...4...
Application Layer Protocols

Learning Outcomes...

- Explain functioning of DNS in internet.
- Explain the components of DNS Architecture.
- Explain the working of Message Transfer Agent.
- Explain the working of Message Access Agent.
- Explain the steps to transfer files using FTP.
- Describe the steps to access remote machine using command line and GUI tool.
- Explain the working of HTTP.
- Explain functions of PGP and Algorithms.

4.0 INTRODUCTION

- Application Layer is the 7th layer of OSI model and responsible for providing services to the user. It provides services that directly support user application such as database access, e-mail, file transfer etc.
- Application layer protocols specify the format and control information necessary for many of the common Internet communication functions.
- Application layer protocols are used to exchange data between the source and destination hosts/devices.
- Application layer uses protocols such as File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Simple Mail Transfer Protocol (SMTP), Internet Message Access Protocol (IMAP), Bootstrap Protocol (BOOTP), Simple Network Management Protocol (SNMP), Border Gateway Protocol (BGP), Hypertext Transfer Protocol (HTTP), TELNET, Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS) etc.
- The application layer in TCP/IP is equivalent to the combined session, presentation and application layers in the OSI model. The application layer allows a user to access the services of the private internet or the global Internet.
- Number of protocols are defined at application layer to provide services such as electronic mail, file transfer, accessing the World Wide Web (WWW) and so on.

4.1 DOMAIN NAME SYSTEM (DNS) [S-23, W-23, S-24, W-24]

- DNS stands for Domain Name System or Domain Network Service. DNS is a distributed Internet directory service.
- DNS is a hierarchical and decentralized naming system for computers, services, or other resources connected to the Internet.

Purpose of DNS:

- To identify an entity, TCP/IP protocols use the IP address. An IP address is uniquely identifying 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.
- When the Internet was small, mapping was done using a host file, which contains only two columns name and address.
- Each and every host could store the host file on its disk and update it periodically from a master host file. When a program or a user wanted to map a name to an address, the host consulted the host file and found the mapping.
- Today, however, it is impossible to have one single host file to relate every address with a name and vice versa. The two solutions are given below:
- To store the entire host file in a single computer and allow access to this centralized information to every computer that needs mapping. But we know that this would create a huge amount of traffic on the Internet.
- Another solution, the one used today, is to divide this huge amount of information into smaller parts and store each part on a different computer.
- In this method, the host that needs mapping can contact the closest computer holding the needed information. This method is used by the Domain Name System (DNS).
- Fig. 4.1 shows how TCP/IP uses a DNS client and a DNS server to map a name to an address; the reverse mapping is similar.

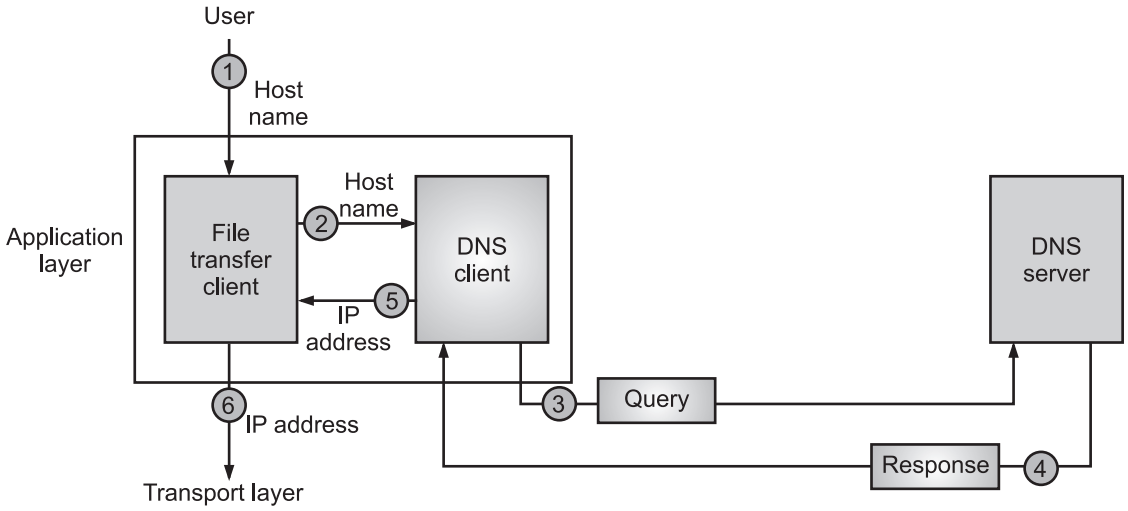

Fig. 4.1: Purpose of DNS

- In Fig. 4.1, a user wants to use a file transfer client to access the corresponding file transfer server running on a remote host.
- The user knows only the file transfer server name, such as forouzan.com. However, the TCP/IP suite needs the IP address of the file transfer server to make the connection.
- The following six steps map the host name to an IP address:

  Step 1: The user passes the host name to the file transfer client.

  Step 2: The file transfer client passes the host name to the DNS client.

Step 3: We know that each computer, after being booted, knows the address of one DNS server. 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.

Step 4: The DNS server responds with the IP address of the desired file transfer server.

Step 5: The DNS client passes the IP address to the file transfer server.

Step 6: The file transfer client now uses the received IP address to access the file transfer server.

- DNS is a network protocol used to translate hostnames into IP addresses.

4.1.1 Domain Name Space

- The domain name space consists of a tree data structure. In this section we study domain name space in detail.

Concept of Name Space:

- Name space is the abstract space or collection of all possible addresses, names, or identifiers of objects on a network, internetwork or the Internet.
- To be unambiguous, the names assigned to machines must be carefully selected from a name space with complete control over the binding between the names and IP addresses.
- A name space that maps each address to a unique name can be organized in two ways namely, flat name space or hierarchical name space.

1. Flat Name Space:

- Fig. 4.2 shows flat name space. In a flat name space, a name is assigned to an address. A name in this space is a sequence of characters without structure.
- The names may or may not have a common section; if they do, it has no meaning.
- The main advantages of flat name space are the names were convenient and short while disadvantage of a flat name space is that it cannot be used in a large system such as the Internet because it must be centrally controlled to avoid ambiguity and duplication.

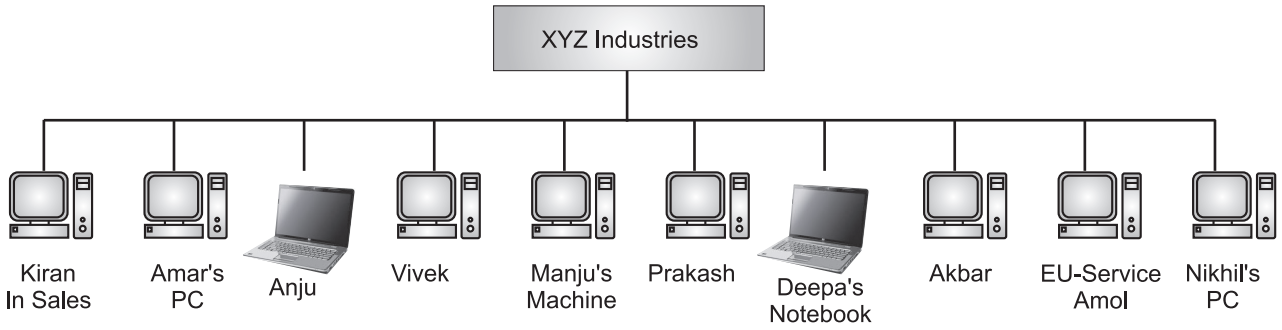

Fig. 4.2

2. Hierarchical Name Space:

- Fig. 4.3 shows hierarchical name space. name space. In a hierarchical name space, each name is made of several parts.
- The first part can define the nature of the organization, the second part can define the name of an organization, and the third part can define departments in the organization and so on.
- In hierarchical name space, the authority to assign and control the name spaces can be decentralized. Authority for names in each partition is passed to each designated agent.

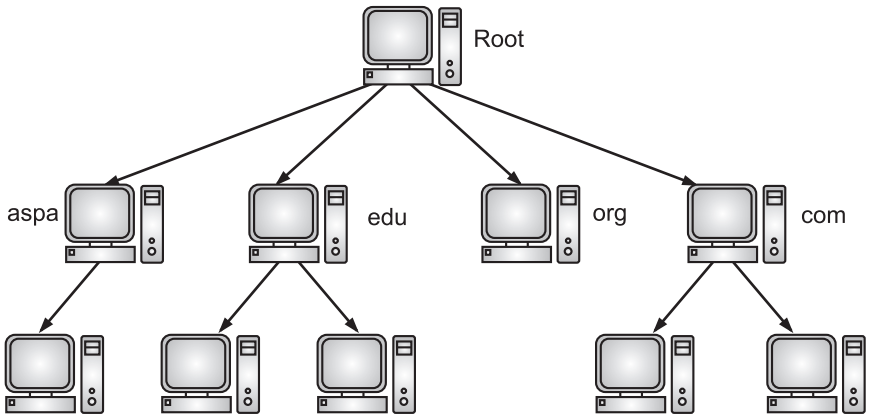

Fig. 4.3

Domain Name Space:

- 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 (See Fig. 4.4). The tree can have only 128 levels: level 0 (root) to level 127.
- In other words, the domain name space refers a hierarchy in the internet naming structure. This hierarchy has multiple levels (from 0 to 127), with a root at the top.

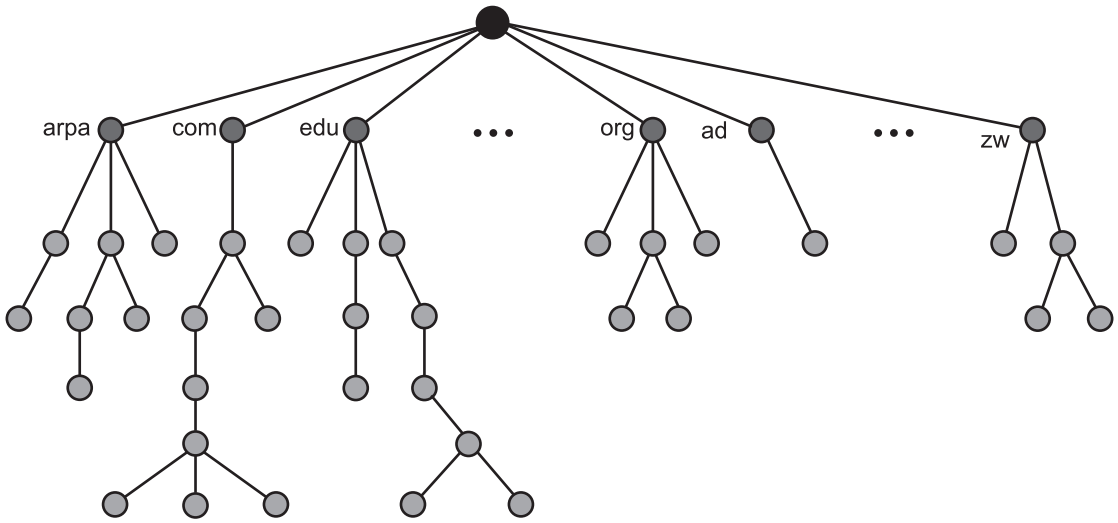

Fig. 4.4: Domain Name Space

Label:

- Each node in the tree, (domain name space) has a label, which is a string with a maximum of 63 characters. The root label is a null string, (empty string).
- DNS requires that children of a node, (nodes that branch from the same node) have different labels, which guarantees the uniqueness of the domain names.

Domain Name:

- Domain Name is a symbolic string associated with an IP address. Each node in the tree, (domain name space) 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). This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing.
- Fig. 4.5 shows some domain names and labels in DNS.

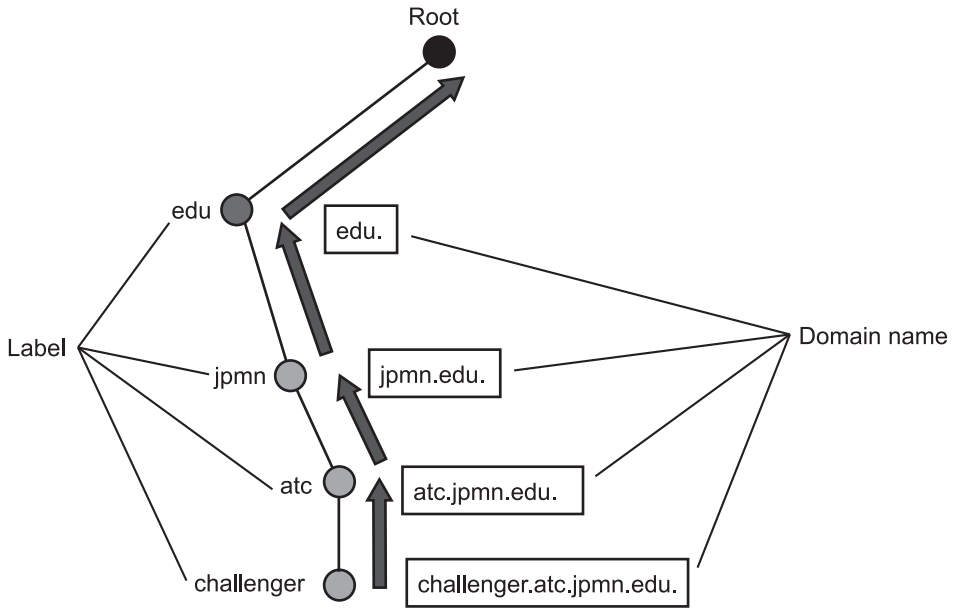

Fig. 4.5: Domain Names and Labels

Domain:

- A domain is a subtree of the domain name space. The name of the domain is the name of the node at the top of the subtree.
- Fig. 4.6 shows some domains. Note that a domain may itself be divided into domains (or subdomains as they are sometimes called).

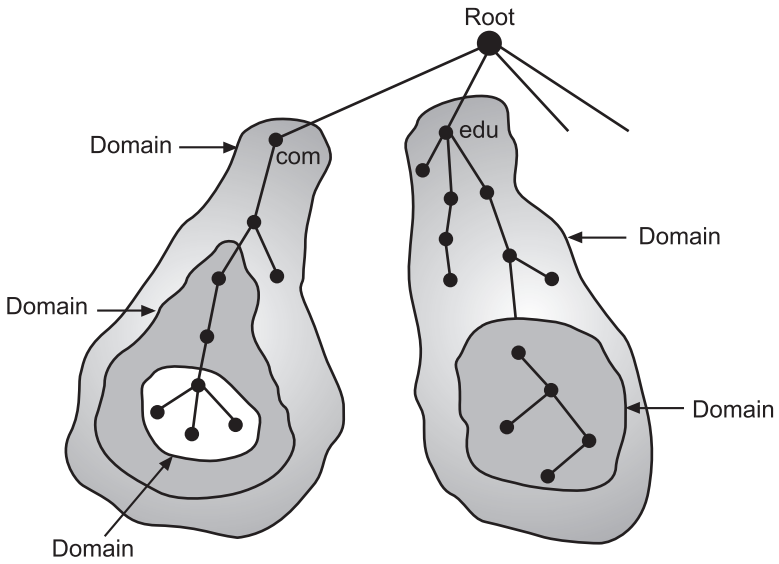

Fig. 4.6: Domains

Zones:

- Zone is collection of nodes (sub domains) under the main domain. The server maintains a database called zone file for every zone.
- Since the complete domain name hierarchy cannot be stored on a single server, it is divided among many servers. What a server is responsible for or has authority over is called a zone. If the domain is not further divided into sub domains, then domain and zone refer to the same thing.
- The information about the nodes in the sub domain is stored in the servers at the lower levels however; the original server keeps reference to these lower levels of servers.

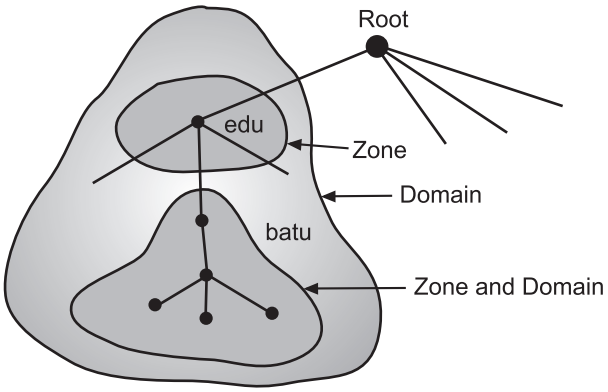

Fig. 4.7: Zones in Domains in DNS

Name Server:

- To distribute the information among many computers DNS uses DNS name servers.
- The Domain Name System is maintained by a distributed database system, which uses the client–server model. The nodes of this database are the name servers.
- Each domain has at least one authoritative DNS server that publishes information about that domain and the name servers of any domains subordinate to it.
- We let the root stand alone and create as many domains (subtrees) as there are first-level nodes. Because a domain created this way could be very large, DNS allows domains to be divided further into smaller domains (subdomains).
- Each and every server can be responsible (authoritative) for either a large or small domain.
- In other words, we have a hierarchy of servers in the same way that we have a hierarchy of names (See Fig. 4.8).

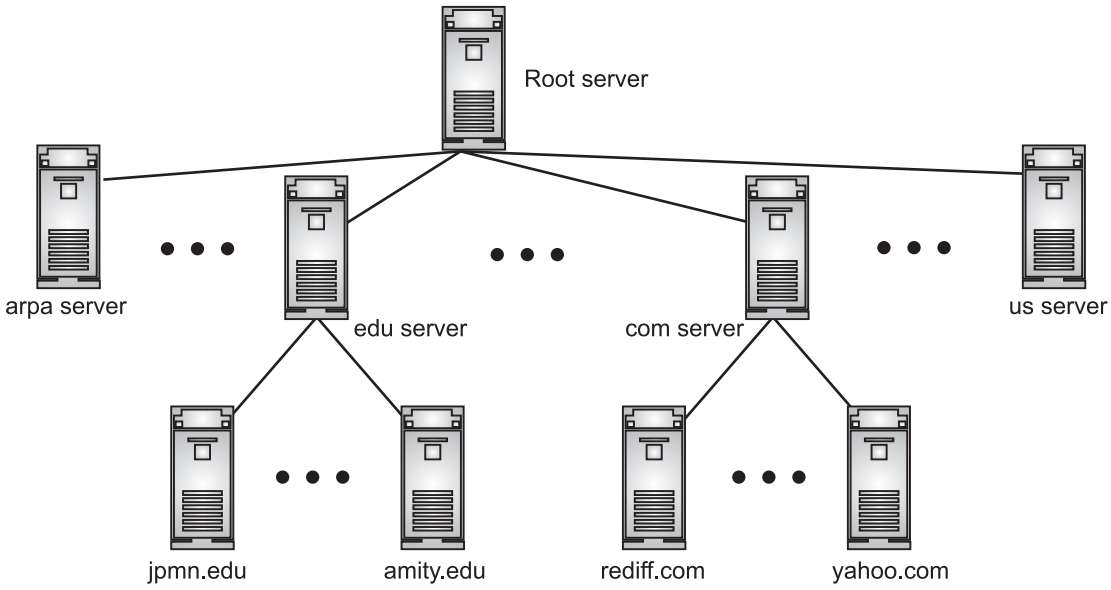

Fig. 4.8: Hierarchy of Name Servers

Types of Name Servers:

- Following are the three categories of Name Servers that manages the entire Domain Name System:
  1. **Root Server:** A root server usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers. There are several root servers, each covering the whole domain name space. The root servers are distributed all around the world.

2. **Primary Server:** A primary server is a server that stores a file about the zone for which it is an authority. It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.
3. **Secondary Server:** A secondary server is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk. The secondary server neither creates nor updates the zone files. If updating is required, it must be done by the primary server, which sends the updated version to the secondary.

- The primary and secondary servers are both authoritative for the zones they serve. The idea is not to put the secondary server at a lower level of authority but to create redundancy for the data so that if one server fails, the other can continue serving clients.

DNS in the Internet:

- In the Internet, the domain name space (tree) is divided into three different sections: generic domains, country domains, and the inverse domain as shown in Fig. 4.9.

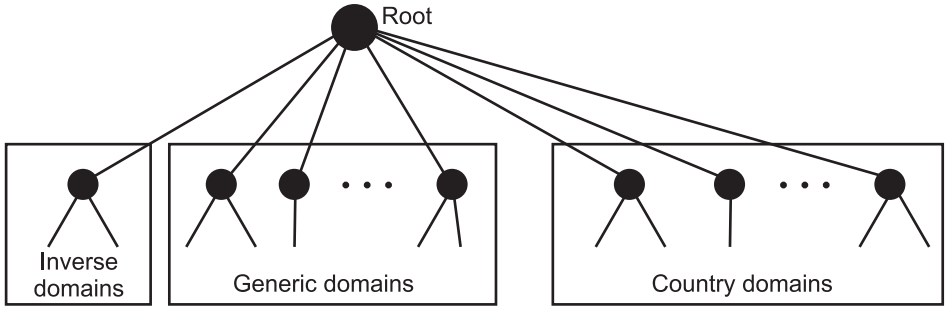

Fig. 4.9 DNS used in the Internet:

4.1.2 DNS Architecture

- DNS is a hierarchical system, with the root at the top and various levels of domains, subdomains and records below.
- The Internet root server manages top-level domains such as .com, .net, and .org at the root level. These top-level domains are responsible for managing their subdomains and records.
- The process of translating IP addresses to corresponding domain names through DNS is called name resolution or DNS resolution. DNS resolution begins with a client's DNS request.
- Fig. 4.10 shows how a client obtains the IP address for a web server via DNS resolution, allowing it to receive web services.
  1. A client requests an IP address www.google.com from a local recursive DNS resolver.
  2. The recursive DNS resolver first checks the address translation in its local cache.
  3. If there is no information in the cache, the recursive DNS resolver requests the IP address of the TLD nameserver from the Root name server.
  4. The Root name server sends back the IP address of the .com name server as a response.
  5. Using this IP address, the recursive DNS Resolver requests the IP address of the SLD nameserver from the .com name server.
  6. The .com name server sends back the IP address of the .google.com name server as a response.
  7. With the IP address, the recursive DNS Resolver requests the IP address for www.google.com from the .google.com name server.
  8. The .google.com name server sends back the own IP address of www.google.com to the recursive DNS resolver.
  9. The recursive DNS resolver sends back the IP address of www.google.com to the client as a response. Finally, with the IP address (172.217.7.197 in this example), the client connects to the www.google.com server.

