

Thibaut LEFRANCOIS D2 - 23/02/23

Practical part SAE 2.03 week 8

HTTP protocol and dialogue with a web server**Question 1 :**

Sources :

- <https://www.ietf.org/standards/rfcs/>
- <https://en.wikipedia.org/wiki/>
- <https://www.webopedia.com/TERM/R/RFC.html>

1.1) A RFC (Request for Comments) is a type of document published by a group called the Internet Engineering Task Force (IETF). RFC documents describe technical specifications, protocols, and procedures for Internet-related technologies. They are collaboratively created by engineers and technology experts working on the development of the Internet. RFCs are considered unofficial standards for Internet-related technologies. They are published online and are available for free to everyone.

1.2) The RFC that defines version 1.0 of the HTTP protocol is RFC 1945. This RFC was published by the Internet Engineering Task Force (IETF) in May 1996. It specifies the syntax and semantics of the HTTP 1.0 protocol, including request and response messages, status codes, and headers. RFC 1945 was replaced in 1999 by RFC 2616, which defines version 1.1 of the HTTP protocol.

1.3) The RFCs that define the version 1.1 of the HTTP protocol are the following:

- RFC 2068: "Hypertext Transfer Protocol -- HTTP/1.1", published in January 1997. This RFC defined the first version of the HTTP/1.1 protocol.
- RFC 2616: "Hypertext Transfer Protocol -- HTTP/1.1", published in June 1999. This RFC brought clarifications and improvements to the HTTP/1.1 protocol.
- RFC 7230 to 7235: These RFCs were published in June 2014 and define the updated specifications for HTTP/1.1, replacing the old RFCs 2616 and 2617.

1.4) According to the HTTP 1.0 RFC, here are the roles of the GET and POST methods, the Content-Type, Last-Modified, and User-Agent header directives, and the 304 and 403 return codes:

- GET and POST methods: Both methods allow requesting a resource from the server. The GET method is used to retrieve a resource while the POST method is used to send data to the server.
- Content-Type header directive: This directive is used to specify the type of content being sent to the server. It is typically used for POST requests and responses.
- Last-Modified header directive: This directive is used to indicate the date and time when the resource was last modified.
- User-Agent header directive: This directive is used to identify the browser or client software being used to send the request.

- 304 return code: This code is used to indicate that the requested resource has not been modified since it was last retrieved. The browser can then use its cached copy rather than retrieving the resource from the server again.
- 403 return code: This code is used to indicate that the server has denied access to the requested resource. This may be due to a configuration error or access restriction.

Question 2 :

- GET: This method is used to retrieve information from the server. It allows requesting a specific resource, such as an HTML page or an image, using a URL. Test the query with the command `GET / HTTP/1.0 :`

```
HTTP/1.1 200 OK
Date: Wed, 22 Feb 2023 14:18:37 GMT
Server: Apache/2.4.7 (Ubuntu)
Last-Modified: Tue, 06 Sep 2022 07:19:16 GMT
ETag: "87b-5e7fd0391c895"
Accept-Ranges: bytes
Content-Length: 2171
Connection: close
Content-Type: text/html
```

- OPTIONS: This method allows requesting information about the options and/or features available for a resource, such as the HTTP methods supported by the server for that resource.

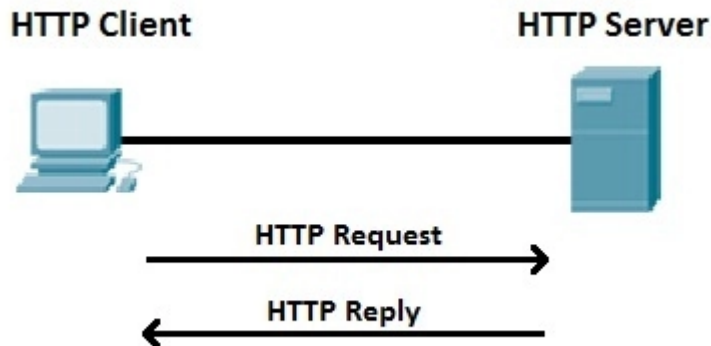
```
HTTP/1.1 200 OK
Date: Wed, 22 Feb 2023 14:40:41 GMT
Server: Apache/2.4.7 (Ubuntu)
Allow: GET,HEAD,POST,OPTIONS
Content-Length: 0
Connection: close
Content-Type: text/html
```

- HEAD: This method is similar to GET, but it only returns the HTTP headers of the response, without including the response body. This can be useful for checking header information, such as content type or modification date, without having to download the entire resource content.

```
HTTP/1.1 200 OK
Date: Wed, 22 Feb 2023 14:43:07 GMT
Server: Apache/2.4.7 (Ubuntu)
Last-Modified: Tue, 06 Sep 2022 07:19:16 GMT
ETag: "87b-5e7fd0391c895"
Accept-Ranges: bytes
Content-Length: 2171
Connection: close
Content-Type: text/html
```

Question 3 :

Diagram of how the HTTP protocol works :

**Question 4 :**

To compare the HTTP headers observed via the browser with those observed in question 2, I opened the Firefox browser development tools with F12 and looked at the information provided in the "Network" tab.

