Introduction to Cisco IOS

Like any other piece of computer hardware, Cisco switches need some kind of operating system software. Cisco calls this OS the Internetwork Operating System (IOS). Cisco IOS also defines an interface for humans called the ***Command Line Interface*** (CLI) which is also called user interface to the IOS program. The Cisco IOS CLI allows the user to use a terminal emulation program, which accepts text entered by the user. When the user presses Enter, the terminal emulator sends that text to the switch. The switch processes the text as if it is a command, does what the command says. The switch CLI can be accessed through three popular methods—the *console*, *Telnet*, and *Secure Shell* (SSH). Two of these methods (Telnet and SSH) use the IP network in which the switch resides to reach the switch. The console is a physical port built specifically to allow access to the CLI.

After a PC is physically connected to the console port, a terminal emulator software package must be installed and configured on the PC. There are many terminal emulator software one of them is called ***Teraterm***. The terminal emulator software treats all data as text. It accepts the text typed by the user and sends it over the console connection to the switch. Similarly, any bits coming into the PC over the console connection are displayed as text for the user to read. The emulator must be configured to use the PC's serial port to match the settings on the switch's console port settings.

Storing Switch Configuration Files

When you configure a switch, it needs to use the configuration. It also needs to be able to retain the configuration in case the switch loses power. Cisco switches contain random-access memory (RAM) to store data while Cisco IOS is using it, but RAM loses its contents when the switch loses power. To store information that must be retained when the switch loses power, Cisco switches use several types of more permanent memory.

The following list details the four main types of memory found in Cisco switches, as well as the most common use of each type:

1. ***RAM:*** Sometimes called DRAM, for dynamic random-access memory, RAM is used by the switch just as it is used by any other computer: for working storage. The running (active) configuration file is stored here.
2. ***ROM:*** Read-only memory (ROM) stores a bootstrap (or boothelper) program that is loaded when the switch first powers on. This bootstrap program then finds the full Cisco IOS image and manages the process of loading Cisco IOS into RAM, at which point Cisco IOS takes over operation of the switch.
3. ***Flash memory:*** flash memory stores fully functional Cisco IOS images and is the default location where the switch gets its Cisco IOS at boot time. Flash memory also can be used to store any other files, including backup copies of configuration files.
4. ***NVRAM:*** Nonvolatile RAM (NVRAM) stores the initial or startup configuration file that is used when the switch is first powered on and when the switch is reloaded.

Switches use multiple configuration files —one file for the initial configuration used when powering on, and another configuration file for the active, currently used running configuration as stored in RAM. The first type of configuration is called ***startup config*** and the other one is ***running config***. Startup-config is stored in NVRAM whereas running-config is stored in RAM.

Before we start how to configure a Cisco device using a CLI, let us first discuss a software called Packet Tracer which we will be working on as there are no switches to work with. Packet Tracer is a protocol simulator developed by Cisco Systems. Packet Tracer is a simulator software that simulates router, switch and other networking devices. It is a powerful and dynamic tool that displays the various protocols used in networking, in

