

HTTP2 VS HTTP1.x

(Specifications & Suggestions)

Girish K. M. ,Akshay R. Muntode

Abstract: This paper is a research conducted on HTTP2 and describes all the features and developments to previous versions. It also states various advantages and its disadvantages. The papers provides a vision on the improvements of HTTP2 by comparing it to its predecessor HTTP 1.x.

Keywords— HTTP, Frame, Stream, Flow Control, Error handling, Server push, Ping, Endpoint, Peer, TCP, Proxy

I. INTRODUCTION

This specification describes an optimized expression of the semantics of the Hypertext Transfer Protocol (HTTP). HTTP/2 enables a more efficient use of network resources and a reduced perception of latency by introducing header field compression and allowing multiple concurrent exchanges on the same connection. It also introduces unsolicited push of representations from servers to clients.

This specification is an alternative to, but does not obsolete, the HTTP/1.1 message syntax. HTTP's existing semantics remain unchanged.

II. OBJECTIVE

The working group charter mentions several goals and issues of concern

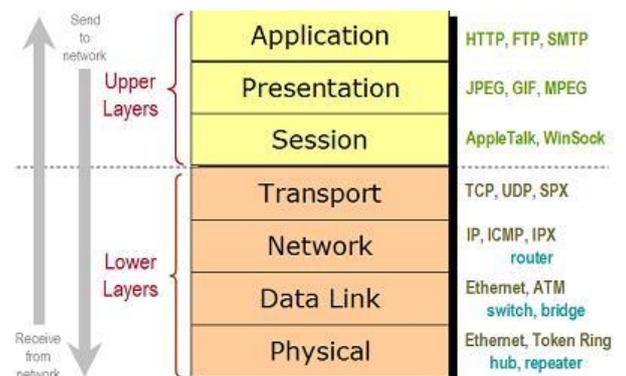
- Negotiation mechanism that allows clients and servers to elect to use HTTP 1.1, 2.0, or potentially other non-HTTP protocols.
- Maintain high-level compatibility with HTTP 1.1 (for example with methods, status codes, and URIs, and most header fields)
- Decrease latency to improve page load speed in web browsers by considering:
 - Data compression of HTTP headers
 - Server push technologies
 - Fixing the head-of-line blocking problem in HTTP 1

- Loading page elements in parallel over a single TCP connection
- Support common existing use cases of HTTP, such as desktop web browsers, mobile web browsers, web APIs, web servers at various scales, proxy servers, reverse proxy servers, firewalls, and content delivery networks of the current designations.

III. PREPARE YOUR PAPER BEFORE STYLING

The HTTP (Hyper Text Transfer Protocol) is a protocol used in the application layer of the OSI model and uses a TCP (Transmission Control Protocol) connection.

The client initiates the connection using a TCP connection via the Transport layer of the OSI model.



HTTP2 uses traditional protocols of “http” and “https”:

- http: is used for sending normal hypertext data.
- https: is used whenever a secure connection was needed, example online payment.

And also HTTP2 uses the same default port i.e. 80 for “http” URI’s and 443 for “https” URI’s. Before the connection is made it has to be validated whether the client browser supports HTTP2. This is done using a process called “Discovery” which is explained in the later part.

Identifying HTTP2 version

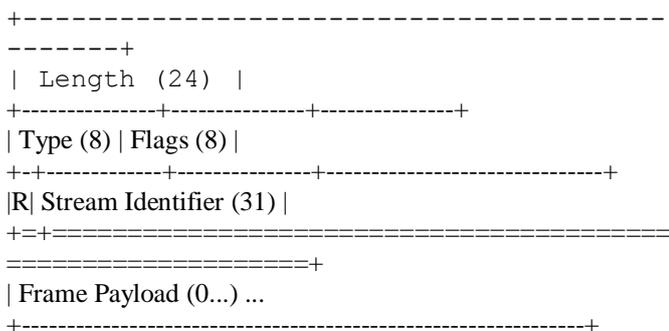
The HTTP2 version can be determined using the “h2” header which is a part of TLS i.e. Transport Layer Security. The HTTP2 uses TLS version 1.2.

