**PRACTICAL NO. 6**

**AIM: Using network protocol**

**1.Configure servers for**

**a.Telnet**

**b.DNS**

**c.HTTP**

**d.SMTP**

**e.FTP Servers**

**f.SNMP**

**THEORY:**

**Network Protocol**

Network Protocols are a set of rules governing exchange of information in an easy, reliable and secure way. Before we discuss the most common protocols used to transmit and receive data over a network, we need to understand how a network is logically organized or designed. The most popular model used to establish open communication between two systems is the Open Systems Interface (OSI) model proposed by ISO.

1. **Telnet**

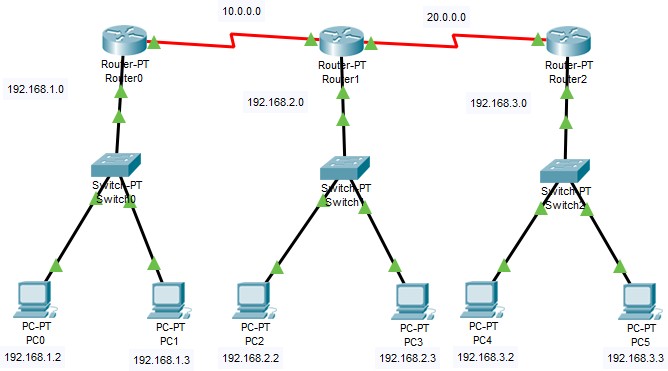
Telnet is an application protocol used on the Internet or local area network to provide a bidirectional interactive text-oriented communication facility using a virtual terminal connection. User data is interspersed in-band with Telnet control information in an 8-bit byte oriented data connection over the Transmission Control Protocol (TCP).

Telnet was developed in 1969 beginning with RFC 15, extended in RFC 855, and standardized as Internet Engineering Task Force (IETF) Internet Standard STD 8, one of the first Internet standards. The name stands for "teletype network".

Historically, Telnet provided access to a command-line interface on a remote host. However, because of serious security concerns when using Telnet over an open network such as the Internet, its use for this purpose has waned significantly in favour of SSH.

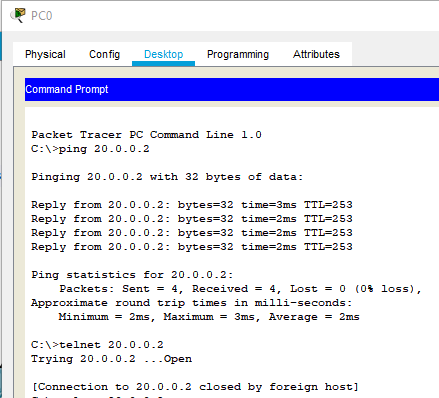
The term telnet is also used to refer to the software that implements the client part of the protocol. Telnet client applications are available for virtually all computer platforms. Telnet is also used as a verb. To telnet means to establish a connection using the Telnet protocol, either with a command line client or with a graphical interface. For example, a common directive might be: "To change your password, telnet into the server, log in and run the passwd command." In most cases, a user would be telnetting into a Unix-like server system or a network device (such as a router).

# TOPOLOGY:

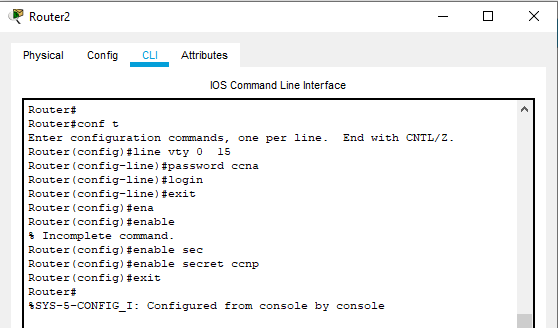


**CONFIGURATION:**

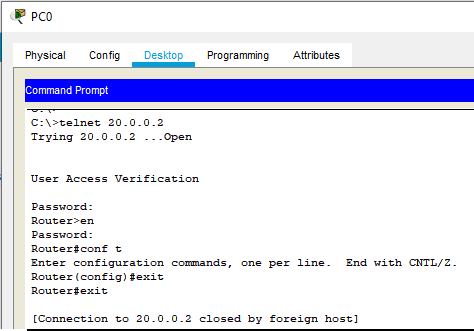
Connection from PC0 to Router2 using telnet.