I then created new requests with the following HTTP methods:

```
GET
http://serverrx.univ-lr.fr/?=http://serverrx.univ-lr.fr/
Status
200
OK
VersionHTTP/1.1
Transferred (2.12 KB size)

Response Headers (248 B) :

Accept-Ranges : bytes
Connection : close
Content-Length :2171
Content-Type :text/html
Date : Thu, 23 Feb 2023 07:45:38 GMT
ETag : "87b-5e7fd0391c895"
Last-Modified : Tue, 06 Sep 2022 07:19:16 GMT
Server : Apache/2.4.7 (Ubuntu)
```

```
HEAD
http://serverrx.univ-lr.fr/?=http://serverrx.univ-lr.fr/
Status
200
OK
```

```
VersionHTTP/1.1
Transferred (0 B size)

Response Headers (248 B) :

Accept-Ranges : bytes
Connection : close
Content-Length : 2171
Content-Type : text/html
Date : Thu, 23 Feb 2023 07:44:10 GMT
ETag : "87b-5e7fd0391c895"
Last-Modified : Tue, 06 Sep 2022 07:19:16 GMT
Server : Apache/2.4.7 (Ubuntu)
```

```
OPTIONS
http://serverrx.univ-lr.fr/?=http://serverrx.univ-lr.fr/
Status
200
OK
VersionHTTP/1.1
Transferred (0 B size)

Response Headers (180 B) :

Allow : GET,HEAD,POST,OPTIONS
Connection : close
Content-Length : 0
Content-Type : text/html
Date : Thu, 23 Feb 2023 07:46:05 GMT
Server : Apache/2.4.7 (Ubuntu)
```

We then notice that the HTTP headers are similar because they have the same Etag (87b-5e7fd0391c895), the same server version (Apache/2.4.7 (Ubuntu)), the same last modification date (Tue, 06 Sep 2022 07:19:16 GMT) and the same content size (2171) and more.

Question 5 :

To compare the HTTP headers observed via the browser with those observed in question 4 for serverrx.univ-lr.fr, I opened the Firefox browser development tools with F12 and looked at the information provided in the "Network" tab.

```
GET
http://sae0203.asc2.org/?=http://sae0203.asc2.org/
Status
503
Service Unavailable
VersionHTTP/1.1
Transferred6.75 KB (6.41 KB size)
```

Response Headers (248 B) :

Connection : keep-alive
Content-Language : en-us
Content-Length : 6567
Content-Type : text/html
Date : Thu, 23 Feb 2023 08:18:20 GMT
Mime-Version : 1.0
Server : squid/3.3.13
Vary : Accept-Language
Via : 1.1 wwwcache.univ-lr.fr (squid/3.3.13)
X-Cache : MISS from wwwcache.univ-lr.fr
X-Squid-Error : ERR_DNS_FAIL 0

HEAD

http://sae0203.asc2.org/?=http://sae0203.asc2.org/
Status
503
Service Unavailable
VersionHTTP/1.1
Transferred 349 B (0 B size)

Response Headers (248 B) :

Connection : keep-alive
Content-Language : en-us
Content-Length : 6568
Content-Type : text/html
Date : Thu, 23 Feb 2023 08:18:20 GMT
Mime-Version : 1.0
Server : squid/3.3.13
Vary : Accept-Language
Via : 1.1 wwwcache.univ-lr.fr (squid/3.3.13)
X-Cache : MISS from wwwcache.univ-lr.fr
X-Squid-Error : ERR_DNS_FAIL 0

OPTIONS

http://sae0203.asc2.org/?=http://sae0203.asc2.org/
Status
503
Service Unavailable
VersionHTTP/1.1
Transferred 349 B (0 B size)

Response Headers (248 B) :

Connection : keep-alive
Content-Language : en-us

```
Content-Length : 6618
Content-Type : text/html
Date : Thu, 23 Feb 2023 08:18:20 GMT
Mime-Version : 1.0
Server : squid/3.3.13
Vary : Accept-Language
Via : 1.1 wwwcache.univ-lr.fr (squid/3.3.13)
X-Cache : MISS from wwwcache.univ-lr.fr
X-Squid-Error : ERR_DNS_FAIL 0
```

So we notice that the HTTP headers are not similar to those found in the previous question on serverrx.univ-lr.fr because they have different sizes, different servers (squid/3.3.13 instead of Apache2) and unavailable.