

## Task 1

1.The Hyper Text Transfer Protocol is an application protocol that has been the de facto standard for communication on the world wide web since its invention in 1989.

### HTTP/1.1:

Developed by Timothy Berners-Lee in 1989 as a communication standard for the World Wide Web, HTTP is a top-level application protocol that exchanges information between a client computer and a local or remote web server. In this process, a client sends a text-based request to a server by calling a *method* like GET or POST. In response, the server sends a resource like an HTML page back to the client.

For example, let's say you are visiting a website at the domain `www.example.com`. When you navigate to this URL, the web browser on your computer sends an HTTP request in the form of a text-based message, similar to the one shown here:

```
GET /index.html HTTP/1.1
```

```
HOST:www.example.com
```

This request uses the GET method, which asks for data from the host server listed after Host:. In response to this request, the `example.com` web server returns an HTML page to the requesting client, in addition to any images, stylesheets, or other resources called for in the HTML. Note that not all of the resources are returned to the client in the first call for data. The requests and responses will go back and forth between the server and client until the web browser has received all the resources necessary to render the contents of the HTML page on your screen.

### HTTP/2

HTTP/2 began as the SPDY protocol, developed primarily at Google with the intention of reducing web page load latency by using techniques such as compression, multiplexing, and prioritization. This

protocol served as a template for HTTP/2 when the Hypertext Transfer Protocol working group httpbis of the IETF (Internet Engineering Task Force) put the standard together, culminating in the publication of HTTP/2 in May 2015. From the beginning, many browsers supported this standardization effort, including Chrome, Opera, Internet Explorer, and Safari. Due in part to this browser support, there has been a significant adoption rate of the protocol since 2015, with especially high rates among new sites.

From a technical point of view, one of the most significant features that distinguishes HTTP/1.1 and HTTP/2 is the binary framing layer, which can be thought of as a part of the application layer in the internet protocol stack. As opposed to HTTP/1.1, which keeps all requests and responses in plain text format, HTTP/2 uses the binary framing layer to encapsulate all messages in binary format, while still maintaining HTTP semantics, such as verbs, methods, and headers. An application level API would still create messages in the conventional HTTP formats, but the underlying layer would then convert these messages into binary. This ensures that web applications created before HTTP/2 can continue functioning as normal when interacting with the new protocol.

The conversion of messages into binary allows HTTP/2 to try new approaches to data delivery not available in HTTP/1.1, a contrast that is at the root of the practical differences between the two protocols. The next section will take a look at the delivery model of HTTP/1.1, followed by what new models are made possible by HTTP/2.

## 2.

**HTTP** (HyperText Transfer Protocol) is the underlying protocol of the World Wide Web. Developed by Tim Berners-Lee and his team between 1989-1991, HTTP has seen many changes, keeping most of the simplicity and further shaping its flexibility. HTTP has evolved from an early protocol to exchange files in a semi-trusted laboratory environment, to the modern maze of the Internet, now carrying images, videos in high resolution and 3D.

## Invention of the World Wide Web

In 1989, while he was working at CERN, Tim Berners-Lee wrote a proposal to build a hypertext system over the Internet. Initially calling it the *Mesh*, it was later renamed to *World Wide Web* during its implementation in 1990. Built over the existing TCP and IP protocols, it consisted of 4 building blocks:

1. A textual format to represent hypertext documents, the Hyper text mark up language (HTML)
2. A simple protocol to exchange these documents, the hyper text transfer protocol (HTTP)
3. A client to display these documents, the first web browser called world wide web
4. A server to give access to the document, an early version of httpd

These four building blocks were completed by the end of 1990, and the first servers were already running outside of CERN by early 1991. On August 6<sup>th</sup> 1991, Tim Berners-Lee's [post](#) on the public *alt.hypertext* newsgroup is now considered as the official start of the World Wide Web as a public project.

HTTP/0.9 The one line protocol:

The initial version of http has no number later it was given a number to differentiate from the later versions. This version was extremely simple

GET /mypage.html

The response is also extremely simple, it only consisted of the file itself

<HTML>

A very simple html page

</HTML>

HTTP/1.0-Building extensibility:HTTP/0.9 was very limited and both browsers and servers quickly extended it to be more versatile

HTTP/1.1 – The standardized protocol:

In parallel to the somewhat chaotic use of the diverse implementations of HTTP/1.0 and since 1995 well before the publication of HTTP/1.0 document the next year proper standardization was in progress the first standardized version of http,Http/1.1 was published in the year 1997 only a few months after HTTP/1.0.This cleared ambiguities and numerous improvements

HTTP/2-A protocol for greater performance:

Over the years,web pages have become much more complex,even becoming applications in their own right.The amount of visual media displayed,the volume of size of scripts adding interactivity,has also increased,much more data is transmitted over significantly more HTTP requests

In the first half of the 2010 s,Google demonstrated an alternative way of exchanging data between client and server,by implementing an experimental protocol spdy.This embossed an intrest from developers working on both browsers and servers.defining an increase in the responsiveness and solving the problem of duplication of the data transmitted,SPDY served as the foundations of the HTTP/2 protocol.

### **3.Browser JS vs Node js**

1.JavaScript is a programming language that is used for writing the scripts on the website while Node Js is javascript run time environment

2.JavaScript can only be run in the browsers while NodeJs can run outside the browser

3.BrowserJs basically used on the client side while Node JS is mostly used on the server side

4.JS is capable enough to add HTML and play with DOM while NodeJs does not have the capability to add HTML tags

5.Js can run in any browser engine while NodeJs can only run in V8 engine of Google Chrome

#### **4.What happens when you type an url in the address bar of the browser?**

DNS:Domain Name system.Like a phonebook,DNS maintains and maps the name of the website i.e URL and particular IP address it links to.Every URL on the internet has a unique IP address it links to.Every URL on the internet has a unique IP address which is of the computer which hosts the server of the website requested

Steps for What happens when we enter an URL:

1. Browser checks cache for DNS entry to find the corresponding Ip address of website.

It looks for following cache. If not found in one, then continues checking to the next until found.

- Browser Cache
- Operating Systems Cache
- Router Cache
- ISP Cache

2. If not found in cache, ISP's (Internet Service Provider) DNS server initiates a DNS query to find IP address of server that hosts the domain name.

The requests are sent using small data packets that contain information content of request and IP address it is destined for.

3. Browser initiates a TCP connection with the server using synchronize and acknowledge messages
4. Browser sends an HTTP requests to the web server. GET or POST request
5. Server on the host computer handles that request and sends back a response. It assembles a response in some format like JSON, XML, HTML
6. Server sends out an HTTP response along with the status of the response
7. Browser Displays HTML content