Configure Router2 for telnet connection.



Verifying Connection from PC0:



# DNS

An application layer protocol defines how the application processes running on different systems, pass the messages to each other.

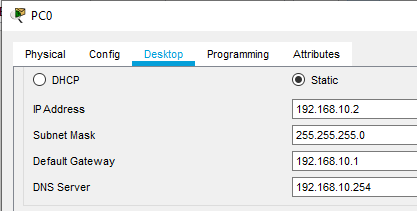
* + DNS stands for Domain Name System.
  + DNS is a directory service that provides a mapping between the name of a host on the network and its numerical address.
  + DNS is required for the functioning of the internet.
  + Each node in a tree has a domain name, and a full domain name is a sequence of symbols specified by dots.
  + DNS is a service that translates the domain name into IP addresses. This allows the users of networks to utilize user-friendly names when looking for other hosts instead of remembering the IP addresses.
  + For example, suppose the FTP site at EduSoft had an IP address of 132.147.165.50, most people would reach this site by specifying ftp.EduSoft.com. Therefore, the domain name is more reliable than IP address.

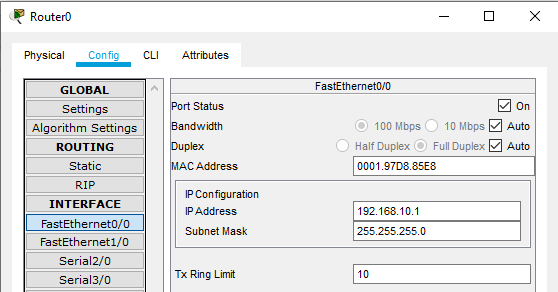
DNS is a TCP/IP protocol used on different platforms. The domain name space is divided into three different sections: generic domains, country domains, and inverse domain.

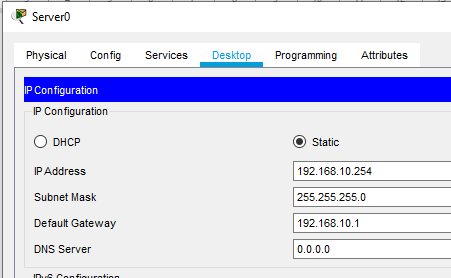
Working of DNS

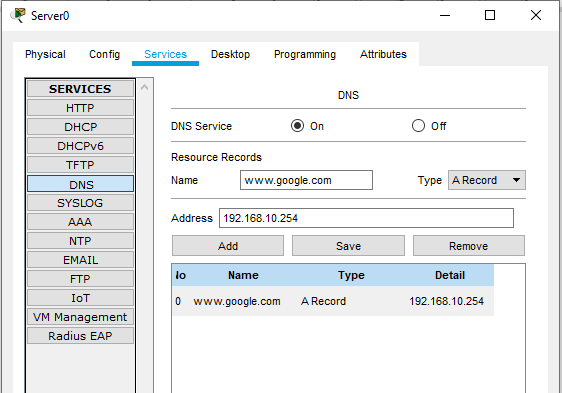
* + DNS is a client/server network communication protocol. DNS clients send requests to the. server while DNS servers send responses to the client.
  + Client requests contain a name which is converted into an IP address known as a forward DNS lookup while requests containing an IP address which is converted into a name known as reverse DNS lookups.
  + DNS implements a distributed database to store the name of all the hosts available on the internet.
  + If a client like a web browser sends a request containing a hostname, then a piece of software such as DNS resolver sends a request to the DNS server to obtain the IP address of a hostname. If DNS server does not contain the IP address associated with a hostname, then it forwards the request to another DNS server. If IP address has arrived at the resolver, which in turn completes the request over the internet protocol.

