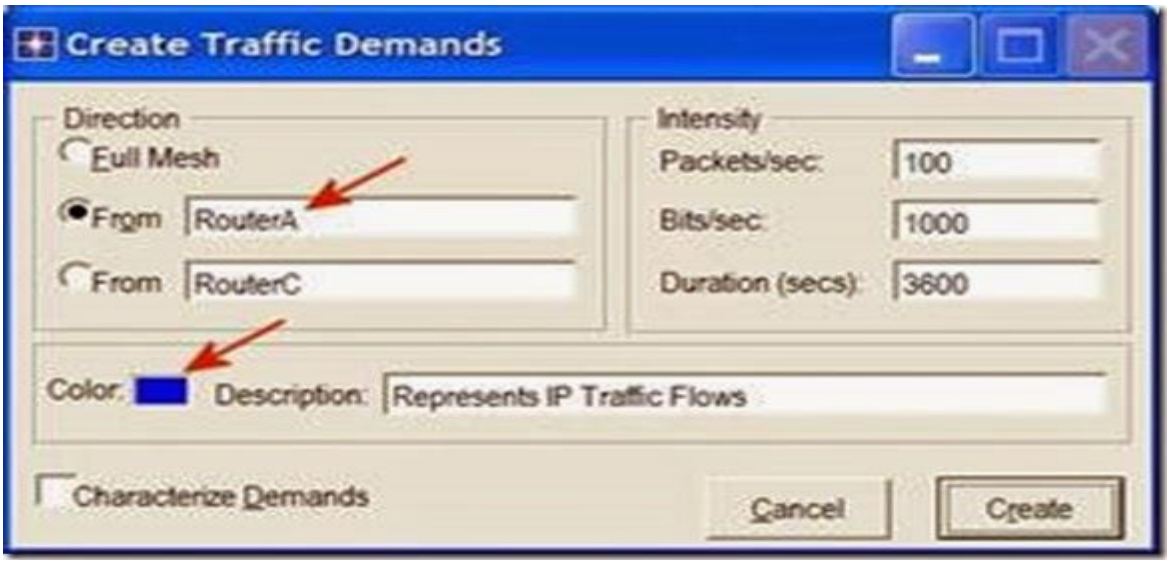
1. Save your project.

   Configure the Traffic Demands:

1. Select both **RouterA** and **RouterC** by shift-clicking on them.

   i. Select the **Protocols** menu ⇒ **IP** ⇒ **Demands** ⇒ **Create Traffic Demands**

   ⇒ Check the **From RouterA** radio button as shown ⇒ Keep the color as blue ⇒ Click **Create**. Now you should see a blue-dotted line representing the traffic demand between **RouterA** and **RouterC**.



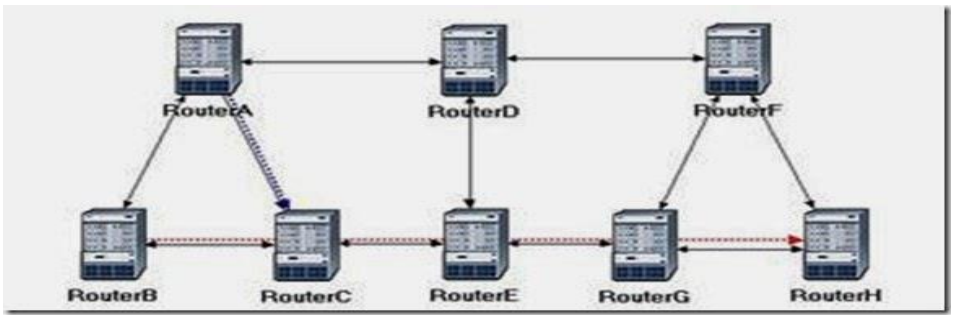
   2. Select both **RouterB** and **RouterH** by shift-clicking on them.

   i. Select the **Protocols** menu ⇒ **IP** ⇒ **Demands** ⇒ **Create Traffic Demands**

   ⇒ Check the **From RouterB** radio button ⇒ Change the color to **red** ⇒ Click

   OK ⇒ Click **Create**.

   Now you can see the lines representing the traffic demands as shown.



3. To hide these lines: Select the **View** menu ⇒ Select **Demand Objects** ⇒ Select

Hide All.

1. Save your project.

Configure the Routing Protocol and Addresses:

Auto-Assign IP Addresses assigns a unique IP address to connected IP interfaces whose IP address is currently set to auto-assigned. It does not change the value of manually set IP addresses.

1. Select the **Protocols** menu ⇒ **IP** ⇒ **Routing** ⇒ **Configure Routing Protocols**.

2. Check the **OSPF** check box ⇒ Uncheck the **RIP** check box ⇒ Uncheck the

Visualize Routing Domains check box, as shown:

3. Click **OK**.

4.Select **RouterA** and **RouterB** only ⇒ Select the **Protocols** menu ⇒ **IP** ⇒**Routing** ⇒ Select **Export Routing Table for Selected Routers** ⇒ Click**OK** on

the *Status Confirm* dialog box.

5. Select the **Protocols** menu ⇒ **IP** ⇒ **Addressing** ⇒ Select **Auto-Assign IP Addresses**.

6. Save your project.

Configure the Simulation

Here we need to configure some of the simulation parameters:

1. Click on and the *Configure Simulation* window should appear.

2. Set the duration to be **10.0 minutes**.

3. Click **OK** and then save your project**.**

Duplicate the Scenario

In the network we just created, all routers belong to one level of hierarchy (i.e., one area). Also, we didn't enforce load balancing for any routes. Two new scenarios will be created. The first new scenario will define two new areas in addition to the backbone area. The second one will be configured to balance the load for the traffic demands between **RouterB** and **RouterH**.

The Areas Scenario:

Loopback interface allows a client and a server on the same host to communicate with each other using TCP/IP.

1. Select **Duplicate Scenario** from the **Scenarios** menu and give it the name

Areas ⇒ Click **OK**.

2. Area **0.0.0.1**:

i. Select the three links that connect **RouterA**, **RouterB**, and **RouterC** by shift-clicking on them ⇒ Select the **Protocols** menu ⇒ **OSPF** ⇒ **Configure Areas** ⇒ Assign the value **0.0.0.1** to the **Area Identifier**, as shown ⇒ Click**OK**.

ii. Right-click on **RouterC** ⇒ **Edit Attributes** ⇒ Expand the **OSPF Parameters** hierarchy ⇒ Expand the **Loopback Interfaces** hierarchy ⇒ Expand the **row0** hierarchy ⇒ Assign **0.0.0.1** to the value of the **Area ID**attribute ⇒ Click **OK**.

3. Area **0.0.0.2**:

i. Click somewhere in the project workspace to disable the selected links and then repeat step 2-i for the three links that connect **RouterF**, **RouterG**, and **RouterH** but assign the value **0.0.0.2** to their **Area Identifier**.

OPNET provides two types of IP load balancing:

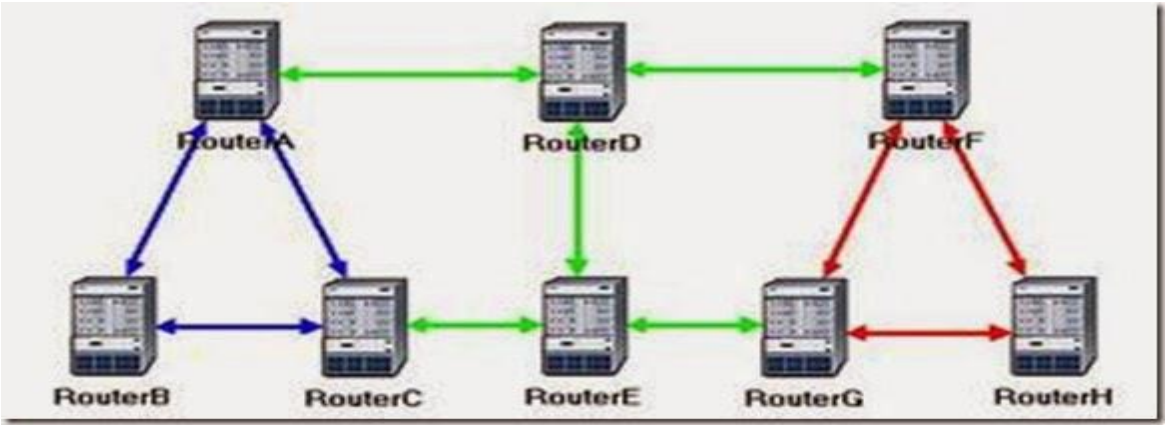
With **Destination Based**, load balancing is done on a per-destination basis. The route chosen from the source router to the destination network is the same for all packets. With **Packet Based**, load balancing is done on a per-packet basis. The route chosen from the source router to the destination network is redetermined for every individual packet.

4. To visualize the areas we just created, select the **Protocols** menu ⇒**OSPF** ⇒ **Visualize Areas** ⇒ Click **OK**. The network should look like the following one with different colors assigned to each area (you may get different colors though).

Note:

- The area you did not configure is the backbone area and its **AreaIdentifier** = 0.0.0.0.

- The figure shows the links with a thickness of 3.



The Balanced Scenario:

1. Under the **Scenarios** menu, **Switch to Scenario** ⇒ Select**No_Areas**.

1. Select **Duplicate Scenario** from the **Scenarios** menu, and give it the name

   Balanced ⇒ Click **OK**.

   3. In the new scenario, select both **RouterB** and **RouterH** by shift-clicking on them.

1. Select the **Protocols** menu ⇒ **IP** ⇒ **Routing** ⇒ **Configure Load Balancing Options** ⇒ Make sure that the option is **Packet-Based**and the radio button **Selected Routers** is selected as shown ⇒ Click**OK**.
2.

5. Save your project.

Run the Simulation

To run the simulation for the three scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Click on the row of each scenario and click the **Collect Results** button. This should change the values under the **Results** column to **<collect>** as shown.

