



CN Unit-5 Notes

Computer Networks (Dr. A.P.J. Abdul Kalam Technical University)



Scan to open on Studocu

Computer Network Unit-5

DOMAIN NAME SPACE (DNS)

To have a hierarchical name space, a domain name space was designed. In this design the names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 (root) to level 127.

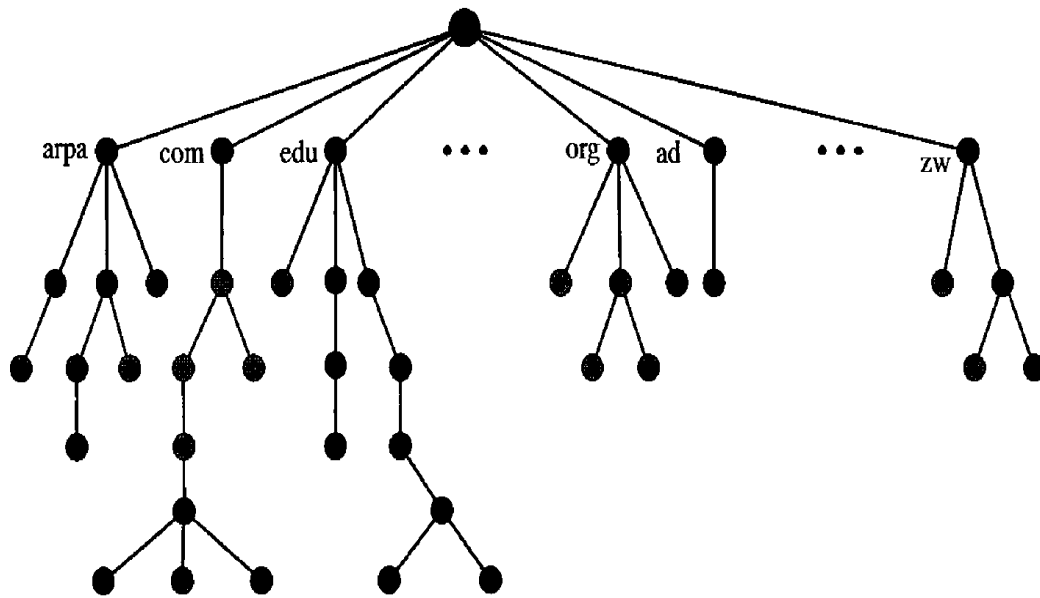


Figure 1

Label

Each node in the tree has a label, which is a string with a maximum of 63 characters. The root label is a null string (empty string).

Domain Name

Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.). The domain names are always read from the node up to the root. The last label is the label of the root (null).

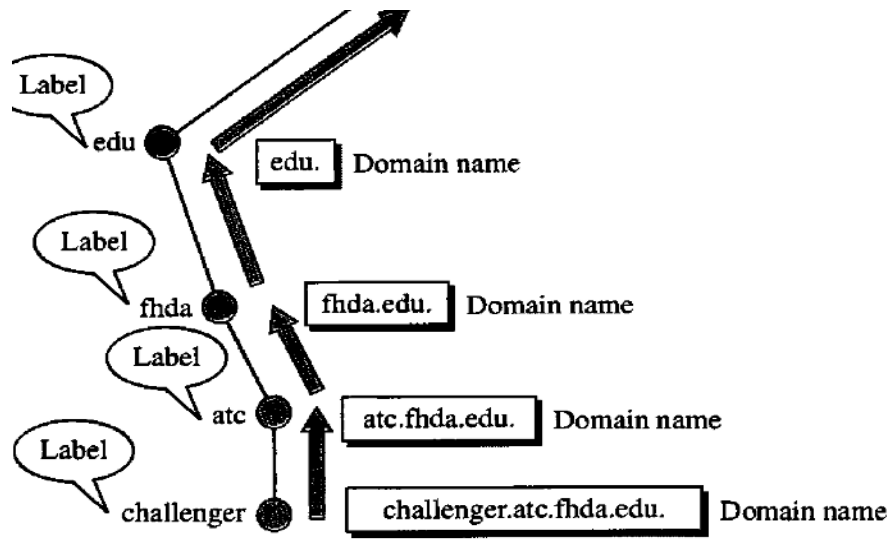


Figure 2

Fully Qualified Domain Name

If a label is terminated by a null string, it is called a fully qualified domain name (FQDN). An FQDN is a domain name that contains the full name of a host.

Example: challenger.atc.fhda.edu.

Partially Qualified Domain Name

If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN). A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client.