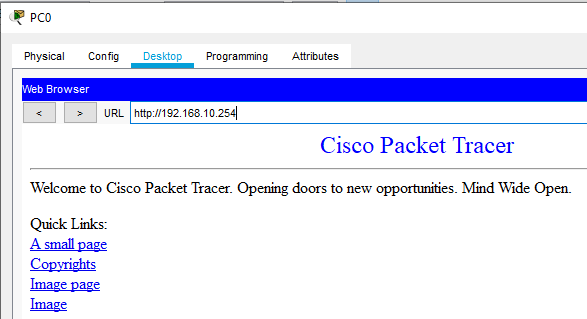
# TOPOLOGY:

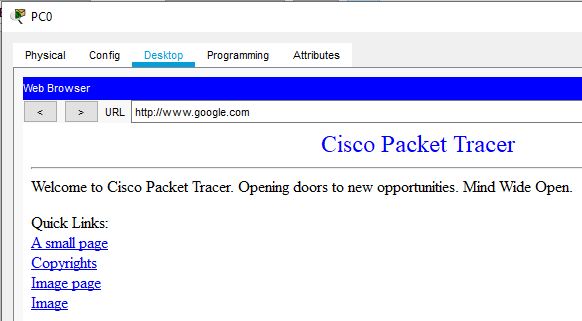
**CONFIGURATION: PC0:**

**ROUTER0:**

**SERVER0:**



**OUTPUT:**



**b.HTTP**

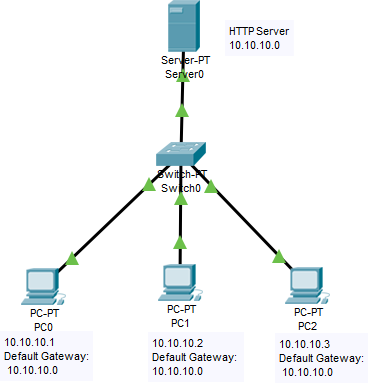
* **HTTP** stands for **H**yper **T**ext **T**ransfer **P**rotocol
* **WWW** is about communication between web **clients** and **servers**
* Communication between client computers and web servers is done by sending **HTTP Requests** and receiving **HTTP Responses**

# HTTP Request / Response

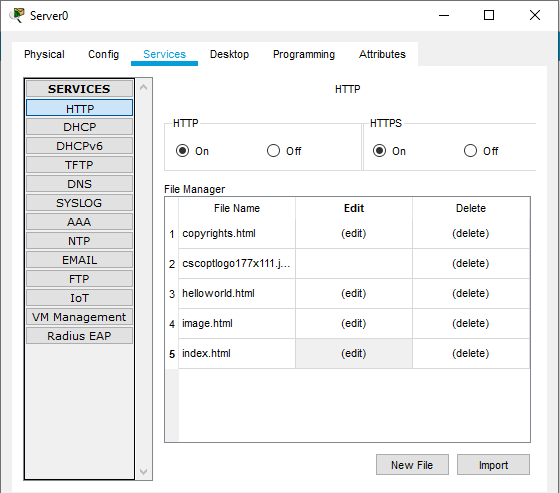
Communication between clients and servers is done by **requests** and **responses**:

1. A client (a browser) sends an **HTTP request** to the web
2. A web server receives the request
3. The server runs an application to process the request
4. The server returns an **HTTP response** (output) to the browser
5. The client (the browser) receives the response

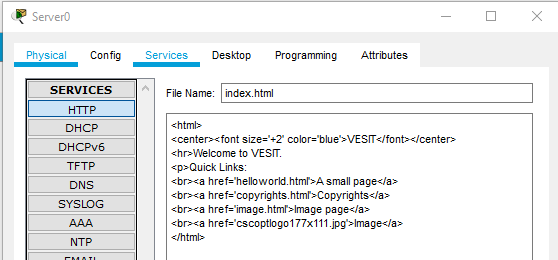
# TOPOLOGY:



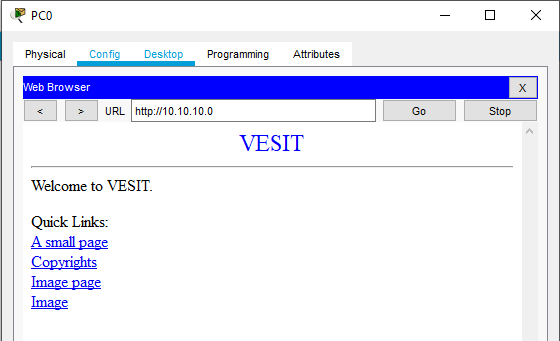
**CONFIGURATION:**

**SERVER0:**

Click on the Edit of index.html to edit the webpage.



# OUTPUT:

Click on Web Browser from Desktop Tab.

# SMTP

SMTP stands for Simple Mail Transfer Protocol.

* + SMTP is a set of communication guidelines that allow software to transmit an electronic mail over the internet is called Simple Mail Transfer Protocol.
  + It is a program used for sending messages to other computer users based on e-mail addresses.
  + It provides a mail exchange between users on the same or different computers, and it also supports:
    - It can send a single message to one or more recipients.
    - Sending message can include text, voice, video or graphics.
    - It can also send the messages on networks outside the internet.
  + The main purpose of SMTP is used to set up communication rules between servers. The servers have a way of identifying themselves and announcing what kind of communication they are trying to perform. They also have a way of handling the errors such as incorrect email address. For example, if the recipient address is wrong, then receiving server reply with an error message of some kind.

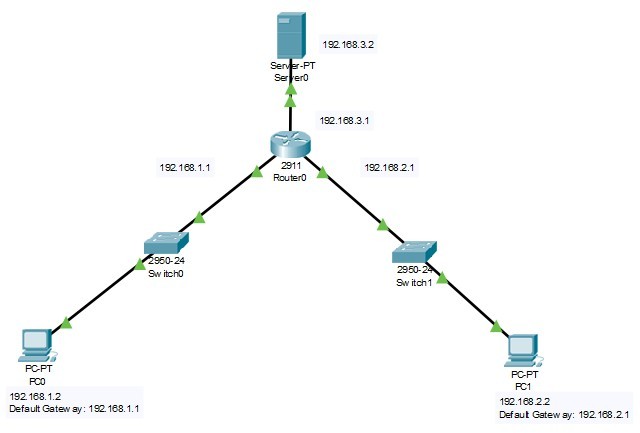
Working of SMTP

* + Composition of Mail: A user sends an e-mail by composing an electronic mail message using a Mail User Agent (MUA). Mail User Agent is a program which is used to send and receive mail. The message contains two parts: body and header. The body is the main part of the message while the header includes information such as the sender and recipient address. The header also includes descriptive information such as the subject of the message. In this case, the message body is like a letter and header is like an envelope that contains the recipient's address.
  + Submission of Mail: After composing an email, the mail client then submits the completed e-mail to the SMTP server by using SMTP on TCP port 25.
  + Delivery of Mail: E-mail addresses contain two parts: username of the recipient and domain name. For example, mca@gmail.com, where "mca" is the username of the recipient and "gmail.com" is the domain name.

If the domain name of the recipient's email address is different from the sender's domain name, then MSA will send the mail to the Mail Transfer Agent (MTA). To relay the email, the MTA will find the target domain. It checks the MX record from Domain Name System to obtain the target domain. The MX record contains the domain name and IP address of the recipient's domain. Once the record is located, MTA connects to the exchange server to relay the message.

* + Receipt and Processing of Mail: Once the incoming message is received, the exchange server delivers it to the incoming server (Mail Delivery Agent) which stores the e-mail where it waits for the user to retrieve it.
  + Access and Retrieval of Mail: The stored email in MDA can be retrieved by using MUA (Mail User Agent). MUA can be accessed by using login and password.

