SYIT PRACTICAL
Computer Network

Prof Ismail H P

Maharashtra College

PRACTICAL NO 1A

An address in a block is given as 180.8.17.9. Find the number of addresses in the block, the first address, and the last address

Solution :
The given address is a Class B address therefore n= 16

1) **No of addresses:**
   $N = 2^{32-n}$
   $= 2^{32-16}$
   $= 2^{16}$
   $= 65536$
   Therefore the number of addresses are = 65536 addresses

2) **First address:**
   For class B address netid = 16
   Therefore network mask is 255.255.0.0
   To find the first address we logically AND the given address with the network mask

Given address	180	8	17	9
Network mask	255	255	0	0
AND operation	180	8	0	0

Therefore the first address is 180.8.0.0

3) **Last address:**
   To find the last address we logically OR the given address with the COMPLEMENT of the network mask
   Network mask = 255.255.0.0

Network mask	255	255	0	0
Complement of mask	0	0	255	255

Given address	180	8	17	9
Complement of mask	0	0	255	255
OR operation	180	8	255	255

Therefore the last address is 180.8.255.255

PRACTICAL NO 1B

An organization is granted the block 130.34.12.64/26. The organization needs four sub networks, each with an equal number of hosts. Design the sub networks and find the information about each network.

Solution :
The given address is address of type classless addressing
with n= 26

1) **No of addresses:**

$N = 2^{32-n}$
$= 2^{32-26}$
$= 2^{6}$
$= 64$
Therefore the number of addresses are = 64 addresses

2) **First address:**

For the given case n = 26
Therefore network mask is 255.255.255.192
To find the first address we logically AND the given address with the network mask

Given address	130	34	12	64
Network mask	255	255	255	192
AND operation	130	34	12	64

Therefore the first address is 130.34.12.64

3) **Creating sub-networks:**

In this case we need to create 4 sub-networks with equal number of hosts
Total number of hosts N = 64
Therefore number of hosts in each sub-network $N_{SUB} = 16$

We calculate the sub-netid for each network as follows

$n_{SUB} = n + \log_2(N / N_{SUB})$

$= 26 + \log_2(64/16)$

$= 28$

Therefore the given sub-networks are

Sub-network	First address	Last address
1	130.34.12.64	130.34.12.79
2	130.34.12.80	130.34.12.95
3	130.34.12.96	130.34.12.111
4	130.34.12.112	130.34.12.127

PRACTICAL NO 2

Static Routing

Static Route

1. Static routing method is most trusted by a router.
2. Static routing is not really a routing protocol.
3. Static routes do not dynamically adapt to network changes, are not particularly scalable, and require manual updating to reflect changes.

Static routing has the following advantages

1. There is no bandwidth usage between routers, which means you could possibly save money on WAN links.
2. There is no overhead on the router CPU, which means you could possibly buy a cheaper router than you would use if you were using dynamic routing.
3. It adds security because the administrator can choose to allow routing access to certain networks only.

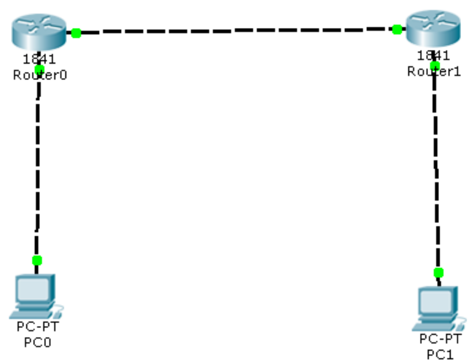
Static routing has the following disadvantages

1. Static routes don't dynamically adapt to network change.
2. If a network is added to the internetwork, the administrator has to add a route to it on all routers—by hand.
3. It's not feasible in large networks because maintaining it would be a full-time job in itself.
4. With static routing, as your network grows, it can be difficult just keep adding static routes makes sure everybody can still get everything.
5. The administrator must really understand the internetwork and how each router is connected in order to configure routes correctly.

There are two different styles to configure an "ip route" command:

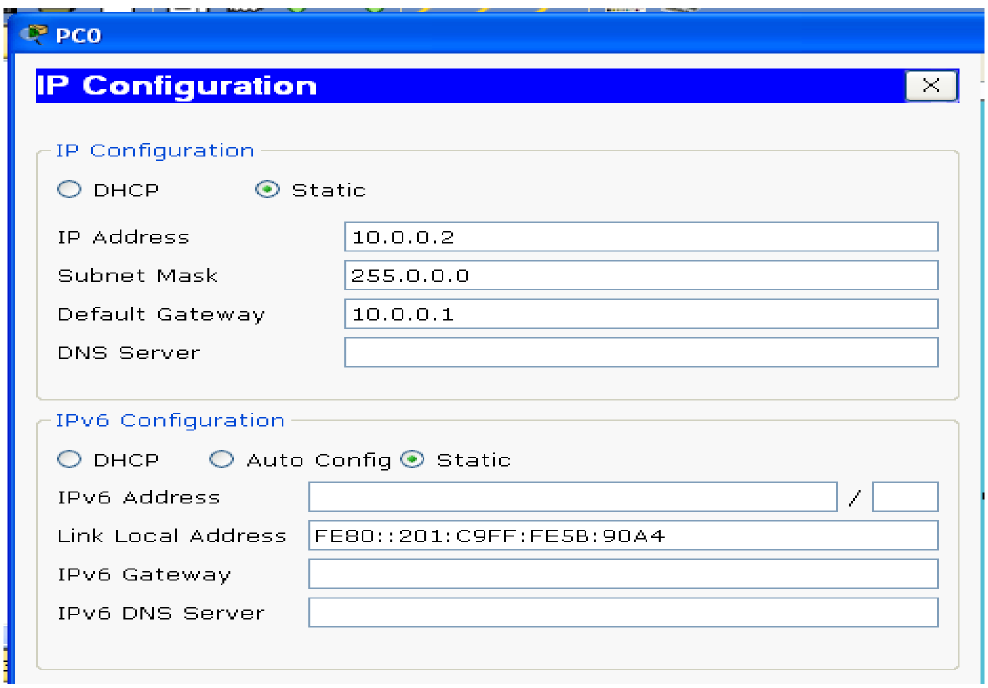
1. Using a next hop IP address
2. Using an outgoing interface

Consider the following network

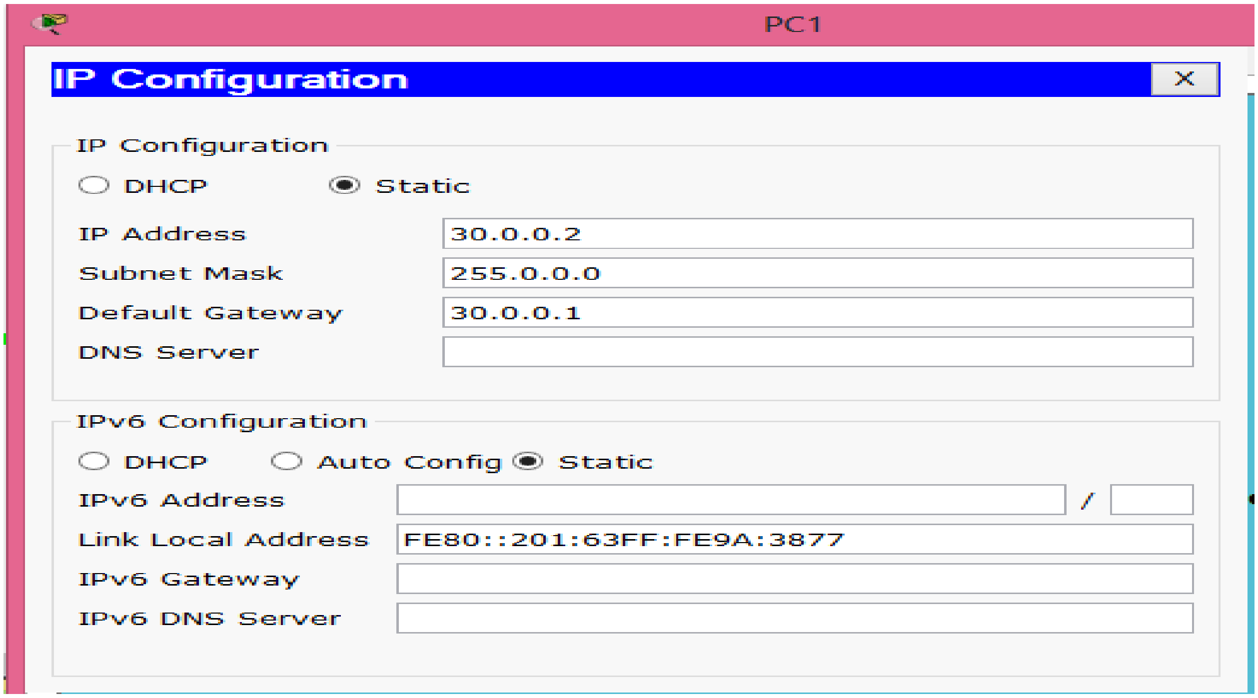


We configure it as follows

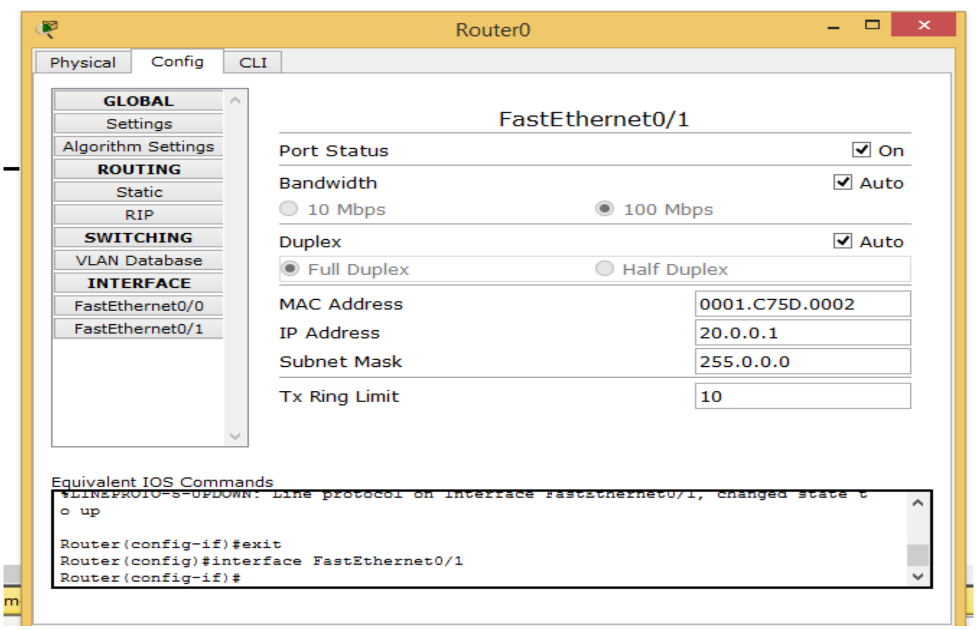
Step 1: (configure PC 0)

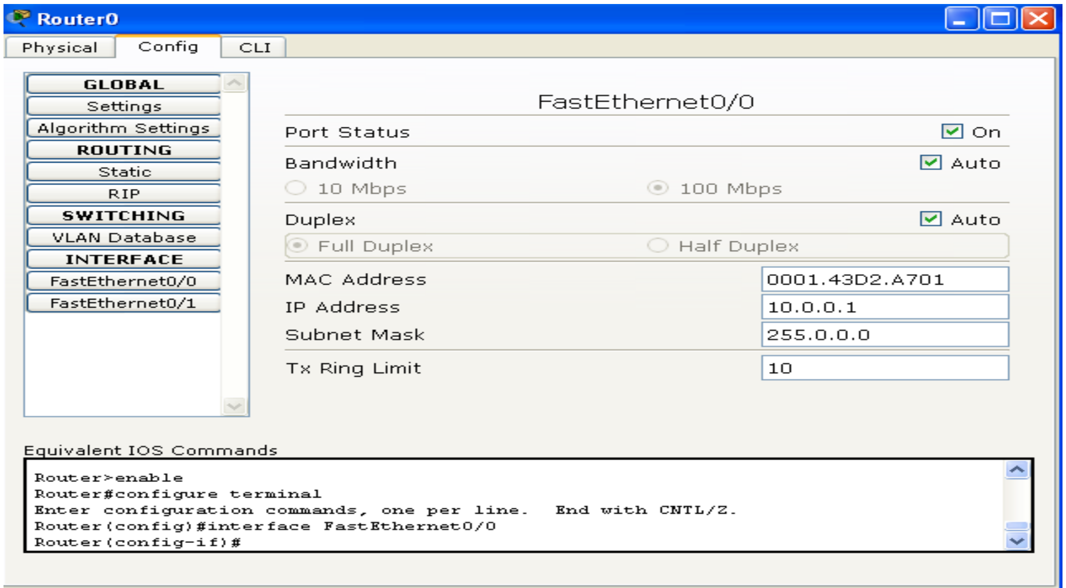


Step 2: (configure PC 1)

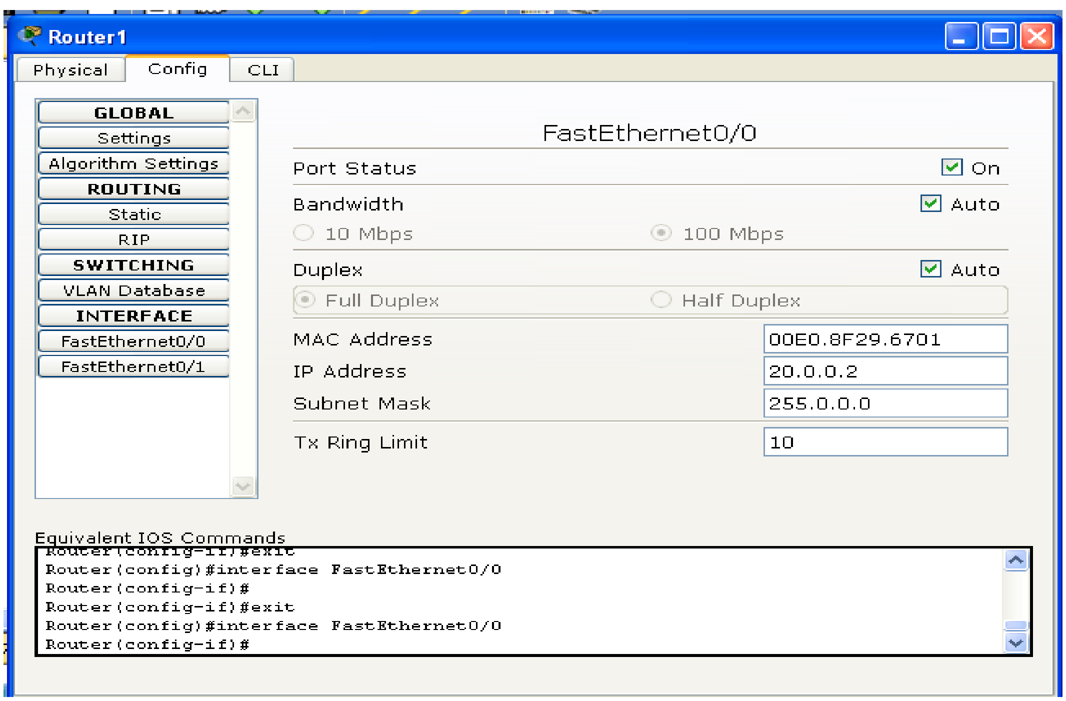


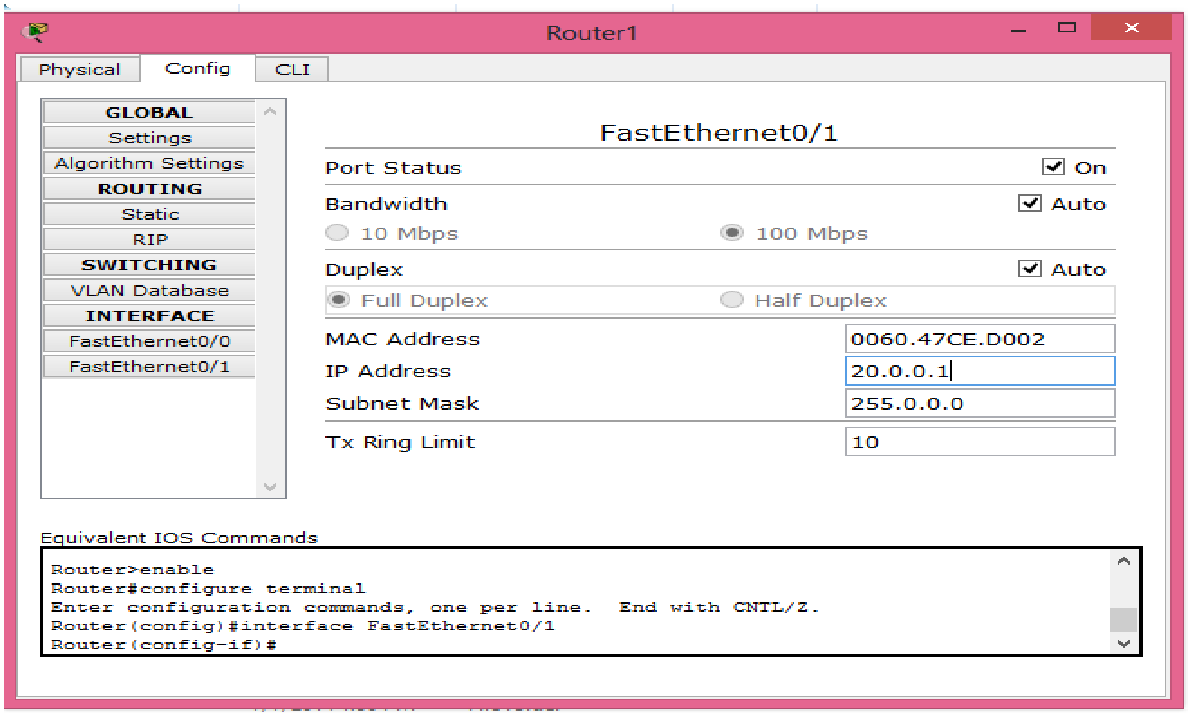
Step 3: (configure Router 0)





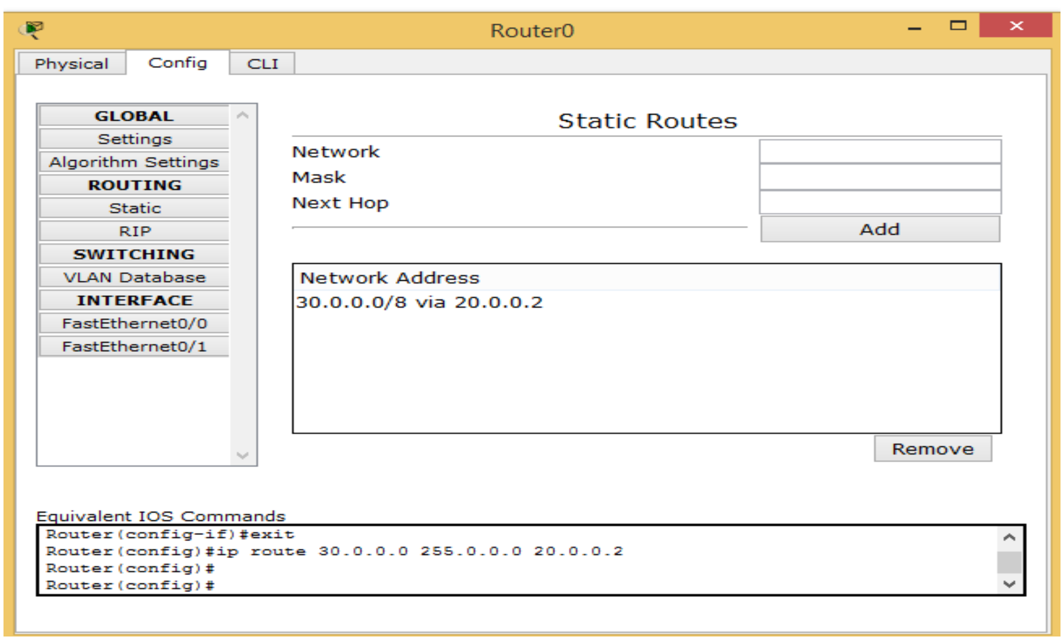
Step 4: (configure Router 1)



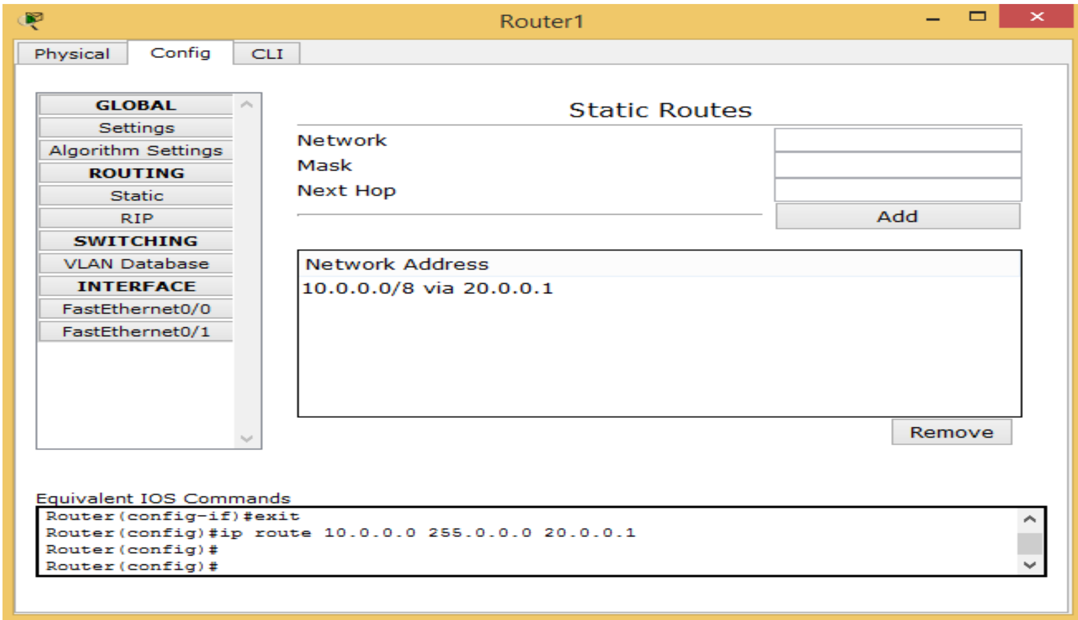


The routing table is configured in the following way

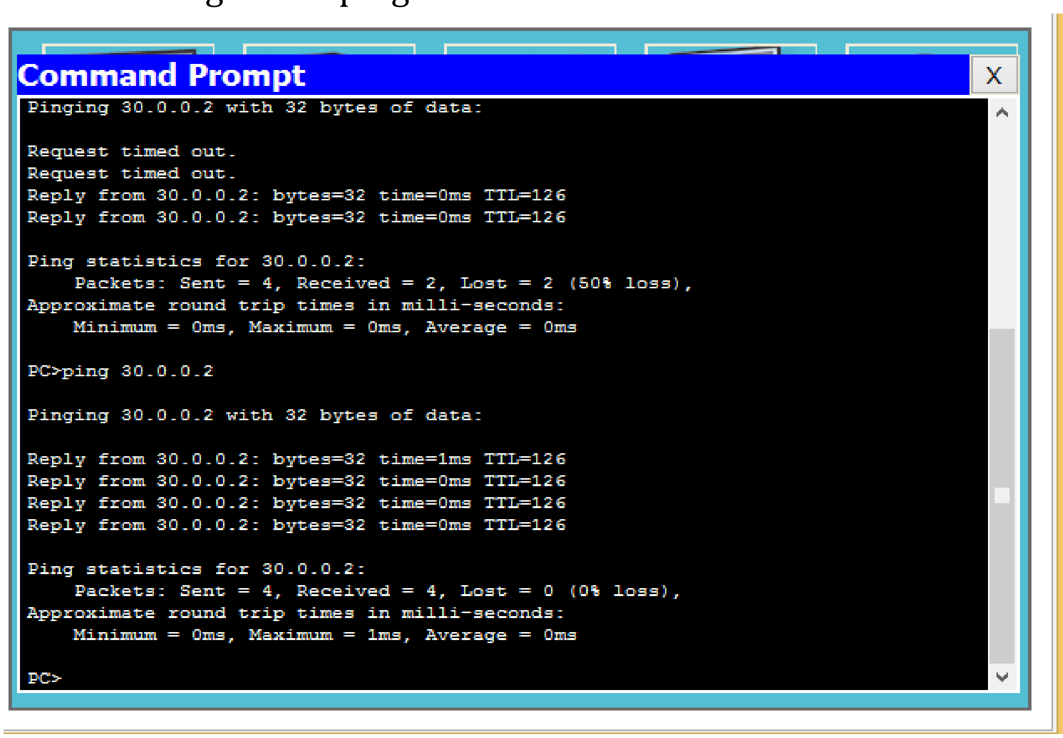
For router 0



For router 1



Now we can give the ping command as shown to check the connectivity



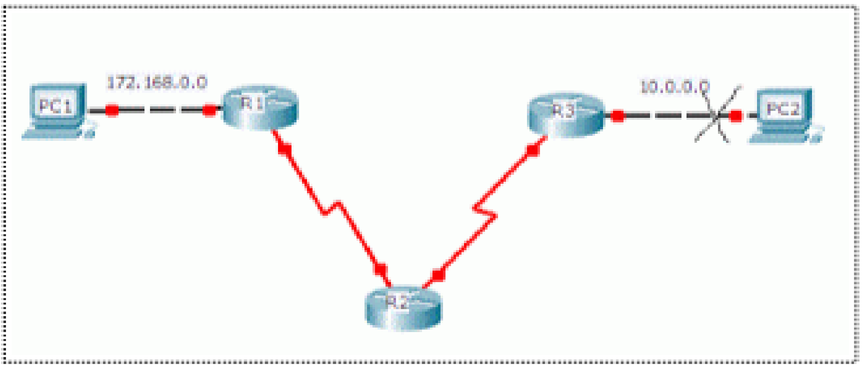
So static routing has been studied

PRACTICAL NO 3

Routing Information Protocol (RIP)