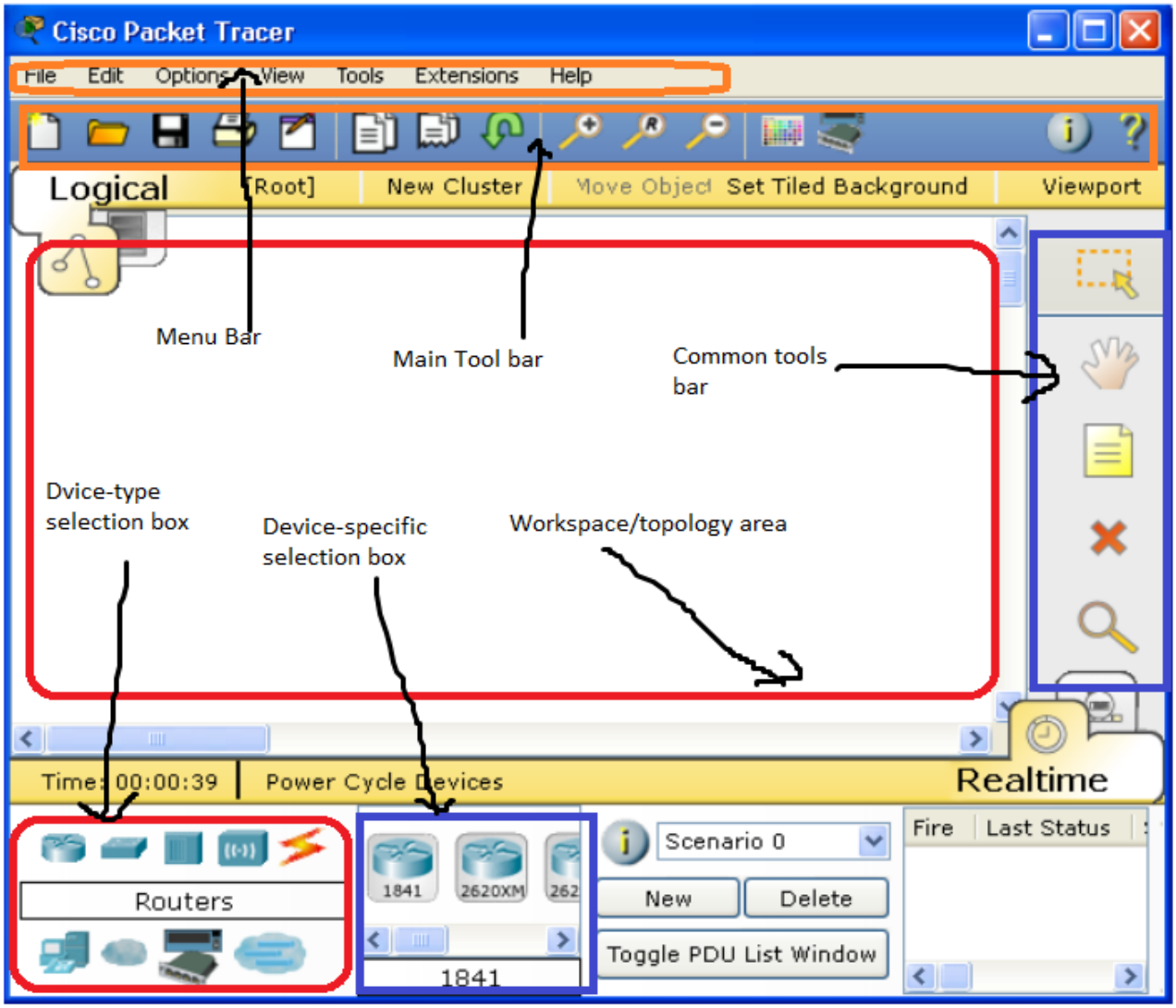
either Real Time or Simulation mode. This includes layer 2 protocols such as Ethernet and PPP, layer 3 protocols such as IP, ICMP, and ARP, and layer 4 protocols such as TCP and UDP. Routing protocols can also be configured.

Lab 1 a): Introduction to Packet Tracer

Objective: The purpose of this lab is to become familiar with the Packet Tracer interface. Learn how to build network topologies of your own.

Activity 1: Identifying the main Packet tracer interface components

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



This initial interface contains components described bellow.

Menu Bar: This bar provides the File, Edit, Options, View, Tools, Extensions, and Help menus. You will find basic commands such as Open, Save, Save as, Print, and Preferences in these menus.

Main Tool Bar: This bar provides shortcut icons to the File and Edit menu commands. This bar also provides buttons for Copy, Paste, Undo, Redo, Zoom, and others.

Common Tools Bar: This bar provides access to these commonly used workspace tools: Select, Move Layout, Place Note, Delete, etc.