Example: challenger

Domain

A domain is a sub-tree of the domain name space. The name of the domain is the domain name of the node at the top of the sub-tree. Figure 3 shows some domains. Note that a domain may itself be divided into domains (or subdomains as they are sometimes called).

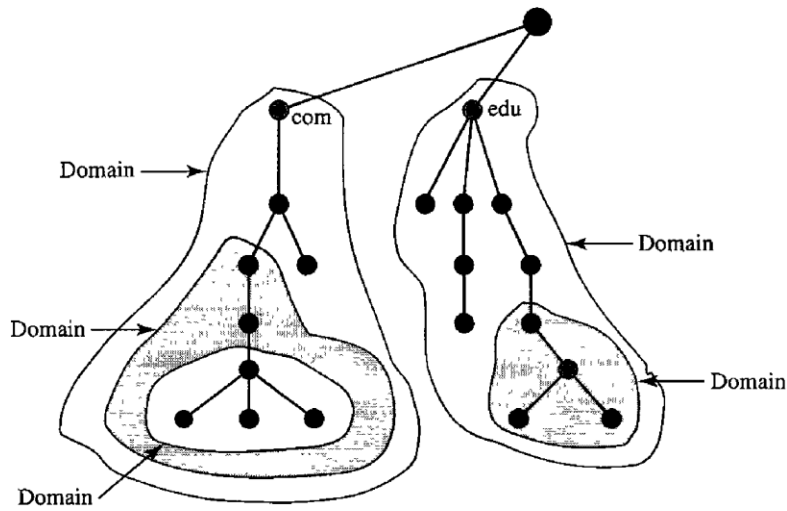


Figure 3

DNS in the Internet

DNS is a protocol that can be used in different platforms. In the Internet, the domain name space (tree) is divided into three different sections: generic domains, country domains, and the inverse domain.

Generic Domains

The generic domains define registered hosts according to their generic behavior. Each node in the tree defines a domain, which is an index to the domain name space database.

Example:

- com -Commercial organizations
- edu-Educational institutions ,
- gov- Government institutions ,
- info -Information service providers
- int-International organizations,
- net - Network support centers ,
- org -Nonprofit organizations

Country Domains

The country domains section uses two-character country abbreviations (e.g., us for United States). Second labels can be organizational, or they can be more specific, national designations. The United States, for example, uses state abbreviations as a subdivision of us.

Inverse Domain

The inverse domain is used to map an address to a name. This may happen, for example, when a server has received a request from a client to do a task. Although the server has a file that contains a list of authorized clients, only the IP address of the client (extracted from the received IP packet) is listed. The server asks its resolver to send a query to the DNS server to map an address to a name to determine if the client is on the authorized list.

TELNET

TELNET is an abbreviation for **TErminaL NETwork**. It is the standard TCP/IP protocol for virtualterminal service as proposed by the International Organization for Standards (ISO). TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system.

Note: TELNET is a general purpose client/server application program.

Timesharing Environment

TELNET was designed at a time when most operating systems, such as UNIX, were operating in a timesharing environment. In such an environment, a large computer supports multiple users.

Logging

In a timesharing environment, users are part of the system with some right to access resources. Each authorized user has identification and probably, a password. The user identification defines the user as part of the system. To access the system, the user logs into the system with a user id or log-in name. The system also includes password checking to prevent an unauthorized user from accessing resources.

When a user wants to access an application program or utility located on a remote machine, she performs remote log-in. Here the TELNET client and server programs come into use. The user sends the keystrokes to the terminal driver, where the local operating system accepts the characters but does not interpret them. The characters are sent to the TELNET client, which transforms the characters to a

universal character set called Network Virtual Terminal (NVT) characters and delivers them to the local TCP/IP protocol stack.