- The DNS framework consists of the following three parts/components:
  1. **Client:** They request IP addresses with domain names through the stub resolver, a client of DNS, and transmits the request to the local DNS server address set on its device.
  2. **Local DNS Server (Recursive DNS Resolver):** They receive the DNS query from clients and obtains the IP address for the domain name from domain name servers. Also, the IP address once found is stored in memory for a certain period. So, it is called Caching Resolver.
  3. **Domain Name Server (Authoritative Name Server):** They have and manage IP addresses for the domain names as well as the information related to the IP addresses. The Authoritative Name Server is composed of more than 3-levels (Root, TLD, Lower-level Domain). Each domain server consists of a single master server and several slave servers.

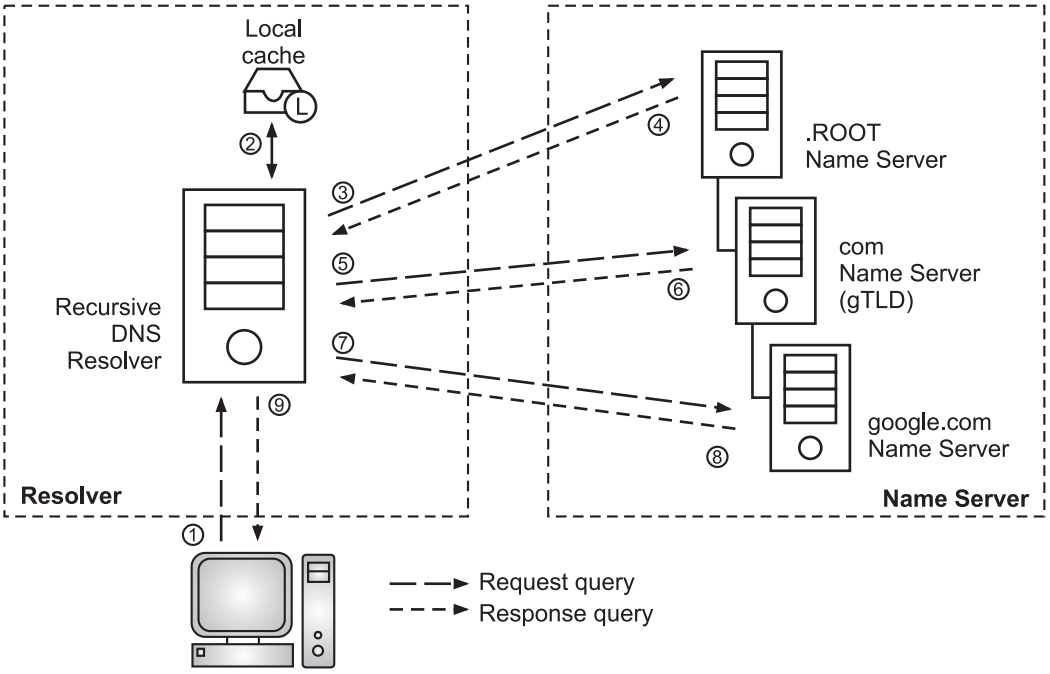

Fig. 4.10: DNS Architecture (DNS, gTLD (general Top Level Domain))

Generic Domains/Types of Domain:

- 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.

Table 4.1: Example of Generic Domains

Sr. No.	Label	Description
1.	com	Commercial organization, such as Hewlett-Packard (hp.com), Sun Microsystems (sun.com), and IBM (ibm.com).
2.	edu	Educational institute, such as U.C. Berkeley (berkeley.edu) and Purdue University (purdue.edu).
3.	gov	Government institute, such as NASA (nasa.gov) and the National Science Foundation (nsf.gov).
4.	int	International Organization, such as NATO (nato.int).
5.	mil	Military groups, such as the U.S. Army (army.mil) and Navy (navy.mil).
6.	net	Network support engineers, such as NSFNET (nsf.net).
7.	org	Nonprofit organization, such as the Electronic Frontier Foundation (eff.org).

- Fig. 4.11 shows generic domains, used by DNS for Internet.

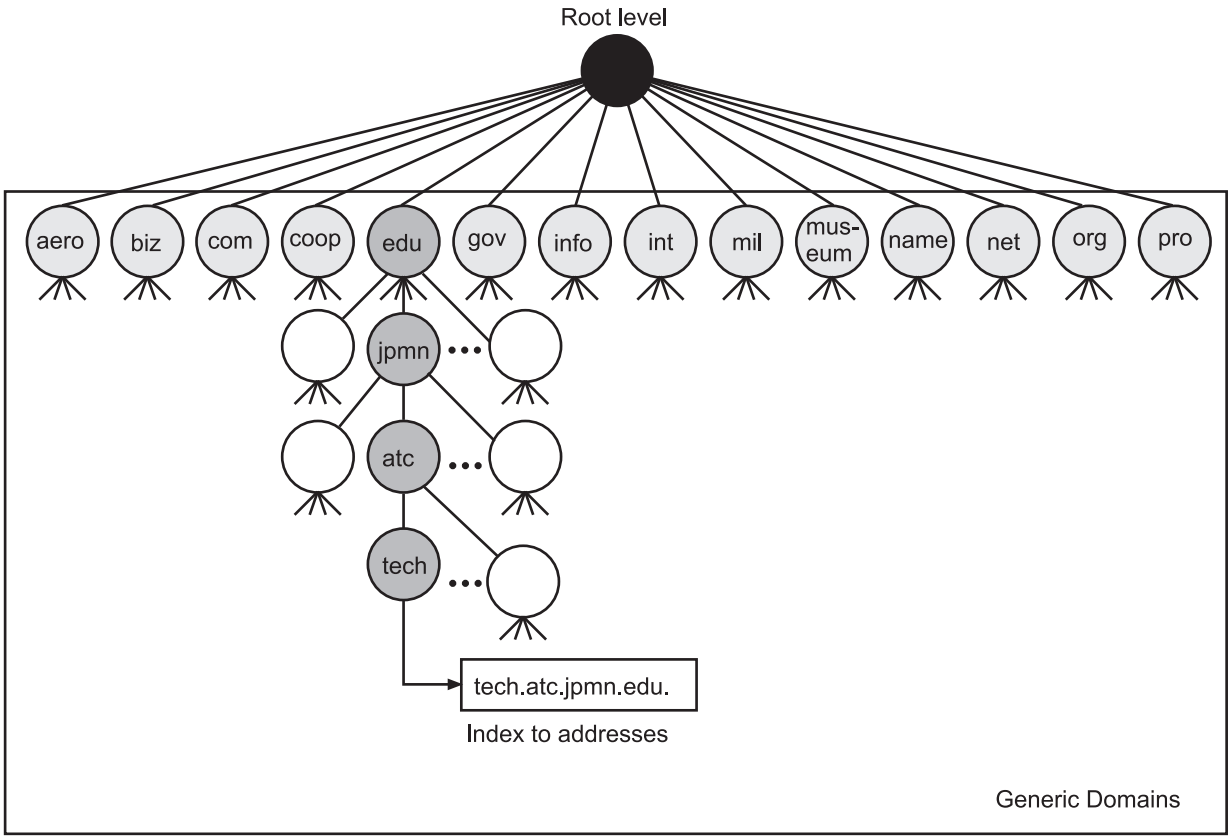

Fig. 4.11

Country Domains:

- The country domain section follows the same format as the generic domains but uses two-character country abbreviations in place of three-character organizational abbreviations at the first level.

Table 4.2: Example of country domains

Sr. No.	Label	Description
1.	au	Australia
2.	ca	Canada
3.	in	India
4.	uk	United Kingdom
5.	fr	France
6.	th	Thailand
7.	us	United States
8.	zw	Zimbabwe

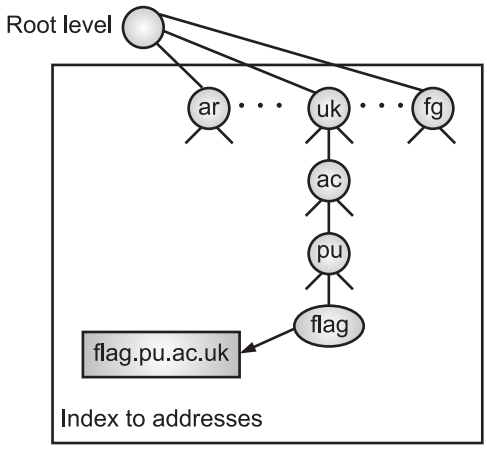

Fig. 4.12

- Fig. 4.12 shows country domains, used by DNS for Internet.

Inverse Domains:

- Inverse domain is used to map an address to a name. For example, a client sends a request to the server for performing a particular task, server finds a list of authorized clients. The list contains only IP addresses of the client.

- The server sends a query to the DNS server to map an address to a name to determine if the client is on the authorized list. This query is called an inverse query. This query is handled by first level node called arpa.

Domain Levels in DNS:

- At the top of the hierarchical structure of the domain name system is the Root Domain which is represented by a dot (.).
- **Top Level Domain:** Below the root domain are situated Top Level Domain. This is the rightmost part of a domain name.

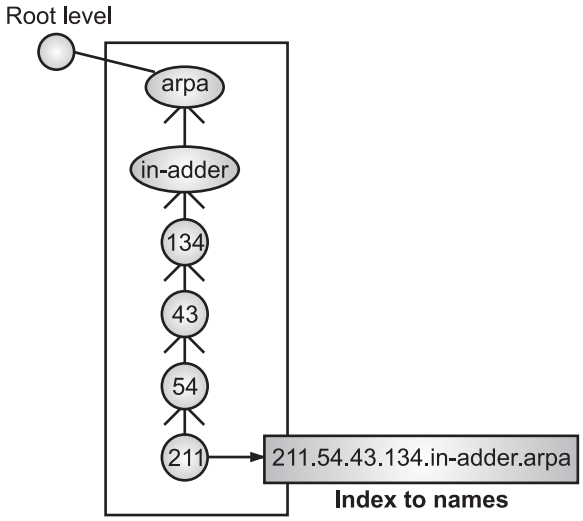

Fig. 4.13: Inverse Domain

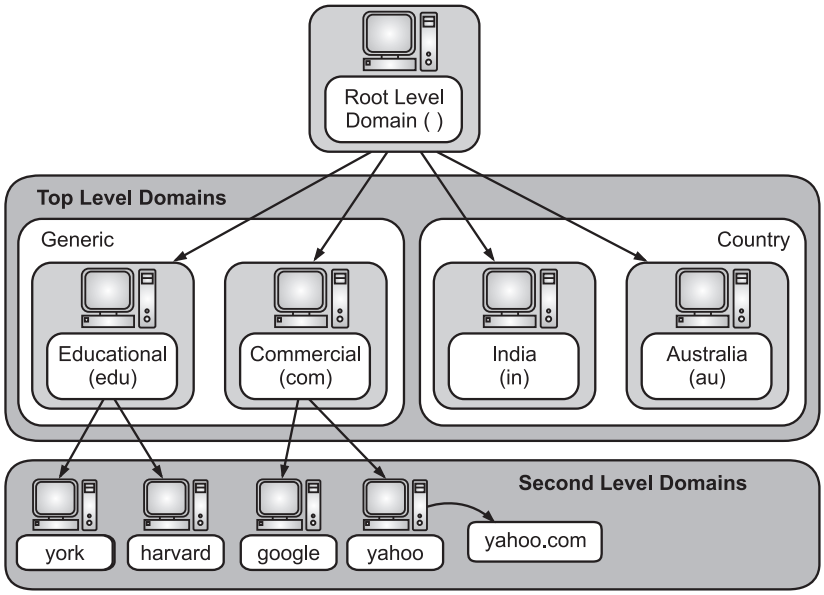

Fig. 4.14

- **Second Level Domain:** This is middle section of a domain name and indicates a unique name for an organization. Different organizations with same top-level domains are distinguished by their second level domain names. For example, google.com and ongc.com are both commercial organizations but differ from each other by their second level domain name Google.
- **Third Level Domain:** To identify closely related divisions of an organization another word called a third-level domain or a sub-domain may be added at the very beginning of a domain name. Example sales.ongc.com, www.yahoo.com etc.

4.1.3 Domain Name Resolution and Mapping to Physical Addresses [S-23]

- Domain name resolution is the process of converting a human-readable domain name like www.google.com into a machine-readable IP address, such as 142.250.190.4.
- Mapping a name to an address or an address to a name is called name-address resolution. DNS is designed as a client-server application.
- A host that needs to map an address to a name or a name to an address calls a DNS client called a resolver and it accesses the closest DNS server with a mapping request.

- If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.
- After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.

Mapping Names to Addresses:

- Number of the time, the resolver gives a domain name to the server and asks for the corresponding address. In this case, the server checks the generic domains or the country domains to find the mapping.
- If the domain name is from the generic domains section, the resolver receives a domain name such as "nirali.atc.fhda.edu.".
- The query is sent by the resolver to the local DNS server for resolution. If the local server cannot resolve the query, it either refers the resolver to other servers or asks other servers directly.
- If the domain name is from the country domains section, the resolver receives a domain name such as "ni.fhda.cu.ca.in.". The procedure is the same.

Mapping Addresses to Names:

- A client can send an IP address to a server to be mapped to a domain name, this is called a PTR query. To answer queries of this kind, DNS uses the inverse domain.
- However, in the request, the IP address is reversed and two labels, in-addr and arpa, are appended to create a domain acceptable by the inverse domain section.
- For example, if the resolver receives the IP address 132.34.45.121, the resolver first inverts the address and then adds the two labels before sending. The domain name sent is "121.45.34.132.in-addr.arpa.", which is received by the local DNS and resolved.

Recursive Resolution in DNS:

- In the recursive resolution the client (resolver) can ask for a recursive answer from a name server. This means that the resolver expects the server to supply the final answer.
- If the server is the authority for the domain name, it checks its database and responds. If the server is not the authority, it sends the request to another server (the parent usually) and waits for the response.
- If the parent is the authority, it responds; otherwise, it sends the query to yet another server. When the query is finally resolved, the response travels back until it finally reaches the requesting client.

Iterative Resolution:

- If the client does not ask for a recursive answer, the mapping can be done iteratively. If the server is an authority for the name, it sends the answer.
- If it is not, it returns (to the client) the IP address of the server that it thinks can resolve the query. The client is responsible for repeating the query to this second server.
- If the newly addressed server can resolve the problem, it answers the query with the IP address; otherwise, it returns the IP address of a new server to the client.
- Now the client must repeat the query to the third server. This process is called iterative because the client repeats the same query to multiple servers.

Caching:

- Each and every time a server receives a query for a name that is not in its domain, it needs to search its database for a server IP address.
- Reduction of this search time would increase efficiency. DNS handles this with a mechanism called caching.
- When a server asks for a mapping from another server and receives the response, it stores this information in its cache memory before sending it to the client.

- If the same or another client asks for the same mapping, it can check its cache memory and resolve the problem.
- However, to inform the client that the response is coming from the cache memory and not from an authoritative source, the server marks the response as unauthoritative.
- Fig. 4.15 shows operation of DNS.

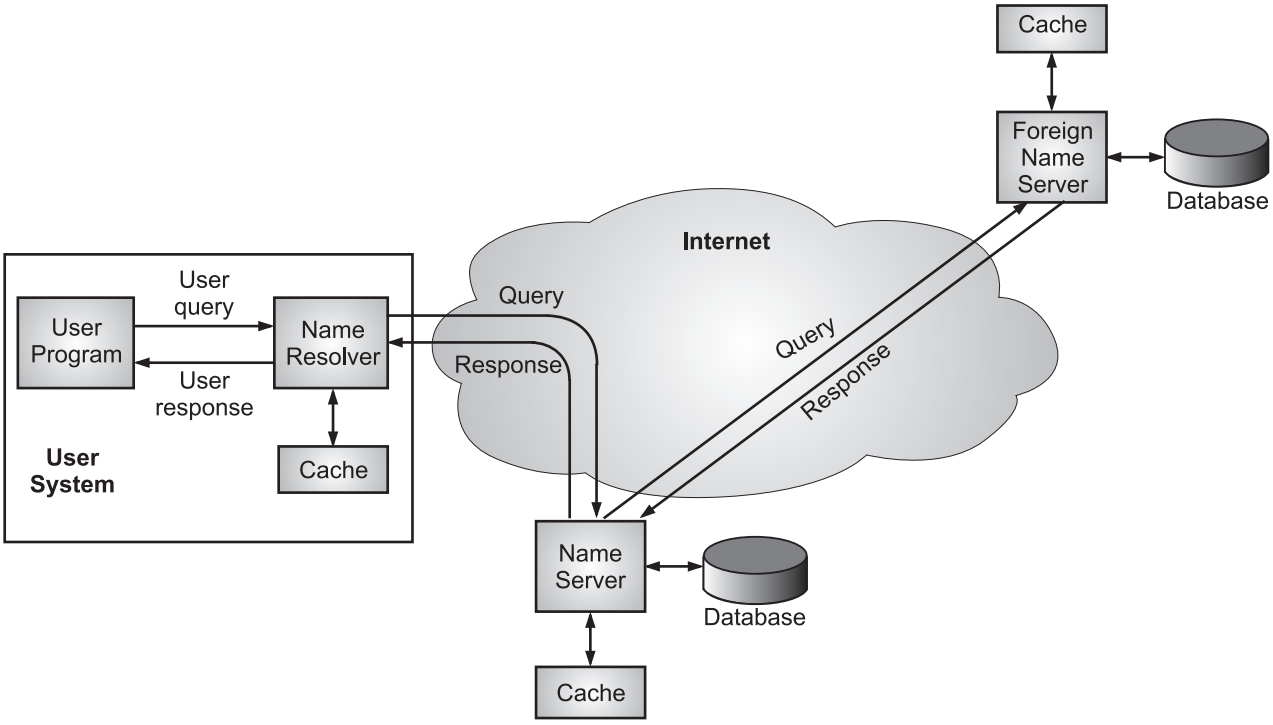

Fig. 4.15: Operation of DNS

- Steps in DNS operation are given below:

  Step 1: A user program requests an IP address for a domain name.

  Step 2: A resolver module in the local host or local ISP formulates a query for a local name server in the same domain as the resolver.

  Step 3: The local name server checks to see if the name is in its local database or cache, and, if so, returns the IP address to the requestor. Otherwise, the name server queries other available name servers, starting down from the root of the DNS tree or as high up the tree as possible.

  Step 4: When a response is received at the local name server, it stores the name/address mapping in its local cache and may maintain this entry for the amount of time specified in the time to live field of the retrieved RR, (Resource Records).

  Step 5: The user program is given the IP address or an error message.

- Typically, single queries are carried over UDP. Queries for a group of names are carried over TCP. The distributed DNS database that supports the DNS functionality must be updated frequently because of the rapid and continued growth of the Internet.

Example:

- A Domain Name System (DNS) server performs the task of resolving a domain name to an IP address. As shown in Fig. 4.16, an end user who wants to navigate to the www.niralipress.com website enters that fully qualified domain name (FQDN) into the web browser.
- The end user's computer sends a DNS request to the DNS server asking for the IP address that corresponds to www.niralipress.com.
- The DNS server responds with 192.168.1.11, and now the end user's computer can send packets to the destination.

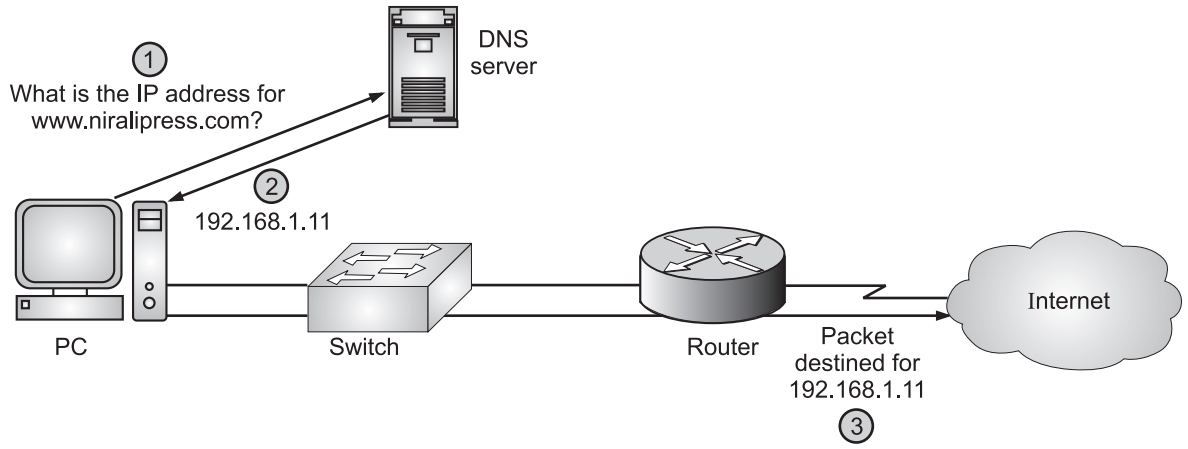

Fig. 4.16: DNS Operation

- A DNS server maintains a database of local FQDNs and their corresponding IP addresses, in addition to pointers to other servers that can resolve IP addresses for other domains.

