Practical 6

AIM: Packet Analysis in network using Wireshark Tool.

1.1. What is Wireshark?

Wireshark is a network packet analyzer. A network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible.
You could think of a network packet analyzer as a measuring device used to examine what's going on inside a network cable, just like a voltmeter is used by an electrician to examine what's going on inside an electric cable (but at a higher level, of course).
In the past, such tools were either very expensive, proprietary, or both. However, with the advent of Wireshark, all that has changed.

Wireshark is perhaps one of the best open source packet analyzers available today.

1.1.1 Some intended purpose

Here is some examples people use Wireshark for:

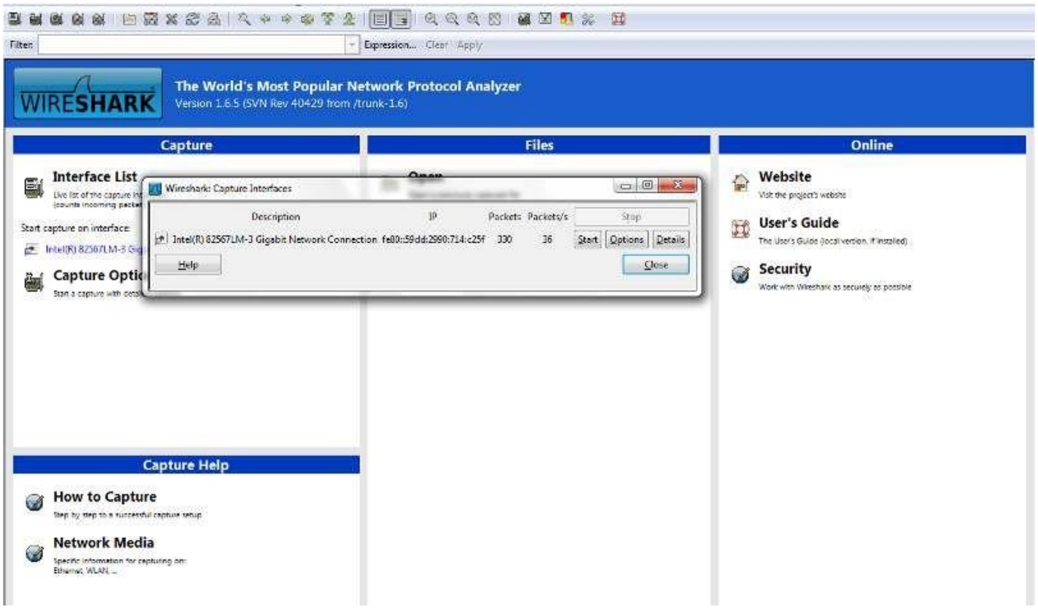
- Network administrators use it to *troubleshoot network problems*
- Network security engineers use it to *examine security problems*
- Developers use it to *debug protocol implementations*
- People use it to *learn network protocol* internals

Beside these examples Wireshark can be helpful in many other situations too.

1.1.2. Features

The following are some of the many features Wireshark provides:

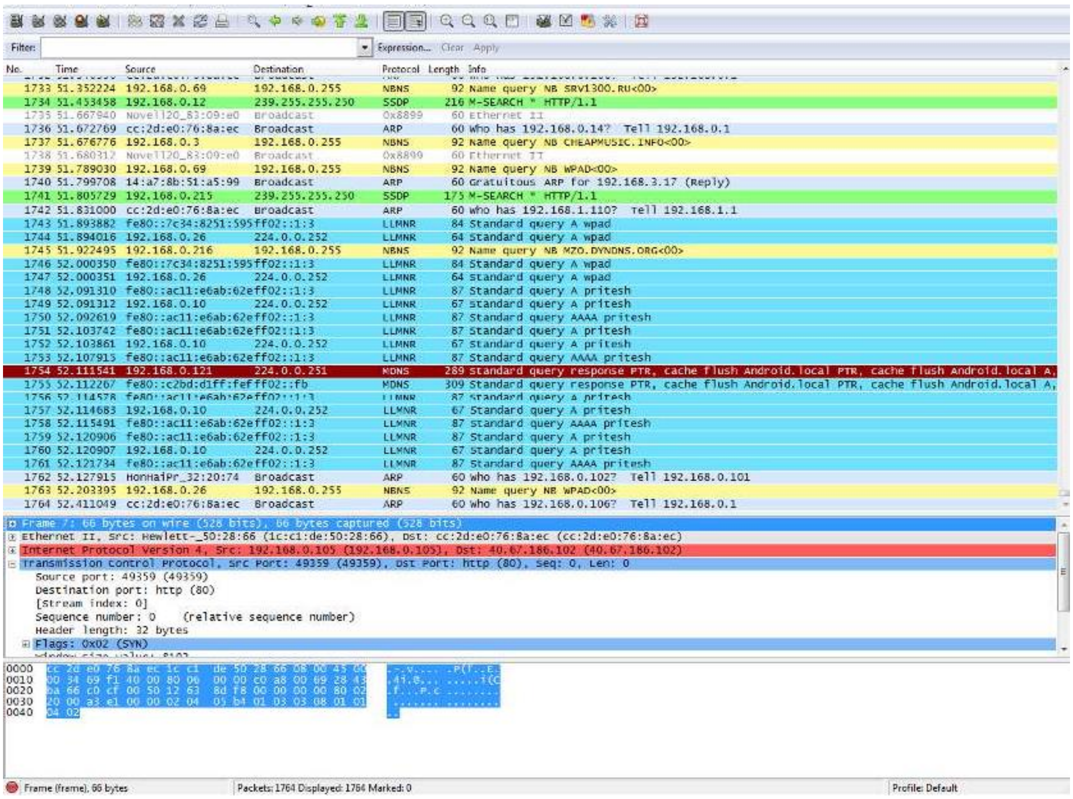
- Available for *UNIX* and *Windows*.
- *Capture* live packet data from a network interface.
- *Open* files containing packet data captured with tcpdump/WinDump, Wireshark, and a number of other packet capture programs.
- *Import* packets from text files containing hex dumps of packet data.
- Display packets with *very detailed protocol information*.
- *Save* packet data captured.
- *Export* some or all packets in a number of capture file formats.
- *Filter packets* on many criteria.
- *Search* for packets on many criteria.
- *Colorize* packet display based on filter


Filter:
Expression... Clear Apply
WIRESHARK
The World's Most Popular Network Protocol Analyzer
Version 1.6.5 (SVN Rev 40429 from /trunk-1.6)
Capture
Files
Online
Interface List
Start capture on interface:
Intel(R) 82567LM-3 Gi
Capture Optio
Website
Visit the project's website
User's Guide
The User's Guide (local version, if installed)
Security
Work with Wireshark as securely as possible
Capture Help
How to Capture
Step by step to a successful capture setup
Network Media
Specific information for capturing on:
Ethernet, WLAN, ...
Wireshark: Capture Interfaces
Stop
Start
Options
Details
Help
Close

```
Description                                  IP                         Packets  Packets/s
Intel(R) 82567LM-3 Gigabit Network Connection fe80::59dd:2990:714:c25f   330      36
```

