

# Qué es el protocolo HTTP y cómo funciona

## RECURSOS

**HTTP** (Hypertext Transfer Protocol), en español: Protocolo de Transferencia de Hipertexto, **es el protocolo por excelencia de transferencia para el hipertexto**. Por lo tanto, es el enlace de normas para transportar archivos de texto, imágenes, gráficas, sonido, video y otros tipos de archivos multimedia que pueden ser consumidos en la World Wide Web.

Desde el primer momento en que abres el navegador de tu preferencia, estás consumiendo este protocolo, ya que el navegador es un consumidor HTTP, que envía solicitudes a diferentes servidores y retorna la respuesta de estos al usuario final, o sea tú o cualquier persona que navegue en internet.

## Qué es HTTP

Es un protocolo de comunicación entre aplicaciones basado en el intercambio de texto. El protocolo HTTP es el que impulsa todo internet. Los navegadores web utilizan este protocolo para solicitar páginas web a los servidores. El servidor devuelve todos los datos necesarios en código HTML para que puedan mostrarse en el navegador.

## Qué es un protocolo

Es conjunto de reglas en las que se van a comunicar dos entidades, en este caso dos computadoras. El modelo **TCP/IP** nos permite esta comunicación entre computadoras.

## Cómo funciona el protocolo HTTP

Así como los humanos nos comunicamos a través del lenguaje, las computadoras se pueden comunicar a través de HTTP gracias al modelo de TCP/IP.

Las reglas principales de HTTP se suelen definir por los *headers*, que se dividen en dos grupos: los *headers* de petición y los *headers* de respuesta:

### Header request (petición)

Suelen estar definidos por el método de petición, hacia donde se hará el *request*, y la información que enviará la petición.

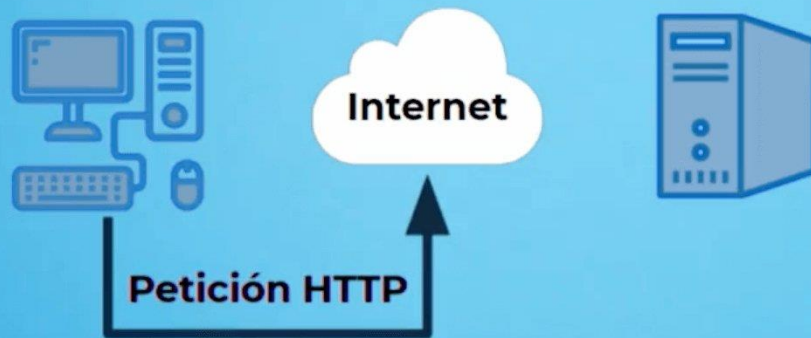
### Header response (respuesta)

Suele estar definido principalmente por la respuesta del servidor y también por las características de la respuesta.

**\*\*Así funciona el protocolo HTTP:**

1. El cliente envía una petición, se transmite por internet.
2. El servidor recibe la petición y genera una respuesta, la envía.
3. El cliente recibe la respuesta (y si estamos en el navegador, interpreta lo recibido).

# Cómo funciona HTTP



## Características del protocolo HTTP

Las características principales del protocolo HTTP son:

### 1. Conexión única

Desde la versión 2.0 de HTTP, este usa una conexión única para generar múltiples solicitudes y respuestas en paralelo, lo cual a la hora de procesar las solicitudes genera una eficiencia mayor en la respuesta.

### 2. Elimina la información redundante

Al eliminar la información redundante se busca evitar el envío de datos repetidos durante una misma conexión, así se consigue un menor consumo de recursos y, por lo tanto, se obtiene una menor latencia.

### 3. Tiene multiplexación

Al ser un protocolo multiplexado, puede enviar y recibir varios mensajes al mismo tiempo, optimizando la comunicación. Esta característica mejora considerablemente la velocidad de carga.

### 4. Es un protocolo binario

HTTP trabaja mediante tramas y al ser un protocolo binario facilita encontrar el comienzo y el final de cada *frame*, lo cual al ser un protocolo de texto podría llegar a ser complicado de identificar. Adicionalmente, los protocolos binarios son más simples, por lo que tienden a ser menos propensos a errores.

### 5. Servicio *server push*

Este servicio de HTTP se basa en estimaciones con las que el servidor es capaz de enviar información al usuario antes de que esta sea solicitada para conseguir que la información esté disponible de forma inmediata.

## 6. Compresión de cabeceras

En versiones anteriores, dados los requerimientos de las solicitudes, las cabeceras tenían un tamaño considerable, lo cual aumentaba el tiempo para obtener respuestas. A partir de la versión 2.0 las cabeceras empezaron a ser comprimidas para mejorar la eficiencia en la respuesta haciendo uso de un algoritmo simple y poco flexible llamado HPACK.

## 7. Priorización de flujos

Un mensaje HTTP se puede dividir en múltiples fragmentos al ir desde el cliente hasta el servidor o viceversa y el orden en que lleguen a su destino es fundamental, es por eso que a cada flujo de le puede asignar un peso que va desde 1 hasta 256 y una dependencia. Haciendo uso de las prioridades y dependencias el protocolo crea un árbol de prioridades para los mensajes y solicitudes.

### Ejemplo de protocolo HTTP

Para este ejemplo, vamos a emplear la terminal de comandos de tu computador. Si tienes Windows, puedes encontrarla simplemente buscando "CMD" en la barra de búsqueda de Windows.

Ya dentro de la terminal, vamos a usar el comando curl:

```
curl https://platzi.com
```

Al ejecutar el comando, estamos enviando una solicitud HTTP al servidor, que nos enviará todo el código HTML del sitio web. Si queremos ver solamente los encabezados, usamos el modificador -v

```
curl https://platzi.com -v
```

Utilizamos redirección (>) para no observar todo el HTML. Aquí tenemos una redirección de salida

```
curl https://platzi.com -v > /dev/null
```