Mapping Domain Names to Physical Addresses:

- While domain name resolution converts a domain name into an IP address, mapping it to a physical address involves additional steps:

1. **IP Address as Network Identifier:**
   - The resolved IP address identifies a specific device or server on a network.
   - For example, in IPv4, 192.168.1.1 could represent a specific web server.
2. **Address Resolution Protocol (ARP):**
   - Within local networks, ARP is used to map an IP address to a physical (MAC) address of a device.
   - This process ensures that data packets are delivered to the correct hardware on a network.
3. **Routing Across Networks:**
   - Once an IP address is resolved, routers use it to forward data packets across networks until they reach their destination.
   - This involves translating between logical addresses (IP) and physical devices (MAC).

4.2 ELECTRONIC MAIL

- Electronic mail (or e-mail) was the first Internet application or service and is still the most popular one. E-mail is a way of sending messages between people or computers through networks of computer connections.
- To send and receive a mail two agents, Message Transfer Agent (MTA) and a Message Access Agent (MAA) are required.
- E-mail on the Internet is analogous to the regular postal system but faster, easy to distribute, and inexpensive.in delivery of mail. E-mail is a way to send and receive messages across the Internet/Web/WWW.
- E-mail is defined as, "the transmission of messages over communication networks". E-mail is a service which allows us to send the message in electronic mode over the Internet.
- As with ordinary postal mail, e-mail is an asynchronous communication medium in which people send and read messages when it is convenient for them, without having to coordinate with other people's schedules.
- Modern e-mail has many powerful features, including messages with attachments, hyperlinks, HTML-formatted text, and embedded photos.

- E-mail has three major components namely, user agents, mail servers and Simple Mail Transfer Protocol (SMTP).

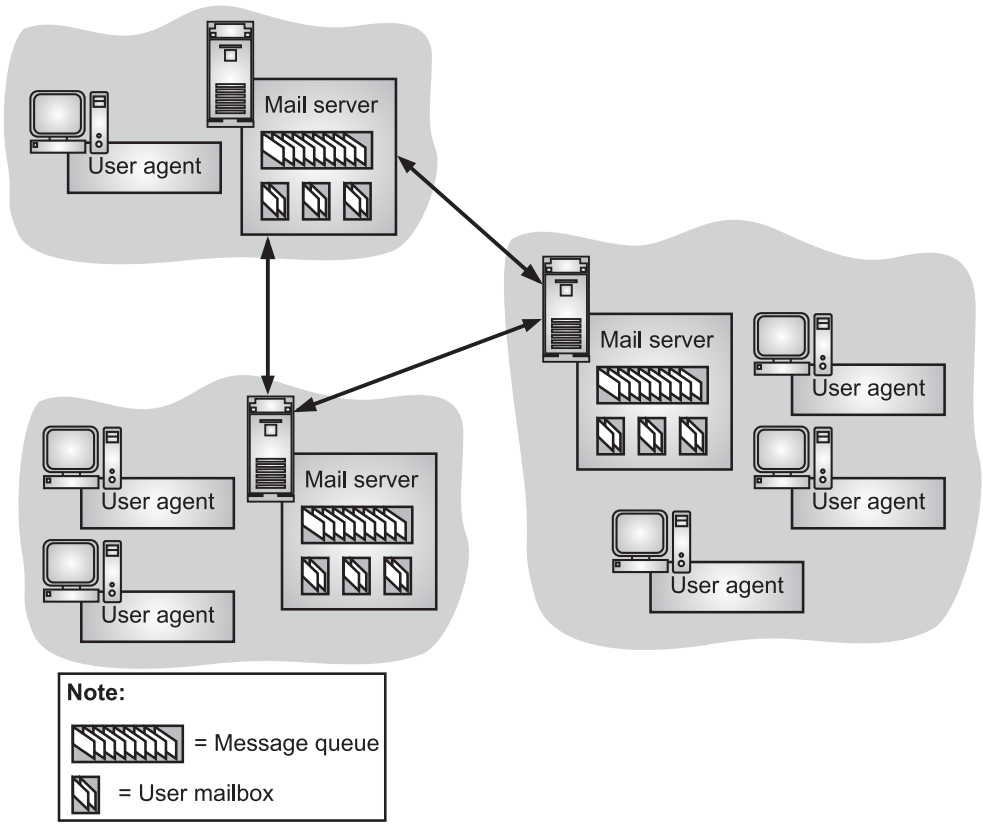

Fig. 4.17: High Level View of the Internet E-Mail System

Functions of E-Mail:

- E-mail system support following five basic functions:
  1. **Composition:** Composition refers to the process of creating messages and answer. Any text editor can be used for the body of the message. When answering a message, the e-mail system can extract the originator's address form the incoming e-mail.
  2. **Transfer:** Transfer refers to moving messages from the originator to the recipient.
  3. **Reporting:** Reporting has to do with telling the originator what happened to the message. Whether, email is delivered or not delivered.
  4. **Displaying:** Displaying incoming messages is needed, so user can read their e-mail.
  5. **Disposition:** Disposition is the final step and concerns what the recipient does with the message after receiving it. It may be read and save or delete or forward the message.
- **Example:** Fig. 4.18 shows an example of e-mail.
- In the Fig. 4.18, when Anju is finished composing her message, her user agent sends the message to her mail server, where the message is placed in the mail servers' outgoing message queue.
- When Anil wants to read a message, his user agent retrieves the message from his mailbox in his mail server. Mail servers form the core of the e-mail infrastructure.
- Each recipient, such as Anil, has a mailbox located in one of the mail servers. Anil's mailbox manages and maintains the messages that have been sent to him.
- A typical message starts its journey in the sender's user agent, travels to the sender's mail server and travels to the recipient's mail server, where it is deposited in the recipient's mailbox.

- When Anil wants to access the messages in his mailbox, the mail server containing his mailbox authenticates Anil, (with usernames and passwords).

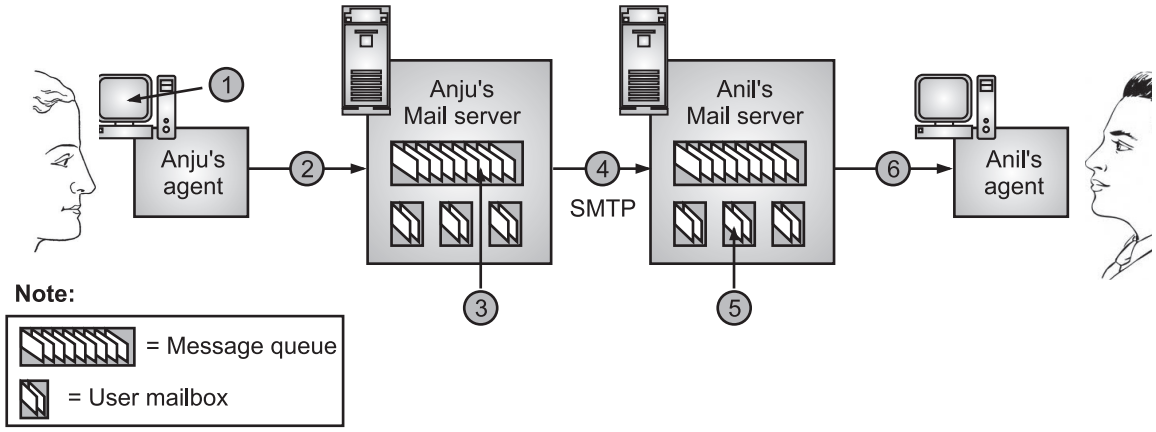

Fig. 4.18: Anju Sends a Message to Anil

- Anju's mail server must also deal with failures in Anil's mail server. If Anju's server cannot deliver mail to Anil's server, Anju's server holds the message in a message queue and attempts to transfer the message later.
- Reattempts are often done every 30 minutes or so; if there is no success after several days, the server removes the message and notifies the sender (Anju) with an e-mail message.
- SMTP is the principal application layer protocol for Internet electronic mail. It uses the reliable data transfer service of TCP to transfer mail from the sender's mail server to the recipient's mail server.
- SMTP has namely, a client side, which executes on the sender's mail server, and a server side, which executes on the recipient's mail server. Both the client and server sides of SMTP run on every mail server.
- When a mail server sends mail to other mail servers, it acts as an SMTP client. When a mail server receives mail from other mail servers, it acts as an SMTP server.

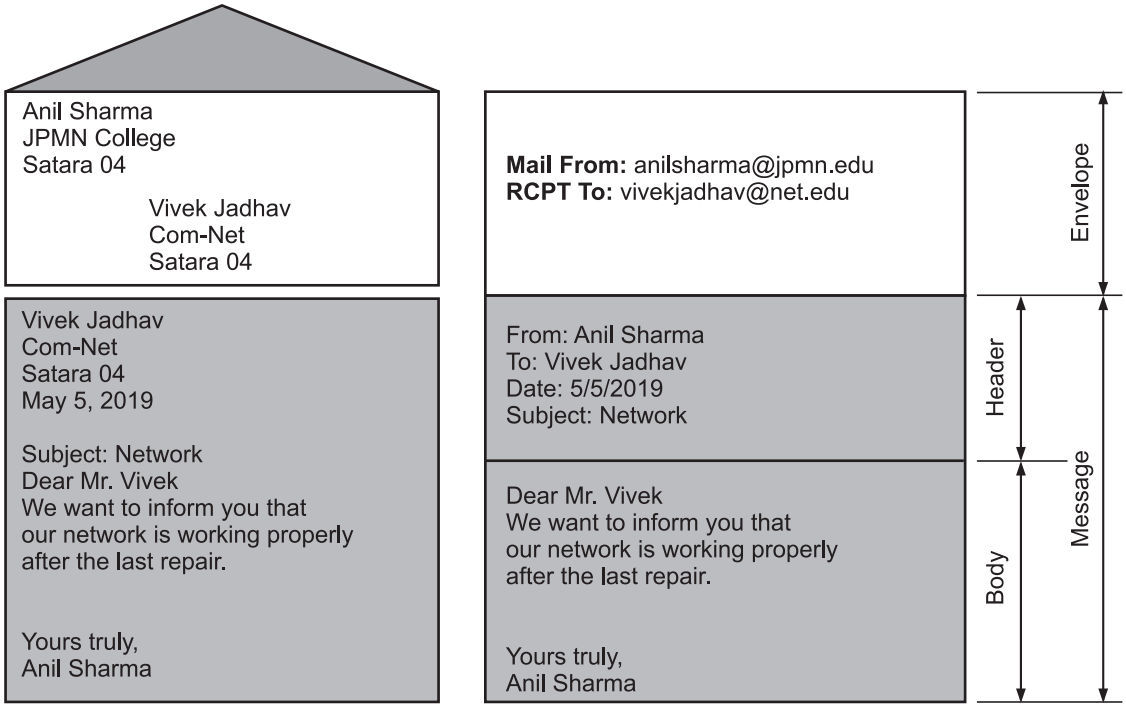

Fig. 4.19: Format of an E-mail

4.2.1 Architecture of E-Mail

- One of the most popular Internet services is e-mail. The designers of the Internet probably never imagined the popularity of this application program.

1. First Scenario:

- In the first scenario, the sender and the receiver of the e-mail are users (or application programs) on the same system; they are directly connected to a shared system.
- The administrator has created one mailbox for each user where the received messages are stored. A mailbox is part of a local hard drive, a special file with permission restrictions.
- Only the owner of the mailbox has access to it. When Anju, a user, needs to send a message to Anil, another user, Anju runs a User Agent (UA) program to prepare the message and store it in Anil's mailbox.
- The message has the sender and recipient mailbox addresses (names of files). Anil can retrieve his convenience, using a user agent.
- Fig. 4.20 shows the concept. This is similar to the traditional memo exchange between employees in an office. There is a mailroom where each employee has a mailbox with his or her name on it.
- When Anju needs to send a memo to Anil, she writes the memo and inserts it into Anil's mailbox. When Anil checks his mailbox, he finds Anju's memo and reads it.

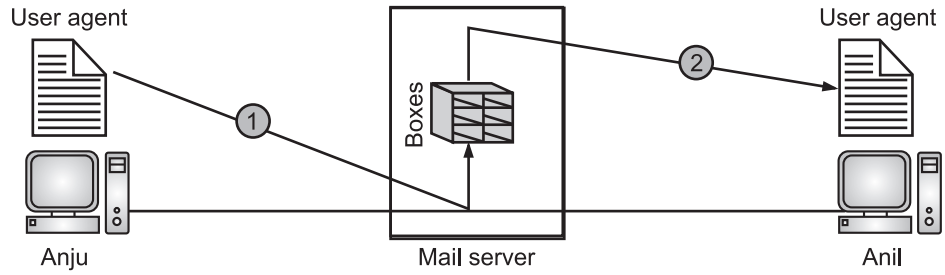

Fig. 4.20: First Scenario

2. Second Scenario:

- In the second scenario, the sender and the receiver of the e-mail are users (or application programs) on two different systems.
- The message needs to be sent over the Internet. Here, we need User Agents (UAs) and Message Transfer Agents (MTAs), as shown in Fig. 4.21.

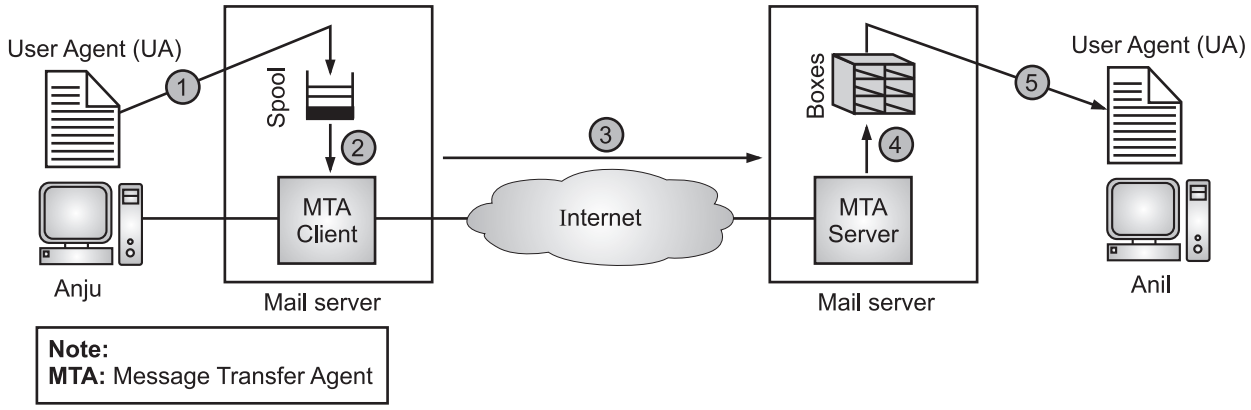

Fig. 4.21: Second Scenario

- Anju needs to use a user agent program to send her message to the system at her own site. The system (sometimes called the mail server) at her site uses a queue to store messages waiting to be sent.

- Anil also needs a user agent program to retrieve messages stored in the mailbox of the system at his site.
- The message, however, needs to be sent through the Internet from Anju's site to Anil's site. Here, two message transfer agents are needed namely, one client and one server.
- Like most client/server programs on the Internet, the server needs to run all the time because it does not know when a client will ask for a connection.
- On the other hand, the client can be alerted by the system when there is a message in the queue to be sent.

3. Third Scenario:

- In the third scenario, Anil as in the second scenario, is directly connected to his system. Anju, however, is separated from her system.
- Either Anju is connected to the system via a point-to-point WAN, such as a dial-up modem, a DSL, or a cable modem; or she is connected to a LAN in an organization that uses one mail server for handling e-mails-all users need to send their messages to this mail server.
- Fig. 4.22 shows the situation.

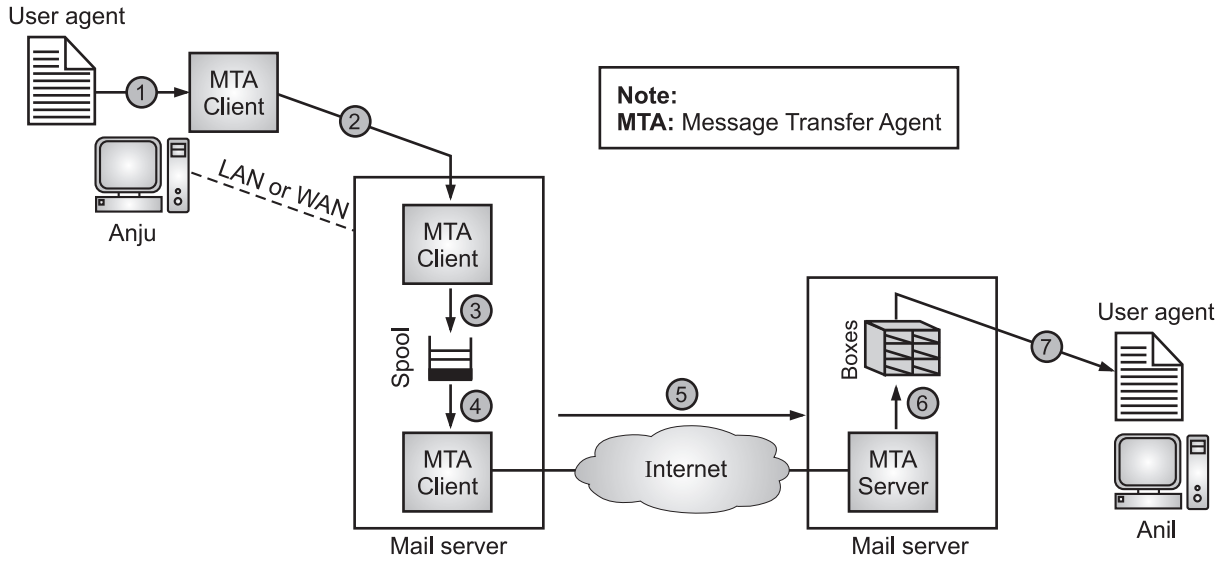

Fig. 4.22: Third Scenario

- Anju still needs a user agent to prepare her message. She then needs to send the message through the LAN or WAN.
- This can be done through a pair of message transfer agents (client and server). Whenever, Anju has a message to send, she calls the user agent which, in tum, calls the MTA client.
- The MTA client establishes a connection with the MTA server on the system, which is running all the time. The system at Anju's site queues all messages received.
- It then uses an MTA client to send the messages to the system at Anil's site; the system receives the message and stores it in Anil's mailbox.
- At his convenience, Anil uses his user agent to retrieve the message and reads it. Note that we need two pairs of MTA client/server programs.

4. Fourth Scenario:

- In the fourth and most common scenario, Anil is also connected to his mail server by a WAN or a LAN.
- After the message has arrived at Anil's mail server, Anil needs to retrieve it. Here, we need another set of client/server agents, which we call Message Access Agents (MAAs).

- Anil 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. The situation is shown in Fig. 4.23.