1.1.2. Live capture from many different network media

Wireshark can capture traffic from many different network media types - and despite its name - including wireless LAN as well. Which media types are supported, depends on many things like the operating system you are using. An overview of the supported media types can be found at https://wiki.wireshark.org/CaptureSetup/NetworkMedia



```
No.   Time      Source               Destination      Protocol Length Info
 1733 51.352224 192.168.0.69         192.168.0.255    NBNS      92 Name query NB SRV1300.RU<00>
 1734 51.453458 192.168.0.12         239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1
 1735 51.667940 Novell20_83:09:e0    Broadcast        0x8899    60 Ethernet II
 1736 51.672769 cc:2d:e0:76:8a:ec    Broadcast        ARP       60 Who has 192.168.0.14?  Tell 192.168.0.1
 1737 51.676776 192.168.0.3          192.168.0.255    NBNS      92 Name query NB CHEAPMUSIC.INFO<00>
 1738 51.680312 Novell20_83:09:e0    Broadcast        0x8899    60 Ethernet II
 1739 51.789030 192.168.0.69         192.168.0.255    NBNS      92 Name query NB WPAD<00>
 1740 51.799708 14:a7:8b:51:a5:99    Broadcast        ARP       60 Gratuitous ARP for 192.168.3.17 (Reply)
 1741 51.805729 192.168.0.215        239.255.255.250  SSDP     175 M-SEARCH * HTTP/1.1
 1742 51.831000 cc:2d:e0:76:8a:ec    Broadcast        ARP       60 Who has 192.168.1.110?  Tell 192.168.1.1
 1743 51.893882 fe80::7c34:8251:595ff02::1:3           LLMNR     84 Standard query A wpad
 1744 51.894016 192.168.0.26         224.0.0.252      LLMNR     64 Standard query A wpad
 1745 51.922495 192.168.0.216        192.168.0.255    NBNS      92 Name query NB MZO.DYNDNS.ORG<00>
 1746 52.000350 fe80::7c34:8251:595ff02::1:3           LLMNR     84 Standard query A wpad
 1747 52.000351 192.168.0.26         224.0.0.252      LLMNR     64 Standard query A wpad
 1748 52.091310 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query A pritesh
 1749 52.091312 192.168.0.10         224.0.0.252      LLMNR     67 Standard query A pritesh
 1750 52.092619 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query AAAA pritesh
 1751 52.103742 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query A pritesh
 1752 52.103861 192.168.0.10         224.0.0.252      LLMNR     67 Standard query A pritesh
 1753 52.107915 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query AAAA pritesh
 1754 52.111541 192.168.0.121        224.0.0.251      MDNS     289 Standard query response PTR, cache flush Android.local PTR, cache flush Android.local A,
 1755 52.112267 fe80::c2bd:d1ff:feff02::fb            MDNS     309 Standard query response PTR, cache flush Android.local PTR, cache flush Android.local A,
 1756 52.114578 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query A pritesh
 1757 52.114683 192.168.0.10         224.0.0.252      LLMNR     67 Standard query A pritesh
 1758 52.115491 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query AAAA pritesh
 1759 52.120906 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query A pritesh
 1760 52.120907 192.168.0.10         224.0.0.252      LLMNR     67 Standard query A pritesh
 1761 52.121734 fe80::ac11:e6ab:62eff02::1:3           LLMNR     87 Standard query AAAA pritesh
 1762 52.127915 HonHaiPr_32:20:74    Broadcast        ARP       60 Who has 192.168.0.102?  Tell 192.168.0.101
 1763 52.203395 192.168.0.26         192.168.0.255    NBNS      92 Name query NB WPAD<00>
 1764 52.411049 cc:2d:e0:76:8a:ec    Broadcast        ARP       60 Who has 192.168.0.106?  Tell 192.168.0.1
```

```
⊞ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
⊞ Ethernet II, Src: Hewlett-_50:28:66 (1c:c1:de:50:28:66), Dst: cc:2d:e0:76:8a:ec (cc:2d:e0:76:8a:ec)
⊞ Internet Protocol Version 4, Src: 192.168.0.105 (192.168.0.105), Dst: 40.67.186.102 (40.67.186.102)
⊟ Transmission Control Protocol, Src Port: 49359 (49359), Dst Port: http (80), Seq: 0, Len: 0
    Source port: 49359 (49359)
    Destination port: http (80)
    [Stream index: 0]
    Sequence number: 0    (relative sequence number)
    Header length: 32 bytes
  ⊞ Flags: 0x02 (SYN)
```

```
0000  cc 2d e0 76 8a ec 1c c1  de 50 28 66 08 00 45 00   .-.v.... .P(f..E.
0010  00 34 69 f1 40 00 80 06  00 00 c0 a8 00 69 28 43   .4i.@... .....i(C
0020  ba 66 c0 cf 00 50 12 63  8d f8 00 00 00 00 80 02   .f...P.c ........
0030  20 00 a3 e1 00 00 02 04  05 b4 01 03 03 08 01 01    ....... ........
0040  04 02                                              ..
```

```
Frame (frame), 66 bytes     Packets: 1764 Displayed: 1764 Marked: 0     Profile: Default
```

.

1.1.3. Import files from many other capture programs

Wireshark can open packets captured from a large number of other capture programs. For a list of input formats see Section 5.2.2, “Input File Formats”.

1.1.2. Export files for many other capture programs

Wireshark can save packets captured in a large number of formats of other capture programs. For a list of output formats see Section 5.3.2, “Output File Formats”.

1.1.3. Many protocol dissectors

There are protocol dissectors (or decoders, as they are known in other products) for a great many protocols: see Appendix C, *Protocols and Protocol Fields*.

1.1.4. Open Source Software

Wireshark is an open source software project, and is released under the GNU General Public License (GPL). You can freely use Wireshark on any number of computers you like, without worrying about license keys or fees or such. In addition, all source code is freely available under the GPL. Because of that, it is very easy for people to add new protocols to Wireshark, either as plugins, or built into the source, and they often do!

1.1.5. What Wireshark is not

Here are some things Wireshark does not provide:

- Wireshark isn’t an intrusion detection system. It will not warn you when someone does strange things on your network that he/she isn’t allowed to do. However, if strange things happen, Wireshark might help you figure out what is really going on.

- Wireshark will not manipulate things on the network, it will only "measure" things from it. Wireshark doesn’t send packets on the network or do other active things (except for name resolutions, but even that can be disabled).