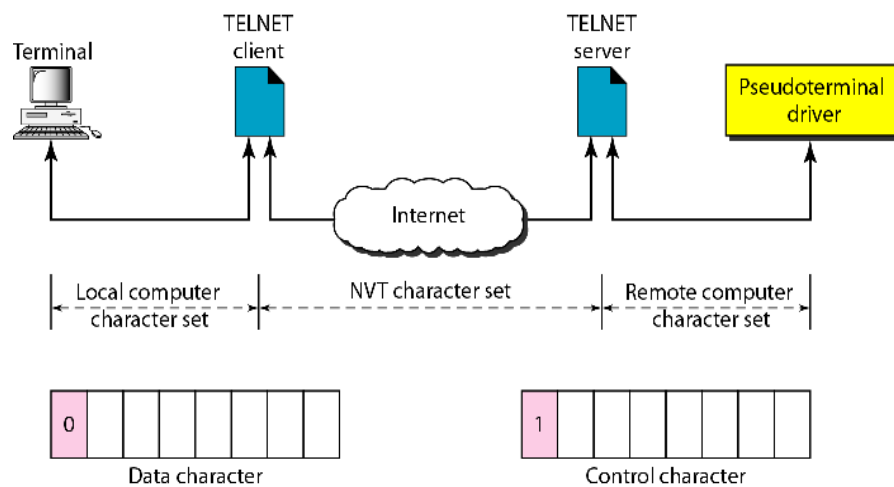


Figure 4

ELECTRONIC MAIL

This is most common scenario, Bob is also connected to his mail server by a WAN or a LAN. After the message has arrived at Bob's mail server, Bob needs to retrieve it. Here, we need another set of client/server agents, which we call message access agents (MAAs). Bob uses an MAA client to retrieve his messages. The client sends a request to the MAA server, which is running all the time, and requests the transfer of the messages.

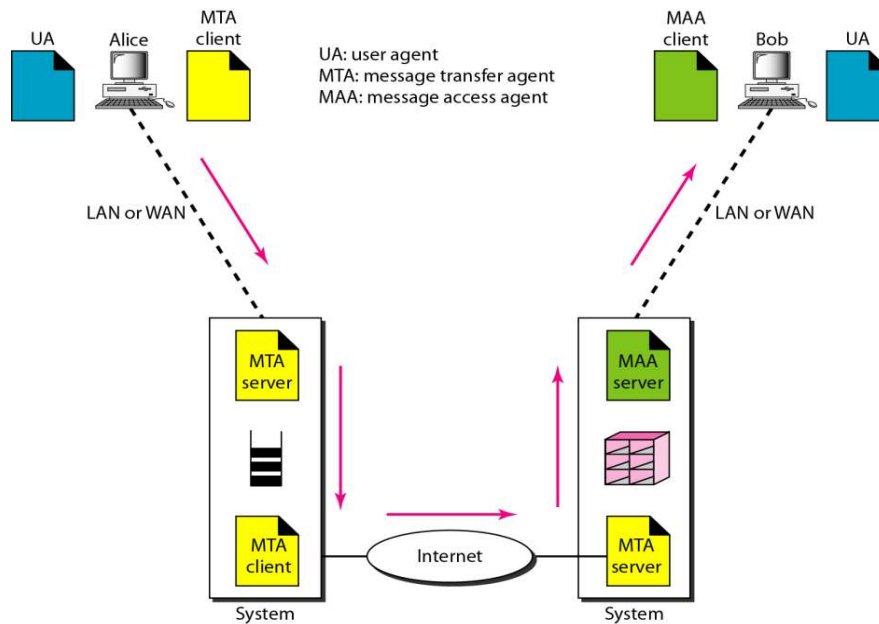


Figure 5

When both sender and receiver are connected to the mail server via a LAN or a WAN, we need two VAs, two pairs of MTAs (client and server), and a pair of MAAs (client and server). This is the most common situation today.

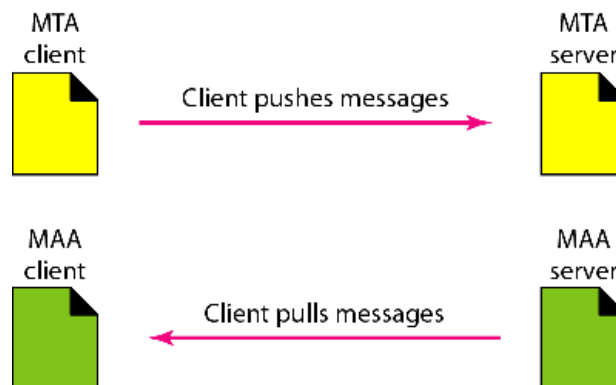


Figure 6

User Agent

The first component of an electronic mail system is the user agent (VA). It provides service to the user to make the process of sending and receiving a

message easier. A user agent is a software package (program) that composes reads, replies to, and forwards messages. It also handles mailboxes

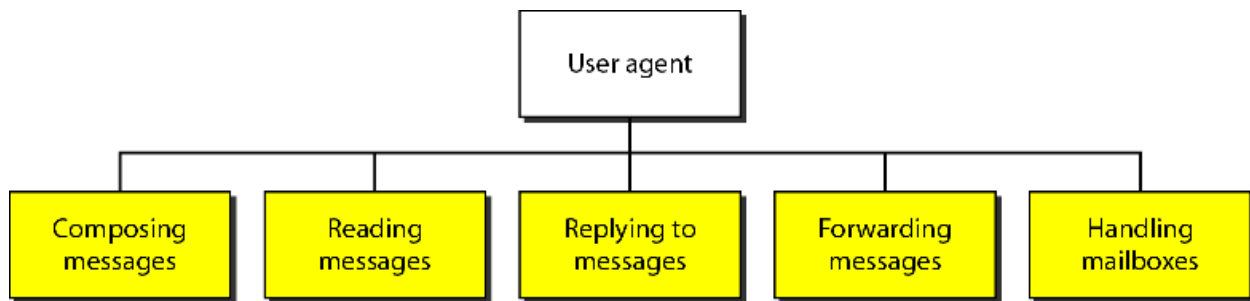


Figure 7

User Agent Types

There are two types of user agents: command-driven and GUI-based.