The string “h2c” identifies the protocol where HTTP2 message is run using cleartext

Starting HTTP2

When a client make an “http” request without any prior knowledge whether the protocol being used is HTTP2, it uses the HTTP-upgrade. When a client make a request using HTTP 1.x, in the packet format of HTTP1.x lies and “Upgrade header” field with the “h2c” token.

A. Http2 packet format



The http2 packet format is no different than is predecessor, it consists of all the same fields that of an HTTP1.x packet format. The only difference lies in the header where the header is to be replaced with “h2” or “h2c” for http cleartext as mentioned above.

B. Features

Http2 inherits the basic features of its counterpart http1.x and has some new additions to those features which make it more reliable and one of the reasons for coming to being.

Stateless: this feature of http1.x/2 states that the http connection between client and server is not recorded thus when the connection is terminated there is no record of a client being connected to a particular server

Connectionless: Being connectionless means this protocol does not need to be connected to the server always, when the client sends a request the connection is disconnected and waits till the server can generate a response for client’s request. When the response is generated the server reconnects to the client and provides required response.

Media Independent: Media independence states that http can transfer any type of data provided it can be processed by the other side.

Multiple connection: This is a new feature introduced in Http2, this feature states that a client can establish more than one connection with a particular server at an instance.

Push Service: The push service is a feature for a server machine where in the server responds with more than one data for a particular requesting thus reducing the overhead in request cycles. With this feature the server can recognize and send data which will be required by the client in future.

C. HTTP1.x v/s HTTP2.0

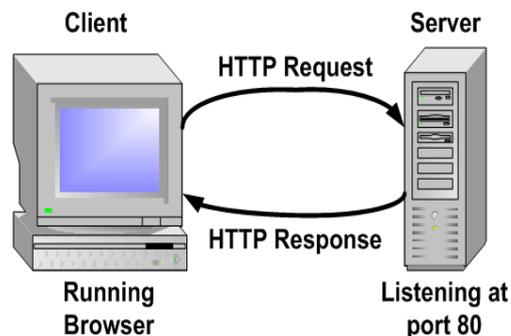
About HTTP:

Http was first introduced in 1995 by Tim Berners-Lee and his team. The first version of Http has only one method, namely GET which was used to send requests to a server, and the response was always an HTML (Hyper Text Markup Language).The first version of HTTP was given the code HTTP v0.9.

Later Dave Ragette, lead the HTTP research group in order to improve http. Dave’s vision was to expand the protocol by adding new features such as- extended negotiations, richer meta-information, along with a security protocol. Thus with these goals in mind HTTP 1.0 was formed in 1996.

HTTP is an application layer protocol. It functions as a request-response protocol in client-server computing. Whenever a client requires a page to be loaded, client sends an HTTP request to the server. The server processes this information and generates a response for the client. This response is an HTML page.

disadvantage according to the author is of multiplexing, as a suggestion author suggests to limit the number of connections from a particular node.



REFERENCES

- [1] M.Belshe, BitGo, HTTPbis Working Group, Hyper Text Transfer Protocol Version 2(HTTP/2) draft-ietf-httpbis-http2-04, Feb 11th, 2015
- [2] http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol

D. Advantages of HTTP 2 over HTTP 1.x

There are two main advantages of HTTP 2 i.e., **Multiple connections** and **Push service**. Multiple connection states that the client can have more than one request running over a server at any particular instant. Push service prevents the loading time or can also be stated as a round trip i.e. the time between request to a server and its reply to the request, this is done by sending or **pushing** data that may be required by the client in near future without the client sending a request for that particular data. Thus reducing the bandwidth and time for request response cycle.

E. Disadvantages of HTTP2

There are things to love about HTTP 2 but there are also things to hate about the new protocol in town.

According to the concept of multiple connections a client may open multiple connections to the same IP address and TCP port using different server name identification values. This feature is a boon as well as a curse, creating multiple connections will consume greater bandwidth and if multiple connections are made using the same machine without closing the unused connection it may lead to a bottle neck situation thus resulting in the termination of all existing connections as well.

Conclusion:

After careful study and observation of the new protocol i.e. HTTP2 the author concludes by saying, this new evolving protocol is the future of browsing as it over comes many a short comings of its predecessor. These new features makes it easier to implement and is more reliable. The main