CONCLUSION

From this practical, we can learn about capturing of packets and analysis using wireshark tool. It gives details about packet information, length, protocol, data and frames.

Signature with date:

____

Practical 7

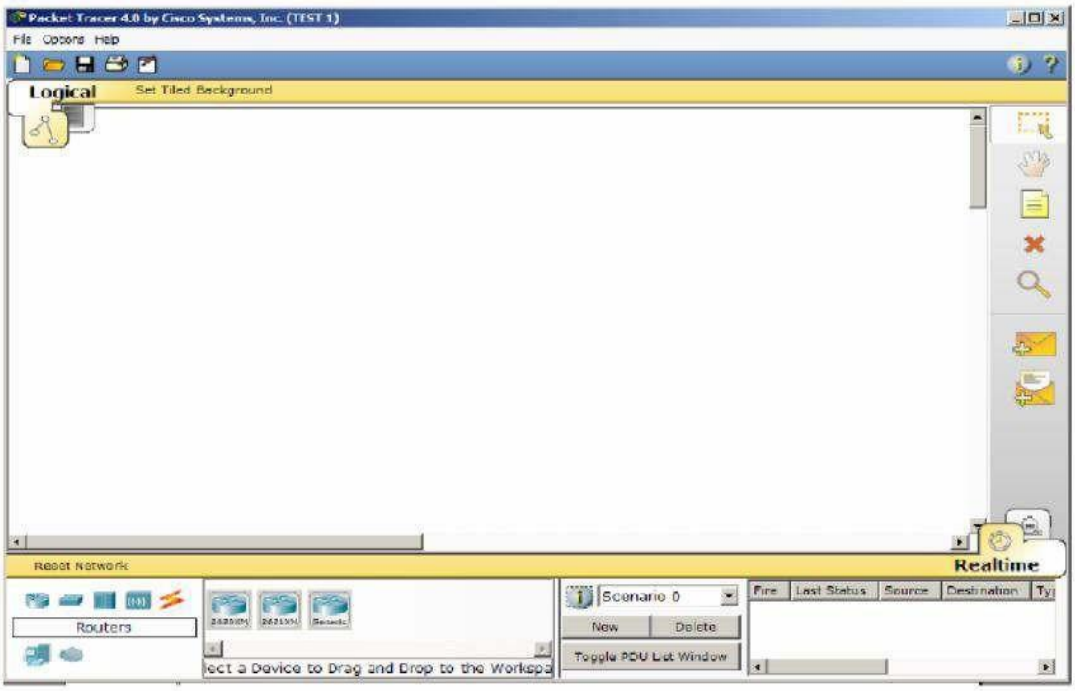
Aim : Introduction to Cisco packet tracer:

Packet Tracer:

Packet Tracer is a protocol simulator developed by Dennis Frezzo and his team at Cisco Systems. Packet Tracer (PT) is a powerful and dynamic tool that displays the various protocols used in networking, in either Real Time or Simulation mode. 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 traced.

Step 1: Start Packet Tracer and Entering

Simulation Mode:

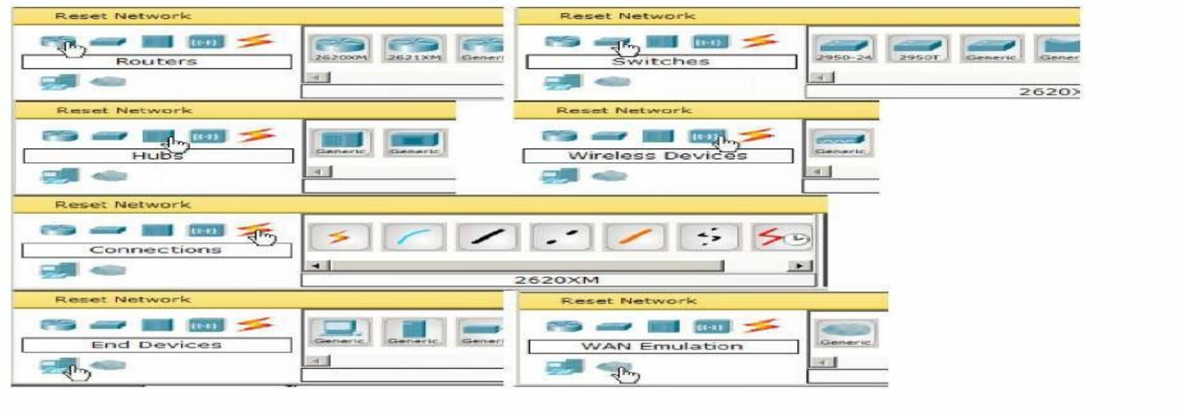


Step 2: Choosing Devices and Connections

We will begin building our network topology by selecting devices and the media in which to connect them.

Several types of devices and network connections can be used. For this lab we will keep it simple by using End Devices, Switches, Hubs, and Connections.

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

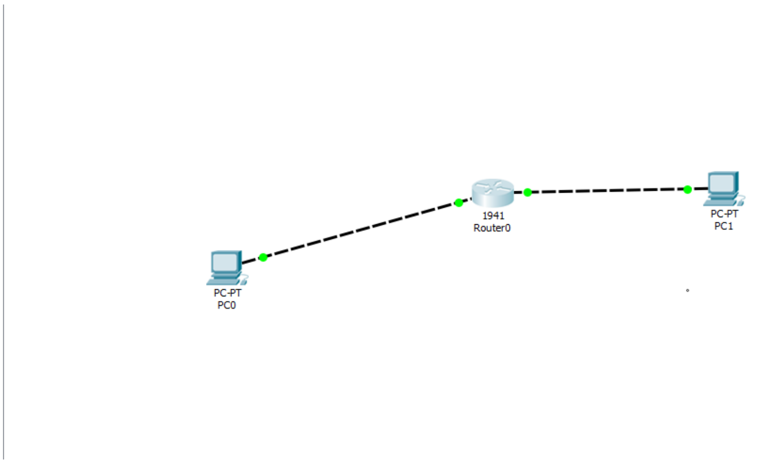


Step 3: Choosing Devices and Connections

We have to select the two “Generic PC” from device section & the one “Generic Router” from router section

Then, make connection between the Generic pc to router using “Crossover cable”

Now, Develop the topology as define in the below figure.

