**Difference between HTTP1.1 vs HTTP2**

**What is HTTP1?**

HTTP 1.1 is **the latest version of Hypertext Transfer Protocol (HTTP)**, the World Wide Web application protocol that runs on top of the Internet's TCP/IP suite of protocols. HTTP 1.1 provides faster delivery of Web pages than the original HTTP and reduces Web traffic.

**What is HTTP2?**

HTTP/2 (originally named HTTP/2.0) is **a major revision of the HTTP network protocol used by the World Wide Web**. It was derived from the earlier experimental SPDY protocol, originally developed by Google.

**What is the difference between HTTP1 1 and HTTP2?**

Multiplexing: HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. In contrast, HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource.

**HTTP1 Vs. HTTP2 .**

HTTP/1.1 has been around for more than a decade. With Google’s SPDY leading the way in 2015, the IETF (Internet Engineering Task Force) gave us HTTP/2, which introduces several features to reduce page load times. Let’s compare HTTP2 Vs. HTTP1.1 in detail.

HTTP/2 achieves faster webpage loading without performance optimizations that require extensive human efforts in terms of development. It significantly reduces the complexities that had crept into HTTP/1.1 and gives us a robust protocol which, though not without its flaws, will perhaps stand the test of time. Before making this leap forward, let’s trace our steps back to when the internet was in its infancy to understand how the different versions evolved into the current form.

**Features of HTTP/1.1:**

* It was no longer required for each connection to be terminated immediately after every request was served with a response; instead, with the keep-alive header, it was possible to have persistent connections. It allowed multiple requests/responses per TCP connection.
* The Upgrade header was used to indicate a preference from the client that made it possible to switch to a more preferred protocol if found appropriate by the server.
* HTTP/1.1 provided support for chunk transfers that allowed streaming of content dynamically as chunks and for additional headers to be sent after the message body. This enhancement was particularly useful in cases where values of a field remained unknown until the content had been produced. For example, when the content had to be digitally signed, it was not possible to do so before the entire content gets generated.
* Other features that reinforced its stability were introduced such as:
* pipelining (the second request is sent before the response to the first is adequately served)
* content negotiation (an exchange between client and server to determine the media type, it also provides the provision to serve different versions of a resource at the same URI)
* cache control (used to specify caching policies in both requests and responses)

**HTTP/2 Features**

* Binary: Meaning commands use 1s and 0s and not text
* Multiplex: Permits multiple requests and responses to be sent at the same time
* Compression: Compresses headers that have been requested previously to make things more efficient
* Stream prioritization: This allows for the exchange of successive streams at one time
* Server push: The server can send additional information needed for a request before it is requested
* Increased security: HTTP/2 is supported through encrypted connections

This diagram compares the function HTTP/1.1 and HTTP/2.



**Conclusion**

The influence and control of HTTP/2 in the cyber world are absolutely inexorable. The core features of HTTP/2 provide greater levels of control that can be used to optimize the web application performance. Certainly, the tech world is rapidly evolving with each passing year, which needs advanced technologies every now and then. HTTP/3 is the upcoming internet protocol developed to fix the shortcomings of its predecessor. However, there is so much left to do, and HTTP/2 is not going away any time soon.