



The Hashemite University
Faculty of Engineering
Computer engineering Department

Computer Networks Lab Manual
110408454

Prepared by: Eng.Salah Abu-Ghalyon.

Eng.Sawsan Harahsheh.

Table of Content

Experiment	Page
<u>Experiment 1:</u> Network Cabling & Troubleshooting Network Connectivity using Windows Networking Utilities	1
<u>Experiment 2:</u> Subnetting & Variable Length Subnet Mask (VLSM)	8
<u>Experiment 3:</u> Introduction to Routers	17
<u>Experiment 4:</u> Packet Tracer Network Simulator	26
<u>Experiment 5:</u> Introduction to Routing Protocols	31
<u>Experiment 6:</u> Routing Protocols (2)	35
<u>Experiment 7:</u> Introduction to Access control lists	41
<u>Experiment 8:</u> Introduction to switches & Virtual LANs (VLANs)	47
<u>Experiment 9:</u> Inter-VLAN Routing	54

Experiment 1:

Network Cabling & Troubleshooting Network Connectivity using Windows Networking Utilities

1. Objectives

1. Troubleshooting Network Connectivity using Windows Networking tools.
2. Become familiar with the different types of cables that are used in the lab.
3. Become familiar with cable tester equipment.
4. Construct and test straight-Through, Crossover, and Rollover Cables.

2. Devices and equipment :

- PC with an Ethernet 10/100 NIC installed.
- UTP cables
- RJ-45 connectors and guides
- Data cable stripper
- Crimp tool
- Cable tester

3. Theoretical Background :

In any computer network, computers and other network devices are connected together using cables. Unshielded twisted pair, or UTP for short, is the most common cable that is used for local area networks. The current standard for UTP cables is Category 5 Enhanced, better known as Cat5e. Cat5e cable is available in several standards that suitable for different wiring applications. Cable runs are terminated with RJ45 connectors. Making network cables is a relatively simple process. In addition to cable and connectors, only a crimper wire trimmer is required.

UTP cable can be constructed to be:

- Straight-through cable
- Crossover cable
- Rolled cable

Straight-Through Cable

A straight-through cable has connectors on each end that are terminated in similar manner in accordance with either the T568A or T568B standards ¹(shown in Figure 1). This means that the color of the wire on

¹ These standards specified by the Electronics Industry Alliance/Telecommunications Industry Association (EIA/TIA).

Pin 1 at one end of the cable will be the same for Pin 1 at the other end. Similarly Pin 2 will have the same color as Pin 2, and so on.

Straight-through cables can be used to connect switch to router, switch to PC or server, and hub to PC or server.

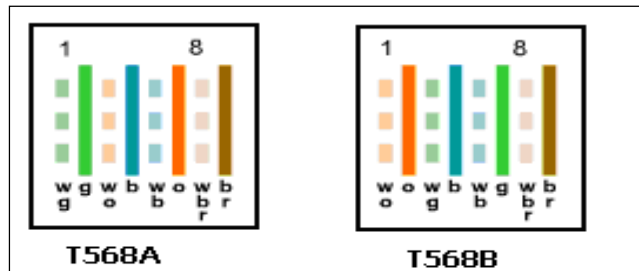


Figure 1: T568A standard and T568B standard

Crossover Cable

One end of the cable should be wired to the T568A standard. And the other end should be wired to the T568B standard. This crosses the transmit pairs and the receive pairs, the second and third pair, to allow communication to take place. Cross cable color arrangement is shown in Figure 2.

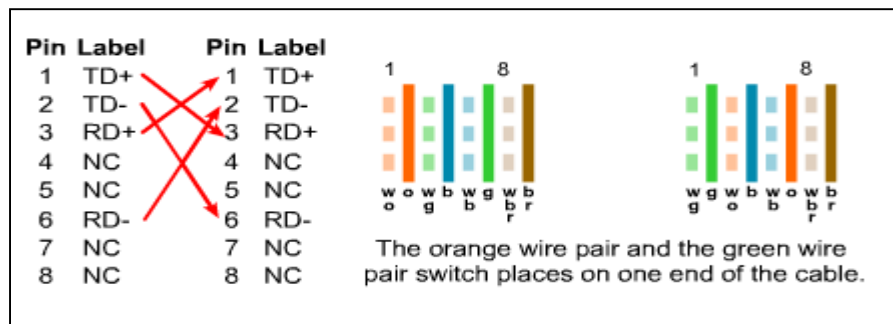


Figure 2: UTP Implementation – Crossover Cable (color arrangement).

Crossover cables are used in cases where we need to connect two devices that have the same interface. For example, connecting a switch to a switch, a hub to a hub, a PC to a PC, hub to switch, and a router directly to a host.

Rollover Cable

Although rollover cable is not used to connect any Ethernet connections. It can be used to connect a PC to a router console serial communication (com) port in order to configure it. Rollover cables are probably the easiest cables to make, because you just cut the end off on one side of a straight-through cable and reverse it. Figure 3 shows the eight wires used in a rolled cable.

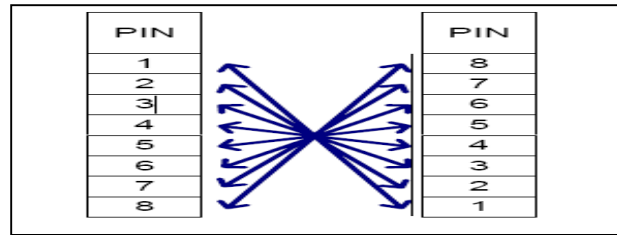


Figure 3: UTP Implementation – Rollover Cable.

4. Methodology:

4.1 Straight-Through Cable Construction

Build a straight-through and a crossover Ethernet cable. The following instructions will help you complete this task.

1. Measure off about 3 feet of CAT5 cable.
2. Remove 2.54 cm (1 inch) of the outer jacket from the end of the cable using the cable stripper.
3. Untwist and align the eight leads according to the color arrangement for straight-through cable as shown in Figure (1).
4. Trim all eight leads to a length of 1/2 inch by simultaneously clipping the wires with the wire cutter.
5. Keep the leads aligned, insert them into the RJ45 connector and press the cable into the connector until the metal wire ends are all visible through the end of the connector.
6. While pressing on the cable, inspect the end of the connector to make sure the metal wire leads can all be seen pressed against the end of the connector. If all of the wire ends are pressed against the end of the connector, squeeze the handles of the crimper as far as they can go and hold them for approximately 3 or 4 seconds.
7. Remove the RJ45 cable from the crimper and inspect the connector to make sure the wires are all aligned properly.
8. Using the same procedure as you used to connect the first connector, attach another RJ45 connector to the other end of the cable to complete construction of a straight-through (T568A/T568B) cable.

4.2 Crossover Cable Construction

Use the color arrangement for cross cable as shown in Figure (2), and the same basic instructions listed in section 4.1 to make a crossover cable.

4.3 Rollover Cable Construction

Use the color arrangement for Rollover cable as shown in Figure (3), and the same basic instructions mentioned in section 4.1 to make a crossover cable.

4.4 Cable Testing

Plug both ends of the cable into the cable tester and check to make sure that the cable is good by pressing the 'mode' button until 'wiremap' appears. A good straight cable should read 12345678 - 12345678, and a crossover cable should read 12345678 - 36145278. If the cable tester indicates a bad cable, clip the RJ45 connector off of one of the ends of the cable (choose the one that is most questionable first) and then attach a new connector to the exposed end of the cable and retest. If the rebuilt cable fails the test, clip off the other connector (that was not replaced) and install a new RJ45 connector on that end of the cable and then retest.

Note: Since it is easy to confuse straight-through cables and crossover cables, it is recommended to clearly label the different types of cables.

Troubleshooting Network Connectivity using Windows Networking Utilities.

TCP/IP includes a variety of utilities that gather information about various protocols in the network. They are usually command-line utilities. In this experiment, you will learn to use them to collect information about networks. Therefore, you will use the command prompt for this experiment.

1. The ping (Packet Internet Groper) Command

The **ping** command is used to verify TCP/IP Network layer connectivity on the local host computer or another device in the network. The command can be used with a destination IP address or qualified name, such as eagle-server.example.com, to test domain name services (DNS)² functionality. For this lab, only IP addresses will be used.

The ping operation is straightforward. The source computer sends an ICMP³ echo request to the destination. The destination responds with an echo reply. If there is a break between the source and destination, a router may respond with an ICMP message that the host is unknown or the destination network is unknown.

Using ping

1. Access the command prompt. (Start > Programs > Accessories > Command Prompt)
2. Ping the IP address of another computer. Output should look similar to that shown in Figure 4.

² DNS is a protocol provides a mapping between cryptic IP address and, host names.

³ The Internet Control Message Protocol (ICMP) is a TCP/IP Network layer protocol used by both **ping** and **tracert** to send messages between devices.

```

C:\> ping 172.16.1.2
Pinging 172.16.1.1 with 32 bytes of data:
Reply from 172.16.1.2: bytes=32 time<1ms TTL=128
Reply from 172.16.1.2: bytes=32 time<1ms TTL=128
Reply from 172.16.1.2: bytes=32 time<1ms TTL=128
Reply from 172.16.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 172.16.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>

```

Figure 4: Output of the ping Command.

Here is a description for each parameter of the Output shown in Figure 4

1. Destination address set to the IP address for the local computer.
2. Reply information:
 - Bytes—size of the ICMP packet.
 - Time—elapsed time between transmission and reply.
 - TTL—default TTL value of the destination device, minus the number of routers in the path. The maximum TTL value is 128.
3. Summary information about the replies:
4. Packets Sent—number of packets transmitted. By default, four packets are sent.
5. Packets Received—number of packets received.
6. Packets Lost —difference between number of packets sent and received.
7. Information about the delay in replies, measured in milliseconds.

2. The tracert Command

The **tracert** command is useful for learning about network latency and path information. Instead of using the ping command to test connectivity of each device to the destination, one by one, the **tracert** command can be used. On Linux and Cisco IOS devices, the equivalent command is traceroute.

Using tracert

1. Access the command prompt.
2. Issue the following command. **C:\> tracert 192.168.254.254**. Output from the tracert command should be similar to that shown in Figure 5.

```

C:\> tracert 192.168.254.254
Tracing route to 192.168.254.254 over a maximum of 30 hops
  0  <1 ms    <1 ms    <1 ms    172.16.255.254
  1  <1 ms    <1 ms    <1 ms    10.10.10.6
  2  <1 ms    <1 ms    <1 ms    192.168.254.254
Trace complete.
C:\>

```

Figure 5: Output of the tracert Command.

Use help with tracert and ping

Try **tracert -?** and then **ping -?** to see the options available for the commands used previously. In looking at the help for ping, notice the **-t** option, which will send continuous pings, not just four. More importantly, notice the two commands to stop it:

- Control-Break
- Control-C

3. The ipconfig Command

The IP Configuration (IPCONFIG) command can be used to display current IP configuration parameters for a host computer. To display these parameters, follow the following steps:

1. Start command prompt.
2. To verify the TCP/IP parameters, type **ipconfig** and press enter, you will see the IP address of your computer, subnet mask, and default gateway.
3. For a detailed configurations, type **ipconfig /all** and press enter, now you will see the host name of your computer, IP address of your computer, subnet mask, default gateway, MAC/Physical address, DNS servers, WINS server IP addresses etc.

4. The ARP Command

Address Resolution Protocol (ARP) is used as a tool for confirming that a computer is successfully resolving network Layer 3 addresses to Media Access Control (MAC) Layer 2 addresses. The TCP/IP network protocol relies on IP addresses like 192.168.14.211 to identify individual devices and to assist in navigating data packets between networks. While the IP address is essential to move data from one LAN to another, it cannot deliver the data in the destination LAN by itself. Local network protocols, like Ethernet or Token Ring, use the MAC, or Layer 2, address to identify local devices and deliver all data.

This is an example of a MAC address:

- 00-02-A5-9A-63-5C

A MAC address is a 48-bit address displayed in Hexadecimal (HEX) format as six sets of two HEX characters separated by dashes. In this format each hex symbol represents 4 bits. With some devices, the 12 hex characters may be displayed as three sets of four characters separated by periods or colons (0002.A59A.635C).

ARP maintains a table in the computer for IP and MAC address combinations. In other words, it keeps track of which MAC address is associated with an IP address. If ARP does not know the MAC address of a local device, it issues a broadcast using the IP address. This broadcast searches for the MAC address that corresponds to the IP address. If the IP address is active on the LAN, it will send a reply from which ARP will extract the MAC address. ARP will then add the address combination to the local ARP table of the requesting computer.

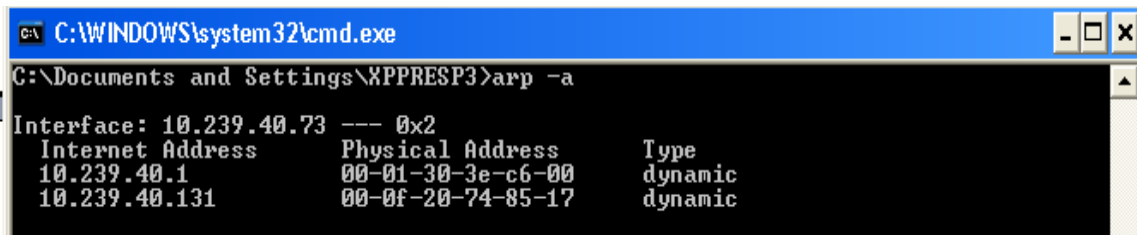
MAC addresses and ARP are only used within the LAN. When a computer prepares a packet for transmission, it checks the destination IP address to see if it is part of the local network. It does this by checking to see if the network portion of the IP address is the same as the local network. If it is, the ARP process is consulted to get the MAC address of the destination device using the IP address. The MAC address is then applied to the data packet and used for delivery. If the destination IP address is not local, the computer will need the MAC address of the default gateway.