Command-driven

User agent normally accepts a one-character command from the keyboard to perform its task. For example, a user can type the character *r*, at the command prompt, to reply to the sender of the message, or type the character *R* to reply to the sender and all recipients. Some examples of command-driven user agents are *mail*, *pine*, and *elm*.

GUI-Based Modem user agents are GUI-based.

They contain Graphical User Interface (GUI) components that allow the user to interact with the software by using both the keyboard and the mouse. They have graphical components such as icons, menu bars, and windows that make the services easy to access. Some examples of GUI-based user agents are Eudora, Microsoft's Outlook, and Netscape.

Addresses

To deliver mail, a mail handling system must use an addressing system with unique addresses. In the Internet, the address consists of two parts: a local part and a domain name, separated by an @ sign.



Figure 8

Local Part

The local part defines the name of a special file, called the user mailbox, where all the mail received for a user is stored for retrieval by the message access agent.

Domain Name

The second part of the address is the domain name. An organization usually selects one or more hosts to receive and send e-mail; the hosts are sometimes called *mail servers* or *exchangers*.

MIME

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 them to the client MTA to be sent through the Internet. The message at the receiving side is transformed back to the original data. We can think of MIME as a set of software functions that transforms non-ASCII data (stream of bits) to ASCII data and vice versa.

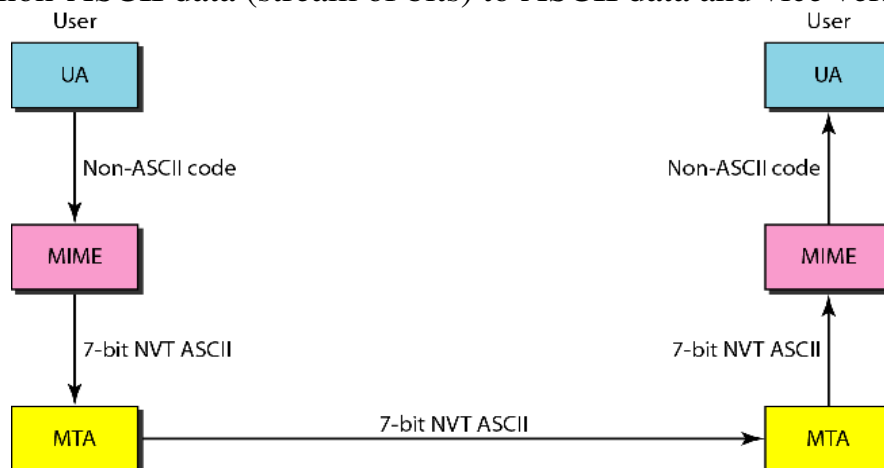


Figure 9

MIME defines five headers that can be added to the original e-mail header section to define the transformation parameters:

1. MIME-Version
2. Content-Type
3. Content-Transfer-Encoding
4. Content-Id
5. Content-Description

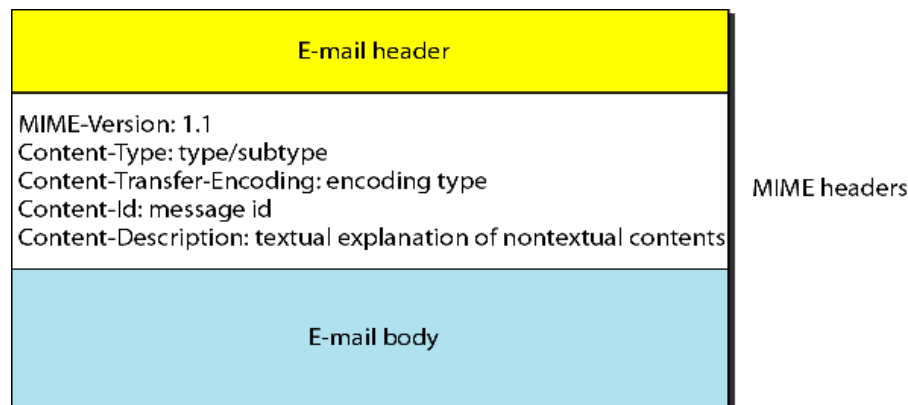


Figure 10

MIME-Version

This header defines the version of MIME used. The current version is 1.1.