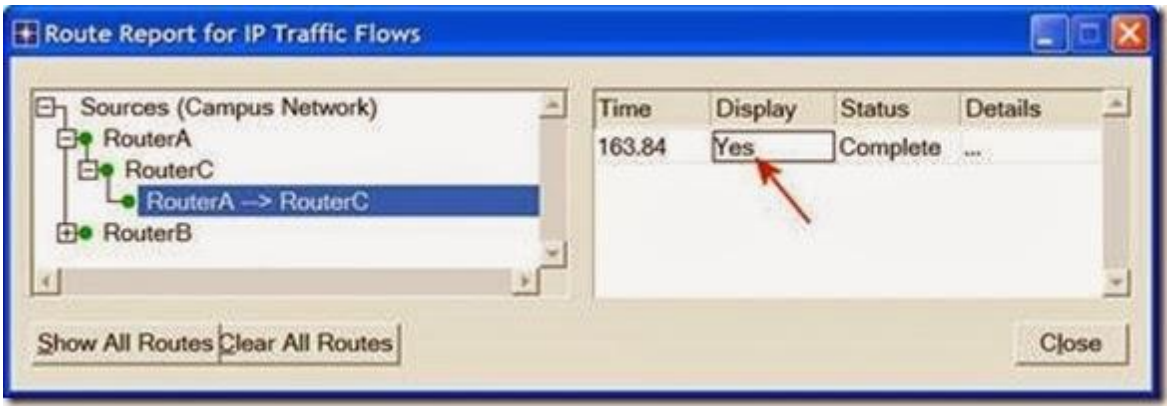
3. Click **OK** to run the three simulations. Depending on the speed of your processor, this may take several seconds to complete.

4. After the three simulation runs complete, one for each scenario, click**Close** and then save your project.

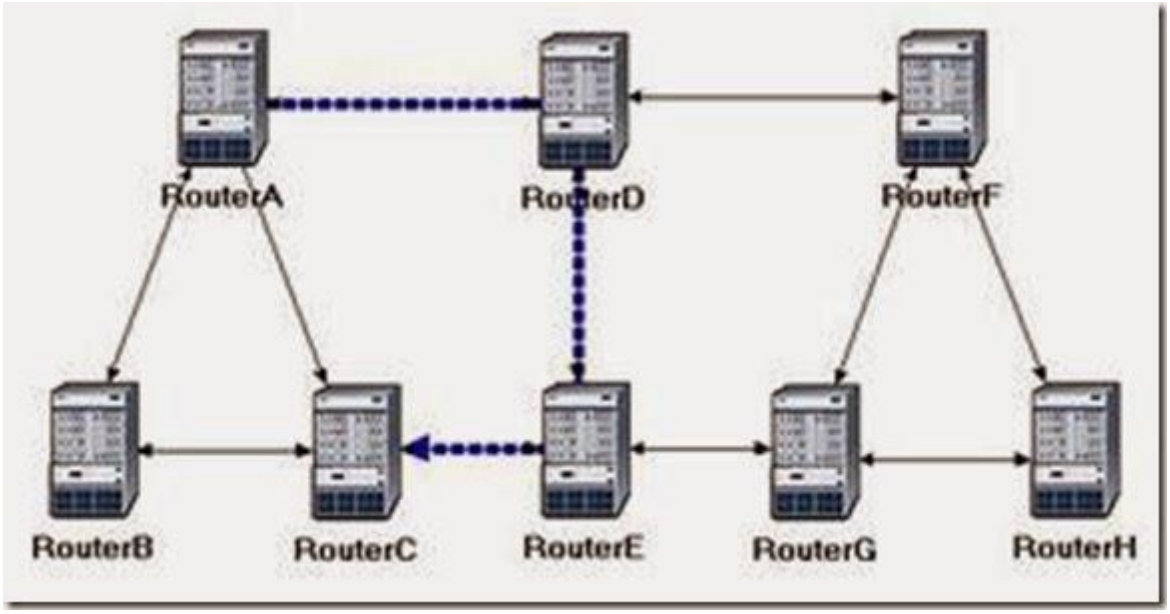
View the Results: *The No_Areas Scenario:*

1. Go back to the **No_Areas** scenario.

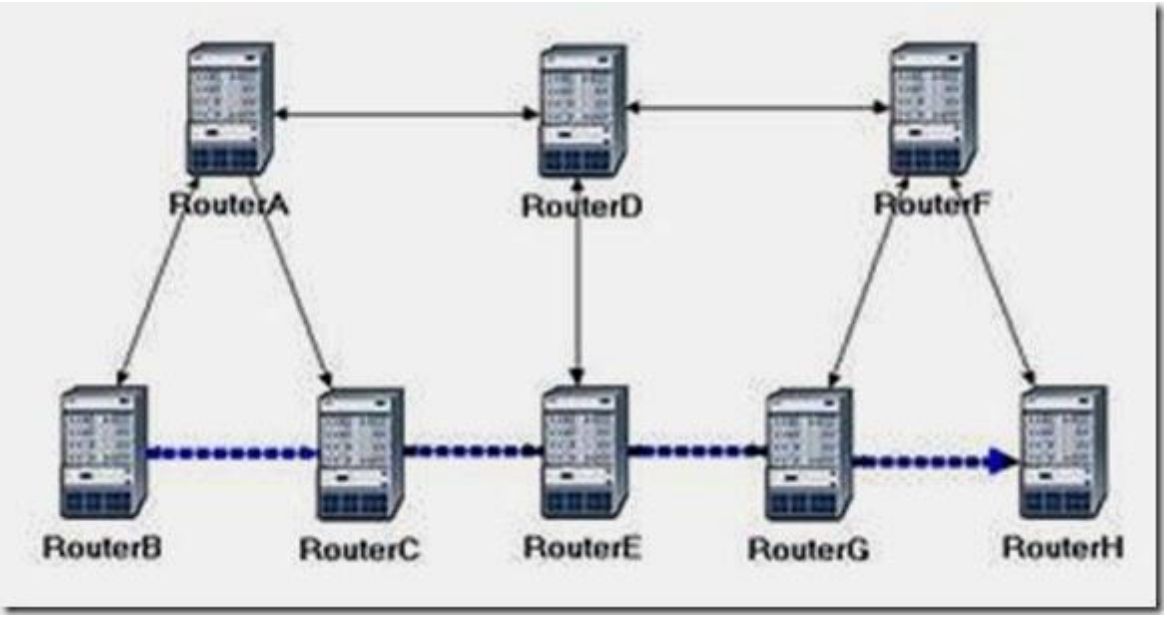
2. To display the route for the traffic demand between **RouterA** and**RouterC**: Select the **Protocols** menu ⇒ **IP** ⇒ **Demands** ⇒ **Display Routes for Configured Demands** ⇒ Expand the hierarchies as shown and select**RouterA** Æ **RouterC** ⇒ Go to the **Display** column and pick **Yes** ⇒ Click**Close**.



3. The resulting route will appear on the network as shown:



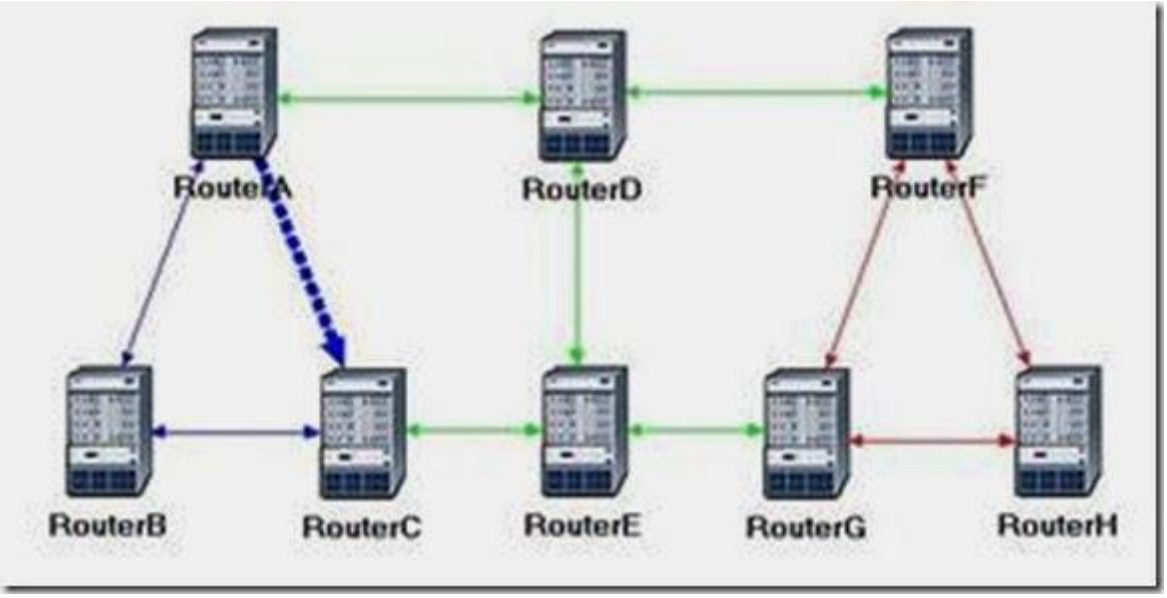
4. Repeat step 2 to show the route for the traffic demand between **RouterB**and **RouterH**. The route is as shown below. (*Note:* Depending on the order in which you created the network topology, the other "equal-cost" path can be used, that is, the *RouterB-RouterA-RouterD-RouterF-RouterH* path).



