**(1) Write a blog on the Difference between HTTP1.1 vs HTTP2**

**Difference between HTTP1.1 vs HTTP2**

**Introduction**

HTTP1.1 and HTTP2 are versions of the HTTP protocol used for data transfer on the web.

**HTTP 1.1**

Officially standardized in 1999, HTTP 1.1 is the most recent version of HTTP.

Just like its predecessor, this application protocol transfers data between web servers

and clients.

However, what makes it different is the incorporation of PUT, DELETE, and OPTIONS,

and it includes features like HTTP pipelining and chunked transfer encoding.

Due to all these features, HTTP 1.1 provides better performance and more flexibility

than HTTP 1.

**Benefits of HTTP 1.1**

There are several benefits of HTTP 1.1 compared to the earlier version HTTP 1.0.

**Persistent Connections**

One of the most significant benefits of HTTP 1.1 is the support for persistent connections,

allowing multiple requests to be sent over the same connection.

This reduces the overhead of establishing new connections for each request and

results in improved performance.

**Improved Caching**

With several features that improve caching, such as the ability to send conditional

GET requests and support for caching negotiation, and clients can cache responses

and avoid unnecessary network traffic.

**More Efficient Request and Response Handling**

With PUT, DELETE, and OPTIONS, fine-grained control over resources on the server is

facilitated. It also supports HTTP pipelining and chunked transfer encoding,

thus reducing the latency of sending large payloads.

**HTTP 2**

Officially standardized in May 2015, HTTP 2 is the second major version of the HTTP

network protocol for transmitting data over the Internet. It was developed to

improve the performance and efficiency of web applications by reducing the amount of

data sent over the wire. This reduces the number of round trips required to load a

webpage and allows for a more efficient application of resources.

**Benefits of HTTP2**

As technology became more evolved, HTTP 2 was developed. It addressed some of the

limitations and inefficiencies of HTTP 1.1.

**Request Multiplexing**

In HTTP 1.1, each request and response were sent over a separate connection.

A new connection had to be established for each request, resulting in many round trips required to load a webpage. HTTP 2 came with multiplexing of requests.

**Header Compression**

HTTP 1.1 did not have built-in support for header compression, so headers were sent

in plain text, they took up a significant amount of bandwidth.

As a result, while sending large numbers of headers or when sending headers over slow

connections, problems were faced.

These are the major reasons why HTTP 1.1 was replaced.

However, there are other differences between the two versions. So first, let us

compare both of them.

**HTTP 1.1 vs. HTTP 2**

There are some parameters based on which we will compare both the HTTP versions.

**Predicting Resource Requests**

In HTTP 1.1, the client-server initiates all requests for resources such as images,

stylesheets, and scripts. The server can only respond to requests that it receives.

This means that the client must first request the HTML of a web page, parse it, and

then make additional requests for any additional resources it needs to render the page.

This results in delayed page load times.

Contrary to this, HTTP 2 allows for server push, so the server proactively pushes

resources to the client without the client needing to request them.

This speeds up page load times as the client starts processing and rendering resources

as soon as they are received.

**Buffer Overflow**

In HTTP 1.1, a buffer overflow can occur when a client sends a request with a header

larger than the server’s buffer size. This can cause the server to crash or become

unresponsive. To prevent buffer overflow, servers typically have a maximum buffer size

for incoming requests and reject requests exceeding this limit.

HTTP 2, on the other hand, uses a more sophisticated approach to prevent buffer overflow.

It uses a flow control mechanism that allows the server to send data to the client in

small chunks rather than sending all the data at once.

**Multiplexing**

HTTP 1.1 uses a separate connection for each request and response, resulting in increased

latency and reduced performance. In contrast, HTTP 2 supports multiplexing, simultaneously sending

multiple requests and responses over a single connection

This helps to reduce the latency and increase the overall performance of the connection.

**Binary Protocol**

HTTP 1.1 uses plain text to encode and transmit data. Though it is easy for humans to

read and understand the data. It can be less efficient than a binary protocol.

HTTP 2 encodes and transmits data rather than plain text using binary codes.

Binary protocols are generally more efficient than text-based protocols because they can

transmit data more compactly.

|  |  |  |
| --- | --- | --- |
|  | **HTTP 1.1** | **HTTP 2** |
| **Development** | It was developed in the year 1997 | It was developed in the year 2015 |
| **Compression** | It compresses data by itself. | It uses HPACK for data compression. |
| **Binary Protocol** | A text-based protocol uses plain text to encode and transmit data. | It works on the binary protocol as binary codes encode and transmit data rather than plain text. |
| **Security** | The client sends a request to a server, and the server responds to the client. | A different underlying protocol called Secure Remote Protocol 2 (SRP2) establishes a secure connection between a client and a server. |
| **Multiplexing** | A separate connection is established for each request and response, which can add overhead and latency to the communication process. | It allows multiplexing so multiple requests and responses can be sent over a single connection. |
| **Buffer Overflow** | HTTP 1.1 cannot handle buffer overflow vulnerabilities due to insufficient measures. | HTTP 2 includes measures to prevent buffer overflow vulnerabilities. |
| **Performance** | HTTP 1.1 does not include any in-built features, so its performance is less efficient. | HTTP 2 is designed to be more efficient and performant than HTTP 1.1. This is because HTTP 2 includes features like multiplexing, binary protocol, and header compression. |

**Concluding Words**

HTTP is the backbone of the internet and enables the creation and sharing of various information and resources. Without HTTP, the internet would not be able to function in the way that it does today. However, with time, HTTP versions kept changing.