Using arp command

1. Access the command prompt.
2. Display the ARP table

a. In the window type **arp -a** and press Enter. Do not be surprised if there are no entries. The message displayed will probably be, 'No ARP Entries Found'. Windows computers remove any addresses that are unused after a couple minutes.

b. Try pinging a couple local addresses and a website URL. Then re-run the command. Figure 6 below shows a possible result of the **arp -a** command. The MAC address for the website will not be listed because it is not local, but that will cause the default gateway to be listed. Notice that for each IP address there is a physical address, or MAC, and type, indicating how the address was learned.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\XPPRESP3>arp -a
Interface: 10.239.40.73 --- 0x2
Internet Address      Physical Address      Type
10.239.40.1          00-01-30-3e-c6-00    dynamic
10.239.40.131        00-0f-20-74-85-17    dynamic
```

Figure 6: arp -a command results.

Experiment 2:

Subnetting & Variable Length Subnet Mask (VLSM)

1. Objectives :

- To become more familiar with the concept of Subnetting.
- To become familiar with the concept of Variable Length Subnet Mask (VLSM).
- To utilize the above concept practically in a networked environment using Packet Tracer.

2. Theoretical Background :

Part 1: Classless Subnetting

When given an IP Address, Major Network Mask, and a Subnet Mask, how can you determine other information such as:

- The subnet address of this subnet
- The broadcast address of this subnet
- The range of Host Addresses for this subnet
- The maximum number of subnets for this subnet mask
- The number of hosts for each subnet
- The number of subnet bits
- The number of this subnet

Let's start with an example:

Host IP Address	138.101.114.250
Major Network Mask	255.255.0.0 (/16)
Major (Base) Network Address	
Major Network Broadcast Address	
Total Number of Host Bits Number of Hosts	
Subnet Mask	255.255.255.192 (/26)
Number of Subnet Bits Number of Usable Subnets (all 0's used, all 1's not used)	
Number of Host Bits per Subnet Number of Usable Hosts per Subnet	
Subnet Address for this IP Address	
IP Address of First Host on this Subnet	
IP Address of Last Host on this Subnet	
Broadcast Address for this Subnet	

Part 1.1: Determine Major Network Information

Before we begin subnetting, let's gather some information regarding the network in general. Using the Major Network Mask, determine the major network Address, the broadcast address for the entire network, and the number of hosts for the entire network.

IP Address: 138.101.114.250
Major Network Mask: 255.255.0.0

Step 1: Translate Host IP Address and Major Network Mask into binary notation

Convert the Host IP Address and Major Network Mask to binary:

	138.	101.	114.	250
IP Address:	10001010	01100101	01110010	11111010
Major Mask:	11111111	11111111	00000000	00000000

Step 2: Major Network Address

1. Draw a line under the major mask
2. Perform a bit-wise AND operation on the IP Address and the Subnet Mask
Note: 1 AND 1 results in a 1, 0 AND anything results in a 0.
3. Express the result in Dotted Decimal Notation
4. The result is the Major Network Address of this for this host IP Address is 138.101.0.0

	138.	101.	114.	250
IP Address:	10001010	01100101	01110010	11111010
Major Mask:	<u>11111111</u>	<u>11111111</u>	<u>00000000</u>	<u>00000000</u>
Network Add.:	10001010	01100101	00000000	00000000
	138.	101.	0.	0

Step 3: Broadcast Address for the Major Network Address

Remember that the network mask separates the network portion of the address from the host portion. The network address has all 0's in the host portion of the address while the broadcast address has all 1's in the host portion of the address.

	Network portion		Host portion	
	138.	101.	0.	0
Network Add.	10001010	01100101	00000000	00000000
Major Mask	11111111	11111111	00000000	00000000
Broadcast.	10001010	01100101	11111111	11111111
	138.	101.	1.	1

By counting the number of host bits we can determine the total number of usable hosts for this network (before subnetting)

Host bits: 16

Total number of hosts: $2^{16} = 65,536$ $65,536 - 2 = 65,534$ (Can't use the all 0's address, network address, or the all 1's address, broadcast address.)

Add this information to our table:

Host IP Address	138.101.114.250
Major Network Mask	255.255.0.0 (/16)
Major (Base) Network Address	138.101.0.0
Major Network Broadcast Address	138.101.255.255
Total Number of Host Bits Number of Hosts	16 bits or 2 ¹⁶ or 65,536 total hosts 65,536 – 2 = 65,534 usable hosts
Subnet Mask	255.255.255.192 (/26)
Number of Subnet Bits Number of Usable Subnets (all 0's used, all 1's not used)	
Number of Host Bits per Subnet Number of Usable Hosts per Subnet	
Subnet Address for this IP Address	
IP Address of First Host on this Subnet	
IP Address of Last Host on this Subnet	
Broadcast Address for this Subnet	

Part 1.2: Determine Subnet Information

Step 1: Translate Host IP Address and Subnet Mask into binary notation

```

138.    101.    114.    250
IP Address: 10001010 01100101 01110010 11111010
Subnet Mask 11111111 11111111 11111111 11000000
255.    255.    255.    192

```

Step 2: Determine the Network (or Subnet) where this Host address lives:

1. Draw a line under the mask.
2. Perform a bit-wise AND operation on the IP Address and the Subnet Mask
Note: 1 AND 1 results in a 1, 0 AND anything results in a 0.
3. Express the result in Dotted Decimal Notation.
4. The result is the Subnet Address of this Subnet which is 138.101.114.192.

