

HTTP

Hypertext Transfer Protocol (HTTP) (o **Protocolo de Transferencia de Hipertexto en español**) es un protocolo de la [capa de aplicación](#) para la transmisión de documentos hipermedia, como HTML. Fue diseñado para la comunicación entre los navegadores y servidores web, aunque puede ser utilizado para otros propósitos también. Sigue el clásico [modelo cliente-servidor](#), en el que un cliente establece una conexión, realizando una petición a un servidor y espera una respuesta del mismo. Se trata de un [protocolo sin estado](#), lo que significa que el servidor no guarda ningún dato (estado) entre dos peticiones. Aunque en la mayoría de casos se basa en una conexión del tipo TCP/IP, puede ser usado sobre cualquier [capa de transporte](#) segura o de confianza, es decir, sobre cualquier protocolo que no pierda mensajes silenciosamente, tal como UDP.

Aprende cómo utilizar HTTP con guías y tutoriales.

[Generalidades de HTTP](#)

Se presentan las características básicas del protocolo y su estructura cliente-servidor: qué puede hacer y cuáles son sus usos.

[HTTP Caché](#)

La gestión de la Caché es fundamental para la eficiencia de sitios Web. En este artículo se presentan los distintos tipos de caché y cómo usar las cabeceras HTTP para su configuración y control.

[HTTP Cookies](#)

El funcionamiento de las cookies se define en [RFC 6265](#). Al recibir una petición HTTP, un servidor puede enviar una cabecera Set-Cookie junto con la respuesta. Posteriormente el cliente devuelve el valor de la cookie en cada petición al mismo servidor en forma de cabecera de solicitud Cookie. La cookie también puede tener una fecha de expiración determinada, o puede estar restringida a un dominio y path específico.

[Control de Acceso HTTP \(CORS\) \(en-US\)](#)

Las **Solicitudes Inter-Sitio HTTP** (Cross-site HTTP requests en inglés), son peticiones HTTP por recursos pertenecientes a un dominio distinto al dominio del recurso que está haciendo la petición. Por ejemplo, una página HTML de un dominio A (<http://dominioa.ejemplo/>) hace una solicitud por una imagen en un dominio B (<http://dominiob.foo/imagen.jpg>) a través del elemento `img`. Hoy en día, las webs utilizan recursos de otros orígenes muy a menudo, incluyendo hojas de estilo CSS, imágenes, scripts y otros recursos. El Control de Acceso HTTP posibilita a los desarrolladores web a controlar cómo su sitio web responde a solicitudes de otros orígenes.

[Evolución de HTTP](#)

Una breve descripción de los cambios del protocolo HTTP desde sus primeras versiones hasta el moderno HTTP/2 y más allá.

[Consejos de Seguridad Web de Mozilla](#)

Una colección de tips para ayudar a equipos de desarrollo con la creación de aplicaciones web seguras.

[Mensajes HTTP](#)

Se describen los tipos de mensajes y distintas estructuras de los mensajes del protocolo HTTP/1.X y HTTP/2

[La típica sesión HTTP](#)

Se muestra y explica cómo es una sesión HTTP típica.

[Manejo de conexión en HTTP/1.X](#)

Se describen los tres tipos de gestiones posibles en una sesión HTTP/1.x, sus principales ventajas e inconvenientes.