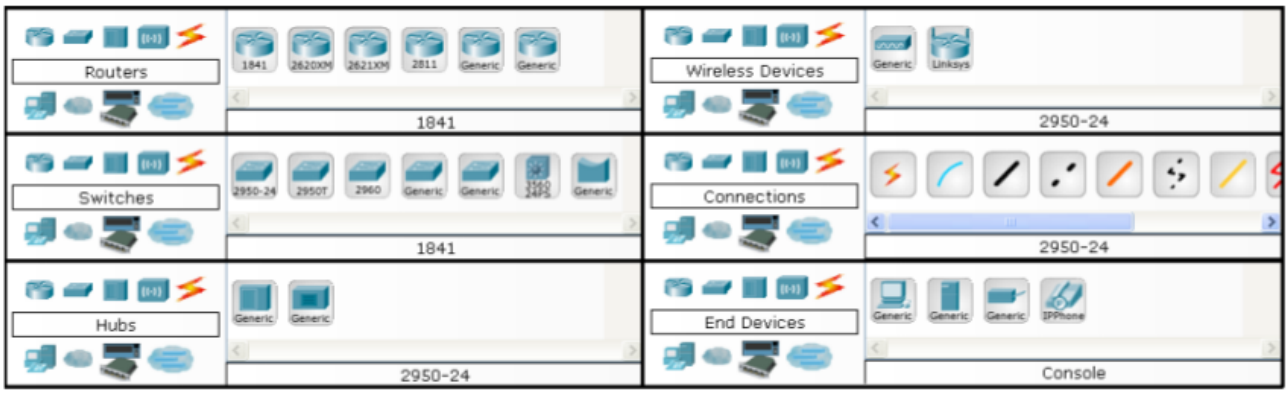
Workspace/topology area: This area is where you will create your network, watch simulations, and view many kinds of information and statistics.

Device-Type Selection Box: This box contains the type of devices and connections available in Packet Tracer. The Device-Specific Selection Box will change depending on which type of device you choose.
Device-Specific Selection Box: This box is where you choose specifically which devices you want to put in your network and which connections to make.

Activity 2: Adding Devices and Connections

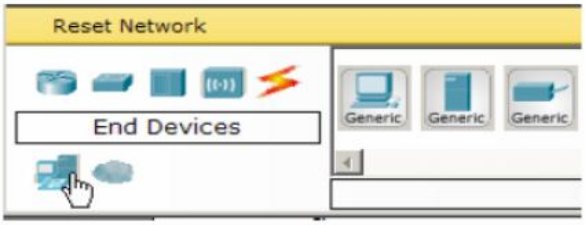
We will begin building our network topology by selecting devices and the media to connect them. Several types of devices and network connections can be used. For this lab, we will keep it simple by using End Devices, Switches, and Connections.

Step 1: Single click on each group of devices and connections to display the various choices.



Step 2: Adding Hosts

Single click on the End Devices.

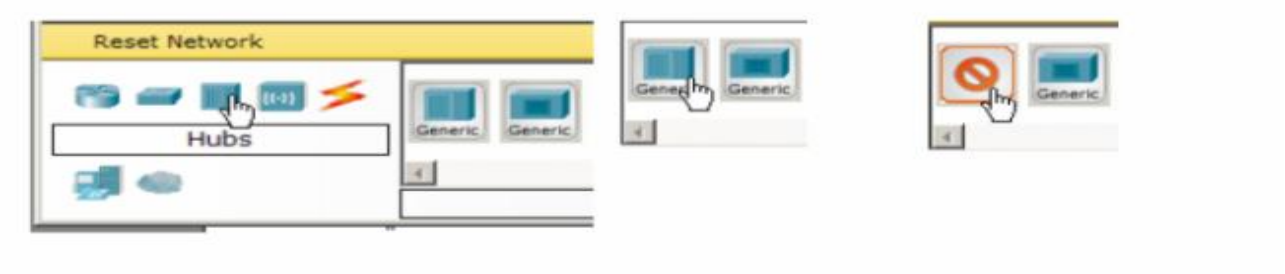


Then single click on the ***Generic*** host.

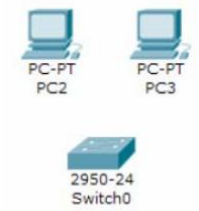
Move the cursor into topology area. You will notice it turns into a plus "+" sign. Single click in the topology area and it the device is placed in the workplace area..

Then repeat step 2 to add three more hosts.

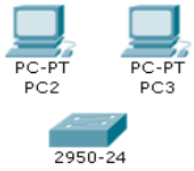
Step 3: Adding a Hub and a switch



Select a hub, by clicking once on ***Hubs*** and once on a ***Generic*** hub.
And move the Generic hub to the working area place it below the generic hosts as shown.



To add a switch follow a similar procedure.
Select a switch, by clicking once on **Switches** from the Device-Type Selection Box and once on a 2950-24 switch.
Add the switch by moving the plus sign "+" below PC2 and PC3 and click once.



