CN Assignment I

1. **With an example explain how cookies are used in e-commerce application to improve the performance.**

   In an e-commerce application, when the user sends a login form to the server, the server authorizes the credentials, which if correct – the server stores information about this session of the user (username, password, session_id, session_start, user_type, …); and sends a *cookie* data to the user which contains information about this session file.

   The cookie data from the server is stored in the browser. For every simultaneous requests made to the server, the browser sends this cookie instead of all the authentication information, so that the server knows which session is to be retrieved from the server's database. If the session information 'expires', or 'fails to authenticate', the server asks for re-login.

   Cookie data can be anything (not just login information) that the server stores and sends a ID to identify that store.

   Cookies prevent the browser to send multiple authentication calls before each request is made to the server, thus decreasing the amount of network calls, and server side processing hence increasing the performance. It also prevents the sending of password, and other critical user information more than once, reducing sniffing risks.

   For example, take Amazon
   - User logs in with his account credentials.
   - Server authenticates it and creates a 'session' on server-side.
   - Server sends a 'cookie' with the session data to the browser.
   - The browser stores the cookie, and sends it with future requests to the server.
   - The user adds some stuff to the cart, the server stores this information, and sends back a cookie with the cart ID.
   - The user uses this cookie to shop and checkout.
   - When the user logs off, the cookie data is deleted on the server, and from the browser.

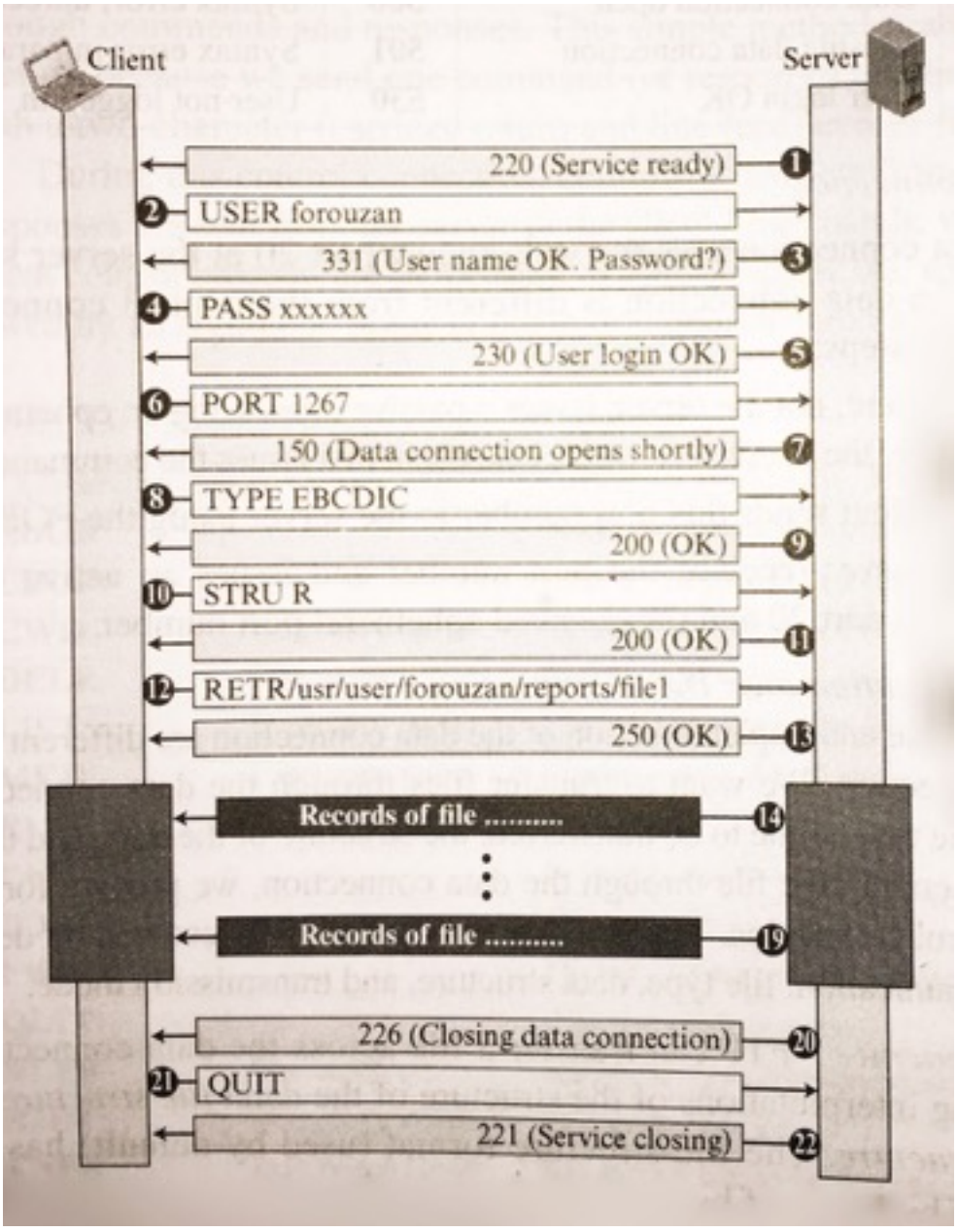
2. **With a diagram explain how commands are exchanged between FTP client and FTP server during retrieval of a file.**