Content-Type

This header defines the type of data used in the body of the message. The content type and the content subtype are separated by a slash.

Content-Transfer-Encoding

This header defines the method used to encode the messages into 0s and 1s for transport.

Content-Id

This header uniquely identifies the whole message in a multiple-message environment.

Content-Description

This header defines whether the body is image, audio, or video.

Message Transfer Agent: SMTP

The actual mail transfer is done through message transfer agents. 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 the 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. SMTP simply defines how commands and responses must be sent back and forth.

Commands and Responses

SMTP uses commands and responses to transfer messages between an MTA client and an MTA server.

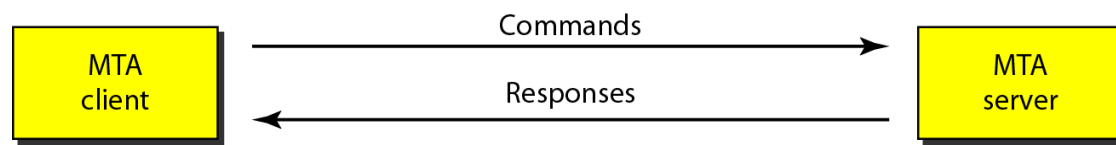


Figure 11

Each command or reply is terminated by a two-character (carriage return and line feed) end-of-line token. Commands are sent from the client to the server. It consists of a keyword followed by zero or more arguments. SMTP defines 14 commands. The first five are mandatory; every implementation must support these five commands.

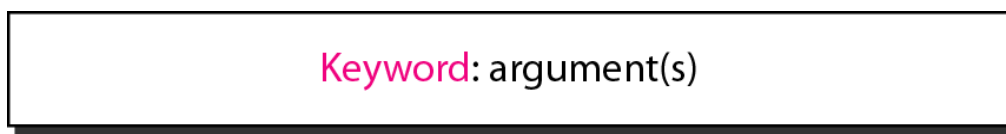


Figure 12

Example:

Keywords	Arguments
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPTTO	Intended recipient of the message

Responses are sent from the server to the client. A response is a three digit code that may be followed by additional textual information.

Code	Description
211	System status or help reply
214	Help message
220	Service ready

Message Access Agent: POP and IMAP

SMTP is a *push* protocol; it pushes the message from the client to the server. In other words, the direction of the bulk data (messages) is from the client to the server. On the other hand, the third stage needs a *pull* protocol; the client must pull messages from the server. The direction of the bulk data is from the server to the client. The third stage uses a message access agent.

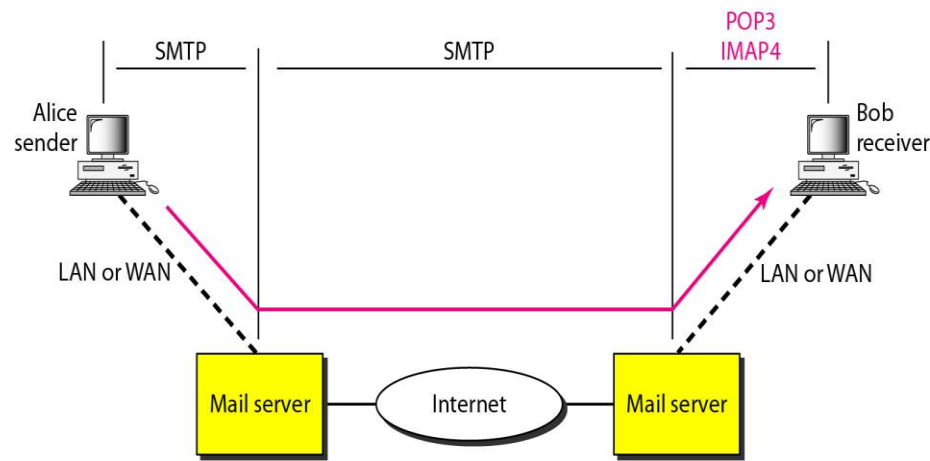


Figure 13

POP3 (POST OFFICE PROTOCOL VERSION 3)

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 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.

POP3 has two modes: the delete mode and the keep mode. In the delete mode, the mail is deleted from the mailbox after each retrieval. In the keep mode, the mail remains in the mailbox after retrieval. The delete mode is normally used when the user is working at her permanent computer and can save and organize the received mail after reading or replying. The keep mode is normally used when the user accesses her mail away from her primary computer (e.g., a laptop). The mail is read but kept in the system for later retrieval and organizing

IMAP4

Another mail access protocol is Internet Mail Access Protocol, version 4 (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. POP3 does not allow the user to partially check the contents of the mail before downloading.