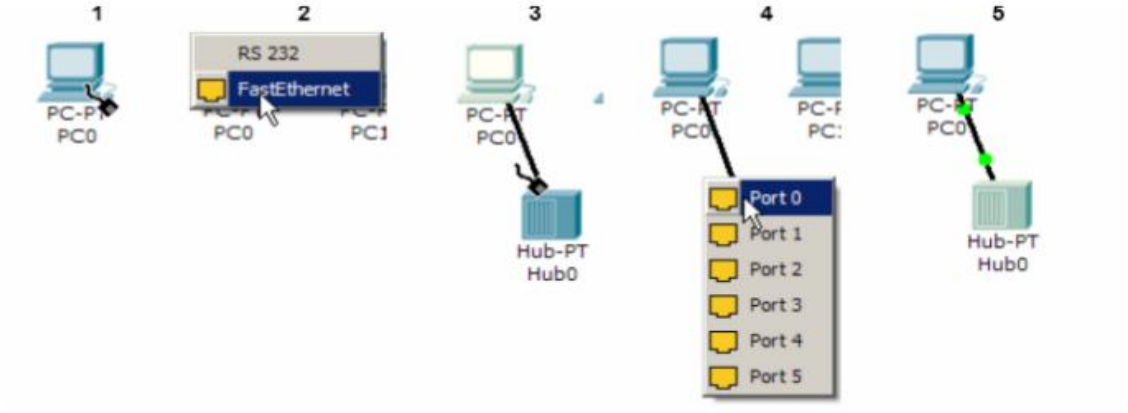
Step 4: adding connections
First Connect PC0 to Hub0 by clicking on the ***connections*** from the Device-Type Selection Box.

Click once on the **Copper Straight-through** cable.

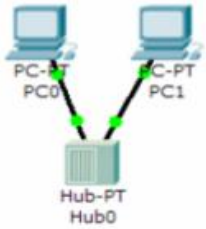
After clicking on the straight-through cable do the following steps to connect PC0 to Hub0, as shown in the following figure.

- Click once on PC0
- Choose FastEthernet (by clicking)
- Drag the cursor to Hub0
- Click once on Hub0 and choose Port 0

By now PC0 and Hub0 are connected and notice the green link lights on both the PC0 Ethernet NIC and the Hub0 Port 0 showing that the link is active.

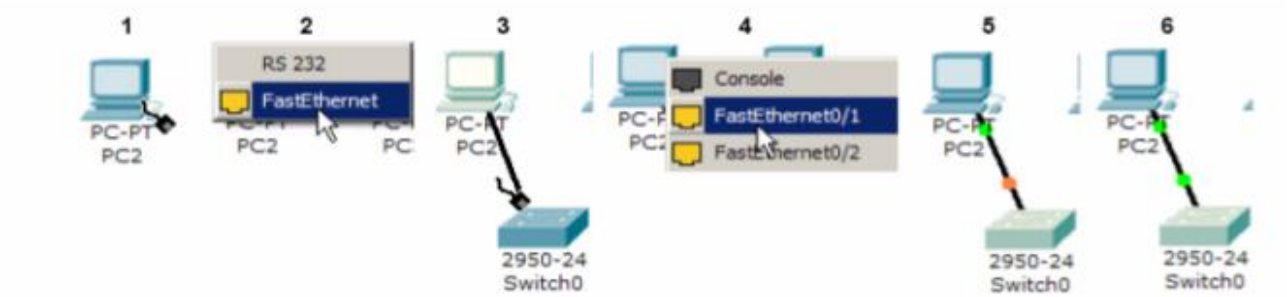


Repeat the steps above for **PC1** connecting it to **Port 1** on **Hub0**. (The actual hub port you choose does not matter.)



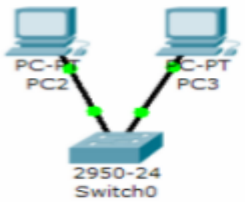
Then now connect PC2 and PC3 to switch0. Follow a similar procedure as connecting PC0 and PC1 to Hub0.
First click on the connections, choose the ***straight-through*** cable and do the following to connect PC2 to switch0.