The Areas Scenario:

1. Go to scenario **Areas**.

1. Display the route for the traffic demand between **RouterA** and**RouterC**. The route is as shown:

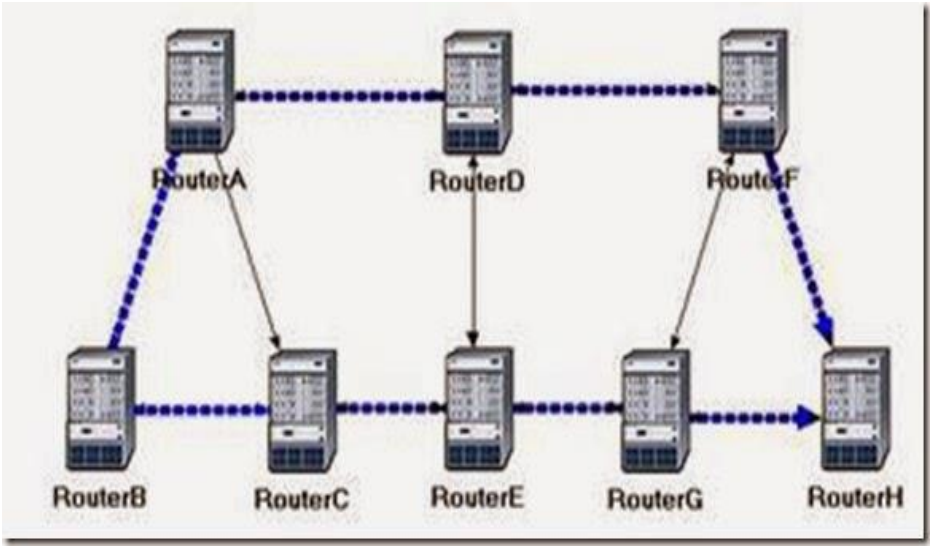


3. Save your project.

The Balanced Scenario:

1. Go to scenario **Balanced**.

1. Display the route for the traffic demand between **RouterB** and**RouterH**. The route is as shown:



Lab 27 & 28 – TCP on IT Guru

Objectives

This lab is designed to demonstrate the congestion control algorithms implemented by the Transmission Control Protocol (TCP). The lab provides a number of scenarios to simulate these algorithms. You will compare the performance of the algorithms through the analysis of the simulation results.

Lab Tasks

Task: Apply TCP on IT guru.

The Internet's TCP guarantees the reliable, in-order delivery of a stream of bytes. It includes a flow-control mechanism for the byte streams that allows the receiver to limit how much data the sender can transmit at a given time. In addition, TCP implements a highly tuned congestion-control mechanism. The idea of this mechanism is to throttle how fast TCP sends data to keep the sender from overloading the network.

The idea of TCP congestion control is for each source to determine how much capacity is available in the network, so that it knows how many packets it can safely have in transit. It maintains a state variable for each connection, called the*congestion window***,** which is used by the source to limit how much data it is allowed to have in transit at a given time. TCP uses a mechanism, called *additive increase/multiplicative decrease,* that decreases the congestion window when the level of congestion goes up and increases the congestion window when the level of congestion goes down. TCP interprets timeouts as a sign of congestion. Each time a timeout occurs, the source sets the congestion window to half of its previous value. This halving corresponds to the*multiplicative decrease* part of the mechanism. The congestion window is not allowed to fall below the size of a single packet (the TCP maximum segment size, or MSS). Every time the source successfully sends a congestion window's worth of packets, it adds the equivalent of one packet to the congestion window; this is the *additive increase*part of the mechanism.

TCP uses a mechanism called *slow start* to increase the congestion window "rapidly" from a cold start in TCP connections. It increases the congestion window exponentially, rather than linearly. Finally, TCP utilizes a mechanism called*fast retransmit and fast recovery*. Fast retransmit is a heuristic that sometimes triggers the retransmission of a dropped packet sooner than the regular timeout mechanism

In this lab you will set up a network that utilizes TCP as its end-to-end transmission protocol and analyze the size of the congestion window with different mechanisms.

Procedure

Create a New Project

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose **New**from the **File** menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_TCP**, and the scenario **No_Drop** ⇒ Click **OK**.

3. In the *Startup Wizard: Initial Topology* dialog box, make sure that **Create Empty Scenario** is selected ⇒ Click **Next** ⇒Select **Choose From Maps** from the *Network Scale* list ⇒Click **Next** ⇒ Choose **USA** from the Map List ⇒ Click **Next**twice ⇒ Click **OK**.

Create and Configure the Network

Initialize the Network:

The **ip32_cloud** node model represents an IP cloud supporting up to 32 serial line interfaces at a selectable data rate through which IP traffic can be modeled. IP packets arriving on any cloud interface are routed to the appropriate output interface based on their destination IP address. The RIP or OSPF protocol may be used to automatically and dynamically create the cloud's routing tables and select routes in an adaptive manner. This cloud requires a fixed amount of time to route each packet, as determined by the

Packet Latency attribute of the node.

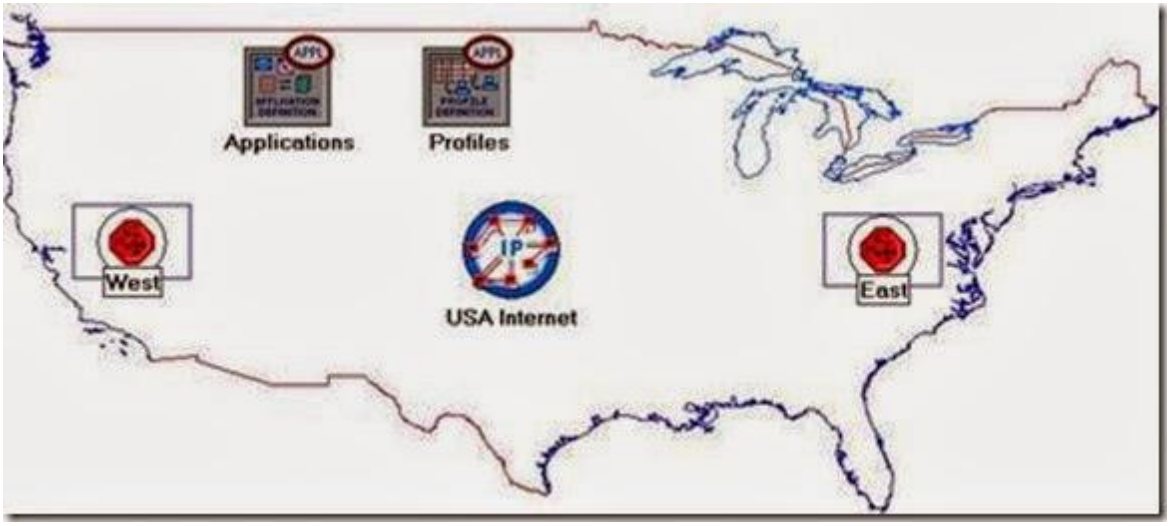
1. The *Object Palette* dialog box should now be on the top of your project space. If it is not there, open it by clicking . Make sure that the **internet_toolbox** item is selected from the pull-down menu on the object palette.

2. Add to the project workspace the following objects from the palette: **Application Config**, **Profile Config**, an **ip32_Cloud,**and two subnets.

a. To add an object from a palette, click its icon in the object palette ⇒ Move your mouse to the workspace ⇒ Click to drop the object in the desired location ⇒ Right-click to finish creating objects of that type.

3. Close the palette.

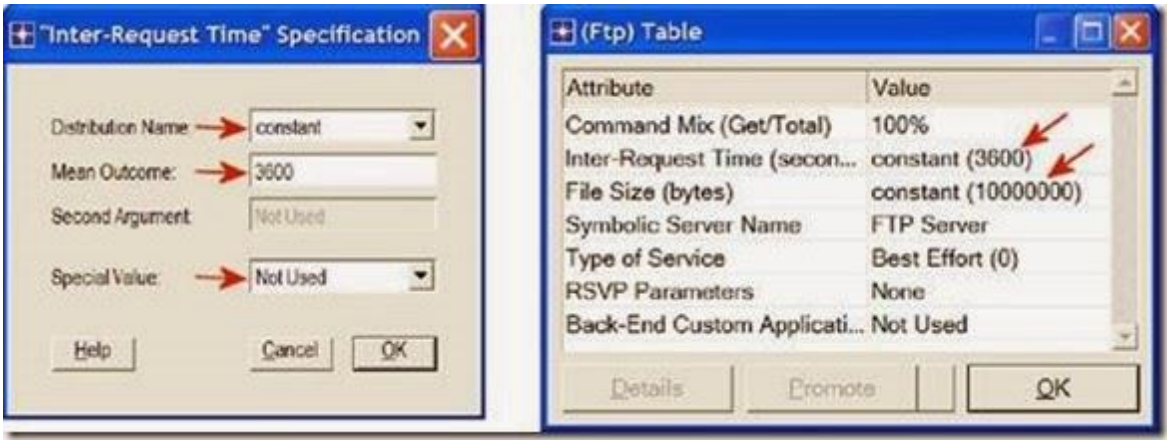
4. Rename the objects you added as shown and then save your project:



Configure the Applications:

1. Right-click on the **Applications** node ⇒ **Edit Attributes** ⇒ Expand the **Application Definitions**attribute and set **rows** to 1 ⇒ Expand the new row ⇒Name the row **FTP_Application**.

   i. Expand the **Description** hierarchy ⇒ Edit the **FTP** row as shown (you will need to set the **Special Value** to **Not Used**while editing the shown attributes):

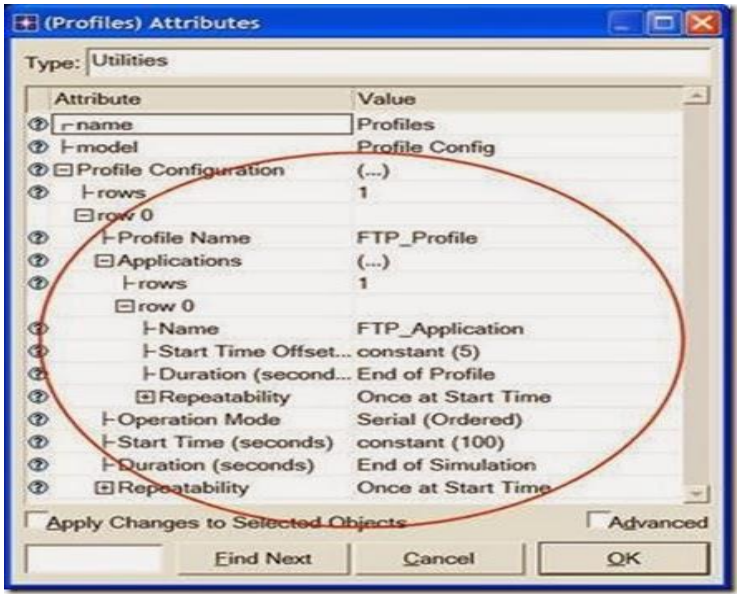


   2. Click **OK** twice and then save your project.

   Configure the Profiles:

1. Right-click on the **Profiles** node ⇒ **Edit Attributes** ⇒ Expand the **Profile Configuration** attribute and set**rows** to 1.

   i. Name and set the attributes of **row 0** as shown ⇒ Click **OK**.