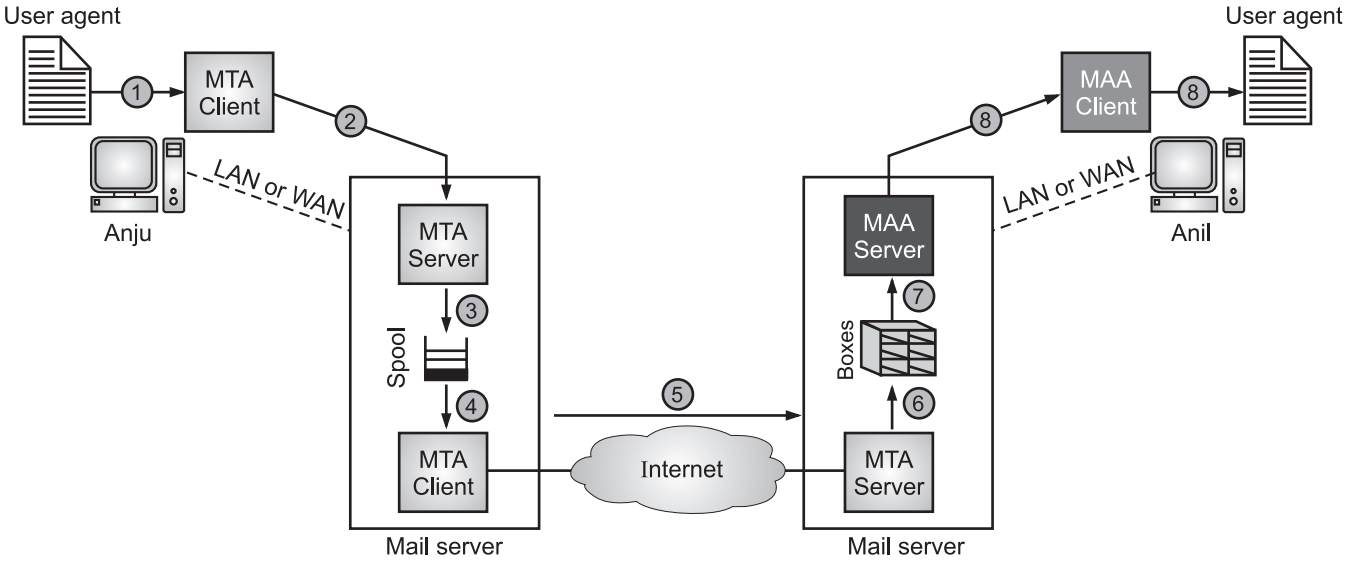

Fig. 4.23: Fourth Scenario

- There are two important points here. First, Anil cannot bypass the mail server and use the MTA server directly.
- To use MTA server directly, Anil would need to run the MTA server all the time because he does not know when a message will arrive.
- This implies that Anil must keep his computer on all the time if he is connected to his system through a LAN. If he is connected through a WAN, he must keep the connection up all the time. Neither of these situations is feasible today.
- Second, note that Anil needs another pair of client/server program: message access programs. This is so because an MTA client/server program is a push program: the client pushes the message to the server. Anil needs a pull program.
- The client needs to pull the message from the server. Fig. 4.24 shows the difference.

(a) Client Pushes Messages **(b) Client Pulls Messages**

Fig. 4.24: Push vs Pull in Electronic Mail

Advantages of using E-mail:

1. **Speed:** The recipient can receive an e-mail within a very short time of sending the same.
2. **Cost:** No extra cost involved in sending e-mail. It is the cheapest mode of sending a message.
3. **Content:** A message can consist of a few lines or several hundred lines of text. Any form of electronic data like text, graphics, audio, etc. can also be sent along with an e-mail.
4. **Delivery:** A single message can be delivered to a single or multiple recipients simultaneously.
5. **Reliability:** It is a reliable mode of communication, as in case a message cannot be sent, the sender gets a notification on the same within a short period of time.
6. **Security:** Unauthorized persons cannot access another person's e-mail. Only the user with his password can open his account or mailbox and view his mails.
7. **Access:** In case of a Web mail service. A user can check his email from any computer/mobile around the world, provided it is connected to the Internet.

Disadvantages of using E-mail:

1. **Information Overload:** People daily receive many mails, majority of which can be junk mails. One can have to go through hundreds of mails to find out which are useful and which are not. Thus, spamming can be unproductive.
2. **Security Risk:** E-mails can be a source of computer viruses. Viruses can come as attachments to mails from unknown sources. When such mails are opened, the computers get infected by the virus. Using antivirus programs and firewalls can prevent such infection.
3. **E-mail Bombing:** It is the intentional sending of a large number of e-mails to a particular target address. This may cause overloading and can finally lead to a server crash.
4. **E-mail Spoofing:** This occurs when the e-mail header is altered to make a message appear to come from a reliable or known source. This is usually done to collect personal information of the recipient such as usernames, passwords, bank account and credit card details, which can then be misused. As stated earlier, the process is called Phishing (pronounced fishing!).
5. **Distraction:** Checking and replying to vast number of e-mails can hamper productivity.

4.2.2 Message Transfer Agent

- MTA (Message Transfer Agent) is a piece of software that is responsible for transferring emails between computers.
- MTA's are essentially mail servers that are known by quite a few different names. It is referred to as a mail transport agent, mail transfer agent, mail relay, mail router, or even internet mailer.
- Technically speaking, a MTA is a program used within a Message Handling System (MHS) for transmitting e-mails between the sender and receiver device or computer.
- The basic function of an MTA is the transmission of mail between users. Here the main support for an MTA is the exchange system with the server architecture that aids the transfer.
- Some common examples of MTA are Microsoft Exchange, Exim, Sendmail, Amazon SES, and Oracle Beehive.
- In the corresponding sections, we will learn more about how the messaging system works and the role of MTA in the process. We will also touch upon the different types of MTA and the main functions they exhibit.

Working of MTA:

- MTA is a component of the message handling system. It works in conjunction with the other components to enable the email delivery process.
- The main elements in the email delivery process are as follows:
  1. **Mail User Agent (MUA):** MUA is the first point of contact in this process. An MUA is usually an application used by users to send or receive emails.
  2. **Mail Submission Agent (MSA):** It is an intermediate element that receives mail from MUA and transfers it to MTA. Practically, most MTAs work as MSA's to perform their functions.
  3. **Message Transfer Agent (MTA):** The MTA in the process can receive mail from either an MSA or another MTA, or even an MUA.
  4. **Message Delivery Agent (MDA):** This is the last stop where mails arrive before being sent into the users' inbox.
- So here, MTAs receive mail from an MSA, which, in turn, receives the mail from an MUA. Once the mail is received by an MTA the relaying process follows.
- If the recipient is not hosted locally the mail is forwarded to other MTAs. Finally, the mail is sent to the MDA which then delivers it into the recipient's inbox.
- The SMTP (Simple Mail Transfer Protocol) is used to send emails between servers (MTA) - also known as an SMTP relay.

- Then the message is routed directly over POP3 (Post Office Protocol - a one-way client-server protocol) or IMAP4 protocols. The IMAP and POP3 protocols are used by email clients to retrieve messages directly from the server.

Functions of a Message Transfer Agent (MTA):

1. Accept emails sent from the MUA (Mail User Agent).
2. Select a mail server to transfer emails, depending on the MX records and the domain name of the recipient.
3. Process deferrals if any and track the delivery status.
4. Send an auto-response to the sender ID, if the delivery fails.

Types of MTA Servers:

- The two main types of MTA servers are classified based on where they are hosted. These types are on-premise MTA servers and cloud-based MTA servers.
- Some mail servers can be physical servers located at an organization's premises. Other mail servers can be third-party virtually hosted servers that organizations can access to meet their email transfer needs.

1. On-Premise MTA Servers:

- On-premise MTA servers are mostly hosted by large enterprises or companies that have an in-house email infrastructure that includes both the software and the hardware component.
- Here, the mail servers use the organization's servers where all transferred emails are saved in an indexed database.
- The basic functionality of MTA remains the same in both cases. But building an on-premise MTA infrastructure can be expensive, although the goal here is to give the company complete control over its email transfer system.

2. Cloud-Based SMPT Servers:

- Cloud-based SMPT servers are, as the name suggests, a cloud-based infrastructure that enables email transfer. Examples of cloud-based SMPT servers are services like SendGrid or even Mailgun.
- The catch, in this case, is that, although the cloud-based servers are inexpensive they don't allow complete control over the mail sending and delivery infrastructure. It is essentially a third-party system that organizations can make use of.

4.2.3 SMTP (Simple Mail Transfer Protocol) [S-22, W-22, S-23, S-24, W-24]

- Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks.
- SMTP is a connection oriented, text based protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically a Transmission Control Protocol (TCP) connection.
- SMTP is standard application layer protocol for delivery of e-mail over a TCP/IP internetwork such as the Internet.
- E-mail system is implemented with the help of Message Transfer Agents (MTA). There are normally two MTA's in each mailing system, (one for sending emails and another for receiving e-mails).
- The formal protocol that defines the MTA client and server in the internet is called Simple Mail Transfer Protocol (SMTP).
- Fig. 4.25 shows the range of SMTP protocol.
- Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks.
- The formal protocol that defines the MTA (Message Transfer Agent) client and server in the Internet is called Simple Mail Transfer Protocol (SMTP).

Characteristics of SMTP:

1. SMTP is a push protocol and uses port number 24.
2. SMTP uses TCP at the transport layer.
3. SMTP uses persistent TCP connections, so it can send multiple emails at once.
4. SMTP is a connection-oriented protocol.
5. SMTP is an in-band protocol.
6. SMTP is a stateless protocol.

- By referring the above diagram, we can say that SMTP is used two times. That is between the sender and sender's mail server and between the sender's mail server and receiver's mail server.
- Another protocol is used between the receiver's mail server and receiver. SMTP simply defines how commands and responses must be sent back and forth.
- SMTP is a simple ASCII protocol. It establishes a TCP connection between a sender and port number 24 of the receiver.
- No checksums are generally required because TCP provides a reliable byte stream. After exchanging all the e-mail, the connection is released.

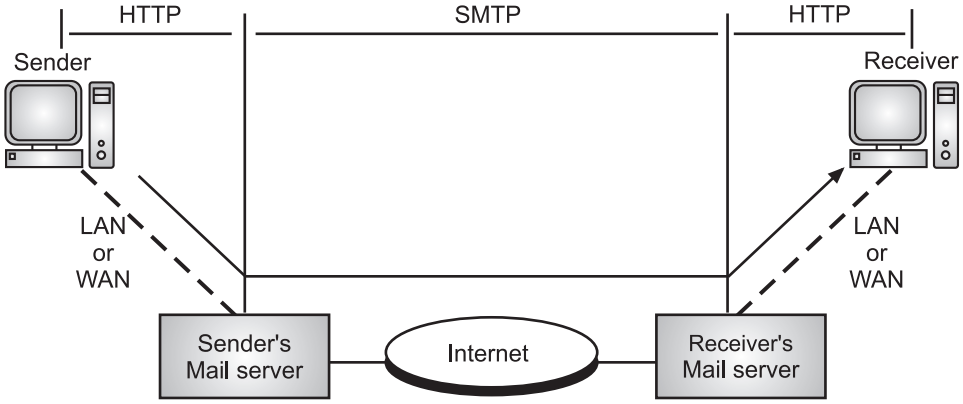

Fig. 4.25: SMTP

Commands and Responses:

- SMTP uses commands and responses to transfer messages between an MTA client and MTA server.
- Each command or reply is terminated by a two-character (carriage return and line feed) End-Of-Line (EOF) token.

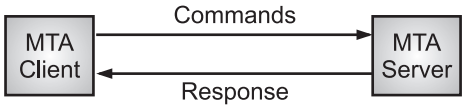

Fig. 4.26: Commands and Responses

- The SMTP provides for a two-way communication between the client (local) and server (remote) MTAs. The client MTA sends commands to the server MTA, which, in turn, sends replies (responses) back to the client MTA.
- Most useful SMTP commands are listed in following table:

Sr. No.	Command	Description
1.	HELO	Identifies the sender SMTP to the receiver SMTP – a hello command. This command is used by the client to identify itself.
2.	MAIL FROM	With this SMTP command the operations begin. The sender states the source email address in the "From" field and actually starts the email transfer. This command is used by the client to identify the sender of the message.
3.	RCPT TO	It identifies the recipient of the email; if there are more than one, the command is simply repeated address by address. This command is used by the client to identify the intended recipient of the message.

4.	DATA	With the DATA command the email content begins to be transferred. This shows the Body of the mail. This command is used to send the actual message. All lines that follow the DATA command are treated as the mail message. The message is terminated by a line containing just one period.
5.	QUIT	It terminates the SMTP conversation. This command terminates the message.
6.	RSET	This command aborts (reset) the current e-mail transaction.
7.	VRFY	The server is asked to verify whether a particular email address or username actually exists. This command is used to verify the address of the recipient, which is sent as the argument. The sender can ask the receiver to confirm that a name identifies a valid recipient.
8.	NOOP	This command is used by the client to check the status of the recipient. It requires an answer from the recipient.
9.	TURN	This command lets the sender and the recipient switch positions, whereby the sender becomes the recipient and vice versa. However, most SMTP implementations today do not support this feature.
10.	EXPN	This command asks the receiving host to expand the mailing list sent as the arguments and to return the mailbox addresses of the recipients that comprise the list.
11.	HELP	This command asks the recipient to send information about the command sent as the argument.

Model of SMTP:

- In the SMTP model user deals with the user agent (UA), for example, Microsoft Outlook, Netscape, Mozilla, etc.
- To exchange the mail using TCP, MTA is used. The user sending the mail doesn't have to deal with MTA as it is the responsibility of the system admin to set up a local MTA.
- The MTA maintains a small queue of mail so that it can schedule repeat delivery of mail in case the receiver is not available.
- The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.

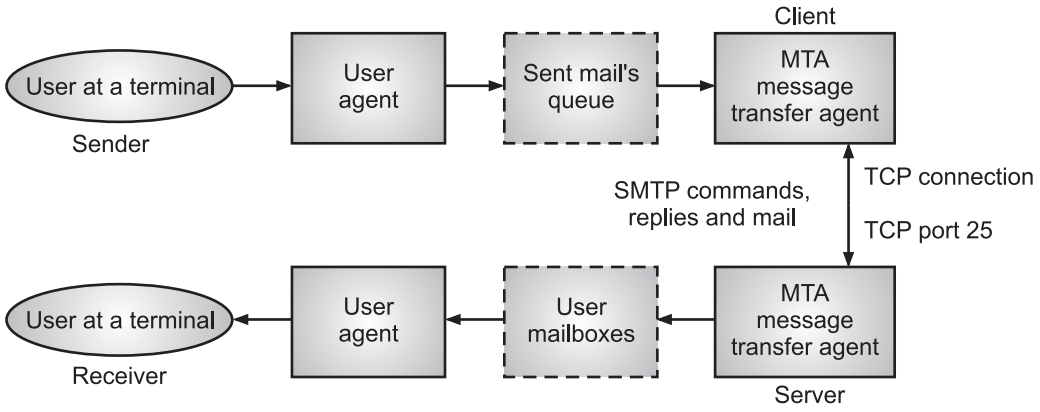

Fig. 4.27: SMTP Model

4.2.3.1 Components of SMTP

- Components of SMTP includes:
  1. **Mail User Agent (MUA):** It is a computer application that helps you in sending and retrieving mail. It is responsible for creating email messages for transfer to the mail transfer agent(MTA).

2. **Mail Submission Agent (MSA):** It is a computer program that receives mail from a Mail User Agent (MUA) and interacts with the Mail Transfer Agent (MTA) for the transfer of the mail.
3. **Mail Transfer Agent (MTA):** It is software that has the work to transfer mail from one system to another with the help of SMTP.
4. **Mail Delivery Agent (MDA):** A mail Delivery agent or Local Delivery Agent is basically a system that helps in the delivery of mail to the local system.

4.2.3.2 Working of SMTP

- The process of transferring a mail message in SNTP occurs in three phases namely, connection establishment, mail transfer and connection termination.

Step 1: Connection Establishment:

- After a client has made a TCP connection to the well-known port 24, the SMTP server starts the connection phase.
- Connection establishment phase in SMTP involves the following three steps, which are shown in Fig. 4.28.
  (i) The server sends code 220 (service ready) to tell the client that it is ready to receive mail. If the server is not ready, it sends code 421 (service not available).

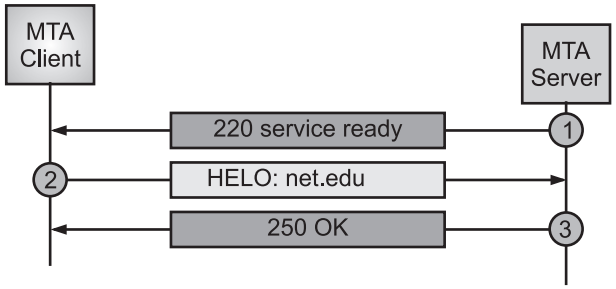

Fig. 4.28: Connection Establishment in SMTP

  (ii) The client sends the HELO message to identify itself using its domain name address. This step is necessary to inform the server of the domain name of the client. Remember that during TCP connection establishment, the sender and receiver know each other through their IP addresses.
  (iii) The server responds with code 240 (request command completed) or some other code depending on the situation.

Step 2: Message Transfer:

- After connection has been established between the SMTP client and server, a single message between a sender and one or more recipients can be exchanged.
- Message transfer phase in SMTP involves eight steps. Steps 3 and 4 are repeated if there is more than one recipient, (See Fig. 4.29).
  (i) The client sends the MAIL FROM message to introduce the sender of the message. It includes the mail address of the sender (mailbox and the domain name). This step is needed to give the server the return mail address for returning errors and reporting messages.
  (ii) The server responds with code 240 or some other appropriate code.
  (iii) The client sends the RCPT TO (recipient) message, which includes the mail address of the recipient.
  (iv) The server responds with code 240 or some other appropriate code.
  (v) The client sends the DATA message to initialize the message transfer.
  (vi) The server responds with code 344 (start mail input) or some other appropriate message.
  (vii) The client sends the contents of the message in consecutive lines. Each line is terminated by a two-character end-of-line token (carriage return and line feed). The message is terminated by a line containing just one period.
  (viii) The server responds with code 240 (OK) or some other appropriate code.

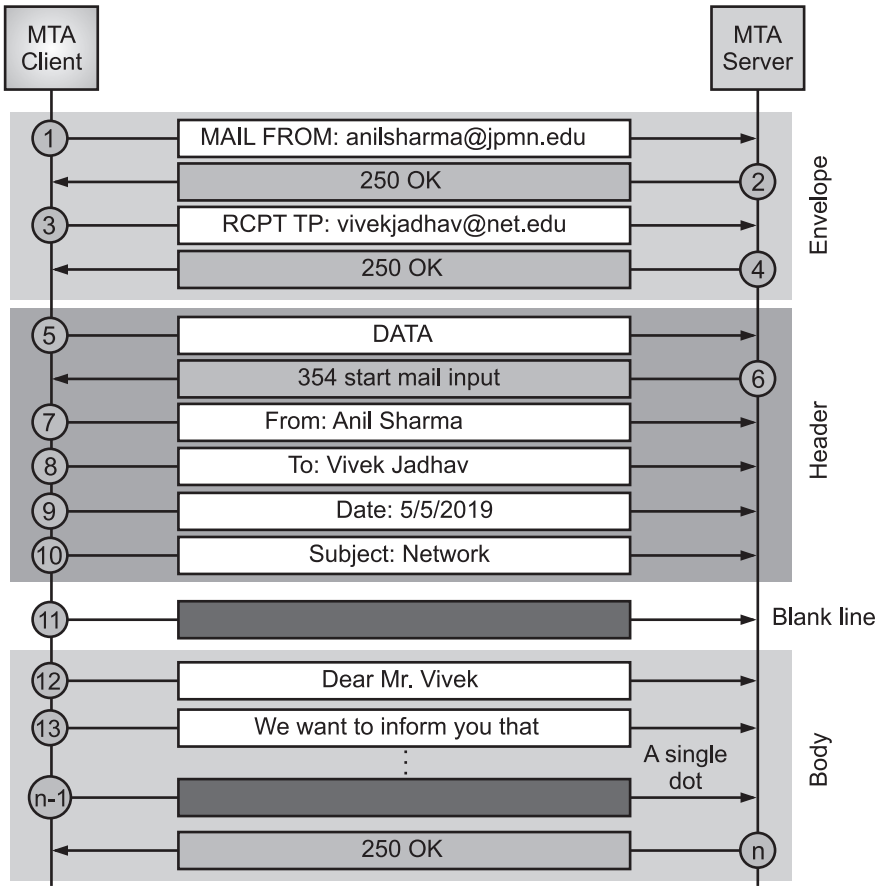

Fig. 4.29: Message Transfer in SMTP

Step 3: Connection Termination:

- In SMTP, after the message is transferred successfully, the client terminates the connection.
- Connection termination phase in SMTP involves following two steps (See Fig. 4.30).
  (i) The client sends the QUIT command.
  (ii) The server responds with code 221 or some other appropriate code.
- After the connection termination phase, the TCP connection must be closed.

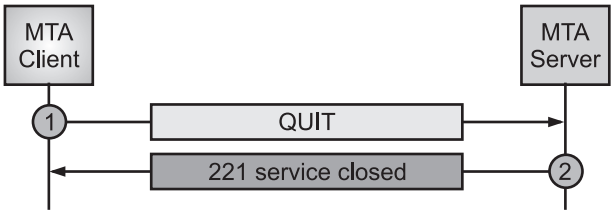

Fig. 4.30: Connection Termination in SMTP

Advantages of SMTP:

1. SMTP is a relatively simple, text-based protocol, in which one or more recipients of a message are specified along with the message text and possibly other encoded objects.
2. Easy to implement and higher speed.

Disadvantages of SMTP:

1. SMTP cannot transmit executable files or other binary objects.
2. SMTP servers may reject mail messages over a certain size.
3. SMTP gateways that translate from ASCII to EBCDIC and vice versa do not use a consistent set of code page mappings, resulting in translation problems.

4.2.4 Message Access Agent

- Message Access Agent (MAA) As we know to send and receive a mail two agents, message transfer agent and a message access agent are required. The message transfer agent transfers the message from client computer to the recipient's mail server.
- Now, it's the work of message access agent to pull the message from the mailbox present on the mail server at recipient's side to the recipient's computer, message access agent is also the final delivery at the recipient side.
- We have one message transfer agent i.e. SMTP (Simple Mail Transfer Agent), and we have two message access agents POP (Post Office Protocol) and IMAP (Internet Mail Access Protocol).