- Click once on PC2
- Choose FastEthernet
- Drag the cursor to Switch0
- Click once on Switch0 and choose FastEthernet0/1



- Notice the green link lights on PC2 Ethernet NIC and amber light Switch0 FastEthernet0/1 port.

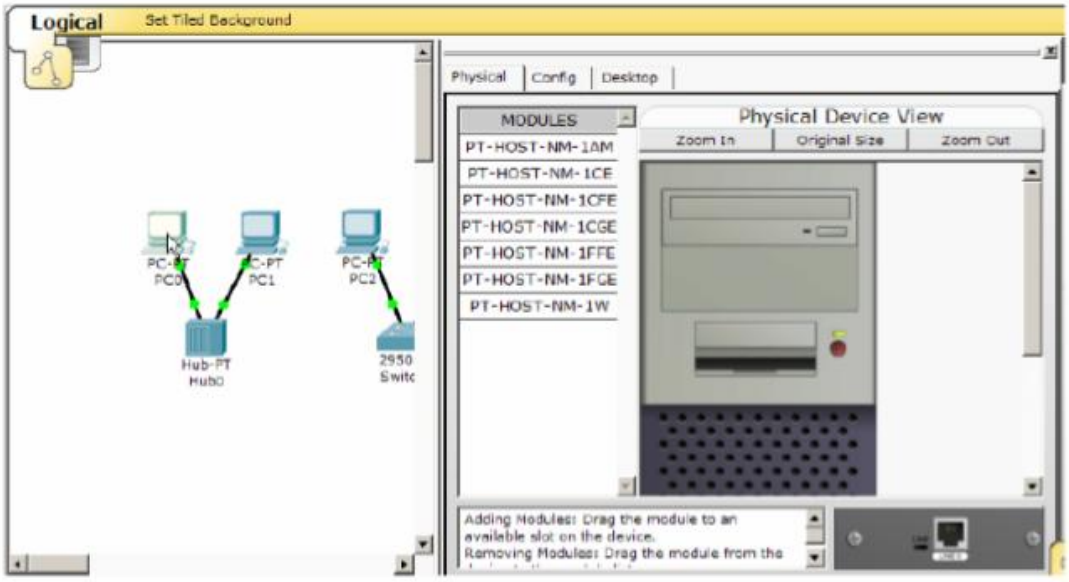
Repeat the steps above for **PC3** connecting it to **Port 3** on **Switch0** on port **FastEtherent0/2**. (The actual switch port you choose does not matter.)

