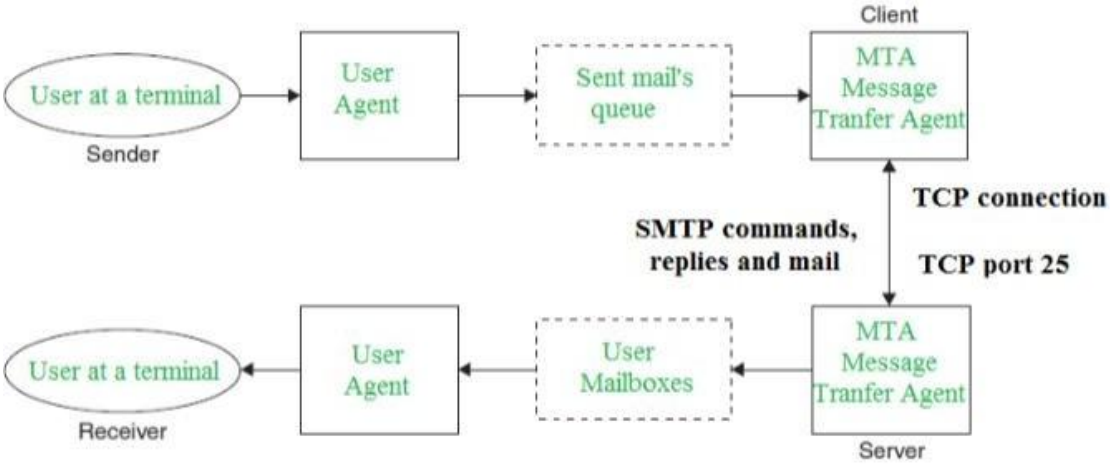
Q.1 Briefly explain the working of SMTP. (Nov-2016)

Ans:

- SMTP stands for Simple Mail Transfer Protocol. It is a TCP/IP protocol that specifies how computers exchange electronic mail. It works with post office protocol (POP).
- SMTP is used to upload mail directly from the client to an intermediate host, but only computers constantly connected such as Internet Service Providers (ISP) to the Internet can use SMTP to receive mail. The ISP servers then offload the mail to the users to whom they provide the Internet service.



Working of SMTP: SMTP is a simple ASCII protocol that is based on client-server model. After establishing the TCP connection, the sending machine, operating as the client, waits for the receiving machine, operating as the server, to talk first.

The server starts by sending a line of text giving its identity and telling whether or not it is prepared to receive mail. If it is not, the client releases the connection and tries again later. If the server is willing to accept e-mail, the client announces whom the e-mail is coming from and destination, the server gives the client the go ahead to send the message. Then the client sends the message and the server acknowledges it.

Mail Transfer Phases:

The basic SMTP operation occurs in three phases:

1. Connection set up
2. Mail transfer
3. Connection termination

Connection Setup:

An SMTP sender will attempt to set up a TCP connection with a target host when it has one or more mail message to deliver to that host. The following sequence occurs during connection setup:

1. The sender opens a TCP connection with the receiver.
2. Once the connection is established, the receiver identifies itself with '220 Service Ready".
3. The sender identifies itself with the HELO command.
4. The receiver accepts the sender's identification with "250 'OK".
5. If the mail service on the destination is not available, the destination host returns a "421 Service Not Available" reply in step 2 and the process is terminated.

Mail transfer:

- Once the connection has been established, the SMTP sender may send one or more messages to the SMTP receiver.
- There are three logical phases to the transfer of a message :
  1. A MAIL command identifies the originator of the message.
  2. One or more RCPT commands identify the recipients of this message.
  3. A DATA command transfers the message text.

Connection termination:

- The SMTP sender closes the connection in the following manner:
  1. The sender sends a QUIT command and waits for a reply.
  2. Sender initiates TCP close operation for the TCP connection.
  3. The receiver initiates its TCP close after sending is reply to the QUIT command.

Q.2 How DNS is useful in Internet? OR Write a short note on DNS OR What is the role of Domain Name Server (DNS) in Internet? OR What is the purpose of domain naming system (DNS)? (May-2016, June-2017, Nov-2017)

Ans:

- DNS is a Domain Name Server.
- It is use IP address (32 bit) - for addressing datagram
- Ex. “name”, e.g., www.yahoo.com used by humans.
- Distributed Database implemented in hierarchy of many name server.
- Application Layer Protocol is hosts, name servers communicate to resolve names (address/name translation)

- DNS uses UDP as transport layer protocol.
- DNS provide services like hostname to IP address translation, host aliasing Ex. canonical, alias names Like www.fb.com, mail server aliasing, load distribution replicated Web servers many IP addresses correspond to one name
- DNS Centralization is single point of failure, traffic volume, distant centralized database, maintenance.

DNS Working :

- Consider HTTP request message contains www.someschool.edu in host name:
  1. The same user machine runs the client side of the DNS application.

2. The browser extracts the hostname, www.someschool.edu, from the URL and passes the hostname to the client side of the DNS application.
3. The DNS client sends a query containing the hostname to a DNS server.
4. The DNS client eventually receives a reply, which includes the IP address for the hostname.
5. Once the browser receives the IP address from DNS, it can initiate a TCP connection to the HTTP server process located at port 80 at that IP address.

- DNS: a distributed, hierarchical database:
- DNS user large number of database organization hierarchical fashion and distributed around the world.
- **There are three class.**
    1. Root DNS
    2. Top-Level DNS (TLD)
    3. Authoritative DNS

Root DNS:

- In the internet there 13 Root DNS which are located in North America and each Root DNS server is actually of replicated server's.
- It is doesn't stored in IP address of the domain but it stores IP address of the TLD.
- Any local DNS server is trusted the Root DNS.

Local DNS:

- It does not strictly belong to hierarchy
- It each ISP (residential ISP, company, university) has one also called "default name server".
- when host makes DNS query, query is sent to its local DNS server has local cache of recent name-to-address translation pairs (but may be out of date!) acts as proxy, forwards query into hierarchy.

TLD :

- It is Top-Level DNS server.