4.2.4.1 POP-Post Office Protocol [W-22, S-23, W-23, S-24, W-24]

- POP stands for Post Office Protocol. The POP is an application-layer Internet standard protocol used by e-mail clients to retrieve e-mail from a mail server.
- POP is generally used to support a single client. There are several versions of POP but the POP 3 is the current standard.
- Post Office Protocol version 3 (POP3) is a message access protocol that enables the client to fetch an e-mail from the remote mail server.
- Post Office Protocol version 3 (POP3) is a standard mail protocol used to receive e-mails from a remote server to a local email client. POP3 allows you to download email messages on your local computer and read them even when you are offline.
- 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.

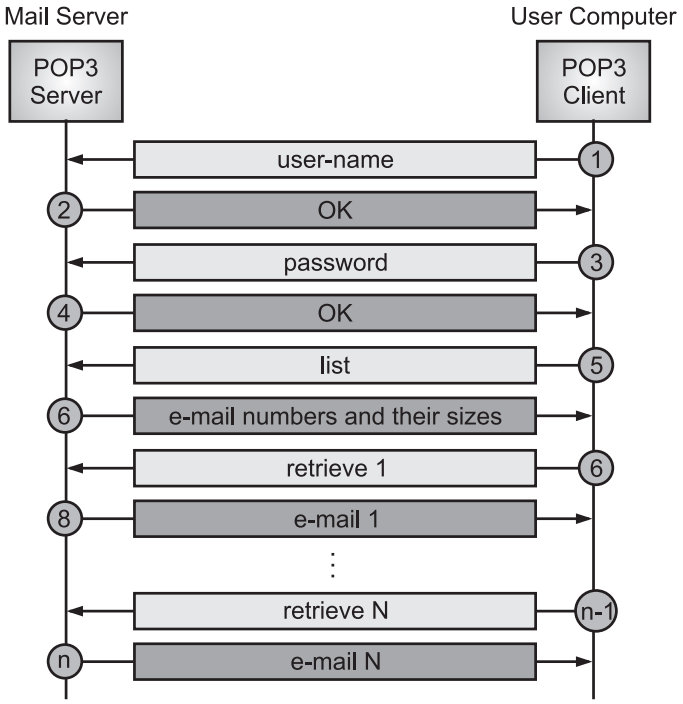

Fig. 4.31: POP3

POP Commands:

Sr. No.	Command	Description
1.	LOGIN	This command opens the connection.
2.	STAT	It is used to display number of messages currently in the mailbox.
3.	LIST	It is used to get the summary of messages where each message summary is shown.
4.	RETR	This command helps to select a mailbox to access the messages.
5.	DELE	It is used to delete a message.
6.	RSET	It is used to reset the session to its initial state.
7.	QUIT	It is used to log off the session.

Modes of POP3:

- POP3 has two modes namely, 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.

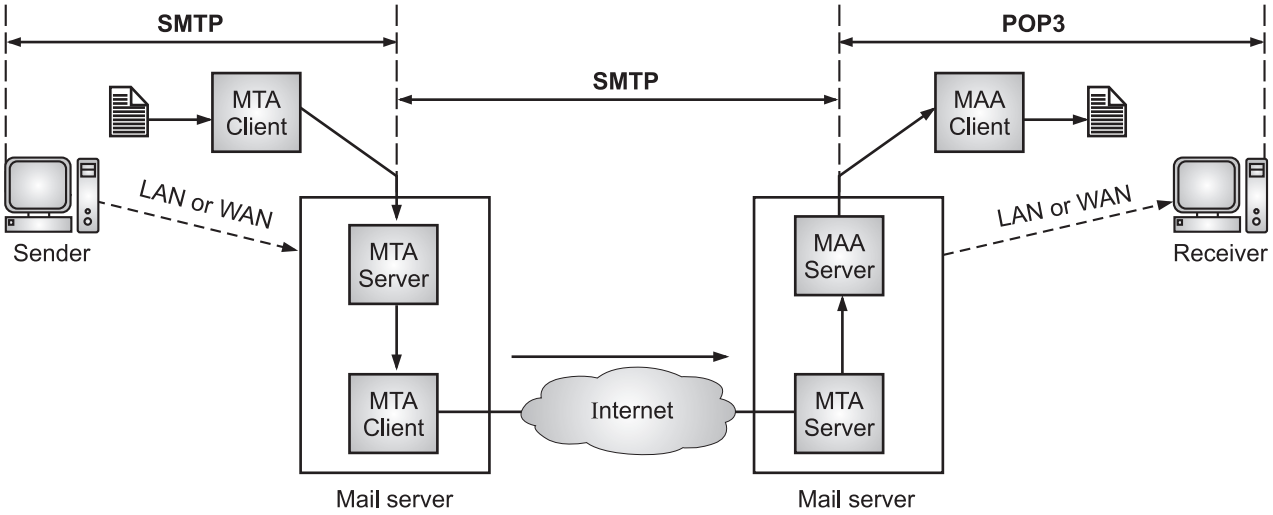

Fig. 4.32

Difference between SMTP and POP3:

Sr. No.	SMTP	POP3
1.	It is message transfer agent.	It is message access agent.
2.	Stands for Simple Mail Transfer Protocol.	Stands for Post Office Protocol version 3.
3.	Between sender and sender mail server and between sender mail server and receiver mail server.	Between receiver and receiver mail server.

Contd...

4.	It transfers the mail from sender's computer to the mail box present on receiver's mail server.	It allows to retrieve and organize mails from mailbox on receiver mail server to receiver's computer.
5.	SMTP is an application layer protocol that is used to send e-mail from the client to the mail server.	POP3 is an application layer protocol used by email systems to retrieve mail from e-mail servers.
6.	SMTP is an Internet protocol for transmitting e-mail over IP networks.	POP3 is an Internet protocol used to retrieve e-mail from a mail server POP3 access incoming mails.
7.	It uses port 24 for transfer of all outgoing e-mail.	An e-mail client connects with a POP3 server via port 110.

4.2.4.2 IMAP [W-23]

- IMAP stands for Internet Mail Access Protocol. IMAP is application layer protocol used to retrieve mail from the e-mail server.
- The IMAP is an Internet standard protocol used by email clients to retrieve email messages from a mail server over a TCP/IP connection.
- IMAP was designed with the goal of permitting complete management of an email box by multiple email clients, therefore clients generally leave messages on the server until the user explicitly deletes them.
- IMAP is another popular email protocol. It is more commonly used on internal networks rather than on the Internet. The current version is IMAP4.
- IMAP was first proposed in 1986. There exist five versions of IMAP as follows:
  - Original IMAP
  - IMAP2
  - IMAP3
  - IMAP2bis
  - IMAP4
- IMAP allows the client program to manipulate the e-mail message on the server without downloading them on the local computer. ·The e-mail is hold and maintained by the remote server.
- IMAP describes a method to retrieve e-mail messages. Unlike POP, when the user connects to an IMAP-capable server, copies of the messages are downloaded to the client application (See Fig. 4.33).
- The original messages are kept on the server until manually deleted. Users view copies of the messages in their e-mail client software. IMAP is a way of accessing electronic mail that is stored on a central server.

Characteristics of IMAP:

1. IMAP is a pull protocol.
2. IMAP uses TCP at the transport layer.
3. IMAP uses port number 143.
4. IMAP uses persistent TCP connections.
5. IMAP is a connection-oriented protocol.
6. IMAP is an in-band protocol.
7. IMAP is a stateful protocol.
8. IMAP distributes mail boxes across multiple servers.

Functions of IMAP4:

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.

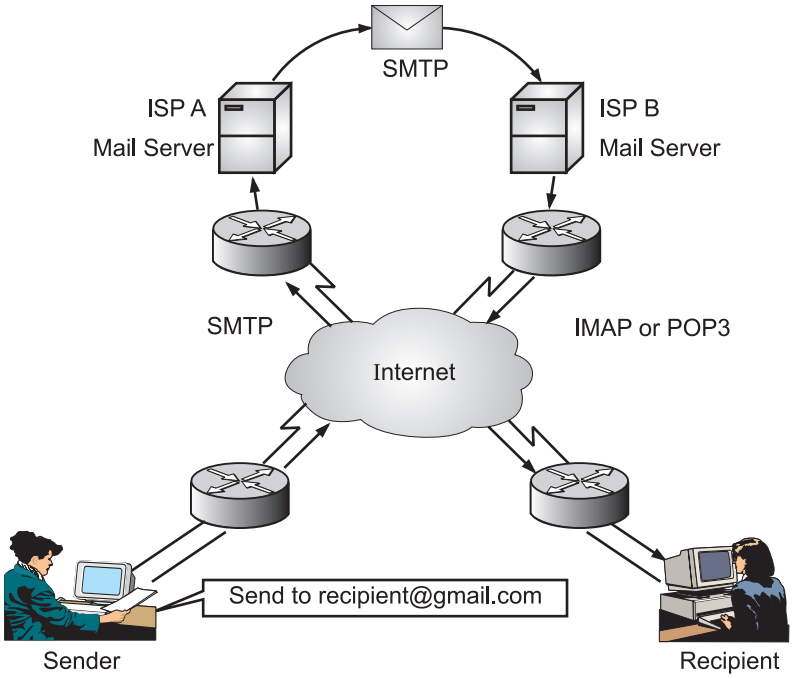

Fig. 4.33: SMTP, POP, and IMAP Operation

IMAP Commands:

Sr. No.	Command	Description
1.	IMAP_LOGIN	This command opens the connection.
2.	SELECT	This command helps to select a mailbox to access the messages.
3.	CREATE	It is used to create mailbox with a specified name.
4.	DELETE	It is used to permanently delete a mailbox with a given name.
5.	RENAME	It is used to change the name of a mailbox.
6.	LOGOUT	This command informs the server that client is done with the session. The server must send BYE untagged response before the OK response and then close the network connection.

Comparison between POP and IMAP:

Sr. No.	POP	IMAP
1.	Generally used to support single client.	Designed to handle multiple clients.
2.	Messages are accessed offline.	Messages are accessed online although it also supports offline mode.
3.	POP does not allow search facility.	It offers ability to search e-mails.
4.	All the messages have to be downloaded.	It allows selective transfer of messages to the client.
5.	Only one mailbox can be created on the server.	Multiple mailboxes can be created on the server.
6.	Not suitable for accessing non-mail data.	Suitable for accessing non-mail data i.e., attachment.
7.	It requires minimum use of server resources.	Clients are totally dependent on server.

Contd...

8.	POP requires less internet usage time.	IMAP requires more internet usage time.
9.	POP is a stateful protocol until the mail is downloaded as well as stateless across sessions.	IMAP is a stateful protocol because the IMAP server has to maintain a folder hierarchy for each of its users.

4.3 FILE TRANSFER PROTOCOL (FTP) [S-22, S-23, S-24]

- Files transferring from one computer to another computer is one of the most common tasks expected from a networking or internetworking environment.
- For transferring files to common protocols used in networking are File Transfer Protocol (FTP) and Trivial File Transfer Protocol (TFTP).
- File Transfer Protocol (FTP) is the standard mechanism provided by TCP/IP for copying a file from one host 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). Separation of commands and data transfer makes FTP more efficient.
- FTP uses two well-known TCP ports namely, Port 21 is used for the control connection, and port 20 is used for the data connection.
- FTP is a network protocol used to transfer files from one computer to another over a TCP network. TFTP is a simple version of FTP. TFTP is a network protocol used to transfer files between remote machines.

Differences between FTP and TFTP: **[S-22, S-23, W-24]**

Sr. No.	Parameters	FTP	TFTP
1.	Stands for	File Transfer Protocol.	Trivial File Transfer Protocol.
2.	Features	Authentication, encryption, and error recovery.	Basic file transfer only.
3.	Protocol Complexity	More complex and heavier.	Less complex and lightweight.
4.	Ports used	FTP works on ports 20 and 21.	TFTP works on port 69.
5.	Protocol used	FTP is based on TCP.	TFTP is based on UDP.
6.	Authentication	Authentication is must for FTP.	Authentication is not required in case of TFTP.
7.	Use Cases	General file transfer, Web servers etc.	Network device configuration, Booting etc.

4.3.1 Concept of FTP

- FTP transfers files between any two networked machines using it. FTP cannot be used to execute remote files as programs. FTP is used to copy files from one host to another.
- The file transfer protocol allows you to connect to a remote computer (host) using an FTP program on your machine, browse a list of files available, retrieve files, and navigate the directory structure of the host system.
- FTP is the simplest and most secure way to exchange files over the Internet. The most common use for FTP is to download files from the Internet.

4.3.1.1 Architecture of FTP

- Fig. 4.34 shows basic architecture of FTP. The server has two major components and the client has two major components.
- The control connection is made between the control processes at server and client side while the data connection is made between the data transfer processes.

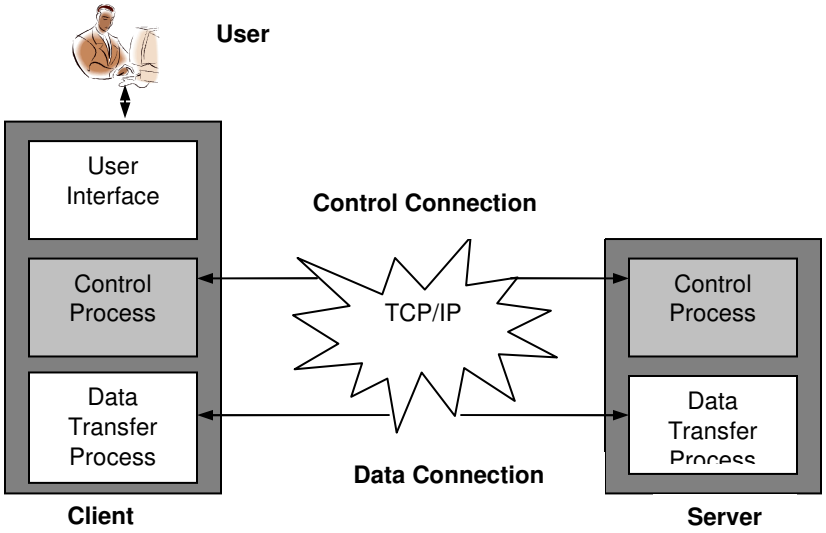

Fig. 4.34: Basic Architecture of FTP

- One more thing that is very important about the control and data connection is that, the control connection remains open during the entire FTP interactive session, while the data connection is opened when the user wants to transmit a file and then it is closed after the file transfer.
- In short, the data connection is opened and closed for each file transferred.

Benefits of FTP:

1. FTP is used to transfer files throughout the network.
2. FTP provides accessing to both directories and files with certain operations.
3. FTP is used to list and manipulate directories, type file contents. copy files between hosts, and other file operations.

4.3.1.2 How FTP Works? [W-23, S-24, W-24]

- File transfer protocol is one of the earliest Internet protocols, and is still used for uploading and downloading files between clients and servers.
- An FTP client is an application that can issue FTP commands to an FTP server, while an FTP server is a service or daemon running on a server that responds to FTP commands from a client.
- FTP commands can be used to change directories, change transfer modes between binary and ASCII, upload files, and download files.
- FTP uses Transmission Control Protocol (TCP) for reliable network communication by establishing a session before initiating data transfer.

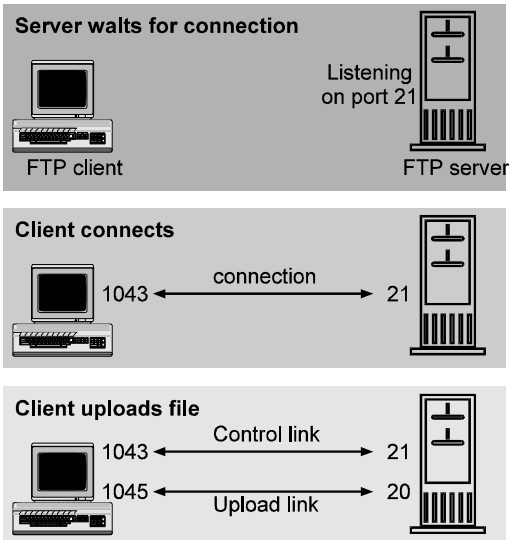

Fig. 4.35: Working of File Transfer Protocol (FTP)

- TCP port number 21 on the FTP server listens for connection attempts from an FTP client and is used as a control port for establishing a connection between the client and server, for allowing the client to send an FTP command to the server, and for returning the server's response to the command.
- Once, a control connection has been established, the server opens port number 20 to form a new connection with the client for transferring the actual data during uploads and downloads.
- We require two types of protocols for transferring the files on the network i.e., FTP and TFTP (Trivial File Transfer Protocol).
- FTP is a standard mechanism provided by TCP/IP for copying a file from one host to another. Fig. 42 shows command processing FTP.
- FTP uses the control connection to establish a communication between the client control process and the server control process.
- During this communication, the commands are sent from the client to the server and the responses are sent from the server to the client as shown in Fig. 4.36.

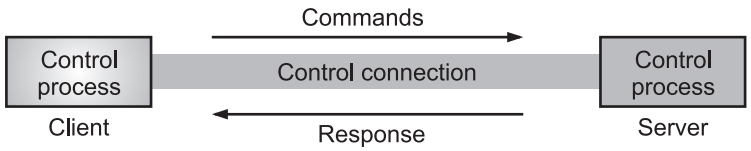

Fig. 4.36: Command Processing in FTP

FTP Commands:

Sr. No.	Command	Meaning
1.	CD	Change the working directory on the remote host.
2.	CLOSE	Closes the FTP connection.
3.	QUIT	Quits FTP.
4.	PWD	Displays the current working directory on the remote host.
5.	DIR or LS	Provides a directory listing of the current working directory.
6.	HELP	Displays a list of all client FTP commands.
7.	REMOTEHELP	Displays a list of all server FTP commands.
8.	TYPE	Allows the user to specify the file type.
9.	STRUCT	Specifies the files structure.

4.3.1.3 File Transfer in FTP [S-24]

- In FTP, file transfer occurs over the data connection under the control of the commands sent over the control connection.
- However, we should remember that file transfer in FTP means one of following three things (See Fig. 4.37).
- A file is to be copied from the server to the client (download). 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 (upload). 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. This 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.

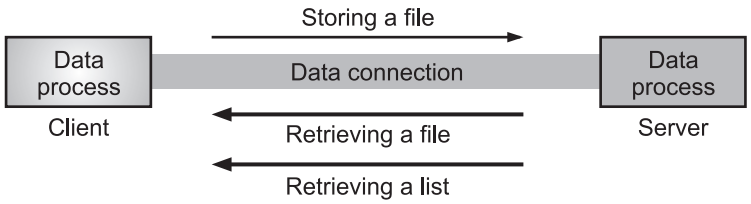

Fig. 4.37: File Transfer in FTP

- **Example:** Fig. 4.38 shows an example of using FTP for retrieving a list of items in a directory.

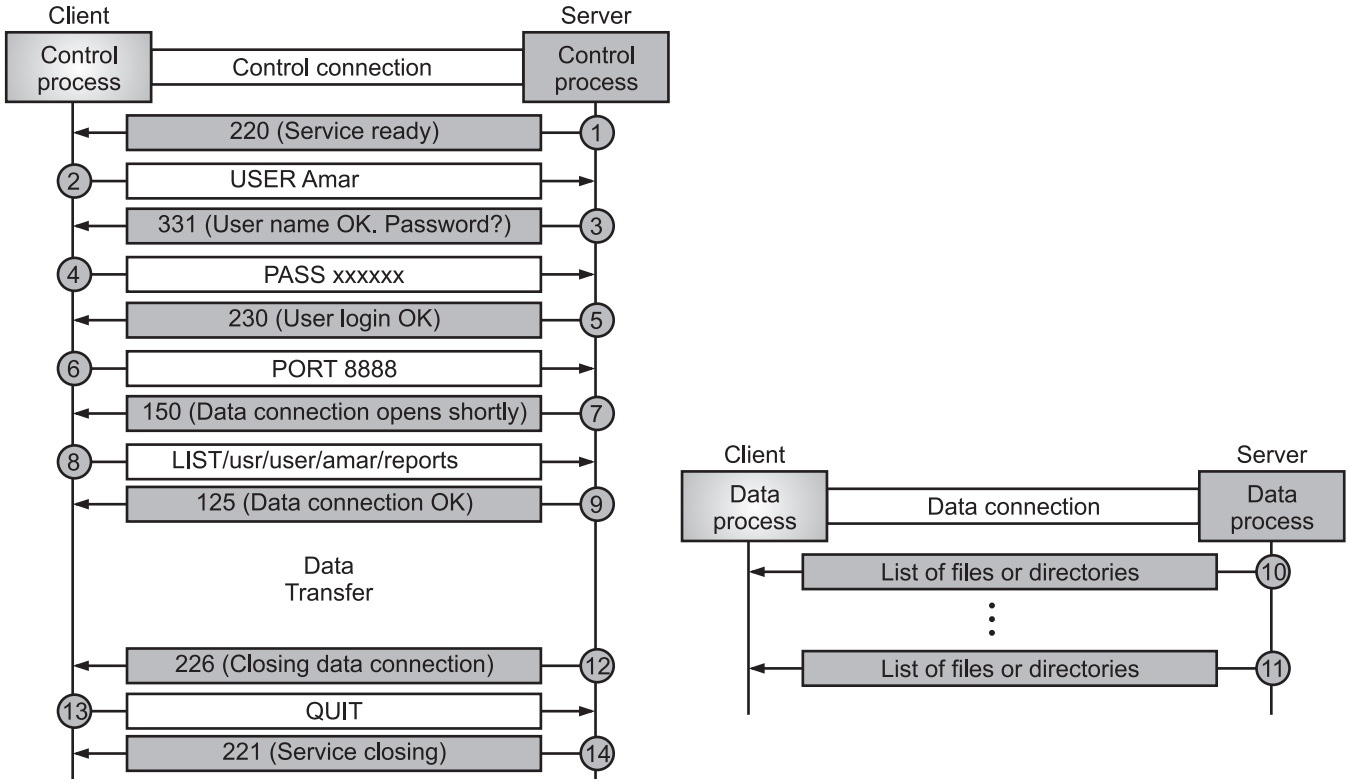

Fig. 4.38

- Steps in file transfer in FTP are given below:

  Step 1: After the control connection to port 21 is created, the FTP server sends the 220 (service ready) response on the control connection.