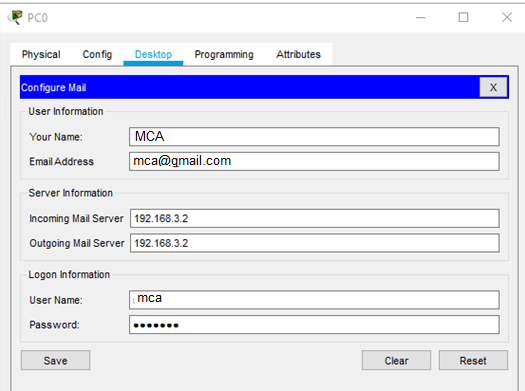
# TOPOLOGY:



**CONFIGURATION:**

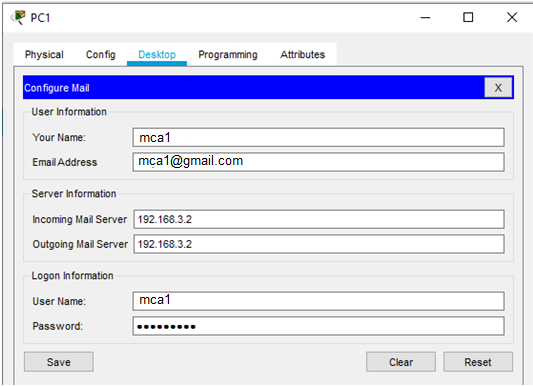
**PC0:**

**Click Desktop – Email – Fill the Details – Username and Password from Server0.**

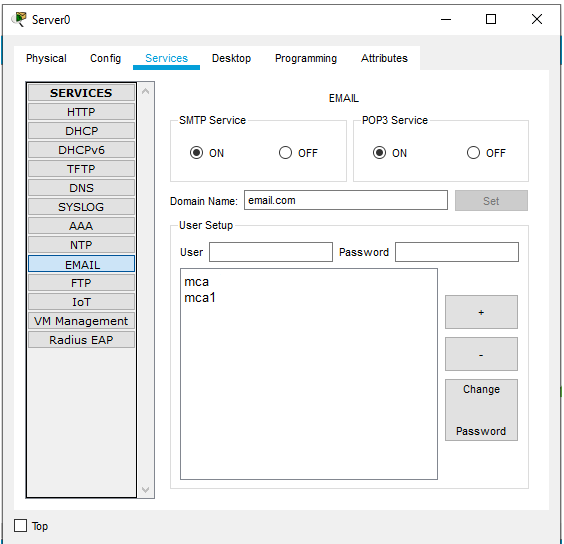


**PC1:**

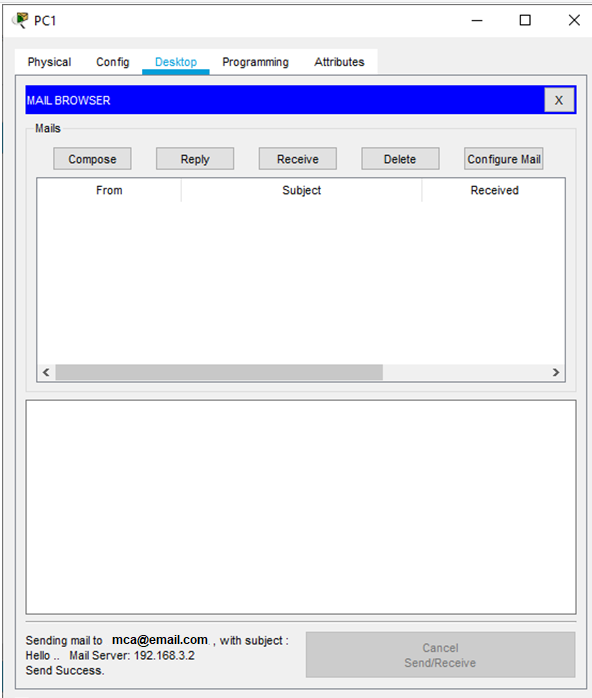
**Click Desktop – Email – Fill the Details – Username and Password from Server0.**