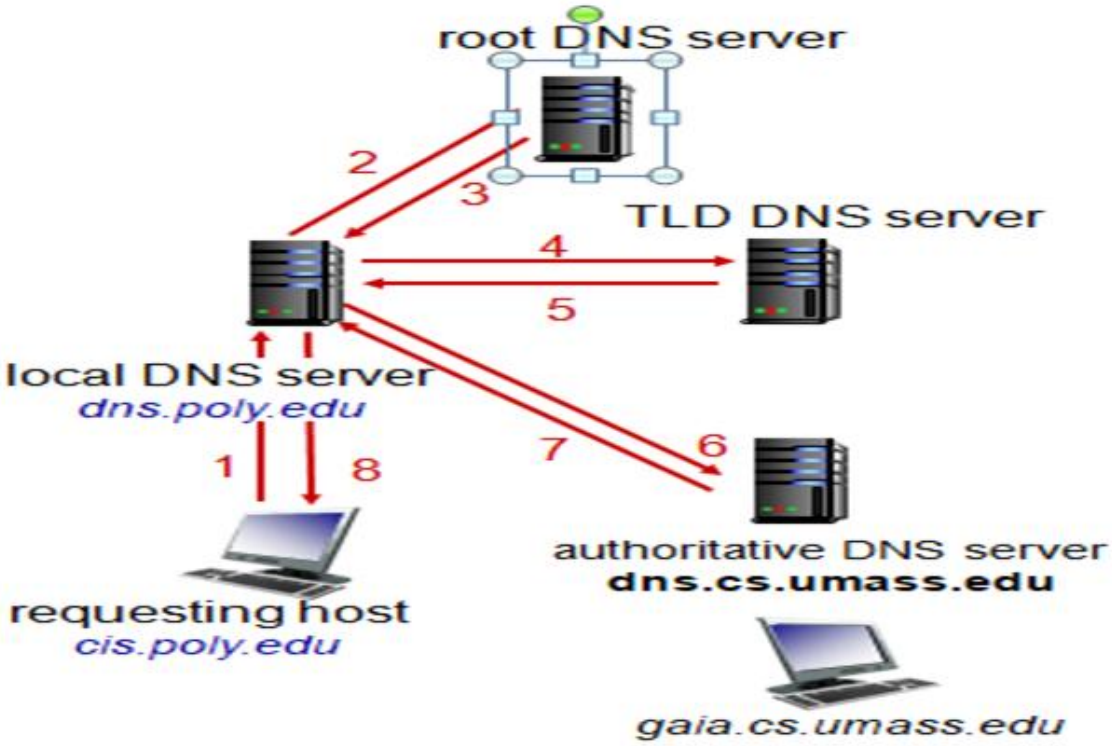
Country Domain	Full name
.com	Commercial
.edu	Education
.ind	India
.pak	Pakistan
.gov	Government
.org	Nonprofit organization
.mil	Defense (Military)

There 2 ways DNS handle Services:

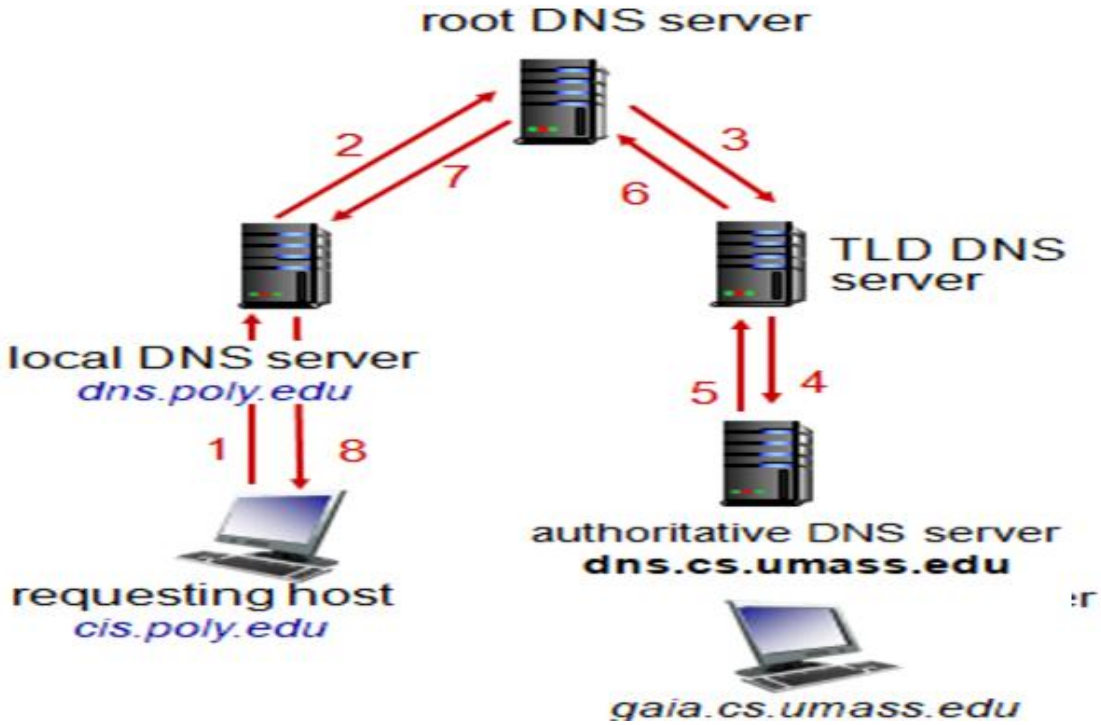
1) Iterated query:
    - Contacted server replies with name of server to contact "I don't know this name, but ask this server".
    - Local DNS sends to query to the Root DNS and it gives reply with TLD to Local DNS.
    - Local DNS sends to requests to the TLD. TLD reply with IP address of

Authoritative DNS. Lastly Local DNS contacts the Authoritative DNS. Authoritative DNS the replies with IP address of Local DNS. Finally Local DNS with provide IP address to the requesting host.

- Figure of Iterative method:



2) Recursive method:
   - Where Local DNS will request to Root DNS and Root DNS will contact to TLD on directly on behalf of Local DNS. TLD will then contact to Authoritative DNS will be forward to TLD and TLD will again forward to Root DNs.
   - Root DNS will again forward it to Local DNS server and Finally Requesting host.
   - Puts burden of name resolution on contacted name server heavy load at upper levels of hierarchy.
   - Figure of Recursive Method:



- **DNS Record:**

- DNS: distributed db storing resource records (RR)

- RR format: **(**name, value, type, ttl**)**

type=A

- Name is hostname.
- Value is IP address.

type=NS

- Name is domain (e.g., foo.com).
- Value is hostname of authoritative name server for this domain.

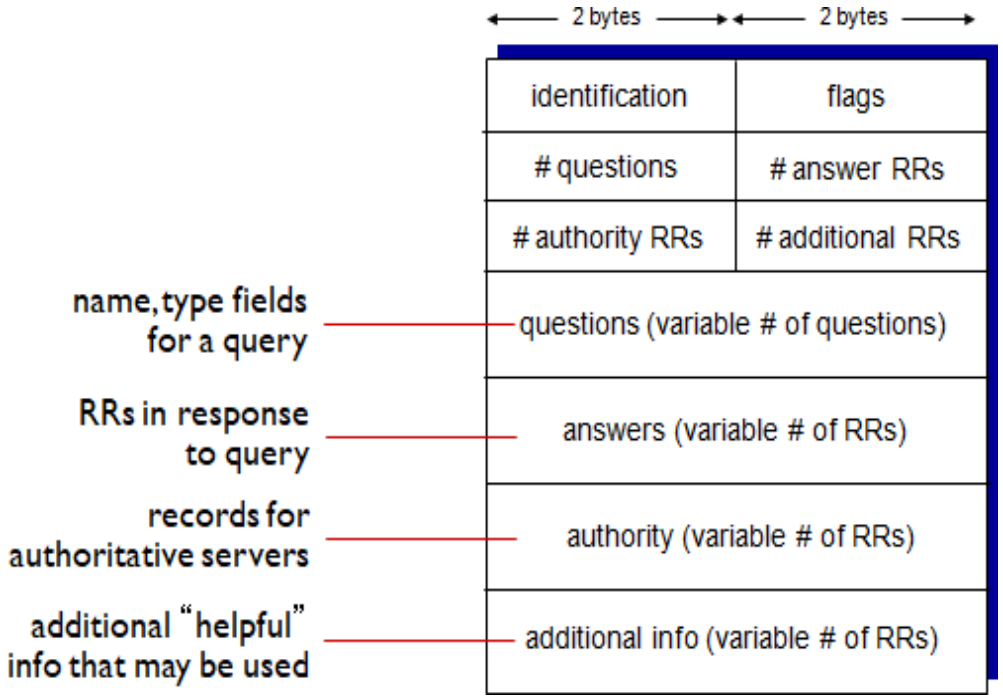
type=CNAME

- Name is alias name for some “canonical” (the real) name.
- www.ibm.com is really.
- servereast.backup2.ibm.com.
- Value is canon.