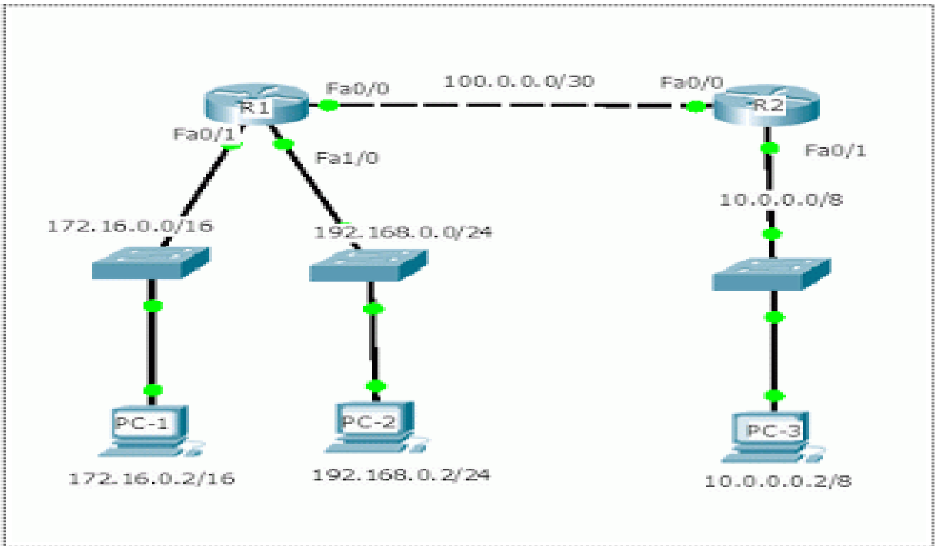
There are two versions of RIP: RIPv1 and RIPv2.
Comparing between RIPv1 and RIPv2
1. Both RIPv1 and RIPv2 have the Administrative distance 120.
2. Both RIPv1 and RIPv2 are distance vector routing protocol.
Both RIPv1 and RIPv2's metric is hop count. Maximum hop count = 15. Max routers = 16.

Consider the following case



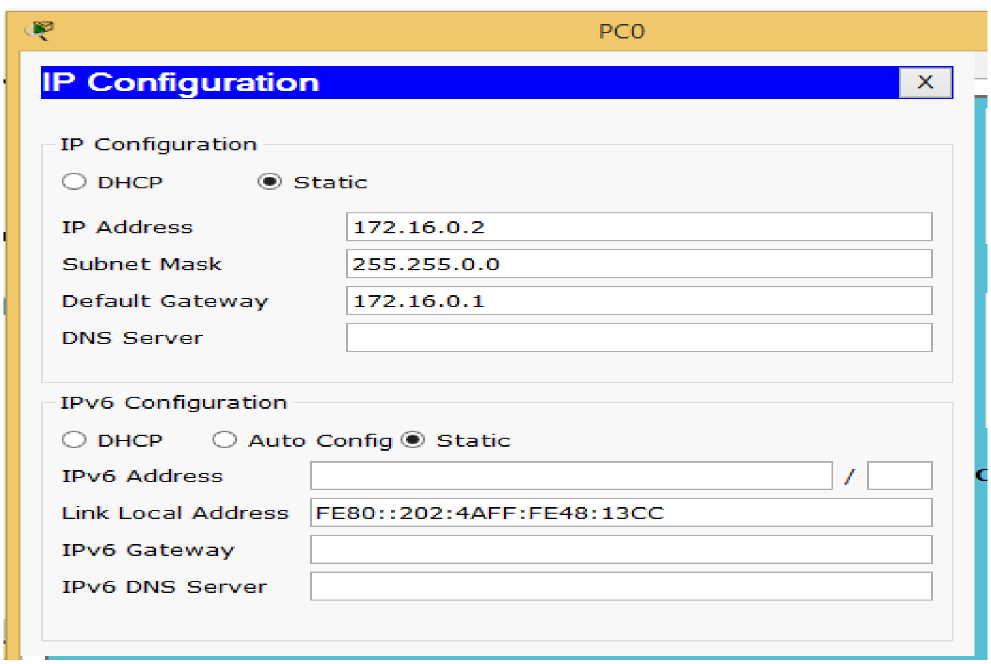
Here all routers are running RIP and network 10.0.0.0 goes down. After hold timer expires, that network will be advertised by metric 16 and everyone will know that the network is down and that network will be seen in routing table as possibly down.
4. Both RIPv1 and RIPv2 send routing updates or complete routing table or broadcast every 30 seconds. i.e. The default routing update period for both version of RIP is 30 seconds. i.e. Both have the same timers.
5. Both RIPv1 and RIPv2 use split horizon to prevent routing loops.
6. Both RIPv1 and RIPv2 are configured with router rip.
7. Network command tells both RIPv1 and RIPv2 to send hellos, out an interface, to find neighbors and to advertise routes.

Consider the following example of RIP using packet tracer

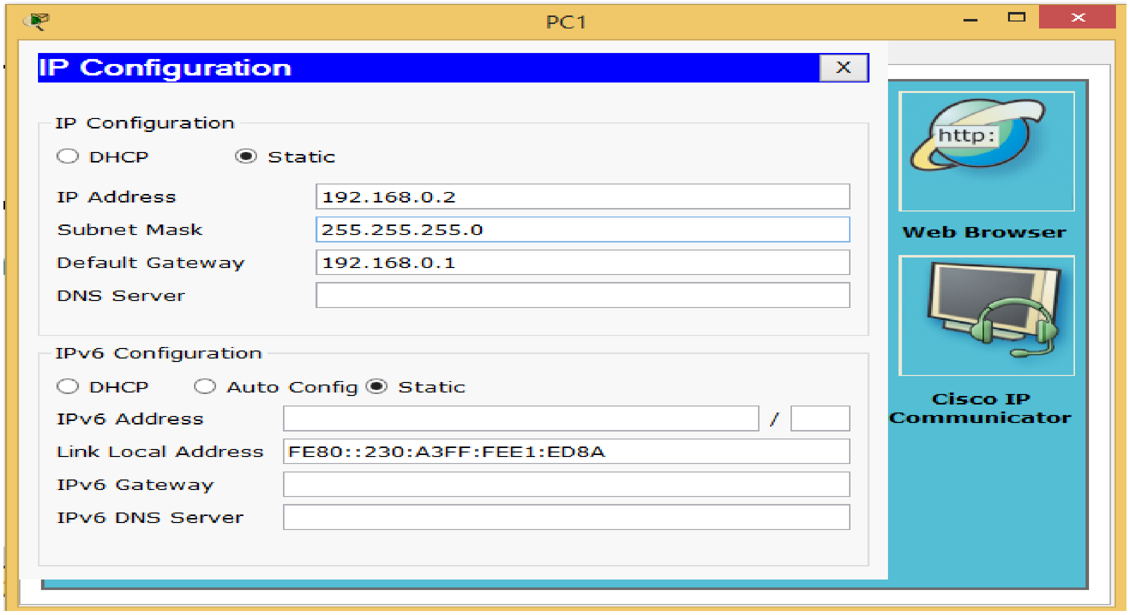


Now we configure the PC's and Routers as follows

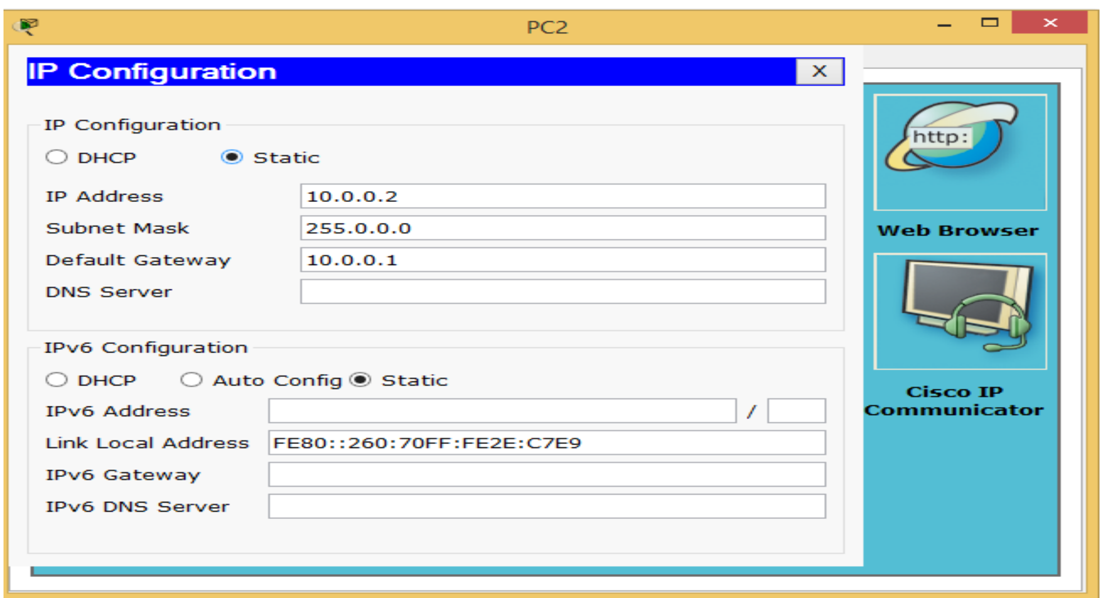
Step 1: (configure PC 0)



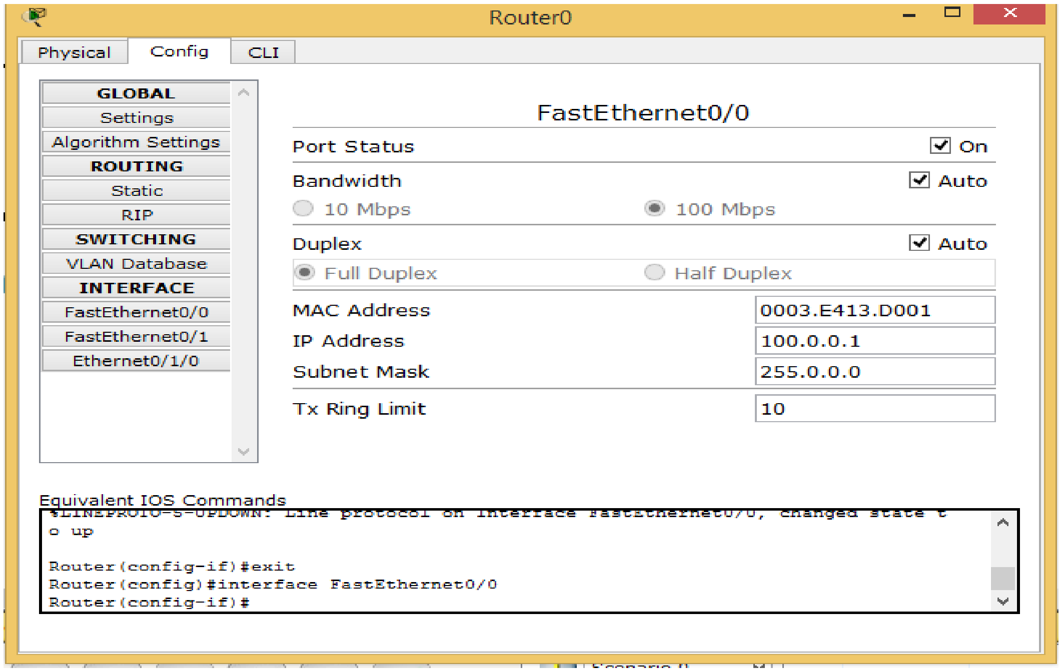
Step 2: (configure PC 1)

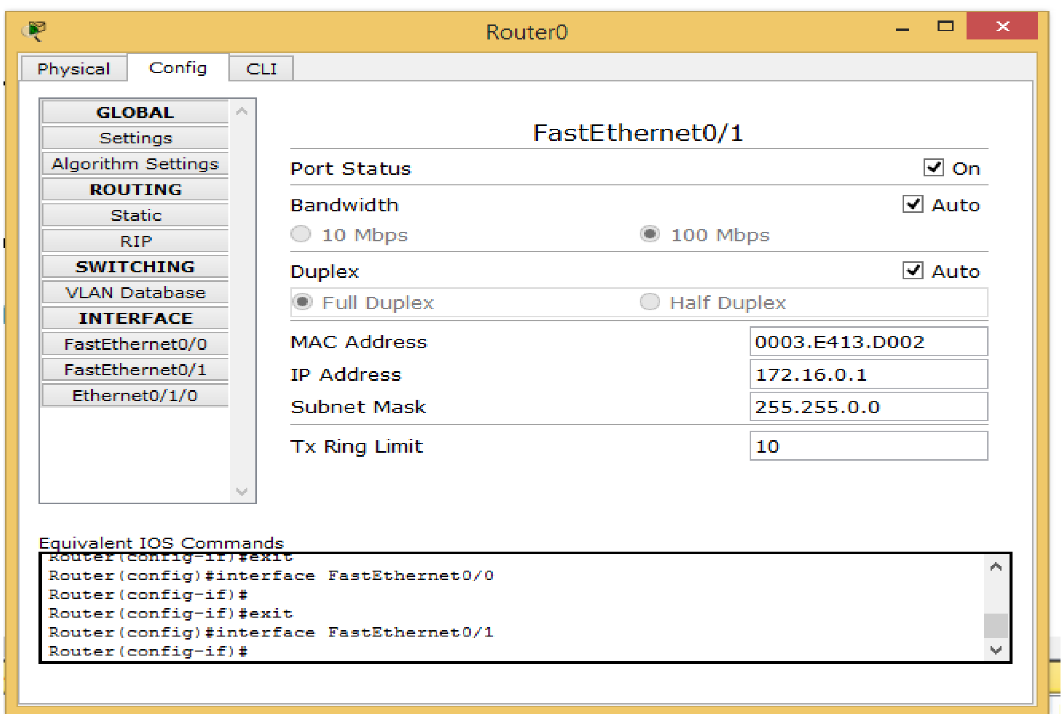


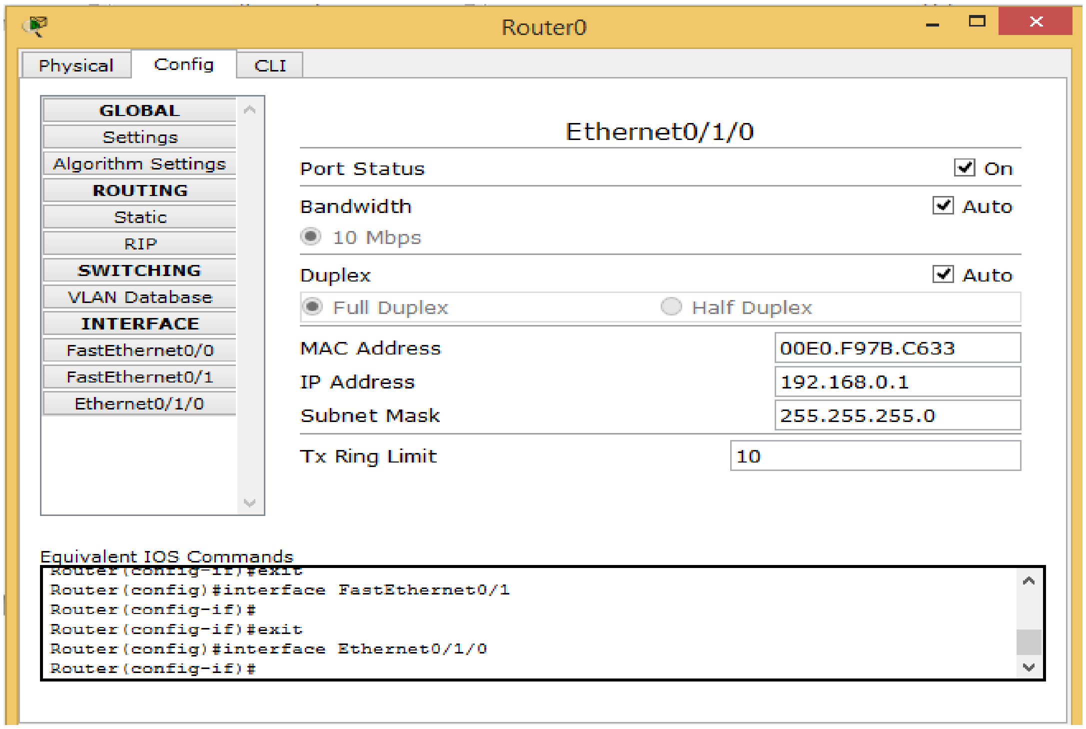
Step 3: (configure PC 2)



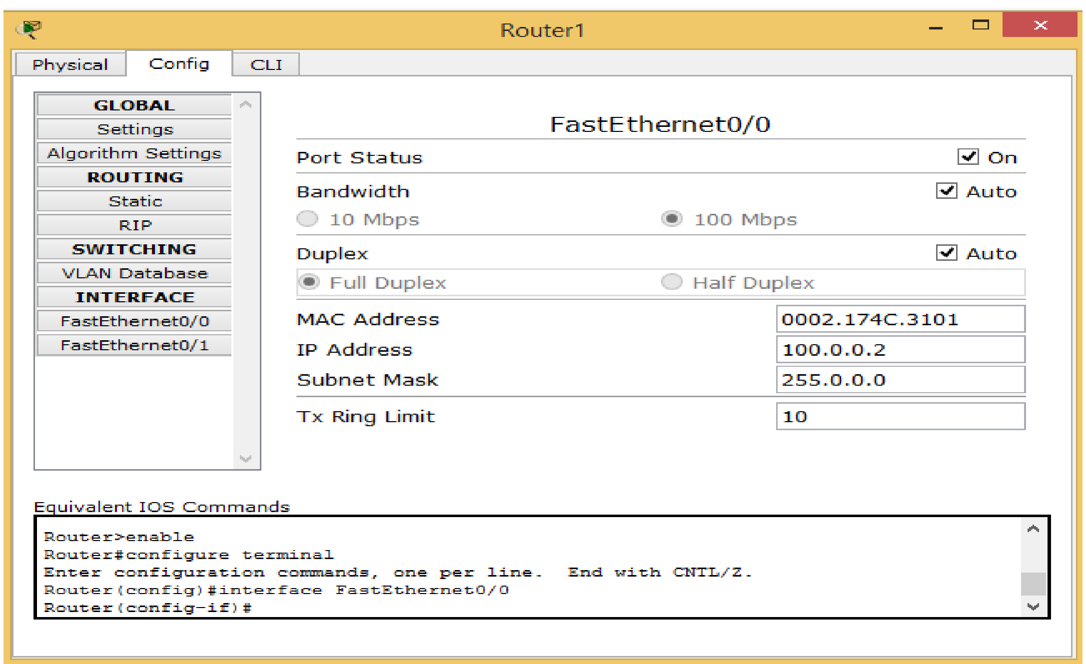
Step 4: (configure Router 0)

