

# TCP / IP – Teil 3: HTTP – Protokoll

Das HTTP (Hypertext Transfer Protocol) baut auf das TCP auf, bildet in seiner Übertragung jedoch weitreichende eigene Konzepte, als das die Übertragung über TCP/IP.

Ebenfalls ist das HTTP ist heute das Standardprotokoll um Webseiten im Webbrowser darzustellen, welches sich laut W3C im 1.1 Standard befindet. HTTP/2 befindet sich aber bereits in Entwicklung.

Eine Ausführung sieht im kurzem immer so aus:

1. Server wartet über einen Port i.d.R. Port 80 auf eine Anfrage
2. Client sendet über POST oder GET eine Anfrage (Request) an den Server
3. Falls dieser Erreichbar ist sendet der Server eine Antwort (Response) zurück

## URL

Eine URL kann z.B. so aussehen:

[crayon-67418f02c0aea301266728/]

http – ist das verwendete Protokoll

www.devandy.de – hostname oder Domain

/meinOrdner/2015-04-01/ – Verzeichnis wo sich die Datei auf dem Server befindet

datei01.php – die Datei an die ein Request gesendet wird

?vorname=peter&nachname=lustig – Querystring, der immer mit einem ? beginnt. vorname= und nachname= bilden dabei die

variablen und peter und lustig sind die darin enthaltenen werte.

## **Übertragungsmethode GET**

Die Übertragung findet über die URI, über den Querystring statt. Die Anweisungen aus der method sind in der URL sichtbar und können für spätere selbe Zwecke gespeichert werden

## **Übertragungsmethode POST**

Die Übertragung über POST hingegen kann man nicht bookmarken, da die Übertragung im Content geschieht. Dabei werden die Informationen im Header weiter gegeben. Ein Dateiupload ist z.B. nur mit Post möglich, ein Textfeld mit einem Roman macht auch nur über Post Sinn, weil dieser nicht begrenzt ist.

### **1. Schritt: Server wartet auf Anfrage**

In der Regel sind die Serverprogramme wie der IIS Server oder Apache standardisiert eingerichtet, so dass immer über Port 80 gelauscht wird. Man kann dies aber Serverseitig auch auf einen anderen Port umlegen.

### **2. Schritt: Client Request**

Der Client wie der Webbrowser sendet an den Host über das http Protokoll eine Anfrage in Form einer URL.

Dabei enthält diese Anfrage 2 Teile. Zum einen die Anforderungszeile (URL) und zum anderen einen Header, der so aussehen kann:

```
[crayon-67418f02c0af0499489496/]
```

Die erste Zeile beinhaltet mit GET die Methode über den Request, index.html die zu aufrufende Datei und dann die Protokollart und Versionsnummer.

Darunter folgen weitere Informationen wie Ursprungsland, Bilder erlauben, Browsertyp und Versionsnr, Referer (welche Seite man vorher besucht hat), usw.

### 3. Schritt Server Response

Ist der Server erreichbar, sendet er eine 3 teilige Antwort wieder. Diese besteht aus

#### 1. Statuszeile

[crayon-67418f02c0af2397379968/]

bestehend aus dem Protokoll, Versionsnummer, Statusnummer und Status.

#### 2. Header

Der Header beinhaltet das Date, Datum und Uhrzeit wann die Antwort geschickt wurde, Server Name und Versionsnummer des Webservers, Content-Length, Länge des Nachrichten-Body in Byte usw.

#### 3. Body

Darin befindet sich der eigentliche HTML Code, der dann im Browser angezeigt wird.

Beispiel Projekt in C#, ein Http Server:

HTTP Server

*Quellen:*

*SelfPHP Get/Post*

*Andreas Olesch*