IMAP4 provides the following extra functions:

1. A user can check the e-mail header prior to downloading.
2. A user can search the contents of the e-mail for a specific string of characters prior to downloading.
3. 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.
4. A user can create, delete, or rename mailboxes on the mail server.
5. A user can create a hierarchy of mailboxes in a folder for e-mail storage.

FILE TRANSFER PTOTOCOL (FTP)

File Transfer Protocol (FTP) is the standard mechanism provided by TCP/IP for copying a file from one host to another. Although transferring files from one system to another.

FTP differs from other client/server applications in that it establishes two connections between the hosts. One connection is used for data transfer, the other for control information (commands and responses).

The well-known port 21 is used for the control connection and the well-known port 20 for the data connection.

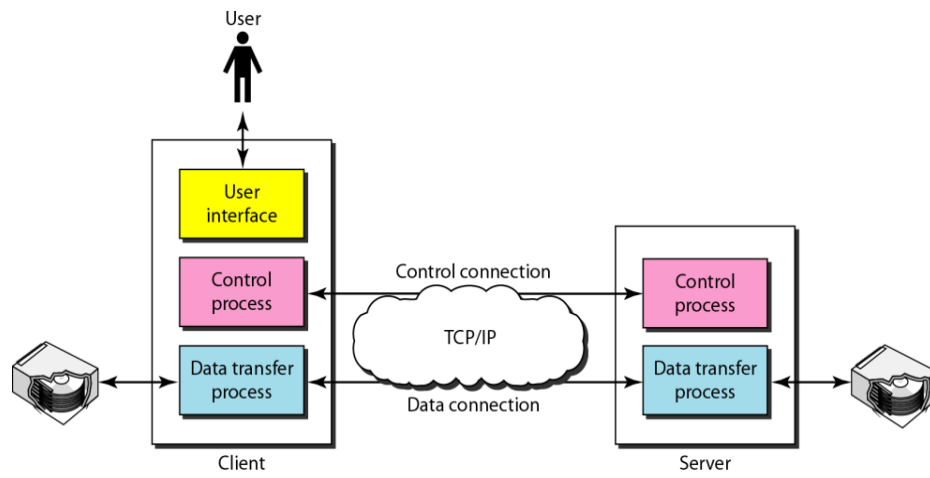


Figure 14

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.

We should remember that file transfer in FTP means one of three things:

- A file is to be copied from the server to the client. This is called *retrieving a file*. It is done under the supervision of the RETR command,
- A file is to be copied from the client to the server. This is called *storing a file*. It is done under the supervision of the STOR command.
- A list of directory or file names is to be sent from the server to the client. It is done under the supervision of the LIST command. Note that FTP treats a list of directory or file names as a file. It is sent over the data connection.

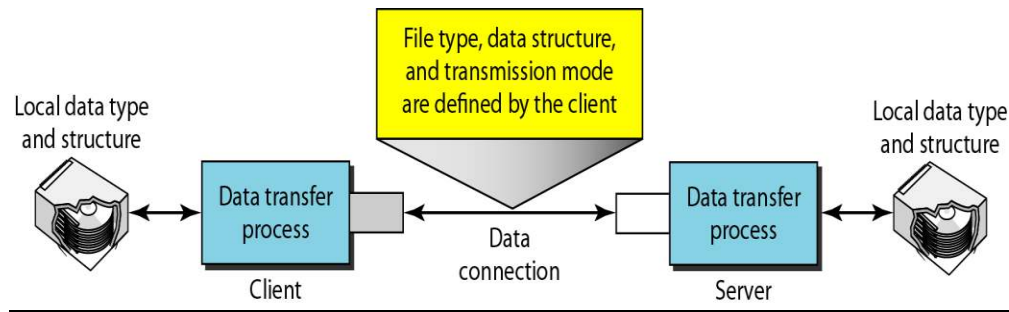


Figure 15

HYPER TEXT TRANSFER PROTOCOL (HTTP)

The Hypertext Transfer Protocol (HTTP) is a protocol used mainly to access data on the World Wide Web. HTTP functions as a combination of FTP and SMTP. It is similar to FTP because it transfers files and uses the services of TCP. However, it is much simpler than FTP because it uses only one TCP connection. There is no separate control connection; only data are transferred between the client and the server. HTTP is like SMTP because the data transferred between the client and the server look like SMTP messages.

