



Universidad Iberoamericana, Ciudad de México.

Arquitectura de información web

Gustavo Antonio Luna Maya

Investigación HTTP

Periodo: Primavera

27/01/2021



## Índice

1. ¿Qué es HTTP y cómo funciona? .....página 3
2. Métodos de petición HTTP.....página 4
3. Ventajas de las versiones más recientes .....página 6

Este documento consta de una investigación sobre el protocolo HTTP o Protocolo de Transferencia de Hipertexto, veremos lo que significa y que es, como funciona y para que sirve, los métodos de petición HTTP y finalmente se hablará sobre las ventajas que tienen las versiones más recientes de HTTP como la versión 3.

¿Qué es HTTP?

*“Es la abreviación de Protocolo de Transferencia de Hipertexto y es un protocolo de aplicación para permitir la comunicación entre sistemas físicamente dispersos.*

*Fue originalmente ideado por Sir Tim Berners Lee en 1989. Ahora está coordinado por el W3C. En su formato más básico, es lo que establece cómo las páginas web se comunican desde el servidor web al navegador del usuario.*

*El Hipertexto es el texto que se usa en las páginas web y que conecta una página con otra, ya sea dentro del mismo sistema o externamente. Son los enlaces que conectan el contenido en Internet.*

¿Cómo funciona HTTP?

*HTTP es un protocolo de aplicación para transferir recursos a través de Internet.*

*HTTP usa el Puerto 80, que es el puerto desde el que el servidor web acepta peticiones.*

*La mayoría de los recursos son archivos (imágenes, etc.) pero puede incluir otro tipo de datos como scripts.*

*Las sesiones HTTP se abren por un cliente HTTP (es decir, el navegador del usuario) a través de un agente de usuario y se envía un mensaje de petición de conexión al servidor HTTP (es decir, el servidor web). El mensaje de petición se conoce también como ‘petición del cliente’ y consiste en las siguientes líneas:*

- *Línea de petición*
- *Encabezados*
- *Línea vacía*
- *Un cuerpo del mensaje opcional*

*Una vez que la respuesta ha sido entregada el servidor web cierra la conexión.*

*Este tipo de conexión es conocido como Stateless. La conexión solo existe por el periodo que dure el intercambio de datos.*

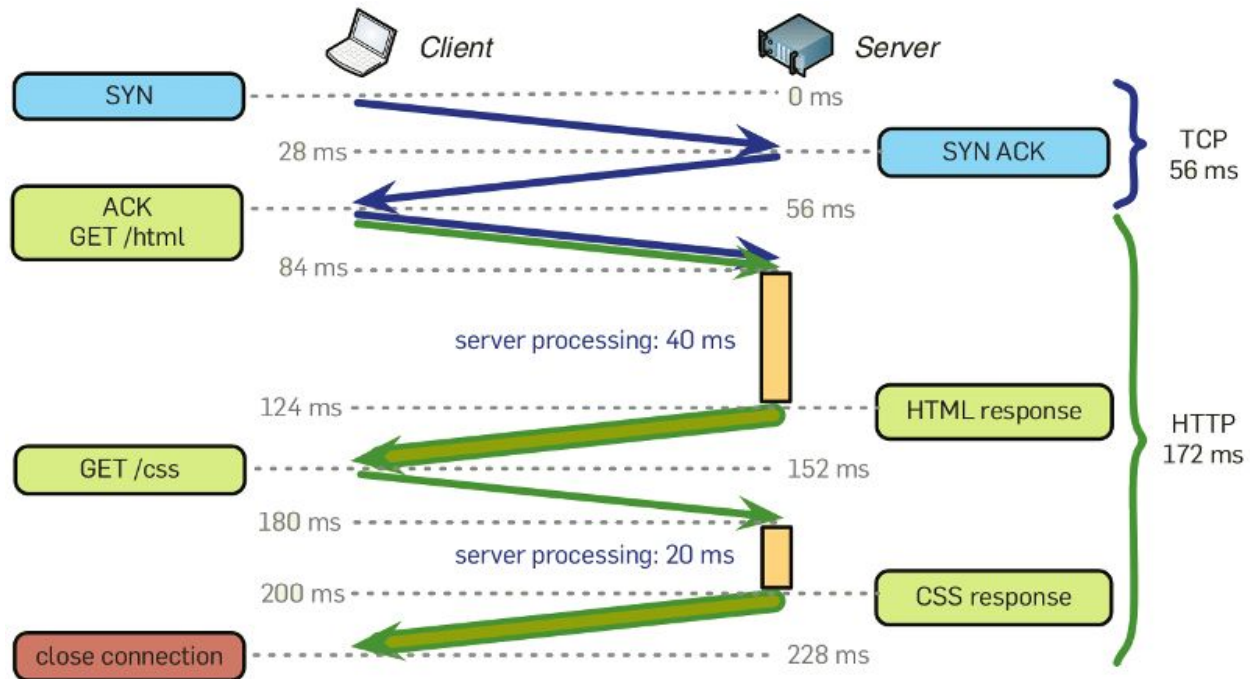
*En función de la disponibilidad o no del recurso, HTTP proporciona un código de estatus apropiado (también conocido como ‘respuesta del servidor’), determinado por el protocolo. Son los siguientes:*