  Step 2: The client sends the USER command.

  Step 3: The server responds with 331 (user name is OK, password is required).

  Step 4: The client sends the PASS command.

  Step 5: The server responds with 230 (user login is OK).

  Step 6: The client issues a passive open on an ephemeral port for the data connection and sends the PORT command (over the control connection) to give this port number to the server.

  Step 7: The server does not open the connection at this time, but it prepares itself for issuing an active open on the data connection between port 20 (server side) and the ephemeral port received from the client. It sends response 140 (data connection will open shortly).

  Step 8: The client sends the LIST message.

  Step 9: Now the server responds with 124 and opens the data connection.

  Step 10: The server then sends the list of the files or directories (as a file) on the data connection. When the whole list (file) is sent, the server responds with 226 (closing data connection) over the control connection.

Step 11: The client now has two choices. It can use the QUIT command to request the closing of the control connection or it can send another command to start another activity (and eventually open another data connection). In our example, the client sends a QUIT command.

Step 12: After receiving the QUIT command, the server responds with 221 (service closing) and then closes the control connection.

FTP transfers files in three different modes: **[S-24]**

1. In **Stream mode**, the FTP handles the data as a string of bytes without separating boundaries.
2. In **Block mode**, the FTP decomposes the entire data into different blocks of data.
3. In **Compressed mode**, the FTP uses the Lempel-Ziv algorithm to compress the data.

Applications of FTP:

1. Uploading webpages to web servers for publishing on the Internet.
2. Browsing and downloading files from public software sites.
3. Organizations use FTP to allow employees to share files across different locations and branch offices.
4. Transferring large files among two parties that are too large for email attachments.
5. Downloading and uploading content like university's assignments via an FTP server.
6. Distributing the latest revisions of programs by software developers.
7. Employees use FTP to securely share files with coworkers and external business partners.
8. The IT teams use FTP to transfer data back to DR (disaster recovery) sites.

4.3.2 Anonymous File Transfer Protocol

- Anonymous File Transfer Protocol (AFTP) is a network protocol used for transmitting files using TCP-based networks.
- Anonymous file transfer protocol lets a user move files anonymously from one computer to another. Anonymous FTP operates at layer 7 in OSI model.
- anonymous FTP permits anonymous external computer users without any designated password or user ID to access the FTP server i.e., when a user accesses a file, they don't need to identify themselves.
- Hence, all the data contained within a website that allows Anonymous FTP should be considered publicly accessible.
- Anonymous File Transfer Protocol (FTP) is a method that lets users access public files from a remote server or archive site without requiring them to identify themselves to the server or site.
- The user uses an FTP program or the FTP command interface and enters "anonymous" as their user ID. The password may be furnished by the FTP server or the user may provide their own.
- Anonymous FTP is a way for remote users to use an FTP server even if they don't have an assigned user ID and password. It enables unprotected access of selected information about a remote system without entering a password.
- The information is usually publicly accessible, which means it can be read by anyone who logs into the server.
- However, the remote site determines what this information would be or how much would be available for general access.
- The person or organization that owns the information and the remote system must control their information and ensure that only appropriate information is made available for public access.
- To access the information, the user logs onto the FTP host server using the user ID anonymous and any password.
- The user account will typically accept any string as a password, including the user's email address. After they log in, the user will have limited access rights to the files on the server.

- The server also imposes some operating restrictions so only certain operations are allowed on the anonymous FTP.
- These include the following:
  - logging onto the FTP server;
  - listing the contents of or files under a limited number of directories; and
  - retrieving files and content from these directories.

How do anonymous FTP sessions work?

- An anonymous FTP session starts when a user logs into a remote server. To start the session, they will use the ftp command and the hostname/IP. The user could use either of the following commands to access the archive site via FTP:
  - ftp openfiles.samplecompany.com
  - ftp 128.103.129.6
- This will invoke the FTP program and establish the user's connection to the remote host. At this point, they can see the contents of the server and retrieve the files they need.
- After they finish, they will exit the FTP program, which will close the connection and terminate the anonymous FTP session.
- Every response the FTP program gives is preceded by a number called a reply code. The user's password -- whatever it may be -- is never shown on the screen.
- Here, is how an anonymous FTP session works step by step.
  - The user logs into the local host and invokes the FTP program.
  - They open a connection to the host using either the host name or its IP address.
  - After connecting to the remote host, they log in with the username "anonymous."
  - They provide a password. This could be "guest," their email address, or anything else that the site requests.
  - They issue the requisite FTP commands depending on what they want to do on the archive site (e.g., change directories or retrieve a file from a particular directory).
  - The user exits the FTP program.
  - The connection to the archive host closes and the anonymous FTP session is terminated.

4.4 REMOTE LOGGING

- Remote logging is the process of collecting and storing log data from multiple remote systems (such as computers, servers, or network devices) onto a centralized log server.
- This allows administrators to monitor, analyze, and troubleshoot issues across multiple systems from a single location.
- Various examples of remote logging tools are: syslog, Windows Event Forwarding (WEF), Graylog, splunk, etc.

How Remote Logging works?

1. **Log Generation**: The remote machines generate log files containing system events, application activities, security logs, etc.
2. **Log Transmission**: Logs are sent over a network to a central log server using protocols such as:
   - Syslog (UDP/TCP port 514).
   - Rsyslog (Advanced version of Syslog).
   - Journalctl (For systemd-based Linux systems).
   - Windows Event Forwarding (WEF).
   - Logstash (For ELK Stack).

3. **Log Storage and Analysis**: The central server stores and processes the logs for real-time monitoring, alerting, or forensic analysis.

4.4.1 Telnet [W-22, S-23, S-24, W-24]

- TELNET stands for TErminaL NETwork. TELNET is a protocol used to log in to remote computer on the internet.
- TELNET is an application layer protocol, which can be used on the internet or LAN (Local Area Network).
- It provides a bi-directional interactive text-oriented communication service by using virtual terminal connection.
- TELNET is basically a client server protocol, which is based on a reliable connection- oriented transport. It uses a port number 23, to establish the connection with TCP (Transmission Control Protocol).
- TELNET provides the ability to perform remote logons to remote hosts. Telnet operates using a client and server.
- TELNET is a general-purpose client-server application program. Fig. 4.39 shows the client-server interaction in TELNET.

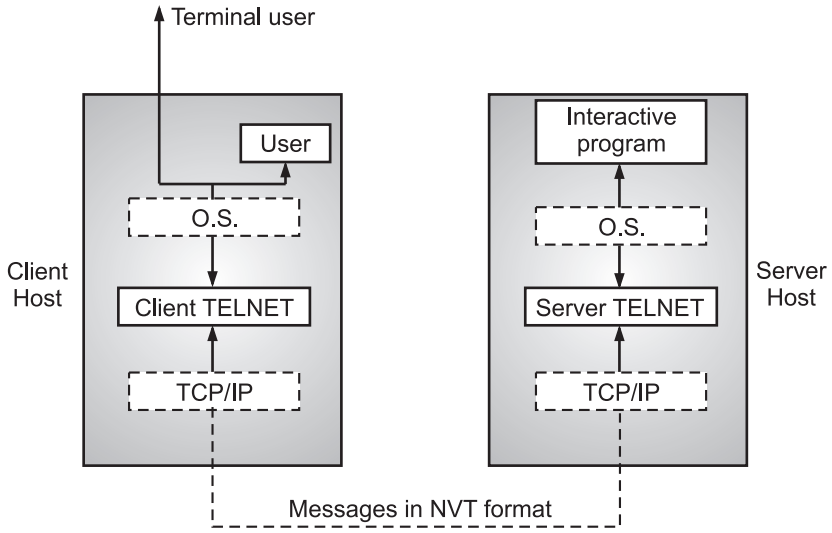

Fig. 4.39: Telnet Client Server Interaction

- The client TELNET protocol is accessed through the local Operating System (OS) either by user or by a user at a terminal.
- It provides services to enable a user to log on to the operating system of a remote machine, to initiate the running of a program on that machine.
- All the commands and data entered at the user terminal are passed by the local operating system to the client TELNET process which then passes them, using the reliable stream service provided by TCP, to the correspondent server TELNET.
- The two TELNET protocols communicate with each other using commands that are encoded in a standard format known as network virtual terminal.
- The character set used for commands is ASCII. All input and output data relating to an interaction is transferred as ASCII strings.
- If this is different from the local character set being used, the corresponding TELNET will carry out any necessary mapping functions. Thus, the two TELNET protocol entities also perform the role of the presentation layer in an OSI stack.

Logging in TELNET:

- The logging process can be further categorized into two parts namely, Local Login and Remote Login.

1. Local Login:

- Whenever a user logs into its local system, it is known as local login.

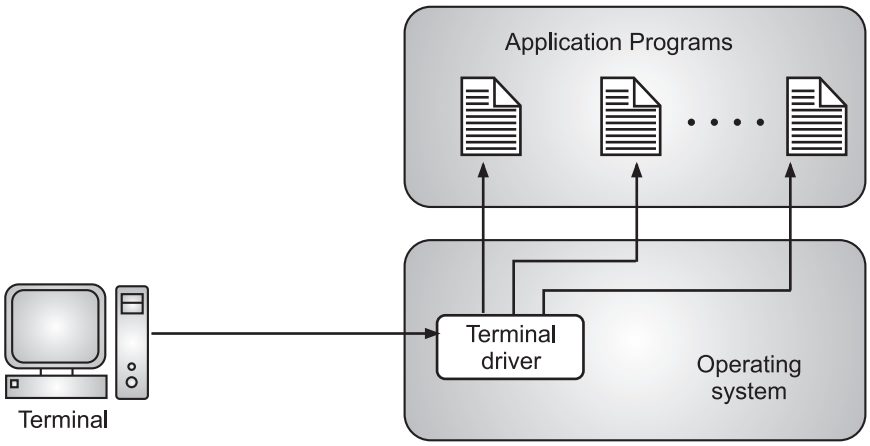

Fig. 4.40: Local Login

- The Procedure of Local Login:
  - Keystrokes are accepted by the terminal driver when the user types at the terminal.
  - Terminal Driver passes these characters to OS.
  - Now, OS validates the combination of characters and opens the required application.

2. Remote Login:

- Remote Login is a process in which users can log in to a remote site i.e. computer and use services that are available on the remote computer.
- With the help of remote login, a user is able to understand the result of transferring the result of processing from the remote computer to the local computer.

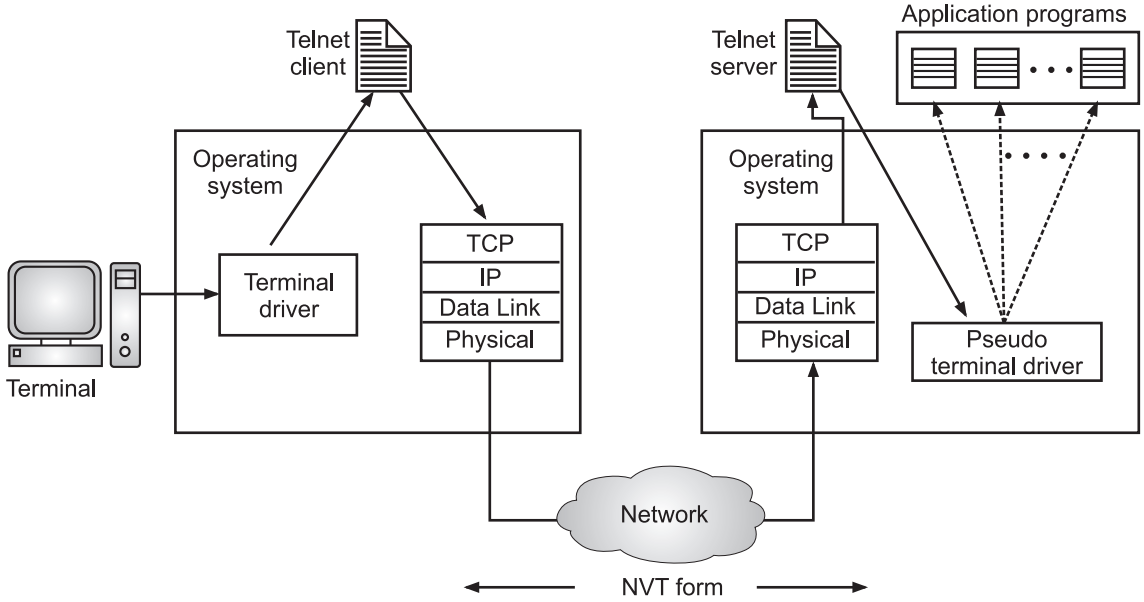

Fig. 4.41: Remote Login-in Logging

- The Procedure of Remote Login:
  - When the user types something on the local computer, the local operating system accepts the character.

- The local computer does not interpret the characters, it will send them to the TELNET client.
- TELNET client transforms these characters to a universal character set called Network Virtual Terminal (NVT) characters and it will pass them to the local TCP/IP protocol Stack.
- Commands or text which are in the form of NVT, travel through the Internet and it will arrive at the TCP/IP stack at the remote computer.
- Characters are then delivered to the operating system and later on passed to the TELNET server.
- Then TELNET server changes those characters to characters that can be understandable by a remote computer.
- The remote operating system receives characters from a pseudo-terminal driver, which is a piece of software that pretends that characters are coming from a terminal.
- The operating system then passes the character to the appropriate application program.

TELNET Commands:

- Commands of Telnet are identified by a prefix character, Interpret As Command (IAC) with code 255. IAC is followed by command and option codes.
- Following are some of the important TELNET commands:

Character	Decimal	Binary	Meaning
WILL	251	11111011	1. Offering to enable. 2. Accepting a request to enable.
WON'T	252	11111100	1. Rejecting a request to enable. 2. Offering to disable. 3. Accepting a request to disable.
DO	253	11111101	1. Approving a request to enable. 2. Requesting to enable.
DON'T	254	11111110	1. Disapproving a request to enable. 2. Approving an offer to disable. 3. Requesting to disable.

- Following are some common options used with the telnet:

Code	Option	Meaning
0	Binary	It interprets as 8-bit binary transmission.
1	Echo	It will echo the data that is received on one side to the other side.
3	Suppress go ahead	It will suppress go ahead signal after data.
5	Status	It will request the status of TELNET.
6	Timing mark	It defines the timing marks.
8	Line width	It specifies the line width.
9	Page size	It specifies the number of lines on a page.
24	Terminal type	It set the terminal type.
32	Terminal speed	It set the terminal speed.
34	Line mode	It will change to the line mode.

Modes of Operations in Telnet:

1. **Default Mode:** If no other modes are invoked then this mode is used. Echoing is performed in this mode by the client. In this mode, the user types a character and the client echoes the character on the screen but it does not send it until the whole line is completed.

2. **Character Mode:** Each character typed in this mode is sent by the client to the server. A server in this type of mode normally echoes characters back to be displayed on the client's screen.
3. **Line Mode:** Line editing like echoing, character erasing, etc. is done from the client side. The client will send the whole line to the server.

Uses of Telnet:

1. Remote Administration and Management.
2. Network Diagnostics.
3. Understanding Command-Line Interfaces.
4. Accessing Bulletin Board Systems (BBS).
5. Automation and Scripting.

Advantages of Telnet:

1. It provides remote access to someone's computer system.
2. Telnet allows the user for more access with fewer problems in data transmission.
3. Telnet saves a lot of time.
4. The oldest system can be connected to a newer system with telnet having different operating systems.

Disadvantages of Telnet:

1. As it is somehow complex, it becomes difficult to beginners in understanding.
2. Data is sent here in form of plain text, that's why it is not so secured.
3. Some capabilities are disabled because of not proper interlinking of the remote and local devices.

4.4.2 Remote Desktop

- A Remote Desktop is a technology that allows a user to access and control a computer from another location over a network or the internet.
- It enables users to interact with a remote computer as if they were physically present in front of it.

Features of Remote Desktop:

1. **Remote Control:** Users can control a distant computer's mouse, keyboard, and applications.
2. **File Transfer:** Some remote desktop solutions allow transferring files between local and remote computers.
3. **Multi-User Support:** Multiple users can access the same system remotely if permissions are granted.
4. **Secure Connection:** Encryption and authentication protocols ensure secure access.

Common Remote Desktop Software:

1. **Microsoft Remote Desktop (RDP):** Built into Windows for accessing other Windows PCs.
2. **TeamViewer:** Popular for remote support and personal use.
3. **AnyDesk:** Lightweight and fast remote access software.
4. **Chrome Remote Desktop:** A free browser-based remote desktop tool from Google.
5. **VNC (Virtual Network Computing):** Open-source remote desktop software.

4.4.2.1 Remote Desktop Access (Using Command Line Interface (CLI))

- We can access a remote machine using either the command line or a GUI-based remote desktop tool.

1. **Windows (Using Remote Desktop Protocol - RDP):**

   Step 1: Open Command Prompt or PowerShell.

   Press Win + R, type cmd or powershell, and hit Enter.

Step 2: Use the `mstsc` Command.

Run the following command:

```
mstsc /v:192.168.1.100
```

This will open the Remote Desktop Connection window and try to connect to the remote machine.

Step 3: Enter Credentials.

When prompted, enter the remote machine's username and password.

2. **Linux & macOS (Using SSH):**

Step 1: Open Terminal.

On Linux/macOS, press Ctrl + Alt + T or open Terminal.

Step 2: Use SSH Command.

Run following command:

```
ssh admin@192.168.1.100
```

If connecting for the first time, type yes to accept the fingerprint.

Enter the remote machine's password when prompted.

Step 3: Access the Remote Shell.

Once logged in, you can run commands on the remote machine.

4.4.2.2 Remote Desktop Access (Using GUI-based Remote Desktop Tool)

- We can access a remote machine using GUI-based remote desktop tool.

1. **Windows Remote Desktop Connection:**

Step 1: Open Remote Desktop Connection

Press Win + R, type mstsc, and press Enter.

Step 2: Enter Remote Machine Details

In the Computer field, enter the IP address or hostname of the remote machine.

Click Connect.

Step 3: Enter Credentials

Input the username and password of the remote machine.

Click OK to log in.

2. **Using TeamViewer or AnyDesk:**

Step 1: Install TeamViewer/AnyDesk

Download and install TeamViewer **or** AnyDesk on both the local and remote machines.

Step 2: Open the Application

Launch TeamViewer or AnyDesk on both machines.

Step 3: Connect to Remote Machine

In TeamViewer, enter the Partner ID and click Connect.

In AnyDesk, enter the Remote Address and click Connect.

Step 4: Authenticate

Enter the password if required.

Grant remote access when prompted.

3. **Chrome Remote Desktop**

Step 1: Install Chrome Remote Desktop Extension

Install the extension from the Chrome Web Store.

Step 2: Set Up Remote Access

Open Chrome Remote Desktop and Enable Remote Access on the remote machine.

Step 3: Connect to the Remote Machine

Open Chrome Remote Desktop on your local machine.

Click on the remote computer's name.

Enter the PIN to access it.

4.5 WORLD WIDE WEB (WWW) AND HYPERTEXT TRANSFER PROTOCOL

- The WWW is a way of exchanging information between computers on the Internet. The WWW is combination of all resources and users on the Internet that are using the Hypertext Transfer Protocol (HTTP).
- WWW is a distributed client server service and HTTP is used to retrieve information from the Web.
- Fig. 4.42 shows communication between client and server using HTTP protocol. Communication between client computers and web servers is done by sending HTTP Requests and receiving HTTP Responses.

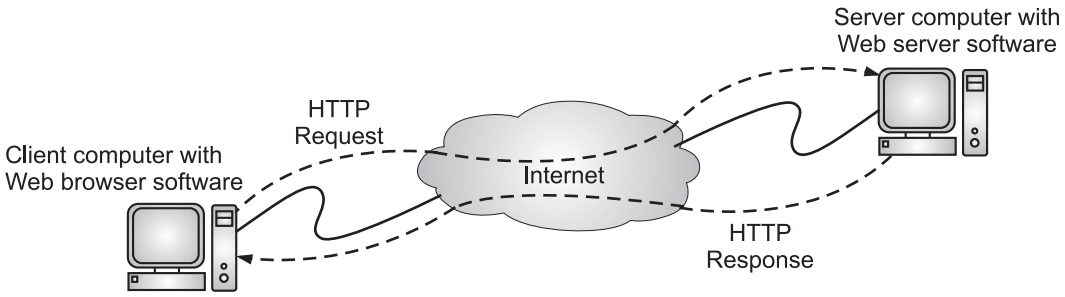

Fig. 4.42: Communication between Client and Server

4.5.1 World Wide Web (WWW) [S-22, W-22, W-23]

- The WWW today is a distributed client-server service, in which a client using a browser can access a service using a server. The WWW) is a repository of information linked together from points all over the world.
- WWW (or Web) refers to the collection of public websites connected to the internet worldwide, together with client devices such as computers and cell phones that access its content.
- The term WWW has a unique combination of flexibility, portability and user-friendly features that distinguish it from other services provided by the Internet.

