1 - Explain who are the IETF, what's their job?

IETF is an open, international community of network designers, operators and researchers concerned with developIng voluntary Internet standards and protocols that ensure the long-term stability of the Internet. thier job is to work together for creating and documenting of the standards

2 - What is a Communication protocol?

A communication protocol enable two or more entities (like computers) to communicate and exchange data in a structured and consistent manner. They ensure that data can be sent and received efficiently, and securely. Communication protocols are used in various domains, including computer networks, the internet and more. Examples of communication protocols include HTTP , TCP/IP UDP and more.3

What is an RFC?

Request for Comments, is a document series that are used to document specifications, best practices, and other technical details related to the Internet and its protocols. They can describe anything specification or proposal is being developed within the IETF and after review and consensus, it may become a formal Internet standard.

4 - what is HTTP used for?

HTTP is a communication protocol that enable the transfer of data between a client and a server. It is the protocol that facilitates the retrieval and display of web pages, as well as the transmission of other resources like images, videos, and documents over the internet. HTTP is based on a request-response model, where a client sends an HTTP request to a server, and the server responds with the requested data, typically in the form of a web page.

5 - Explain how does HTTP differ from HTTPS? IN DETAIL (TCP IP TLS SSL)

HTTP does not provide any encryption or security measures. Data transmitted via HTTP is in plain text, making it vulnerable. HTTPS is working with tcp / ip communication, (the same as http) but it has a layer of TLS/SSL protocols that ensures that data transmitted between them is encrypted and secure from eavesdropping.

6 - What is the general model of the HTTP protocol?

. the general model of http protocol has a client that initiates communication with the server by sending an HTTP request to a web server. a server that processes the request and sends back an HTTP response according to the client request. request that has a request method ( GET, POST) indicating the action the client wants to perform and URL. a response that has status (404, 200) headers ,and the body message that has the requested data

7 - Explain: user-agent, Web server, Proxies

User-Agent is an HTTP header sent by the client in an HTTP request that identifies the client's software and version. web server receives, processes, and responds to HTTP requests from clients. Proxies are intermediaries that sit between a client and a server. They can reduce the load on the server and speed up responses to clients, enhancing privacy and security and to Load Balancing.

8 - What are proxies used for? How?

Proxies provide a valuable layer of security for your computer. They can be set up as web filters or firewalls, protecting your computer from internet threats like malware. The end-user or a network administrator can choose a proxy designed to protect data and privacy. This examines the data going in and out of your computer or network. web requests go to the proxy, which then reaches out and gets what you want from the internet. proxies can reducing the load on web servers and improving response times for clients. they can mask a client's IP address which enhances privacy and security. can filtering out malicious content, blocking access to certain websites and can be used to route traffic through specific geographic locations, enabling region-specific content delivery or testing.

9 - What does it mean for protocols to have a state?

When a protocol has a state, it means that it maintains information on communication between two parties (like a client and a server). This information helps the protocol keep track of the progress and history of the communication. Protocols with state remember details such as the current position in a conversation and the data exchanged so far

10 - Does HTTP have a state?

HTTP doesn't have any state because HTTP doesn't maintain any information about the previous requests or responses in the communication between the client and server. Each HTTP request sent by a client to a server is independent, and the server processes it without considering previous requests.

11 - Explain the flow of HTTP

the client initiates communication with the server by sending an HTTP request. the server receives the request, processes it and creates an http response with the status request, headers and the requested data at the end the client receives the HTTP response.

12 - What are request headers? HTTP request headers provide information about the request and help the server understand the client's needs.

13 - What are cookies?

are typically found in the HTTP request headers. They are sent by the client to the server in the request headers. cookies are included in the "Cookie" header of the HTTP request. Cookies are used to store information about a user's activity on a website. Cookies can maintain user sessions, allowing users to remain authenticated when they are using the website. They can also store user preferences, settings and tracking the user behavior, so the manager of the site will know and understand how users interact with their site.

14 - What elements does an HTTP request consist of?

An HTTP request consists few elements: Request Line with the HTTP method, the URL, and the HTTP version. Headers that provide more information about the request, and the body message that may contain the data to be sent to the server.

15 - What elements does an HTTP response consist of?

An HTTP response consists of elements: Status Line with the status code (405, 404, 200) and a status message. Headers that provide data about the response, and the message body that contains the data that the server is sending as a response.

16 - What is HTTP3 and how does it differ from HTTP2?

HTTP2 uses TCP as its transport protocol HTTP3 uses the QUIC transport protocol, which is built on top of the UDP. HTTP3 has few advantages over HTTP2: HTTP3 allows multiple requests and responses to be sent simultaneously over a single connection. This enhances performance compared to HTTP2, which uses multiplexing but is limited by TCP's limitations. HTTP3 includes encryption by default in contrast to HTTP2 that doesn't inherently enforce encryption. HTTP2 also requires the setup of multiple TCP connections to allow parallelism, which can add complexity.