type=MX

- Value is name of mail server associated with Name

DNS Message Header:



Q.3: What is client server architecture? Discuss its merits and demerits. (Nov-2016)

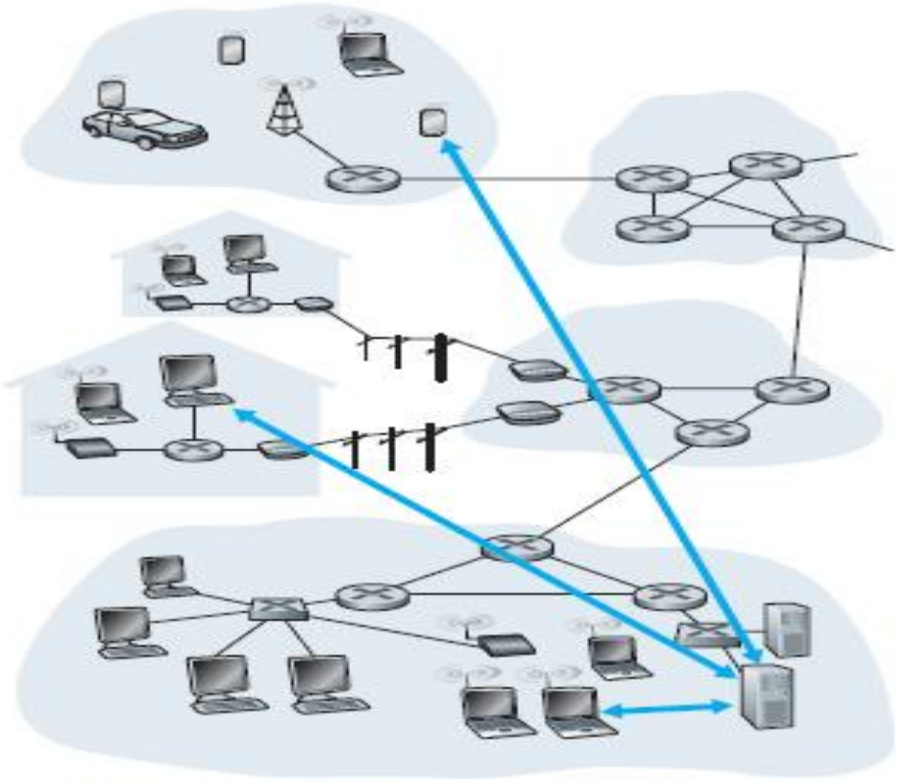
Ans:

- In a client-server architecture, there is an always-on host, called the server, which services requests from many other hosts, called clients.
- A classic example is the Web application for which an always-on Web server services requests from browsers running on client hosts.
- When a Web server receives a request for an object from a client host, it responds by sending the requested object to the client host.
- Note that with the client-server architecture, clients do not directly communicate with each other; for example, in the Web application, two browsers do not directly communicate.
- Another characteristic of the client-server architecture is that the server has a fixed, well-known address, called an IP address (which we'll discuss soon). Because the server has a fixed, well-known address, and because the server is always on, a client can always contact the server by sending a packet to the server's IP address.
- Some of the better-known applications with a client-server architecture include the Web, FTP, Telnet, and e-mail.Often in a client-server application, a single-server host is incapable of keeping upwith all the requests from clients.
- For example, a popular social-networking site can quickly become overwhelmed if it has only one server handling all of its requests. For this reason, a data center, housing a large number of hosts, is often used to create a powerful virtual server.
- The most popular Internet services—such as search engines (e.g., Google and Bing), Internet commerce (e.g., Amazon and e-Bay), Web-based email (e.g., Gmail and Yahoo Mail), social networking (e.g., Facebook and Twitter)— employ one or more data centers., Google has 30 to 50data centers distributed around the world, which collectively handle search, YouTube,Gmail, and other services.
- A data center can have hundreds of thousands of servers,which must be powered and maintained. Additionally, the service providers must payrecurring interconnection and

bandwidth costs for sending data from their data centers.

- **Definition:**
- The term ‘client-server’ refers to the network architecture. Where one or more computers are connected a server. That one computer (the client) or more sends a service request to another computer(the server).

- **Merits of client-server Architecture :**
- Improved Data sharing.
- Easy Maintenance.
- Increased security.
- Integration of Services.
- Data Processing capability despite of location.
- Best price/performance on Clients machines.
- Resources are centralized on a server – easier to find and to distribute.
-
- **Demerits of client-server Architecture :**
- More expensive to implement.
- Single point of failure.
- The more clients there are the more crowded the server and response time is longer.
- Require more complex setup and Management.
- Overloaded servers.
- Impact of centralized architecture.



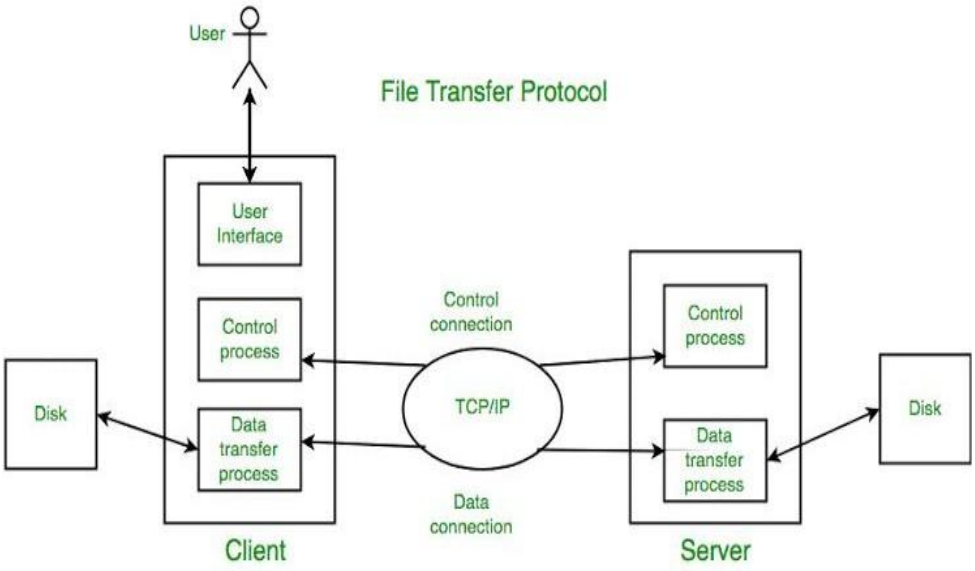
a. Client-server architecture

Q.4 Explain the movement of file between local and remote system using FTP. (Nov 2016)