   During retrieval of a file, the following things happen:
   1. The FTP client pings the server, which if available, responds with a **220** code (service ready).
   2. The client sends the username to the server. (**USER** method)
   3. The server checks the validity of the username, and asks the user for the password.
   4. The client sends the password to the server. (**PASS** method)

5. The server authenticates the password, and on success sends a login-ok response (**230**)
6. Data connection uses port 20 on the server side; The client issues a passive open using a ephemeral port, and sends this port information to the server.
7. The server issues a active open response using the port 20 and the received port.
8. The client sends the default type of the files to the server. (**TYPE**)
9. Server says OK (**200**).
10. The client defines the data file organization to the server (**STRU**)
11. Server says OK (**200**).
12. The client asks the server to retrieve a file. (**RETR**)
13. The server responds with OK (**250**) if the file is found, followed by the file's contents.

14. … 19. File contents

20. After the file's contents end, the server sends a closing data connection – telling the client that file has finished transfer.
21. The client may send a QUIT to the server to close the service.
22. The server responds with **221** (Service closing) and closes the connection.



3. With a diagram explain the different phases during transfer of a mail message.

A mail message transfer occurs in three phases – **connection establishment**, **mail transfer**, and **connection termination**.

Connection Establishment:

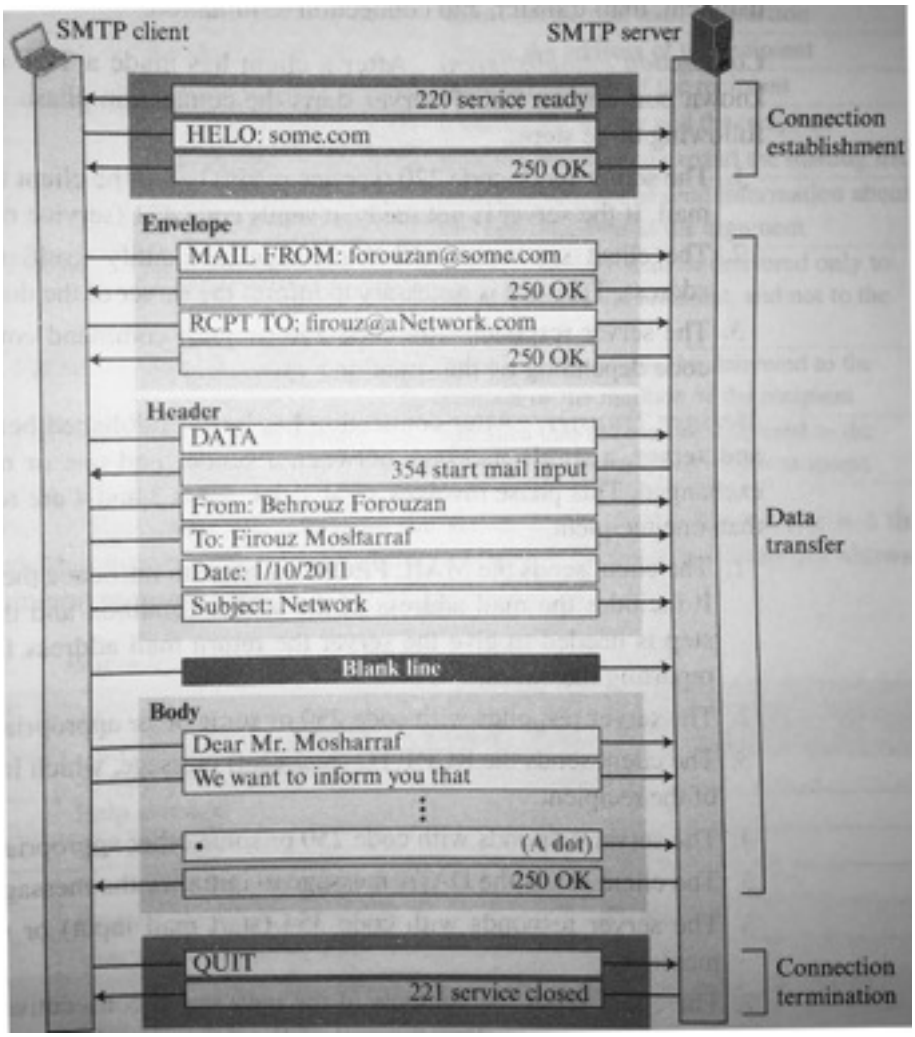