HTTP uses the services of TCP on well-known port 80.

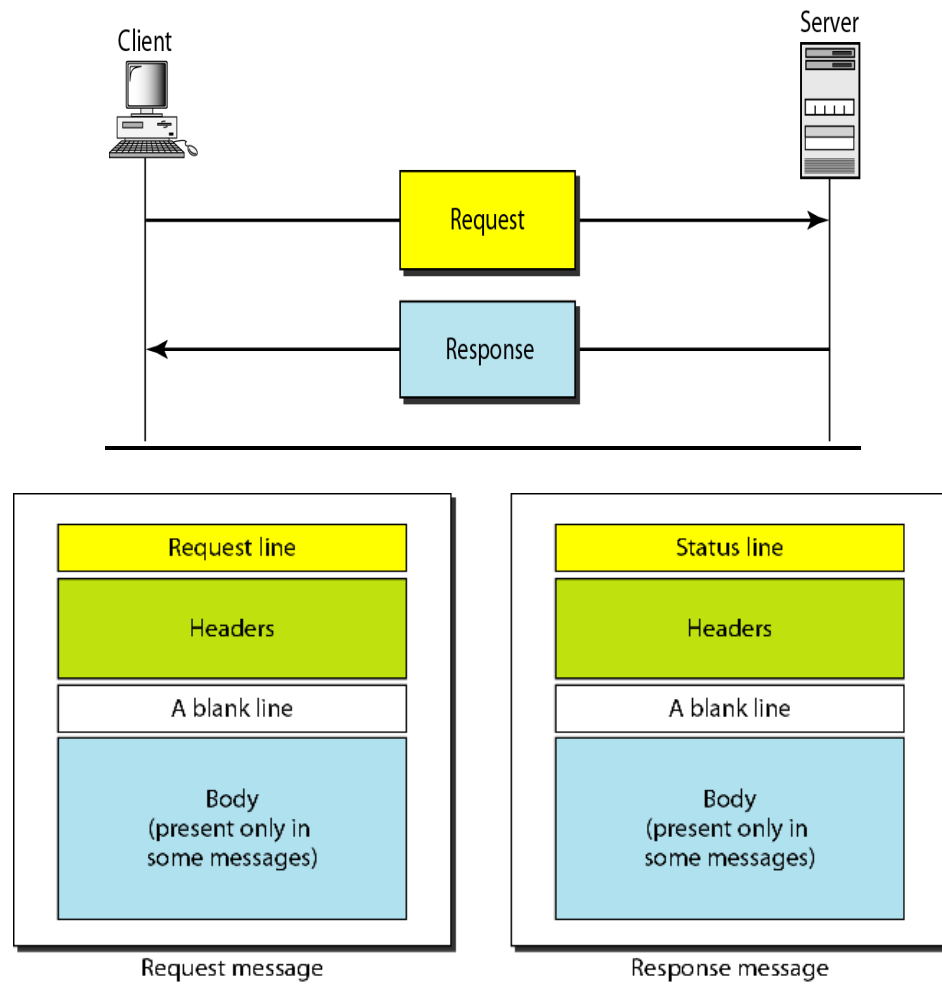


Figure 16

Request and Status Lines

The first line in a request message is called a request line; the first line in the response message is called the status line.

Request type

This field is used in the request message. In version 1.1 of HTTP, several request types are defined. The request type is categorized into *methods* as follows

<i>Method</i>	<i>Action</i>
GET	Requests a document from the server
HEAD	Requests information about a document but not the document itself
POST	Sends some information from the client to the server
PUT	Sends a document from the server to the client
TRACE	Echoes the incoming request
CONNECT	Reserved
OPTION	Inquires about available options

Figure 17

Version

The most current version of HTTP is 1.1.

Status code

This field is used in the response message. The status code field is similar to those in the FTP and the SMTP protocols. It consists of three digits.

Whereas the codes in the 100 range are only informational, the codes in the 200 range indicate a successful request. The codes in the 300 range redirect the client to another URL, and the codes in the 400 range indicate an error at the client site.

Header

The header exchanges additional information between the client and the server.

For example, the client can request that the document be sent in a special format, or the server can send extra information about the document. The header can consist of one or more header lines. Each header line has a header name, a colon, a space, and a header value.

Nonpersistent Connection

In a nonpersistent connection, one TCP connection is made for each request/response.

The following lists the steps in this strategy:

- The client opens a TCP connection and sends a request.
- The server sends the response and closes the connection.
- The client reads the data until it encounters an end-of-file marker; it then closes the connection.