Ans:

- File Transfer Protocol (FTP) is the commonly used protocol for exchanging files over the Network or Internet.
- FTP uses the Internet's TCP/IP protocols to enable data transfer. FTP uses client-server architecture. FTP promotes sharing of files via remote computers with reliable and efficient data transfer.
- Fig. shows FTP moves files between local and remote file systems In above figure user interacts with FTP through an FTP user agent.
- The user first provides the hostname of the remote host, causing the FTP client process in the local host to establish a TCP connection with the FTP server process in the remote host.
- The user then provides the user identification and password, which are sent over the TCP connection as part of FTP commands.
- Once the server has authorized the user, the user copies one or more files stored in the local file system into the remote file system (or vice versa). FTP uses two parallel TCP connections to transfer a file,

  - Control connection
  - Data connection



Q.5 Consider the following HTTP message and answer the following questions. (May-2016)

1) **From which browser URL is requested.**
2) **Does browser request a non-persistent or a persistent connection.**
3) **Which is the URL of the document requested by the user.**
4) **Which HTML method is used to receive the requested URL.**

GET/home.asp HTTP/1.1
Host:gtu.ac.in
Accept-encoding:gzip,deflate,sdch

Accept-language:en-us,en;q=0.8
Cookie:OGPC=5061921-11:5061952-13:5061985-24:5061983-27:5061068:137:5062009-6:5062022-12:;
SID=DQAAALgBAAA3RAje9UILOOSuH0G91uzL5JOJNUYU2aVOml6jEV
User Agent:chrome/49.0.2623.110
X-client-data:CIS2yQEIprbJAQjDtskBCP2VygE
Connection:keep-alive

Ans. 1) chrome 2) persistent connection 3) gtu.ac.in 4) GET

Q.6 Explain the way of DNS servers to handle the recursive DNS query using suitable diagram. (May 2015)
Ans:

- The basic concept of DNS name resolution is fairly simple. Every Web site is assigned a unique IP address. In order to access a website, a client needs to know what the site's IP address is. Of course users don't usually enter an IP address into their Web browser, but rather enter the site's domain name instead. In order to access the requested website, the Web browser must be able to convert the site's domain name into the corresponding IP address. This is where DNS comes into play. The client computer is configured with the address of a preferred DNS server. The requested URL is forwarded to the DNS server, and the DNS server returns the IP address for the requested website. The client is then able to access the requested site.
- As you can see, the name resolution process is pretty cut and dry. However, there are countless websites in the world, and new sites are being created every day. It is impossible for your DNS server to know the IP address of every single website. When a DNS server does not know the address for a requested site, it uses one of two methods to determine the site's IP address.
- The preferred name resolution method is called recursion. Generally speaking, recursion refers to the process of having the DNS server itself to make queries to other DNS servers on behalf of the client who made the original request. In essence, the DNS server becomes a DNS client. Some administrators prefer to disable recursion for performance reasons. If recursion is disabled, then the DNS server uses a process called iteration to resolve the name request.

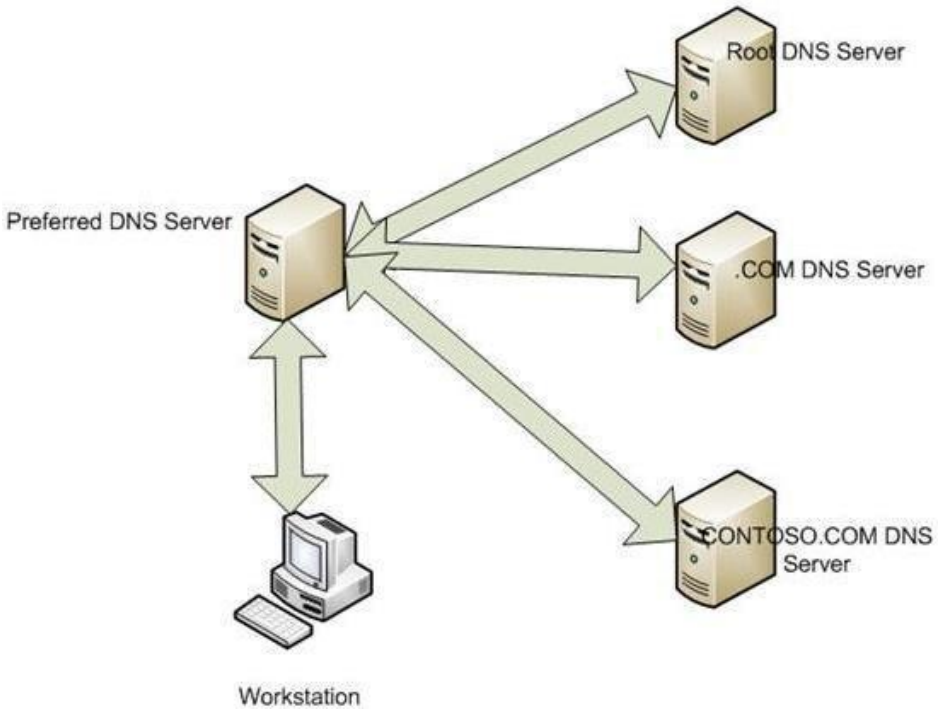


Figure A: This is how DNS recursion works

- The process begins when the user enters a URL into their Web browser. For the purpose of this example, let's assume that the user has entered www.contoso.com as the URL. Upon doing so, the request to resolve the Contoso.com domain into an IP address is passed to the workstation's preferred DNS server. Often times the preferred DNS server will have already cached the requested record, but for the sake of this example, let's assume that the preferred DNS server has no information related to CONTOSO.COM.
- Assuming that DNS recursion is enabled, the DNS server begins acting as a DNS client and launches a series or iterative queries against other DNS servers. I will discuss the difference between iterative and recursive queries later on, but for now just realize that the process as a whole is considered to be recursive because the client only makes one request to the preferred DNS server.
- At any rate, the workstation's preferred DNS server doesn't know the IP address of the www.contoso.com Web site, and it doesn't know the IP address of a DNS server that is authoritative for the Contoso.com domain (and would therefore know the IP address of the www.contoso.com Web site). What the DNS server does know is the IP address to a root level DNS server (thanks to the root hints file). Therefore, the preferred DNS server forwards the request to the root DNS server.
- The root DNS server doesn't have a clue as to the IP address of the www.contoso.com Web server. What it does know is the IP address of a DNS server that is responsible for the .COM domain. The root DNS server returns the IP address of the DNS server responsible for the .COM domain to the preferred DNS server. The preferred DNS server then sends the client's request to the .COM DNS server. The .COM DNS server doesn't know the IP address of the www.contoso.com Web site, but it does know the IP address of the DNS server that is authoritative for the Contoso.com domain. The .com domain server returns the IP address of the DNS server that is authoritative for

the Contoso.com domain. The client's preferred DNS server then sends the request to the Contoso.com DNS server, which in turn returns the IP address for the requested Web site. This address is then returned to the client who requested it.