```

138.    101.    114.    250
IP Address: 10001010 01100101 01110010 11111010
Subnet Mask: 11111111 11111111 11111111 11000000
Subnet Add.: 10001010 01100101 01110010 11000000
138.    101.    114.    192

```

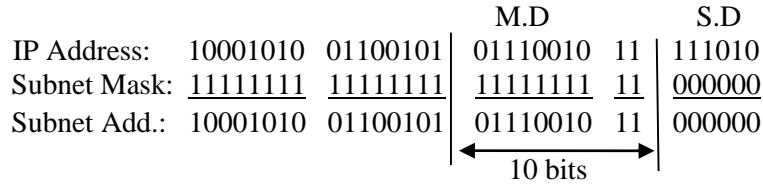
Add this information to our table:

Subnet Address for this IP Address	138.101.114.192
------------------------------------	-----------------

Step 3: Determine which bits in the address contain Network information and which contain Host information:

1. Draw the “Major Divide” (M.D) as a wavy line where the 1’s in the Major (Base) Network Mask ends (also the mask if there was no subnetting). In our example, the Major Network Mask is 255.255.0.0 or the first 16 left-most bits.

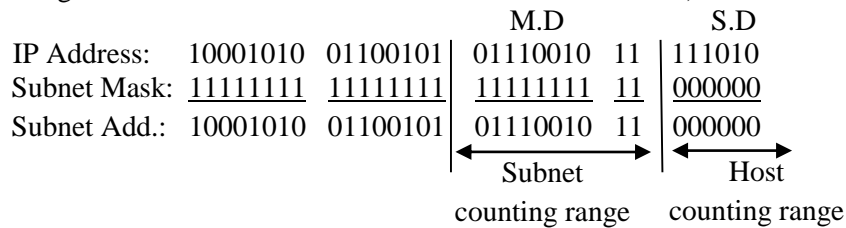
2. Draw the “Subnet Divide” (S.D.) as a straight line where the 1’s in the given Subnet Mask ends, the network information ends where the 1’s in the mask ends.



3. The result is the “Number of Subnet Bits” may be determined by simply counting the number of bits between the M.D. and S.D., which in this case is 10 bits.

Step 4: Determine bit ranges that are for subnets and for hosts:

1. Label the “subnet counting range” between the M.D. and the S.D. (these are the bits that are being incremented to make the subnet numbers or addresses).
2. Label the “host counting range” between the S.D. and all of the way to the end on the right (these are the bits that are being incremented to make the host numbers or addresses).



Step 5: Determine the range of host addresses available on this subnet, and the broadcast address on this subnet:

1. Copy down all of the network/subnet bits of the Network Address (i.e. all bits before the S.D.).
2. In the host portion (to the right of the S.D.) make the host bits all 0’s except for the right most bit (or least significant bit), which you make a 1. This gives you the first Host IP Address on this subnet, which is the first part of the result for “Range of Host Addresses for This Subnet,” or in our example 138.101.114.193.
3. Now, in the host portion (to the right of the S.D.) make the host bits all 1’s except for the right most bit (or least significant bit), which you make a 0. This gives you the last Host IP Address on this subnet, which is the last part of the result for “Range of Host Addresses for This Subnet,” or in our example 138.101.114.254.
4. In the host portion (to the right of the S.D.) make the host bits all 1’s. This gives you the Broadcast IP Address on this subnet. This is the result for “Broadcast Address of This Subnet,” or in our example 138.101.114.255.

IP Address:	10001010	01100101	01110010	11	111010
Subnet Mask:	<u>11111111</u>	<u>11111111</u>	<u>11111111</u>	<u>11</u>	<u>000000</u>
Subnet Add.:	10001010	01100101	01110010	11	000000
			← Subnet	→ Host	
			counting range	counting range	

First Host	10001010	01100101	01110010	11	000001
	138.	101.	114.		193
Last Host	10001010	01100101	01110010	11	111110
	138.	101.	114.		254
Broadcast	10001010	01100101	01110010	11	111111
	138.	101.	114.		255

Let's add some of this information to our table:

Host IP Address	138.101.114.250
Major Network Mask	255.255.0.0 (/16)
Major (Base) Network Address	138.101.0.0
Major Network Broadcast Address	138.101.255.255
Total Number of Host Bits	16 bits or 2 ¹⁶ or 65,536 total hosts
Number of Hosts	65,536 – 2 = 65,534 usable hosts
Subnet Mask	255.255.255.192 (/26)
Number of Subnet Bits	
Number of Usable Subnets (all 0's used, all 1's not used)	
Number of Host Bits per Subnet	
Number of Usable Hosts per Subnet	
Subnet Address for this IP Address	138.101.114.192
IP Address of First Host on this Subnet	138.101.114.193
IP Address of Last Host on this Subnet	138.101.114.254
Broadcast Address for this Subnet	138.101.114.255

Step 6: Determine the number of usable subnets

The number of usable subnets depends upon the equipment and the network administrator. Subtract 0 to use all subnets, subtract 1 if not using either the all 0's or all 1's subnets, subtract 2 if not using the all 0's and all 1's subnets.

The number of subnets is determined by how many bits are in the subnet counting range (in this example, 10 bits) minus 1 for the last subnets, the "all ones subnets" which is sometimes not used. The first subnets, known as the "all zeroes subnets" is a usable subnets in this example.

1. Use the formula $2^n - 1$, where n is the number of bit in the subnet counting range.
2. $2^{10} - 1 = 1024 - 1 = 1023$
3. Subtract 1 from the number of usable subnets (the "all zeroes" subnets)

Number of Subnet Bits	10 bits
Number of Usable Subnets (all 0's used, all 1's not used)	$2^{10} - 1 = 1024 - 1 = 1023$ usable subnets

Step 7: Determine the number usable hosts per subnet

The number of hosts per subnet is determined by the number of host bits (in this example, 6 bits) minus 2 (1 for the subnet address and 1 for the broadcast address of the subnet).

$2^6 - 2 = 64 - 2 = 62$ hosts per subnet.

Number of Host Bits per Subnet	6 bits
Number of Usable Hosts per Subnet	$2^6 - 2 = 64 - 2 = 62$ hosts per subnet

Final Answers

Host IP Address	138.101.114.250
Major Network Mask	255.255.0.0 (/16)
Major (Base) Network Address	138.101.0.0
Major Network Broadcast Address	138.101.255.255
Total Number of Host Bits	16 bits or 2 ¹⁶ or 65,536 total hosts
Number of Hosts	$65,536 - 2 = 65,534$ usable hosts
Subnet Mask	255.255.255.192 (/26)
Number of Subnet Bits	10 bits
Number of Usable Subnets (all 0's used, all 1's not used)	$2^{10} - 1 = 1024 - 1 = 1023$ usable subnets
Number of Host Bits per Subnet	6 bits
Number of Usable Hosts per Subnet	$2^6 - 2 = 64 - 2 = 62$ hosts per subnet
Subnet Address for this IP Address	138.101.114.192
IP Address of First Host on this Subnet	138.101.114.193
IP Address of Last Host on this Subnet	138.101.114.254
Broadcast Address for this Subnet	138.101.114.255

Part 2: Introduction to VLSM

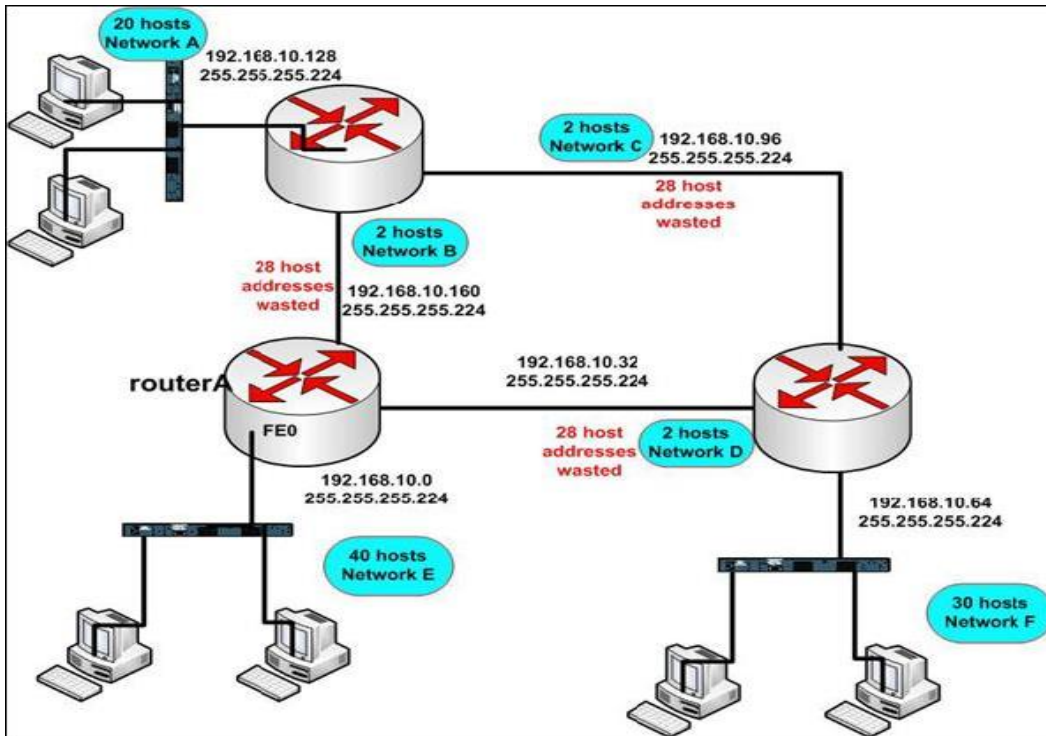
We will subnet an already subnetted network into multiple subnets with variable subnet masks and then allocate them within our sample network.

The Benefits of VLSM

VLSM provides the ability to subnet an already subnetted network address. The benefits that arise from this behavior include:

1. **Efficient use of IP addresses:** IP addresses are allocated according to the host space requirement of each subnet.
IP addresses are not wasted; for example, a Class C network of 192.168.10.0(/24) subnetted with a mask of 255.255.255.224 (/27) allows you to have eight equal size subnets, each with 32 IP addresses (30 of which could be assigned to devices). What if we had a few WAN links in our network (WAN links need only one IP address on each side, hence a total of two IP addresses

per WAN link are needed). Without VLSM that would be impossible. With VLSM we can subnet one of the subnets, 192.168.10.32, into smaller subnets with a mask of 255.255.255.252 (/30). This way we end up with eight subnets with only two available hosts each that we could use on the WAN links. The /30 subnets created are: 192.168.10.32/30, 192.168.10.36/30, 192.168.10.40/30, 192.168.10.44/30, 192.168.10.48/30, 192.168.10.52/30, 192.168.10.56/30, 192.168.10.60/30.



2. Support for better route summarization: VLSM supports hierarchical addressing design.

Therefore; it can effectively support route aggregation, also called route summarization. The latter can successfully reduce the number of routes in a routing table by representing a range of network subnets in a single summary address. For example subnets 192.168.10.0/24, 192.168.11.0/24 and 192.168.12.0/24 could all be summarized into 192.168.8.0/21. Meaning, in the routing table instead of having three entries all pointing to the same exit interface, we have one entry that covers all the three, thus reducing the size of the routing table.

Address Waste without VLSM

The following diagram shows a sample internetwork which uses a network class C address 192.168.10.0(/24) subnetted into 8 equal size subnets (32 available IP addresses each) to be allocated to the various portions of the network. This specific network consists of 3 WAN links that are allocated a subnet address range each from the pool of available subnets. Obviously 30 IP address are wasted (28 host addresses) since they are never going to be used on the WAN links.

Implementing VLSM

In order to be able to implement VLSMs in a quick and efficient way, you need to understand and memorize the IP address blocks (total addresses) and available hosts for various subnet masks. Remember that we cannot use both the broadcast address and the network address as legitimate host addresses. Create a small table with all of this information and use it to create your VLSM network. The following table shows the block sizes used for subnetting a Class C subnet.

Subnet Prefix	Mask	Hosts	Block Size
/26	255.255.255.192	62	64
/27	255.255.255.224	30	32
/28	255.255.255.240	14	16
/29	255.255.255.248	6	8
/30	255.255.255.252	2	4

Having this table in front of you is very helpful. For example, if you have a subnet with 28 hosts then you can easily see from the table that you will need a block size of 32. For a subnet of 40 hosts you will need a block size of 64.

VLSM Rules

There are several ways of performing VLSM. Here are our preferred rules:

1. Work out the required size for each network. Remember to leave room for the identity address and the broadcast address.
2. Allocate networks (subnets) from the **biggest requirements down to the smallest**. This is very important.
3. After each allocation, there will be leftover addresses. Use from the biggest remaining down to the smallest for future allocations.
4. Try to keep networks of the same size adjacent in the numbering. But, also try to keep networks connected to the same router adjacent in the numbering, too. This will allow route summarization.

Other Rules

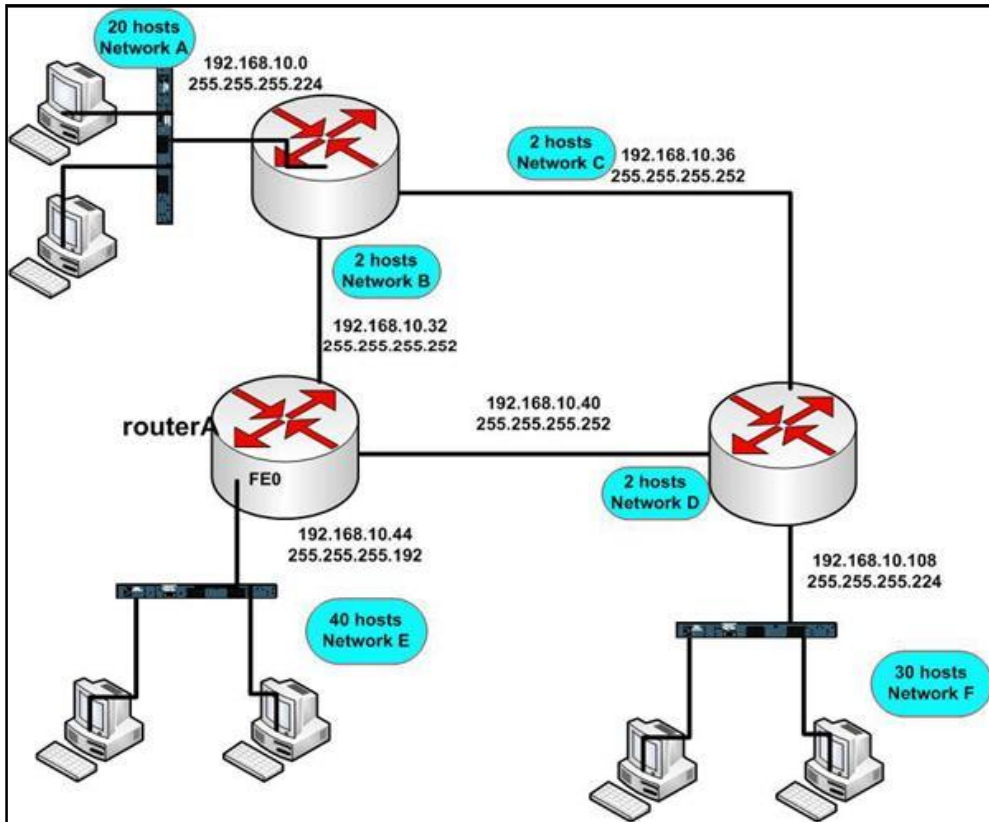
1. If you split a /N range into two, you end up with two /N+1 ranges.
2. Split into 4, get /N+2 ranges. Split into 8, get /N+3 range etc.
3. Remember to watch the step factor as you are subdividing and allocating addresses!
4. Remember that links between 2 routers are also networks. Allocate /30 to each one.
5. Place point-to-point router address allocations at the bottom of the original address range.

Example: Create a VLSM Network

Let us use the sample network provided above to implement VLSM. According to the number of hosts in each subnet, identify the addressing blocks required. You should end up with the following VLSM table for this Class C network 192.168.10.0/24.

Network	Subnet Prefix	Mask	Hosts	Block Size
A	/27	255.255.255.224	20	32
B	/30	255.255.255.252	2	4
C	/30	255.255.255.252	2	4
D	/30	255.255.255.252	2	4
E	/26	255.255.255.192	40	64
F	/27	255.255.255.224	30	32

We have identified the necessary block sizes for our sample network. The final step is to allocate the actual subnets to our design and construct our VLSM network. We will take into account that subnet-zero



can be used in our network design, therefore the following solution will really allow us to save unnecessary addressing waste:

Network	Mask
A	192.168.10.64/27
B	192.168.10.128/30
C	192.168.10.132/30
D	192.168.10.136/30
E	192.168.10.0/26
F	192.168.10.96/27
Next subnet	192.168.10.140/?

With VLSM we have occupied 140 addresses. Nearly half of the address space of the Class C network is saved. The address space that remains unused is available for any future expansion. Isn't that amazing? We have reserved a great amount of addresses for future use. Our sample network diagram is finalized as shown on the following diagram:

Experiment 3:

Introduction to Routers

1. Objectives :

- Cable devices and establish console connections.
- Introduce different access levels and commands of routers.
- Erase and reload the routers.
- Perform basic router configuration.
- Verify and test configurations using show commands, ping and traceroute.

2. Devices and Equipments :

- Two PCs with an Ethernet 10/100 NIC installed.
- Cisco 2811 Router.
- Ethernet straight-through, crossover cables, and console cables.

3. Theoretical Background :

You could say that a router is nothing more than a small PC with a smaller operating system and no direct user interface hardware, such as keyboards or video monitors. If you look at what a router does for networking, it is essentially the same as a personal computer. A router looks and acts like a PC in many ways. Like a PC, the router is built with input/output (I/O) ports, it has a processor and memory chips, it provides a set of instructions that tells the router what to do, and it has an operating system that runs the router.

In this experiment, you will review previously learned skills like cabling devices. And you will also learn to establish a console connection, basic IOS command line interface operation and configuration commands, and how to save configuration files and capture your configurations to a text file.

3.1 Overview of Cisco Router Hardware

The Cisco 2811 router is a multiple-chip standalone cryptographic module. The router has a processing speed of 350MHz. The interfaces for the router are located on the rear and front panels as shown in Figure 1 and Figure 2, respectively.

The front panel contains the following:

- (1) Power inlet
- (2) Power switch
- (3) Optional RPS input
- (4) Console and auxiliary ports
- (5) Two Universal Serial Bus (USB) ports.
- (6) Compact Flash (CF) drive.
- (7) Four LEDs that output status data about the system power, auxiliary power, system activity, and compact flash busy status.

The back panel contains the following:

- (1) Ground connector

- (2) and (3) Ethernet ports and LEDs
- (4)-(7) High-speed WAN interface card (HWIC) slots.
- (8) Enhanced Network Module (ENM) slot.

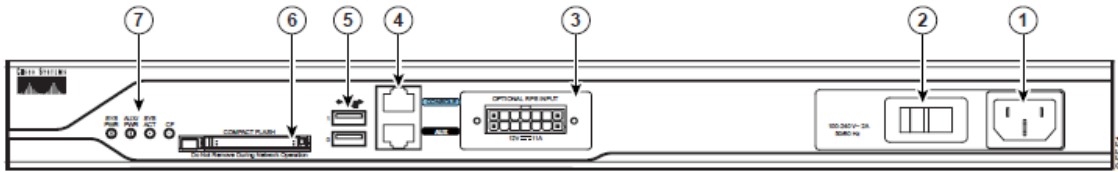


Figure 1: Cisco 2811 Front Panel Physical Interfaces.

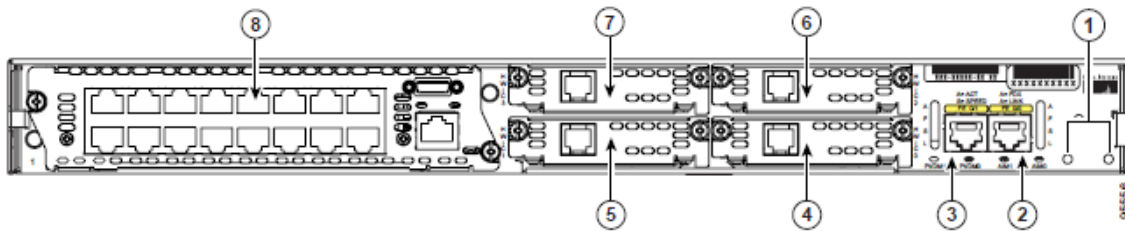


Figure 2: Cisco 2811 Rear Panel Physical Interfaces.

Figure

Although there are several different types and models of routers, every router has the same general hardware components. Depending on the model, those components are located in different places inside the router. To see the internal router components, you must unscrew the metal cover and take it off the router. Usually you do not need to open the router unless you are upgrading memory.

3.1.1 Router Components and their Functions

Routers have many of the same hardware and software components that are found in other computers including:

1. **CPU:** The CPU executes operating system instructions, such as system initialization, routing functions, and switching functions.
2. **RAM:** RAM stores the instructions and data needed to be executed by the CPU. RAM is used to store these components:
 - **Operating System:** The Cisco IOS (Internetwork Operating System) is copied into RAM during bootup.
 - **Running Configuration File:** This is the configuration file that stores the configuration commands that the router IOS is currently using. With few exceptions, all commands configured on the router are stored in the running configuration file, known as running-config.
 - **IP Routing Table:** This file stores information about directly connected and remote networks. It is used to determine the best path to forward the packet.
 - **ARP Cache:** This cache contains the IPv4 address to MAC address mappings. The ARP cache is used on routers that have LAN interfaces such as Ethernet interfaces.

-Packet Buffer: Packets are temporarily stored in a buffer when received on an interface or before they exit an interface.

RAM is volatile memory and loses its content when the router is powered down or restarted.

3. **ROM:** ROM is a form of permanent storage. ROM used to store:

- The bootstrap instructions
- Basic diagnostic software
- Scaled-down version of IOS

ROM uses firmware, which is software that is embedded inside the integrated circuit. Firmware includes the software that does not normally need to be modified or upgraded, such as the bootup instructions. ROM does not lose its contents when the router loses power or is restarted.

4. **Flash Memory:** Flash memory is nonvolatile computer memory that can be electrically stored and erased. Flash is used as permanent storage for the operating system, Cisco IOS. In most models of Cisco routers, the IOS is permanently stored in flash memory and copied into RAM during the bootup process, where it is then executed by the CPU. Some older models of Cisco routers run the IOS directly from flash.
5. **NVRAM:** NVRAM (Nonvolatile RAM) does not lose its information when power is turned off. NVRAM is used as permanent storage for the startup configuration file (startup-config). All configuration changes are stored in the running-config file in RAM, and with few exceptions, are implemented immediately by the IOS. To save those changes in case the router is restarted or loses power, the running-config must be copied to NVRAM, where it is stored as the startup-config file. NVRAM retains its contents even when the router reloads or is powered off.

3.2 Overview of Cisco Router Software

3.2.1 Internetwork Operating System (IOS)

The operating system software used in Cisco routers is known as Cisco Internetwork Operating System (IOS). Like any operating system on any computer, Cisco IOS manages the hardware and software resources of the router, including memory allocation, processes, security, and file systems. Cisco IOS is a multitasking operating system that is integrated with routing, switching, internetworking, and telecommunications functions.

Although the Cisco IOS may appear to be the same on many routers, there are many different IOS images. An IOS image is a file that contains the entire IOS for that router. Cisco creates many different types of IOS images, depending upon the model of the router and the features within the IOS. Typically the more features in the IOS, the larger the IOS image, and therefore, the more flash and RAM that is required to store and load the IOS. For example, some features include the ability to run IPv6 or the ability for the router to perform NAT (Network Address Translation).

As with other operating systems Cisco IOS has its own user interface. Although some routers provide a graphical user interface (GUI), the command line interface (CLI) is a much more common method of configuring Cisco routers.

3.2.2 Accessing the Cisco Routers thru CLI

Cisco IOS Software provides a relatively simple command-line interface (CLI) that is used both to accept user commands and to display router output.

This CLI environment is accessible through several methods:

- Through a console session. A console uses a low-speed serial connection directly from a computer or terminal to the console connection on the router. This method does not require the configuration of network services on the router.
- Through a dialup connection by using a modem or a null modem connected to the router AUX port. This method does not require the configuration of network services on the router.
- Through a Telnet connection to the router as a virtual terminal. To establish a Telnet session to the router, at least one interface must be configured for IP. Virtual terminal sessions also must be configured for login and passwords.

3.2.3 Overview of Users Levels and Modes

Cisco IOS Software provides a command interpreter service known as the command executive (EXEC). After each command is entered, the EXEC validates and executes the command.

As a security feature, Cisco IOS Software separates EXEC sessions into two different access levels: user EXEC level and privileged EXEC level.

1. User EXEC level allows you to access only basic monitoring commands.
2. Privileged EXEC level allows you to access all router commands. It can be password protected to allow only authorized users the ability to configure or manage the router.

To assist you in navigation through the Cisco IOS CLI, the command prompt changes to reflect your position within the command hierarchy. This setup allows you to easily identify where within the command structure you are at any given moment. The following table is a summary of command prompts and the corresponding location within the command structure.

IOS Command mode	Role of command mode	How to Enter This Mode	How to Leave This Mode	Command Prompt
User EXEC mode.	- Basic commands which do not change system parameters (e.g., ping, telnet, traceroute)	-telnet to IP address of router (requires terminal password) -direct serial connection through console port (may require a password)	exit	Router>
Privileged EXEC mode	- Manage configuration files, examine state of router.	enable (requires additional enable password)	Disable	Router#
Global Configuration mode	- Change system wide configuration parameters.	configure term	CTRL-z	Router(config)#
Interface Configuration mode	- modify configuration of specific interface.	Router (config) interface <type> <number>	end	Router(config-if)#
Router Configuration mode	- modify configuration of specific routing protocol.	Router (config) router <routing protocol>	exit	Router(config-router)#
Line Configuration mode	- modify configuration at a Line level (vty).	Router (config) line <type>	exit	Router(config-line)#

Table 1: IOS command prompts.

3.3 Basic Router Configuration

All CLI configuration changes to a Cisco router are made from global configuration mode. The following command moves the router into global configuration mode and allows entry of commands from the terminal:

```
Router#configure terminal
Router (config) #
```

1- Configuring a router Name

One of the first basic configuration tasks is to name the router. Naming a router helps to better manage the network by uniquely identifying each router within the network. The router is named in global configuration mode. The name of the router is called the host name and is displayed as the system prompt. If a router is not named, the system default is Router.

This task is accomplished in global configuration mode with the following command:

```
Router (config) #hostname Networklab
Networklab (config) #
```

When the Enter key is pressed, the prompt will change from the default host name, which is Router, to the newly configured host name, which is **Networklab**.

2- Configuring router passwords

Access passwords are set for the privileged exec mode and user entry point such as console, aux, and virtual lines. The privileged exec mode password is the most critical password, since it controls access to the configuration mode.

Configure the privileged exec password.

Cisco IOS supports two commands that set access to the privileged exec mode. One command, **enable password**, contains weak cryptography and should never be used if the **enable secret** command is available. The **enable secret** command uses a very secure MD5 cryptographic hash algorithm.

The following example set the privileged exec password to **cisco**.

```
Router1 (config) # enable secret cisco
```

Configure the console password

The console password controls console access to the router. To set the console access password to **class**.

```
Router1 (config) # line console 0
Router1 (config-line) # password class
Router1 (config-line) # login
```

Configure the virtual line password.

The virtual line password controls Telnet access to the router. In early Cisco IOS versions, only five virtual lines could be set, 0 through 4. In newer Cisco IOS versions, the number has been expanded. The following example set the virtual line access password to **user**.

```
Router1(config-line)# line vty 0 4
Router1(config-line)# password user
Router1(config-line)# login
```

Sometimes it is undesirable for passwords to be shown in clear text in the output from the show running-config or show startup-config commands. This command is used to encrypt passwords in configuration output:

```
Router(config)#service password-encryption
```

3-Configuring Banners

An IOS banner is used to give information to users or administrators when they log in to a router via a terminal line. We are going to cover three types of banners:

- MOTD (Message Of The Day)
- Login
- Exec

An MOTD banner is sent to a terminal as soon as the terminal's connection becomes active. A login banner is also sent to a terminal when a terminal becomes operative. The login banner is displayed after an MOTD banner if there is one. An exec banner is displayed to a terminal immediately after a person has successfully logged in. We use the global configuration mode command banner to create a banner.

```
Router(config)# banner {exec | login | motd} dc4 message dc
```

For example

```
Router(config)# banner motd %
Enter TEXT message. End with the character '%'.
This is the motd banner.
You have accessed a private system.
%
```

4- Configure Cisco Router Interfaces.

A router interface can be configured from the console or a virtual terminal line. Each interface must have an IP address and subnet mask to route IP packets.

⁴The argument **dc** is a delimiting character. The delimiting character can be any character as long as it is not part of the message, and it must be the same at the end as it is at the beginning of the *message*.

To configure an interface follow these steps:

- Enter global configuration mode.
- Enter interface configuration mode.
- Specify the interface address and subnet mask.
- Enable the interface.

By default, interfaces are turned off, or disabled. To turn on or enable an interface, the command `no shutdown` is entered. If an interface needs to be disabled for maintenance or troubleshooting, use the `shutdown` command to turn off the interface.

Configuring an Ethernet interface

The following example configures the router fa0/0 interface.

```
Router1(config)# interface fa0/0
Router1(config-if)# ip address <ip address > <subnetmask >
Router1(config-if)# no shutdown
Router1(config-if)# end
Router1#
```

Configuring a serial interface

Serial interfaces require a clock signal to control the timing of the communications. In most environments, a DCE device such as a CSU/DSU will provide the clock. By default, Cisco routers are DTE devices but they can be configured as DCE devices.

On serial links that are directly interconnected, as in a lab environment, one side must be considered a DCE and provide a clocking signal. The clock is enabled and speed is specified with the `clock rate` command. The available clock rates in bits per second are 1200, 2400, 9600, 19200, 38400, 56000, 64000, 72000, 125000, 148000, 500000, 800000, 1000000, 1300000, 2000000, or 4000000. Some bit rates might not be available on certain serial interfaces. This depends on the capacity of each interface.

The following example configures the router serial 0/0 interface.

```
Router(config)#interface serial 0/0
Router(config-if)#ip address <ip address > <netmask >
Router(config-if)# clockrate 56000
Router(config-if)# no shutdown
```

Interface descriptions

A description of an interface can help a network user remember specific information about the interface, such as what network the interface services. The description will appear in the configuration files that exist in the router memory. However, it will not affect the operation of a router.

The following example configures a description to the router serial 0/0 interface.

```
Router(config)#interface serial 0/0
Router(config-if)#description This interface connected to
```

5- Save the Router Configuration File.

Cisco IOS refers to RAM configuration storage as running-configuration, and NVRAM configuration storage as startup-configuration. For configurations to survive rebooting or power restarts, the RAM configuration must be copied into non-volatile RAM (NVRAM). This does not occur automatically, NVRAM must be manually updated after any changes are made.

Compare router RAM and NVRAM configurations.

Use the Cisco IOS **show** command to view RAM and NVRAM configurations.

Display the contents of NVRAM. If the output of NVRAM is missing, it is because there is no saved configuration.:

```
Router1# show startup-config
```

Display the contents of RAM.

```
Router1#show running-config
```

Save RAM configuration to NVRAM.

For a configuration to be used the next time the router is powered on or reloaded, it must be manually saved in NVRAM. Save the RAM configuration to NVRAM:

```
Router1# copy running-config startup-config
Destination filename [startup-config]? <ENTER> Building configuration...
[OK]
Router1#
```

6- Removing/Undoing commands

The Cisco IOS Software provides an easy way to remove commands from a configuration: simply navigate to the proper mode and type **no** followed by the command to be removed.

3.4 CLI Help

The Cisco IOS command-line interface CLI offers context-sensitive help, a useful tool if you are a new user because at any time during an EXEC session, you can type a question mark (?) to get help. Two types of context-sensitive help are available: **word help** and command **syntax help**.

Word help can be used to obtain a list of available commands that begin with a character (or string of characters) that you have begun to type. To use word help, enter? Immediately after the character, or characters, in question, without a space. Let's try the show command and see what Will do for us:

```
Router# show v?
version vines vpdn
Router# show v
```

Command syntax help can be used to obtain a list of command, keyword, or argument options that are available based on the syntax you have already entered. To use command syntax help, enter ? in the place of a keyword or argument. See this example

```
Router(config-if)# ip add ?
  A.B.C.D    IP address
Router(config-if)# ip add 192.168.1.1 ?
  A.B.C.D    IP subnet mask
Router(config-if)# ip add 192.168.1.1 255.255.255.0 ?
```

Command Line Completion

Another CLI feature is command line completion, the function of the [tab] key. The tab key will fill in with the command that matches the text you enter. Let's take the show version command and try it again, but this time let's put the [tab] key in the place of the question mark:

```
Router# show ve[tab]
Router# show version
```

Syntax Checking

Automatic syntax checking is built into the CLI. If a command is improperly spelled, or is not a valid command, the router will respond by placing a caret symbol below the errant letter, word, or argument. If you were to type in show version like this example, here is what you would receive in response:

```
Router# show versoin
                ^
% Invalid input detected at '^' marker.
```

Command History

The Router keeps the last 10 commands you issued in its HISTORY, which is a special memory buffer that holds the "Command History." You can:

- Press the UP Arrow key to go to the next most recent command.
- Press the DOWN Arrow key to go back down through the previous commands (after pressing UP arrow).

Experiment 4:

Packet Tracer Network Simulator

1. Objectives :

- Develop an understanding of the basic functions of Packet Tracer.
- Create/model a simple Ethernet network using two hosts and a hub.
- Observe traffic behavior on the network.
- Observe data flow of ARP broadcasts and pings.

2. Devices and Equipments :

- A PC with Packet Tracer program installed on it.

3. Theoretical Background :

Prior to performing routing and switching experiments, you must install and familiarize yourself with the Packet Tracer program. This is a tool designed by Cisco Systems for the Cisco Networking Academy. It may be used in the classroom for many purposes such as: Student use to become familiar with initial commands, and when needed to use more than one device (such as router and switch) to perform a lab.

What is Packet Tracer?

Packet Tracer is a simulation-based learning environment for networking novices to design, configure, and troubleshoot computer networks. 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.

Students can use Packet Tracer to drag and drop networking devices (nodes) such as routers, switches, and workstations into logical topology space (the Logical Workspace). They can then specify the types of interconnections between these devices (links) and configure the devices they created. Once they have designed and configured a network of nodes and links, they can then launch sample data packets into the network, either in real time, or in a user-controlled simulation mode. The packets are displayed graphically. The student can step the packet through the network, examining the processing decisions made by networking devices as they switch and route the packet to its destination.

4. Procedures:

Packet Tracer Installation is straight forward. Simply click the setup executable and follow the prompts. There is no installation key at this time. After installation, the desktop icon should be visible on the desktop. Double click it to run the Packet Tracer program.

Step 1: Start Packet Tracer

When you open Packet Tracer, by default you will be presented with the following interface in Figure 1. This initial interface contains ten components. If you are unsure of what a particular interface item does, move your mouse over the item and a help balloon will explain the item. Table 1 list these components as numbered in Figure 1.

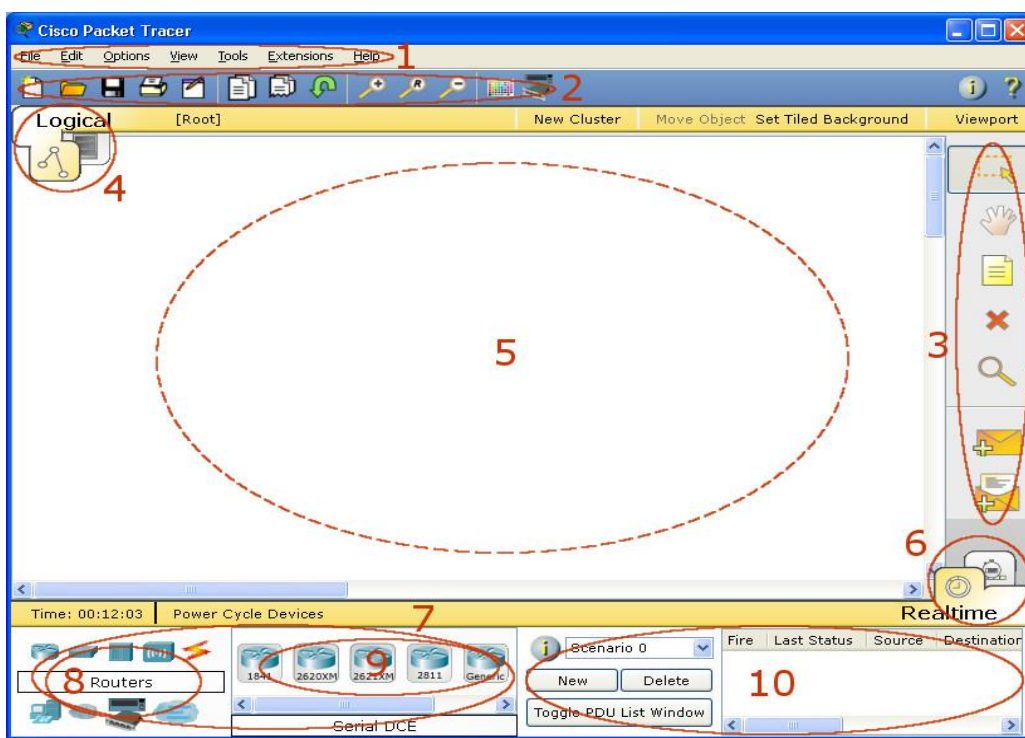


Figure 1: PT interface

#	component
1	Menu Bar
2	Main Tool Bar
3	Common Tools Bar
4	Logical/Physical Workspace and Navigation Bar
5	Workspace
6	Realtime/Simulation Bar
7	Network Component Box
8	Device-Type Selection Box
9	Device-Specific Selection Box
10	User Created Packet Window

Table (1): PT interface components.

Step 2: Create a logical network diagram with two PCs and a hub

The bottom left-hand corner of the Packet Tracer screen displays eight icons that represent device categories or groups, such as Routers, Switches, or End Devices.

Moving the cursor over the device categories will show the name of the category in the box.

To select a device, first select the device category. Once the device category is selected, the options within that category appear in the box next to the category listings. Select the device option that is required.

a) Select **End Devices** from the options in the bottom left-hand corner (see **Figure 1** below). Drag and drop two generic PCs onto your design area.

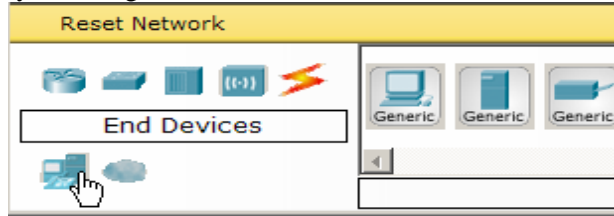


Figure 1: End devices categories.

b) Select **Hubs** from the options in the bottom left-hand corner. Add a hub to the prototype network by dragging and dropping a generic hub onto the design area.

c) Select **Connections** from the bottom left-hand corner (shown in **Figure 2**). Choose a **Copper Straight-through** cable type.



Figure 2: Connections categories.

d) Click the first host, **PC0**, and assign the cable to the **FastEthernet** connector. Click the hub, **Hub0**, and select a connection port, **Port 0**, to connect to **PC0**.(see **Figure 3** below)

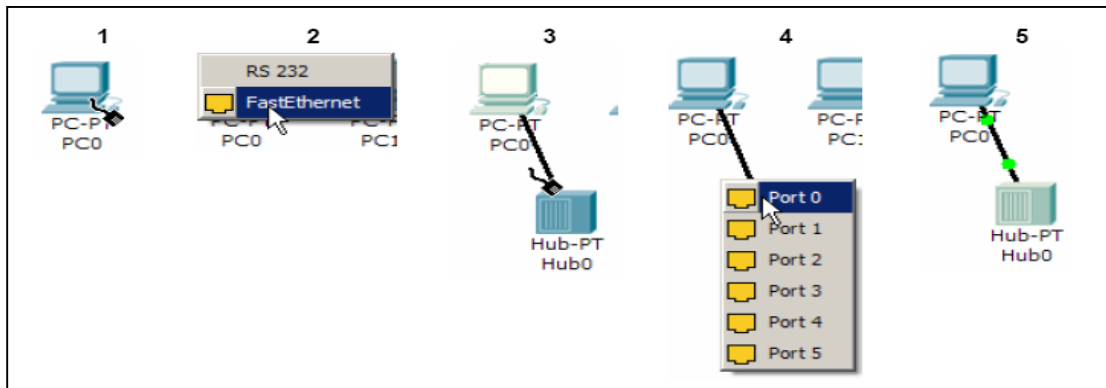


Figure 3: PC to Hub connection.

e) Repeat Step c for the second PC, **PC1**, to connect the PC to **Port 1** on the hub.

Note: There should be green dots at both ends of each cable connection. If not, check the cable type selected.

Step 3: Configure host names and IP addresses on the PCs

- a) Click PC0. A PC0 window will appear.
- b) From the PC0 window, select the **Config** tab (shown in **Figure 4**). Change the PC **Display Name** to **PC-A**. (An error message window will appear warning that changing the device name may affect scoring of the activity. Ignore this error message.) Select the **FastEthernet** tab on the left and add the IP address of **192.168.1.1** and subnet mask of **255.255.255.0**. Close the PC-A configuration window by selecting the **x** in the upper right-hand corner.

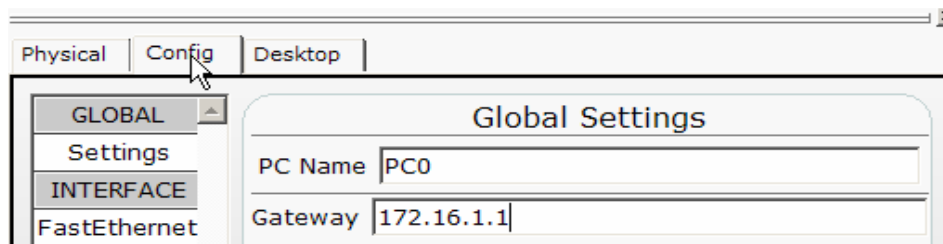


Figure 4: Config tab.

- c) Click PC1.
- d) Select the **Config** tab. Change the PC **Display Name** to **PC-B**. Select the **FastEthernet** tab on the left and add the IP address of **192.168.1.2** and subnet mask of **255.255.255.0**. Close the PC-B configuration window.

Step 4: Observe the flow of data from PC-A to PC-B by creating network traffic

- a) Switch to **Simulation** mode by selecting the tab (shown in **Figure 5**) that is partially hidden behind the **Realtime** tab in the bottom right-hand corner. The tab has the icon of a stopwatch on it.
- b) Click the **Edit Filters** button in the **Edit List Filters** area. Clicking the **Edit Filters** button will create a pop-up window (shown in **Figure 6**). In the pop-up window, click the **Show All/None** box to deselect every filter. Select just the **ARP** and **ICMP** filters.



Figure 5: Simulation mode tab.

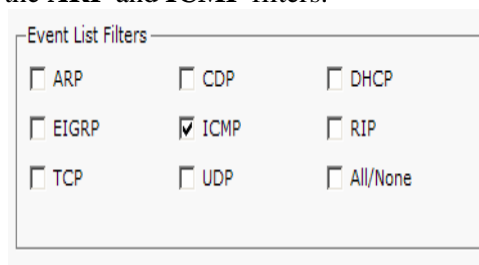


Figure 6: Edit List Filters window.



Figure 7: Simple PDU button

- c) Select a **Simple PDU** by clicking the closed envelope on the right vertical toolbar (shown in **Figure 7**). Move your cursor to the display area of your screen. Click **PC-A** to establish the source. Move your cursor to **PC-B** and click to establish the destination.

Note: Notice that two envelopes are now positioned beside PC-A. One envelope is ICMP, while the other is ARP. The Event List in the Simulation Panel will identify exactly which envelope represents ICMP and which represents ARP.

d) Select **Auto Capture / Play** from the **Play Controls** area of the Simulation Panel. Below the **Auto Capture / Play** button is a horizontal bar, with a vertical button that controls the speed of the simulation. Dragging the button to the right will speed up the simulation, while dragging is to the left will slow down the simulation.

e) The animation will run until the message window *No More Events* appears. All requested events have been completed. Select OK to close the message box.

f) Choose the **Reset Simulation** button in the Simulation Panel. Notice that the ARP envelope is no longer present. This has reset the simulation but has not cleared any configuration changes or dynamic table entries, such as ARP table entries. The ARP request is not necessary to complete the **ping** command because PC-A already has the MAC address in the ARP table.

g) Choose the **Capture / Forward** button. The ICMP envelope will move from the source to the hub and stop. The **Capture / Forward** button allows you to run the simulation one step at a time. Continue selecting the **Capture / Forward** button until you complete the event.

h) Choose the **Power Cycle Devices** button on the bottom left, above the device icons.

i) An error message will appear asking you to confirm reset. Choose **Yes**. Now both the ICMP and ARP envelopes are present again. The **Reset Network** button will clear any configuration changes not saved and will clear all dynamic table entries, such as the ARP and MAC table entries.

Step 5: View ARP Tables on each PC

a) Choose the **Auto Capture / Play** button to repopulate the ARP table on the PCs. Click **OK** when the *No More Events* message appears.

b) Select the magnifying glass on the right vertical tool bar.

c) Click **PC-A**. The ARP table for PC-A will appear. Notice that PC-A does have an ARP entry for PC-C. View the ARP tables for PC-B and PC-C as well. Close all ARP table windows.

d) Click the **Select Tool** on the right vertical tool bar. (This is the first icon present in the toolbar.)

e) Click **PC-A** and select the **Desktop** tab.

f) Select the **Command Prompt** and type the command **arp -a** and press *enter* to view the ARP table from the desktop view. Close the PC-A configuration window.

g) Examine the ARP table for **PC-B**.

h) Close the PC-B configuration window.

i) Click the **Check Results** button at the bottom of the instruction window to verify that the topology is correct.

Step 6: Saving the Topology

To save the topology (uses .pkt file extension).

Experiment 5:

Introduction to Routing Protocols

1. Objectives :

- Understand the concept of static routing.
- To understand the concept of dynamic routing.
- Configure static routes between routers.
- Configure the RIP and EIGRP dynamic routing protocols on routers.

2. Devices and Equipments :

- Two PCs with an Ethernet 10/100 NIC installed.
- Two 2811 Cisco routers
- Crossover, and console cables.
- Smart serial DB60 cable.

3. Theoretical Background :

Routing is the process that a router uses to forward packets toward the destination network. The routing process is based on the destination IP address of a packet. All devices along the way use the destination IP address to send the packet in the right direction to reach its destination. To make the correct decisions, routers must learn how to reach remote networks.

When static routing is used, a network administrator configures information about remote networks manually. When routers use dynamic routing, this information is learned from other routers. In this case the IOS learns about a remote network and the interface that it will use to reach that network, it adds that route to the routing table as long as the exit interface is enabled.

3.1 Static Routes

Static route knowledge is administered manually by a network administrator who enters it into a router's configuration. The administrator must update this static route entry manually whenever an internetwork topology change requires an update.

Since static routes are configured manually, network administrators must add and delete static routes to reflect any network topology changes. In a large network, the manual maintenance of routing tables could require a lot of administrative time. On small networks with few possible changes, static routes require very little maintenance. Static routing is not as scalable as dynamic routing because of the extra administrative requirements. Even in large networks, static routes that are intended to accomplish a specific purpose are often configured in conjunction with a dynamic routing protocol.

Configuring Static Routes

Use the following steps to configure static routes:

Step 1: Determine all desired destination networks, their subnet masks, and their gateways. A gateway can be either a local interface or a next-hop address that leads to the desired destination.

Step 2: Enter global configuration mode.

Step 3: Use the ip route command The correct syntax for the *ip route* command is as follows:

```
Router(config)#ip route network-address subnet-mask {ip-address | exit-interface }
```

The following parameters are used:

- *network-address* - Destination network address of the remote network to be added to the routing table .
- *subnet-mask* - Subnet mask of the remote network to be added to the routing table. The subnet mask can be modified to summarize a group of networks.

One or both of the following parameters must also be used:

- *ip-address* - Commonly referred to as the next-hop router's IP address
- *exit-interface*- Outgoing interface that would be used in forwarding packets to the destination network.

Step 4: Repeat Step 3 for as many destination networks as were defined in Step 1.

Step 5: Exit global configuration mode.

Step 6: Save the active configuration to NVRAM by using the *copy running-config startup-config* command.

Configuring Default Route

Default routes route packets with destinations that do not match any of the other routes in the routing table. Routers typically are configured with a default route for Internet-bound traffic because it is often impractical and unnecessary to maintain routes to all networks in the Internet. A default route is actually a special static route that uses the following format:

```
ip route 0.0.0.0 0.0.0.0 [next-hop-address | outgoing interface]
```

3.2 Dynamic Routes

Dynamic route knowledge works differently. After a network administrator enters configuration commands to start dynamic routing, the route knowledge automatically is updated by a routing process

whenever new information is received from the internetwork. Changes in dynamic knowledge are exchanged between routers as part of the update process.

Dynamic routing protocols are usually used in larger networks to ease the administrative and operational overhead of using only static routes. Typically, a network uses a combination of both a dynamic routing protocol and static routes. In most networks, a single dynamic routing protocol is used, however there are cases where different parts of the network may use different routing protocols.

IP Routing Protocols

There are several dynamic routing protocols for IP. Here are some of the more common dynamic routing protocols for routing IP packets:

RIP (Routing Information Protocol)
IGRP (Interior Gateway Routing Protocol)
EIGRP (Enhanced Interior Gateway Routing Protocol)
OSPF (Open Shortest Path First)
IS-IS (Intermediate System-to-Intermediate System)
BGP (Border Gateway Protocol)

Dynamic Routing Protocols Classification

There are three classes of routing protocols:

Distance vector: The distance-vector protocols find the best path to a remote network by judging distance. Each time a packet goes through a router, that's called a hop. The route with the least number of hops to the network is determined to be the best route. The vector indicates the direction to the remote network. Both RIP and IGRP are distance-vector routing protocols. They send the entire routing table to directly connected neighbors.

Link-state: In link-state protocols, also called shortest-path-first protocols, the routers each create three separate tables. One of these tables keeps track of directly attached neighbors, one determines the topology of the entire internetwork, and one is used as the routing table. Link-state routers know more about the internetwork than any distance-vector routing protocol. OSPF is an IP routing protocol that is completely link state. Link state protocols send updates containing the state of their own links to all other routers on the network.

Hybrid: Hybrid protocols use aspects of both distance vector and link state—for example, EIGRP.

RIP

RIP is a distance vector routing protocol. RIP uses hop count as the metric for path selection. If the hop count is greater than 15, the packet is discarded. By default; routing updates are broadcast every 30 seconds.

Commands to Enable RIP

To enter the router configuration mode for RIP, enter *router rip* at the global configuration prompt. Notice that the prompt changes from a global configuration prompt to the following:

```
Router(config)#router rip
R1 (config-router)#
```

By entering the RIP router configuration mode, the router is instructed to run RIP. But the router still needs to know which local interfaces it should use for communication with other routers, as well as which locally connected networks it should advertise to those routers. To enable RIP routing for a network, use the network command in the router configuration mode and enter the network address for each directly connected network.

```
Router (config-router)#network directly-connected-network-address
```

If you need to completely remove the RIP routing process from a device, negate the command with *no router rip*. This command stops the RIP process and erases all existing RIP configurations.

Example (1) below shows the process of enabling RIP and specifying directly connected networks.

```
BHM(config)#router rip
! selects RIP as the routing protocol
BHM(config-router)#network 1.0.0.0
! specifies a directly connected network)
BHM(config-router)#network 2.0.0.0
! specifies a directly connected network
```

Example (1): RIP Configuration commands.

Verify Routing Setting:

Table (1) list a group of commands used to verify Routing Protocols.

Command	Effect
Show ip protocol	Verify the activated protocols.
Show ip route	Display all the routes in the routing table.
Show ip neighbor	Verify OSPF Neighbors

Table (1): commands used for monitoring Routing protocols.

Experiment 6:

Routing Protocols (2)

1. Objectives :

- To understand the concept of dynamic routing.
- Identify the EIGRP packets and their jobs.
- Understand the mechanism of the OSPF routing protocol.
- Configure the EIGRP and OSPF dynamic routing protocols on routers.

2. Devices and Equipments :

- Two PCs with an Ethernet 10/100 NIC installed.
- Two 2811 Cisco routers
- Crossover, and console cables.
- Smart serial DB60 cable.

3. Theoretical Background :

Routing is the process that a router uses to forward packets toward the destination network. The routing process is based on the destination IP address of a packet. All devices along the way use the destination IP address to send the packet in the right direction to reach its destination. To make the correct decisions, routers must learn how to reach remote networks.

When static routing is used, a network administrator configures information about remote networks manually. When routers use dynamic routing, this information is learned from other routers. In this case the IOS learns about a remote network and the interface that it will use to reach that network, it adds that route to the routing table as long as the exit interface is enabled.

EIGRP

EIGRP is a balanced hybrid routing protocol developed by Cisco as an enhanced version of IGRP. EIGRP has characteristics in common with both distance vector protocols and link-state protocols. EIGRP calculates the best route to each network or subnet and provides alternative routes that can be used if the current route fails. EIGRP also transmits the subnet mask for each routing entry.

EIGRP packet types:

The EIGRP use s6 different packet types, while communicating with the neighbor EIGRP routers.

- **Hello packets**
- **Update packets**

- **Query packet**
- **Reply packets**
- **Request packets**
- **Acknowledgment packet**

Hello packets:

The hello packet is sent between the EIGRP neighbors for the recovery and neighbor discovery. It helps to keep the existing neighboring relationship alive. The EIGRP hello packet is the multicast to 224.0.0.10. It does not require acknowledgement. The EIGRP neighbor ship is maintained and discovered by the hello packets. If a router fails to get the hello packet within a hold timer, then the corresponding router can be declared as dead.

Update packets:

The update packets are used to convey the reachability of destination. When the new neighbor is discovered, then the update packet is sent so a neighbor can build up their accurate routing as well as topology table. In that case, the update packet is unicast. In some other cases, like the link cost change, then the update is multicast. The updates are mostly transmitted reliably. So the update packets can be sent to single neighbor as well as a group of neighbors too. This packet has the routing information to whatever router that needs this information. The update packet is assigned an OP code of 1.

Query packet:

The query packet is used when the EIGRP router lost its information about the specific network and it does not have any backup routes. This query packets are always multicast unless it is sent in response to the received query. In that case, it is the unicast back to a successor which originated the query. The EIGRP packets are used to reliable request the routing information. The packets are sent to the neighbors when the route is lost as mentioned above or not at all available as well as the router requires to ask about the route status for the faster convergence. If a router which sends out the does not get the response from any neighbor, then it will resend the query as the unicast packet to a non-responsive neighbor. If there is no response in 16 attempts, then the EIGRP neighbor relationships are reset. The EIGRP query packet is assigned an OP code of 3.

Reply packets:

The EIGRP reply packet is sent in response to the query packets. Then the reply packet is used to reliably respond to the query packet. The EIGRP reply packet is assigned an OP code of 4. The replies are unicast to the originator of the query. The replay packet indicates that the new route to a destination has been found.

Request packets:

The request packet is sent in response to the query packets. Then the reply packet is used to reliably respond to the query packet. The packets are unicast to the query originator. The EIGRP reply packet is assigned an OP code of 4. This request packet is used to receive certain information from 1 or more neighbors and is used in the route server application. This type of packet can also be sent through multicast or unicast, but are transmitted unreliably.

Acknowledgment packet:

The acknowledgement packets are themselves simply the hello packets which have no data. This packet will acknowledge as a receipt of update, query, and replay packets. The ACK uses the OP code as same as hello packets because, it is just a hello which contains no information, so that the OP code is 5. It packets cannot be sent through multicast.

Commands to Enable EIGRP

To enter the router configuration mode for EIGRP, enter `router eigrp autonomous-system` at the global configuration prompt.

```
Router(config)#router eigrp autonomous-system
```

Then the network command is used in router configuration mode.

```
Router(config-router)#network A.B.C.D E.F.G.H
```

Where A.B.C.D if the network id of a directly connected network, and E.F.G.H is the wild card of that network.

The autonomous system parameter is a number chosen by the network administrator between 1 and 65535. The number chosen is the process ID number and is important because all routers in this EIGRP routing domain must use the same process ID number (autonomous-system number).

Example (1) below shows the process of enabling EIGRP and specifying directly connected networks.

```
BHM(config)#router eigrp 1
! selects eigrp as the routing protocol
BHM(config-router)#network 1.0.0.0
! specifies a directly connected network)
BHM(config-router)#network 2.0.0.0
! specifies a directly connected network
```

Example (1): EIGRP Configuration commands.

OSPF

OSPF (Open Shortest Path First) is an interior gateway routing protocol deployed typically in upper tier ISPs for intra-AS routing. The OSPF is a Link State Routing Protocol, it uses the “cost” metric to choose routes which is a value assigned to each route based on the speed of that link. The OSPF sends periodic routing updates rather than send the whole routing table

The OSPF maintains three tables:

Neighbor table: stores information about OSPF neighbors.

Topology table: stores the topology structure of a network.

Routing table: stores the best routes.

OSPF Areas:

OSPF uses the concept of areas. An area is a logical grouping of contiguous networks and routers. All routers in the same area have the same topology table, but they don't know about routers in the other areas. The main benefits of creating areas is that the size of the topology and the routing table on a router is reduced, and routing updates are also reduced.

OSPF Router ID:

OSPF Router ID is an IPv4 address (32-bit binary number) assigned to each router running the OSPF protocol to provide a unique identity to the OSPF Router.

OSPF Router ID selection algorithm works as below:

- Any manually configured OSPF Router ID in OSPF Process is selected as the OSPF Router ID.
- If there is no OSPF Router ID configured, the highest IP address on any of the Routers Loopback Interfaces is selected as the OSPF Router ID.
- If there is no Loopback Interfaces configured, the highest IP address on its active interfaces is selected as the OSPF Router ID.

OSPF Election Process:

If all routers formed adjacencies with every other attached router, a large number of link-state advertisements (LSAs) would be sent over the network. OSPF avoids synchronizing between every pair of routers in the network by using the concept of the Designated Router (DR) and Backup Designated Router and (BDR). In this way, adjacencies are formed only to the DR and BDR.

On LANs, DR and BDR have to be elected. Two rules are used to elect a DR and BDR:

1. Router with the highest OSPF priority will become a DR. By default, all routers have a priority of 1
2. If there is a tie, a router with the highest router ID wins the election

The router with the second highest OSPF priority or router ID will become a BDR.

Steps of OSPF Operation:

1. Establish Router Adjacencies.
2. Elect the DR and the BBDR.
3. Discover Routes.
4. Select the best Routes.
5. Maintain Routing Table.

Commands to Enable OSPF

To enter the router configuration mode for OSPF, enter `router ospf processID` at the global configuration prompt.

```
Router(config)#router ospf ospf processID
```

the process ID number is similar to the autonomous system parameter used in EIGRP and its value should be between 1 and 65535.

Then the network command is used in router configuration mode.

```
Router(config-router)#network A.B.C.D E.F.G.H area X
```

Where A.B.C.D if the network id of a directly connected network, and E.F.G.H is the wild card of that network. And X is the number of the area which in this lab will be always 0.

Example (2) below shows the process of enabling EIGRP and specifying directly connected networks.