In this strategy, for N different pictures in different files, the connection must be opened and closed N times. The nonpersistent strategy imposes high overhead on the server because the server needs N different buffers and requires a slow start procedure each time a connection is opened.

Persistent Connection

HTTP version 1.1 specifies a persistent connection by default. In a persistent connection, the server leaves the connection open for more requests after sending a response. The server can close the connection at the request of a client or if a time-out has been reached. The sender usually sends the length of the data with each response. However, there are some occasions when the sender does not know the length of the data. This is the case when a document is created dynamically or actively. In these cases, the server informs the client that the length is not known and closes the connection after sending the data so the client knows that the end of the data has been reached.

Note: HTTP version 1.1 specifies a persistent connection by default.

NETWORK MANAGEMENT

We can define network management as monitoring, testing, configuring, and troubleshooting network components to meet a set of requirements defined by an organization. These requirements include the smooth, efficient operation of the network that provides the predefined quality of service for users.

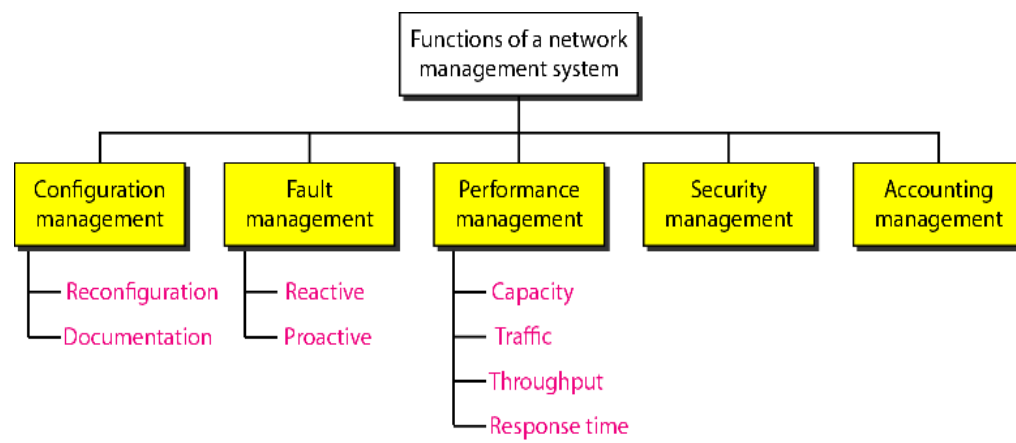


Figure 18

SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP)

The Simple Network Management Protocol (SNMP) is a framework for managing devices in an internet using the TCP/IP protocol suite. It provides a set of fundamental operations for monitoring and maintaining an internet.

SNMP uses the concept of manager and agent. That is, a manager, usually a host, controls and monitors a set of agents, usually routers.

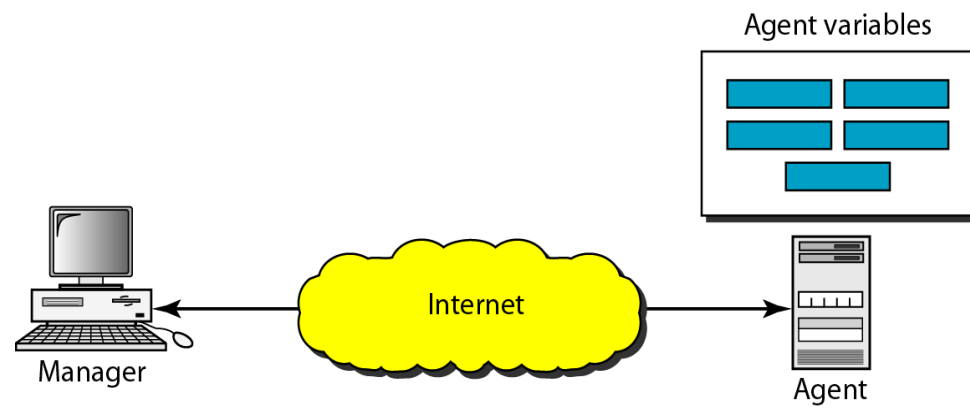


Figure 19

SNMP is an application-level protocol in which a few manager stations control a set of agents. The protocol is designed at the application level so that it can monitor devices made by different manufacturers and installed on different physical networks

Managers and Agents

A management station, called a manager, is a host that runs the SNMP client program. A managed station, called an agent, is a router (or a host) that runs the SNMP server program. Management is achieved through simple interaction between a manager and an agent. The agent keeps performance information in a database. The manager has access to the values in the database. For example, a router can store in appropriate variables the number of packets received and forwarded. The manager can fetch and compare the values of these two variables to see if the router is congested or not. The manager can also make the router perform certain actions.