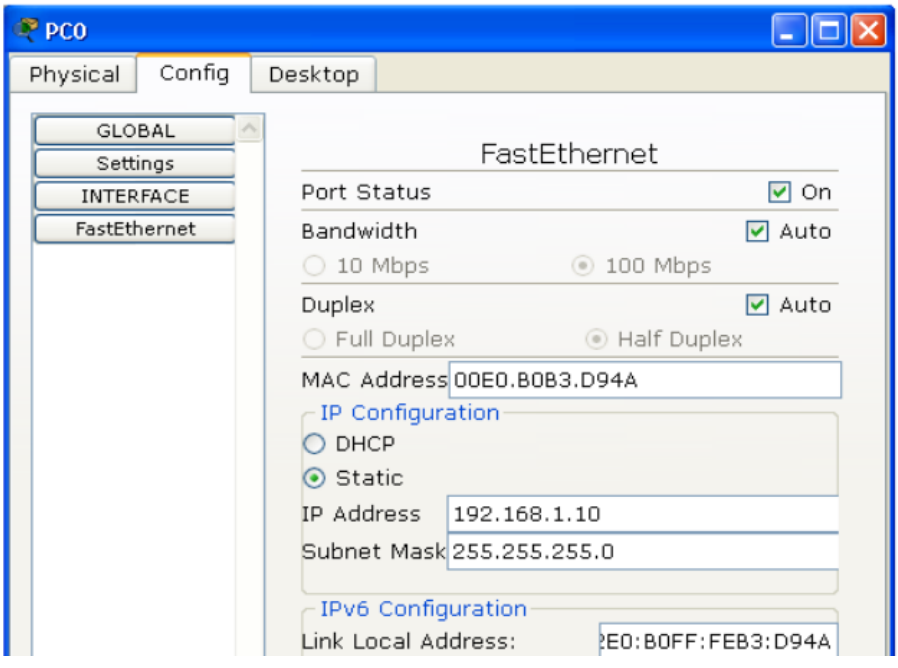


Activity 3: Configuring IP Addresses and Subnet Masks on the Hosts

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

Click once on PC0.
Choose the ***Config tab***. It is here that you can change the name of PC0. It is also here where you would enter a Gateway IP Address, also known as the default gateway. We will discuss this later, but this would be the IP address of the local router. If you want, you can enter the IP Address 172.16.1.1, although it will not be used in this lab.
Click on ***FastEthernet.*** Although we have not yet discussed IP Addresses, add the IP Address to 172.16.1.10. Click once in the Subnet Mask field to enter the default Subnet Mask. You can leave this at 255.255.0.0. We will discuss this later.
N.B: *You may also use the* ***Desktop tab*** *to configure IP address for hosts.*

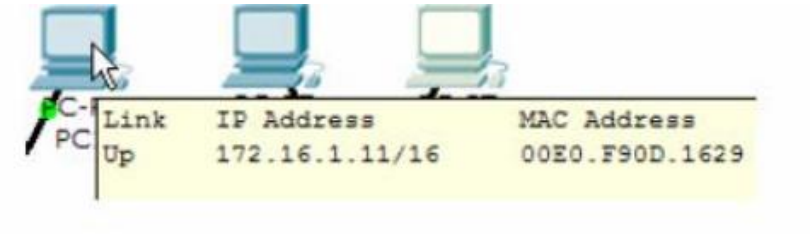




Repeat these steps for the other hosts. Use the information below for IP Addresses and Subnet Masks.

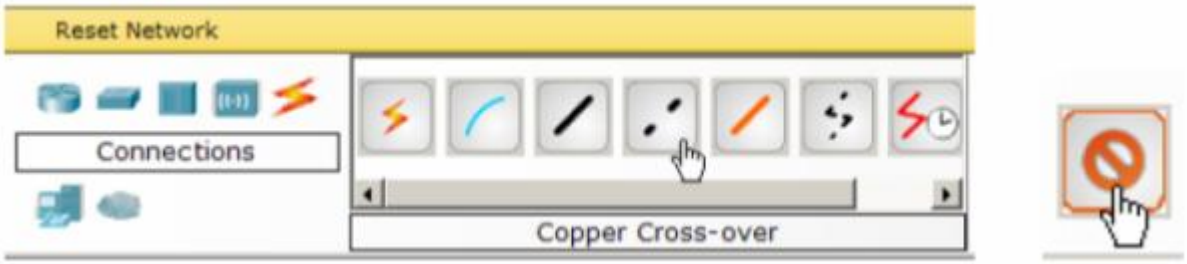
Host	IP Address	Subnet Mask
PC0	172.16.1.10	255.255.0.0
PC1	172.16.1.11	255.255.0.0
PC2	172.16.1.12	255.255.0.0
PC3	172.16.1.13	255.255.0.0

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

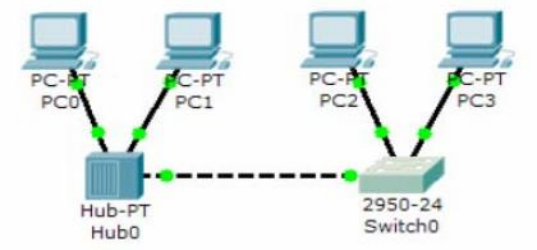


Activity 4: Connecting Hub0 to Switch0

To connect like-devices, like a Hub and a Switch, we will use a ***Cross-over*** cable. Click once the Cross-over Cable from the Connections options.



Move the connections cursor to Hub0 and click ones and select port 5 (it does not matter which port to use in this case). Then move the Connections cursor to Switch0 and then Click once on Switch0 and choose FastEthernet0/4. By now, we have finished building our topology.