```
BHM(config)#router ospf 1
! selects ospf as the routing protocol
BHM(config-router)#network 192.168.10.0 0.0.0.255 area 0
! specifies a directly connected network)
BHM(config-router)#network 192.168.20.0 0.0.0.255 area 0
! specifies a directly connected network
```

Example (2): EIGRP Configuration commands.

Verify OSPF:

Table (1) list a group of commands used to verify OSPF setting.

Command	Effect
Show ospf	Verify OSPF Protocol Settings.
Show ospf interfaces	Verify OSPF Process Information.
Show ip neighbor	Verify OSPF Neighbors

Table (1): commands used for monitoring OSPF.

Experiment 7:

Introduction to Access control lists

1. Objectives :

- Understand the concept of Access control lists.
- Configure, and apply a standard ACL to permit or deny specific traffic.
- Test the ACL to determine if the desired results were achieved.

2. Devices and Equipments :

- Two PCs with an Ethernet 10/100 NIC installed.
- Cisco 2811 Router.
- Ethernet straight-through, and crossover cables.

3. Theoretical Background :

Network administrators must be capable of denying unwanted access to the network while allowing appropriate access. Although security tools such as passwords and physical security devices are helpful, they often lack the flexibility of basic traffic filtering and the specific controls that most administrators prefer. For example, a network administrator might want to allow user's access to the Internet but might not want external user's telnetting into the LAN.

In this experiment, you will understand the concept of Access control lists (ACL). You will also learn how create and apply different types of ACL.

3.5 ACL Overview

Routers provide basic traffic-filtering capabilities, such as blocking Internet traffic, with access control lists (ACLs). An ACL is a sequential collection of permit or deny statements that apply to addresses or upper-layer protocols.

ACLs are lists of instructions that you apply to a router's interface. These lists tell the router what kinds of packets to accept and what kinds of packets to deny. Acceptance and denial can be based on certain specifications, such as source address, destination address, and TCP/UDP port number.

ACLs can be created for all routed network protocols, such as Internet Protocol (IP) and Internetwork Packet Exchange (IPX), to filter packets as the packets pass through a router. ACLs can be configured at the router to control access to a network or subnet.

It is important to configure ACLs correctly and to know where to place ACLs on the network. ACLs serve multiple purposes in a network. Common ACL functions include the following:

- Filtering packets internally.

- Protecting the internal network from illegal Internet access.
- Restricting access to virtual terminal ports.

There are a few important rules that a packet follows when it's being compared with an access list:

- It's always compared with each line of the access list in sequential order—i.e., it'll always start with the first line of the access list, then go to line 2, then line 3, and so on.
- It's compared with lines of the access list only until a match is made. Once the packet matches the condition on a line of the access list, the packet is acted upon, and no further comparisons take place.
- There is an implicit “deny” at the end of each access list—this means that if a packet doesn't match the condition on any of the lines in the access list, the packet will be discarded.

3.6 Access Lists Types

There are two main types of access lists:

1-Standard access lists

These use only the source IP address in an IP packet as the condition test. All decisions are made based on source IP address. This means that standard access lists basically permit or deny an entire suite of protocols. They don't distinguish between any of the many types of IP traffic such as WWW, Telnet, UDP, etc.

2- Extended access lists

Extended access lists can evaluate many of the other fields in the layer 3 and layer 4 headers of an IP packet. They can evaluate source and destination IP addresses, the protocol field in the Network layer header, and port number at the Transport layer header. This gives extended access lists the ability to make much more granular decisions when controlling traffic.

3-Named access lists

Technically there really are only two types of ACLs since named access lists are either standard or extended and not actually a new type.

Named access list allow standard and extended ACLs to be given names instead of numbers. The advantages that a named access list provides are as follows:

- Intuitively identifies an ACL using an alpha or alphanumeric name
- Eliminates the limit of 99 simple and 100 extended ACLs
- Enables administrators to modify ACLs without having to delete and then reconfigure them.

3.7 Placing ACLs

You can assign only one access list per interface per protocol per direction. This means that when creating IP access lists, you can only have one inbound access list and one outbound access list per interface.

1. Inbound access lists

When an access list is applied to inbound packets on an interface, those packets are processed through the access list before being routed to the outbound interface. Any packets that are denied will not be routed because they are discarded before the routing process is invoked.

2. Outbound access lists

When an access list is applied to outbound packets on an interface, those packets are routed to the outbound interface and then processed through the access list before being queued.

Access lists are designed to filter traffic going through the router. They will not filter traffic that has originated from the router. Place IP standard access lists as close to the destination as possible. This is the reason we don't really want to use standard access lists in our networks. You cannot put a standard access list close to the source host or network because you can only filter based on source address and nothing would be forwarded.

Place IP extended access lists as close to the source as possible. By placing this list as close to the source address as possible, you can filter traffic before it uses up your precious bandwidth.

3.8 Access Lists Configuration

1. Configuring a Standard Access Lists

To configure numbered standard ACLs on a Cisco router, you must first create the standard ACL and then activate the ACL on an interface. The access-list global configuration command defines a standard ACL with a number in the range of 1 to 99. The full syntax of the standard ACL command is as follows:

```
Router(config)#access-list access-list-number [deny | permit]source [source-wildcard]
```

Example (2) shows four ACL statements, all of which belong to access list 2, although this combination is not likely it illustrates how several different statement can work. Also remember that if a packet does not match any of these tests, there is an implicit (unseen) deny any at end of the ACL.

```
access-list 2 deny 172.16.1.1
access-list 2 permit 172.16.1.0 0.0.0.255
access-list 2 deny 172.16.1.1 0.0.255.255
access-list 2 permit 172.16.1.1 0.255.255.255
```

Example (2): Standard ACL Statements.

In the first ACL statement, notice that there is no wildcard mask. In situations like this, when no wildcard mask is shown, the default mask is used, which is 0.0.0.0. This statement denies the IP address 172.16.1.1.

The second statement permits the specific host 172.16.1.0 or any host from the 172.16.1.0 subnet.

The third statement denies any host from the 172.16.0.0 network, and the fourth statement permits any host from any network starting with 172.

2. Configuring Extended Access Lists

The procedural steps for configuring extended ACLs are the same as for standard ACLs, you first create the extended ACL and then activate it on an interface.

However, the command syntax and parameters are more complex to support the additional features provided by extended ACLs.

The following is the simple syntax of the extended ACL command:

```
Router(config)# access-list access-list-number{deny | permit| remark }  
protocol source [source-wildcard] destination [destination wildcard][operator  
operand]5 [port port-number or name]
```

In Example (3) ACL 103 It allows traffic coming from any address on the 192.168.10.0 network to go to any destination, subject to the limitation that traffic goes to ports 80 (HTTP) and 443 (HTTPS) only.

```
access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80  
access-list 103 permit tcp 192.168.6.10 0.0.0.255 any eq 443
```

Example (3): Extended ACL Statements

3. Configuring named Access Lists

A named ACL is created with the **ip access-list** command. The named ACL syntax is as follows:

```
ip access-list {extended | standard} name
```

This places the user in ACL configuration mode. In this mode, you can specify one or more conditions for permitting or denying access to a packet.

Example (4) demonstrates creating a named ACL. This access list is given the name server-access. It enables users to access the mail and DNS server only; all other requests are denied.

```
Rt(config)# ip access-list extended server-access  
Rt(config-ext-nacl)# permit tcp any host 131.108.101.99 eq smtp  
Rt(config-ext-nacl)# permit tcp any host 131.108.101.99 eq domain  
Rt(config-ext-nacl)# deny ip any any log  
Rt(config-ext-nacl)# ^Z
```

Example (4): Named ACL Statements

⁵ (Optional) Compares source or destination ports. Possible operands include *lt* (less than), *gt* (greater than), *eq* (equal), *neq* (not equal), and *range* (inclusive range).

To remove a named extended ACL, use the `no ip access-list extended name` global configuration command.

3.9 Applying an ACL to Interfaces

After an ACL is configured, it is linked to an interface using the `ip access-group` command:

```
Router(config-if)#ip access-group {access-list-number | access-list-name} {in  
| out}
```

3.10 Removing an ACL

To remove an ACL from an interface, first enter the `no ip access-group` command on the interface, and then enter the global `no access-list` command to remove the entire ACL.

```
Router(config)# no access-list access-list-number
```

3.11 Using Wildcard Mask Bits

A wildcard mask is a 32-bit quantity that is divided into four octets, with each octet containing 8 bits. A wildcard mask bit of 0 means “check the corresponding bit value,” and a wildcard mask bit of 1 means “do not check (ignore) that corresponding bit value”. A wildcard mask is paired with an IP address, similar to how a subnet mask is paired with an IP address.

ACLs use wildcard masking to identify a single address or multiple addresses for permit or deny tests.

Using the Wildcard any

Assume that you want to specify that any destination address will be permitted in an ACL test. To indicate any IP address, you would enter 0.0.0.0, then, to indicate that the ACL should ignore (that is, allow without checking) any value, the corresponding wildcard mask bits for this address would be all 1s (that is, 255.255.255.255).

You can use the abbreviation of **any** to communicate this same test condition on Cisco IOS Software. Instead of typing 0.0.0.0 255.255.255.255, you can use the word **any** by itself as the keyword. For example, instead of using this:

```
Router(config)#access-list 1 permit 0.0.0.0 255.255.255.255
```

You can use this:

```
Router(config)#access-list 1 permit any
```

Using the Wildcard host

Suppose that you want to specify that a unique host IP address will be permitted in an ACL test. To indicate a host IP address, you would enter the full address (for example, 172.30.16.29). Then, to indicate that the ACL should check all the bits in the address, the corresponding wildcard mask bits for this address would be all 0s (that is, 0.0.0.0).

You can use the abbreviation of *host* to communicate this same test condition on Cisco IOS Software. In the example, instead of typing 172.30.16.29 0.0.0.0, you can use the word *host* in front of the address. For example, instead of using this:

```
Router(config)#access-list 1 permit 172.30.16.29 0.0.0.0
```

You can use this:

```
Router(config)#access-list 1 permit host 172.30.16.29
```

3.12 Monitoring Access Lists

Table (1) list a group of commands used for monitoring ACL.

Command	Effect
Show access-list	Display all access list and their parameters configured on the router. This command does not show you which interface the list is set on.
Show access-list 110	Shows only the parameters for the access list 110.
Show ip access-list	Show only the IP access lists configured on the router.
Show ip interface	Show which interfaces have access list set.
Show running config	Show the access lists and which interfaces have access list set.

Table (1): commands used for monitoring ACL.

Experiment 8:

Introduction to switches & Virtual LANs (VLANs)

1. Objectives :

- Create a basic switch configuration.
- Learn how VLAN mechanism partitions a LAN.
- Learn how to configure a VLAN.
- Verify and test configurations using show commands, and ping.

2. Devices and Equipments :

- PCs with an Ethernet 10/100 NIC installed.
- Cisco 2960 Switches.
- Ethernet straight-through, crossover, and console cables.

3. Theoretical Background :

Switches are dedicated, specialized computers that contain a central processing unit (CPU), random access memory (RAM), and an operating system. Switches usually have several ports that hosts can connect to, as well as specialized ports for the purpose of management. Switches can be managed and the configuration can be viewed and changed through the console port. Switches typically have no power switch to turn them on and off. They simply connect or disconnect from a power source. There are several switches models from Cisco with 12-port or 24-port or 48-port. Cisco catalyst 2960 switch model with 24 port will used in this lab.

3.13 Overview of Cisco Switch Hardware

As shown in Figure (1); the front panel of 2960 switch has several LEDs to help monitor system activity and performance:

- System LED
- Remote Power Supply (RPS) LED
- Port Mode LEDs
- Port Status LEDs

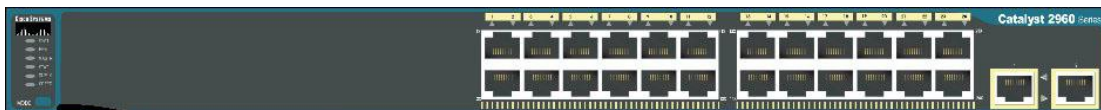


Figure (1): Front panel of 2960 switch.

The System LED shows whether the system is receiving power and functioning correctly. The RPS LED indicates whether or not the remote power supply is in use. The Mode LEDs indicate the state of the

Mode button. The modes are used to determine how the Port Status LEDs are interpreted. To select or change the port mode, press the Mode button repeatedly until the Mode LEDs indicate the desired mode.

Once the power cable is connected, the switch initiates a series of tests called the power-on self test (POST). The System LED indicates the success or failure of POST. If the System LED is off but the switch is plugged in, then POST is running. If the System LED is green, then POST was successful. If the System LED is amber, then POST failed. POST failure is considered to be a fatal error. Reliable operation of the switch should not be expected if POST fails.

The Port Status LEDs also change during POST. The Port Status LEDs turn amber for about 30 seconds as the switch discovers the network topology and searches for loops. If the Port Status LEDs turn green, the switch has established a link between the port and a target, such as a computer. If the Port Status LEDs turn off, the switch has determined that nothing is plugged into the port.

3.14 VLANs Overview

In this part of the experiment, we will introduce VLANs. A VLAN allows a network administrator to create groups of logically networked devices that act as if they are on their own independent network, even if they share a common infrastructure with other VLANs.

When you configure a VLAN, you can name it to describe the primary role of the users for that VLAN. As another example, all of the student computers in a school can be configured in the "Student" VLAN. Using VLANs, you can logically segment switched networks based on functions, departments, or project teams. This improves the performance and manageability of LANs. VLAN provides network administrators flexible control over traffic associated with devices in the LAN.

VLANs allow multiple IP networks and subnets to exist on the same switched network. For computers to communicate on the same VLAN, each must have an IP address and a subnet mask that is consistent for that VLAN.

3.15 Types of VLANs

There are several types of VLANs: a default VLAN, a management VLAN, native VLANs, user/data VLANs, and voice VLANs.

Default VLAN

All switch ports become a member of the default VLAN after the initial boot up of the switch. This allows any device connected to any switch port to communicate with other devices on other switch ports. The default VLAN for Cisco switches is VLAN 1. VLAN 1 has all the features of any VLAN, except that you cannot rename it and you can not delete it.

Note: Some network administrators use the term "default VLAN" to mean a VLAN other than VLAN 1.

Native VLAN

A native VLAN is assigned to an 802.1Q trunk port. An 802.1Q trunk port supports traffic coming from many VLANs (tagged traffic) as well as traffic that does not come from a VLAN (untagged traffic). The 802.1Q trunk port places untagged traffic on the native VLAN.

Management VLAN

A management VLAN is any VLAN you configure to access the management capabilities of a switch. VLAN 1 would serve as the management VLAN if another unique VLAN define to serve as the management VLAN.

Assigning a management VLAN an IP address and subnet mask allows IP communication between the switches, and also allows any host connected to a port assigned to the management VLAN to connect to the switches.

3.16 Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports are used for managing the physical interface and associated Layer 2 protocols. They do not handle routing or bridging. Switch ports belong to one or more VLANs.

Switch port configure to forward a frame to a specific VLAN. It can configure to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the VLANs to which it can belong

3.17 VLAN Trunks

A trunk is a point-to-point link between two network devices that carries more than one VLAN. A VLAN trunk allows you to extend the VLANs across an entire network. Cisco supports IEEE 802.1Q trunking protocol for coordinating trunks on Fast Ethernet and Gigabit Ethernet interfaces.

A VLAN trunk does not belong to a specific VLAN, rather it is a conduit for VLANs between switches and routers.

802.1Q Frame Tagging

Remember that switches are Layer 2 devices. They only use the Ethernet frame header information to forward packets. The frame header does not contain information about which VLAN the frame should belong to.

Subsequently, when Ethernet frames are placed on a trunk they need additional information about the VLANs they belong to. This is accomplished by using the 802.1Q encapsulation header. This header adds a tag to the original Ethernet frame specifying the VLAN to which the frame belongs.

When the switch receives a frame on a port configured in access mode with a static VLAN, the switch takes apart the frame and inserts a VLAN tag, and then sends the tagged frame out a trunk port.

3.18 Switch Configuration

In order to configure or check the status of a switch, connect a computer to the switch through a console connection in order to establish a communication session.

Note: Since the operating system software IOS used in Cisco switches is same as once in Cisco routers which introduced in the previous experiments; the command modes and the basic CLI commands is nearly the same. For VLAN configuration you need to learn a new command which explained in the next sections.

3.19 Configuring VLANs & Trunks

In this section, you will learn the key Cisco IOS commands needed to create, delete, and verify VLANs and VLAN trunks.

Use the following steps to configure and verify VLANs and trunks on a switched network:

1. Create the VLAN.
2. Assign switch ports to a VLAN statically.
3. Verify VLAN configuration.
4. Enable trunking on the inter-switch connections.
5. Verify Trunk configuration.

1. Add a VLAN

To create a static VLAN on a Cisco Catalyst switch using VLAN global configuration mode. You will configure VLANs with IDs in the normal range. Recall there are two ranges of VLAN IDs. The normal range includes IDs 1 to 1001, and extended range consists of IDs 1006 to 4094. VLAN 1 and 1002 to 1005 are reserved ID numbers.

The Cisco IOS commands used to add a VLAN to a switch.

<code>switch#configure terminal</code>	Enter global configuration mode.
<code>switch(config)#vlan vlan id</code>	Create a VLAN. Vlan id is the VLAN number that is to be created. Switches to VLAN configuration mode for VLAN
<code>switch(config-vlan)#name vlan name</code>	(Optional) Specify a unique VLAN name to identify the VLAN.
<code>switch(config-vlan)#end</code>	Return to privileged EXEC mode.

2. Assign a Switch Port

After you have created a VLAN, assign one or more ports to the VLAN. When you manually assign a switch port to a VLAN, it is known as a static access port. A static access port can belong to only one VLAN at a time.

switch# configure terminal	Enter global configuration mode.
switch(config)# interface <i>interface id</i>	Enter the interface to assign the VLAN.
switch(config-if)# switchport mode access	Define the VLAN membership mode for the port.
switch(config-if)# switchport access vlan <i>vlan id</i>	Assign the port to a VLAN.
switch(config-if)# end	Return to privileged EXEC mode.

3. Verify VLAN Configuration

After you configure the VLAN, you can validate the VLAN configurations using Cisco IOS *show* commands.

You can use the *show vlan brief* command with the following syntax