- There are two things worth noting in this example. First, as I explained earlier, the client only made a single DNS query. It was completely unaware of the DNS server's iterative queries on its behalf. Second, the DNS server that is authoritative for the CONTOSO.COM domain would not necessarily be owned by Contoso. Typically, this DNS server would be owned by a Web hosting company and would be authoritative for any sites hosted by the company. That's why the preferred DNS server can't skip a step and just give the client the address for the DNS server that is authoritative for the domain; at least not in this case.
- If a DNS server is configured to not support recursive queries, then clients will perform iterative queries by default.
- If you are interested in obtaining the best performance, then you should configure your DNS server to allow recursive queries. The reason is because if clients are forced to use iterative queries, then they could potentially issue three or four queries to the DNS server for every name resolution request. The DNS server will have to perform all these queries whether recursive or iterative queries are being used, but when recursion is used, most of the name resolution requests are handled by your DNS server and are kept off of your network. This reduces the amount of traffic flowing across the network, thereby improving performance.

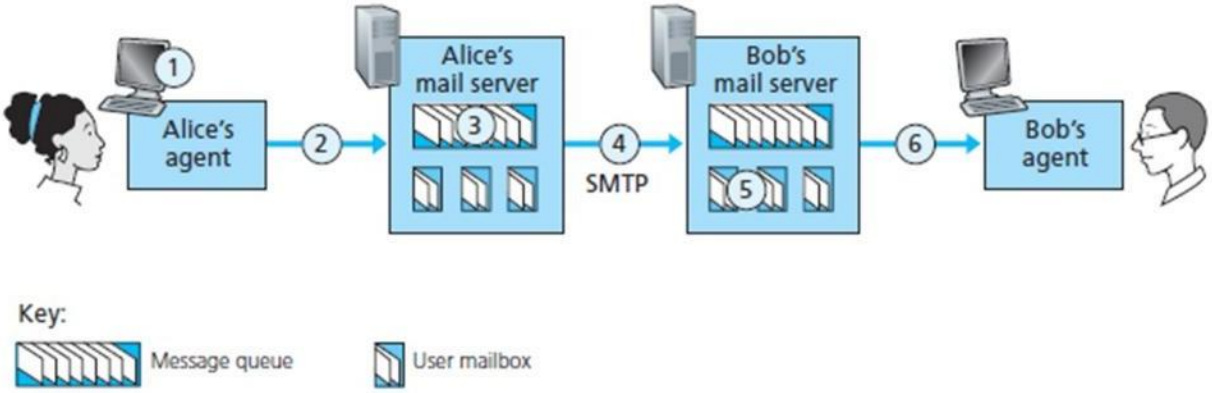
Q.7 Explain working of electronic mail protocols SMTP, IMAP and POP3 in brief with suitable diagram. (May-2015)

Ans:

SMTP (Simple Mail Transfer Protocol)

- SMTP transfers messages from senders' mail servers to the recipients' mail servers.
- It restricts the body (not just the headers) of all mail messages to simple 7-bit ASCII.

(Alice send a message to Bob)



- To illustrate the basic operation of SMTP, let's take a common scenario. Suppose Alice wants to send Bob a simple ASCII message.
  1. Alice invokes her user agent for e-mail, provides Bob's e-mail address (for example, bob@someschool.edu), composes a message and instructs the user agent to send the message.
  2. Alice's user agent sends the message to her mail server, where it is placed in a message queue.

3. The client side of SMTP, running on Alice's mail server, sees the message in the message queue. It opens a TCP connection to an SMTP server, running on Bob's mail server.
4. After some initial SMTP handshaking, the SMTP client sends Alice's message into the TCP connection.
5. At Bob's mail server, the server side of SMTP receives the message. Bob's mail server then places the message in Bob's mailbox.
6. Bob invokes his user agent to read the message at his convenience.

- SMTP does not normally use intermediate mail servers for sending mail, even when the two mail servers are located at opposite ends of the world.
- If Bob's mail server is down, the message remains in Alice's mail server and waits for a new attempt and the message does not get placed in some intermediate mail server.

How SMTP transfers a message from a sending mail server to a receiving mail server

- First, the client SMTP (running on the sending mail server host) has TCP establish a connection to port 25 at the server SMTP (running on the receiving mail server host).
  - If the server is down, the client tries again later.
  - Once this connection is established, the server and client perform some application-layer handshaking, just as humans often introduce themselves before transferring information from one to another.
  - During this SMTP handshaking phase, the SMTP client indicates the e-mail address of the sender (the person who generated the message) and the e-mail address of the recipient.
  - Once the SMTP client and server have introduced themselves to each other, the client sends the message.
  - SMTP can count on the reliable data transfer service of TCP to get the message to the server without errors.
  - The client then repeats this process over the same TCP connection if it has other messages to send to the server; otherwise, it instructs TCP to close the connection.

Mail Access Protocols (POP3 and IMAP)

POP3 (Post Office Protocol)

- POP3 is an extremely simple mail access protocol.
- POP3 begins when the user agent of the client opens a TCP connection to the mail server of the server on port 110.
- With the TCP connection established, POP3 progresses through three phases: authorization, transaction and update.
- During the first phase, authorization, the user agent sends a username and a password to authenticate the user.
- During the second phase, transaction, the user agent retrieves messages; also during this phase, the user agent can mark messages for deletion, remove deletion marks and obtain mail statistics.
- The third phase, update, occurs after the client has issued the quit command, ending the POP3 session; at this time, the mail server deletes the

messages that were marked for deletion.

- POP3 is designed to delete mail on the server as soon as the user has downloaded it.
- However, some implementations allow users or an administrator to specify that mail be saved for some period of time. POP can be thought of as a "store-and-forward" service.
- With the TCP connection established, POP3 progresses through three phases: authorization, transaction and update.
- During the first phase, authorization, the user agent sends a username and a password to authenticate the user.
- During the second phase, transaction, the user agent retrieves messages; also during this phase, the user agent can mark messages for deletion, remove deletion marks and obtain mail statistics.
- The third phase, update, occurs after the client has issued the quit command, ending the POP3 session; at this time, the mail server deletes the messages that were marked for deletion.
- POP3 is designed to delete mail on the server as soon as the user has downloaded it.
- However, some implementations allow users or an administrator to specify that mail be saved for some period of time. POP can be thought of as a "store-and-forward" service.