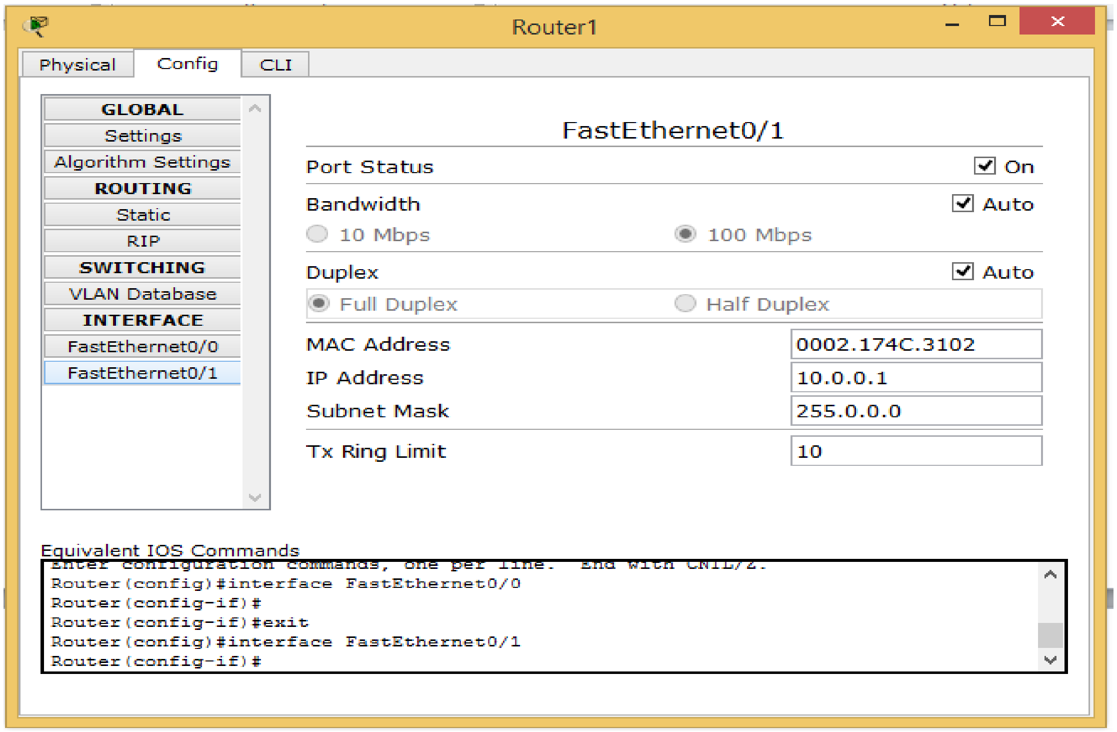






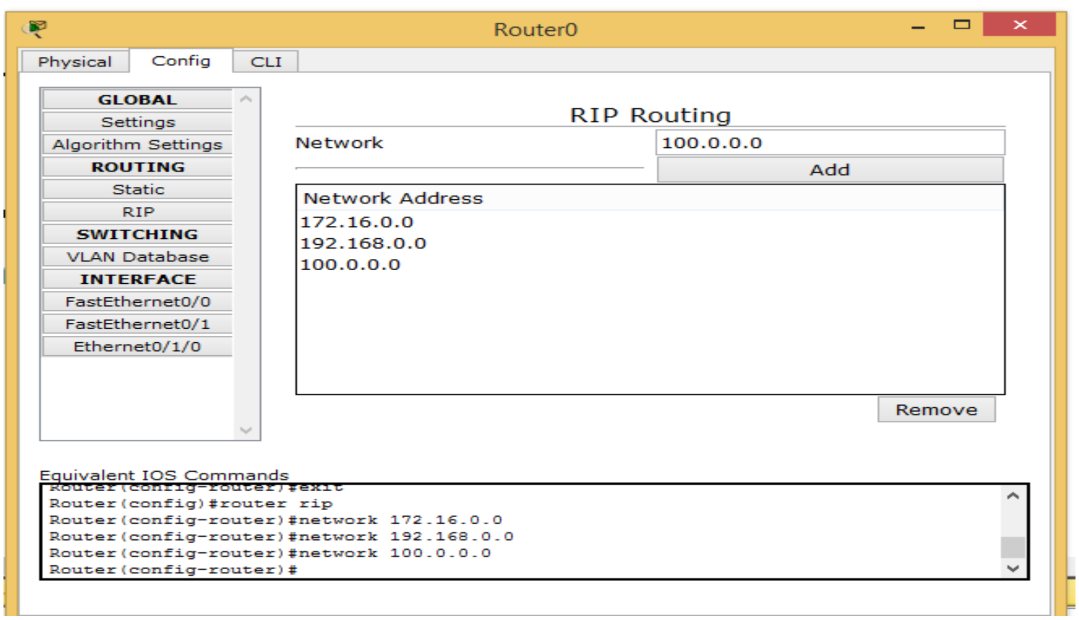
Step 5: (configure Router 1)



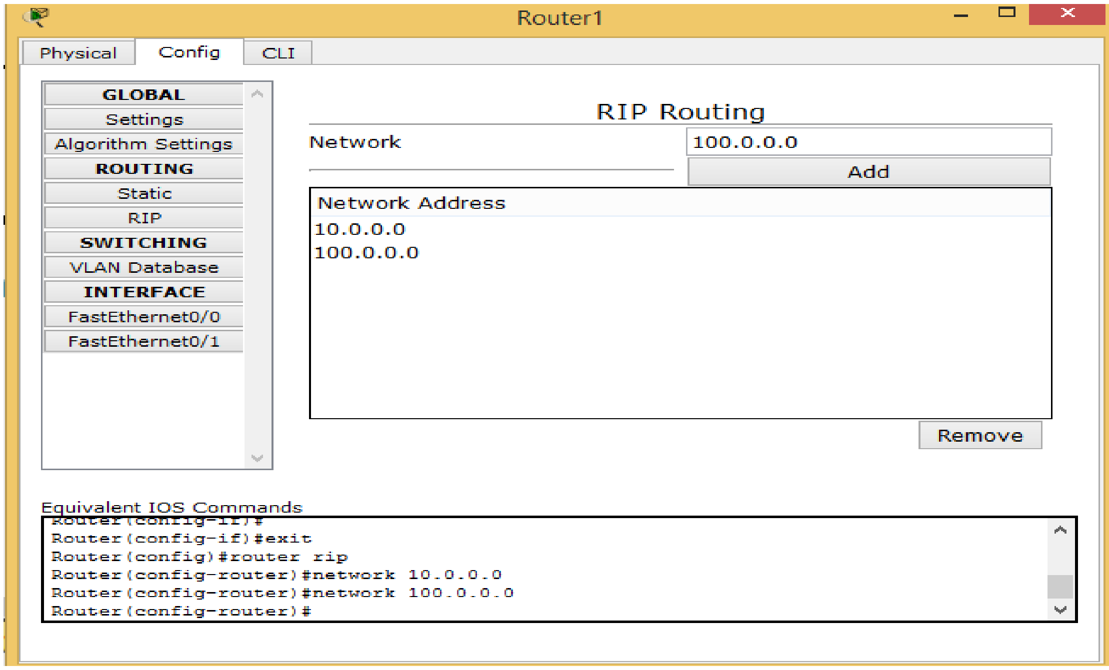


Now we configure the routing table for both the routers

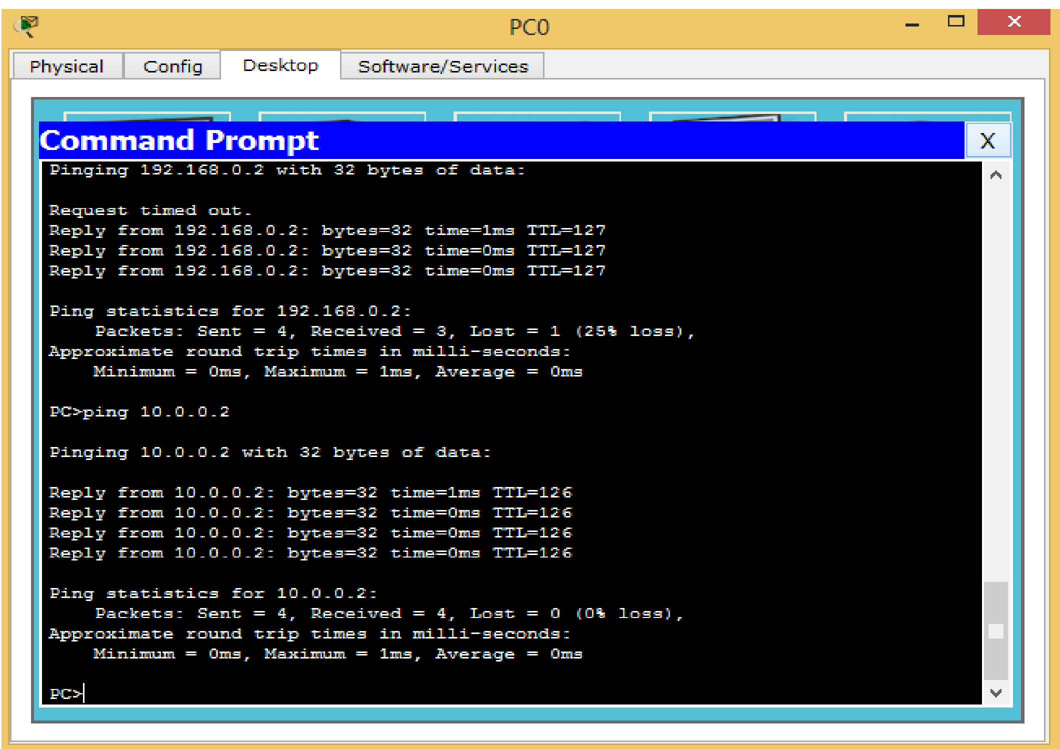
Routing table for Router 1



Routing table for Router 2



Now we use the ping command to check the working

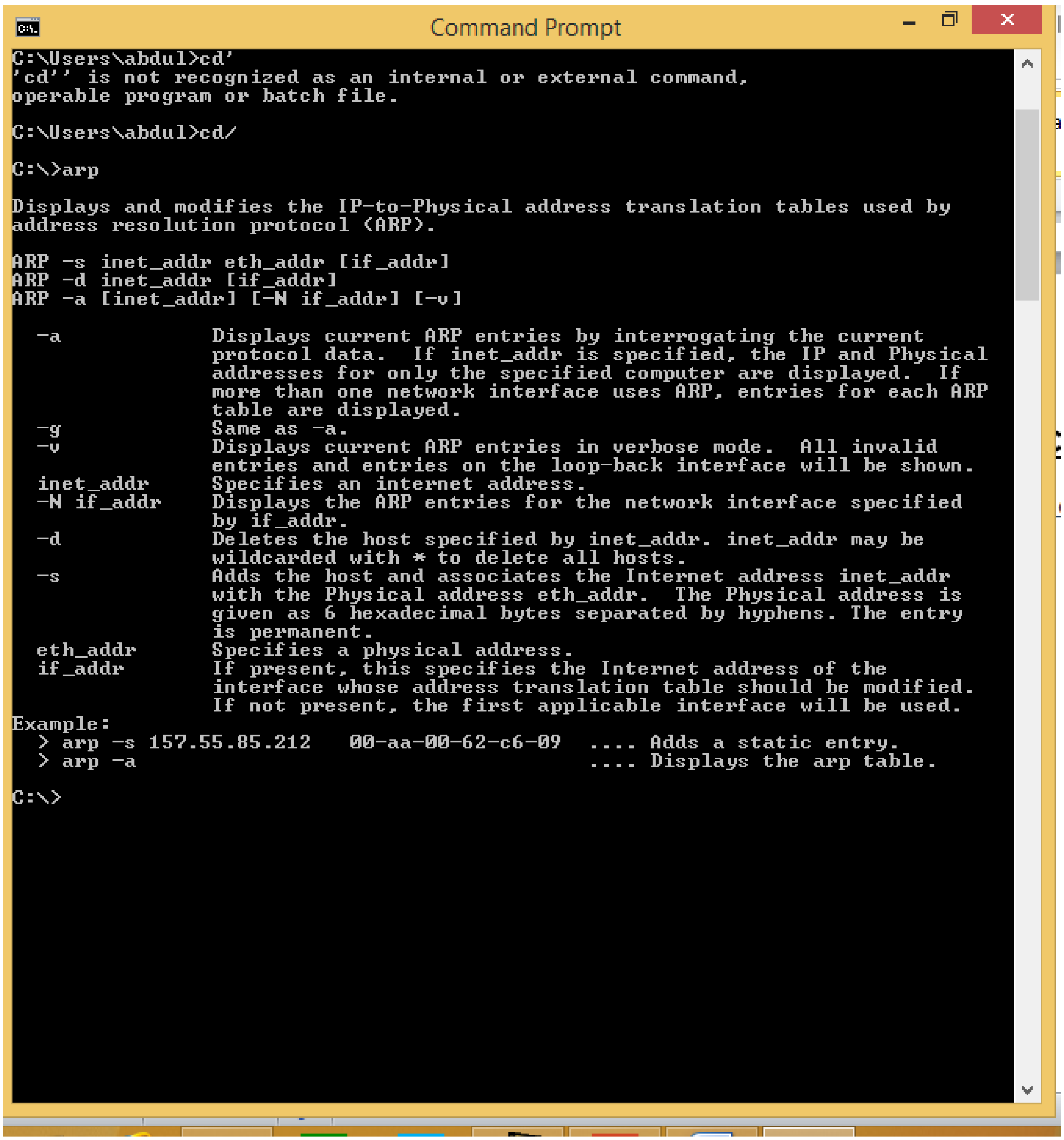


Hence the RIP protocol has been studied

PRACTICAL NO 4

Using the various command utilities

1) **arp** : This diagnostic command displays and modifies the IP-to-Ethernet or Token Ring physical address translation tables used by the Address Resolution Protocol (ARP).



```
C:\Users\abdul>cd'
'cd'' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\abdul>cd/

C:\>arp

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp table.

C:\>
```

2) **hostname**: This diagnostic command prints the name of the host on which the command is used.

Syntax

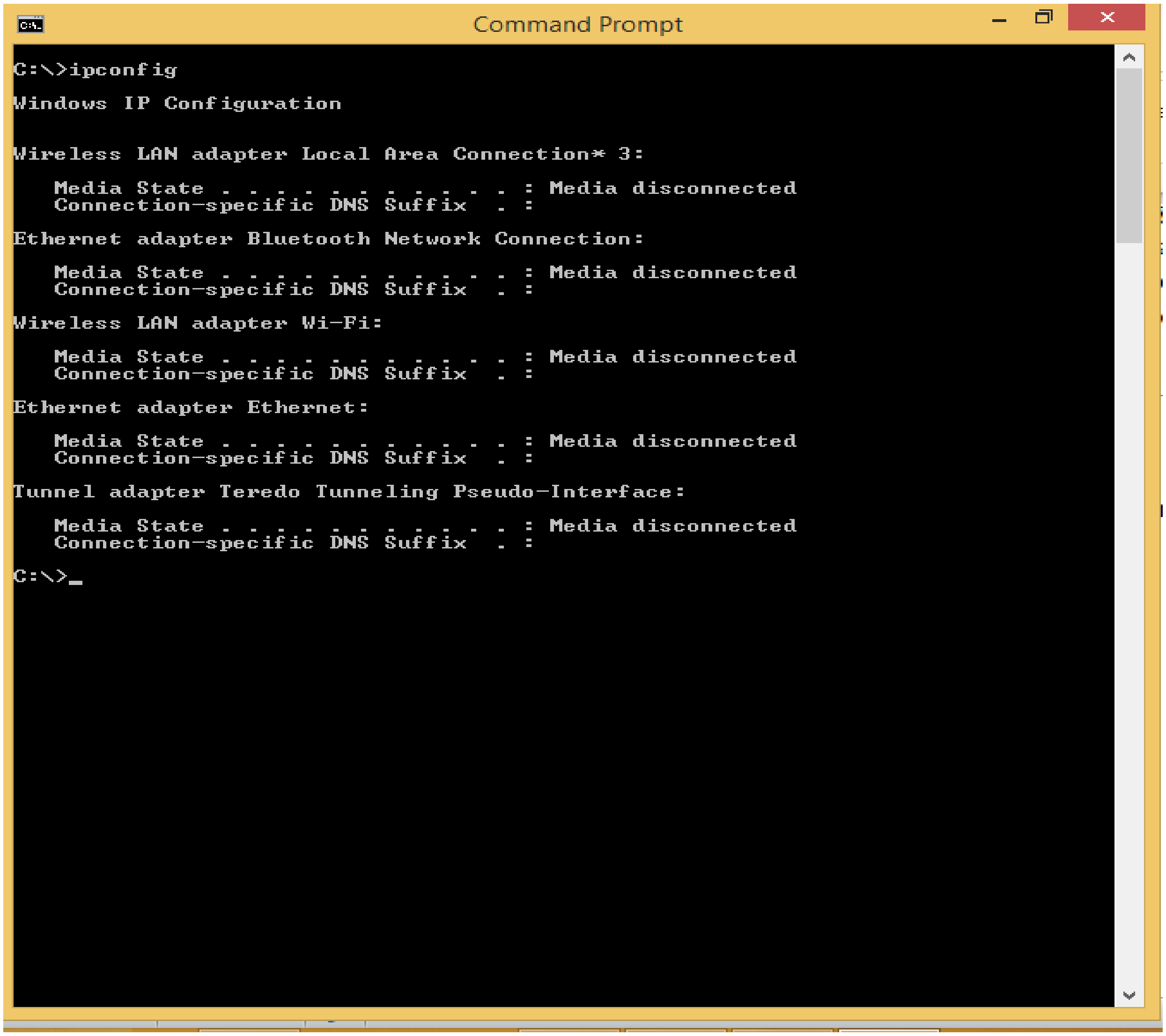
hostname -- This command has no parameters.

3) ipconfig:

This diagnostic command displays all current TCP/IP network configuration values. This command is useful on computers running DHCP because it enables users to determine which TCP/IP configuration values have been configured by DHCP. If you enter only ipconfig without parameters, the response is a display of all of the current TCP/IP configuration values, including IP address, subnet mask, and default gateway.

Syntax

ipconfig [/all | /renew [adapter] | /release [adapter]



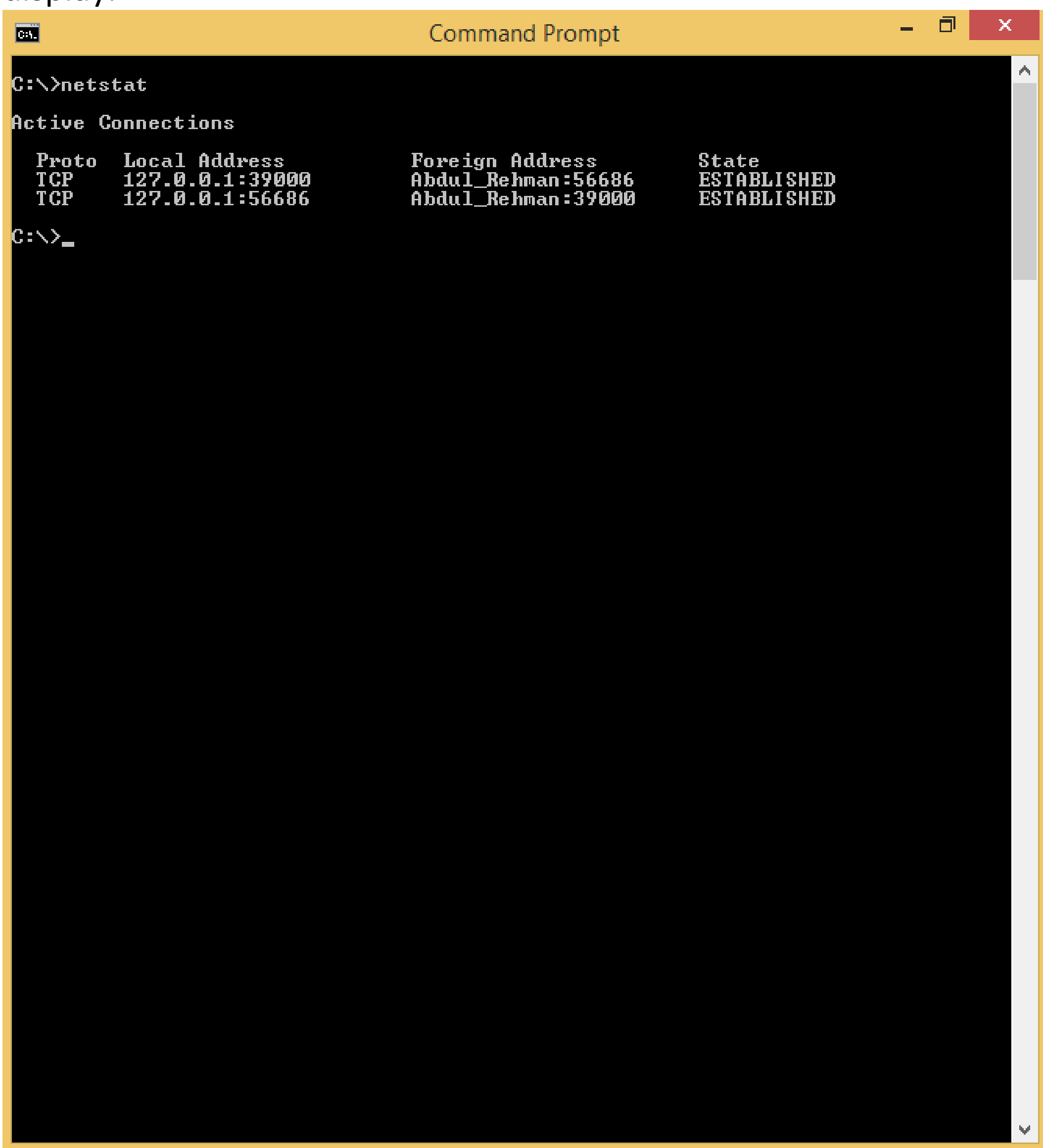
4) netstat:

This diagnostic command displays protocol statistics and current TCP/IP network connection

Syntax

netstat [-a] [-e][-n][-s] [-p protocol] [-r] [interval]

Parameters
-a Displays all connections and listening ports; server connections are usually not shown. -e Displays Ethernet statistics. This can be combined with the -s option. -n Displays addresses and port numbers in numerical form (rather than attempting name lookups). -s Displays per-protocol statistics. By default, statistics are shown for TCP, UDP, ICMP, and IP; the -p option can be used to specify a subset of the default.
-p protocol Shows connections for the protocol specified.
-r Displays the contents of the routing table.
Interval Redisplays selected statistics, pausing interval seconds between each display.



5) ping: This diagnostic command verifies connections to one or more remote computers.

Syntax

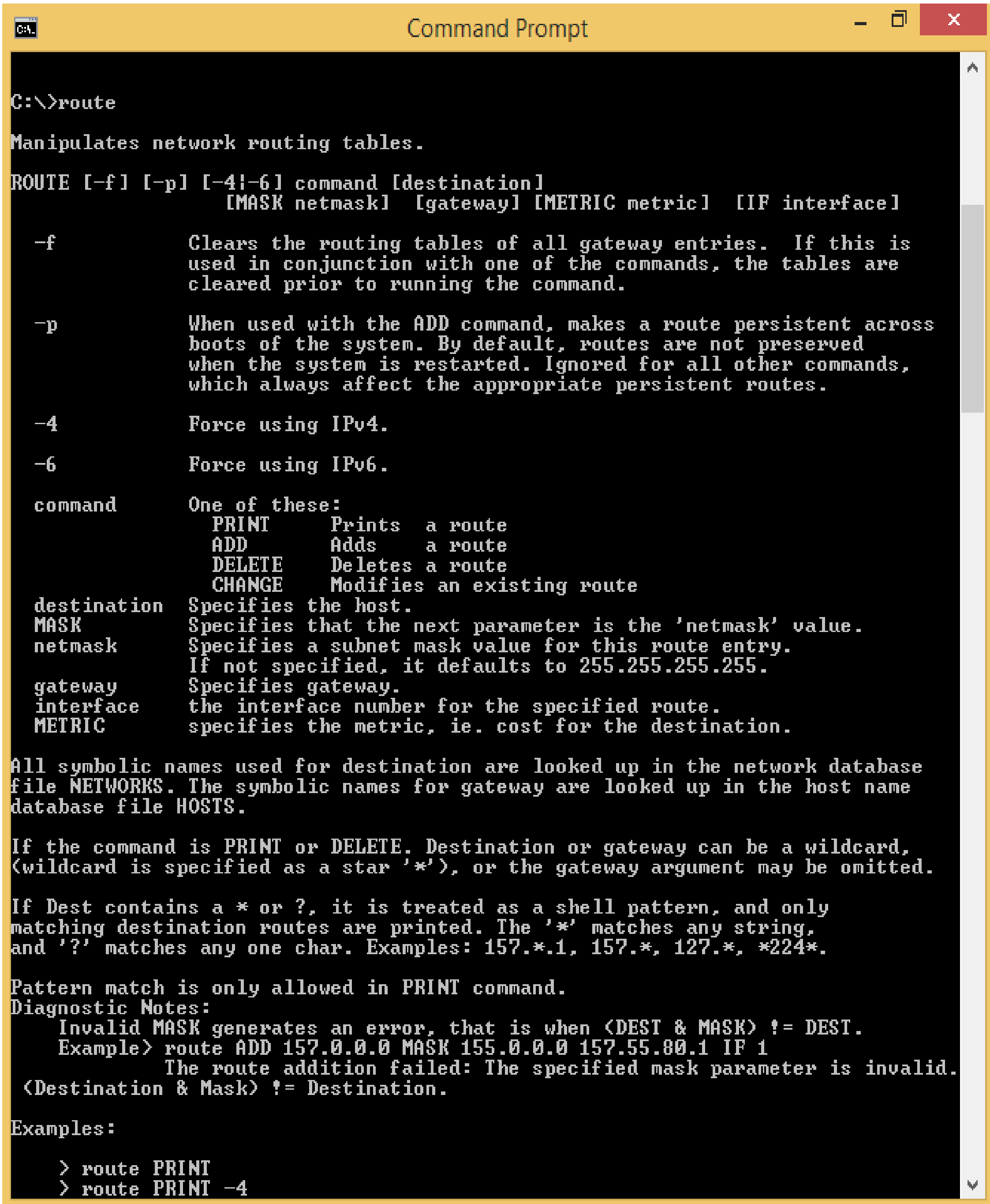
ping [-t] [-a] [-n count] [-l length] [-f] [-i ttl] [-v tos] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] destination-list

6) route:

This diagnostic command manipulates network routing tables.