The **ethernet4_slip8_ gtwy** node model represents an IP-based gateway supporting four Ethernet hub interfaces and eight serial line interfaces.

Configure the West Subnet:

1. Double-click on the **West** subnet node. You get an empty workspace, indicating that the subnet contains no objects.

1. Open the object palette and make sure that the**internet_toolbox** item is selected from the pull-down menu.

1. Add the following items to the subnet workspace: one**ethernet_server**, one **ethernet4_slip8_gtwy** router, and connect them with a bidirectional **100_BaseT** link ⇒**Close** the palette ⇒ Rename the objects as shown.

1. Right-click on the **Server_West** node ⇒ **Edit Attributes**:

i. Edit **Application: Supported Services** ⇒ Set **rows** to 1 ⇒ Set **Name** to **FTP_Application** ⇒ Click **OK**.

ii. Edit the value of the **Server Address** attribute and write down **Server_West**.

iii. Expand the **TCP Parameters** hierarchy ⇒ Set both **Fast Retransmit** and

Fast Recovery to **Disabled**.

1. Click **OK** and then save your project.

Now, you have completed the configuration of the **West**subnet. To go back to the top level of the project, click the **Go to next higher level** button.

Configure the East Subnet:

1. Double-click on the **East** subnet node. You get an empty workspace, indicating that the subnet contains no objects.

1. **Open** the object palette and make sure that the**internet_toolbox** item is selected from the pull-down menu.

1. Add the following items to the subnet workspace: one**ethernet_wkstn**, one **ethernet4_slip8_gtwy** router, and connect them with a bidirectional **100_BaseT** link ⇒**Close** the palette ⇒ Rename the objects as shown.

1. Right-click on the **Client_East** node ⇒ **Edit Attributes**:

i. Expand the **Application: Supported Profiles** hierarchy ⇒ Set **rows** to 1 ⇒ Expand the **row 0** hierarchy ⇒ Set **Profile Name** to **FTP_Profile.**

ii. Assign **Client_ East** to the **Client Address** attributes.

iii. Edit the **Application: Destination Preferences** attribute as follows:

Set **rows** to 1 ⇒ Set **Symbolic Name** to **FTP Server** ⇒ Edit**Actual Name** ⇒

Set **rows** to 1 ⇒ In the new row, assign **Server_West** to the**Name** column.

1. Click **OK** three times and then save your project.

1. You have now completed the configuration of the **East**subnet. To go back to the project space, click the **Go to next higher level** button.

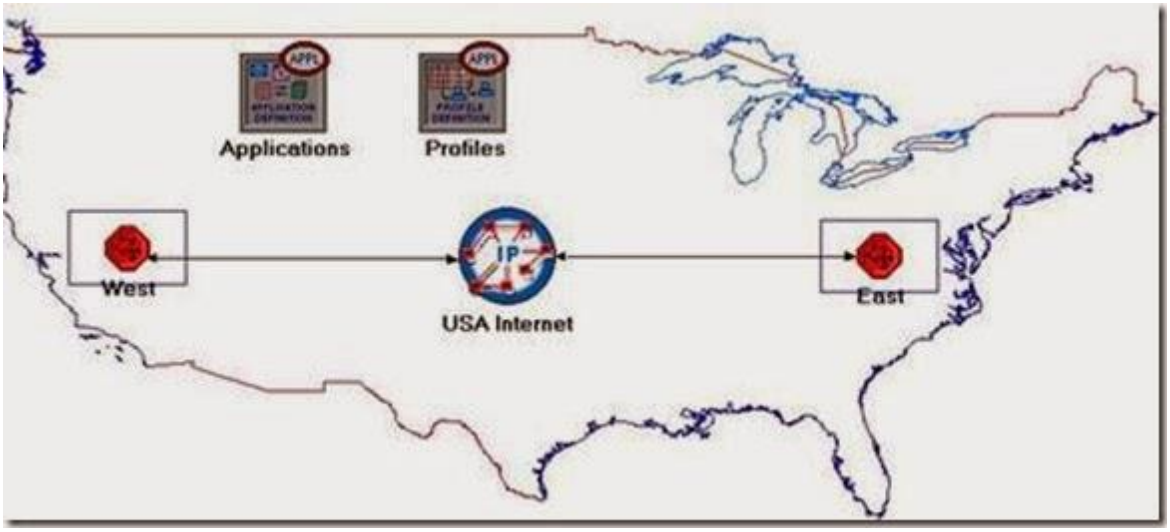
Connect the Subnets to the IP Cloud:

1. Open the object palette .

1. Using two **PPP_DS3** bidirectional links connect the**East** subnet to the **IP Cloud** and the **West** subnet to the**IP Cloud**.

1. A pop-up dialog box will appear asking you what to connect the subnet to the **IP Cloud** with. Make sure to select the "routers."

1. Close the palette.



Choose the Statistics

OPNET provides the following capture modes:

All values—collects every data point from a statistic.

Sample—collects the data according to a user-specified time interval or sample count. For example, if the time interval is 10, data is sampled and recorded every 10th second. If the sample count is 10, every 10th data point is recorded. All other data points are discarded.

Bucket—collects all of the points over the time interval or sample count into a "data bucket" and generates a result from each bucket. This is the default mode.

1. Right-click on **Server_West** in the **West** subnet and select**Choose Individual Statistics** from the pop-up menu.

2. In the *Choose Results* dialog box, choose the following statistic:

TCP Connection ⇒ Congestion Window Size (bytes) and**Sent Segment Sequence Number**.

3. Right-click on the **Congestion Window Size (bytes)**statistic ⇒ Choose **Change Collection Mode** ⇒ In the dialog box check **Advanced** ⇒ From the drop-down menu, assign **all values** to **Capture mode** as shown ⇒ Click **OK**.

4. Right-click on the **Sent Segment Sequence Number**statistic ⇒ Choose

Change Collection Mode ⇒ In the dialog box check**Advanced** ⇒ From the drop-down menu, assign **all values** to**Capture mode**.

5. Click **OK** twice and then save your project.

6. Click the **Go to next higher level** button.

Configure the Simulation

Here we need to configure the duration of the simulation:

1. Click on and the *Configure Simulation* window should appear.

2. Set the duration to be **10.0 minutes**.

3. Click **OK** and then save your project**.**

Duplicate the Scenario

With **fast retransmit,**

TCP performs a retransmission of what appears to be the missing segment, without waiting for a retransmission timer to expire.

After fast retransmit sends what appears to be the missing segment, congestion avoidance but not slow start is performed. This is the **fast recovery** algorithm.

The fast retransmit and fast recovery algorithms are usually implemented together (RFC 2001).

In the network we just created we assumed a perfect network with no discarded packets. Also, we disabled the fast retransmit and fast recovery techniques in TCP. To analyze the effects of discarded packets and those congestion-control techniques, we will create two additional scenarios.

1. Select **Duplicate Scenario** from the **Scenarios** menu and give it the name

   Drop_NoFast ⇒ Click **OK**.

1. In the new scenario, right-click on the **IP Cloud** ⇒ **Edit Attributes** ⇒ Assign

   0.05% to the **Packet Discard Ratio** attribute.

1. Click **OK** and then save your project.

1. While you are still in the **Drop_NoFast** scenario, select**Duplicate Scenario** from the **Scenarios** menu and give it the name **Drop_Fast**.

5. In the **Drop_Fast** scenario, right-click on **Server_ West**, which is inside the **West** subnet ⇒ **Edit Attributes** ⇒ Expand the **TCP Parameters** hierarchy ⇒ **Enable** the **Fast Retransmit** attribute ⇒ Assign **Reno** to the **Fast Recovery**attribute.

1. Click **OK** and then save your project.

Run the Simulation

To run the simulation for the three scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Change the values under the **Results** column to **<collect>**(or **<recollect>**) for the three scenarios. Compare to the following figure.

3. Click **OK** to run the three simulations. Depending on the speed of your processor, this may take several minutes to complete.