```
switch# show vlan [brief | id vlan-id | name vlan-name | summary]
```

Also you can use the **show interfaces** command that determine which VLAN is assigned to a specific port (interface).

```
switch# show interfaces [interface-id | vlan vlan-id] | switchport
```

4. Manage Port Memberships

Remove VLAN

To remove a VLAN from specific interface; you can use the *no switchport access vlan* command (with the syntax shown below) in interface configuration mode. After this command the VLAN (for example VLAN 20) will still active. It has only been removed from the interface (the access VLAN for this interface will be reset to VLAN 1).

```
switch#configure terminal  
switch(config)#interface interface id  
switch(config-if)#no switchport access vlan  
switch(config-if)#end
```

Reassign the VLAN to Another Port

Static access port can only have one VLAN. So you do not need to first remove a port from a VLAN to change its VLAN membership. When you reassign a static access port to an existing VLAN, the VLAN is automatically removed from the previous port.

5. Delete VLANs

To remove a specific VLAN from the system use the global configuration command

```
switch(config)#no vlan vlan-id
```

Alternatively, the entire *vlan.dat* file can be deleted using the command

```
switch#delete flash:vlan.dat
```

After the switch is reloaded, the previously configured VLANs will no longer be present. This effectively places the switch into its "factory default" concerning VLAN configurations.

6. Configure an 802.1Q Trunk

To configure a trunk on a switch port, use the *switchport mode trunk* command. The Cisco IOS command syntax (shown below) to specify a native VLAN other than VLAN 1.

switch# configure terminal	Enter global configuration mode.
switch(config)# interface interface id	Enters the interface configuration mode.
switch(config-if)# switchport mode trunk	Force the link connecting the switches to be a trunk link.
switch(config-if)# switchport trunk native vlan vlan id	Specify another VLAN as the native VLAN for untagged for IEEE 802.1Q trunks
switch(config-if)# end	Return to privileged EXEC mode.

Managing a Trunk Configuration

In the table below, the commands to reset the allowed VLANs and the native VLAN of the trunk to the default state are shown. The command to reset the switch port to an access port and, in effect, deleting the trunk port is also shown.

<code>switch(config-if)#no switchport trunk allowed vlan</code>	Use this command in the interface configuration mode to reset all of the VLANs configured on the trunk
<code>switch(config-if)#no switchport trunk native vlan</code>	Use this command in the interface configuration mode to reset the native VLAN back to VLAN 1.
<code>switch(config-if)#switchport mode access</code>	Use this command in the interface configuration mode to reset the trunk port interface back to a static access mode port.

Reset Example

In the example below; the commands used to reset all trunking characteristics of a trunking interface to the default settings.

```
switch(config-if)#no switchport trunk allowed vlan
switch(config-if)#no switchport trunk native vlan
```

The *show interfaces f0/1 switchport* command reveals that the trunk has been reconfigured to a default state.

Remove Example

The example below shows the commands used to remove the trunk feature from the F0/1 switch port on switch S1.

```
switch(config-if)#switchport mode access
```

The *show interfaces f0/1 switchport* command reveals that the F0/1 interface is now in static access mode.

Experiment 9:

Inter-VLAN Routing

1. Objectives :

- Describe the three primary options for enabling inter-VLAN routing.
- Configure traditional inter-VLAN routing.
- Configure router-on-a-stick inter-VLAN routing.
- Configure inter-VLAN routing using Layer 3 switching.

2. Devices and Equipments :

- PCs with an Ethernet 10/100 NIC installed.
- Cisco 2960 Switches.
- Ethernet straight-through, crossover, and console cables.

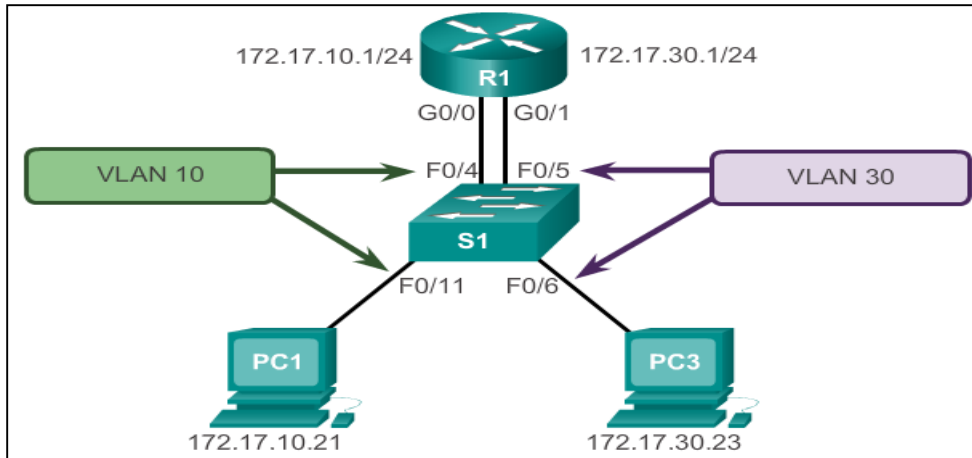
3. Theoretical Background :

After configuring VLANs, the next step will be to enable devices from different VLANs to communicate with each other, and as those devices are in different broadcast domain there should be a mechanism to enable their communication and that mechanism called the Inter-VLAN routing. There are three ways to enable Inter-VLAN routing in a network:

- The traditional Inter-VLAN routing.
- The Router-on-Stick method.
- The Layer-3 switch method.

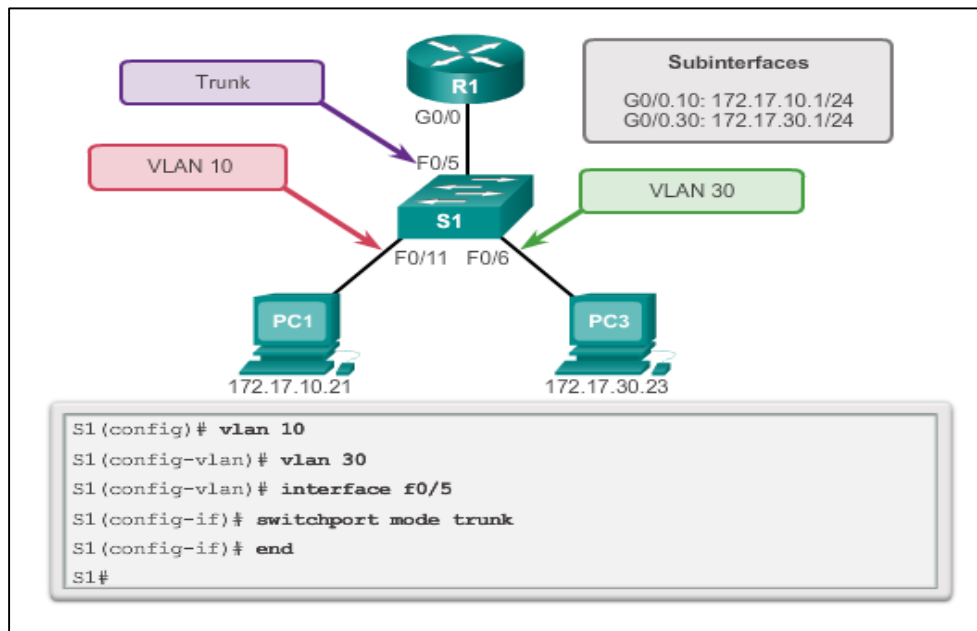
3.1 The traditional Inter-VLAN routing.

In this method each VLAN was considered a separated physical LAN and connected to a different physical router interface. Packets would arrive on the router through one through interface, be routed and leave through another. Because the router interfaces were connected to VLANs and had IP addresses from that specific VLAN, routing between VLANs was achieved. This solution is simple straightforward solution and need no special configuration, but it is not practical for large networks with large number of VLANs since it will require many router interfaces.



3.2 Router-on-a-Stick Inter-VLAN Routing.

In this method One of the router's physical interfaces is configured as a 802.1Q trunk port so it can understand VLAN tags and logical subinterfaces are created; one subinterface per VLAN, each subinterface is configured with an IP address from the VLAN it represents. VLAN members (hosts) are configured to use the subinterface address as a default gateway. Only one of the router's physical interface is used.



```
R1 (config)# interface g0/0.10
R1 (config-subif)# encapsulation dot1q 10
R1 (config-subif)# ip address 172.17.10.1 255.255.255.0
R1 (config-subif)# interface g0/0.30
R1 (config-subif)# encapsulation dot1q 30
R1 (config-subif)# ip address 172.17.30.1 255.255.255.0
R1 (config)# interface g0/0
R1 (config-if)# no shutdown
```

3.3 Multilayer Switch Inter-VLAN Routing.

Multilayer switches can perform Layer 2 and Layer 3 functions, replacing the need for dedicated routers. Multilayer switches support dynamic routing and inter-VLAN routing. The multilayer switch must have IP routing enabled.

Multilayer switching is more scalable than any other inter-VLAN routing implementation. This is because routers have a limited number of available ports to connect to networks. Additionally, for interfaces that are configured as a trunk line, limited amounts of traffic can be accommodated on that line at one time.

With a multilayer switch, traffic is routed internal to the switch device, which means packets are not filtered down a single trunk line to obtain new VLAN-tagging information. A multilayer switch does not, however, completely replace the functionality of a router. Routers support a significant number of additional features, such as the ability to implement greater security controls. Rather, a multilayer switch can be thought of as a Layer 2 device that is upgraded to have some routing capabilities.