Syntax

route [-f] [command [destination] [MASK netmask] [gateway] [METRIC metric]]



```
Command Prompt

C:\>route

Manipulates network routing tables.

ROUTE [-f] [-p] [-4|-6] command [destination]
                  [MASK netmask]  [gateway] [METRIC metric]  [IF interface]

  -f           Clears the routing tables of all gateway entries.  If this is
               used in conjunction with one of the commands, the tables are
               cleared prior to running the command.

  -p           When used with the ADD command, makes a route persistent across
               boots of the system. By default, routes are not preserved
               when the system is restarted. Ignored for all other commands,
               which always affect the appropriate persistent routes.

  -4           Force using IPv4.

  -6           Force using IPv6.

  command      One of these:
                 PRINT     Prints  a route
                 ADD       Adds    a route
                 DELETE    Deletes a route
                 CHANGE    Modifies an existing route
  destination  Specifies the host.
  MASK         Specifies that the next parameter is the 'netmask' value.
  netmask      Specifies a subnet mask value for this route entry.
               If not specified, it defaults to 255.255.255.255.
  gateway      Specifies gateway.
  interface    the interface number for the specified route.
  METRIC       specifies the metric, ie. cost for the destination.

All symbolic names used for destination are looked up in the network database
file NETWORKS. The symbolic names for gateway are looked up in the host name
database file HOSTS.

If the command is PRINT or DELETE. Destination or gateway can be a wildcard,
(wildcard is specified as a star '*'), or the gateway argument may be omitted.

If Dest contains a * or ?, it is treated as a shell pattern, and only
matching destination routes are printed. The '*' matches any string,
and '?' matches any one char. Examples: 157.*.1, 157.*, 127.*, *224*.

Pattern match is only allowed in PRINT command.
Diagnostic Notes:
    Invalid MASK generates an error, that is when (DEST & MASK) != DEST.
    Example> route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1 IF 1
             The route addition failed: The specified mask parameter is invalid.
 (Destination & Mask) != Destination.

Examples:

    > route PRINT
    > route PRINT -4
```

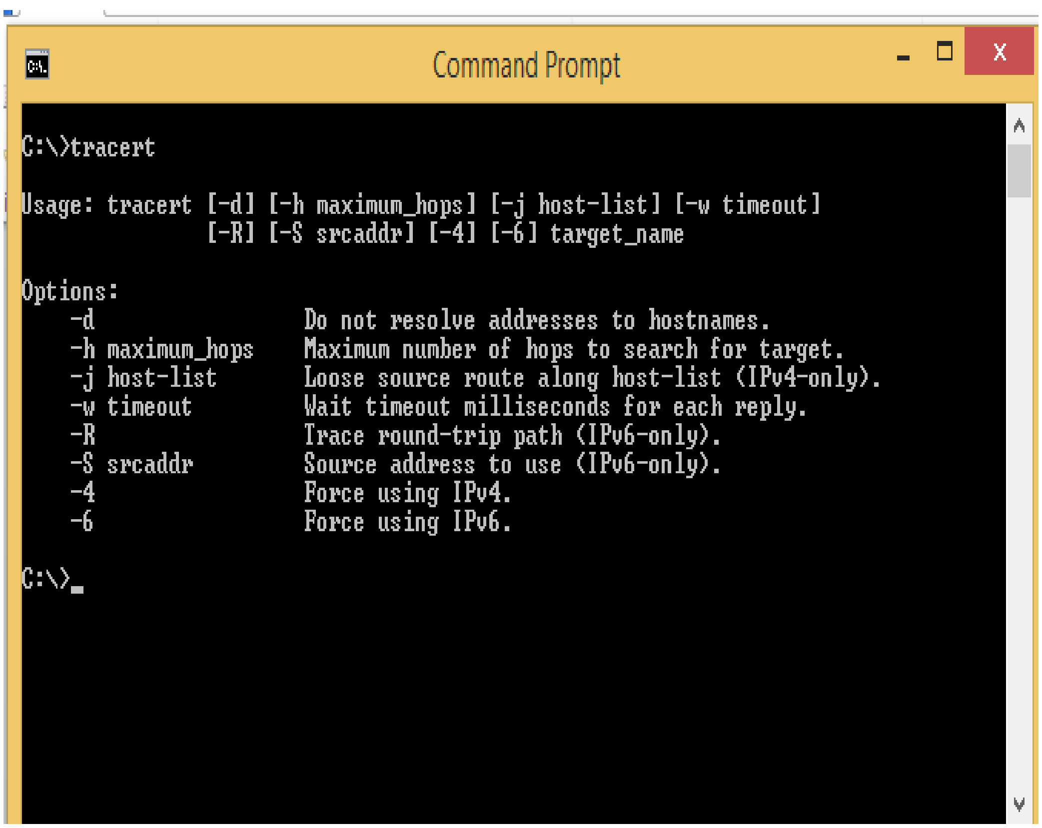
7) tracert:

This diagnostic utility determines the route taken to a destination by sending Internet Control Message Protocol (ICMP) echo packets with varying time-to-live (TTL) values to the destination. Each router along the path is required to decrement the TTL on a packet by at least 1 before forwarding it, so the TTL is effectively a hop count. When the TTL on a packet reaches 0, the router is supposed to send back an ICMP Time Exceeded message to the source computer.

tracert determines the route by sending the first echo packet with a TTL of 1 and incrementing the TTL by 1 on each subsequent transmission until the target responds or the maximum TTL is reached. The route is determined by examining the ICMP Time Exceeded messages sent back by intermediate routers. Notice that some routers silently drop packets with expired TTLs and will be invisible to tracert.

Syntax

tracert[-d] [-h maximum_hops] [-j host-list] [-w timeout] target_name



PRACTICAL NO 5

Configuring DHCP and DNS

DHCP

The **Dynamic Host Configuration Protocol (DHCP)** is a client/server protocol designed to provide the four pieces of information for a diskless computer or a computer that is booted for the first time. DHCP is a successor to BOOTP and is backward compatible with it. Although BOOTP is considered deprecated, there may be some systems that may still use BOOTP for host configuration. The part of the discussion in this chapter that does not deal with the dynamic aspect of DHCP can also be applied to BOOTP.

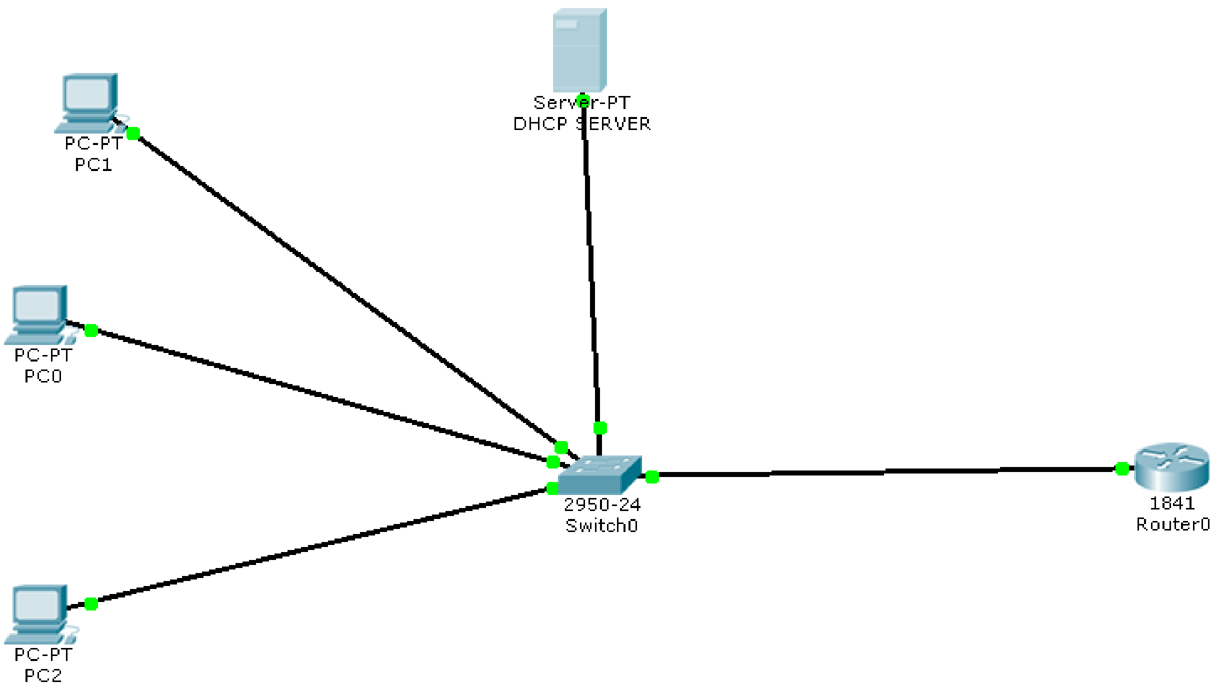
The DHCP client and server can either be on the same network or on different networks. When on same network it works as follows.

1. The DHCP server issues a passive open command on UDP port number 67 and waits for a client.

2. A booted client issues an active open command on port number 68. The message is encapsulated in a UDP user datagram, using the destination port number 67 and the source port number 68. The UDP user datagram, in turn, is encapsulated in an IP datagram. The reader may ask how a client can send an IP datagram when it knows neither its own IP address (the source address) nor the server's IP address (the destination address). The client uses all 0s as the source address and all 1s as the destination address.

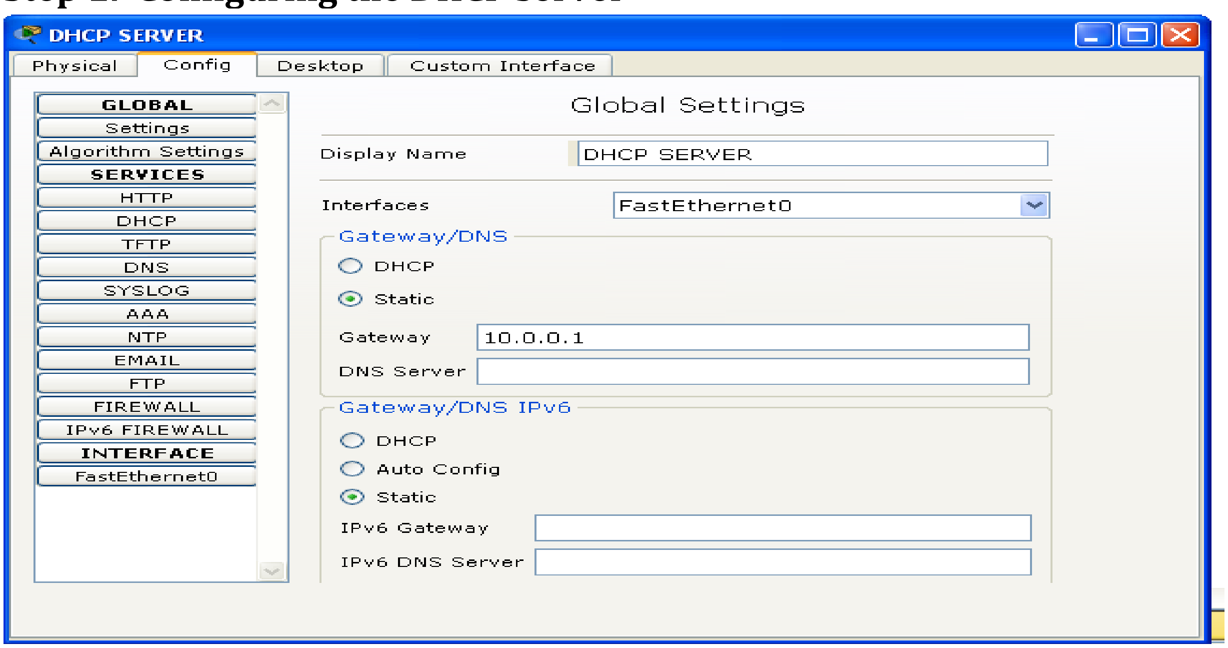
3. The server responds with either a broadcast or a unicast message using UDP source port number 67 and destination port number 68. The response can be unicast because the server knows the IP address of the client. It also knows the physical address of the client, which means it does not need the services of ARP for logical to physical address mapping. However, some systems do not allow the bypassing of ARP, resulting in the use of the broadcast address

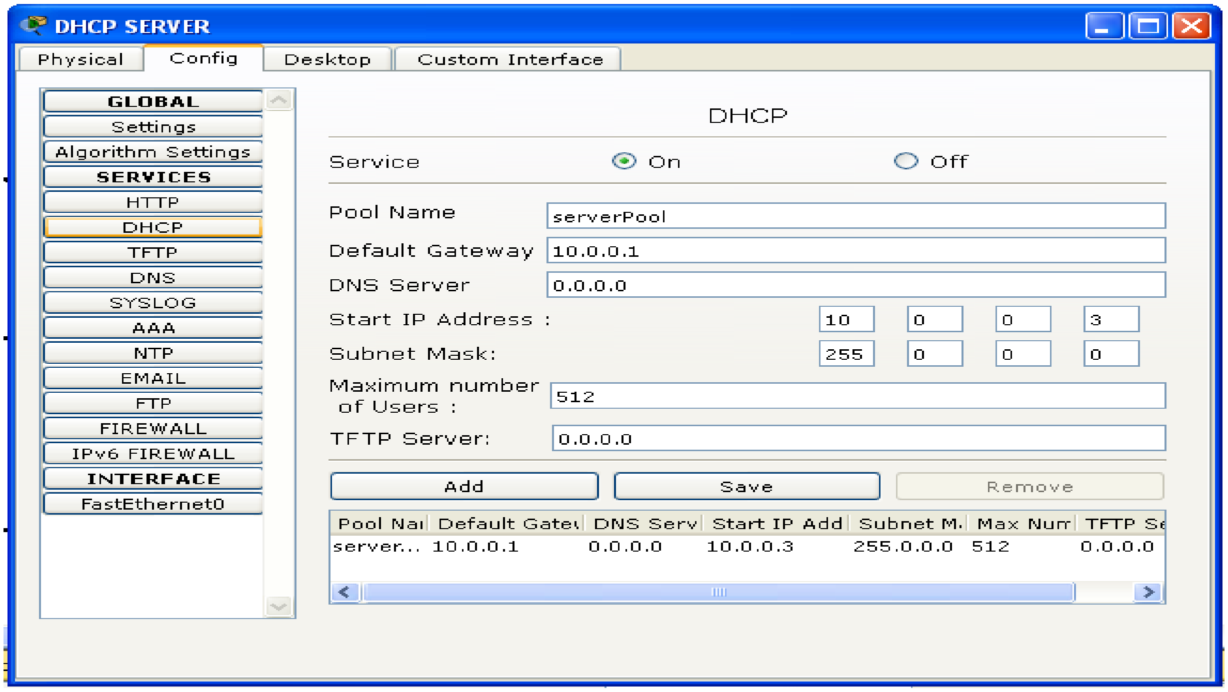
We can study the working of DHCP using the cisco packet tracer using the following example.



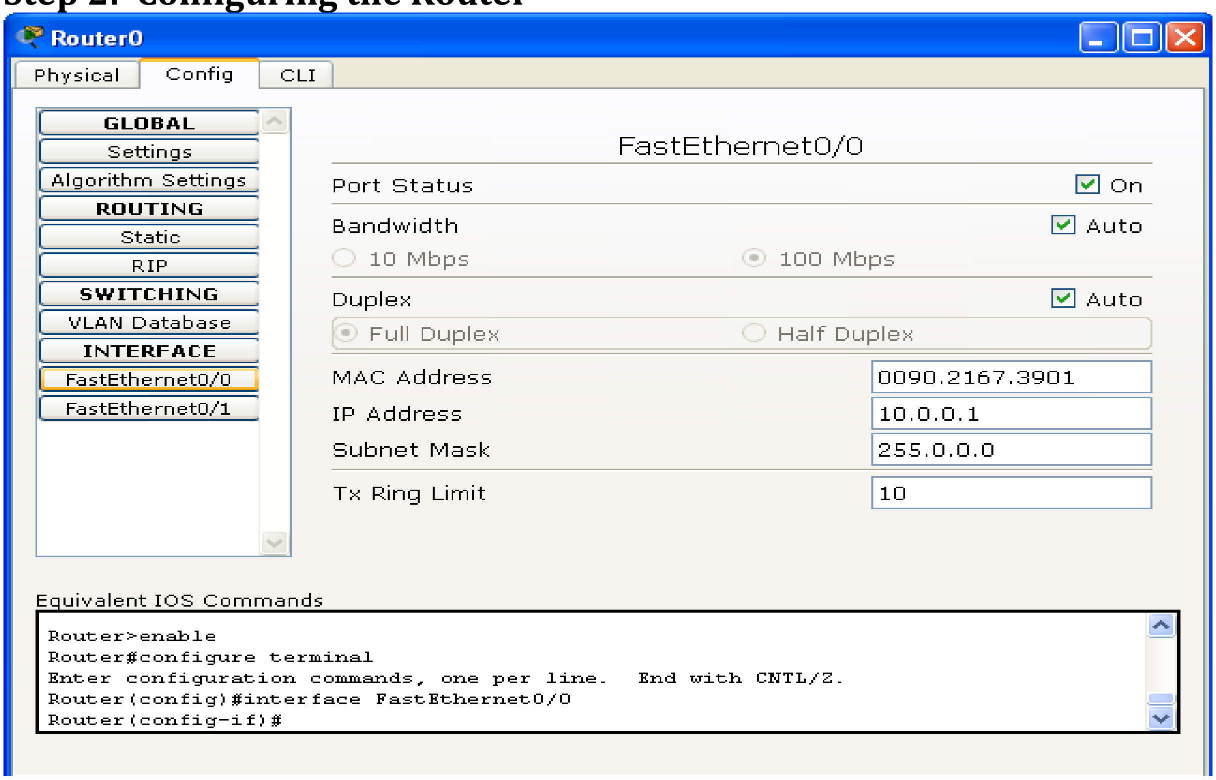
We configure the various components through the following steps

Step 1: Configuring the DHCP server



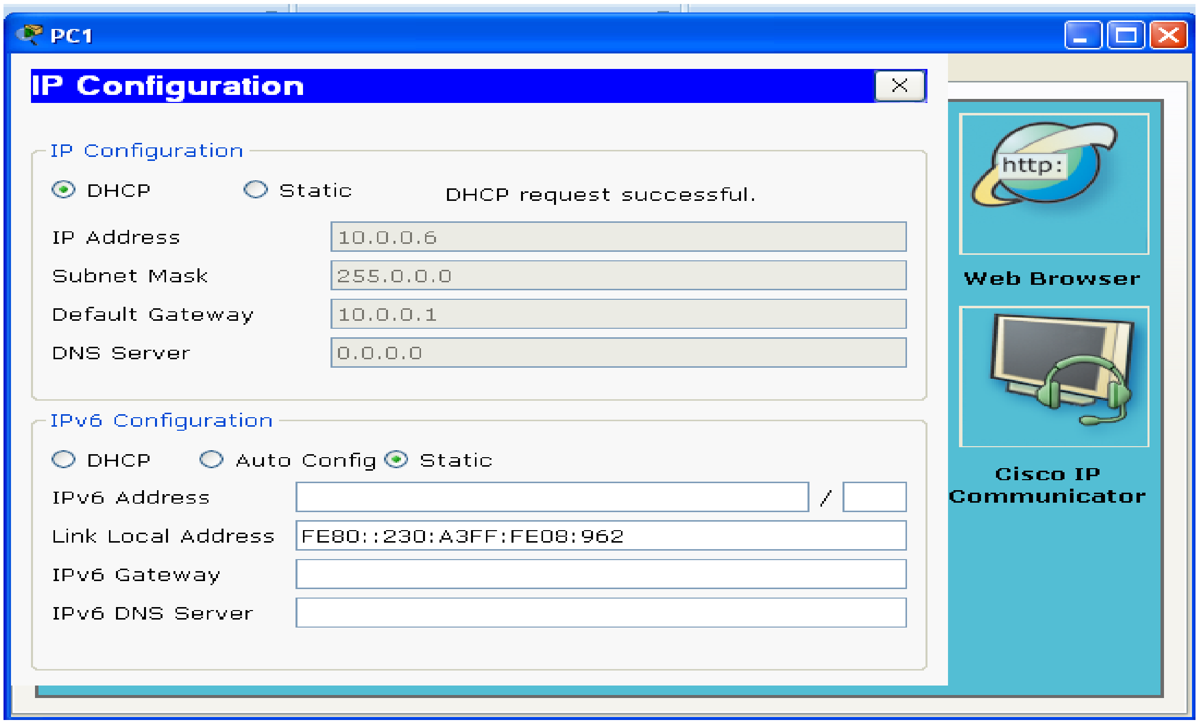


Step 2: Configuring the Router



Now we test the working of the DHCP server by sending a DHCP request from any of the PC as shown