4. After the three simulation runs complete, one for each scenario, click **Close** ⇒ Save your project.

View the Results

To switch to a scenario, choose **Switch to Scenario** from the

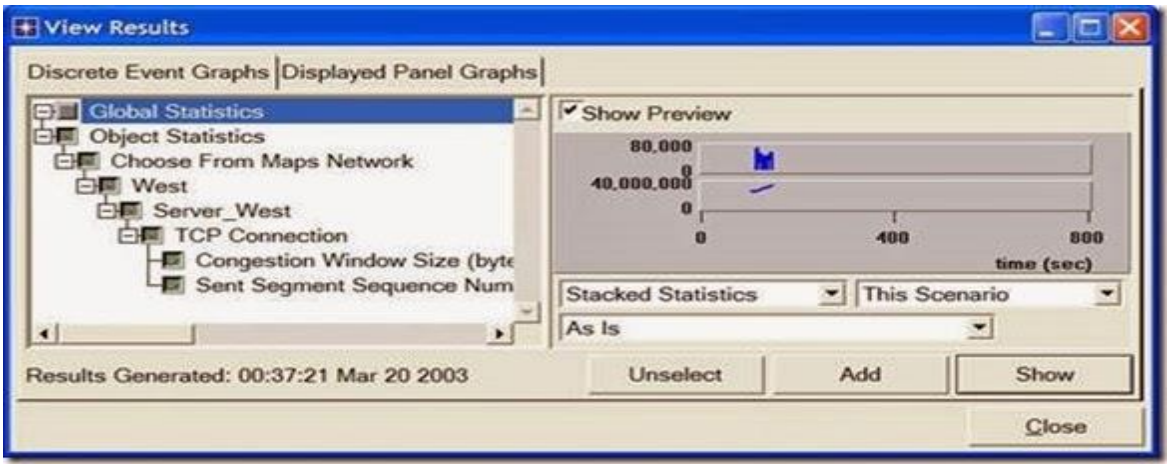
Scenarios menu or just press **Ctrl+<scenario number>.**

To view and analyze the results:

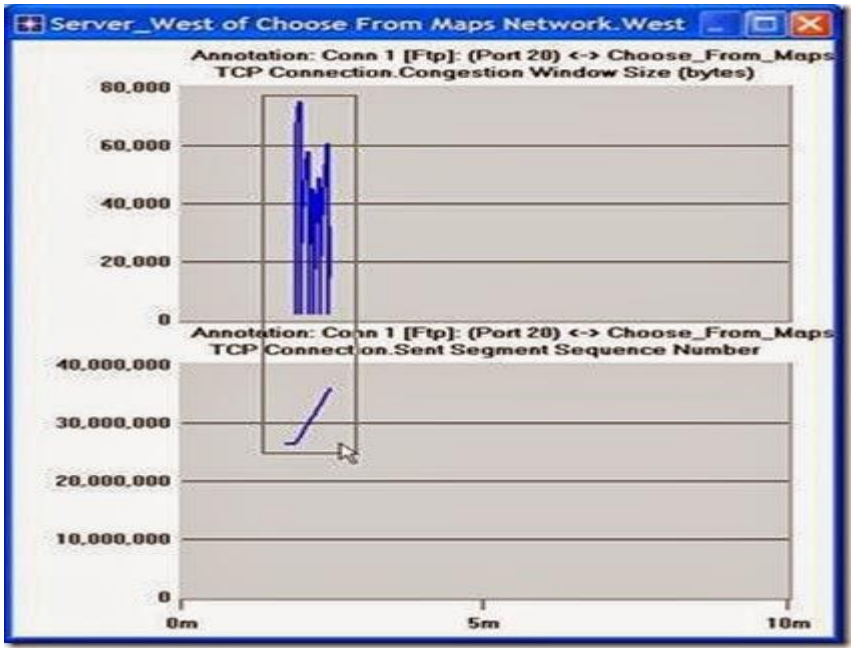
1. Switch to the **Drop_NoFast** scenario (the second one) and choose **View Results** from the **Results** menu.

2. Fully expand the **Object Statistics** hierarchy and select the following two results:

Congestion Window Size (bytes) and **Sent Segment Sequence Number**.

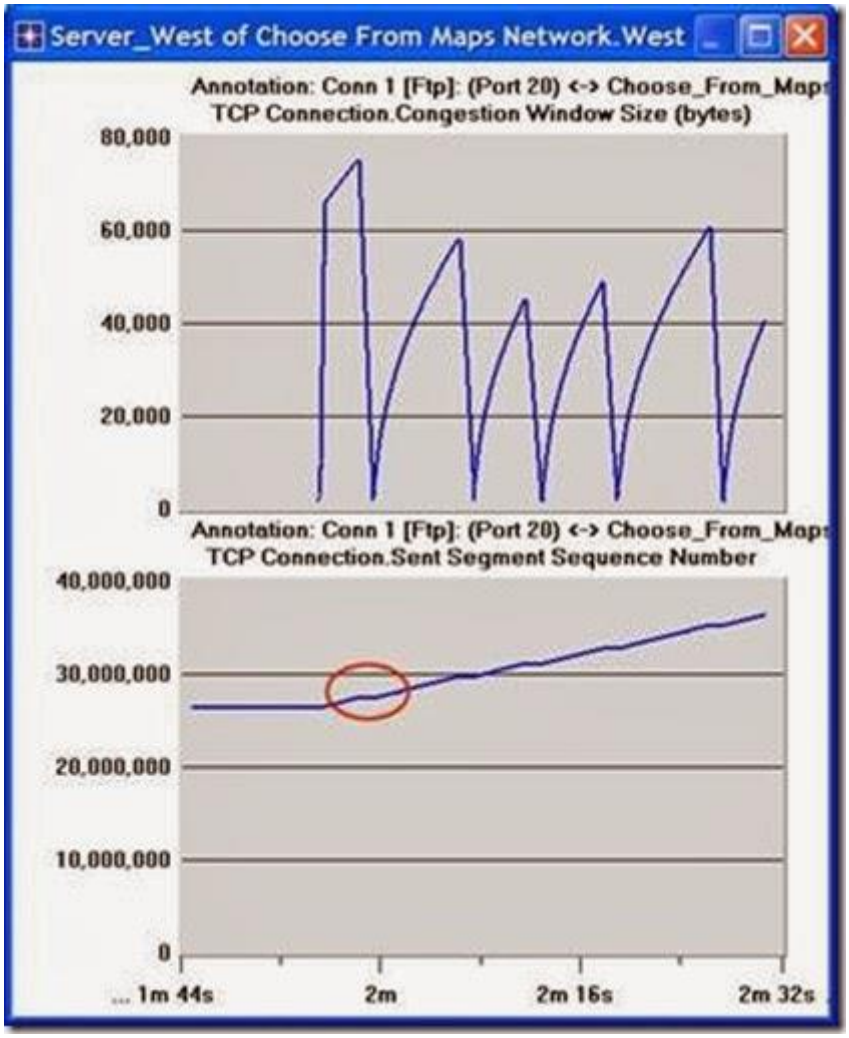


3. Click **Show**. The resulting graphs should resemble the ones below.



4. To zoom in on the details in the graph, click and drag your mouse to draw a rectangle, as shown above.

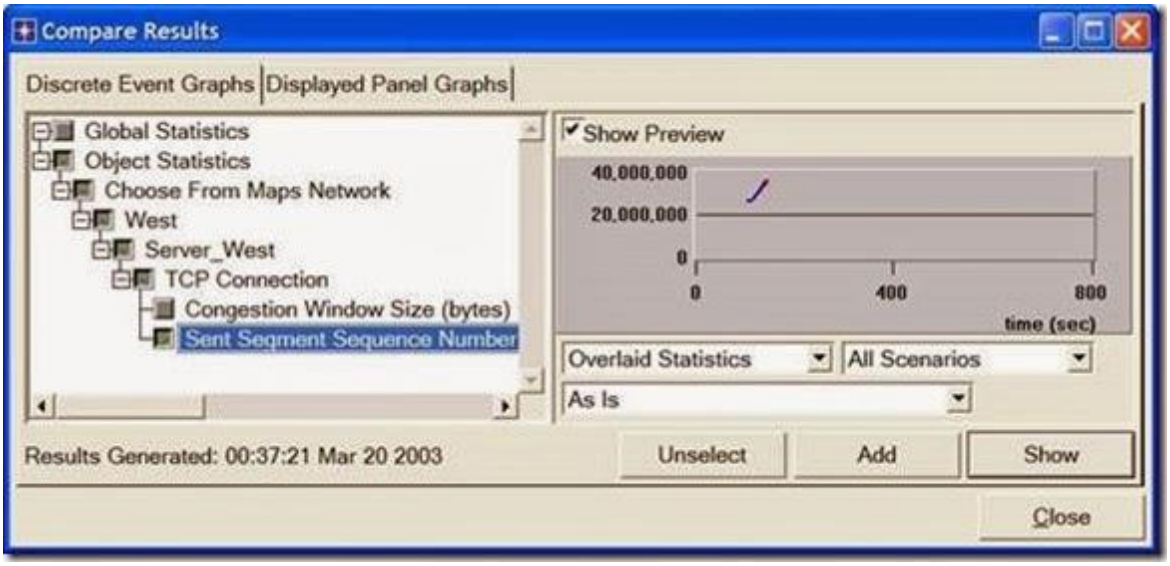
5. The graph should be redrawn to resemble the following one:



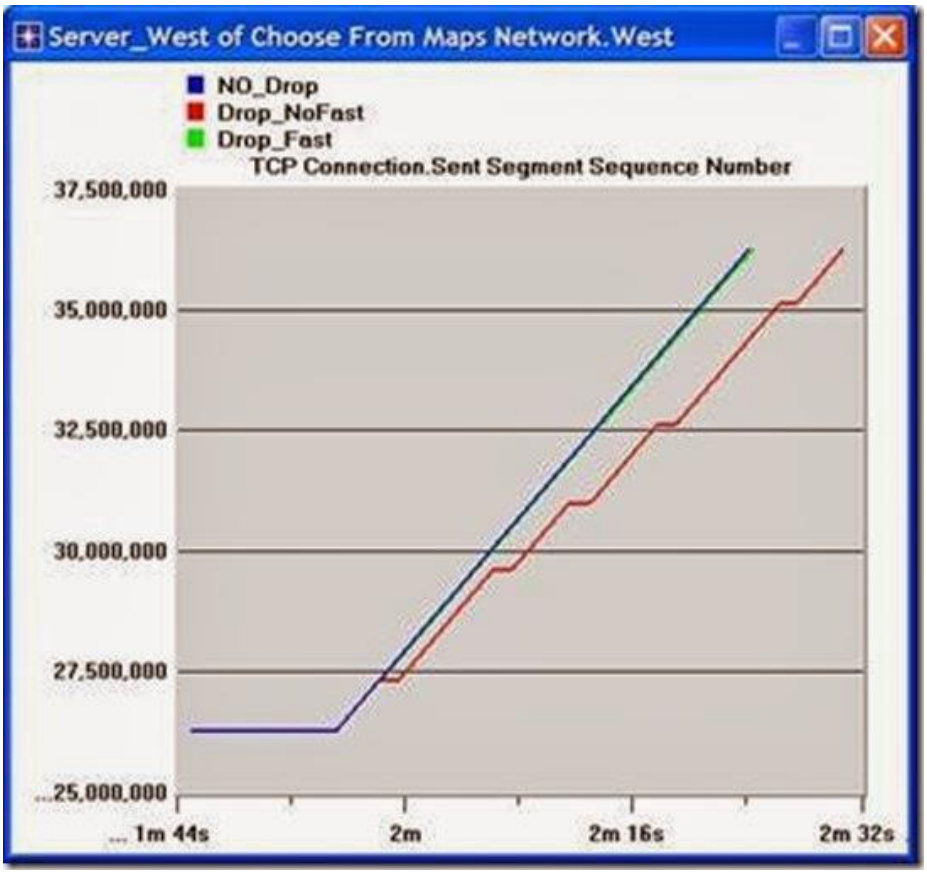
6. Notice the **Segment Sequence Number** is almost flat with every drop in the congestion window.