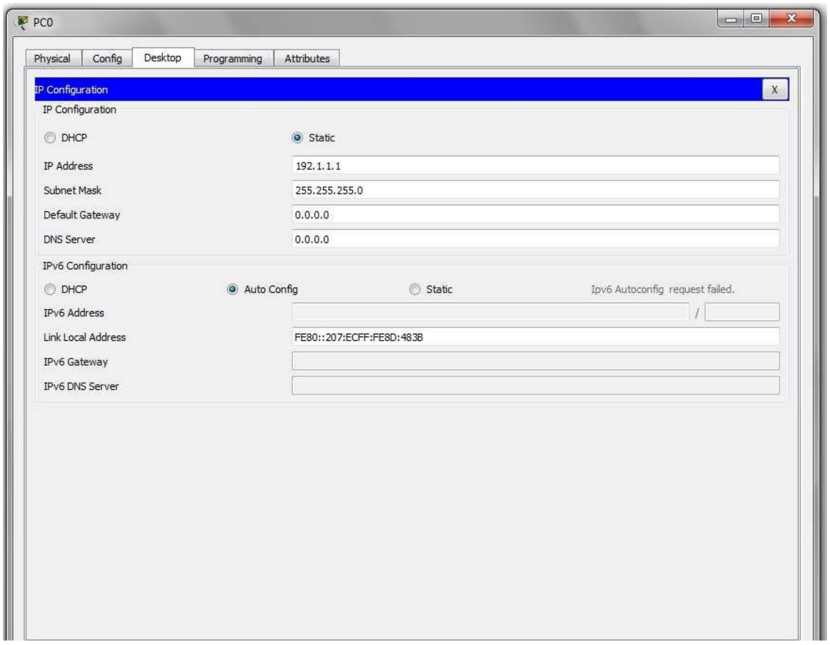


Step 4: Assigning the IP Address to each generic PCs

After Designing the topology, we have to assigning the IP Address to the PCs

For that we have to go to the PC0 -> Desktop -> Ip configuration -> IP Address then assign the “class C” Ip Address to the PCs Follow same step to other PC1 also.

The Simple shown in below figure.



IP Configuration

○ DHCP ● Static

IP Address: 192.1.1.1

Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

IPv6 Configuration

○ DHCP ● Auto Config ○ Static Ipv6 Autoconfig request failed.

IPv6 Address: /

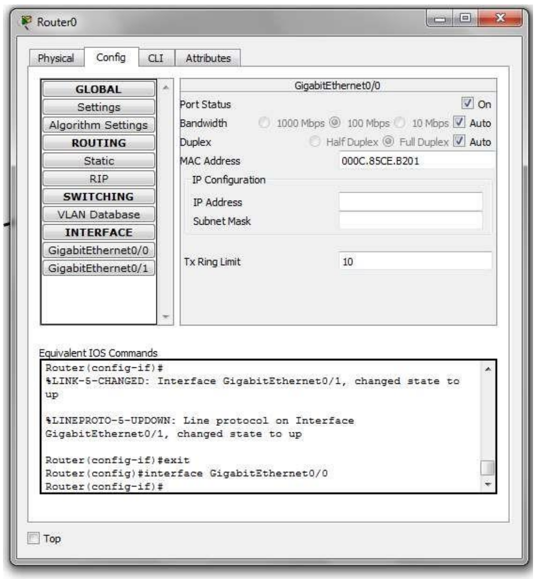
Link Local Address: FE80::207:ECFF:FE8D:483B

IPv6 Gateway:

IPv6 DNS Server:

Step 5: Activate the connection in the router

In this step we have to turn on the connection in the router properties. For the we want to go into Router0 -> Config -> FastEthernet0/0 or FastEthernet0/1 -> Port Status -> tick as ON



GigabitEthernet0/0

Port Status ☑ On

Bandwidth ○ 1000 Mbps ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

MAC Address: 000C.85CE.B201

IP Configuration

IP Address:

Subnet Mask:

Tx Ring Limit: 10

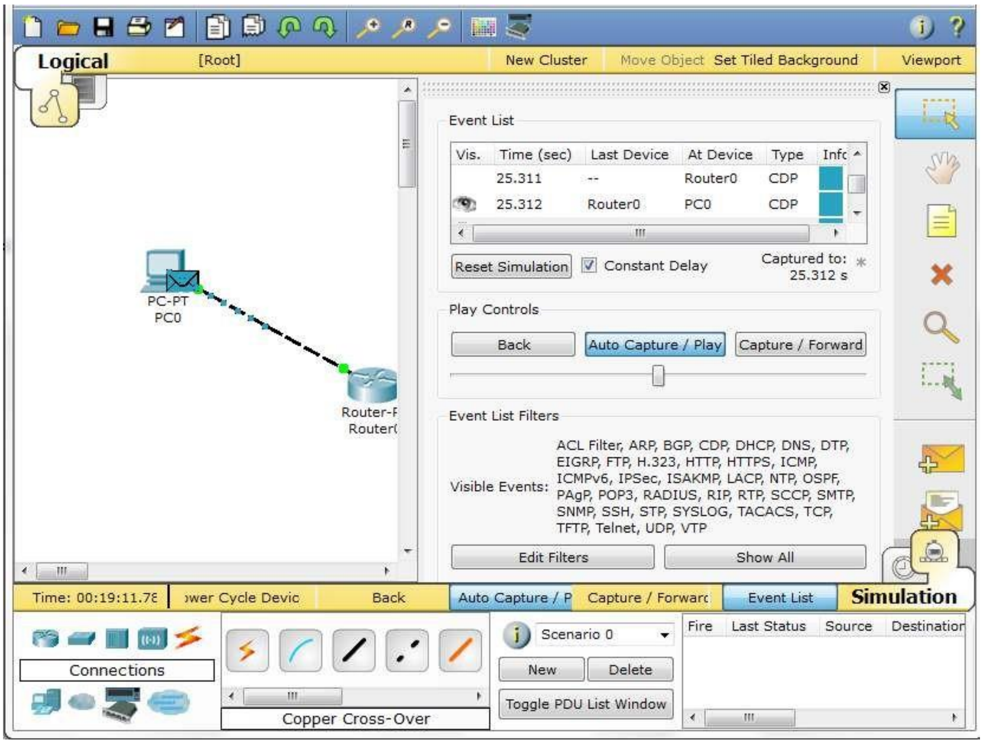
Equivalent IOS Commands

```
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to
up

%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/1, changed state to up

Router(config-if)#exit
Router(config)#interface GigabitEthernet0/0
Router(config-if)#
```

Step 6:Simulation of the connection

In this we are going to send the massage from Generic PC1 To Generic PC0 via the Generic Router0

The Result of the simulation is shown in the below figure.



Vis.	Time (sec)	Last Device	At Device	Type	Infc
	25.311	--	Router0	CDP	🟦
	25.312	Router0	PC0	CDP	🟦

☑ Constant Delay

Captured to: 25.312 s

Visible Events: ACL Filter, ARP, BGP, CDP, DHCP, DNS, DTP, EIGRP, FTP, H.323, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NTP, OSPF, PAgP, POP3, RADIUS, RIP, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP

Time: 00:19:11.7ɛ

Scenario 0

Fire	Last Status	Source	Destination

CONCLUSION

In this practical, we learn about Cisco packet tracer basics and configuration of pc using router.