Step 3: Sending DHCP request



Hence we have configured a DHCP server and also verified its operation

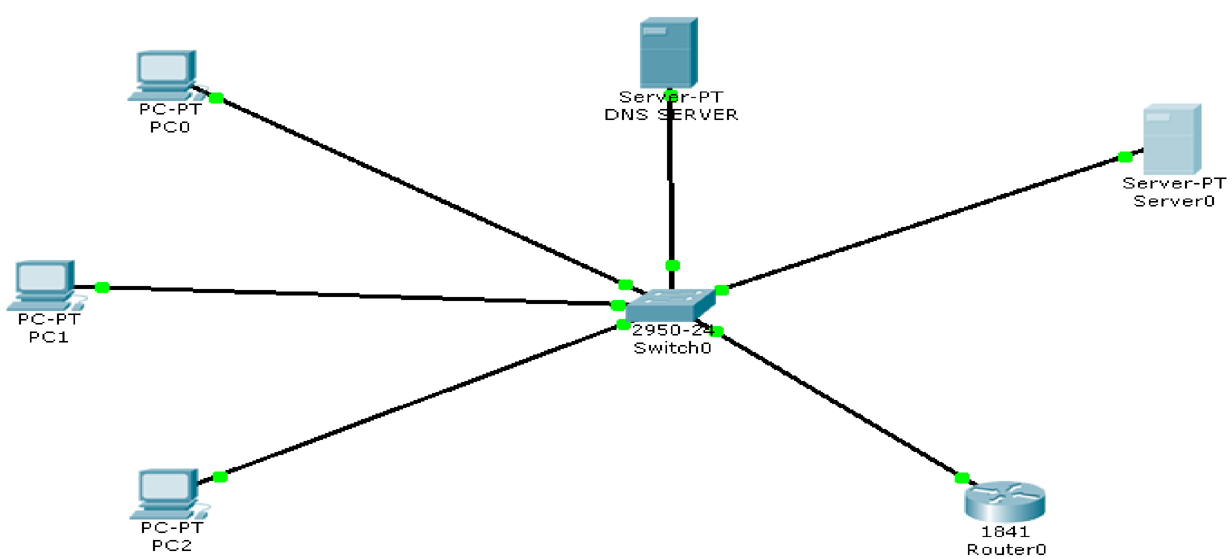
DNS

To identify an entity, TCP/IP protocols use the IP address, which uniquely identifies the connection of a host to the Internet. However, people prefer to use names instead of numeric addresses. Therefore, we need a system that can map a name to an address or an address to a name. One of the solution, is to divide this huge amount of information into smaller parts and store each part on a different computer. In this method, the hostthat needs mapping can contact the closest computer holding the needed information. This method is used by the **Domain Name System (DNS)**

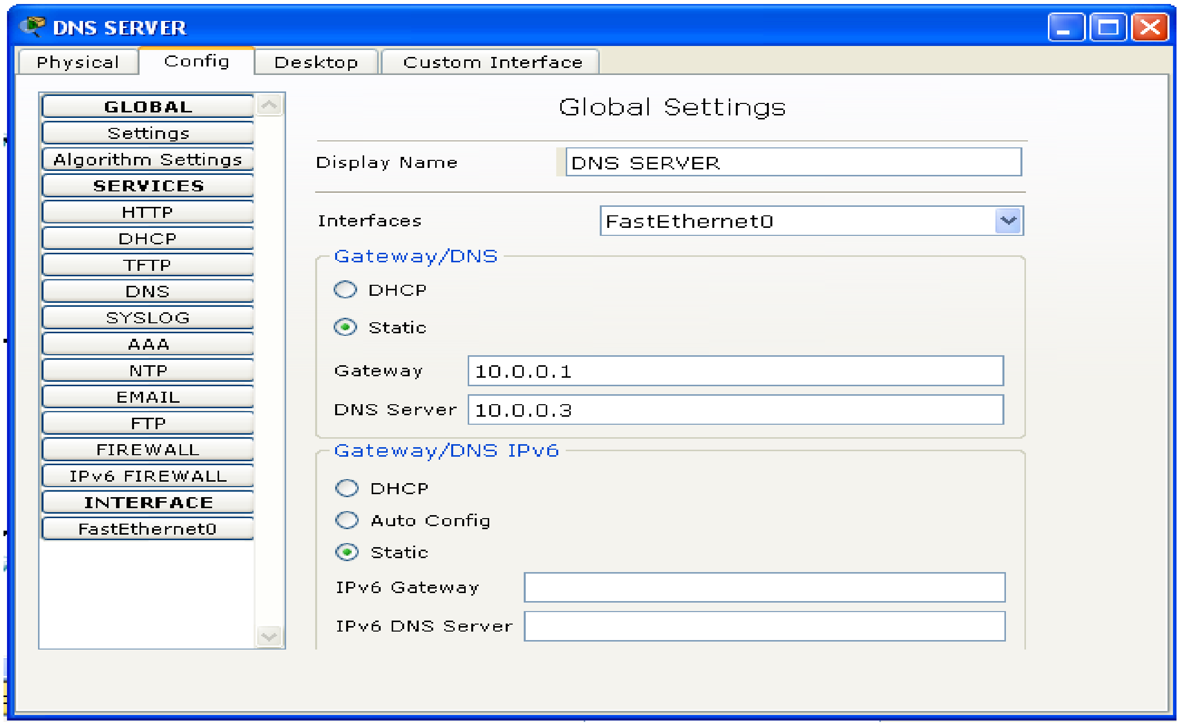
The DNS consists of the following steps

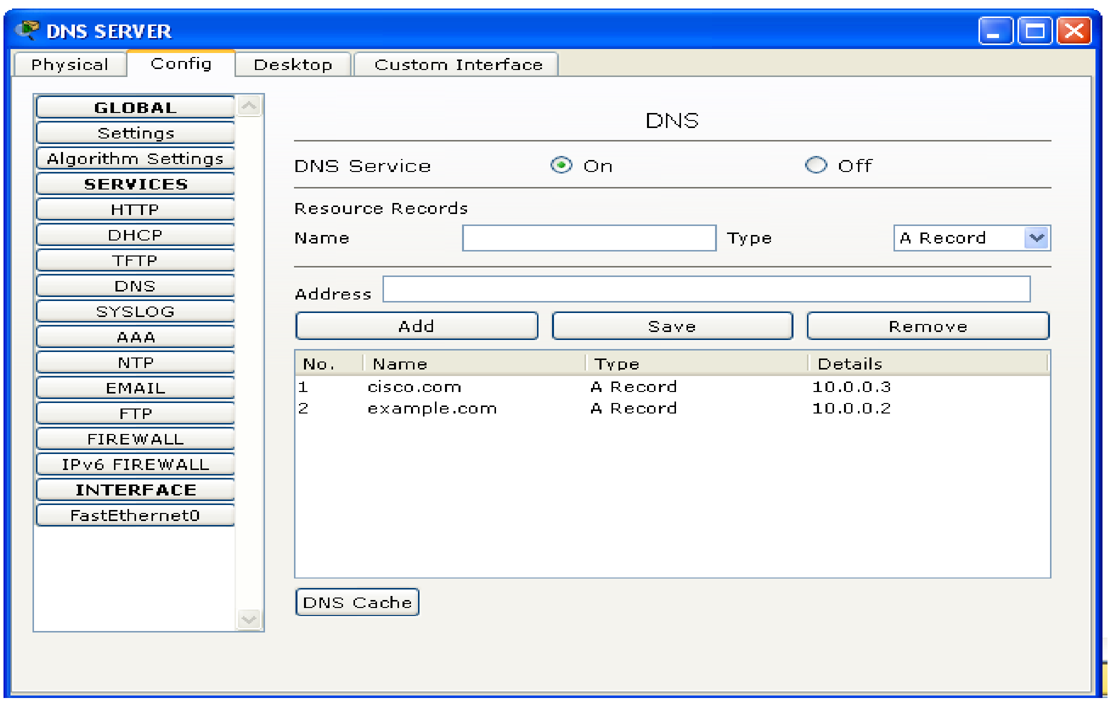
1. The user passes the host name to the file transfer client.
2. The file transfer client passes the host name to the DNS client.
3. The DNS client sends a message to a DNS server with a query that gives the file transfer server name using the known IP address of the DNS server.
4. The DNS server responds with the IP address of the desired file transfer server.
5. The DNS client passes the IP address to the file transfer server.
6. The file transfer client now uses the received IP address to access the file transfer server.

We can study the working of DNS using the cisco packet tracer using the following example.

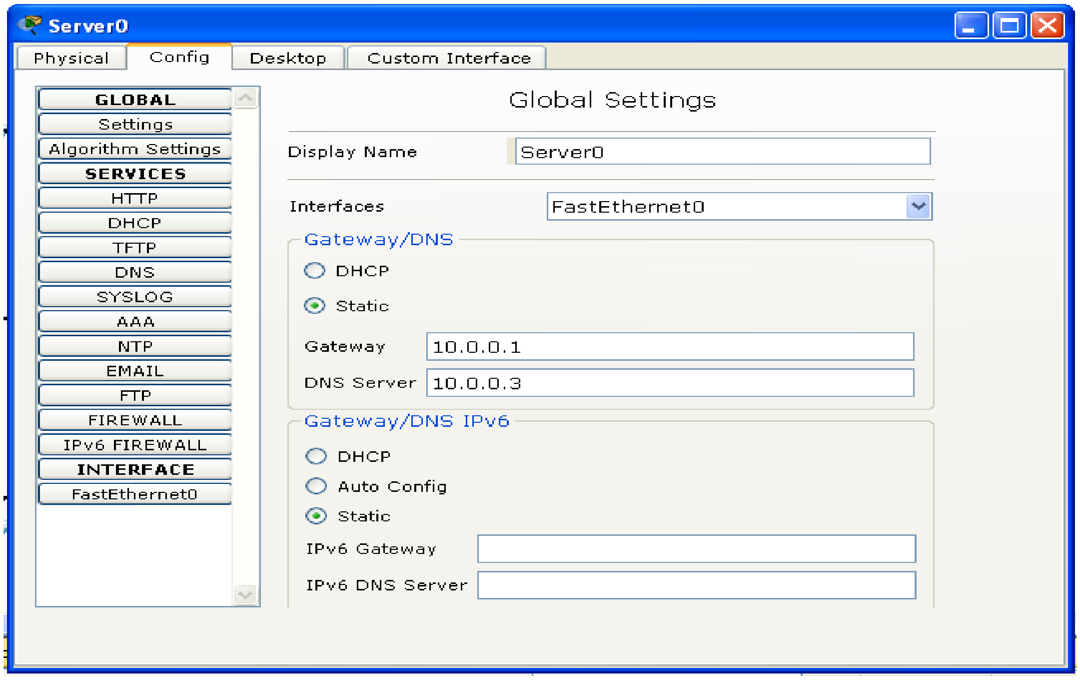


Step 1: Configuring the DNS server

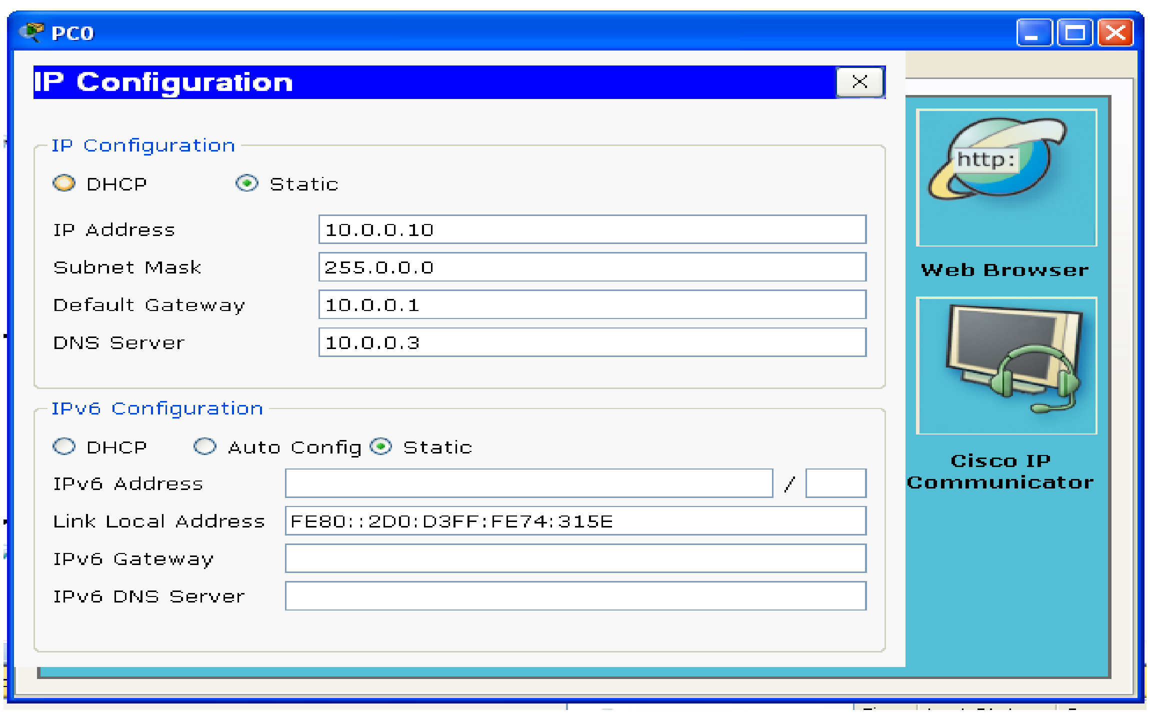




Step 2: Configuring the server 0

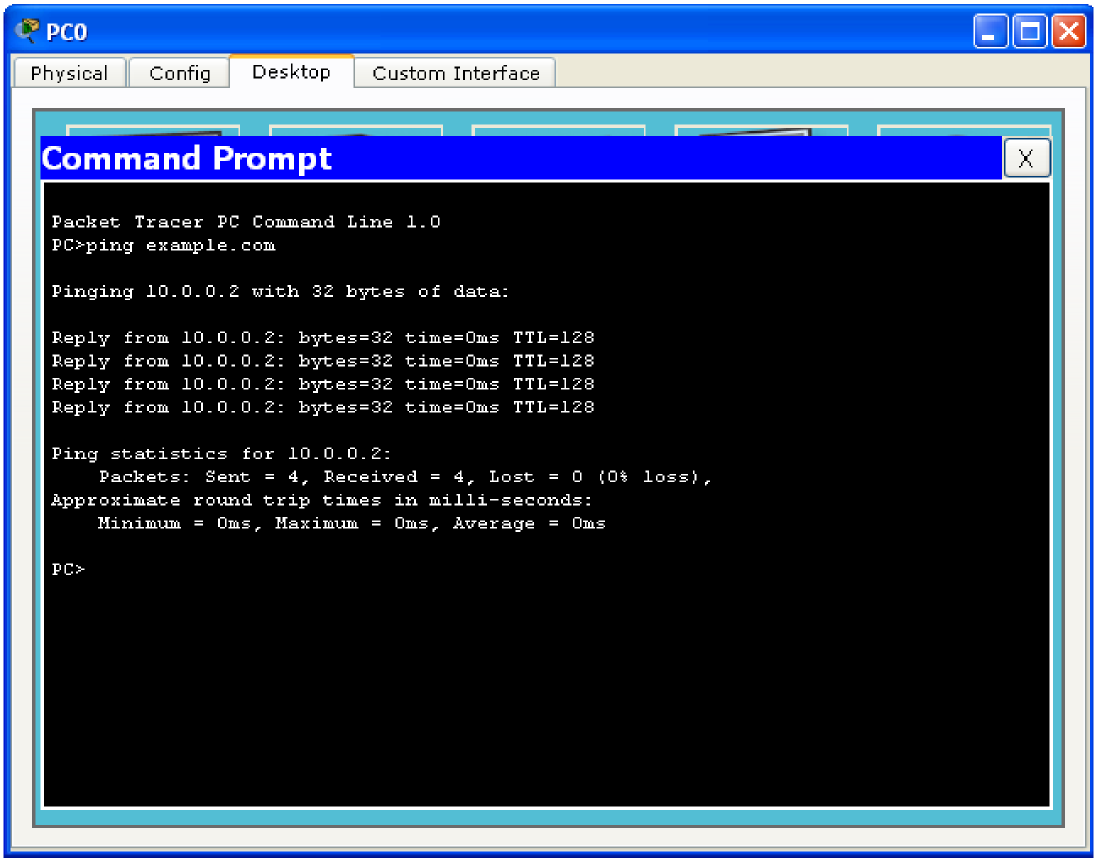


Step 3: Configuring PC0



Similarly we can assign any IP address to the PC1 and PC2

Now we can verify the working as follows



Hence the working of DNS has been studied

PRACTICAL NO 6

Configure SMTP, POP3, IMAP and MIME

SMTP (SIMPLE MAIL TRANSFER PROTOCOL)

The actual mail transfer is done through message transfer agents (MTAs). To send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA. The formal protocol that defines the MTA client and server in the Internet is called **Simple Mail Transfer Protocol (SMTP).**
SMTP is used two times, between the sender and the sender's mail server and between the two mail servers. Another protocol is needed between the mail server and the receiver.
SMTP simply defines how commands and responses must be sent back and forth.

POP3 (POST OFFICE PROTOCOL)

Post Office Protocol, version 3 (POP3) is simple and limited in functionality. The client POP3 software is installed on the recipient computer; the server POP3 software is installed on the mail server. Mail access starts with the client when the user needs to download its e-mail from the mailbox on the mail server. The client opens a connection to the server on TCP port 110. It then sends its user name and password to access the mailbox. The user can then list and retrieve the mail messages, one by one

IMAP4 (Internet Mail Access Protocol)

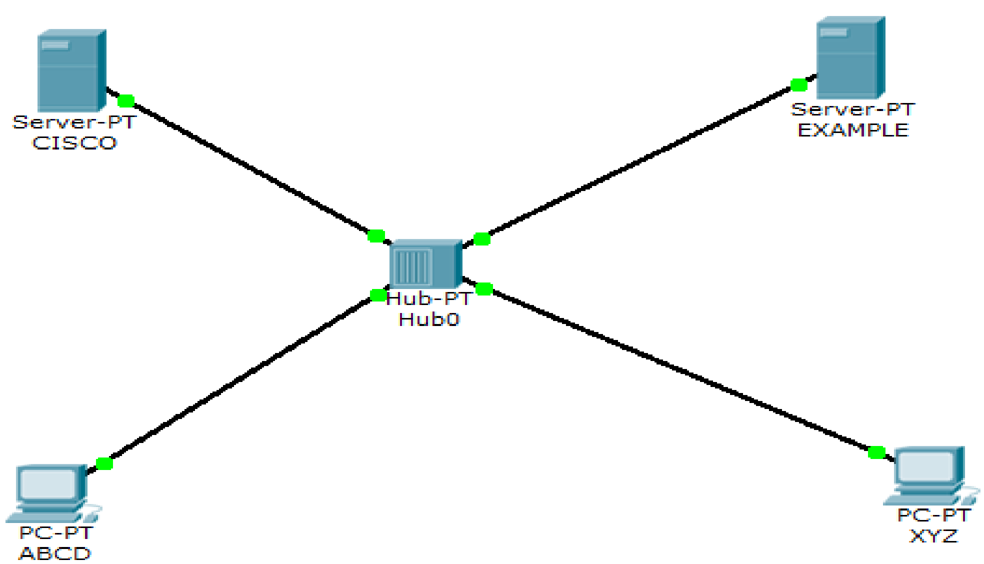
Another mail access protocol is **IMAP4.** IMAP4 is similar to POP3, but it has more features; IMAP4 is more powerful and more complex. POP3 is deficient in several ways. It does not allow the user to organize her mail on the server; the user cannot have different folders on the server. In addition, POP3 does not allow the user to partially check the contents of the mail before downloading. IMAP4 provides some extra functions as compared to POP3 which are as follows

1) A user can search the contents of the e-mail for a specific string of characters prior to downloading.
2) A user can partially download e-mail. This is especially useful if bandwidth is limited and the e-mail contains multimedia with high bandwidth requirements.
3) A user can create, delete, or rename mailboxes on the mail server.

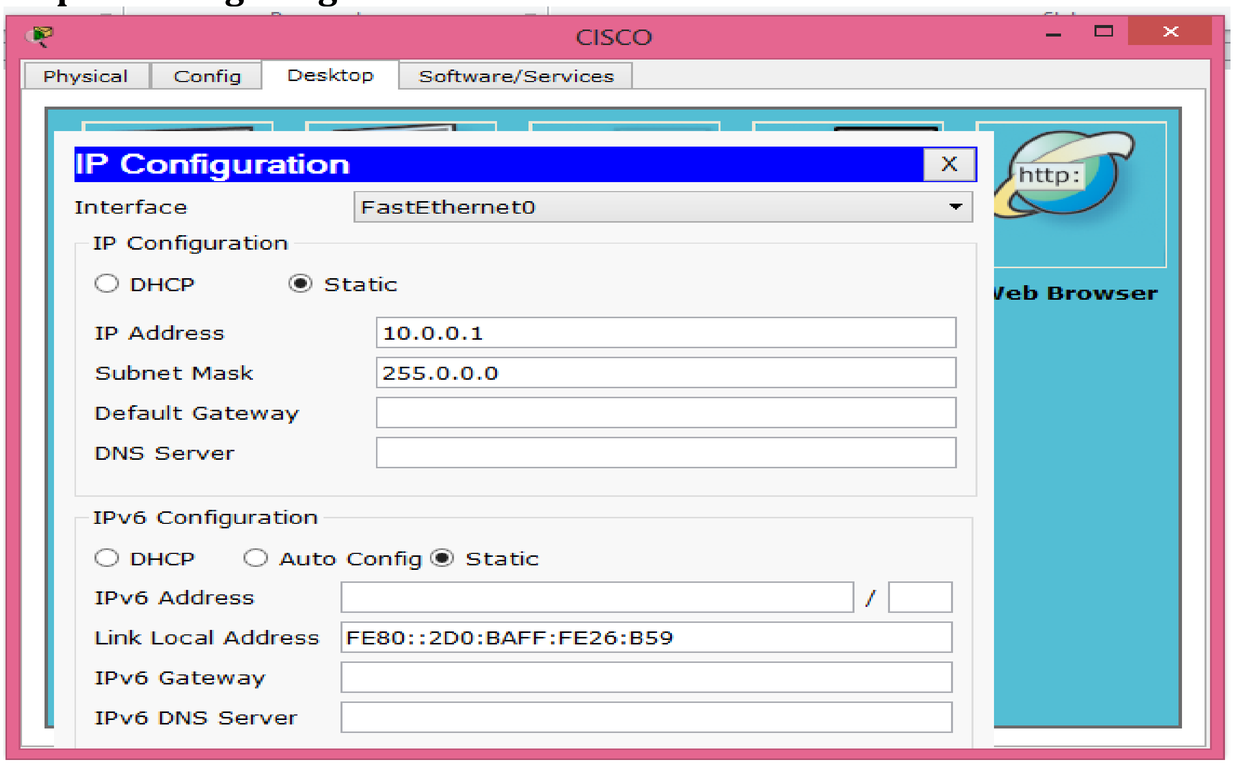
4) A user can create a hierarchy of mailboxes in a folder for e-mail storage

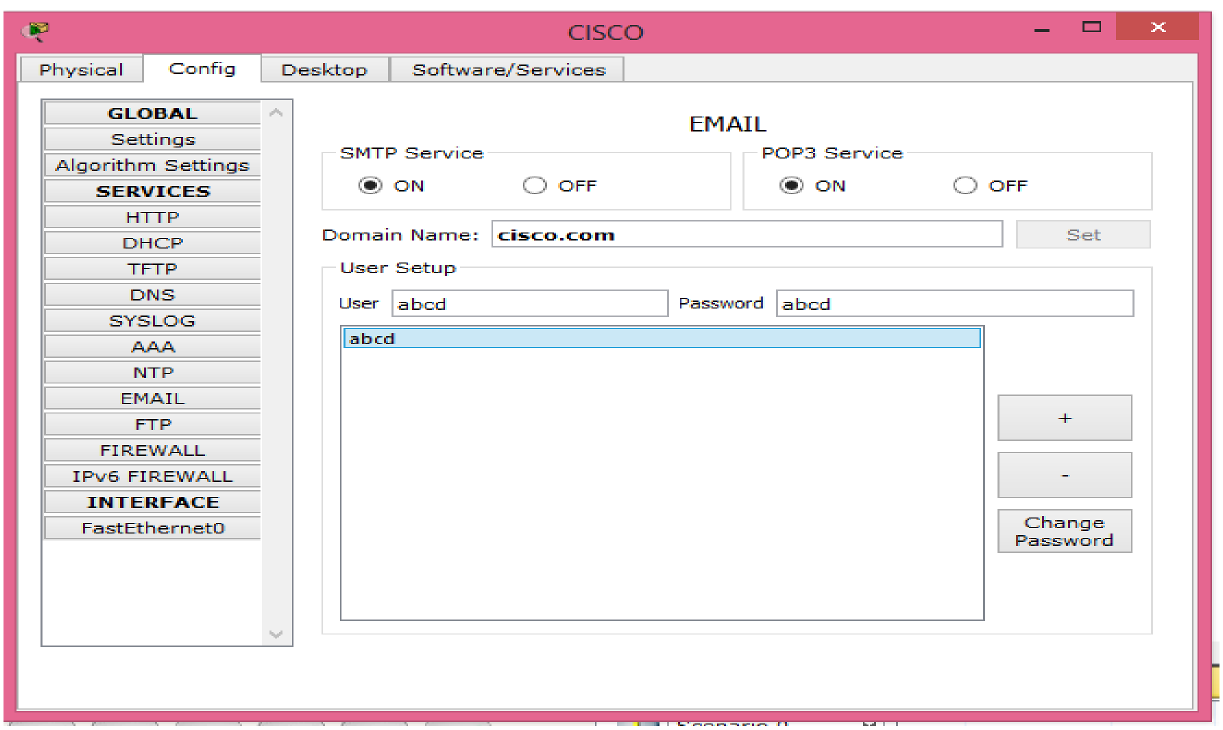
MIME (Multipurpose Internet Mail Extensions)
Electronic mail has a simple structure. Its simplicity, however, comes with a price. It can send messages only in NVT 7-bit ASCII format. In other words, it has some limitations. It cannot be used for languages other than English (such as French, German, Hebrew, Russian, Chinese, and Japanese). Also, it cannot be used to send binary files or video or audio data.