7. Close the *View Results* dialog box and select **Compare Results** from the **Result** menu.

8. Fully expand the **Object Statistics** hierarchy as shown and select the following result: **Sent Segment Sequence Number**.



9. Click **Show**. After zooming in, the resulting graph should resemble the one below.



Lab 29 & 30 – Network Security and Virtual Private Networks

Objectives

The objective of this lab is to study the role of firewalls and Virtual Private Networks (VPNs) in providing security to shared public networks such as the Internet.

Lab Tasks

Task: Apply Firewall and VPN on IT Guru

Computer networks are typically a shared resource used by many applications for many different purposes. Sometimes the data transmitted between application processes is confidential, and the application users would prefer that others not be able to read it.

A firewall is a specially programmed router that sits between a site and the rest of the network. It is a router in the sense that it is connected to two or more physical networks and it forwards packets from one network to another, but it also filters the packets that flow through it. A firewall allows the system administrator to implement a security policy in one centralized place. Filter-based firewalls are the simplest and most widely deployed type of firewall. They are configured with a table of addresses that characterize the packets they will and will not forward.

A VPN is an example of providing a controlled connectivity over a public network such as the Internet. VPNs utilize a concept called an *IP tunnel*—a virtual point-to-point link between a pair of nodes that are actually separated by an arbitrary number of networks. The virtual link is created within the router at the entrance to the tunnel by providing it with the IP address of the router at the far end of the tunnel. Whenever the router at the entrance of the tunnel wants to send a packet over this virtual link, it encapsulates the packet inside an IP datagram. The destination address in the IP header is the address of the router at the far end of the tunnel, while the source address is that of the encapsulating router.

In this lab you will set up a network where servers are accessed over the Internet by customers who have different privileges. You will study how firewalls and VPNs can provide security to the information in the servers while maintaining access for customers with the appropriate privilege.

Procedure

Create a New Project

1. Start **OPNET IT Guru Academic Edition** ⇒ Choose**New** from the **File** menu.

2. Select **Project** and click **OK** ⇒ Name the project <**your initials**>**_VPN**, and the scenario **NoFirewall** ⇒ Click **OK**.

3. Click **Quit** on the *Startup Wizard*.

4. To remove the world background map, select the**View** menu ⇒ **Background** ⇒ **Set Border Map** ⇒Select **NONE** from the drop-down menu ⇒ Click **OK**.

Create and Configure the Network

The **ppp_server** and **ppp_wkstn** support oneunderlying SLIP (Serial Line Internet Protocol) connection at a selectable data rate.

PPP DS1 connects two nodes running IP. Its data rate is 1.544 Mbps.

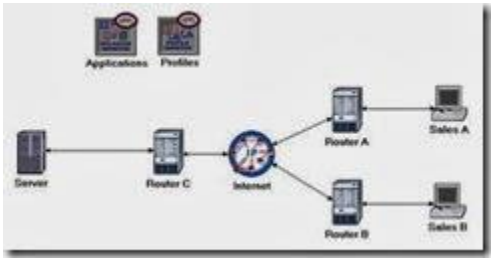
Initialize the Network:

1. Open the *Object Palette* dialog box by clicking . Make sure that the **internet_toolbox** item is selected from the pull-down menu on the object palette.

2. Add the following objects, from the palette, to the project workspace (see figure below for placement):**Application Config**, **Profile Config**, an **ip32_cloud**, one **ppp_server**, three **ethernet4_slip8_gtwy** routers, and two **ppp_wkstn** hosts.

a. To add an object from a palette, click its icon in the object palette ⇒ Move your mouse to the workspace and click where you want to place the object ⇒ Right-click to indicate you are done creating objects of this type.

1. Rename the objects you added and connect them using **PPP DS1** links, as shown below:



4. Save your project.

Several example application configurations are available under the **Default** setting. For example, "Web Browsing (Heavy HTTP1.1)" indicates a Web browsing application performing heavy browsing using HTTP 1.1 protocol..

Configure the Nodes:

1. Right-click on the **Applications** node ⇒ **Edit Attributes** ⇒ Assign **Default** to the **Application Definitions** attribute ⇒ Click **OK**.

1. Right-click on the **Profiles** node ⇒ **Edit Attributes** ⇒ Assign **Sample Profiles** to the**Profile Configuration** attribute ⇒ Click **OK**.

1. Right-click on the **Server** node ⇒ **Edit Attributes**⇒ Assign **All** to the

   Application: Supported Services attribute ⇒ Click **OK**.

   4. Right-click on the **Sales A** node ⇒ **Select Similar Nodes** (make sure that both **Sales A** and **Sales B** are selected).

   i. Right-click on the **Sales A** node ⇒ **Edit Attributes** ⇒ Check the **Apply Changes to Selected Objects** check-box.

   ii. Expand the **Application: Supported Profiles**attribute ⇒ Set **rows** to 1 ⇒ Expand the **row 0**hierarchy ⇒ **Profile Name** = **Sales Person** (this is one of the "sample profiles" we configured in the **Profiles**node).

   iii. Click **OK**.

   5. Save your project.

   Choose the Statistics

   DQ Query Response Time is measured from the time when the database query application

   sends a request to the server to the time it receives a response packet.

   HTTP Page Response Time specifies the time required to retrieve the entire page with all the contained inline objects.

   1. Right-click anywhere in the project workspace and select **Choose Individual Statistics** from the pop-up menu.

   2. In the *Choose Results* dialog, check the following statistics:

   i. **Global Statistics ⇒ DB Query ⇒ Response Time (sec).**

   ii. **Global Statistics ⇒ HTTP ⇒ Page Response Time (seconds).**

   3. Click **OK**.

4. Right-click on the **Sales A** node and select **Choose Individual Statistics** from the pop-up menu.

5. In the *Choose Results* dialog, check the following statistics:

i. **Client DB ⇒ Traffic Received (bytes/sec).**

ii. **Client Http ⇒ Traffic Received (bytes/sec).**

6. Click **OK**.

7. Right-click on the **Sales B** node and select **Choose Individual Statistics** from the pop-up menu.

8. In the *Choose Results* dialog, check the following statistics:

i. **Client DB ⇒ Traffic Received (bytes/sec).**

ii. **Client Http ⇒ Traffic Received (bytes/sec).**

9. Click **OK** and then save your project.

The Firewall Scenario

In the network we just created, the **Sales Person** profile allows both sales sites to access applications such as Database Access, Email, and Web Browsing from the server (check the **Profile Configuration** of the **Profiles**node). Assume that we need to protect the database in the server from external access, including the salespeople. One way to do that is to replace Router C with a firewall as follows:

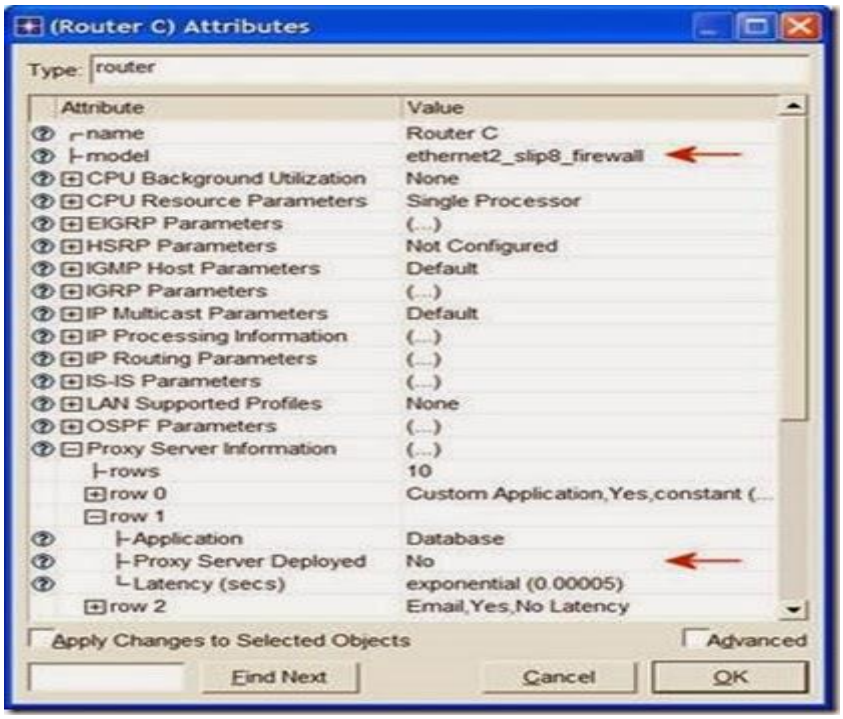
1. Select **Duplicate Scenario** from the **Scenarios** menu and name it **Firewall ⇒**

Click **OK**.

