

ISEN-LILLE

NETWORK PROGRAMMING

PROJET

---

# HTTP Protocol

---

*Étudiants :*

Hugo CAILLE

Louis GÉRARD

Aymeric MASSE

Édouard SAMYN

*Référent :*

Julien ROLAND

Année 2015 - 2016

# 1 Overview

HTTP stands for Hypertext Transfer Protocol and is a protocol for client/server. Its variant HTTPS, provides a security layer on top of it

The standard port for HTTP is 80, and the standard for HTTPS is 443

This protocol is part of the application layer of the OSI model. There are several kinds of HTTP requests, which are defined by verbs, which are GET, POST, DELETE, PUT, HEAD, OPTIONS, CONNECT, TRACE, PATCH

The most commonly used verbs are GET and POST for web servers, however DELETE and PUT verbs are used mainly with Rest APIs.

## 2 HTTP Request

In order to access a resource from a web server, the client must send a GET request to the server. This request has a header which contains several fields :

- First is the type of request, in our case it will be GET, but it can also be POST as far as web servers are concerned. Nevertheless, the HTTP Protocol also supports PUT and DELETE requests.
- Host : The web server address
- User-Agent : Is used to identify the client architecture
- Referer : Previous address visited
- Accept : Refers to the MIME type accepted by the client
- Accept-Encoding : Various encodings accepted by the client
- DNT : Do not track
- Connection : when is set to keep-alive, will tell the server not to close the socket immediately, which is useful for performance, because we don't need to open the stream again
- Pragma and Cache-Control : Is used for cache policy

In the case of a GET request, the body of the request is empty. When the request type is POST, the client sends its data through the body of the request. However, in our case, the web server will only handle GET request.

### 3 HTTP Response

Once the server has received a request, it will reply to the client. This response has also several headers which we are going to describe :

- Status code : An integer to define the status of the response, for example 200 for OK and 404 for Not Found
- Content Encoding : Self-explanatory, It is the encoding used for the content
- Cache-Control : Tells the client how long it should keep the page in its cache
- Content-Type : The type of content embedded in the reply, for example text/html
- Date : The date and time of the reply
- Etag : Hash useful for the client's cache, used to detect potential differences
- Expires : The date at which the cache should be cleaned
- Last-Modified : The last time the page was changed
- Server : Identifies the server software and potentially its version, which, if outdated, may lead to use of vulnerability against the server
- Vary :
- x-Cache : Is not a standardized header, it is used to tell the client that the page was delivered by a Content Delivery Network(CDN) and not origins server
- Content-Length : Size of the body in bytes

The body of the reply will then be the content itself, for example if the type is text/html, the body will be html code.