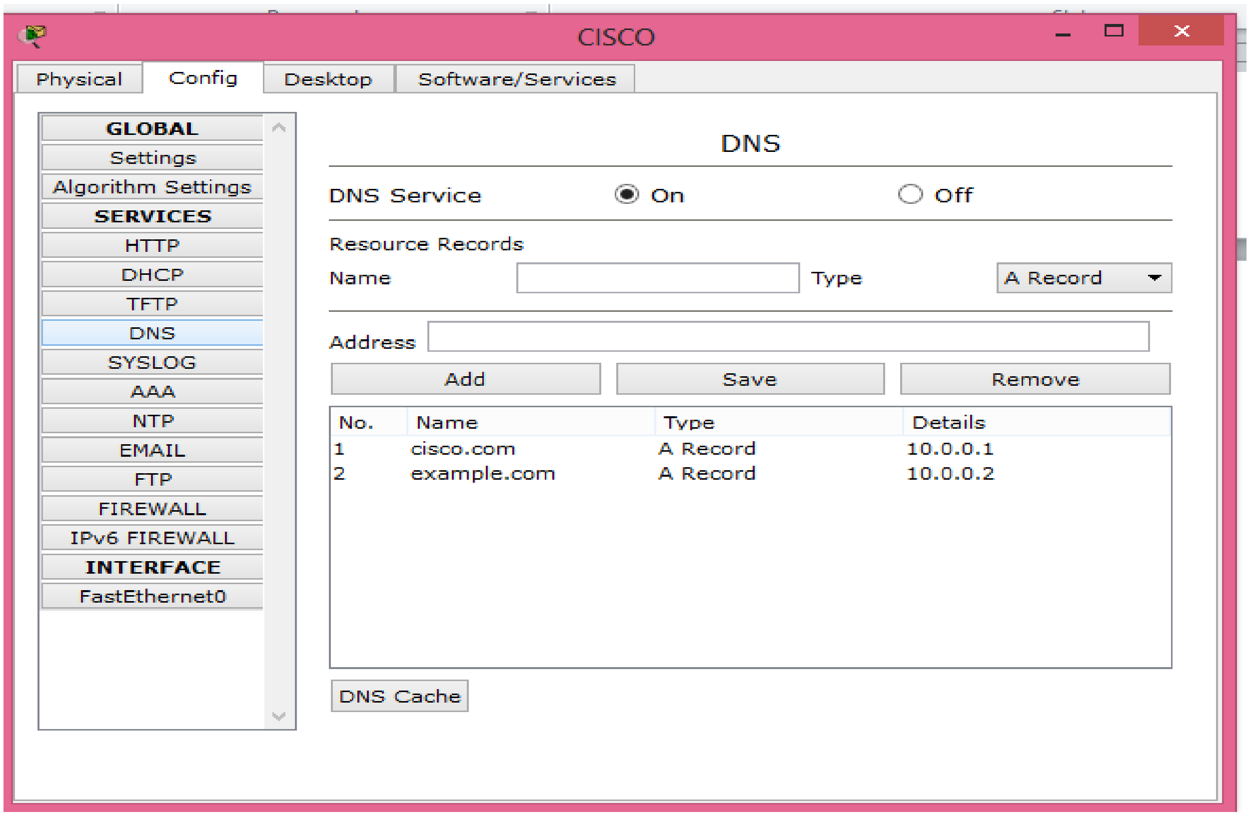
Multipurpose Internet Mail Extensions (MIME) is a supplementary protocol that allows non-ASCII data to be sent through e-mail. MIME transforms non-ASCII data at the sender site to NVT ASCII data and delivers it to the client MTA to be sent through the Internet. The message at the receiving site is transformed back to the original data. We can think of MIME as a set of software functions that transforms non-ASCII data to ASCII data and vice versa.
We study the above protocols using packet tracer using the following network



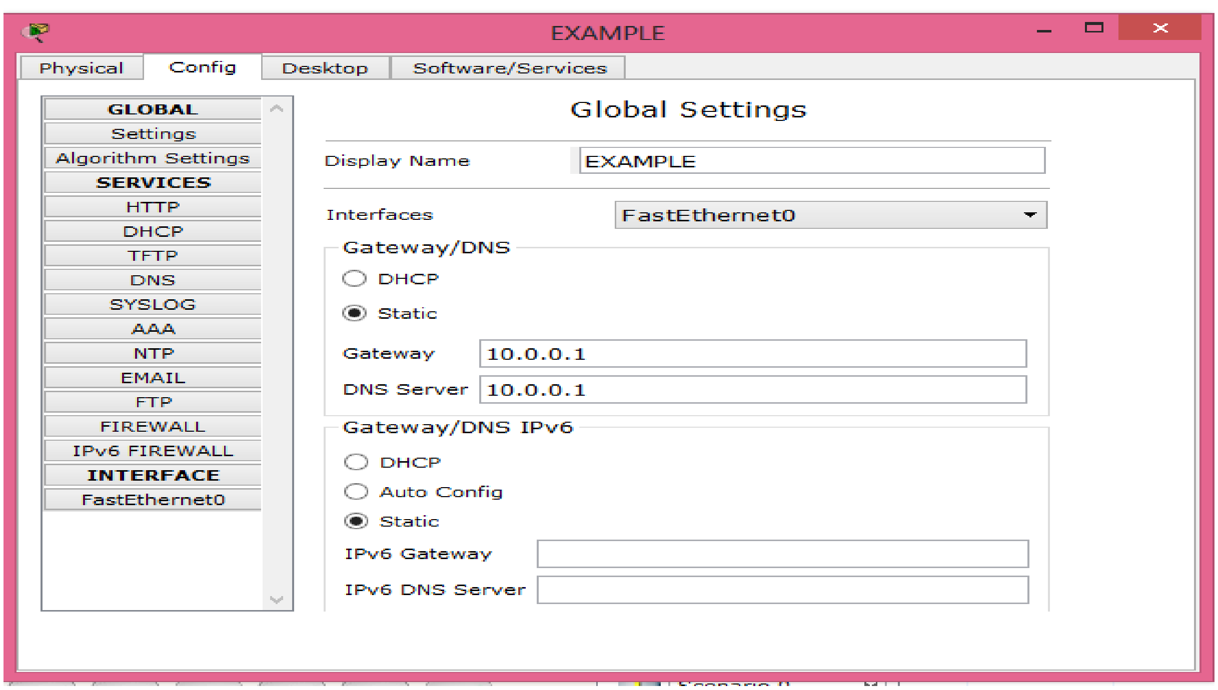
Step 1: Configuring the CISCO server

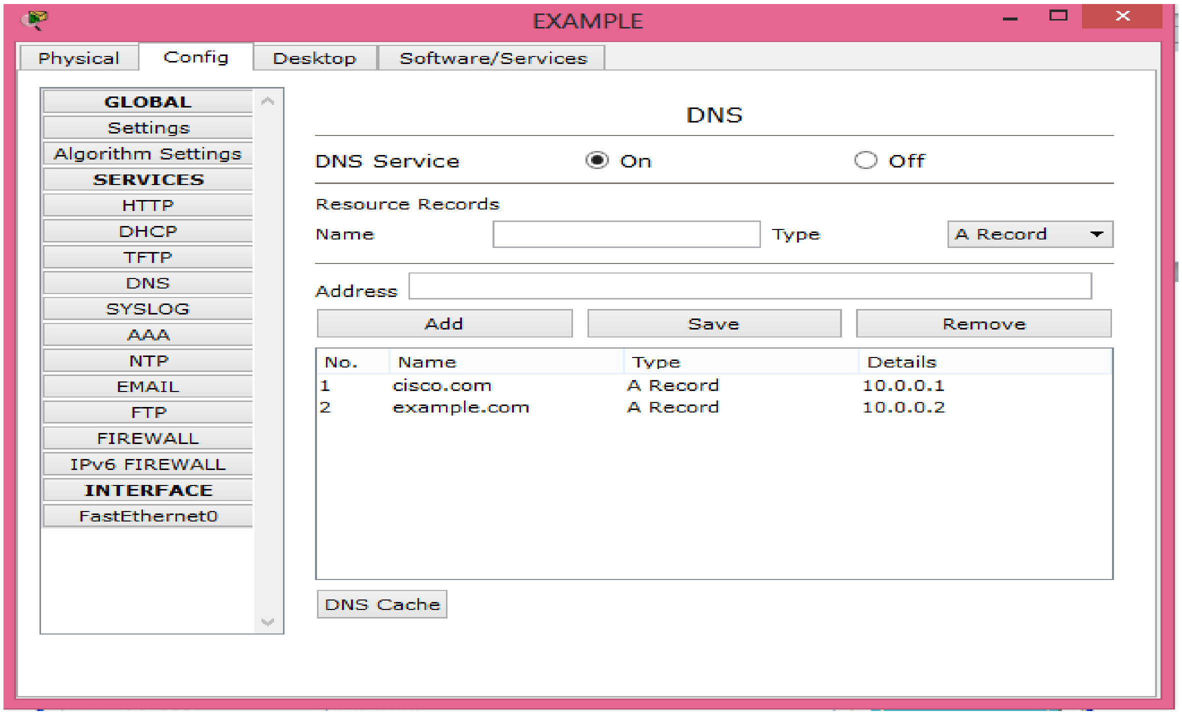


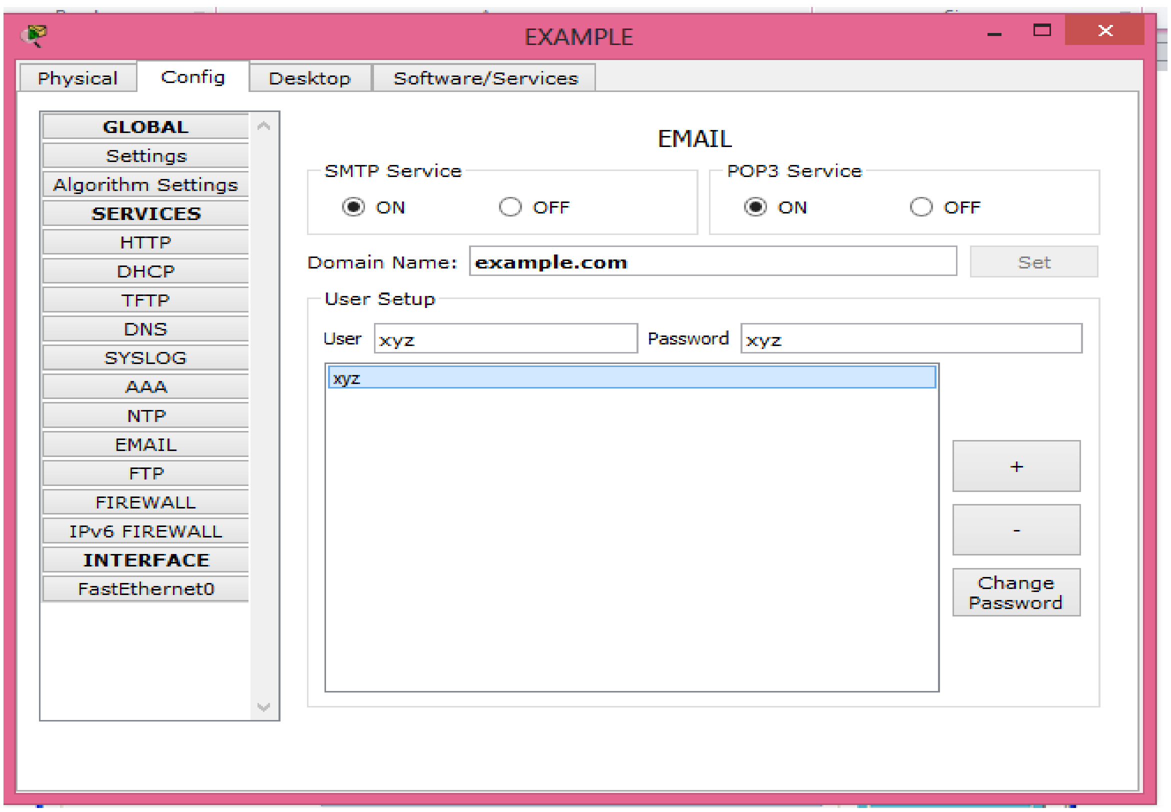




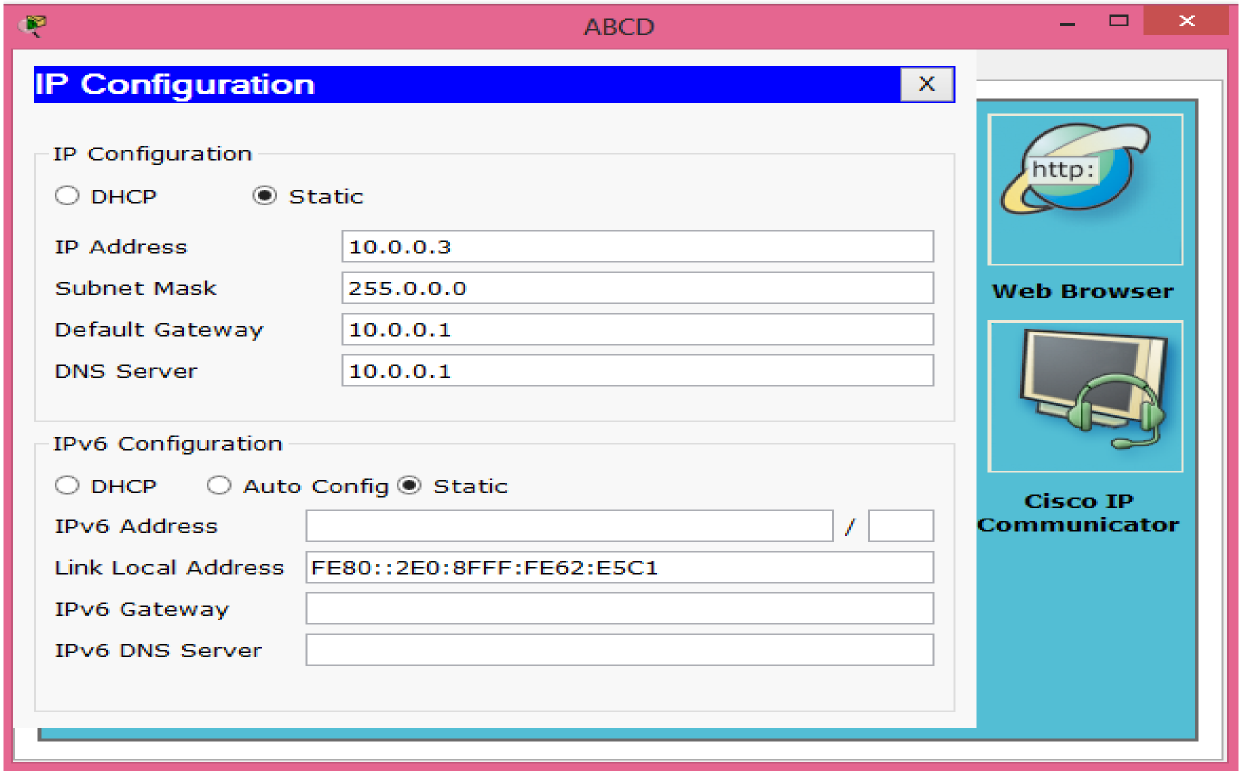
Step 2: Configuring the EXAMPLE server

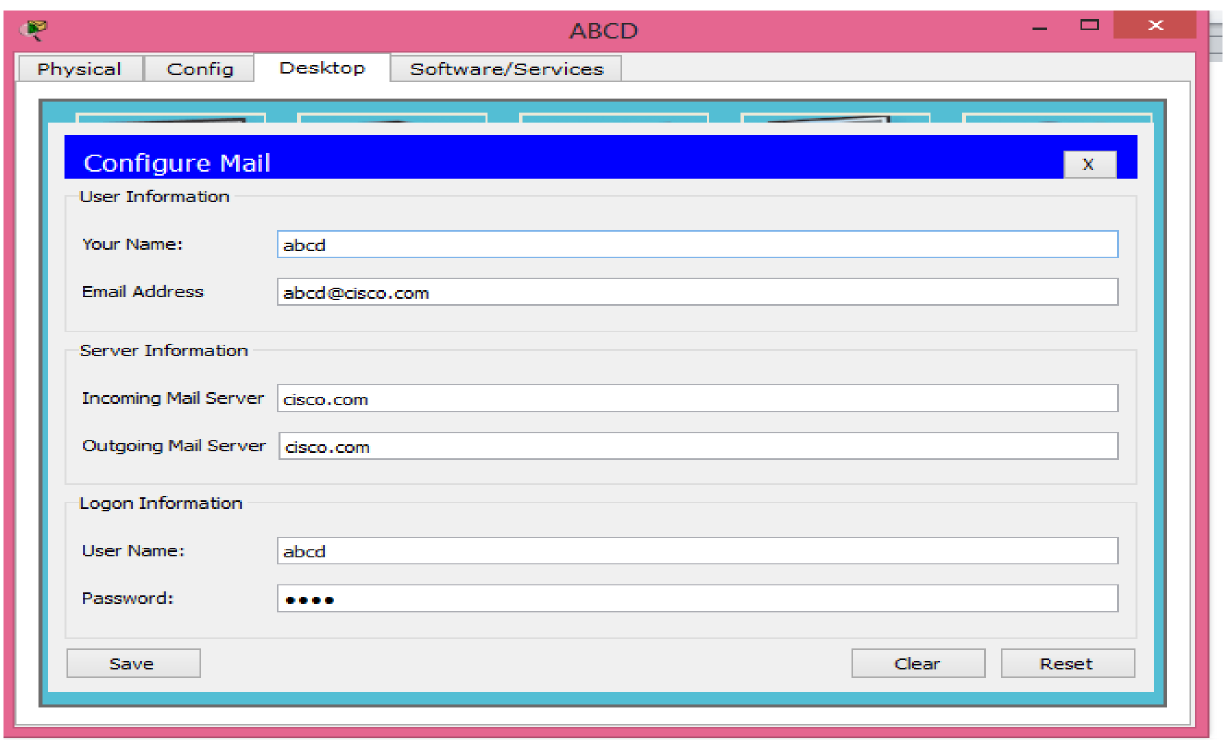



EXAMPLE
Physical
Config
Desktop
Software/Services
GLOBAL
Settings
Algorithm Settings
SERVICES
HTTP
DHCP
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
FIREWALL
IPv6 FIREWALL
INTERFACE
FastEthernet0
DNS
DNS Service
On
Off
Resource Records
Name
Type
A Record
Address
Add
Save
Remove
No. Name Type Details
1 cisco.com A Record 10.0.0.1
2 example.com A Record 10.0.0.2
DNS Cache

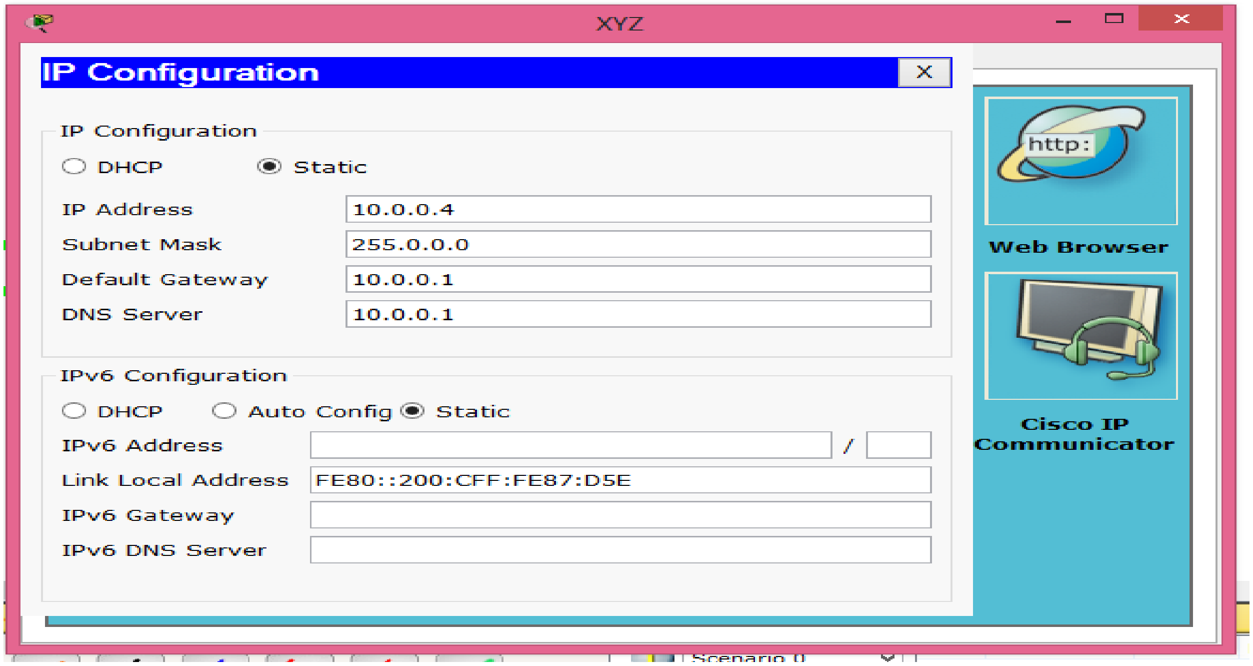

EXAMPLE
Physical
Config
Desktop
Software/Services
GLOBAL
Settings
Algorithm Settings
SERVICES
HTTP
DHCP
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
FIREWALL
IPv6 FIREWALL
INTERFACE
FastEthernet0
EMAIL
SMTP Service
ON
OFF
POP3 Service
ON
OFF
Domain Name: example.com
Set
User Setup
User xyz
Password xyz
xyz
+
-
Change Password

