

HTTP

Bedeutung

HTTP steht für Hypertext Transfer Protocol und ist ein Protokoll, welches der Browser benutzt, um Websites aufzurufen.

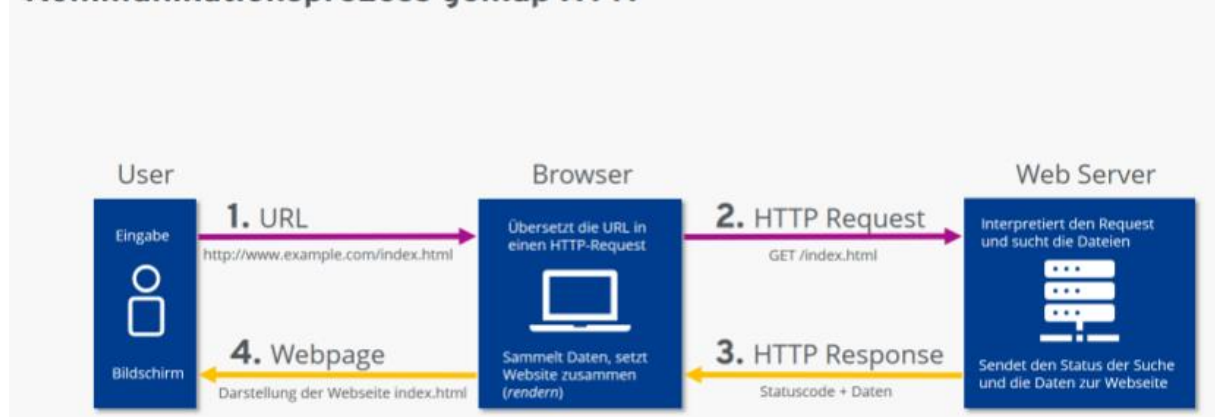
Zweck

Der Client (Browser) schickt eine HTTP-Anfrage an den Server, welche dann eine Antwort zurückschickt. In dieser Antwort sind Statusinformationen über die Anfrage und evtl. auch der angeforderter Inhalt erhalten. Es ist die Sprache, womit der Browser mit dem Webserver spricht, um ihm mitzuteilen, was verlangt wird.

Funktionsweise

1. Nutzer gibt im Webbrowser die Adressenzeile ein, z.B. wikipedia.org.
2. Der Browser sendet dann an den zuständigen Webserver, der die Domäne verwaltet, eine Anfrage, genannt HTTP-Request. Meistens wird in dem Request angefragt, ob der Webserver die Datei schicken kann, es ist aber auch möglich, dass der Client fragt, ob der Webserver die Datei auch einfach hat.
3. Nachdem der Webserver die Datei bekommen hat, sucht er nach der gewünschten Datei (z.B. die Startseite von Wikipedia, also die Datei index.html) und sendet dann den Header. Der Header beinhaltet Status-Codes, welche das Resultat seiner Suche nach der Datei mitteilt. Wenn die Datei gefunden wurde (und der Client die Datei auch wirklich will und nicht nur nach der Existenz der Datei fragt), sendet der Server nach dem Header den Message Body. Dies ist der eigentliche Inhalt der Webseite. Hier wäre es die Datei index.html.
4. Der Browser empfängt die Datei und stellt sie als Webseite dar.

Kommunikationsprozess gemäß HTTP



Verwendung

Ursprünglich wurde es benutzt, um HTML-Dokumente von Webservern anzufordern. Heute hat es aber mehrere Zwecke:

- Browser fordern nun auch andere Medien an, z.B. Text, Bilder, Programmcode usw.
- Anwendungsprogramme nutzen HTTP, um Dateien und Updates von entfernten Servern zu laden

- Bei Machine-to-Machine-Kommunikation wird http als Protokoll für die Kommunikation zwischen Webservices verwendet
- Medienplayer nutzen HTTP
- Zugriffe auf eine Datenbank im Web können über HTTP erfolgen

GET vs. POST

GET wird benutzt, um Daten von einer spezifischen Ressource anzufragen. Es ist Teil der URL.

POST wird hingegen benutzt, um Daten zu einem Server zu senden um eine Ressource zu erzeugen/updaten. Die Daten, welche zum Server gesendet werden, sind im Body des http-Request gespeichert.

	GET	POST
Beschränkungen auf Daten	Es ist Teil vom URL, deswegen hat es eine Limite (max. Länge URL: 2048 Zeichen) und es dürfen nur ASCII-Zeichen verwendet werden	Hat keine Beschränkungen auf die Länge oder Datentype der Daten
BACK-button / Reload	Keine Gefahr	Daten werden neu eingereicht und der Browser sollte eine Warnmeldung geben
Sicherheit	Weniger sicher als POST, da die Daten Teil der URL (für jeder sichtbar) sind und im Suchverlauf stehen	Ein bisschen sicherer als GET, weil Daten nicht im Verlauf oder auf Webserver Logs stehen

PUT vs. POST

Wie die POST Methode wird auch die PUT Methode verwendet, um Daten zu einem Server zu schicken eine Ressource zu erzeugen/updaten. Der Unterschied der beiden Methoden liegt darin, dass die PUT Methode idempotent ist. D.h. falls man mehrmals die gleiche PUT Anfrage schickt, wird das Resultat immer gleich bleiben. Falls man jedoch das gleiche mit einer POST Anfrage macht, kann das Nebenwirkungen geben, da dieselbe Ressource mehrmals erstellt wird.