**Hypertext Transfer Protocol (HTTP)** is a method for encoding and transporting information between a client (such as a web browser) and a [web server](https://www.nginx.com/products/nginx/). HTTP is the primary protocol for transmission of information across the Internet.

**HTTP/1.1:**

The first standardized version of HTTP was called HTTP/1.1. This version is still in use on the web. Even though the HTTP/1.1 protocol was refined over two revisions, [RFC 2616](https://datatracker.ietf.org/doc/html/rfc2616) published in June 1999 and [RFC 7230](https://datatracker.ietf.org/doc/html/rfc7230)-[RFC 7235](https://datatracker.ietf.org/doc/html/rfc7235) published in June 2014 before the release of HTTP/2, it was extremely stable for more than 15 years.

**HTTP/2:**

In 2015, a new version of HTTP called HTTP/2 was created. HTTP/2 solves several problems that the creators of HTTP/1.1 did not anticipate. In particular, HTTP/2 is much faster and more efficient than HTTP/1.1. One of the ways in which HTTP/2 is faster is in how it prioritizes content during the loading process.

The HTTP/2 protocol differs from HTTP/1.1 in a few ways:

* It's a binary protocol rather than a text protocol. It can't be read and created manually. Despite this hurdle, it allows for the implementation of improved optimization techniques.
* It's a multiplexed protocol. Parallel requests can be made over the same connection, removing the constraints of the HTTP/1.x protocol.
* It compresses headers. As these are often similar among a set of requests, this removes the duplication and overhead of data transmitted.
* It allows a server to populate data in a client cache through a mechanism called the server push.

Officially standardized in May 2015, HTTP/2 use peaked in January 2022 at 46.9% of all websites. High-traffic websites showed the most rapid adoption in an effort to save on data transfer overhead and subsequent budgets.

This rapid adoption was likely because HTTP/2 didn't require changes to websites and applications. To use it, only an up-to-date server that communicated with a recent browser was necessary. Only a limited set of groups was needed to trigger adoption, and as legacy browser and server versions were renewed, usage was naturally increased, without significant work for web developers.