L5 HTTP AND THE WEB

HTTP: HyperText Transfer Protocol

- HTTP is the protocol for retrieving HTML files from servers (Also Image, Sound, Video)

- Implemented in servers(Nginx, Apache, MSFT IIS)
- Clients (Chrome, MSFT Edge, Apple Safari)

Mutiple Usage

- Web Brower

- Video streaming via DASH on YouTube.com

- REST (Representational state transfer)

  Representational state transfer REST is a software architectual style that defines a set constraints to be used for creating Web services

- Chat apps like Slack

HTTP Request

- Requests and responses both look like:

  START_LINE <CRLF> ← carriage-return-line-feed

  MESSAGE_HEADER <CRLF>

  <CRLF>

  MESSAGE_BODY <CRLF>

- The first line (START LINE) indicates whether this is a request message or a response message.

There is a link that may be a little bit clearer.

Request Content

1. **Request**

- The operation (called method) to be performed
- The web page the operation should be performed on

- The version of HTTP being used.

Examples:

• GET /index.html HTTP/1.0 • GET /images/catimg23.jpg HTTP/1.1 •. GET /contracts/contract3.txt HTTP/1.1

2. **Request Header**

After the start line are request headers

- Text based, key and value separated by a colon

Example 1:

GET /index.html HTTP/1.0 User Agent: Firefox 23.3.1

Example 2:

GET /images/cat2.jpg HTTP/1.1 Host: www.cs.ucsd.edu User Agent: Chrome 12.1

HTTP Response

Also begins with a single START LINE.

- The version of HTTP being used
- A three digit status code
- Text string giving the reason for the response.

Example:

HTTP/1.1 200 OK Content Type: text/html Content Length: 291

Get ur hands Dirty!

Here we try to retrieve a webpage with http request.

We use **telnet** in command line. Telnet is a computer protocol that was built for interacting with remote computers.

```
borabora:~ gmporter$ telnet oec-vmweb09.ucsd.edu 80
Trying 132.239.8.67...
Connected to oec-vmweb09.ucsd.edu.
Escape character is '^]'.
GET /index.html HTTP/1.0
```

Request

```
HTTP/1.1 200 OK
Date: Mon, 12 Jan 2015 19:36:37 GMT
Server: Apache/2.2.22 (Ubuntu)
Last-Modified: Thu, 28 Feb 2013 17:35:36 GMT
ETag: "fc7b21-a-4d6cc51858aec"
Accept-Ranges: bytes
Content-Length: 10
Vary: Accept-Encoding
Connection: close
Content-Type: text/html

It works!
Connection closed by foreign host.
borabora:~ gmporter$
```

Response

HTTP Request Code

HTTP RESPONSE CODES

Code	Type	Example Reasons
1xx	Informational	request received, continuing process
2xx	Success	action successfully received, understood, and accepted
3xx	Redirection	further action must be taken to complete the request
4xx	Client Error	request contains bad syntax or cannot be fulfilled
5xx	Server Error	server failed to fulfill an apparently valid request

- For project 1:
  - 200: OK
  - 400: Client Error
  - 403: Forbidden
  - 404: Not Found

Connection: Reuse or Create a new one?

1. **HTTP/1.0: Create a new One**

   Easier, but overhead is huge

2. **HTTP/1.1: Reuse connections**

   More complex to framing and parsing

HTTP PIPELINING (VERSION HTTP/1.1)

- HTTP/1.0 opened a new connection for every data item it retrieved

- Overhead in establishing a new connection to the same server over and over again

- HTTP/1.1 Persistent Connections

  - Reuse connection over many requests/responses

  - But more complex in terms of framing/parsing

    - How to know when one request ends and the next begins?

    - This is part of the 1.1 spec

Pipelined request/response
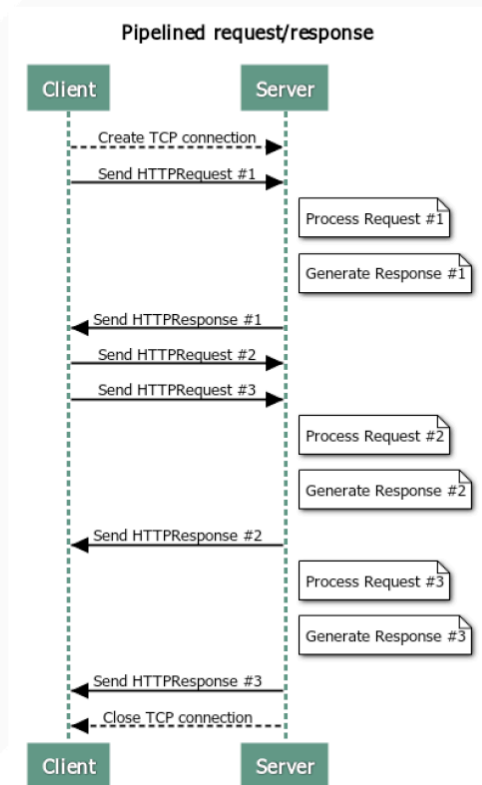
Required Header for host and server

Host: Request Header

Indicates the name of the server you are accessing

Server: Response Header

1. **Server:**

   Identifies the server

   Eg: Server: Apache/2

2. **Content Length:**

How many octets (byte) in the response

3. **Content Type:**

- text/html
- image/jpeg
- image/ png

Curl Learning

curl -v -o /dev/null http://

-v: verbose, useful for debugging

-o: output file name

```
(base) ➜  Downloads curl -o trial.html http://www.baidu.com
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  2381  100  2381    0     0  33069      0 --:--:-- --:--:-- --:--:-- 33535
```