Terminologies in WWW:

- A document on the Web is called a **Web page** and is identified by a unique address called the Uniform Resource Locator (URL).
- In other words, the **World Wide Web (WWW)** consists of files, called pages or web pages, which contain information and links to resources throughout the Internet.
- A web page is an electronic document written in a computer language called **HTML (HyperText Markup Language)**. The service provided is distributed over many locations called sites. Each site holds one or more documents, referred to as Web pages.
- The term **Hypertext** means creating documents that refer to other documents. In a hypertext document, a part of text can be defined as a link to another document. When a hypertext is viewed with a browser, the link can be clicked to retrieve the other document.
- The term **Hypermedia** is applied to document that contains links to other textual document or documents containing graphics, video or audio.

- Each Web page, however, can contain some **links** to other Web pages in the same or other sites.
- A Web page can be simple or composite. A simple Web page has no link to other Web pages; a composite Web page has one or more links to other Web pages. Each Web page is a file with a name and address.

Types of Web Documents: **[S-22]**

- Web pages are also known as HTML documents. A web page can contain huge information including text, graphics, audio, video and hyperlinks. There are following types of web pages:
- A **static web page** (sometimes called a flat page/ stationary page) is a web page that is delivered to the user exactly as stored. **[S-22]**

  Static documents are fixed-content documents that are created and stored in a server. The client can get a copy of the document only. User cannot do any modification or interact with the information on static web page. Static documents are prepared using one of the several languages such as HyperText Markup Language (HTML),

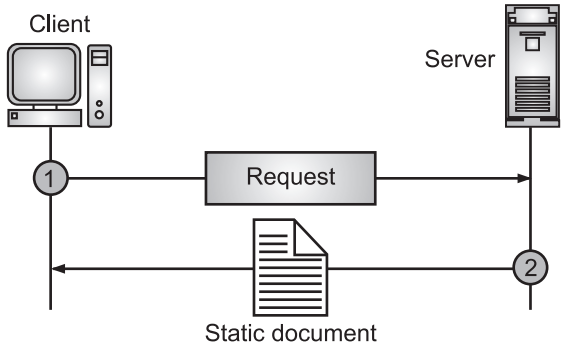

Fig. 4.43: Static Web Page

  Extensible Markup Language (XML), Extensible Style Language (XSL), and Extended Hypertext Markup Language (XHTML).

- A **dynamic web page** is a web page with web content that varies based on parameters provided by a user or a computer program. A dynamic document is created by a Web server whenever a browser requests the document. When a request arrives, the Web server runs an application program or a script that creates the dynamic document. The server returns the output of the program or script as a response to the browser that requested the document. In dynamic web pages, possible to change a portion/content of a web page without loading the entire web page. **[S-22]**

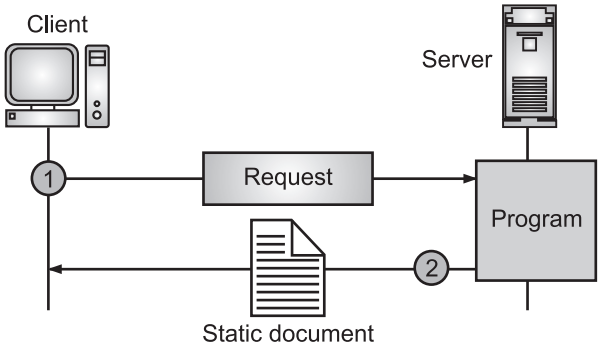

Fig. 4.44: Dynamic Document using CGI

- A URL is also commonly referred to as a Web address. A URL is a type of Uniform Resource Identifier (URI), which is a generic term for many types of names and addresses on the Web.
- A **Uniform Resource Locator (URL)** or **web address** is the address of a resource on the Internet. A URL is a type of uniform resource identifier (URI).
- A URL contains the following information:
  - The **protocol** used to an access the resource. Many different protocols can retrieve a document; among them are Gopher, FTP, HTTP, News, and TELNET. The most common today is HTTP.
  - **Host** is the location of the server (whether by IP address or domain name). Web pages are usually stored in computers, and computers are given domain name aliases that usually begin with the characters "www".
  - The **port number** on the server (optional). If the port is included, it is inserted between the host and the path, and it is separated from the host by a colon.
  - **Path** is the location of the resource in the directory structure of the server.

Fig. 4.45: URL

- **Example:** http://www.sun.com the anatomy of this address is shown in Fig. 4.46.

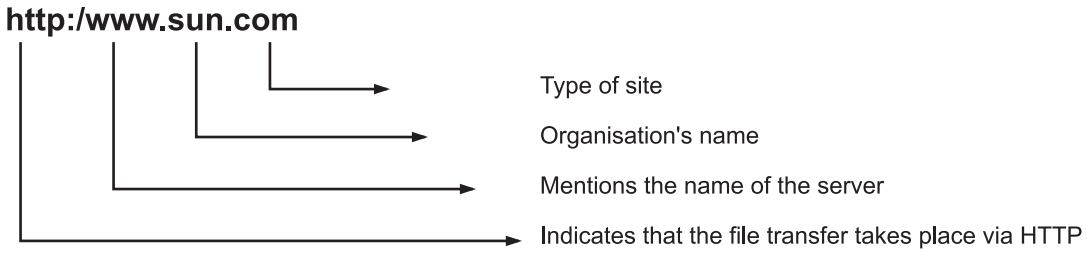

Fig. 4.46: Example of URL

- A **Website** is a location on the Internet where an individual, company, or organization keeps their Web pages and related files (such as text, graphic and video files). We display a Web page on the computer screen by using a program called a **Web browser**.
- A person user can retrieve and open a Web page in a Web browser either by entering a URL in the Web browser's address box or by clicking a hypertext link. When a user wants to access a Web page using either method, the user's Web browser sends a Web server a request for the Web page.
- **Web server** is a computer where the web content is stored. A Web server is a computer that delivers Web pages. The Web server's reaction to the user's request is called the response.
- **Web publishing** or **Online publishing**, is the process of publishing content on the Internet. It includes creating and uploading websites, updating web pages, and posting blogs online and the published content may include text, images, videos and other types of media.

4.5.1.1 Architecture of WWW [W-22, W-23]

- WWW works on client server architecture, in which a client using a browser can access a service using a server. Fig. 4.47 shows architecture of WWW.
- Today, the WWW is a distributed client server service.
- The service provided is distributed over many locations called sites and each site holds one or more documents i.e., Web pages.
- Client sends a request through its browser to the server using HTTP protocol which specifies the way the browser and web server communicates.

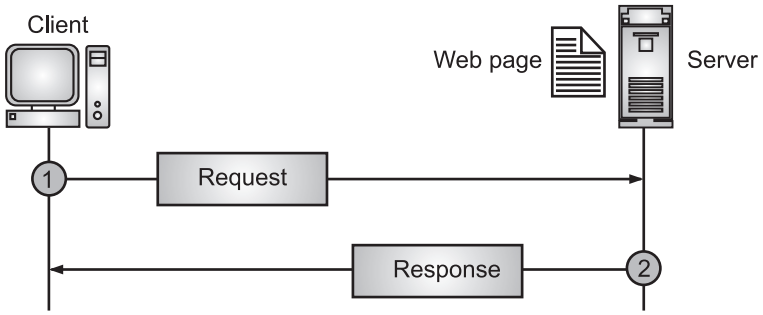

Fig. 4.47

- Then server receives request using HTTP protocol and checks its search for the requested web page. If found it returns it back to the web browser and close the HTTP connection.
- Now the browser receives the web page, it interprets it and display the contents of web page in web browser's window.
- Fig. 4.48 shows how WWW works. The main web document and the image are stored in two separate files in the same site (file X and file Y) and the referenced text file is stored in another site (file Z).
- Since, we are dealing with three different files, (namely, X, Y and Z) we need three transactions if we want to see the whole document. The first transaction (request/response) retrieves a copy of the main document (file X), which has a reference (pointer) to the second and the third files.
- When a copy of the main document is retrieved and browsed, the user can click on the reference to the image to invoke the second transaction and retrieve a copy of the image (file Y).
- If the user further needs to see the contents of the referenced text file, she can click on its reference (pointer) invoking the third transaction and retrieving a copy of the file Z.

- Note that although files X and Y both are stored in site X, they are independent files with different names and addresses. Two transactions are needed to retrieve them.

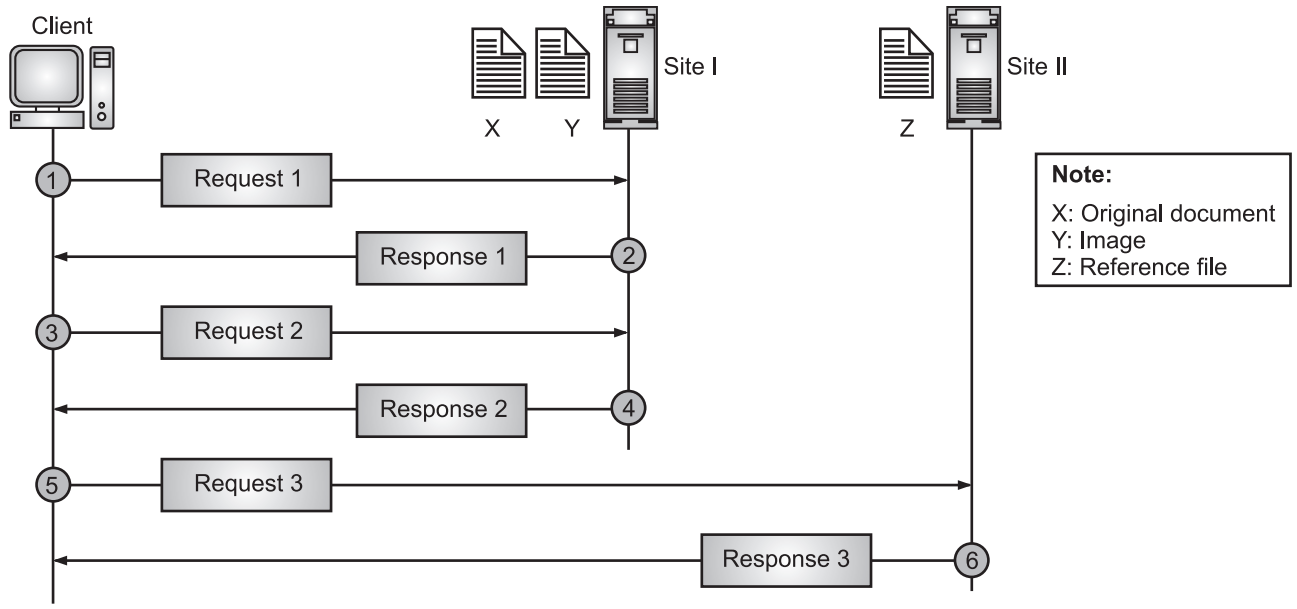

Fig. 4.48

4.5.2 HTTP (HyperText Transfer Protocol) [W-22, W-24]

- HTTP application layer protocol is used to transfer data across the Web or WWW. HTTP is the foundation for data communication for the World Wide Web (i.e. internet) since 1990.
- HTTP is the underlying protocol used by the World Wide Web and this protocol defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands.
- HTTP was designed for communication between web browsers and web servers. HTTP follows a classical client-server model, with a client opening a connection to make a request, then waiting until it receives a response.
- HTTP is a stateless protocol, meaning that the server does not keep any data (state) between two requests.
- The standard web transfer protocol is HyperText Transfer Protocol (HTTP) used mainly to access data on World Wide Web (WWW). It is similar to FTP because it transfers files and uses the services of TCP.
- HTTP is like SMTP because the data transferred between the client and the server look like SMTP messages.
- The HTTP protocol consists of two fairly distinct items: the set of requests from browsers to servers and the set of responses going back the other way.
- HTTP use the services of TCP on well-known port 80. All the newer versions of HTTP support following two kinds of request:
  1. **Simple Request:** A simple request is just a single GET line naming the page desired without the protocol version. The response is just a raw page with no headers, no MIME (Multipurpose Internet Mail Extensions), and no encoding.
  2. **Full Request:** Full request are indicated by the presence of the protocol version on the GET request line. Requests may consist of multiple lines, followed by a blank line to indicate the end of the request. The first line of a full request contains the command protocol/version. Subsequent lines contain RFC 822 headers.

Characteristics of HTTP:

1. **Application Level:** HTTP operates at the application level. It assumes a reliable connection-oriented transfer protocol such as TCP but does not provide reliability or retransmission itself.
2. **Request/Response:** Once, a transport session has been established, the browser must send an HTTP request to which the other side responds.
3. **Stateless:** Each HTTP request is self-content. The server does not keep a history of pervious request or sessions.
4. **Bidirectional Transfer:** HTTP allows transfer from a server to a browser and also vice-versa.
5. **Support for Caching:** To improve response time, a browser caches a copy of each web page it retrieves. If user requests a page again, HTTP allows the browser to interrogate the server to determine whether the contents of the page have changed since the copy was cached.
6. **Support for Intermediaries:** HTTP allows a machine along the path between a browser and a server to act as a proxy server that caches web pages and answers a browser's request from its cache.

4.5.2.1 HTTP Transaction

- Fig. 4.49 shows the HTTP transaction between the client and server.
- HTTP uses the services of TCP, HTTP itself is a stateless protocol, which means that the server does not keep information about the client.
- The client initializes the transaction by sending a request.
- The server replies by sending a response.

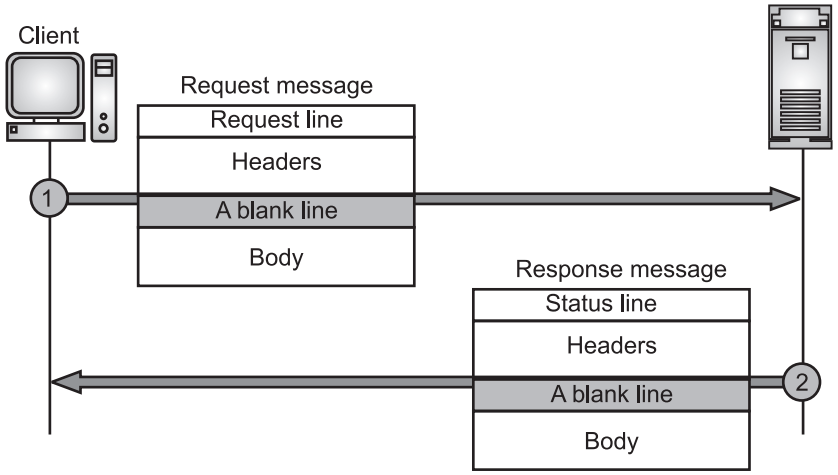

Fig. 4.49: HTTP Transaction

- **Request Message:** The format of request is shown in Fig. 4.50. A request message consists of a request line, a header and sometimes a body.

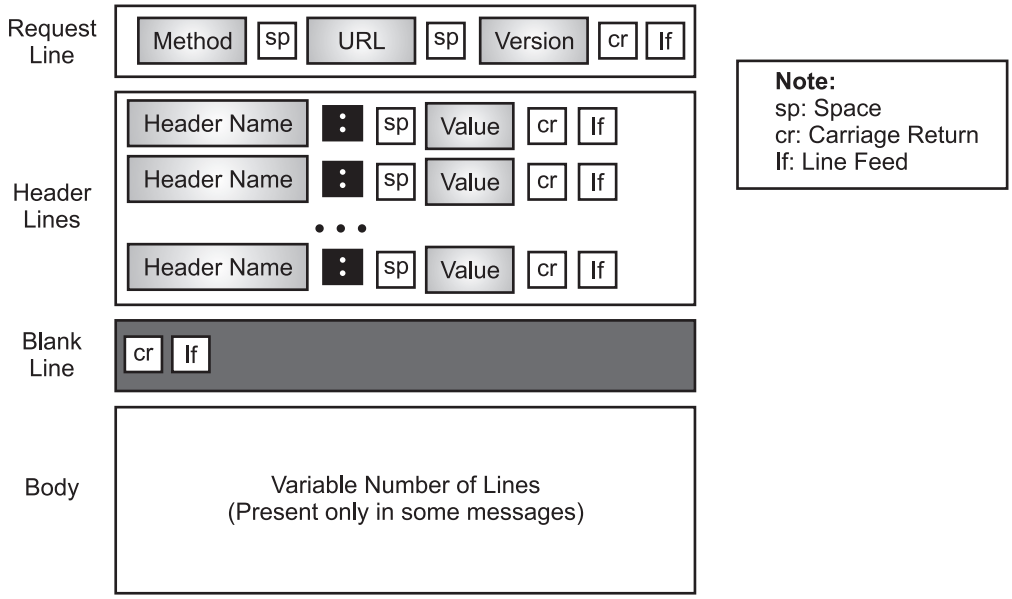

Fig. 4.50: Format of Request Message

- **Request Line:** The first line in a request message is called request line. There are three fields in this line separated by some character delimiter. The fields are called methods, URL and version. The method field defines the request type.
- The built in HTTP request methods are:

Sr. No.	Method	Description
1.	GET	Request a document from the server.
2.	HEAD	Request information about a document but not the document itself.
3.	PUT	Sends the document from the server to the client.
4.	POST	Sends some information from the client to the server.
5.	TRACE	Echoes the incoming request.
6.	DELETE	Remove the web page.
7.	LINK	Connects two existing resources.
8.	UNLINK	Breaks an existing connection between two resources.

- **Response Message:** A response message consists of a status line, header lines, a blank line and sometimes a body. **[W-22, S-24]**

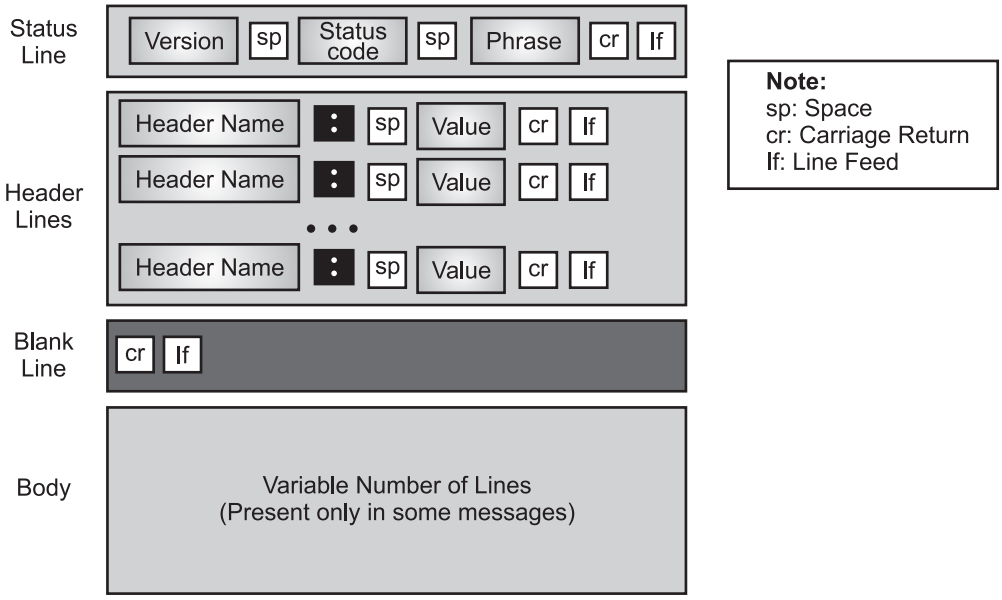

Fig. 4.51: Format of the Response Message

- **Status Line:** The first line in a response message is called the status line. There are three fields in this line separated by spaces and terminated by a carriage return and linefeed. The first field defines the version of HTTP protocol, currently 1.1. The status code field defines the status of the request. 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. Finally, the codes in the 400 range indicate an error at the server site.
- **Body**: The body contains the document to be sent from the server to the client. The body is present unless the response is an error message.

Difference between FTP and HTTP:

Sr. No.	FTP	HTTP
1.	FTP is used to access and transfer files.	HTTP is used to view websites.
2.	FTP is efficient in transferring larger files.	HTTP is efficient in transferring smaller files like web pages.
3.	FTP can be accessed via the command line or graphical client of its own.	The common HTTP client is the browser.
4.	FTP establishes two connection one for data and one for the control connection.	HTTP establishes data connection only.
5.	FTP uses TCP's port number 20 and 21.	HTTP uses TCP's port number 80.
6.	If you are using FTP, ftp will appear in URL.	If you are using HTTP, http will appear in URL.
7.	FTP session (stateful).	No session (stateless).
8.	FTP is comparatively simple.	Web clients and servers became very complex since they need to support many protocols, scripting languages, file types etc. Complexity is also a security problem.
9.	FTP is better suited (faster, more efficient) for large files.	HTTP is better suited for the transfer of many small files.
10.	FTP has a control and a data connection and communicates TCP port numbers for data connection in control connection.	HTTP uses a single TCP connection for control and data.
11.	FTP requires a password.	HTTP does not require authentication.
12.	FTP transmits data as ASCII or binary.	HTTP always sends data in binary format.