- *1xx: un simple mensaje informativo*
- *2xx: éxito de algún tipo (ej: 200 OK – file found)*
- *3xx: el cliente a otra URL (ej: 301 moved permanently)*
- *4xx: un error en el lado del cliente (ej: 404 archivo no encontrado)*

- 5xx: un error en el lado del servidor (ej: 500 server error)”<sup>1</sup>

El protocolo HTTP funciona para hacer la conexión entre el cliente (navegador) y el servidor de la página web a la que el usuario del cliente quiere acceder, para esto se utiliza el puerto 80.

En la siguiente imagen se muestra el proceso general de conexión a una página web:



## Métodos de petición HTTP

“HTTP define un conjunto de **métodos de petición** para indicar la acción que se desea realizar para un recurso determinado. Aunque estos también pueden ser sustantivos, estos métodos de solicitud a veces son llamados HTTP verbs. Cada uno de ellos implementa una semántica diferente, pero algunas características similares son compartidas por un grupo de ellos: ej. un request method puede ser safe, idempotent, o cacheable.

### GET

El método **GET** solicita una representación de un recurso específico. Las peticiones que usan el método **GET** sólo deben recuperar datos.

### HEAD

El método **HEAD** pide una respuesta idéntica a la de una petición **GET**, pero sin el cuerpo de la respuesta.

<sup>1</sup> ¿Qué es HTTP? (2018, 4 mayo). Pickaweb. <https://www.pickaweb.es/ayuda/que-es-http/>

## **POST**

El método *POST* se utiliza para enviar una entidad a un recurso en específico, causando a menudo un cambio en el estado o efectos secundarios en el servidor.

## **PUT**

El modo *PUT* reemplaza todas las representaciones actuales del recurso de destino con la carga útil de la petición.

## **DELETE**

El método *DELETE* borra un recurso en específico.

## **CONNECT**

El método *CONNECT* establece un túnel hacia el servidor identificado por el recurso.

## **OPTIONS**

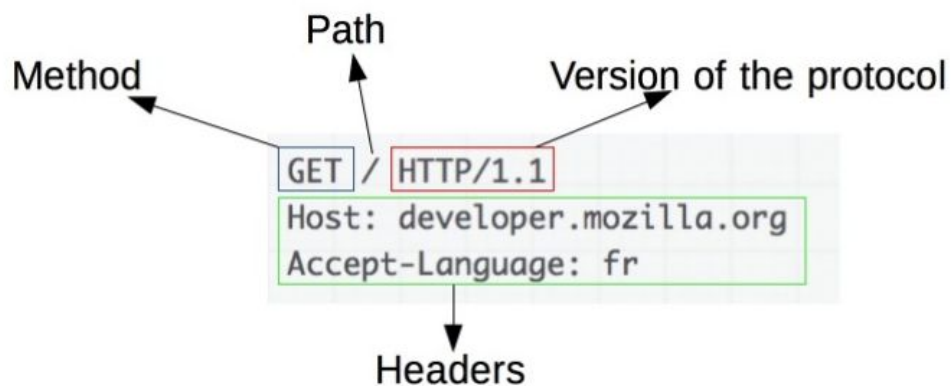
El método *OPTIONS* es utilizado para describir las opciones de comunicación para el recurso de destino.

## **TRACE**

El método *TRACE* realiza una prueba de bucle de retorno de mensaje a lo largo de la ruta al recurso de destino.

## **PATCH**

El método *PATCH* es utilizado para aplicar modificaciones parciales a un recurso.”<sup>2</sup>



Los métodos de petición son una clase de comandos o frases que el protocolo HTTP interpreta según la solicitud del cliente realizada por el usuario, por ejemplo si el cliente

<sup>2</sup> Métodos de petición HTTP - HTTP | MDN. (2020, 15 octubre). MDN Web Docs.  
<https://developer.mozilla.org/es/docs/Web/HTTP/Methods>

solicita al servidor información se utiliza el método GET el cual solo recupera datos del servidor.

## **Ventajas de las versiones más recientes**

La versión de **HTTP 2.0** mantiene los mismos códigos de estado (1XX, 2XX, 3XX, 4XX, 5XX), métodos (GET, POST, PUT, HEAD, etc) y la misma estructura de la cadena de llamadas.