Signature with date:

____

Practical 8

TITLE: Implementation of star topology using switch and Hub in Packet Tracer.

OBJECTIVES:

After completing study of this practical student will be familiarized with…

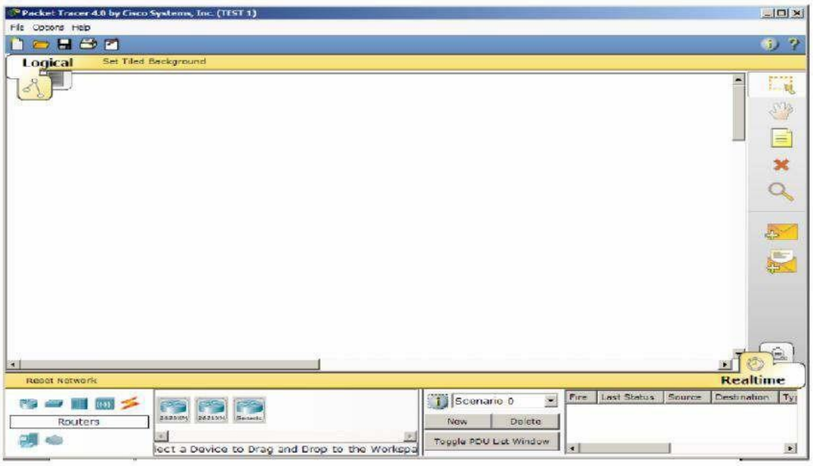
- ➢ Concept of Packet Tracer Simulator.
- ➢ Concept of Basic N/W.

THEORY:

Packet Tracer:

Packet Tracer is a protocol simulator developed by Dennis Frezzo and his team at Cisco Systems. Packet Tracer (PT) is a powerful and dynamic tool that displays the various protocols used in networking, in either Real Time or Simulation mode. 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 traced.

Step 1: Start Packet Tracer and Entering Simulation Mode



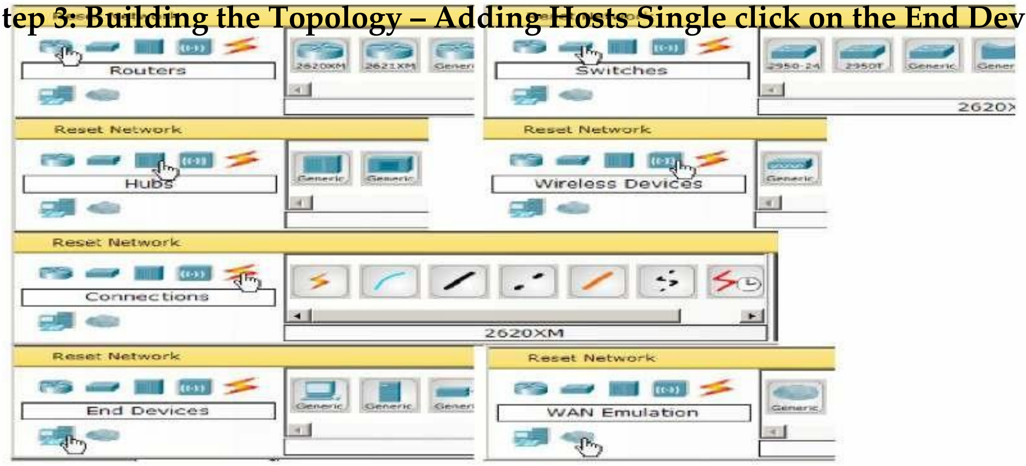
Step 2: Choosing Devices and Connections

We will begin building our network topology by selecting devices and the media in which to connect them.

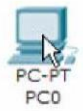
Several types of devices and network connections can be used. For this lab we will keep it simple by using End Devices, Switches, Hubs, and Connections.

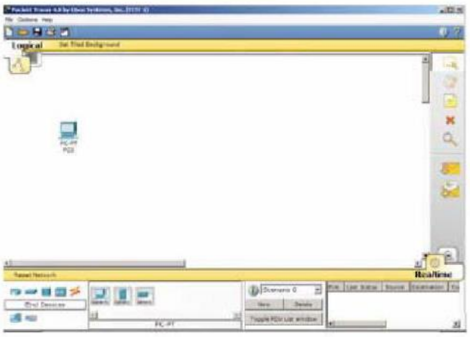
Single click on each group of devices and connections to display the various choices.

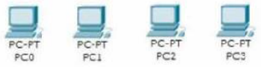
Step 3: Building the Topology – Adding Hosts Single click on the End Devices.



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





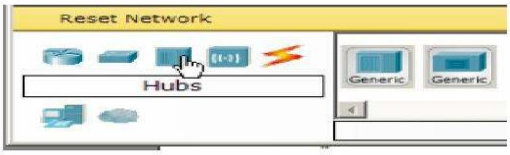


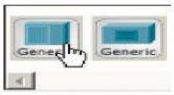
Add three more hosts.

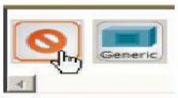
Step 4: Building the Topology – Connecting the Hosts to Hubs and Switches

<u>Adding a Hub</u>

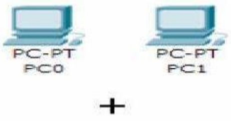
Select a hub, by clicking once on Hubs and once on a Generic hub.

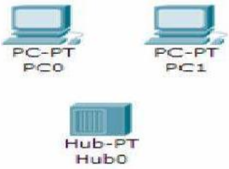




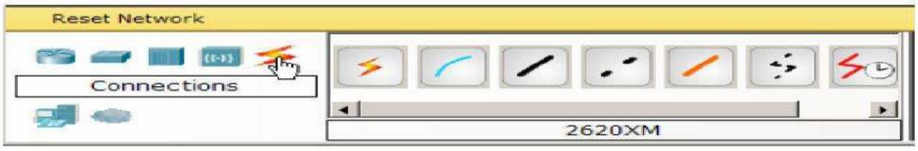


Add the hub by moving the plus sign "+" below PC0 and PC1 and click once.

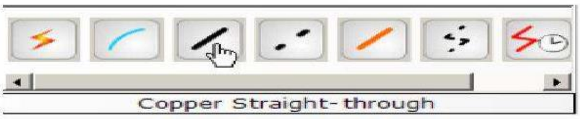




Connect PC0 to Hub0 by first choosing **Connections.**

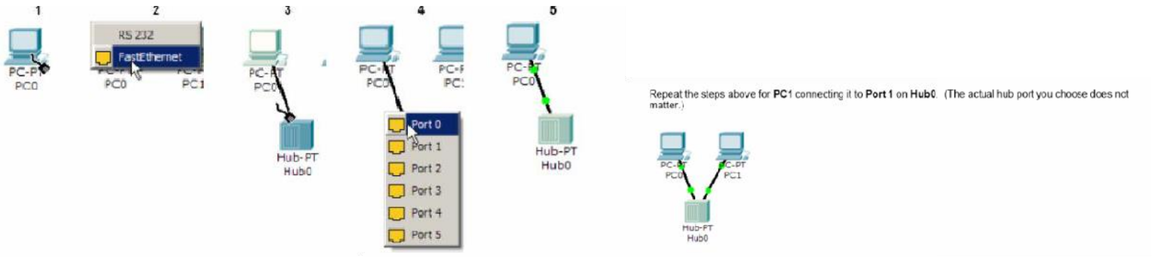


Click once on the **Copper Straight-through** cable.



