**Name: Susmit Jaiswal Class 6th-C 0103CS171169**

**Q1. Explain HTTP protocol with its header format and features.**

Ans-The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. This is the foundation for data communication for the World Wide Web (i.e. internet) since 1990. HTTP is a generic and stateless protocol which can be used for other purposes as well using extensions of its request methods, error codes, and headers.

Basically, HTTP is a TCP/IP based communication protocol, that is used to deliver data (HTML files, image files, query results, etc.) on the World Wide Web. The default port is TCP 80, but other ports can be used as well. It provides a standardized way for computers to communicate with each other. HTTP specification specifies how clients' request data will be constructed and sent to the server, and how the servers respond to these requests.

Basic Features

There are three basic features that make HTTP a simple but powerful protocol:

* HTTP is connectionless: The HTTP client, i.e., a browser initiates an HTTP request and after a request is made, the client waits for the response. The server processes the request and sends a response back after which client disconnect the connection. So client and server knows about each other during current request and response only. Further requests are made on new connection like client and server are new to each other.
* HTTP is media independent: It means, any type of data can be sent by HTTP as long as both the client and the server know how to handle the data content. It is required for the client as well as the server to specify the content type using appropriate MIME-type.
* HTTP is stateless: As mentioned above, HTTP is connectionless and it is a direct result of HTTP being a stateless protocol. The server and client are aware of each other only during a current request. Afterwards, both of them forget about each other. Due to this nature of the protocol, neither the client nor the browser can retain information between different requests across the web pages.

HTTP headers let the client and the server pass additional information with an HTTP request or response. An HTTP header consists of its case-insensitive name followed by a colon (:), then by its value. Whitespace before the value is ignored

Headers can be grouped according to their contexts:

* General headers apply to both requests and responses, but with no relation to the data transmitted in the body.
* Request headers contain more information about the resource to be fetched, or about the client requesting the resource.
* Response headers hold additional information about the response, like its location or about the server providing it.
* Entity headers contain information about the body of the resource, like its content length or MIME type.

Headers can also be grouped according to how proxies handle them:

* Connection
* Keep-Alive
* Proxy-Authenticate
* Proxy-Authorization
* TE
* Trailer
* Transfer-Encoding
* Upgrade.

End-to-end headers

These headers *must* be transmitted to the final recipient of the message: the server for a request, or the client for a response. Intermediate proxies must retransmit these headers unmodified and caches must store them.

Hop-by-hop headers

These headers are meaningful only for a single transport-level connection, and *must not* be retransmitted by proxies or cached. Note that only hop-by-hop headers may be set using the Connection general header.

**Q2. Discuss FTP protocol with its header format and salient points.**

Ans->File Transfer Protocol(FTP) is an application layer protocol which moves files between local and remote file systems. It runs on the top of TCP, like HTTP. To transfer a file, 2 TCP connections are used by FTP in parallel: control connection and data connection.

When a FTP session is started between a client and a server, the client initiates a control TCP connection with the server side. The client sends control information over this. When the server receives this, it initiates a data connection to the client side. Only one file can be sent over one data connection. But the control connection remains active throughout the user session. As we know HTTP is stateless i.e. it does not have to keep track of any user state. But FTP needs to maintain a state about its user throughout the session.

Data Structures : FTP allows three types of data structures :

1. File Structure – In file-structure there is no internal structure and the file is considered to be a continuous sequence of data bytes.
2. Record Structure – In record-structure the file is made up of sequential records.
3. Page Structure – In page-structure the file is made up of independent indexed pages.

FTP Commands – Some of the FTP commands are :

USER – This command sends the user identification to the server.

PASS – This command sends the user password to the server.

CWD – This command allows the user to work with a different directory or dataset for file storage or retrieval without altering his login or accounting information.

RMD – This command causes the directory specified in the path-name to be removed as a directory.

MKD – This command causes the directory specified in the pathname to be created as a directory.

PWD – This command causes the name of the current working directory to be returned in the reply.

RETR – This command causes the remote host to initiate a data connection and to send the requested file over the data connection.

STOR – This command causes to store a file into the current directory of the remote host.

LIST – Sends a request to display the list of all the files present in the directory.

ABOR – This command tells the server to abort the previous FTP service command and any associated transfer of data.

QUIT – This command terminates a USER and if file transfer is not in progress, the server closes the control connection.