After the client has made a TCP connection to the well known port 25; the SMTP server starts the connection phase.

- The server tells the client that it's ready to receive mail by sending a code (220).
- The client sends the HELO message to identify itself, using the domain name address.
- The server verifies the client identity, and sends appropriate response.

Message Transfer:

After the connection has been established between the SMTP client and the server; a single message between the sender and the recipients can be exchanged.

- The client sends the MAIL FROM to introduce the server to the sender.
- The server responds accordingly.
- The client sends the RCPT TO to tell the server about the recipients.
- Server validates the recipients.



- The client sends the DATA message to the server to initialize the transfer.
- The server responds with some some code (354) to let the client know to start data transfer.
- The client sends the contents of the message in consecutive lines. Each line is terminated by a two character end of line token. Message is terminated with a line containing a single period.
- The server responds with OK or something.

Connection Termination:

After the message is transferred successfully, the client terminates the connection.

- The client sends the QUIT command.
- The server responds with some code acknowledging it.

4. With a diagram explain the three components of SSH.

SSH or Secure Shell is a secure application program that can be used for purposes such as remote logging and file transfer.

Components of SSH include Transport-Layer Protocol (**SSH-TRANS**), Authentication Protocol (**SSH-AUTH**) and Connection Protocol (**SSH-CONN**).