2. In the new scenario, right-click on **Router C ⇒ Edit Attributes**.

3. Assign **ethernet2_slip8_firewall** to the **model**attribute.

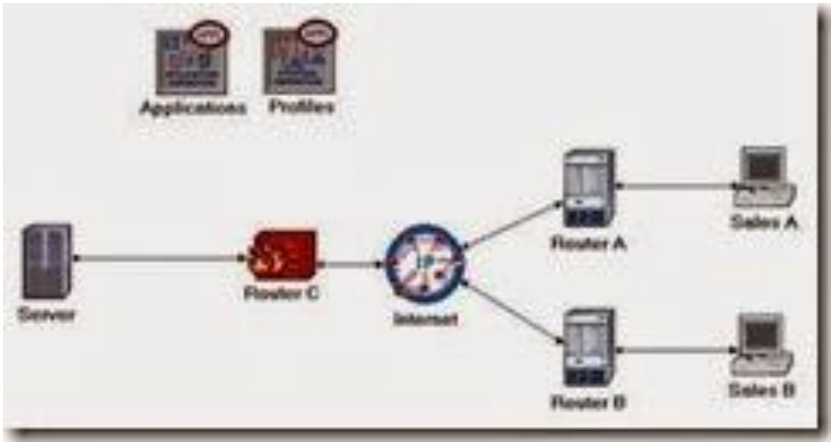
4. Expand the hierarchy of the **Proxy Server Information** attribute ⇒ Expand the **row 1**, which is for the Database application, hierarchy ⇒ Assign **No** to the**Proxy Server Deployed** attribute as shown:



Proxy Server Information is a table defining the configuration of the proxy servers on the firewall. Each row indicates whether a proxy server exists for a certain application and the amount of additional delay that will be introduced to each forwarded packet of that application by the proxy server.

5. Click **OK** and then save your project.

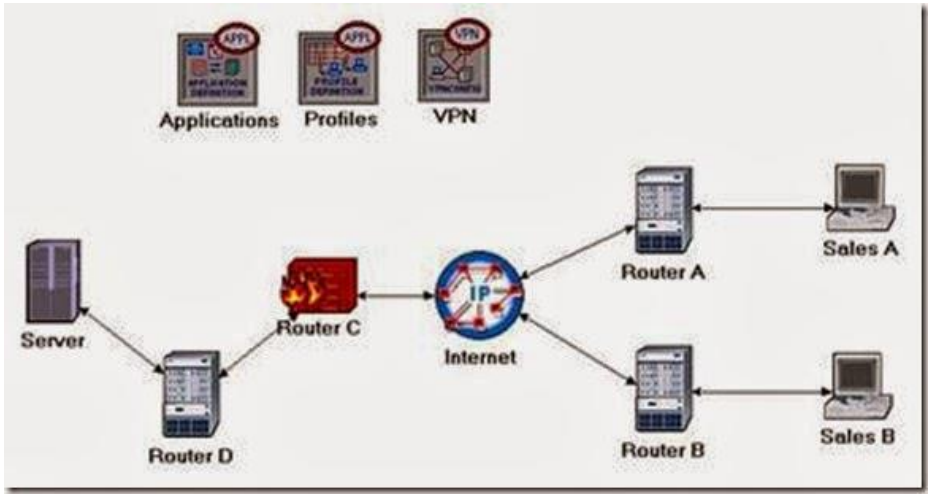
Our **Firewall** configuration does not allow database-related traffic to pass through the firewall (it filters such packets out). This way, the databases in the server are protected from external access. Your **Firewall** scenario should look like the following figure.



The Firewall_VPN Scenario

In the **Firewall** scenario, we protected the databases in the server from "any" external access using a firewall router. Assume that we want to allow the people in the**Sales A** site to have access to the databases in the server. Since the firewall filters all database-related traffic regardless of the source of the traffic, we need to consider the VPN solution. A virtual tunnel can be used by Sales A to send database requests to the server. The firewall will not filter the traffic created by **Sales A**because the IP packets in the tunnel will be encapsulated inside an IP datagram.

1. While you are in the **Firewall** scenario, select**Duplicate Scenario** from the **Scenarios** menu and give it the name **Firewall_VPN** ⇒ Click **OK**.
2. Remove the link between **Router C** and the **Server**.
3. Open the *Object Palette* dialog box by clicking . Make sure that the opened palette is the one called**internet_toolbox**.
i. Add to the project workspace one**ethernet4_slip8_gtwy** and one **IP VPN Config** (see the figure below for placement).
ii. From the *Object Palette*, use two **PPP DS1** links to connect the new router to **Router C** (the firewall) and to the **Server,** as shown below.
iii. Close the *Object Palette* dialog box.
4. Rename the **IP VPN Config** object to **VPN**.
5. Rename the new router to **Router D** as shown:



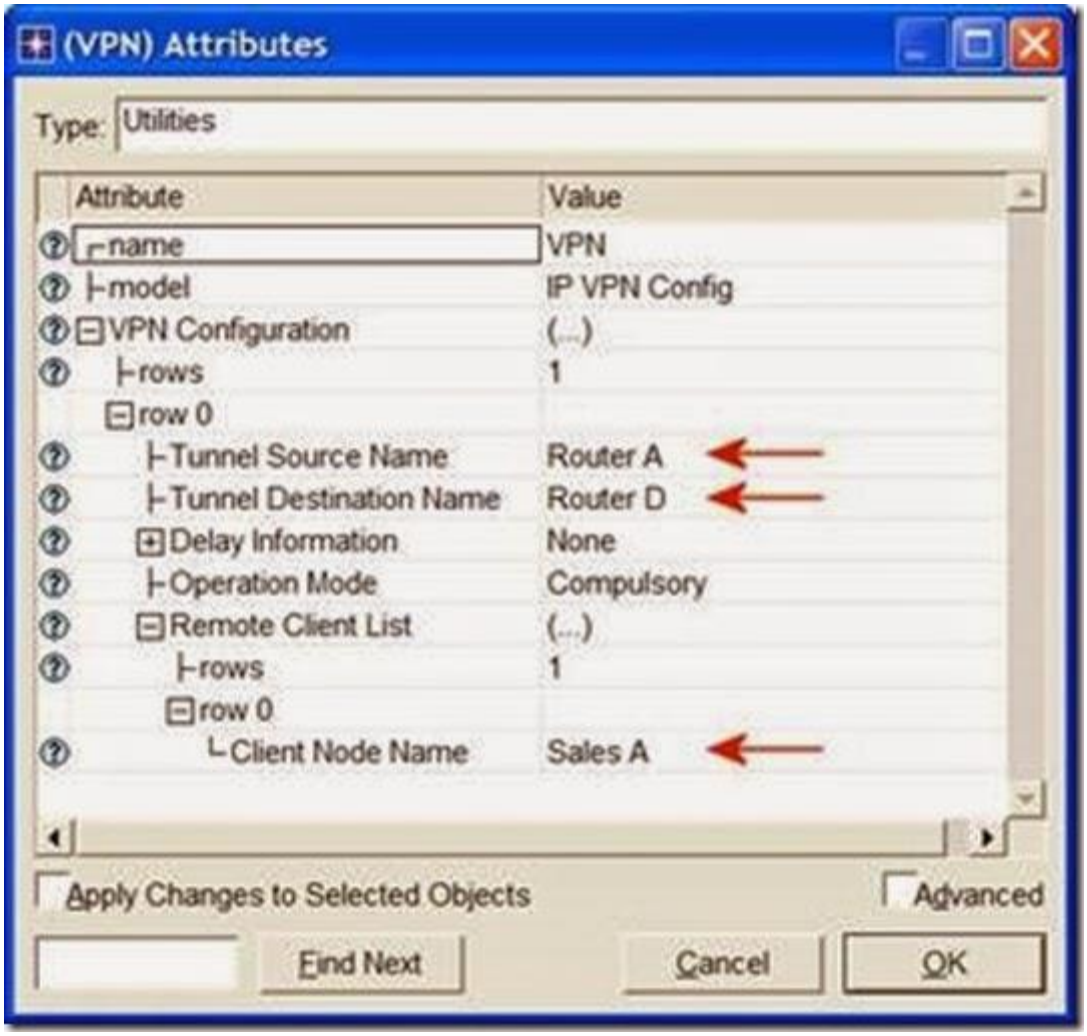
Configure the VPN:

1. Right-click on the **VPN** node ⇒ **Edit Attributes.**

i. Expand the **VPN Configuration** hierarchy ⇒ Set**rows** to 1 ⇒ Expand **row 0** hierarchy ⇒ Edit the value of **Tunnel Source Name** and write down **Router A** ⇒Edit the value of **Tunnel Destination Name** and write down **Router D**.

ii. Expand the **Remote Client List** hierarchy ⇒ Set**rows** to 1 ⇒ Expand **row 0** hierarchy ⇒ Edit the value of **Client Node Name** and write down **Sales A**.

iii. Click **OK** and then save your project.

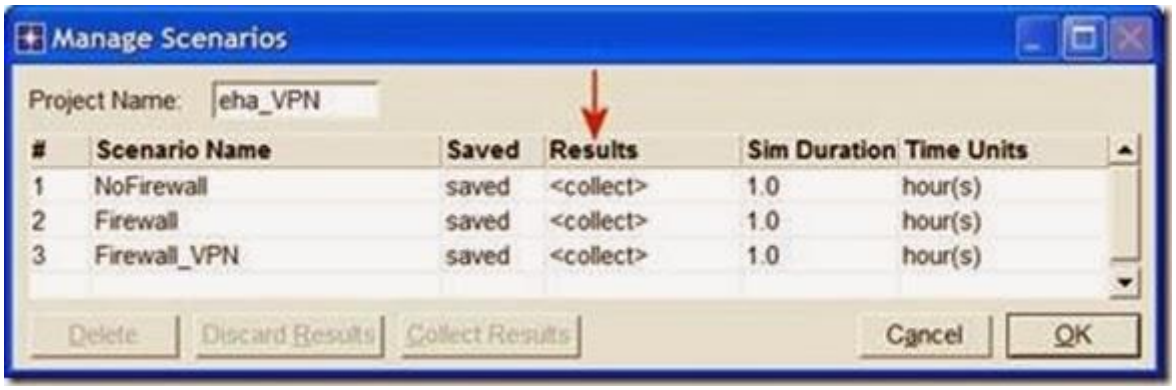


Run the Simulation

To run the simulation for the three scenarios simultaneously:

1. Go to the **Scenarios** menu ⇒ Select **Manage Scenarios**.

2. Change the values under the **Results** column to**<collect>** (or **<recollect>**) for the three scenarios. Keep the default value of the **Sim Duration** (1 hour). Compare to the following figure.