IMAP (Internet Message Access Protocol)

- With POP3 access, once receiver has downloaded his messages to the local machine, he can create mail folders and move the downloaded messages into the folders.
- Receiver can then delete messages, move messages across folders, and search for messages (by sender name or subject).
- But this paradigm—namely, folders and messages in the local machine—poses a problem for the roaming user, who would prefer to maintain a folder hierarchy on a remote server that can be accessed from any computer.
- This is not possible with POP3—the POP3 protocol does not provide any means for a user to create remote folders and assign messages to folders.
- An IMAP server will associate each message with a folder; when a message first arrives at the server, it is associated with the recipient's INBOX folder.
- The recipient can then move the message into a new, user-created folder, read the message, delete the message, and so on.
- The IMAP protocol provides commands to allow users to create folders and move messages from one folder to another.
- IMAP also provides commands that allow users to search remote folders for messages matching specific criteria.
- Another important feature of IMAP is that it has commands that permit a user agent to obtain components of messages. For example, a user agent can obtain just the message header of a message or just one part of a multipart MIME message.
- This feature is useful when there is a low-bandwidth connection (for example, a slow-speed modem link) between the user agent and its mail server.

- With a low bandwidth connection, the user may not want to download all of the messages in its mailbox, particularly avoiding long messages that might contain, for example, an audio or video clip.

Q.8 What is HTTP? Differentiate its persistent and non- persistent types with request-response behavior of HTTP. (DEC-2015, Nov-2016, Nov-2107)

Ans:

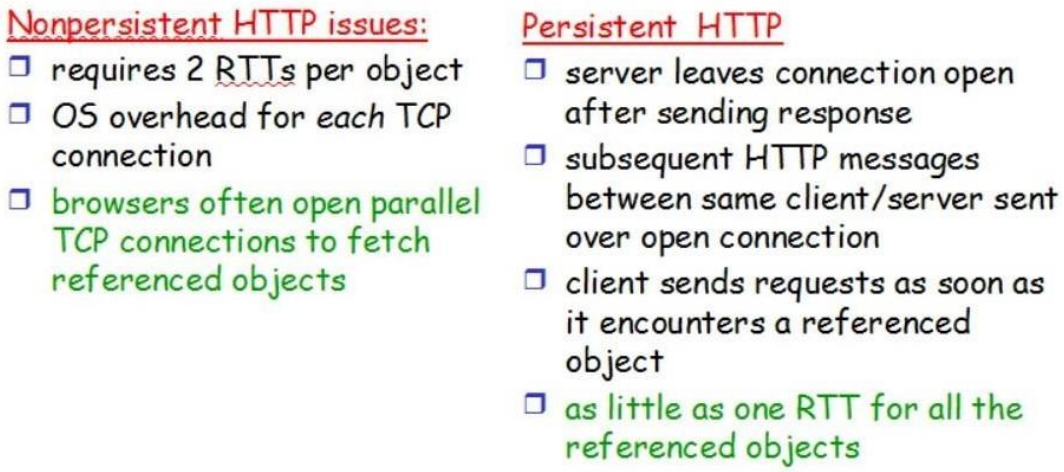
HTTP

- The Hypertext Transfer Protocol (HTTP) is an application-level protocol that uses TCP as an underlying transport and typically runs on port 80, HyperText Transfer Protocol HTTP is an application protocol for distributed, collaborative, and hypermedia information systems. 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 various commands to respond.HTTP is a stateless protocol i.e. server maintains no information about past client requests.

HTTP Connections

- Non-Persistent
- Persistent

Difference between Persistent & Non-Persistent connection.



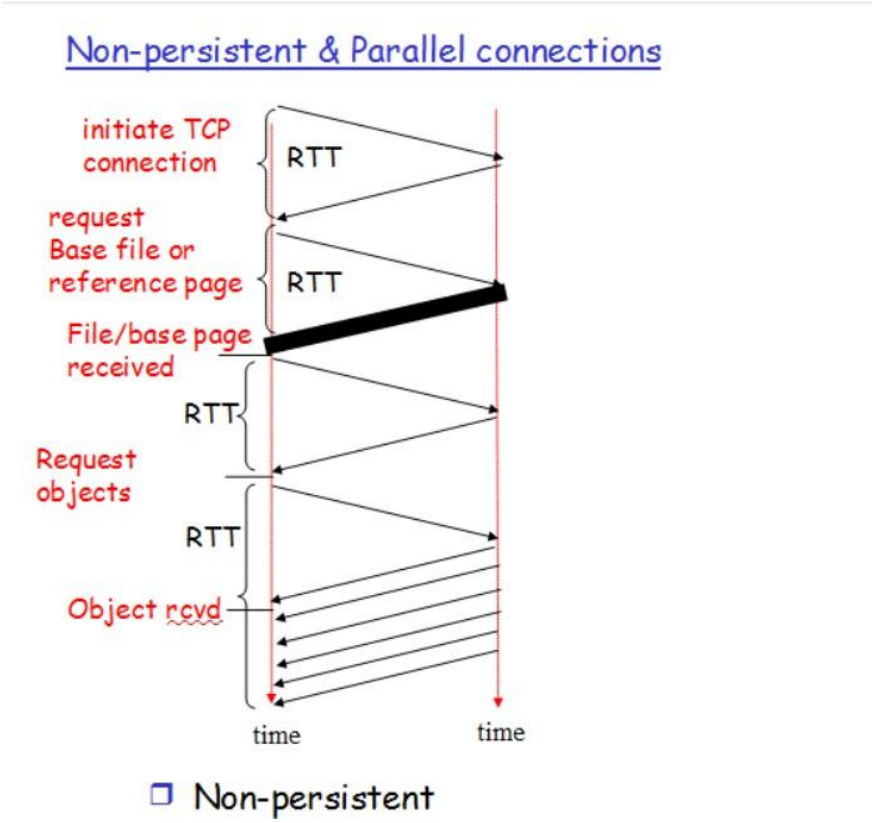
Non-Persistent Connection

- non - persistent A nonpersistent connection is the one that is closed. After the server sends the request Nonpersistent connections are the default made for HTTP / 1. 0
  - Without parallel connection
  - With parallel connection

Without parallel connection Non-Persistent

- Each objection takes two RTT (assuming no window limit) one for TCP connection and other for HTTP image/text file.