4.6 PRETTY GOOD PRIVACY (PGP) [W-22]

- PGP stands for Pretty Good Privacy (PGP) which is invented by Phil Zimmermann. PGP provide e-mail with privacy, integrity, and authentication. PGP can be used to create a secure e-mail message.
- PGP was designed to provide all four aspects of security, i.e., privacy, integrity, authentication, and non-repudiation in the sending of email.
- PGP uses a digital signature (a combination of hashing and public key encryption) to provide integrity, authentication, and non-repudiation.
- PGP uses a combination of secret key encryption and public key encryption to provide privacy. Therefore, we can say that the digital signature uses one hash function, one secret key, and two private-public key pairs.
- PGP is an open source and freely available software package for e-mail security. PGP provides authentication through the use of Digital Signature.
- It provides confidentiality through the use of symmetric block encryption. It provides compression by using the ZIP algorithm, and EMAIL compatibility using the radix-64 encoding scheme.

4.6.1 Services of PGP

- PGP includes the following services which are as follows:
  1. **Authentication:** The hash function used is SHA-1 which makes a 160 bit message digest. EP (DP) defines public encryption (decryption) and the algorithm used can be RSA or DSS. The set of SHA-1 and RSA supports an effective digital signature scheme. Because of the strength of RSA the recipient is guaranteed that only the possessor of the connecting private key can make the signature. Because of the strength of SHA-1 the recipient is guaranteed that no one else can create a new message that connects the hash code and therefore the signature of the original message.
  2. **Confidentiality:** It is a service supported by PGP is confidentiality which is provided by encrypting messages to be transmitted or to be saved locally as files. In some cases, the user has a best of CAST-128, IDEA or 3DES in 64 bit cipher feedback (CFB) mode. The symmetric key is used only once and is generated as a random number with the required number of bits. It is acquired along with the message and is encrypted using the recipient's public key.
  3. **Confidentiality and Authentication:** The both services can be used for the same message. First, a signature is produced for the plaintext message and prepended to the message. Therefore, the plaintext message plus signature is encrypted using CAST-128 (or IDEA or 3DES), and the session key is encrypted using RSA. This sequence is desirable to the opposite encrypting the message and thus producing a signature of the encrypted message. It is usually more convenient to save a signature with a plaintext version of a message. Moreover, for the goals of third-party verification, if the signature is implemented first, a third party need not be concerned with the symmetric key when testing the signature.
  4. **Compression:** As a default, PGP restrict the message after using the signature but before encryption. This has the advantage of storing space both for e-mail transmission and for file storage.
  5. **E-mail compatibility:** Some electronic mail systems only allows the use of blocks including ASCII text. When PGP is used, minimum part of the block to be transmitted is encrypted.
  6. **Segmentation:** E-mail facilities are restricted to a maximum message length. For instance, some facilities accessible throughout the internet set a maximum length of 50,000 octets. Some message higher than that should be broken up into smaller segments, each of which is mailed independently.

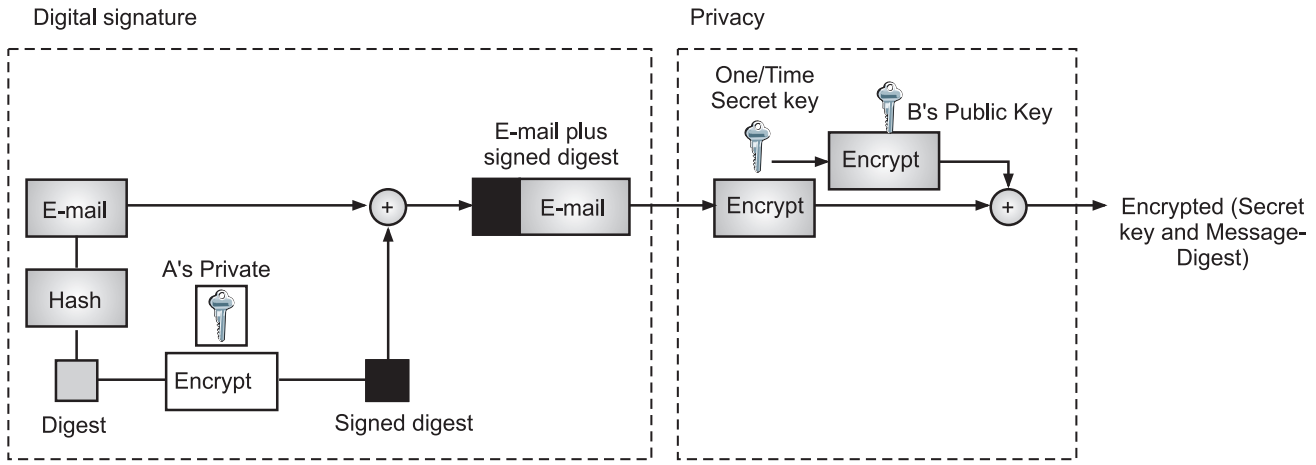

Fig. 4.52: PGP at Sender Side

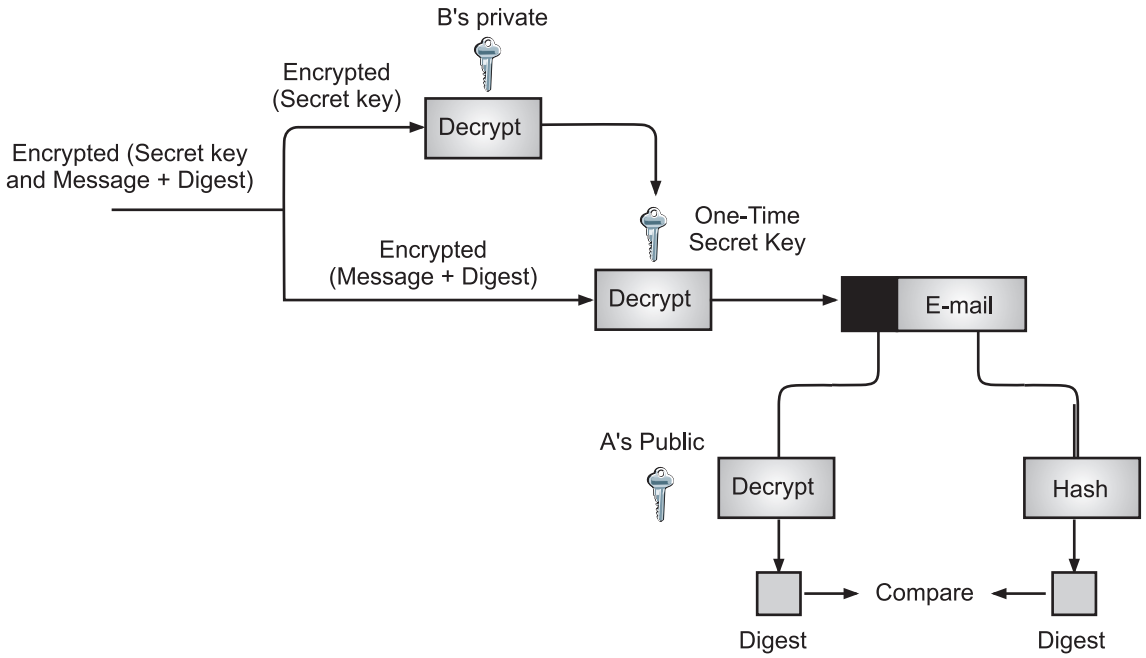

Fig. 4.53: PGP at Receiver Site

- Following are the steps taken by PGP to create secure e-mail at the sender site:
  - The e-mail message is hashed by using a hashing function to create a digest.
  - The digest is then encrypted to form a signed digest by using the sender's private key, and then signed digest is added to the original email message.
  - The original message and signed digest are encrypted by using a one-time secret key created by the sender.
  - The secret key is encrypted by using a receiver's public key.
  - Both the encrypted secret key and the encrypted combination of message and digest are sent together.
- Following are the steps taken to show how PGP uses hashing and a combination of three keys to generate the original message:
  - The receiver receives the combination of encrypted secret key and message digest is received.
  - The encrypted secret key is decrypted by using the receiver's private key to get the one-time secret key.
  - The secret key is then used to decrypt the combination of message and digest.
  - The digest is decrypted by using the sender's public key, and the original message is hashed by using a hash function to create a digest.
  - Both the digests are compared if both of them are equal means that all the aspects of security are preserved.

Disadvantages of PGP Encryption:

1. **Difficult to Administration:** The different versions of PGP complicate the administration.
2. **Compatibility Issues:** Both the sender and the receiver must have compatible versions of PGP. For example, if you encrypt an email by using PGP with one of the encryption techniques, the receiver has a different version of PGP which cannot read the data.
3. **Complexity:** PGP is a complex technique. Other security schemes use symmetric encryption that uses one key or asymmetric encryption that uses two different keys. PGP uses a hybrid approach that implements symmetric encryption with two keys. PGP is more complex, and it is less familiar than the traditional symmetric or asymmetric methods.

4. **No Recovery:** Computer administrators face the problems of losing their passwords. In such situations, an administrator should use a special program to retrieve passwords. For example, a technician has physical access to a PC which can be used to retrieve a password. However, PGP does not offer such a special program for recovery; encryption methods are very strong so, it does not retrieve the forgotten passwords results in lost messages or lost files.

4.6.2 Security Parameters Services

- PGP ensures secure communication using several key security parameters. These parameters help protect confidentiality, integrity, authentication, and non-repudiation in digital communication.

1. **Encryption Algorithms (Confidentiality):** PGP uses hybrid encryption to secure messages:
   - Symmetric Encryption (Fast Encryption of Data)
   - AES (Advanced Encryption Standard)
   - IDEA (International Data Encryption Algorithm)
   - Triple DES (3DES)
   - CAST-128

   Asymmetric Encryption (Key Exchange and Signature Verification):
   - RSA (Rivest-Shamir-Adleman)
   - DSA (Digital Signature Algorithm)
   - ElGamal
2. **Digital Signatures (Authentication & Integrity):** PGP uses digital signatures to verify the sender and ensure data integrity:
   - SHA-256, SHA-512 (Secure Hash Algorithms) – Used for message integrity.
   - MD5 (Obsolete due to vulnerabilities) – Older hashing algorithm.
3. **Key Management (Public and Private Keys):** PGP uses public-key cryptography for secure communication:
   - Public Key (Shared with others to encrypt messages).
   - Private Key (Kept secret by the owner to decrypt messages and sign data).
   - Key Pair Generation (Generated using RSA, DSA, or ElGamal).
4. **Web of Trust (Key Authentication):** PGP does not rely on a central authority like a CA (Certificate Authority). Instead, it uses:
   - User-based Key Authentication (Users sign each other's keys to verify authenticity).
   - Trust Levels:
     - Fully Trusted (Verified by multiple users).
     - Marginally Trusted (Verified by a few users).
     - Untrusted (Unknown key authenticity).
5. **Compression (Efficiency and Security):**
   - PGP compresses data before encryption using:
   - ZIP (Default Compression Algorithm)
   - ZLIB (Alternative Compression Method)
6. **Session Keys (One-Time Encryption Keys):**
   - PGP generates a unique symmetric key (session key) for each message, which is then encrypted with the recipient's public key.
7. **Key Revocation (Compromised Key Management):** If a PGP key is lost or compromised, it can be revoked:
   - Revocation Certificates (Pre-generated keys that can disable a compromised key).
   - Key Expiry Dates (Keys can be set to expire after a specific time).

4.6.3 PGP Key Rings

- In Pretty Good Privacy (PGP), Key Rings are databases that store cryptographic keys used for encryption, decryption, and authentication. PGP maintains two types of key rings.
- a "key ring" refers to a file that stores either public or private keys, used for encrypting and decrypting data and verifying digital signatures.
- In Pretty Good Privacy (PGP), a key ring is a data structure used to store cryptographic keys. It is critical for managing the public and private keys required for encryption, decryption and digital signatures.
- PGP uses two types of key rings namely, the public key ring and the private (or secret) key ring.
  1. **Public Key Ring:** Stores the public keys of other users as well as the user's own public key.
  2. **Private Key Ring:** Stores the user's private keys, which are used for decrypting messages and creating digital signatures.

How PGP Uses Key Rings:

- When you want to send an encrypted message to someone, you use their public key, which is stored in their public key ring.
- When you receive an encrypted message, you use your private key, stored in your private key ring, to decrypt it.
- Similarly, when you want to sign a message, you use your private key, and the recipient can verify the signature using your public key.
- Example, Yogita may need to send messages to many people; she needs key rings. In this case, Yogita needs a ring of public keys, with a key belonging to each person with whom Alice needs to correspond (send or receive messages). In addition, the PGP designers specified a ring of private/public keys. One reason is that Yogita may wish to change her pair of keys from time to time. Another reason is that Yogita may need to correspond with different groups of people (friends, colleagues, and so on). Yogita may wish to use a different key pair for each group. Therefore, each user needs to have two sets of rings: a ring of private/public keys and a ring of public keys of other people.

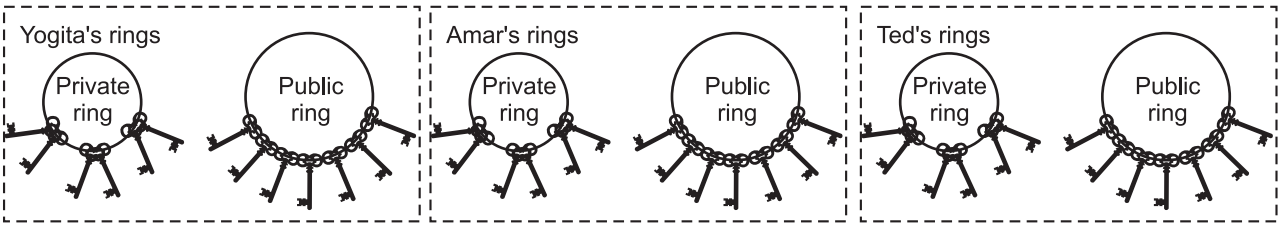

Fig. 4.54: Key Rings in PGP

- Yogita for example, has several pairs of private/public keys belonging to her and public keys belonging to other people. Note that everyone can have more than one public key. Two cases may arise.
  1. Yogita needs to send a message to another person in the community.
     a. She uses her private key to sign the digest.
     b. She uses the receiver's public key to encrypt a newly created session key.
     c. She encrypts the message and signed digest with the session key created.
  2. Yogita receives a message from another person in the community.
     a. She uses her private key to decrypt the session key.
     b. She uses the session key to decrypt the message and digest.
     c. She uses her public key to verify the digest.

4.6.4 PGP Algorithms

- PGP defines a set of asymmetric-key and symmetric-key algorithms, cryptography hash functions and compression methods.
- The key Algorithms used in PGP are explained below:

1. **Public-Key Cryptography (e.g., RSA):**

- PGP uses public-key cryptography for key exchange and encryption/decryption.
- Each user has a pair of keys: a public key for encrypting messages and a private key for decrypting them.
- The sender encrypts the message with the recipient's public key, and only the recipient with the corresponding private key can decrypt it.

2. **Symmetric-Key Cryptography (e.g., 3DES, AES):**

- Once a session key is established using public-key cryptography, PGP uses a symmetric-key algorithm to encrypt the actual message data.
- This is done for efficiency, as symmetric algorithms are faster than public-key algorithms for encrypting large amounts of data.

3. **Hashing (e.g., SHA-1, SHA-256):**

- PGP uses hashing algorithms to create a "fingerprint" or message digest of the data, which is then digitally signed.
- This ensures that the message hasn't been tampered with during transit.

4. **Compression (e.g., ZIP):**

- PGP can optionally compress the data before encryption to reduce the size of the message.

5. **Digital Signatures:**

- PGP uses digital signatures to verify the authenticity and integrity of messages.
- The sender signs the message digest with their private key, and the recipient can verify the signature using the sender's public key.

4.6.5 PGP Certificates

- A PGP Certificate is a digitally signed document that contains a user's public key, identity information, and trust indicators. It allows others to verify the authenticity of a user's public key in a Web of Trust system.
- In Pretty Good Privacy (PGP), certificates are used to associate a public key with the identity of its owner.
- They play a crucial role in verifying the authenticity of public keys and ensuring secure communication.
- PGP certificates are part of the "web of trust" model, which relies on decentralized trust rather than a centralized certificate authority.

A PGP certificate includes:

1. **Public Key:** Used for encryption and signature verification.
2. **User Identity (Name, Email, etc.):** Helps associate the key with a real person.
3. **Key ID & Fingerprint:** Unique identifiers for the public key.
4. **Key Expiry Date:** Optional, defines the key's validity period.
5. **Digital Signature:** The key is signed by the key owner or others to verify authenticity.
6. **Trust Level:** Shows how much the key is trusted in the Web of Trust.

How PGP certificates works?

- **Key Generation:** A user generates a public-private key pair. The public key is packaged into a PGP certificate.
- **Certificate Distribution:** The certificate is shared via email, key servers, or personal exchange.
- **Verification and Signing:** Other users can sign the certificate to establish trust.
- **Encryption and Authentication:** The public key in the certificate is used to encrypt messages. It also helps verify digital signatures.

Applications of PGP Certificates:

1. **Email Encryption**: Ensure secure communication by encrypting emails using verified public keys.
2. **Digital Signatures**: Authenticate messages and verify their integrity using certificates.
3. **File Encryption**: Protect sensitive files by associating them with trusted public keys.

Practice Questions

1. Enlist application layer protocols.
2. What is WWW? Define it?
3. With the help of diagram explain architecture WWW?
4. Define the following terms:
   (i) Web page
   (ii) URL
   (iii) Web site
   (iv) Hypermedia.
5. Enlist types of web pages.
6. What is HTTP? How it Works? Explain its messages diagrammatically.
7. What is meant by file transfer? Which protocols used for file transfer?
8. What is FTP?
9. Explain the working of FTP with diagram.
10. What TFTP? Enlist its features.
11. Differentiate between FTP and TFTP.
12. What is e-mail? Give its structure.
13. Describe the term web-based mail in detail.
14. Describe the following protocols with their frame/packet/UDP format:
    (i) SNMP
    (ii) POP
    (iii) IMAP
    (iv) DHCP
    (v) SMTP.
15. Compare POP and SMTP.
16. What is meant by remote login?
17. Describe DHCP with its operation and static and dynamic allocation.
18. What is TELNET? Describe in detail.
19. Describe PGP in detail.
20. Explain PGP algorithms.

MSBTE Questions with Answers

Summer 2023

1. Differentiate between FTP and TFTP (2 points). **[2 M]**

Ans. Refer to Section 4.3.

2. Describe the working of TELNET. **[4 M]**

Ans. Refer to Section 4.4.1.

3. Distinguish between SMTP and POP3 protocol. **[4 M]**

Ans. Refer to Sections 4.2.3 and 4.2.4.1.

4. Explain the process of resolving the given host name into IP address using DNS. **[4 M]**

Ans. Refer to Section 4.1.

Winter 2023

1. Elaborate need of domain name system. **[2 M]**

Ans. Refer to Section 4.1.

2. Explain working of world wide web. **[4 M]**

Ans. Refer to Section 4.5.1.1.

3. Construct a suitable diagram for each below commands of FTP to show its use (i) get (ii) mget (iii) put (iv) mput. **[4 M]**

Ans. Refer to Section 4.3.1.2.

4. Compare POP3 with IMAP on below points:

(i) TCP port used

(ii) E-mail stored at

(iii) Time required to connect

(iv) Multiple mail boxes. **[4 M]**

Ans. Refer to Sections 4.2.4.1 and 4.2.4.2.

5. Describe HTTP response message format. **[4 M]**

Ans. Refer to Section 4.5.2.1.

Summer 2024

1. State the need of domain name system. **[2 M]**

Ans. Refer to Section 4.1.

2. State the transmission modes of FTP. **[2 M]**

Ans. Refer to Section 4.3.1.3.

3. Describe SMTP with suitable diagram. **[4 M]**

Ans. Refer to Section 4.2.3.

4. Explain the working of TELNET. **[4 M]**

Ans. Refer to Section 4.4.1.

5. Describe HTTP response message format. **[4 M]**

Ans. Refer to Section 4.5.2.1.

6. Distinguish between SMTP and POP3 protocol. **[4 M]**

Ans. Refer to Sections 4.2.3 and 4.2.4.1.

Winter 2024

1. Compare SMTP and HTTP w.r.t. (i) use of port number (ii) used for type of service. **[2 M]**

Ans. Refer to Section 4.2.3 and 4.5.2.

2. State the use of following FTP commands:
(i) mget (ii) site. **[2 M]**

Ans. Refer to Section 4.3.1.2.

3. Compare FTP and TFTP w.r.t. (i) Authentication (ii) Protocol used (iii) Ports (iv) Data Transfer. **[4 M]**

Ans. Refer to Section 4.3.

4. Explain the process of resolving the host name www.msbte.org into IP address using DNS. **[4 M]**

Ans. Refer to Section 4.1.

5. Explain the working of TELNET. **[4 M]**

Ans. Refer to Section 4.4.1.

6. Explain the operations of POP3. **[4 M]**

Ans. Refer to Section 4.2.4.1.

❖❖❖