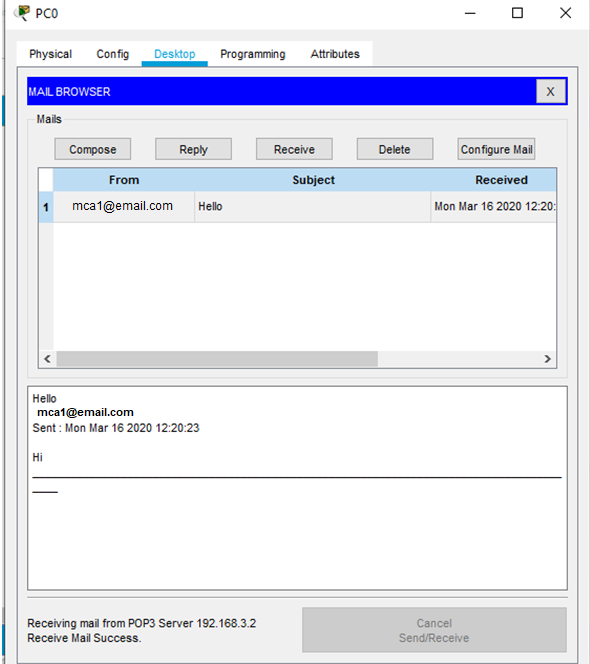


**SERVER0:**



**OUTPUT:**





1. **FTP Servers.**

**DESCRIPTION:**

The File Transfer Protocol (FTP) is a standard network protocol used for the transfer of computer files between a client and server on a computer network.

FTP is built on a client-server model architecture using separate control and data connections between the client and the server.[1] FTP users may authenticate themselves with a clear-text sign- in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it. For secure transmission that protects the username and password, and encrypts the content, FTP is often secured with SSL/TLS (FTPS) or replaced with SSH File Transfer Protocol (SFTP).

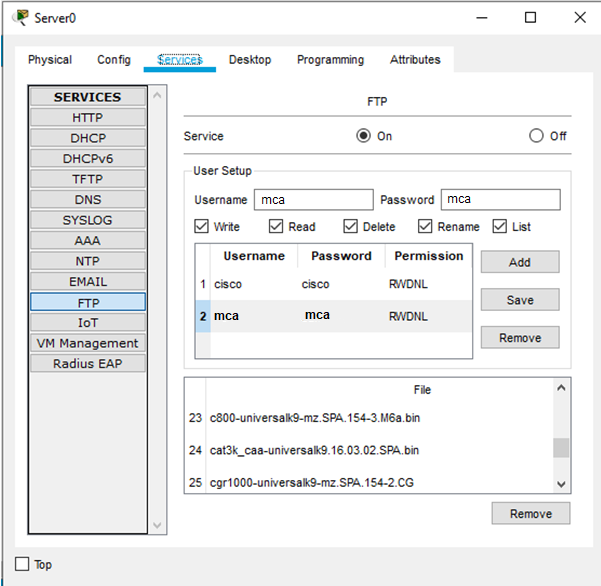
The first FTP client applications were command-line programs developed before operating systems had graphical user interfaces, and are still shipped with most Windows, Unix, and Linux operating systems.[2][3] Many FTP clients and automation utilities have since been developed for desktops, servers, mobile devices, and hardware, and FTP has been incorporated into productivity applications, such as HTML editors.