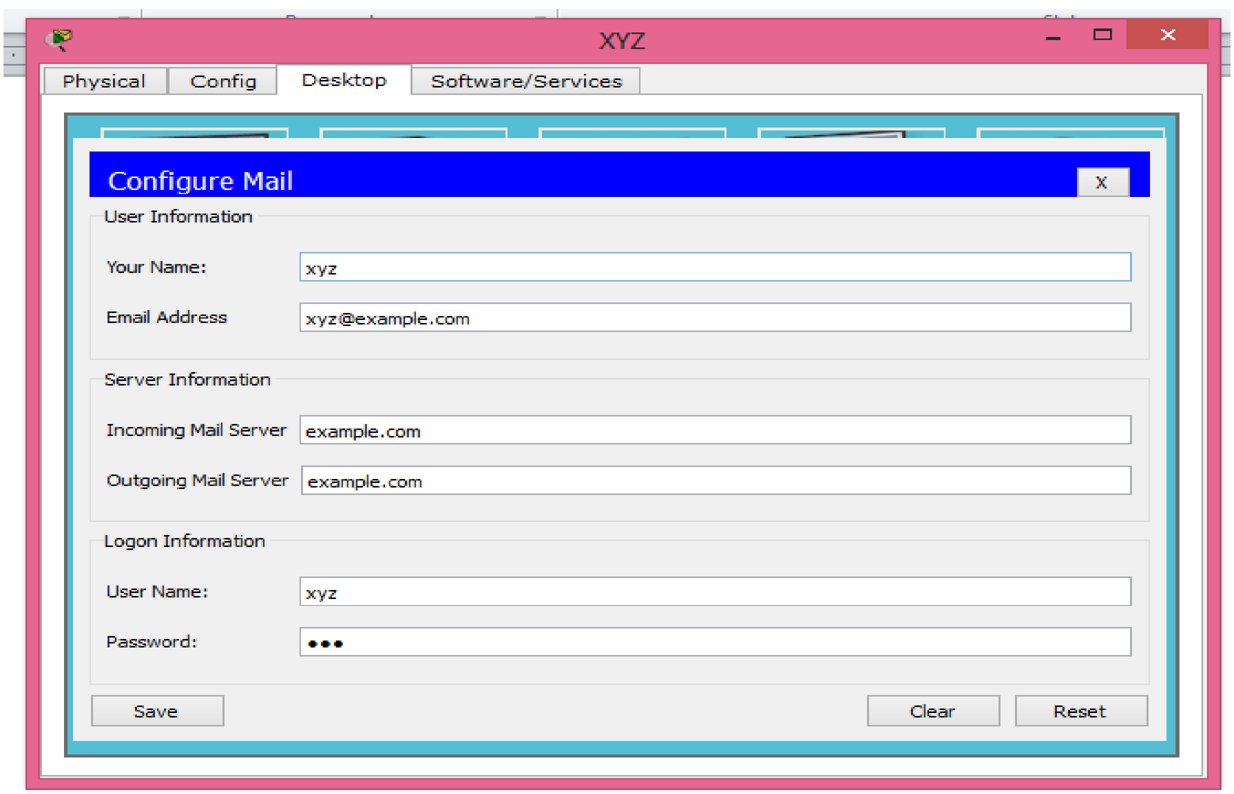
Step 3: Configuring the ABC



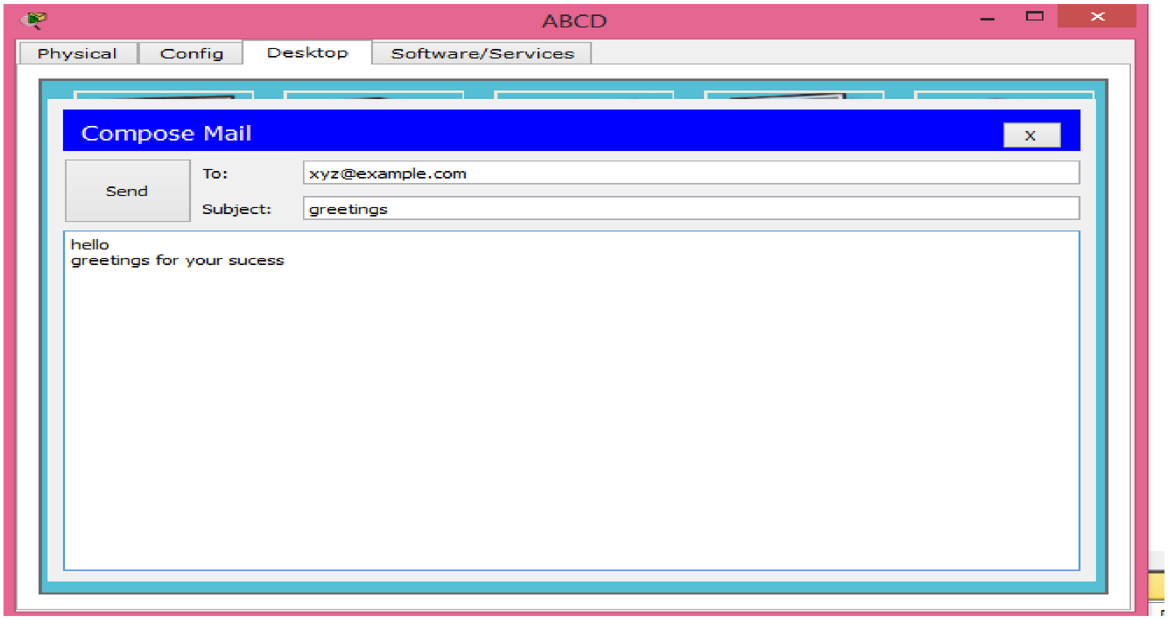


Step 4: Configuring the XYZ

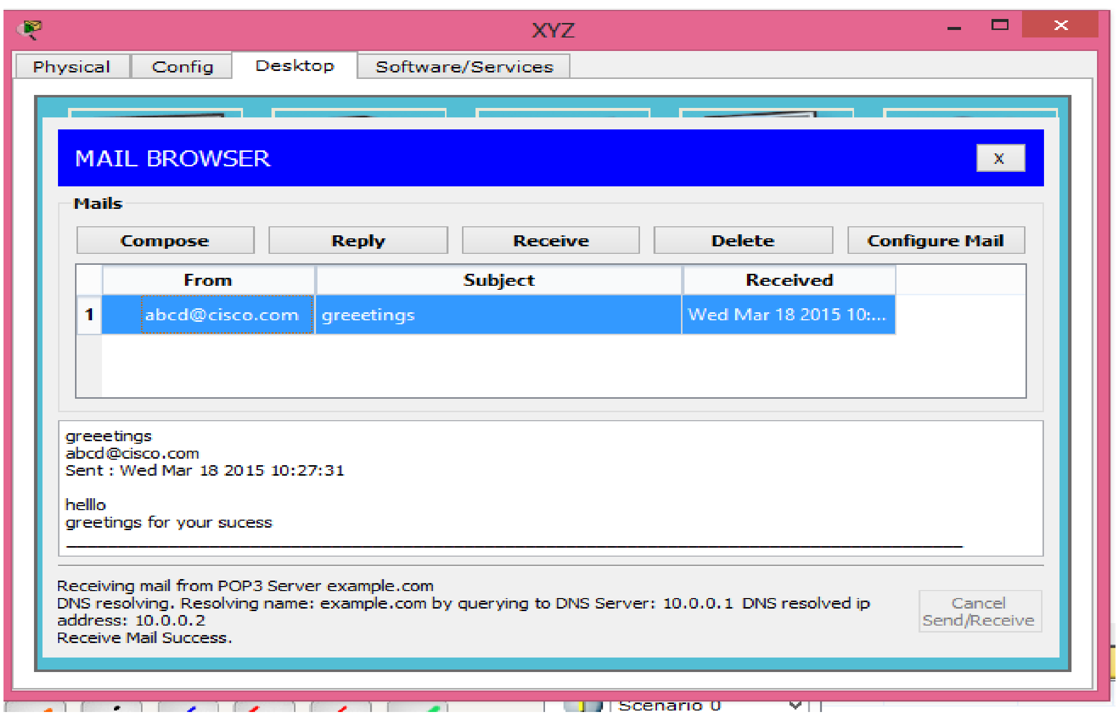




Now we check the working of the mail servers by sending mail as follows



Now we click the send button and check the receive mail at the xyz



Hence we have successfully created a mail server and check the working of all the protocols concerned with email

PRACTICAL NO 7

Configuring UDP and TCP

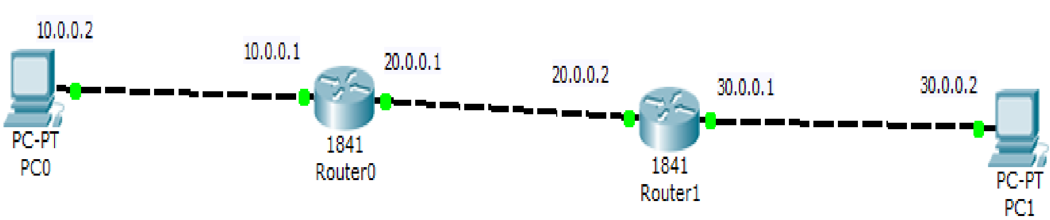
USER DATAGRAM PROTOCOL (UDP)

UDP is a transport protocol that creates a process-to-process communication. UDP is a (mostly) unreliable and connectionless protocol that requires little overhead and offers fast delivery. The UDP packet is called a user datagram.
UDP's only attempt at error control is the checksum. Inclusion of a pseudoheader in the checksum calculation allows source and destination IP address errors to be detected. UDP has no flow-control mechanism.
A user datagram is encapsulated in the data field of an IP datagram. Incoming and outgoing queues hold messages going to and from UDP.
UDP uses multiplexing to handle outgoing user datagrams from multiple processes on one host. UDP uses demultiplexing to handle incoming user datagrams that go to different processes on the same host.
A UDP package can involve five components: a control-block table, a control block module, input queues, an input module, and an output module. The input queues hold incoming user datagrams. The control-block module is responsible for maintenance of entries in the control-block table. The input module creates input queues; the output module sends out user datagrams.

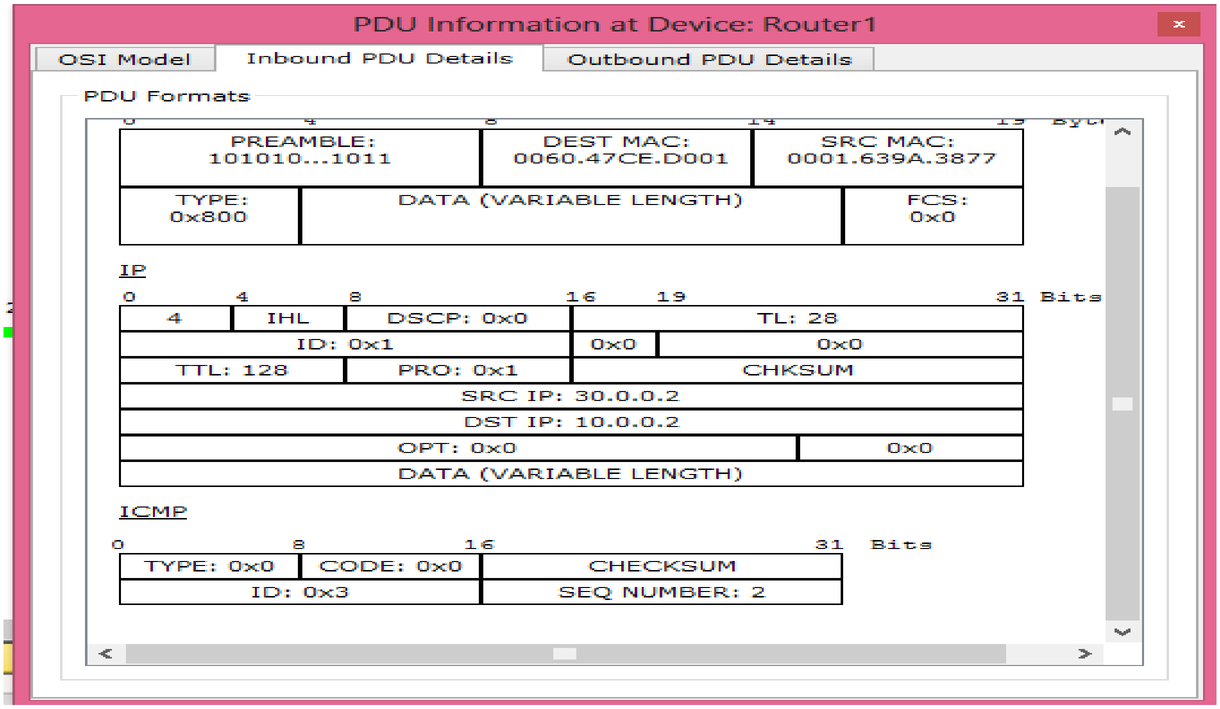
TRANSMISSION CONTROL PROTOCOL (TCP)

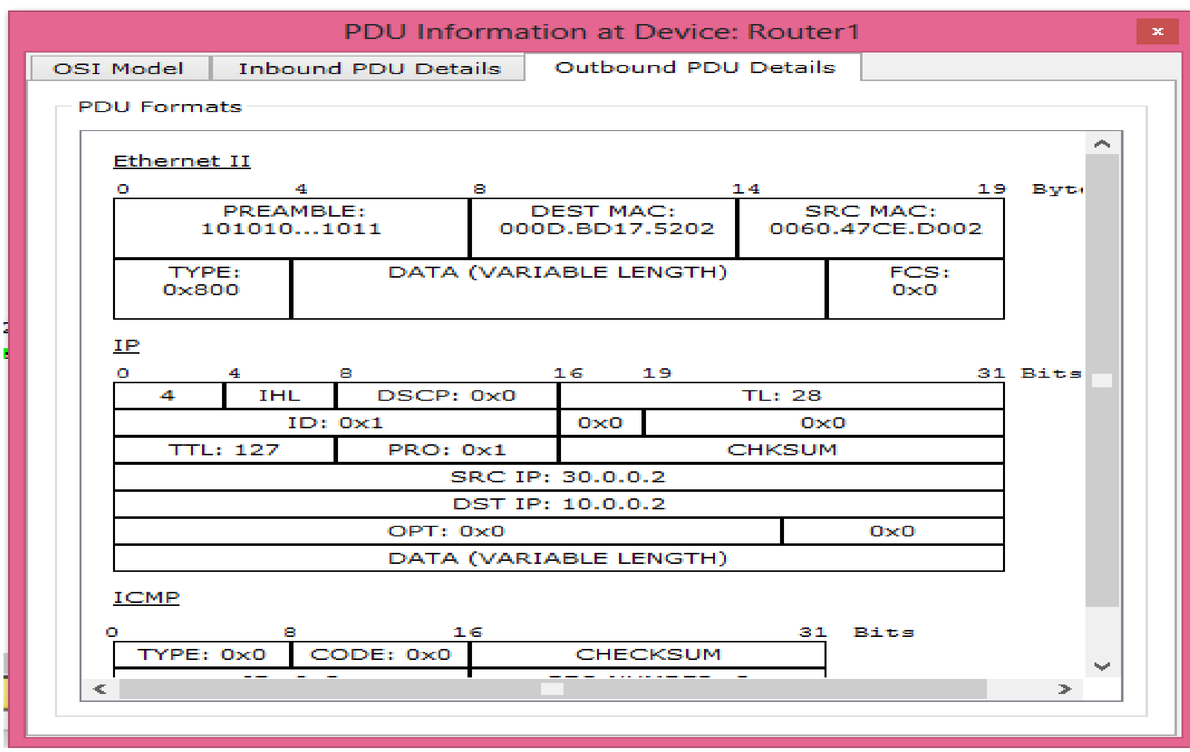
Transmission Control Protocol (TCP) is one of the transport layer protocols in the TCP/IP protocol suite. TCP provides process-to-process, full-duplex, and connection- oriented service. The unit of data transfer between two devices using TCP software is called a segment; it has 20 to 60 bytes of header, followed by data from the application program.
A TCP connection consists of three phases: connection establishment, data transfer, and connection termination. Connection establishment requires three-way handshaking; connection termination requires three- or four-way handshaking.
TCP software is normally implemented as a finite state machine (FSM).
TCP uses flow control, implemented as a sliding window mechanism, to avoid Over whelming a receiver with data. The TCP window size is determined by the receiver-advertised window size (rwnd) or the congestion window size

(cwnd), whichever is smaller. The window can be opened or closed by the receiver, but should not be shrunk. The bytes of data being transferred in each connection are numbered by TCP. The numbering starts with a randomly generated number.
TCP uses error control to provide a reliable service. Error control is handled by checksums, acknowledgment, and time-outs. Corrupted and lost segments are eventually retransmitted and duplicate segments are discarded. Data may arrive out of order and temporarily stored by the receiving TCP, but TCP guarantees that no out-of-order segment is delivered to the process. In modern implementations, a retransmission occurs if the retransmission timer expires or three duplicate ACK segments have arrived.
TCP uses congestion control to avoid and detect congestion in the network. The slow start (exponential increase), congestion avoidance (additive increase), and congestion detection (multiplicative decrease) strategies are used for congestion control. In the slow start algorithm the size of the congestion window increases exponentially until it reaches a threshold. In the congestion avoidance algorithm the size of the congestion window increases additively until congestion is detected.
Different TCP implementations react differently to congestion detection. If detection is by time-out, a new slow start phase starts. If detection is by three duplicate ACKs, a new congestion avoidance phase starts.
We study the above using the example below



In the above example we configure the routing table through RIP and send the messages as shown


PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
PREAMBLE: 101010...1011
DEST MAC: 0060.47CE.D001
SRC MAC: 0001.639A.3877
TYPE: 0x800
DATA (VARIABLE LENGTH)
FCS: 0x0
IP
0 4 8 16 19 31 Bits
4
IHL
DSCP: 0x0
TL: 28
ID: 0x1
0x0
0x0
TTL: 128
PRO: 0x1
CHKSUM
SRC IP: 30.0.0.2
DST IP: 10.0.0.2
OPT: 0x0
0x0
DATA (VARIABLE LENGTH)
ICMP
0 8 16 31 Bits
TYPE: 0x0
CODE: 0x0
CHECKSUM
ID: 0x3
SEQ NUMBER: 2

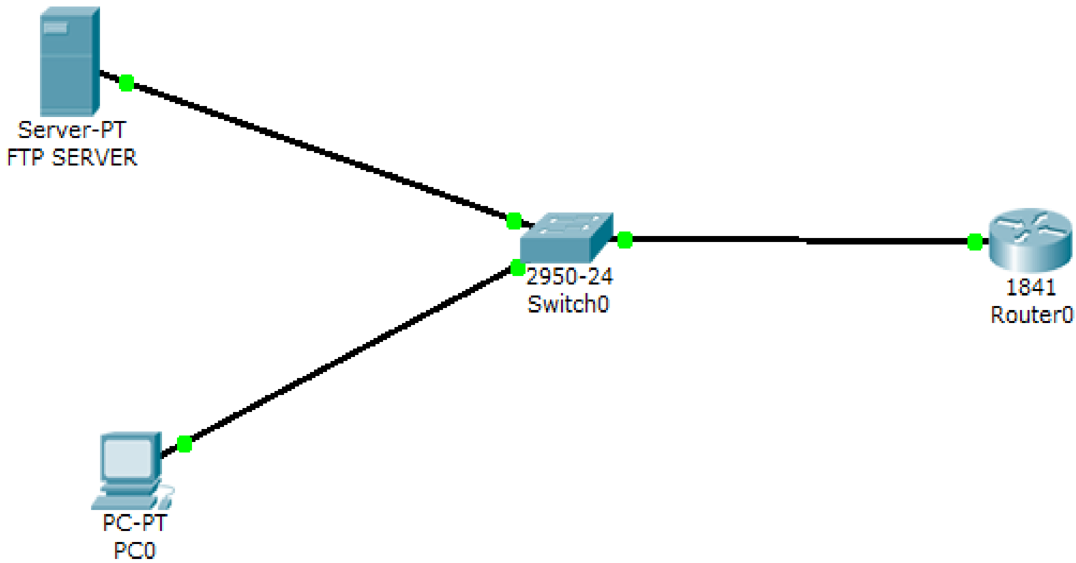

PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
0 4 8 14 19
PREAMBLE: 101010...1011
DEST MAC: 000D.BD17.5202
SRC MAC: 0060.47CE.D002
TYPE: 0x800
DATA (VARIABLE LENGTH)
FCS: 0x0
IP
0 4 8 16 19 31 Bits
4
IHL
DSCP: 0x0
TL: 28
ID: 0x1
0x0
0x0
TTL: 127
PRO: 0x1
CHKSUM
SRC IP: 30.0.0.2
DST IP: 10.0.0.2
OPT: 0x0
0x0
DATA (VARIABLE LENGTH)
ICMP
0 8 16 31 Bits
TYPE: 0x0
CODE: 0x0
CHECKSUM

PRACTICAL NO 7

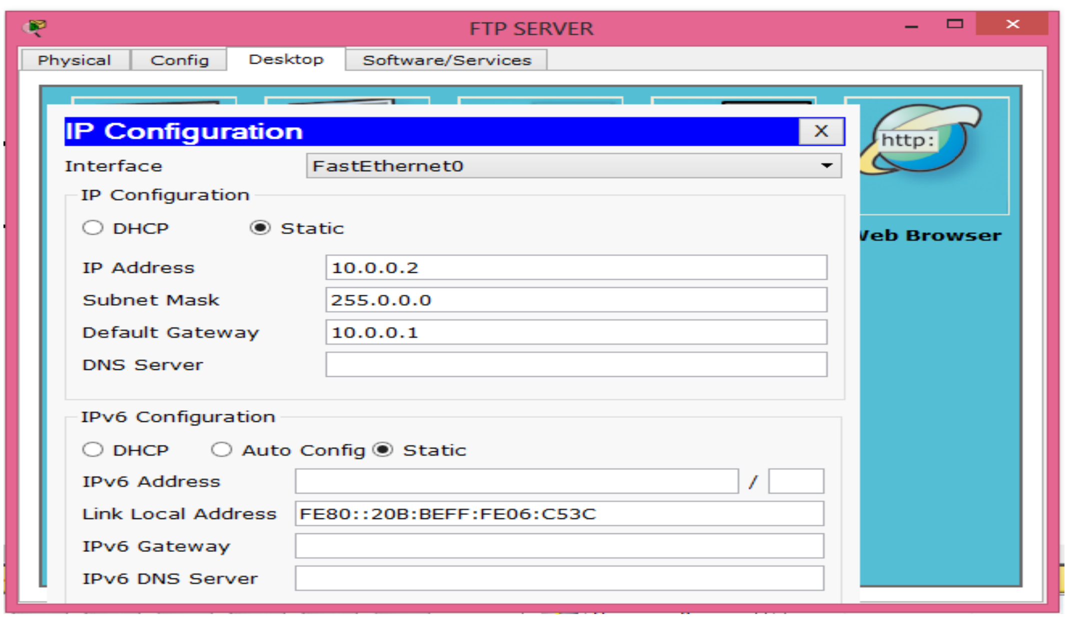
Configuring FTP and HTTP. Run Telnet and SSH

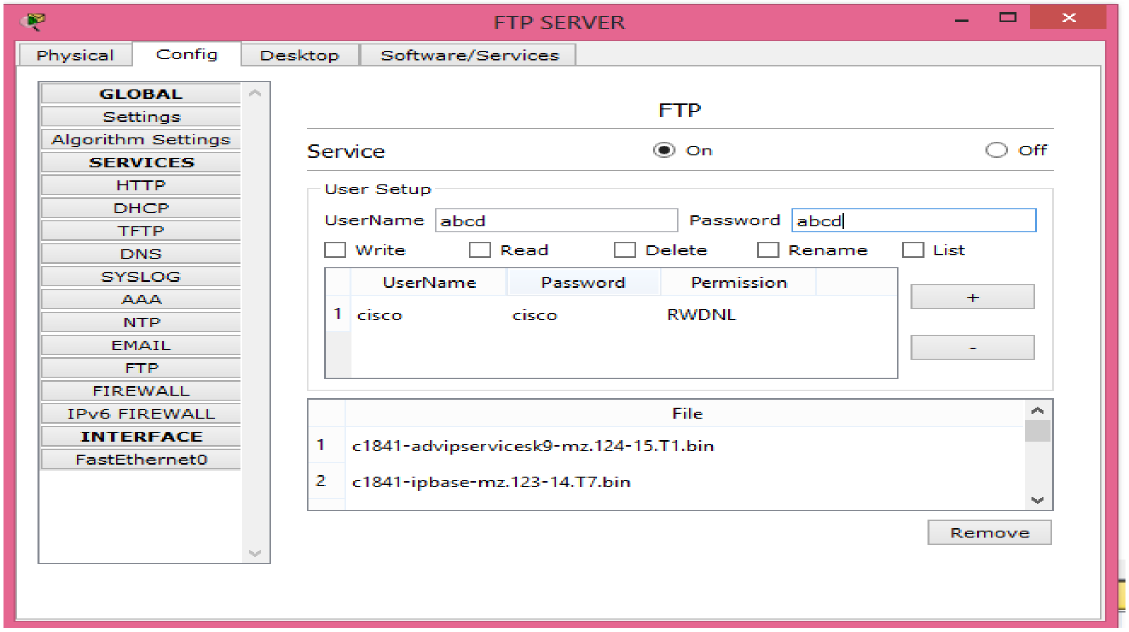
File Transfer Protocol (FTP)

File Transfer Protocol (FTP) is a TCP/IP client-server application for copying files from one host to another. FTP requires two connections for data transfer: a control connection and a data connection. FTP employs NVT ASCII for communication between dissimilar systems. Prior to the actual transfer of files, the file type, data structure, and transmission mode are defined by the client through the control connection.
FTP uses two well-known TCP ports: Port 21 is used for the control connection, and port 20 is used for the data connection
The **control connection** remains connected during the entire interactive FTP session. The data connection is opened and then closed for each file transferred. It opens each time commands that involve transferring files are used, and it closes when the file is transferred. In other words, when a user starts an FTP session, the control connection opens. While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred
There are six classes of commands sent by the client to establish communication with the server: access commands, file management commands, data formatting commands, port defining commands, file transferring commands, and miscellaneous commands. There are three types of file transfer: server-to-client file transfer, client-to-server file transfer, transfer of list of directories.
Transferring files with FTP is not secure. One solution to provide security is to add a Secure Socket Layer (SSL) between the FTP application layer and the TCP layer.
Another solution is to use a completely independent file transfer application called sftp that is one of the application in SSH protocol.
We can study the protocol using the following example.