Perform the following steps to connect PC0 to Hub0:

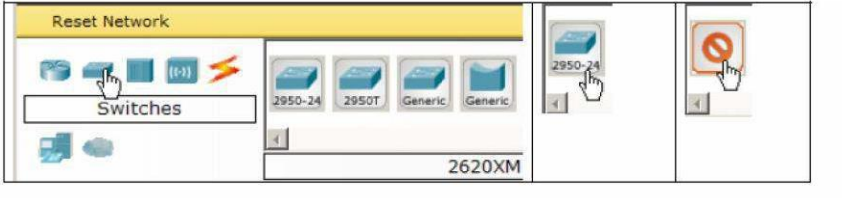
1. Click once on PC0
2. Choose FastEthernet
3. Drag the cursor to Hub0
4. Click once on Hub0 and choose Port 0
5. Notice the green link lights on both the PC0 Ethernet NIC and the Hub0 Port 0 showing - that the link is active.



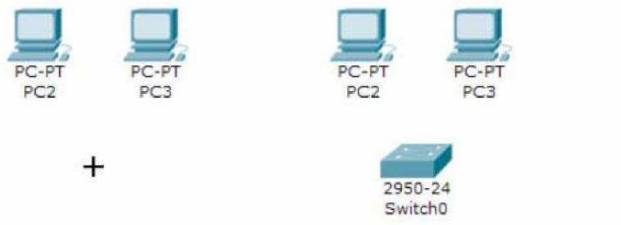
Repeat the steps above for **PC1** connecting it to **Port 1** on **Hub0**. (The actual hub port you choose does not matter.)

Adding a Switch

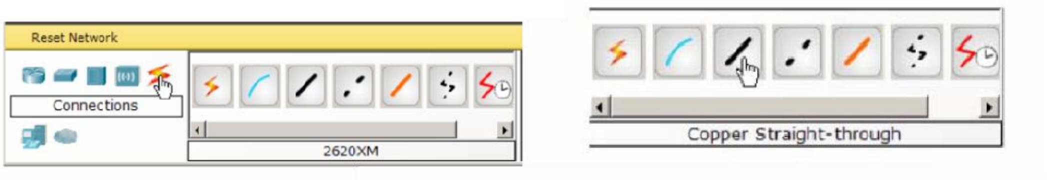
Select a switch, by clicking once on **Switches** and once on a **2950-24** switch.

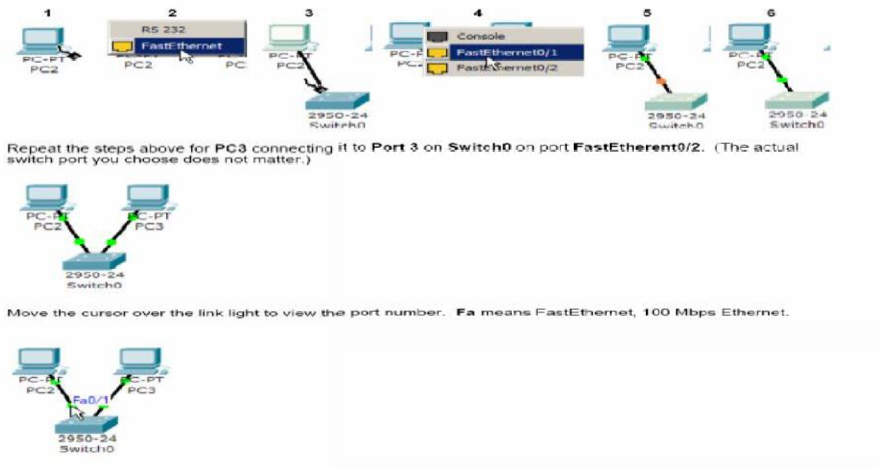


Add the switch by moving the plus sign "+" below PC2 and PC3 and click once.



Connect PC2 to Switch0 by first choosing **Connections.** Click once on the **Copper Straight-through** cable.





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

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

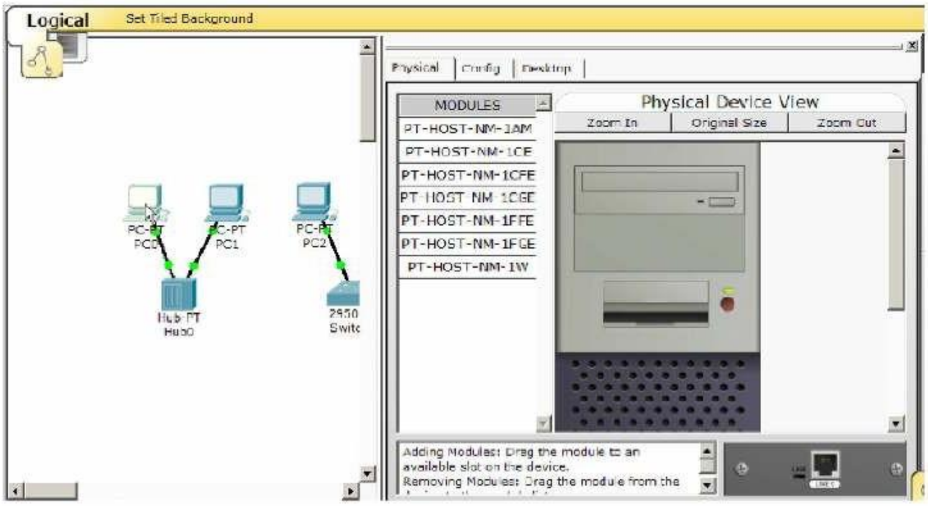
Perform the following steps to connect PC2 to Switch0:

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

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

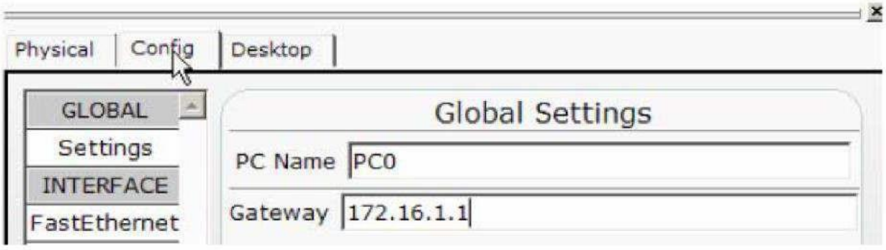
Before we can communicate between the hosts we need to configure IP Addresses and Subnet Masks on the devices.