# TOPOLOGY:

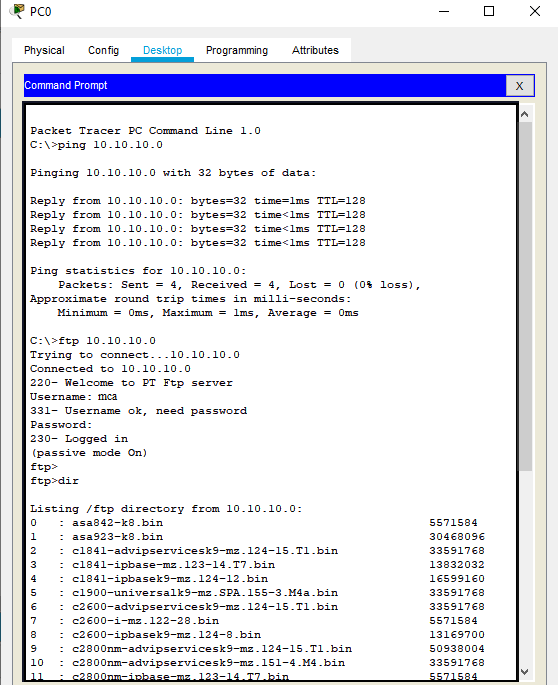
**CONFIGURATION:**

**SERVER0:**

**Click Services Tab and select FTP. Fill the Details as shown below and click on Add.**



**OUTPUT:**



1. **SNMP DESCRIPTION:**

**Simple Network Management Protocol (SNMP) –**

SNMP is an application layer protocol which uses UDP port number 161/162.SNMP is used to monitor the network, detect network faults and sometimes even used to configure remote devices.

# SNMP components –

There are 3 components of SNMP:

1. SNMP Manager –

It is a centralised system used to monitor network.It is also known as Network Management Station (NMS)

1. SNMP agent –

It is a software management software module installed on a managed device. Managed devices can be network devices like PC, router, switches, servers etc.

1. Management Information Base –

MIB consists of information of resources that are to be managed. These information is organised hierarchically. It consists of objects instances which are essentially variables.

# SNMP messages –

Different variables are:

1. GetRequest –

SNMP manager sends this message to request data from SNMP agent. It is simply used to retrieve data from SNMP agent. In response to this, SNMP agent responds with requested value through response message.

1. GetNextRequest –

This message can be sent to discover what data is available on a SNMP agent. The SNMP manager can request for data continuously until no more data is left. In this way, SNMP manager can take knowledge of all the available data on SNMP agent.

1. GetBulkRequest –

This message is used to retrieve large data at once by the SNMP manager from SNMP agent. It is introduced in SNMPv2c.

1. SetRequest –

It is used by SNMP manager to set the value of an object instance on the SNMP agent.

1. Response –

It is a message send from agent upon a request from manager. When sent in response

to Get messages, it will contain the data requested. When sent in response to Set message, it will contain the newly set value as confirmation that the value has been set.

1. Trap –

These are the message send by the agent without being requested by the manager. It is sent when a fault has occurred.

1. InformRequest –

It was introduced in SNMPv2c, used to identify if the trap message has been received by the manager or not. The agents can be configured to set trap continuously until it receives an Inform message. It is same as trap but adds an acknowledgement that trap doesn’t provide.

# SNMP security levels –

It defines the type of security algorithm performed on SNMP packets. These are used in only SNMPv3. There are 3 security levels namely:

1. noAuthNoPriv –

This (no authentication, no privacy) security level uses community string for authentication and no encryption for privacy.

1. AuthNopriv – This security level (authentication, no privacy) uses HMAC with Md5 for authentication and no encryption is used for privacy.
2. AuthPriv – This security level (authentication, privacy) uses HMAC with Md5 or SHA for authentication and encryption uses DES-56 algorithm.

# SNMP versions –

There are 3 versions of SNMP:

1. SNMPv1 –

It uses community strings for authentication and use UDP only.

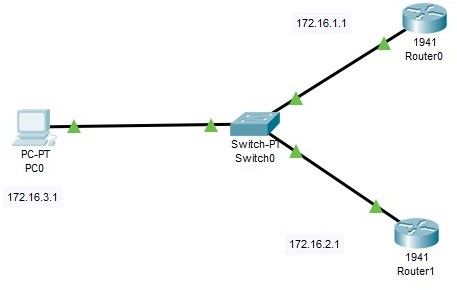
1. SNMPv2c –

It uses community strings for authentication. It uses UDP but can be configured to use TCP.