SSH Transport Layer Protocol (SSH TRANS):

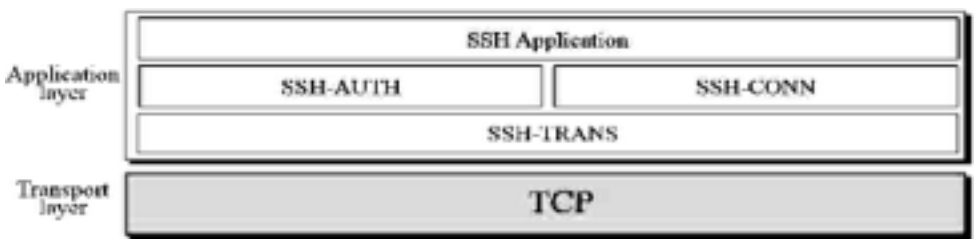
Since TCP is not secured; SSH uses the first protocol that creates a secured channel on top of the TCP. This new layer is an independent protocol referred to as SSH-TRANS. When the procedure implementing this protocol is called, it first connects insecurely on TCP, then it uses several security parameters to establish a secure channel on top of TCP. This protocol authenticates the server for the client.

SSH Authentication Protocol (SSH AUTH):

After a secure channel is established between the client and the server, and the server is authenticated for the client; SSH can call another procedure which can authenticate the client for the server. Client sends a request message to the server, including client ID, server name, method of authentication, and required data. The server in turn verifies this data with either a success or a failure message (repeat).

SSH Connection Protocol (SSH CONN):

After both the server and client are authenticated for each other, SSH can call a piece of software that implements SSH-CONN which provides various services such as multiplexing. SSH-CONN takes the secured channel established by the above two layers and lets the client create multiple logical channels over it, where each channel can be used for a different purpose such as remote logging, file transfer and so on.



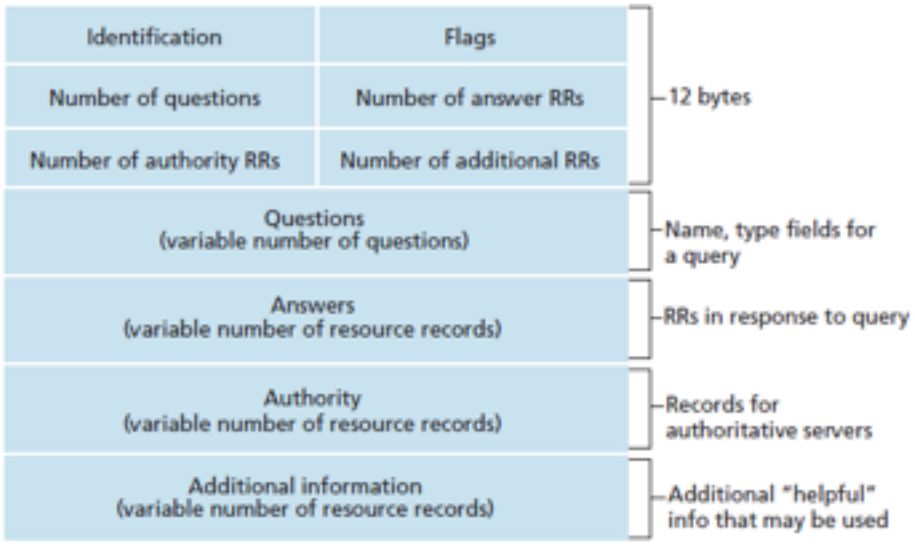
5. With a diagram explain the DNS message format.

The Domain Name System (DNS) is a directory lookup service that provides a mapping between the name of a host on the Internet and its numerical address. DNS is essential to the functioning of the Internet.

The IP address of a resource uniquely identifies the resource. This numerical naming scheme is however not convenient for users. A DNS server maps the domain name to an IP address.

To retrieve information about the hosts, the DNS server uses two types of messages, *query* and *response*.

The *query* message only contains the question, and the *response* message may contain all of the following fields, with the copy of the question.
The **identification** field is used by the client to match the response with the query.



The **flag** field determines whether the message is query or response, with error code if any.

Four header fields: Questions, Answers, Authority and Additional, define each type of messages.

The **question** contains one or more query records, sent in the query messages, and repeated back in the response.

The **answer** contains one or more resource records, only returned in response.

The **authority** section gives information about the domain name about authoritative servers for the query.

Additional information might contain some information to help the resolver.