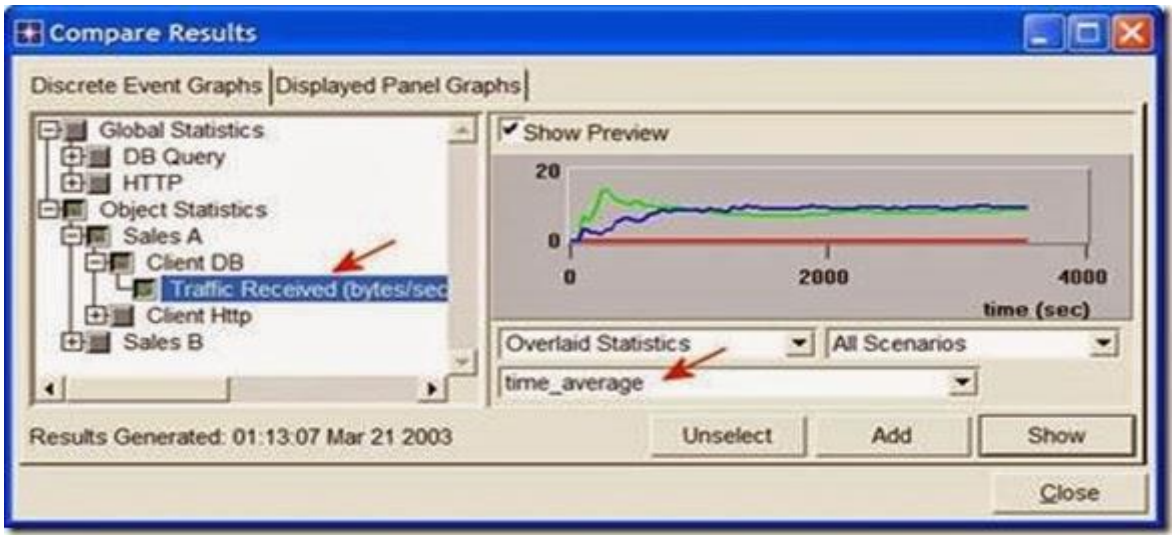
3. Click **OK** to run the three simulations. Depending on the speed of your processor, this may take several minutes to complete.

4. After the three simulation runs complete, one for each scenario, click **Close** ⇒ Save your project.

View the Results

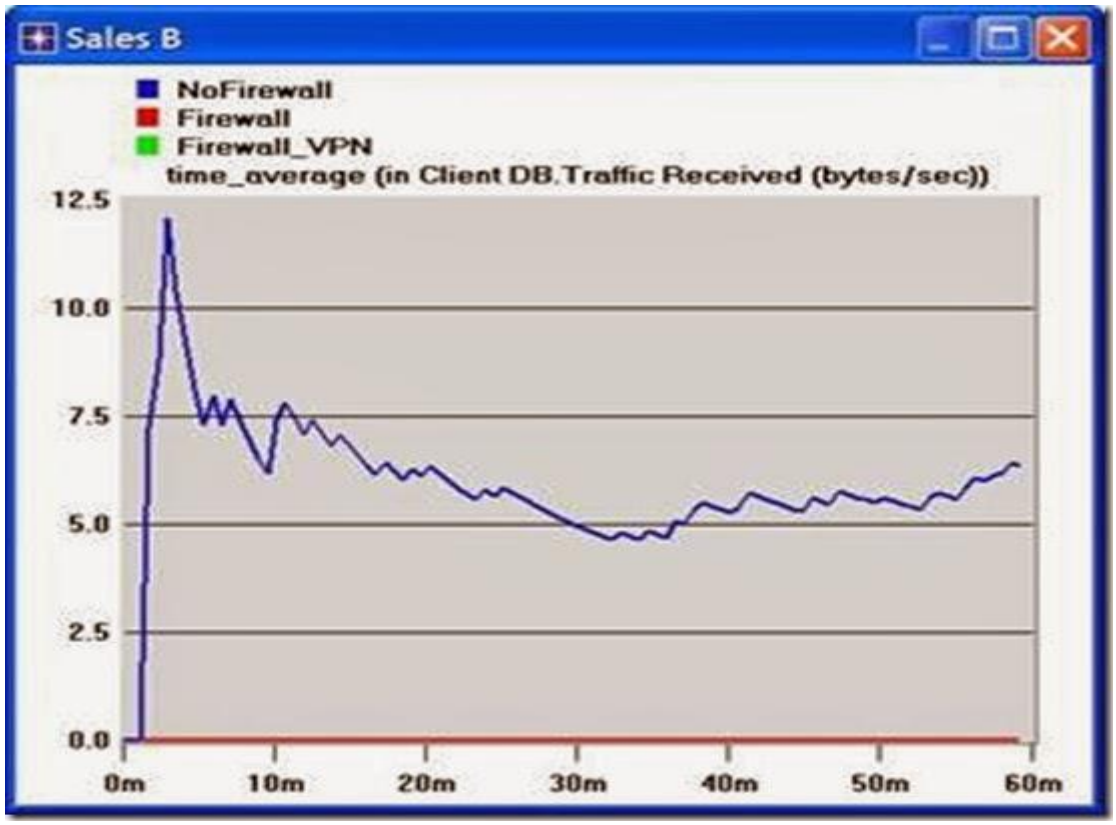
To view and analyze the results:

1. Select **Compare Results** from the **Results** menu.
2. Expand the **Sales A** hierarchy ⇒ Expand the **Client DB** hierarchy ⇒ Select the **Traffic Received** statistic.
3. Change the drop-down menu in the middle-lower part of the **Compare Results** dialog box from **As Is** to**time_average** as shown.

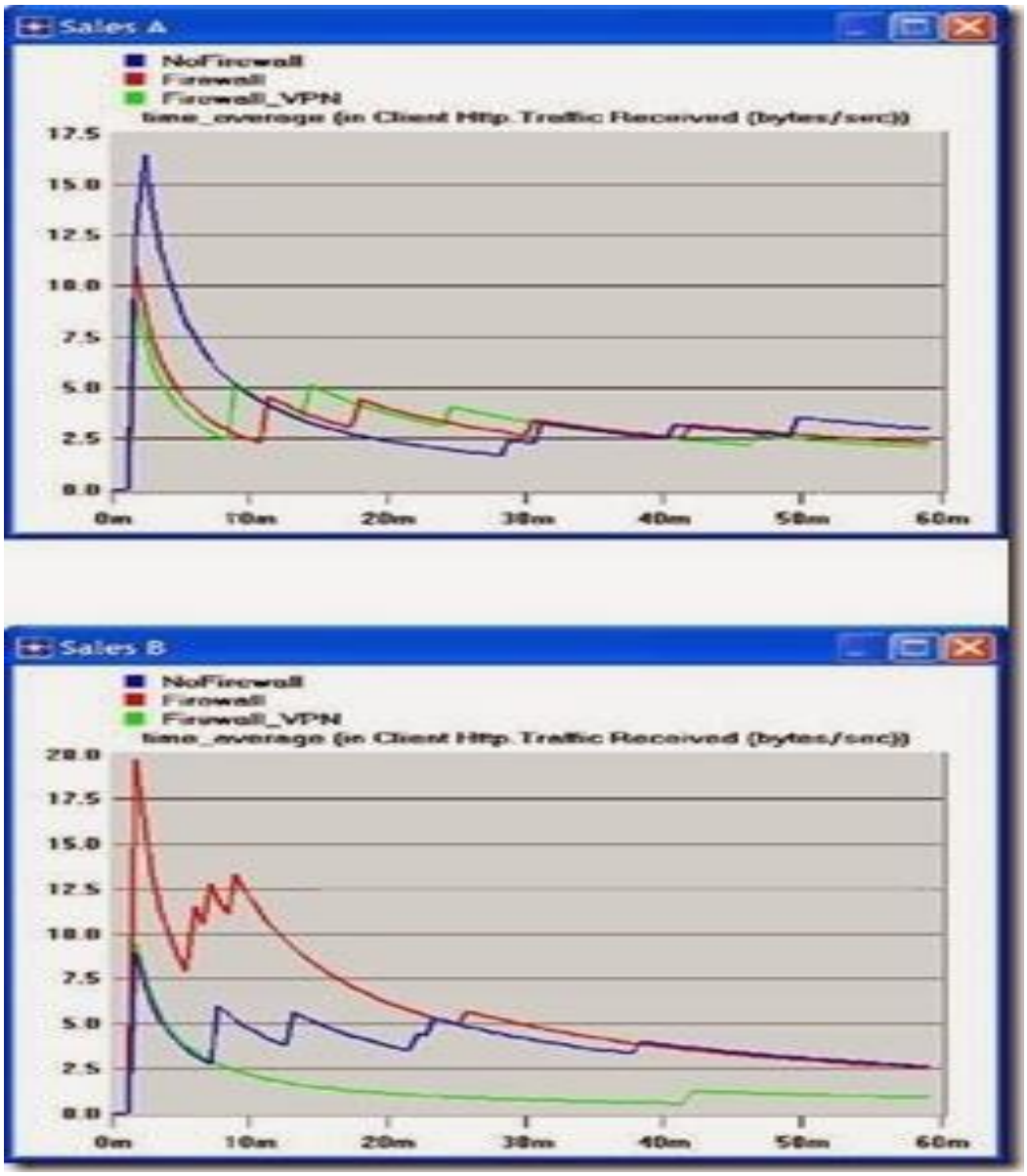


4. Press **Show** and the resulting graph should resemble the following one:

5. Create a graph similar to the previous one, but for Sales B:



1. Create two graphs similar to the previous ones to depict the Traffic Received by the **Client Http** for**Sales A** and **Sales B**.



Note: Results may vary slightly due to different node placemen