Activity 5: Checking connectivity between hosts

Use a ***ping*** program to check for connectivity. Click on PC0 and click on **Desktop tab**, and then click on **Command Prompt**. From the command prompt type **ping 172.16.1.13**. this will check whether there is a connectivity between PC0 and PC3.

```
PC>ping 172.16.1.13
Pinging 172.16.1.13 with 32 bytes of data:
Reply from 172.16.1.13: bytes=32 time=2ms TTL=128
```

```
Reply from 172.16.1.13: bytes=32 time=0ms TTL=128
Reply from 172.16.1.13: bytes=32 time=3ms TTL=128
Reply from 172.16.1.13: bytes=32 time=0ms TTL=128
Ping statistics for 172.16.1.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 1ms
```

Introduction to configuration of Cisco devices using CLI

Quick guide: in this guide, the basic commands which are commonly used are discussed. And you can refer to this guide when necessary.

User EXEC mode*:** When we start a session on switch, we begin in ***user EXEC mode. Prompt for user mode ends in >.

```
Switch>
```

Use ? for help. The ? will list all the commands that can be used from that mode.

Privileged EXEC mode: To have access to all commands, we must be in the privileged mode. Verification of system information and configuration are only possible form this mode. Privileged mode ends in **#**. To enter privileged mode, type enable command from user mode.

```
Switch>enable
Switch#
```

Global configuration mode: used to configure for parameters that apply the entire switch. To enter the global configuration, enter configure terminal command from privileged mode. And the prompt changes to something like <hostname>(config)#

```
Switch#configure terminal
Switch(config)#
```

E.g: configuring a hostname, after we enter the command, the prompt changes

```
Switch(config)#hostname Testswitch
Testswitch(config)#
```

Interface configuration: to configure parameters that affect individual interface, then we enter the *interface configuration mode*. To enter interface configuration mode, enter the following command from global configuration mode. interface <interface_type> <interface_no>

E.g: to configure the first interface of Cisco 2960 switch, use

```
Testswitch(config)#interface fastEthernet 0/1
Testswitch(config-if)#
```

To shut down fast ethernet 0/1, use shutdown command from the interface configuration mode.

```
Testswitch(config)#interface fastEthernet 0/1
Testswitch(config-if)#shutdown
Testswitch(config-if)#
```

Use exit command to quit the existing mode and go to the previous mode.

E.g:
```
Testswitch(config-if)# exit
   Testswitch(config)#
```

Verification: use show commands to verify your configurations or modifications. Use show commands from privileged

```
Testswitch#show running-config  !shows the configuration that resides in RAM
Testswitch#sh startup-config  !displays configuration in NVRAM
Testswitch#sh ip interface brief !displays Interfaces information in brief
Testswitch#sh interfaces fastEthernet 0/2 !displays detailed information about that interface
Testswitch#sh history  !displays recently entered commands
```

Use ? after the show command to see the possible verification commands.

Enable Secret/Password configuration: Enable password and enable secret are used to limit access to privileged mode. Enable secret is more secure than enable password. Better to use enable secret password.

```
Testswitch(config)#enable password cisco   ! cisco is the enable password
Testswitch(config)#enable secret cisco123  !cisco123 is the enable secret
```

Console(line) password: limits access to the user exec mode when we access the device through console.

```
Testswitch(config)#line console 0        !enter line configuration mode.
Testswitch(config-line)#password 123     !setting password value for the console
  Testswitch(config-line)#login          !login command is used to enable the password
```

Saving configuration:unless saved, the running configuration will be lost if the switch boots(or if power goes off)

Testswitch#copy running-config startup-config !or simply use **write** command

Undo a command: use the no command before the command we want to remove. E.g. to undo the command we used to shutdown interface 0/1, use no shutdown

```
Testswitch#
Testswitch#conf t
Testswitch(config)#int fastEthernet 0/1
Testswitch(config-if)#no shutdown
```

Or to remove the hostname, use no hostname command

```
Testswitch(config)#no hostname
Switch(config)#          ! the hostname we configured is removed
```

Assigning IP address to a router's interface: use ip address <ip> <subnet-mask> command from the interface configuration mode after selecting the interface type and number.

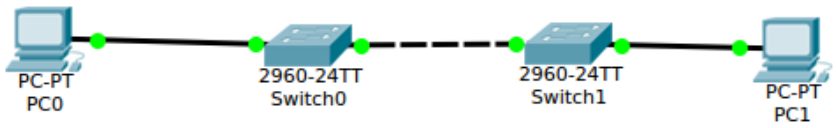
```
Router>en
Router#
Router#conf t
Router(config)#int fa 0/0
Router(config-if)#no shutdown
Router(config-if)#ip address 192.168.1.1 255.255.255.0
```

Lab 1 b): Basic Cisco switch Configuration

Objective: The main objective of this lab is to introduce you to basic configuration of a Cisco switch which includes

- traversing different modes of the IOS
- configuring basic settings like hostname, passwords, and others

Use the following network topology for this lab.