Click once on PC0.



Adding Modules: Drag the module to an available slot on the device.
Removing Modules: Drag the module from the

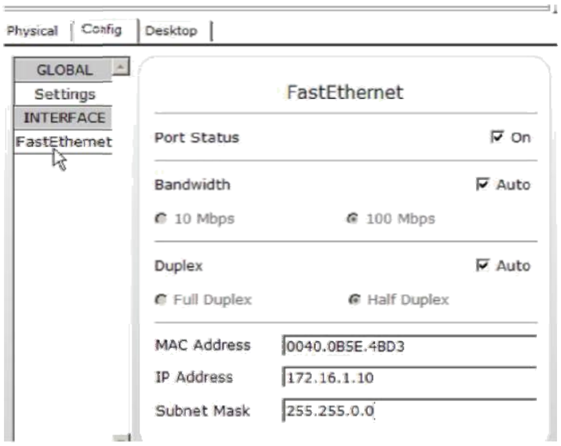
Choose the Config tab. It is here that you can change the name of PC0. It is also here whereyou would enter a Gateway IP Address, also known as the default gateway. We will discuss this later, but this would be the IP address of the local router. If you want, you can enter the IP Address 172.16.1.1,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



PC Name PC0

Gateway 172.16.1.1

Click on FastEthernet. Although we have not yet discussed IP Addresses, add the IPAddress 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.



FastEthernet

Port Status ☑ On

Bandwidth ☑ Auto

○ 10 Mbps ● 100 Mbps

Duplex ☑ Auto

○ Full Duplex ● Half Duplex

MAC Address 0040.0B5E.4BD3

IP Address 172.16.1.10

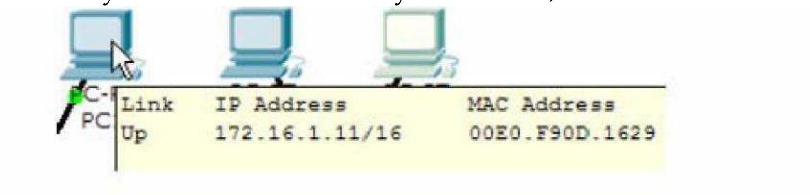
Subnet Mask 255.255.0.0

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.



```
Link   IP Address          MAC Address
Up     172.16.1.11/16      00E0.F90D.1629
```

Also verify the connection by ping command.

CONCLUSION

In this practical, we learn about implementation of star topology using hub and switch in Cisco packet tracer.

Signature with date:

____

Practical 9

TITLE: Implementation of Dynamic Routing (Using RIP)

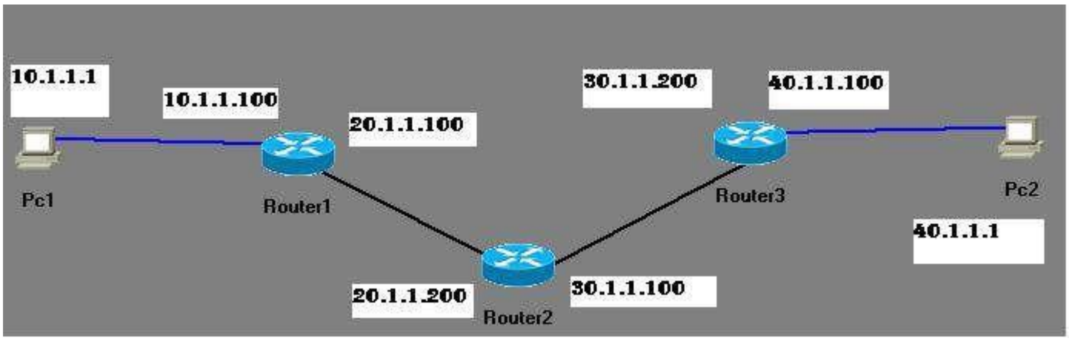
Routing protocols can be classified into different groups according to their characteristics. Specifically, routing protocols can be classified by their:

- **Purpose**: Interior Gateway Protocol (IGP) or Exterior Gateway Protocol (EGP)
- **Operation**: Distance vector protocol, link-state protocol, or path-vector protocol
- **Behavior**: Classful (legacy) or classless protocol

	RIPv1	**RIPv2**
Interior/ Exterior	Interior	Interior
Type	Distance vector	Distance vector
Speed of Convergence	Slow	Slow
Scalability – Size of Network	Small	Small
Use of VLSM	No	Yes
Resource Usage	Low	Low
Implementation and Maintenance	Simple	Simple
Classful/ Classless	Classful	Classless
Metric	Hop	Hop
Time Period	30 sec	30 sec
Administrative Distance(AD)	120	120
Algorithm	Bellman-Ford	Bellman-Ford
Updates	Full Table	Full Table

- RIP is a standardized vector distance routing protocol and uses a form of distance as hop count metric. It is a distance vector. Through limiting the number of hop counts allowed in paths between sources and destinations, RIP prevents routing loops. Typically, the maximum number of hops allowed for RIP is 15. However, by achieving this routing loop prevention, the size of supporting networks is sacrificed. Since the maximum number of hop counts allowed for RIP is 15, as long as the number goes beyond 15, the route will be considered as unreachable.
- When first developed, RIP only transmitted full updates every 30 seconds. In the early distributions, traffic was not important because the routing tables were small enough. As networks become larger, massive traffic burst becomes more likely during the 30 seconds period, even if the routers had been initialized at different times. Because of this random initialization, it is commonly understood that the routing updates would spread out in time, but that is not the case in real practice.

RIP Configuration



Router configuration:

Router-1 Configuration

```
Router>
Router>enable
Router#configure terminal
Router(config)#hostname Router1
Router1(config)#interface fastethernet 0/0
Router1(config-if)#ip address 10.1.1.100 255.0.0.0
Router1(config-if)#no shutdown
Router1(config-if)#exit
Router1(config)#interface fastethernet 1/0
Router1(config-if)#ip address 20.1.1.100 255.0.0.0
Router1(config-if)#no shutdown
Router1(config-if)#exit
Router1(config)#router rip
Router1(config-router)#network 10.0.0.0
Router1(config-router)#network 20.0.0.0
```

Router-2 Configuration

```
Router>
Router>enable
Router#configure terminal
Router(config)#hostname Router2
Router2(config)#interface fastethernet 0/0
Router2(config-if)#ip address 20.1.1.200 255.0.0.0
Router2(config-if)#no shutdown
Router2(config-if)#exit
Router2(config)#interface fastethernet 1/0
Router2(config-if)#ip address 30.1.1.100 255.0.0.0
Router2(config-if)#no shutdown
Router2(config-if)#exit
Router2(config)#router rip
```

```
Router2(config-router)#network 20.0.0.0
Router2(config-router)#network 30.0.0.0
```

Router-3 Configuration

```
Router>
Router>enable
Router#configure terminal
Router(config)#hostname Router3
Router3(config)#interface fastethernet 0/0
Router3(config-if)#ip address 30.1.1.200 255.0.0.0
Router3(config-if)#no shutdown
Router3(config-if)#exit
Router3(config)#interface fastethernet 1/0
Router3(config-if)#ip address 40.1.1.100 255.0.0.0
Router3(config-if)#no shutdown
Router3(config-if)#exit
Router3(config)#router rip
Router3(config-router)#network 30.0.0.0
Router3(config-router)#network 40.0.0.0
```

CONCLUSION

In this practical, we learn about implementation of dynamic routing algorithm using Cisco packet tracer.