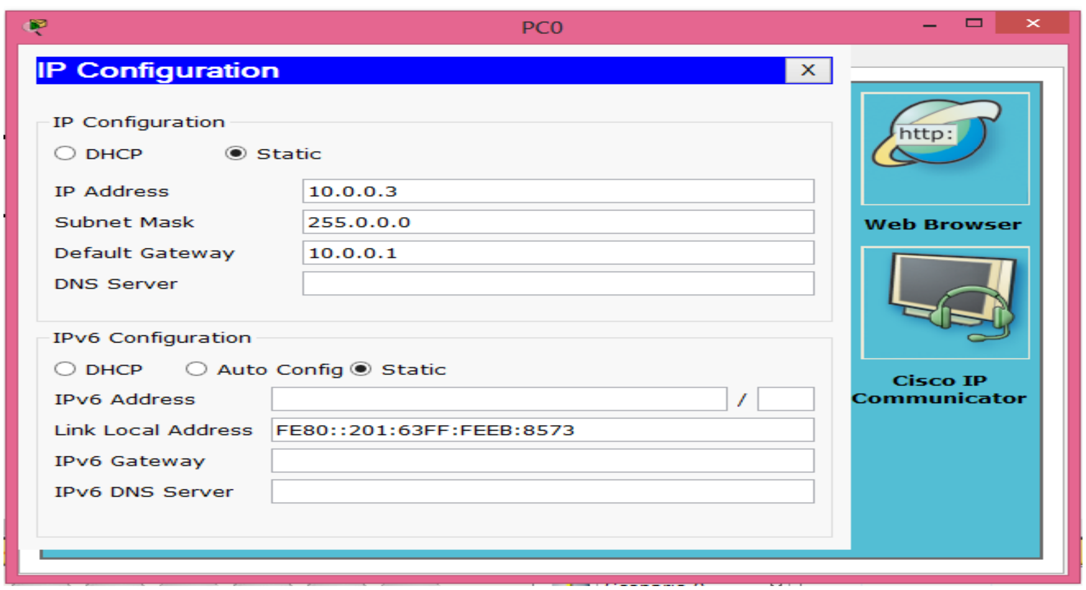


Step 1: Configuring the FTP server

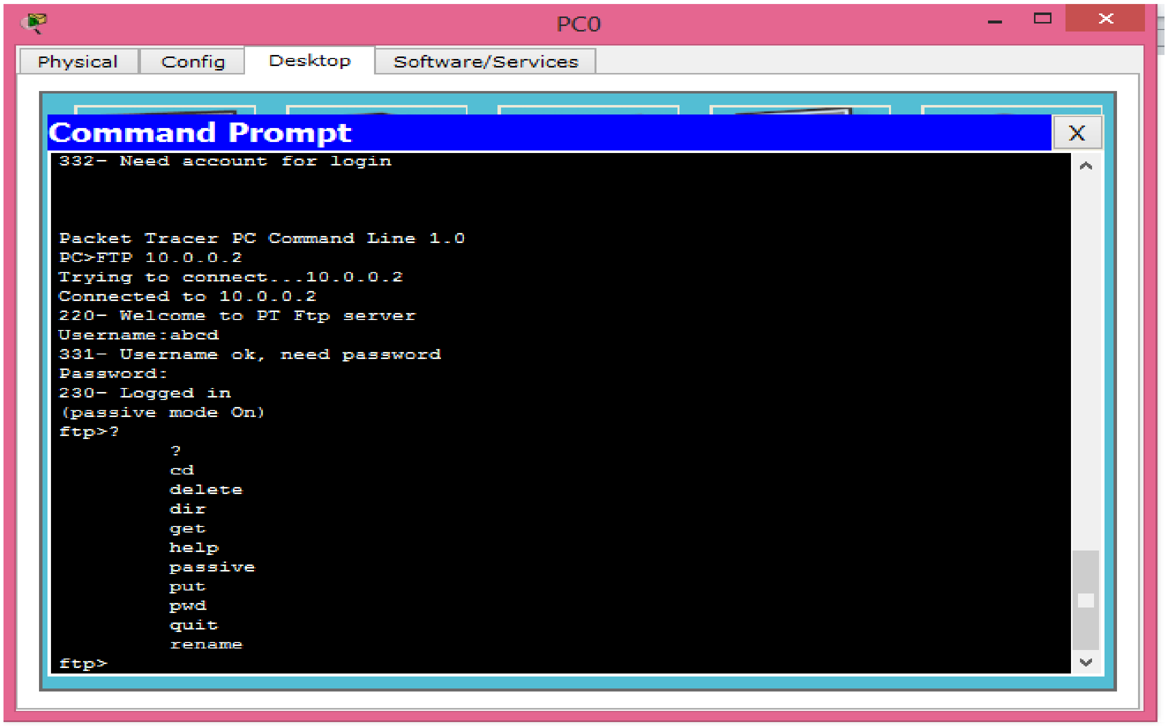




Step 2: Configuring PC0



Now we can verify the working as follows



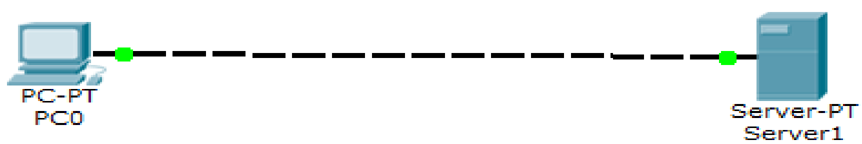
Hyper Text Transfer Protocol (HTTP)

The World Wide Web (WWW) is a repository of information linked together from points all over the world. Hypertext and hypermedia documents are linked to one another through pointers. The WWW architecture is made up of clients and servers. A client or a browserinterprets and displays a Web document. A browser consists of a controller, clientprograms, and interpreters. A server stores Web pages. A Web document can be classified as static, dynamic, or active. A static documentis one in which the contents are fixed and stored in a server. A dynamic Web document is created by a server only at a browser request. An active document is acopy of a program retrieved by the client and run at the client site.

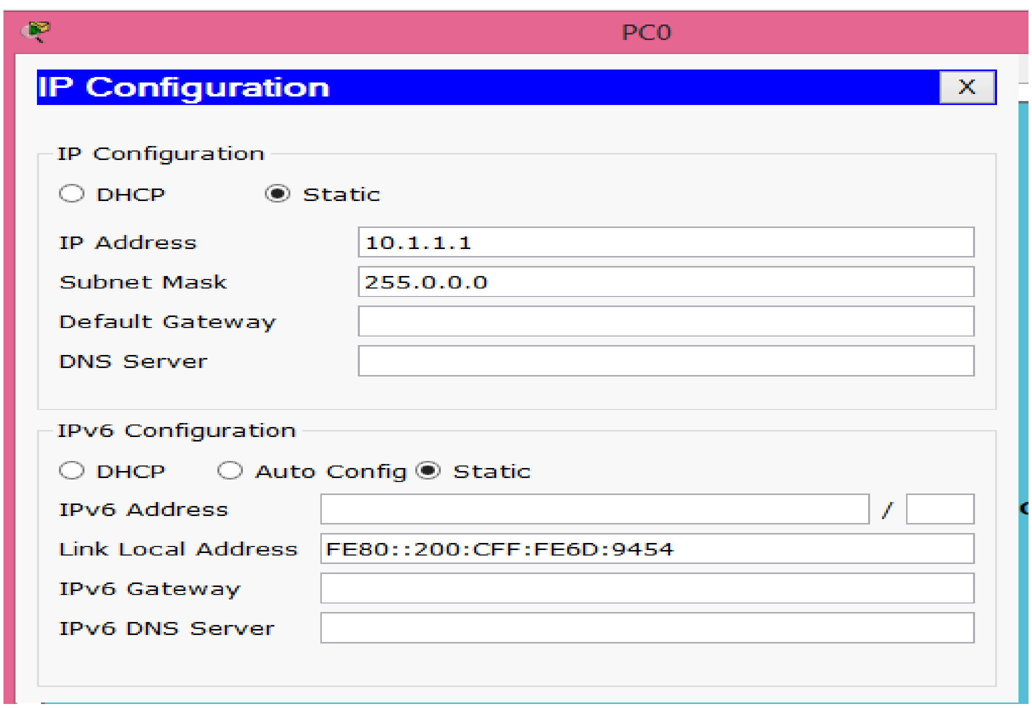
The Hypertext Transfer Protocol (HTTP) is the main protocol used to access dataon the World Wide Web (WWW). HTTP uses a TCP connection to transfer files. HTTP transactions are made of request and response messages.

HTTP can be used in two modes: nonpersistent and persistent. The nonpersistent mode uses a new TCP connection for each transaction; the persistent mode uses only one connection. The default in the new version of HTTP is the persistent mode. HTTP can use cookies to keep the state of the transactions. The server sends a cookie that can be stored in the client and be retrieved later by the server. Web caching using proxy servers improves the efficiency of the HTTP. The proxy servers are installed in the client sites.

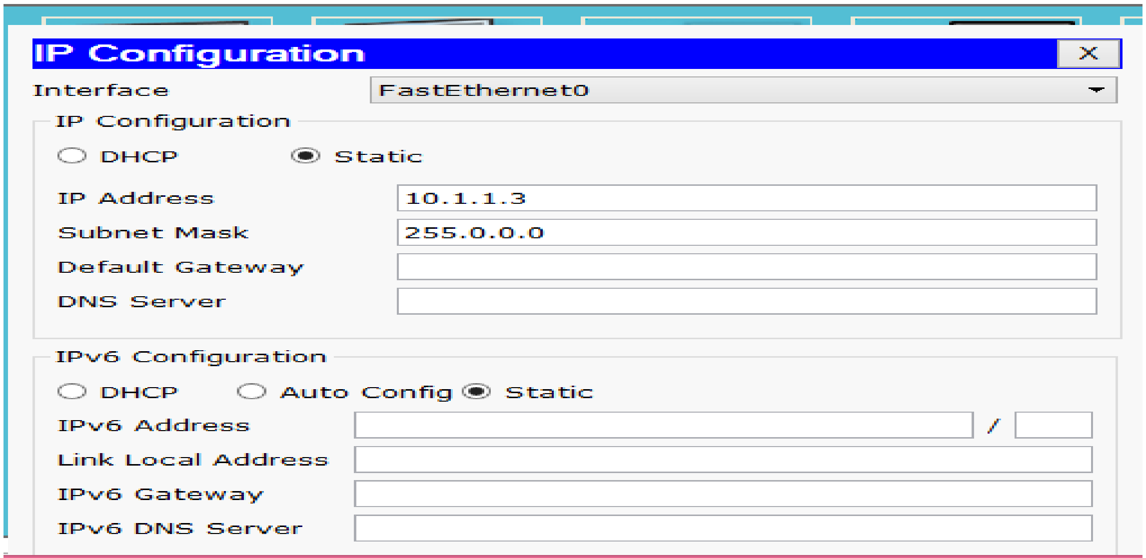
We can verify the given protocol using the following example



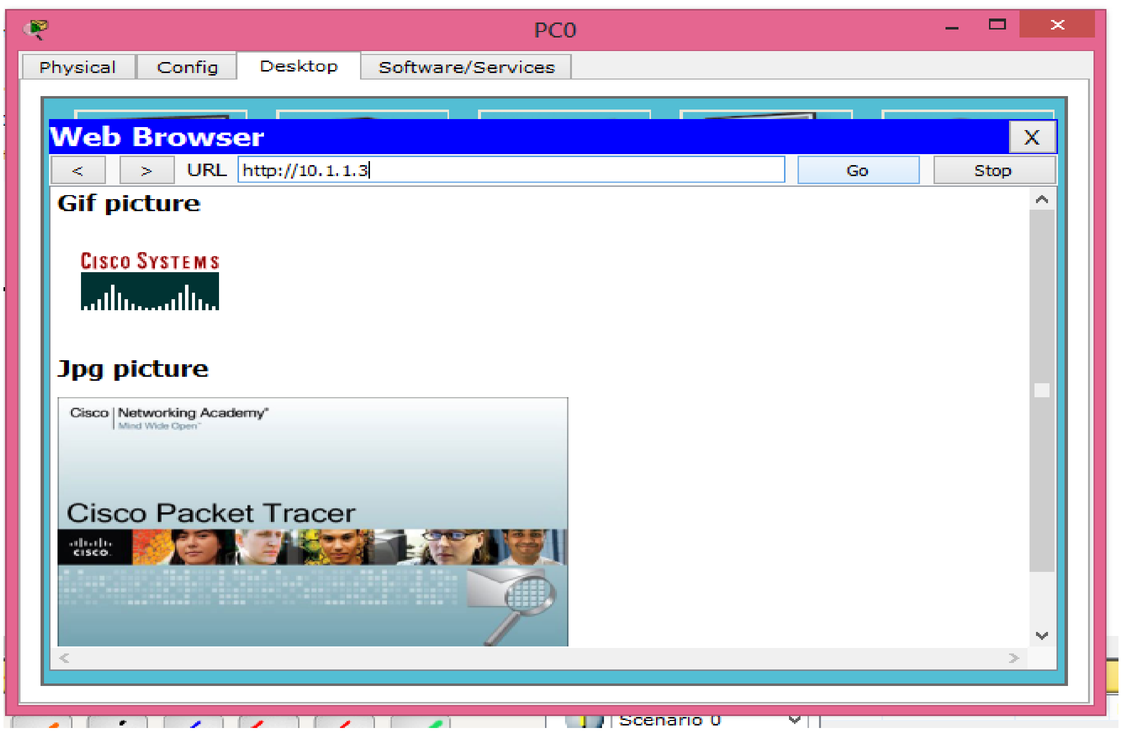
Step 1: Configuring PC0



Step 2: Configuring Server (note HTTP must be enabled)



Now we verify the given protocol using the following



TELNET

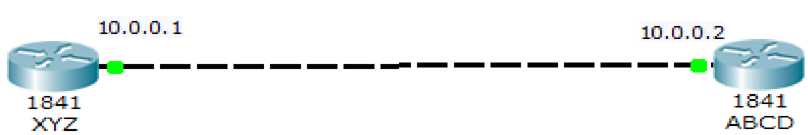
TELNET is a client-server application that allows a user to log on to a remote machine, giving the user access to the remote system. When a user accesses a remote system via the TELNET process, this is comparable to a time-sharing environment. A terminal driver correctly interprets the keystrokes on the local terminal or terminal emulator. This may not occur between a terminal and a remote terminal driver.

TELNET uses the Network Virtual Terminal (NVT) system to encode characters on the local system. On the server machine, NVT decodes the characters to a form acceptable to the remote machine. NVT uses a set of characters for data and a set of characters for control.

Options are features that enhance the TELNET process. TELNET allows negotiation to set transfer conditions between the client and server before and during the use of the service. Some options can only be enabled by the server, some only by the client, and some by both. An option is enabled or disabled through an offer or a request. An option that needs additional information requires the use of suboption characters.

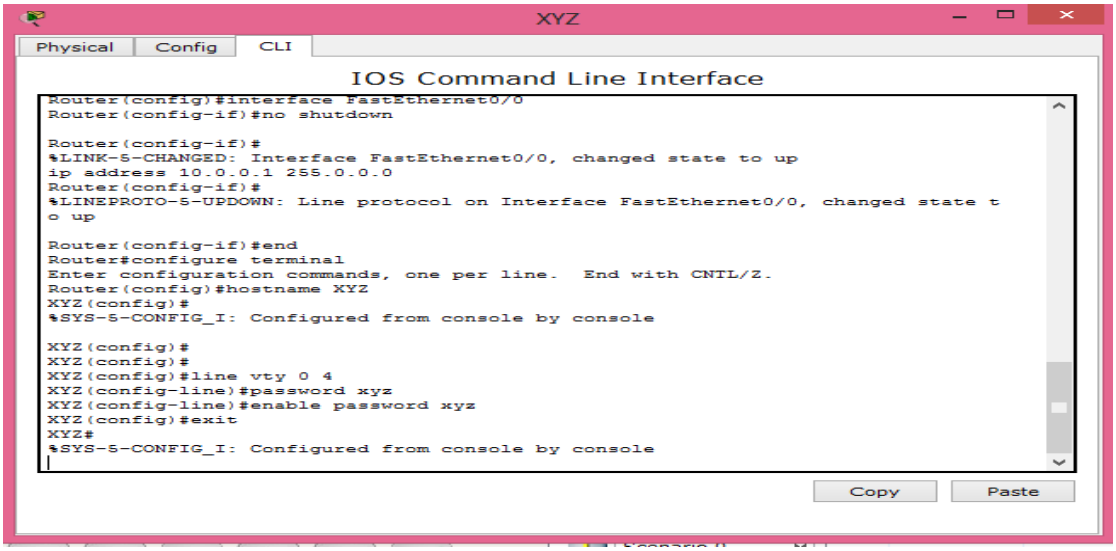
A TELNET implementation operates in the default, character, or line mode. In the default mode, the client sends one line at a time to the server and waits for the go ahead (GA) character before a new line from the user can be accepted. In the character mode, the client sends one character at a time to the server. In the line mode, the client sends one line at a time to the server, one after the other, without the need for an intervening GA character.

We study TELNET through the following example

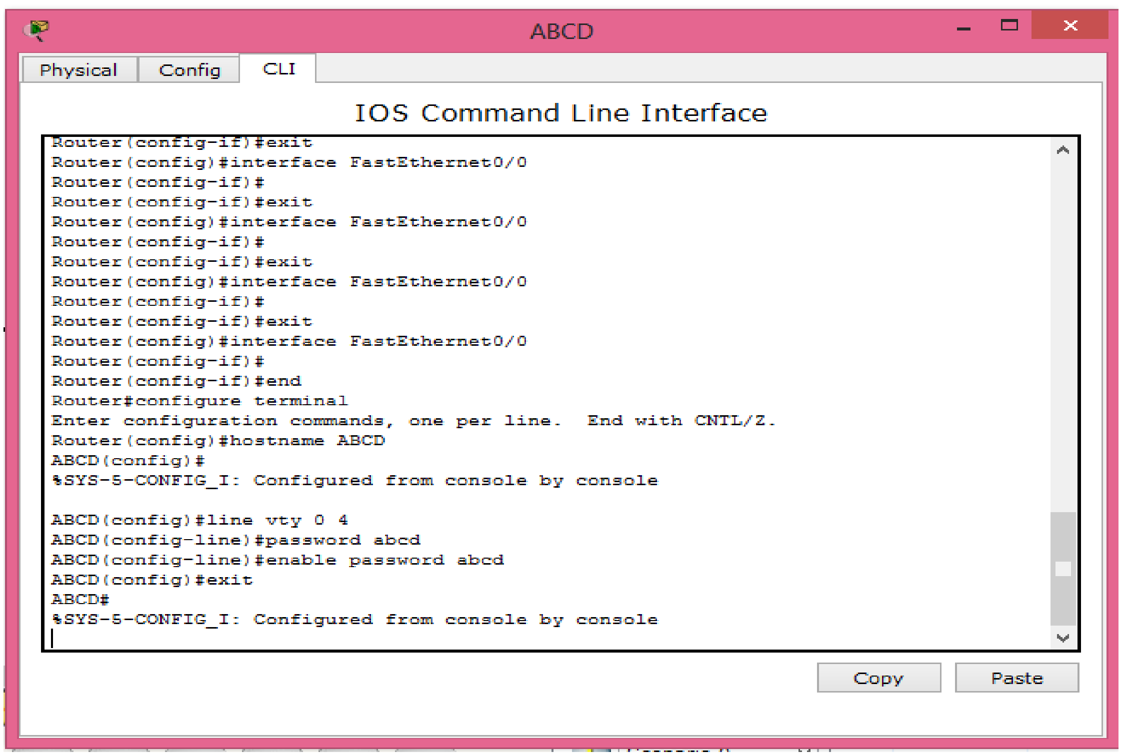


We configure the routers with the given IP addresses

The commands for XYZ is entered as follows



The commands for ABCD is entered as follows



We verify the working using the commands as follows