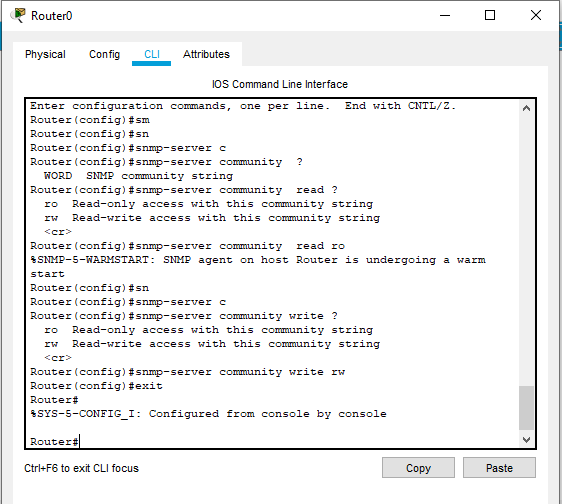
1. SNMPv3 –

It uses Hash based MAC with MD5 or SHA for authentication and DES-56 for privacy.This version uses TCP. Therefore, conclusion is the higher the version of SNMP, more secure it will be.

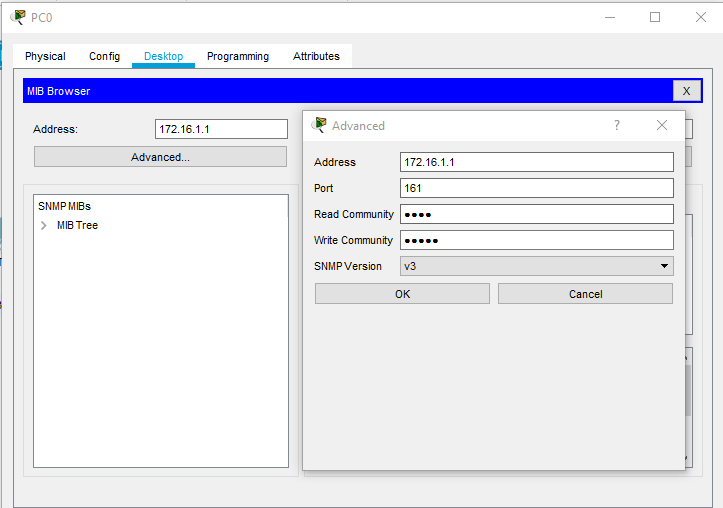
# TOPOLOGY:

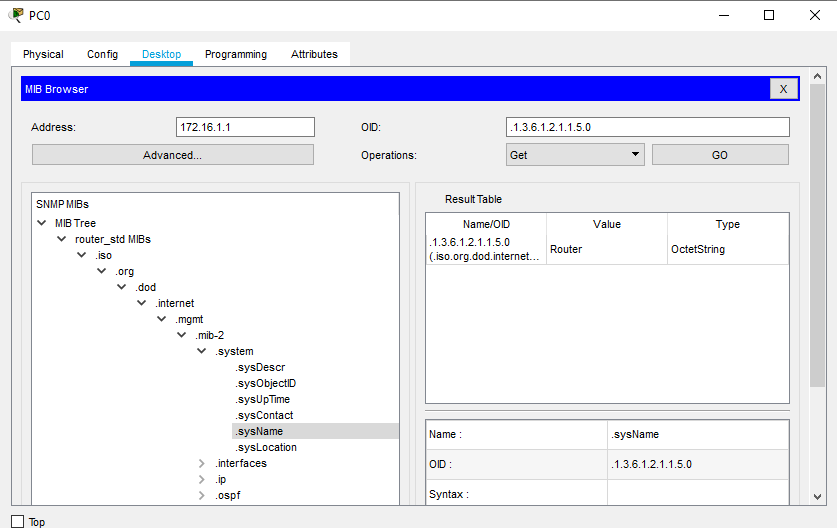


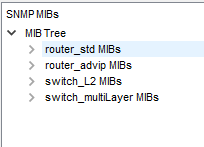
**CONFIGURATION: ROUTER0:**



**PC0:**

**Open Desktop Tab . Select MIB Browser. Enter the Address as shown below and click on Advance:**

**OUTPUT:**



**CONCLUSION:**

We have successfully configured servers for Telnet, Dns, HTTP, SMTP, FTP Server and SNMP using network protocols.