Signature with date:

____

Practical 10

TITLE: Implementation of VLAN

OBJECTIVES:

After completing study of this practical the students will be familiarized with…

- ➢ Concept of VLAN

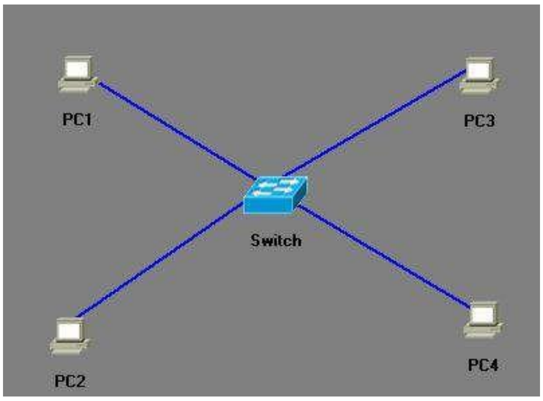
THEORY:

VLAN:

- VLAN refers to Virtual Local Area Network
- VLAN that extends its functionalities beyond a single LAN through VLAN a network is divided into different logical segments which known as broadcast domains.
- In technical terms, a VLAN is a broadcast domain created by switches.
- All devices, by default, are in VLAN 1.
- For devices in different VLAN's to communicate, you must use a router or Layer 3 switch.
- The standard range consists of VLANs 1 to 1024.
- The extended range consists of VLANs 1025 to 4096.

Create Simple VLAN

We are creating simple VLAN. We will take Four PC & one switch. We will create two VLAN named "VLAN8" and "VLAN9".Then we put ports 1 & 2 into VLAN8 and ports 3 & 4 into VLAN9.Then we will check how the communication is done between different nodes.



Step 1: configuration of VLAN in Switch

```
Switch>enable
Switch#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname Vlan_Switch
Vlan_Switch(config)#vlan 8
VLAN 8 added:     Name:VLAN0008
Vlan_Switch(config)#vlan 8 name rikita
Vlan_Switch(config)#vlan 9 name nikita
```

```
VLAN 9 added:     Name:nikita
Vlan_Switch(config)#interface fastethernet0/1
Vlan_Switch(config-if)#switchport mode access
Vlan_Switch(config-if)#switchport access vlan 8
Vlan_Switch(config-if)#exit
Vlan_Switch(config)#interface fastethernet0/2
Vlan_Switch(config-if)#switchport mode access
Vlan_Switch(config-if)#switchport access vlan 8
Vlan_Switch(config-if)#exit
Vlan_Switch(config)#interface fastethernet0/3
Vlan_Switch(config-if)#switchport mode access
Vlan_Switch(config-if)#switchport access vlan 9
Vlan_Switch(config-if)#exit
Vlan_Switch(config)#interface fastethernet0/4
Vlan_Switch(config-if)#switchport mode access
Vlan_Switch(config-if)#switchport access vlan 9
Vlan_Switch(config-if)#exit
```

Step 2: configuration of PC.

PC1

```
    IP Address.......................: 10.1.1.1
    Subnet Mask...................... : 255.0.0.0
    Default Gateway.................. : 10.1.1.4
```

PC2

```
IP Address.......................: 10.1.1.2
    Subnet Mask...................... : 255.0.0.0
    Default Gateway.................. : 10.1.1.4
```

PC3

```
IP Address.......................: 10.1.1.3
    Subnet Mask...................... : 255.0.0.0
    Default Gateway.................. : 10.1.1.4
```

PC4

```
IP Address.......................: 10.1.1.4
    Subnet Mask...................... : 255.0.0.0
    Default Gateway.................. : 10.1.1.4
```

Step 3: Check connectivity.

PC1:>ping 10.1.1.2

```
Pinging 10.1.1.2 with 32 bytes of data:

Reply from 10.1 1.2: bytes=32 time=60ms TTL=241
Reply from 10.1.1.2: bytes=32 time=60ms TTL=241

Reply from 10.1.1.2: bytes=32 time=60ms TTL=241
Reply from 10.1.1.2: bytes=32 time=60ms TTL=241
Reply from 10.1.1.2: bytes=32 time=60ms TTL=241

Ping statistics for 10.1.1.2:     Packets: Sent = 5, Received = 5, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 50ms, Maximum = 60ms, Average = 55ms

PC1:>ping 10.1.1.3
Pinging 10.1.1.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.1.1.3:
    Packets: Sent = 5, Received = 0, Lost = 5 (100% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC1:>ping 10.1.1.4
Pinging 10.1.1.4 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 10.1.1.4:
    Packets: Sent = 5, Received = 0, Lost = 5 (100% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Here we get pinging from PC1 to PC2 only.

So Here Switch is decided into two logical segment.

Step 4: Verify Configuration.

```
Vlan_Switch#
```

```
Vlan_Switch#showvlan



VLAN Name                          Status  Ports
---- ------------------------------ --------- ------------------------------
1    default                        active    Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                              Fa0/9, Fa0/10, Fa0/11, Fa0/12
8    rikita                         active    Gi0/-11, Gi0/-10
9    nikita                         active    Gi0/-9, Gi0/-8
1002 fddi-default                   active
1003 token-ring-default             active
1004 fddinet-default                active
1005 trnet-default                  active

VLAN Type SAID       MTU Parent RingNoBridgeNoStpBrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet 100001      1500    -    -      -      -  -      0    0
8    enet 100008      1500    -    -      -      -  -      0    0
9    enet 100009      1500    -    -      -      -  -      0    0
1002 fddi 101002      1500 -    -     -     -  -      0     0
1003 tr 101003        1500 -    -     -     -  -      0    0
1004 fdnet 101004     1500 -    -     -   ieee -      0     0
1005 trnet 101005     1500 -    -     -   ibm -       0     0
```

CONCLUSION

In this practical, we learn about implementation of virtual lan(VLAN) using Cisco packet tracer.

Signature with date:

____