

## OPDRACHT 8. HTTP

In deze opdracht gaan we het HyperText Transfer Protocol ([HTTP](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)<sup>1</sup>) verkennen.

HTTP is een Internet applicatie protocol. Het is het werkpaard van het World Wide Web. HTTP wordt gebruikt door Browsers om webpagina's (en plaatjes die daarin staan) op te halen van Websites. Het wordt echter ook gebruikt om vanuit software programma's remote Webservices te benaderen.

HTTP is een client/server protocol. Een HTTP server (http daemon) luistert op een poort (normaliter poort 80) naar requests van clients. Elke request wordt beantwoordt in de vorm van een respons. Een HTTP Client die een bepaald resource in een bepaald formaat (b.v. HTML, XML of JSON) wil ontvangen maakt een connectie naar de betreffende host/port, stuurt een http request, en wacht op de respons van de HTTP Server.

### 1. Voorbereidingen

Bestudeer het HTTP protocol. Goede startpunten zijn de volgende links :

- HTTP Made Really Easy  
<http://www.jmarshall.com/easy/http/>
- How the web works: HTTP and CGI explained  
<http://www.garshol.priv.no/download/text/http-tut.html>
- Learn REST: A Tutorial  
<http://rest.elkstein.org/>

Installeer (sudo apt-get install) de volgende tools:

- [PuTTY](http://www.chiark.greenend.org.uk/~sgtatham/putty/)<sup>2</sup>  
PuTTY is een flexibele Telnet and SSH client. Het wordt normaliter gebruikt als terminal emulator, maar je kunt er ook, in raw mode, mee communiceren met een server.
- [cURL](http://curl.haxx.se/)<sup>3</sup>.  
"cURL groks URLs", d.w.z. met dit tool kun je resources ophalen van willekeurige URL's

### 2. Taken

#### 2.1 HTTP handmatig

Eerst gaan we handmatig via PuTTY met een webserver communiceren.

Doe de volgende handelingen om een http request via PuTTY te versturen naar een HTTP server:

- Enter 'Host Name (or IP address)': e.g. [www.example.com](http://www.example.com)  
(enter 'localhost' if your server is running on your own PC)
- Enter 'Port': 80 (could be different on some host)
- Select 'Connection Type': Raw
- Select 'Close window on exit': Never
- Press 'Open'

Doe de volgende experimenten en beantwoordt de vragen:

1. Open your favourite browser and enter the URL [www.example.com](http://www.example.com). Look at the URL field in the Browser. What happens?
2. Let us do what the browser does, but manually, step by step. Connect to [www.example.com](http://www.example.com) (with PuTTY). Type the following two lines in the PuTTY window and after that hit the <enter> key twice:

```
GET /index.html HTTP/1.1
Host: www.example.com
```

What respons do you get? What does it mean?

3. Write down and execute the logical next GET query (manually).

<sup>1</sup> HTTP: [http://en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol)

<sup>2</sup> PuTTY: <http://www.chiark.greenend.org.uk/~sgtatham/putty/>

<sup>3</sup> cURL: <http://curl.haxx.se/>

Do we need any more steps or are we finished now, i.e. did we get to the same page as the browser?

## 2.2 HTTP via cURL

We gaan nu cURL gebruiken om resources op te halen bij een HTTP server.

Hier kun je informatie vinden hoe cURL te gebruiken:

- <http://curl.haxx.se/docs/httpscripting.html>
- <http://www.codediesel.com/tools/6-essential-curl-commands/>
- <http://www.thegeekstuff.com/2012/04/curl-examples/>

Note: cURL built-in help kun je opvragen met het commando 'curl --help'

Doe nu de volgende opdrachten ( en test of je oplossingen ook echt werken!):

- Define a cURL command to do everything (except nice display) that the browser did in assignment 1.1.
- Define a cURL command to download and save the Google logo from the main page of the website [www.google.com](http://www.google.com). What extension (based on image type) should the downloaded file have? How can you find that in the HTTP response header?
- Define a cURL command that does the same POST as the HTML-form in the following w3schools.com example:  
[http://www.w3schools.com/tags/tryit.asp?filename=tryhtml\\_form\\_method\\_post](http://www.w3schools.com/tags/tryit.asp?filename=tryhtml_form_method_post)
- Define a cURL command to get the welcome page of the website <https://fhict.fontys.nl> (do not include password in the command, just the username!)

## 2.3 Communicatie met een REST Webservice

De [REST](#)<sup>4</sup> webservices style wordt vanwege zijn eenvoud steeds populairder. Er is geen complexe software voor nodig, maar eenvoudige HTTP client software en een JSON of XML parser. Allerlei Google services en de meest bekende social media websites kun je zo benaderen via RESTful WebAPI's.

Je opdracht is als volgt:

Zoek een geschikte online webservice met een REST API en doe daarmee het volgende experiment:

- Construeer een cURL commando om een bepaalde API call te doen. Doe de call vanaf de commandline.
- Schrijf vervolgens een [libcurl](#)<sup>5</sup> gebaseerd console program in C om hetzelfde te doen. Hint: cURL heeft een '`--libcurl <file>`' option.
- Doe en beschrijf wat experimenten, d.w.z. enkele API requests en bijbehorende JSON output.

Een voorbeeld webservice die je kunt gebruiken is de "The Google Geocoding API" van Google Maps. Daarmee kun je adressen opzoeken. De specificatie van die API kun je vinden op:

<https://developers.google.com/maps/documentation/geocoding/>

Meer informatie over het JSON formaat en bijbehorende libraries kun je vinden op: <http://json.org/>

## 3. Opleveren

Je dient een document op te leveren waarin je de bovenstaande experimenten uitwerkt. Voeg, waar nodig, screendumps toe om een en andere duidelijk te maken. Lever ook de bijbehorende source code in van de programma's die je gemaakt of aangepast hebt.

---

<sup>4</sup> REST: [http://en.wikipedia.org/wiki/Representational\\_state\\_transfer](http://en.wikipedia.org/wiki/Representational_state_transfer)

<sup>5</sup> Libcurl: <http://curl.haxx.se/libcurl/>