With parallel connection Non-Persistent



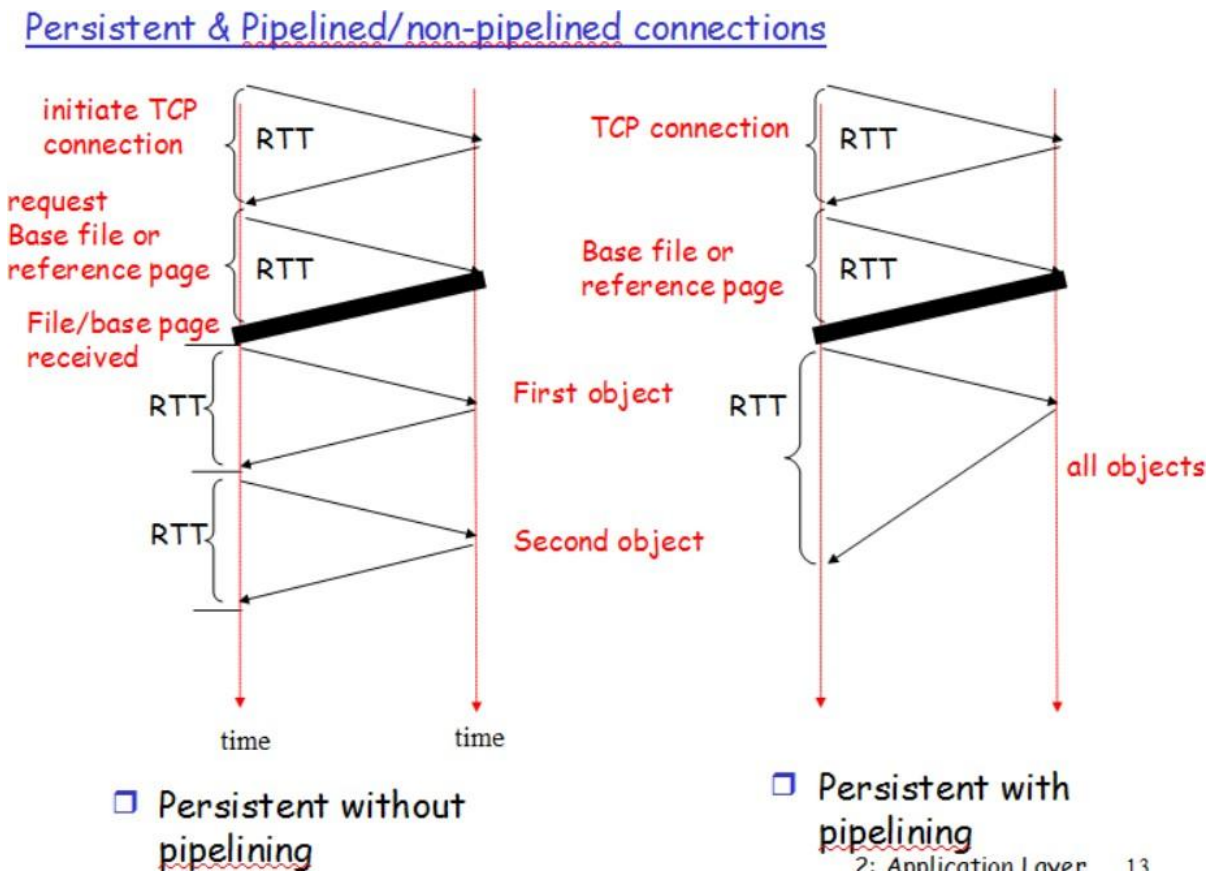
❒ Non-persistent

Persistent connection

- Persistent connections, the server closes the TCP connection and then the same client and server between the subsequent requests and responses sent to the server. The connection is only when the configurable amount of time is used for the server.
- With persistent connections, the performance is improved by 20%. persistent connections are the default mode for HTTP / 1 1.

 Non-Pipelined

 Pipelined



- In Non-pipeline connection we first establish connection which takes two RTT then we send all the objects images/text files which takes 1 RTT each (TCP for each object is not required).
- In Pipelined connection 2RTT for connection establishment and then 1RTT(assuming no window limit) for all the objects i.e. images/text.

Advantages of persistent connections:

- Lower CPU and memory usage because there are less number of connections.
- Allows HTTP pipelining of requests and responses.
- Reduced network congestion (fewer TCP connections).
- Reduced latency in subsequent requests (no handshaking).
- Errors can be reported without the penalty of closing the TCP connection.

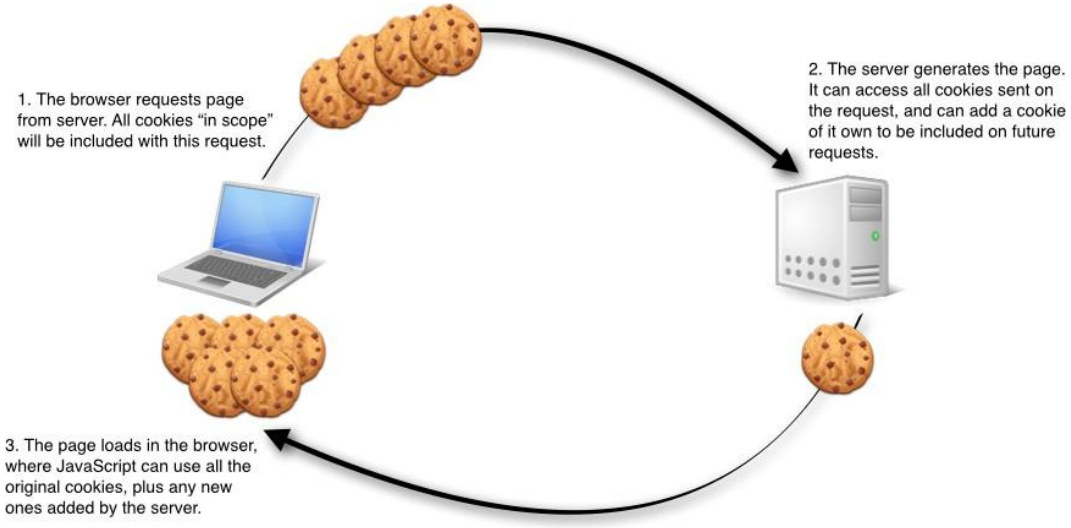
Disadvantages of persistent connections:

- Resources may be be kept occupied even when not needed and may not be available to others.

- Most of the modern browsers like Chrome, Firefox and Internet Explorer use persistent connections.

Q.9 Explain Concept of cookies and its components with suitable example. (Dec-2015)

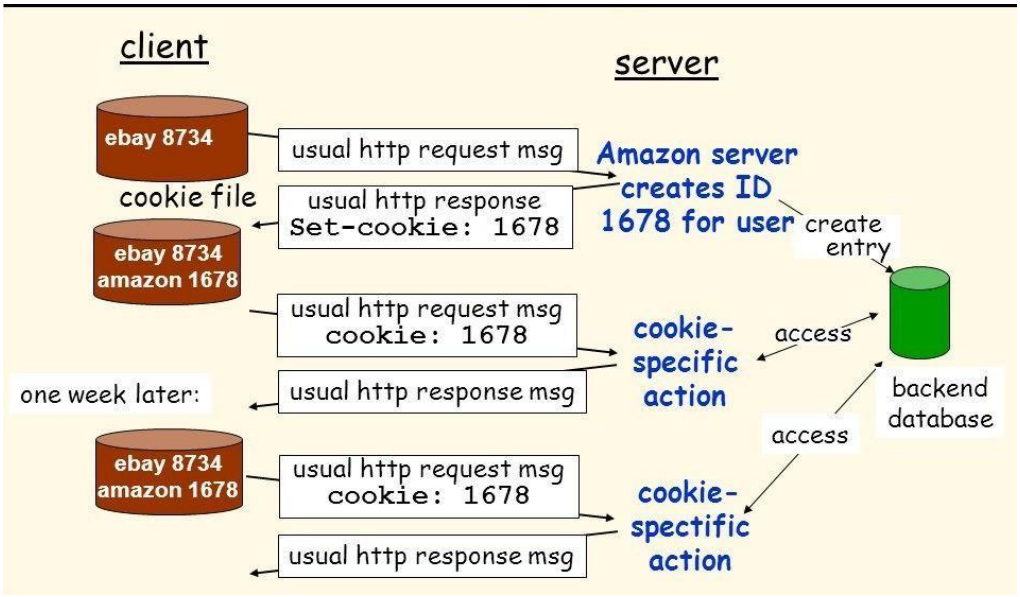
Ans:



- A small text file created by a Web site that is stored in the user's computer either temporarily for that session only or permanently on the hard disk (persistent cookie).
- Cookies provide a way for the Web site to recognize you and keep track of your preferences.