Device	Interface	IP address	Subnet mask
PC0	Fa0	10.10.10.10	255.255.255.0
PC1	Fa0	10.10.10.20	255.255.255.0

Activities:

1. Create the topology shown above by using the correct cabling.
2. Assign the IP addresses for the computers.
3. Configure Switch0
    a. configure the hostname to be **testswitch0**
    b. configure switch0 for enable password of **123** and then logout of switch0 and login again. The switch should ask you for a password after you enter the enable command.
    c. configure Enable secret password of **cisco** and logout from the switch and try to access the CLI again. Which password worked for you this time and why?
    d. configure console password of **cisco123** and logout of the switch and try to login again. Which passwords are required for successful login?
4. Verify the configuration. (look at the contents of the running-config and startup-config)
5. Save your configurations
6. Verify the configuration. (look at the contents of the running-config and startup-config)
7. check the connectivity between the two PCs using ping command.
8. Now shutdown the interface of switch0 connecting to PC0 using a command and check connectivity using ping.
9. You may repeat steps 3 to 7 on switch1 as well.

Configuration on switch0

```
Switch>

Switch>enable
Switch#configure terminal
Switch(config)#hostname testswitch0
testswitch0(config)#
testswitch0(config)#enable password 123
```

Now logout from the command line interface. Enter exit command twice.

```
testswitch0(config)#exit
testswitch0#exit
```

Now login again and the system will ask for enable password which is 123 in this case.

```
testswitch0>enable
Password:
testswitch0#
```

Next, configure the enable secret password on top of the enable password.

```
testswitch0#conf t
testswitch0(config)#enable secret cisco
```

And logout again and login.

```
testswitch0(config)#exit
testswitch0#exit
```

After we enter the enable command, it will ask for password. Now only cisco will work which is the enable secret which has higher priority over enable password.

Configuring console password

```
testswitch0#conf t
testswitch0(config)#line console 0
testswitch0(config-line)#password cisco123
testswitch0(config-line)#login
testswitch0(config-line)#
```

Then logout and login again and the prompt changes. Two passwords will be required. One to access the user mode (cisco123) and the other to access the privileged mode (cisco).

For verification of the configuration use show run and show start from the privileged mode for displaying the running configuration and the start-up configuration respectively.

At this time, the start-up config is not present. Because we haven't save our configuration yet.

Saving the configuration

```
testswitch0#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
testswitch0#
```

Checking the connectivity between the pcs:

The ping request has replies which means the two pcs can reach to each other.

```
pc:\>ping 10.10.10.20

Pinging 10.10.10.20 with 32 bytes of data:
Reply from 10.10.10.20: bytes=32 time<1ms TTL=128
Reply from 10.10.10.20: bytes=32 time<1ms TTL=128
Reply from 10.10.10.20: bytes=32 time<1ms TTL=128
Reply from 10.10.10.20: bytes=32 time<1ms TTL=128
Ping statistics for 10.10.10.20:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
pc:\>
```

To shutdown the interface connected to pc0, first identify the interface number and go to the interface configuration mode and use the shutdown command. The command will produce with a link status that the interface is down.

```
testswitch0#conf t
testswitch0(config)#interface fa 0/1
testswitch0(config-if)#shutdown
testswitch0(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to administratively down
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to
down
```

Now the first interface of switch0 is shut down and pc0 and pc1 can no longer be able to communicate until we enable it.

You can also verify the status of the interface using the sh ip int br command. From the output of the command, we can see that that interface is down by the administrator.

```
testswitch0#sh ip int br
Interface        IP-Address      OK?     Method     Status                Protocol
FastEthernet0/1    unassigned    YES     manual   administratively down    down
FastEthernet0/2   unassigned     YES     manual       up                    up
FastEthernet0/3   unassigned     YES     manual       down                  down
FastEthernet0/4   unassigned     YES     manual       down                  down
FastEthernet0/5    unassigned    YES     manual       down                  down
```