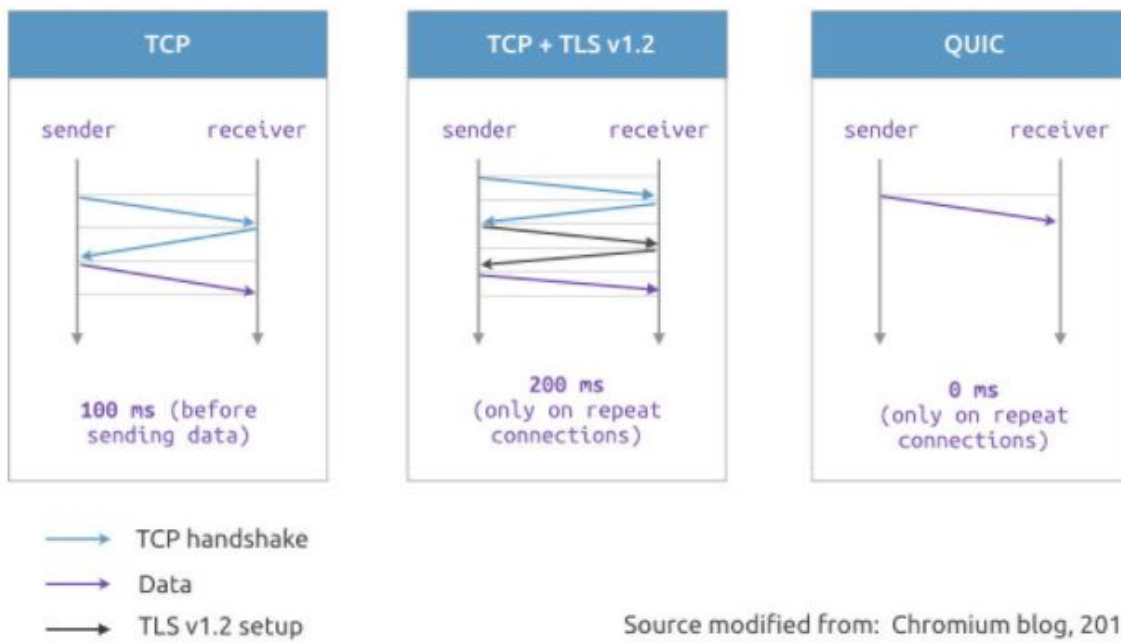
Algunas de las diferencias son: que ahora las cabeceras se encuentran en un formato binario en lugar de formato de texto como en las versiones 1.0 y 1.1, las cabeceras están comprimidas lo que permite una menor cantidad de información transferida por cada petición, cuenta con tecnología server push la cual carga los archivos HTML, CSS, Js entre otros recursos sin que el cliente lo solicite lo cual hace el proceso de carga de las páginas sea más rápido y cuenta con multiplexación lo cual permite que se resuelvan varias peticiones a la vez, esto le permite al cliente que solo necesite una conexión para cargar la página. según: Guillem, V. (2016, 18 enero).

*HTTP/2: hablando sobre sus novedades y ventajas.* Genbeta.  
<https://www.genbeta.com/desarrollo/http-2-hablando-sobre-sus-novedades-y-ventajas>

HTTP/3: Esta version ya no usará el protocolo TCP por primera vez en los protocolos HTTP, ahora usará el protocolo QUIC o Quick UDP Internet Connections el cual está basado en un protocolo anterior que es UDP este tiene como diferencia que no necesita estar intercambiando información de manera continua entre el emisor y el receptor del paquete de informacion, tambien este protocolo ya no se hace cargo de preservar la integridad de los datos si no que ahora esa responsabilidad de heredar a cada aplicación que utilice el protocolo HTTP/3, en otra palabra sería como si no hubiera un handshaking osea que no el emisor no espera a que el paquete o paquetes lleguen al receptor. Una desventaja de este protocolo podría ser que está cifrado en TSL 1.3 por el cual no se puede estimar el RTT o Round Trip Time, osea el tiempo en el que tarda un paquete en salir del emisor llegar al receptor y volver al emisor.(retraso o latencia), Esto no quiere decir que el retraso sea mayor, al contrario tiende a ser menor.

En la imagen se muestra el comportamiento de los protocolos TCP, TCP + TSL y QUIC al enviar y recibir datos entre el emisor y receptor.

## Comparison between TCP, TCP + TLS, and QUIC



3

<sup>3</sup> Referencias utilizadas para la versión HTTP/3: Fernández, Y. (2020, 6 marzo). *HTTP/3: qué es, de dónde viene, y qué es lo que cambia para buscar un Internet más rápido*. Xataka. <https://www.xataka.com/basics/http-3-que-donde-viene-que-que-cambia-para-buscar-internet-rapido>