Cookie technology has four components

- a cookie header line in the HTTP response message
- a cookie header line in the HTTP request message
- a cookie file kept on the user's end system and managed by the user's browser
- a back-end database at the Web site



- Suppose Dhaval, access Amazon.com for the first time.
- Let us suppose that in the past she has already visited the eBay site.
- When the request comes into the Amazon Web server, the server creates a unique identification number and creates an entry in its back-end database by the identification number.
- The Amazon Web server then responds to Dhaval's browser, including in the HTTP response a set-cookie: header, which contains the identification number.
- For example, the header line might be Set-cookie: 1678.
- When Dhaval's browser receives the HTTP response message, it sees the Set- cookie: header.

The browser then appends a line to the special cookie file that it manages.

- This line includes the hostname of the server and the identification number in the Set-cookie: header.
- The cookie file already has an entry for eBay, since Dhaval has visited that site in the past.
- As Dhaval continues to browse the Amazon site, each time she requests a Web page, her browser consults her cookie file, extracts her identification number for this site, and puts a cookie header line that includes the identification number in the HTTP request.
- Specifically, each of her HTTP requests to the Amazon server includes the header line: Cookie:1678.
- In this manner, the Amazon server is able to track Dhaval's activity at the Amazon site.

Use of cookies:

- authorization
- recommendations
- shopping carts
- user session state (Web e-mail)

Q.9 Explain the high-level view of Internet e-mail system and its major components. (Dec-2015)

Ans:

- Along with the Web, electronic mail is one of the most popular Internet applications. Just like ordinary "snail mail," email is asynchronous – people send and read messages when it is convenient for them, without having to coordinate with other peoples' schedules.
- In contrast with snail mail, electronic mail is fast, easy to distribute, and inexpensive. Moreover, modern electronic mail messages can include hyperlinks, HTML formatted text, images, sound and even video.
- In this section we will examine the application-layer protocols that are at the heart of Internet electronic mail. But before we jump into an in-depth discussion of these protocols, let's take a bird's eye view of the Internet mail system and its key components.

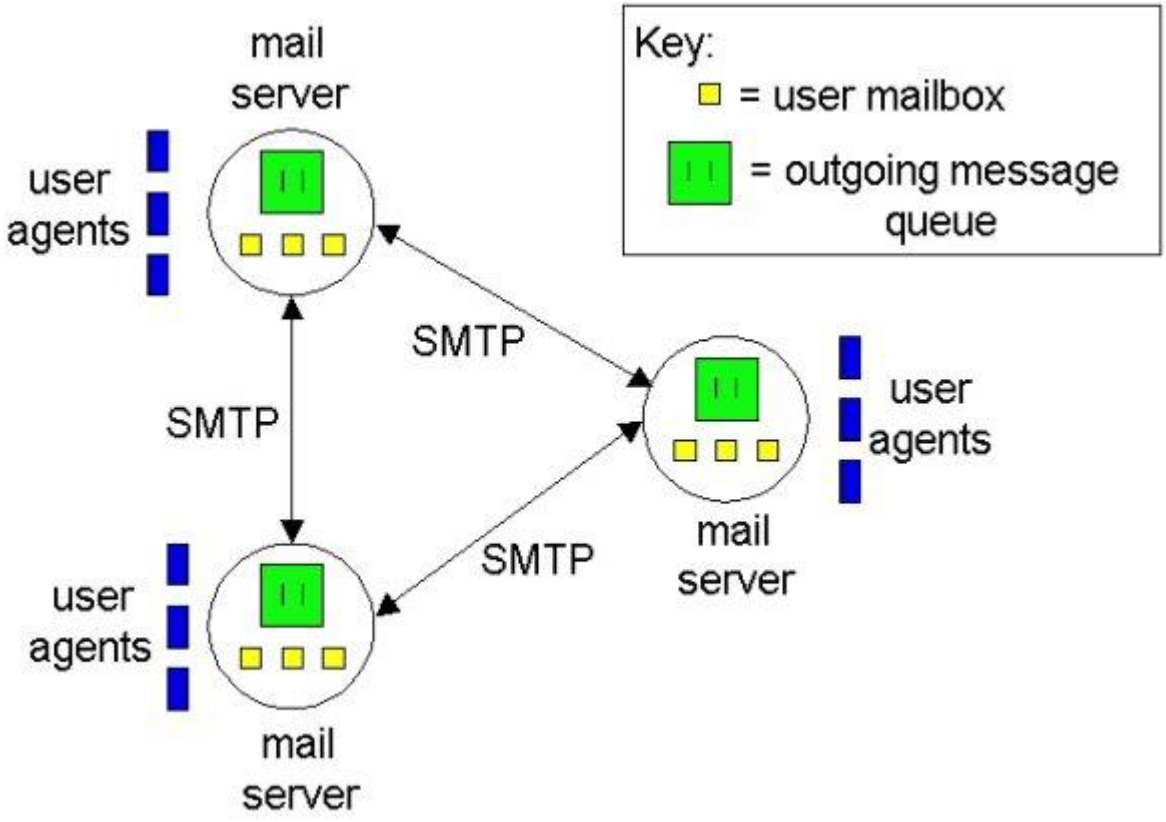


Figure 2.4-1: A bird's eye view of the Internet e-mail system.

- Figure 2.4-1 presents a high-level view of the Internet mail system. We see from this diagram that it has three major components: **user agents**, **mail servers**, and the **Simple Mail Transfer Protocol (SMTP)**.
- We now describe each of these components in the context of a sender, Alice , sending an email message to a recipient, Bob . User agents allow users to read, reply to, forward, save, and compose messages. (User agents for electronic mail are sometimes called *mail readers*, although we will generally avoid this term in this book).
- When Alice is finished composing her message, her user agent sends the message to her mail server, where the message is placed in the mail server's outgoing message queue.
- When Bob wants to read a message, his user agent obtains the message from his mailbox in his mail server.
- In the late 1990s, GUI (graphical user interface) user agents became popular, allowing users to view and compose multimedia messages.
- Currently, Eudora, Microsoft's Outlook Express, and Netscape's Messenger are among the popular GUI user agents for email. There are also many text-based email user interfaces in the public domain, including mail.
- Mail servers form the core of the e-mail infrastructure. Each recipient, such as Bob, has a **mailbox** located in one of the mail servers.

- Bob'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 then travels to the recipient's mail server, where it is deposited in the recipient's mailbox.
- When Bob wants to access the messages in his mailbox, the mail server containing the mailbox authenticates Bob (with user names and passwords). Alice's mail server must also deal with failures in Bob's mail server.
- If Alice's server cannot deliver mail to Bob's server, Alice'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 (Alice) with an email messaage.