HTTP Basics

Marcin Chodkowski

Welcome

HTTP Basics

HTTP - intro

HTTP Basics

TCP/IP

HTTP Basics

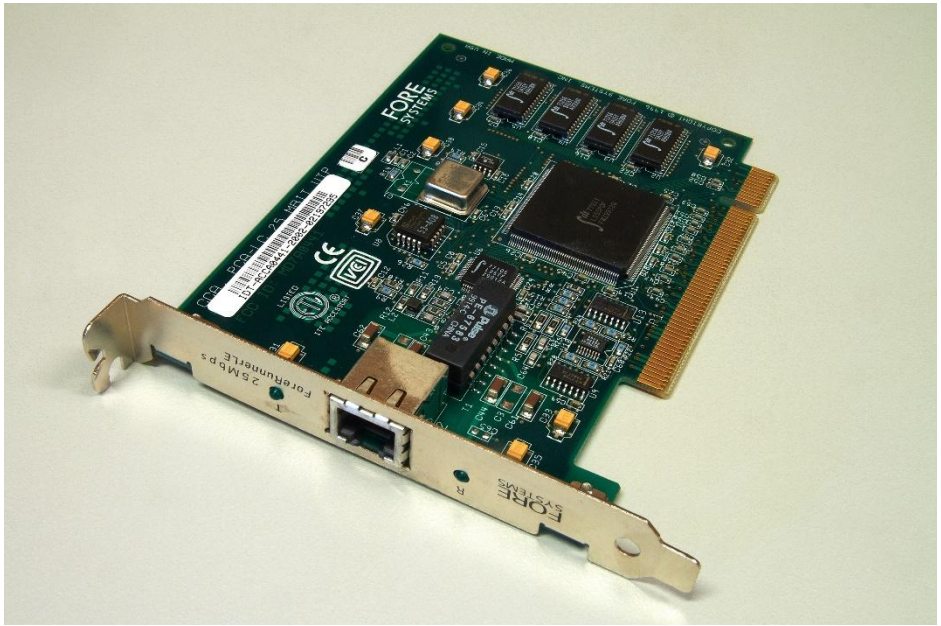
In this video you will learn:

- How to connect to a network?
- About TCP/IP network types
- With what equipment we build TCP/IP networks?
- What are: server, router, and application server?

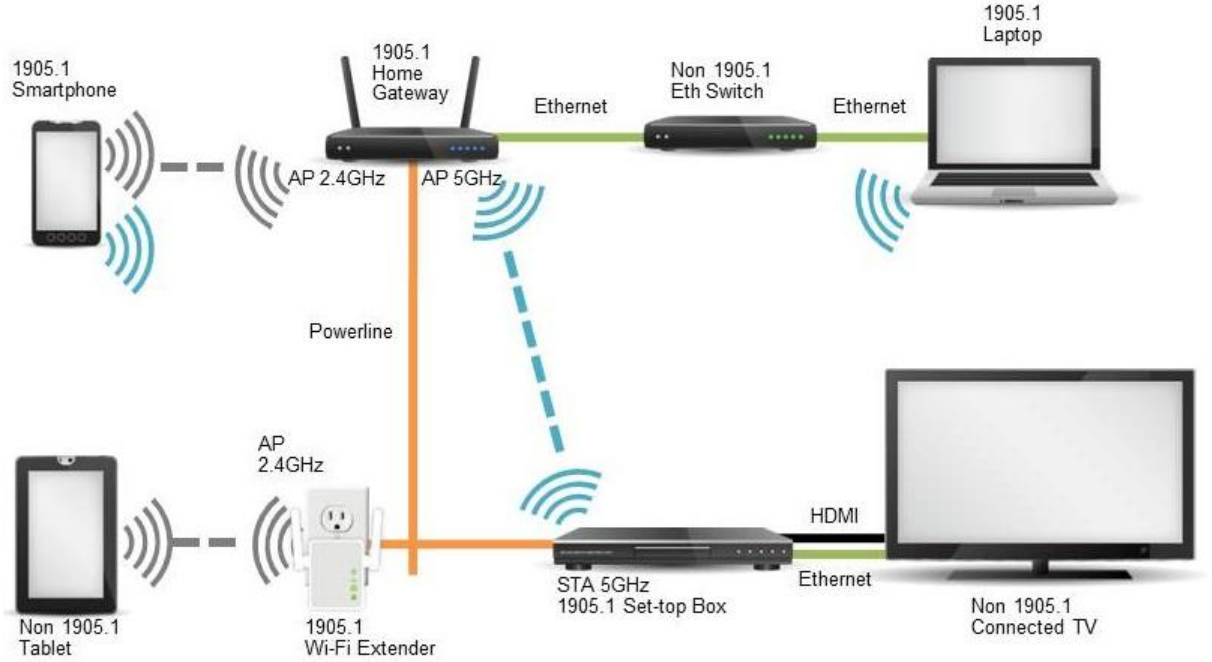
TCP/IP

- Network card
  - MAC (**98:01:a7**:ad:14:49)

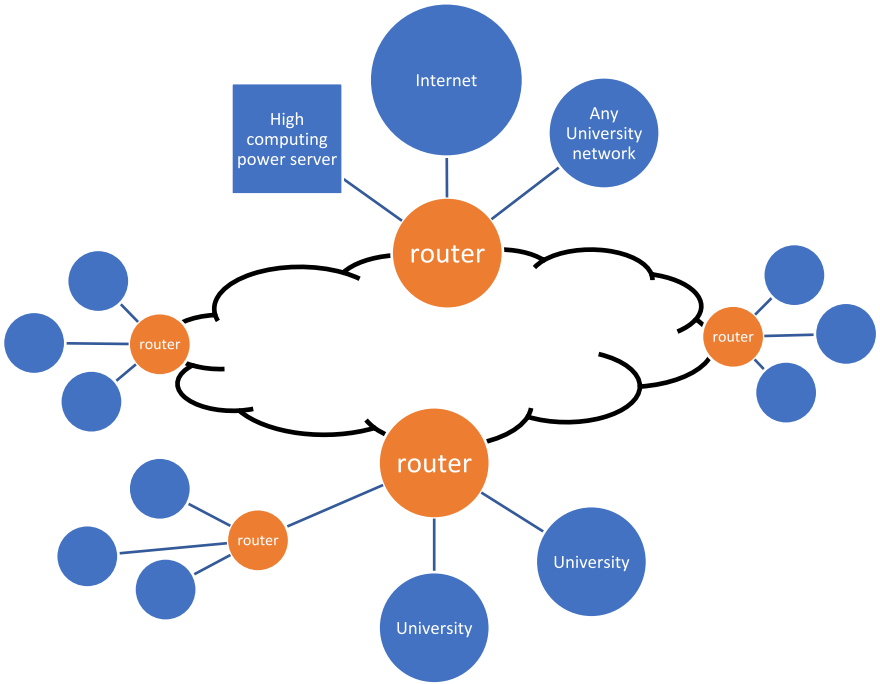
MAC => 192.168.0.2



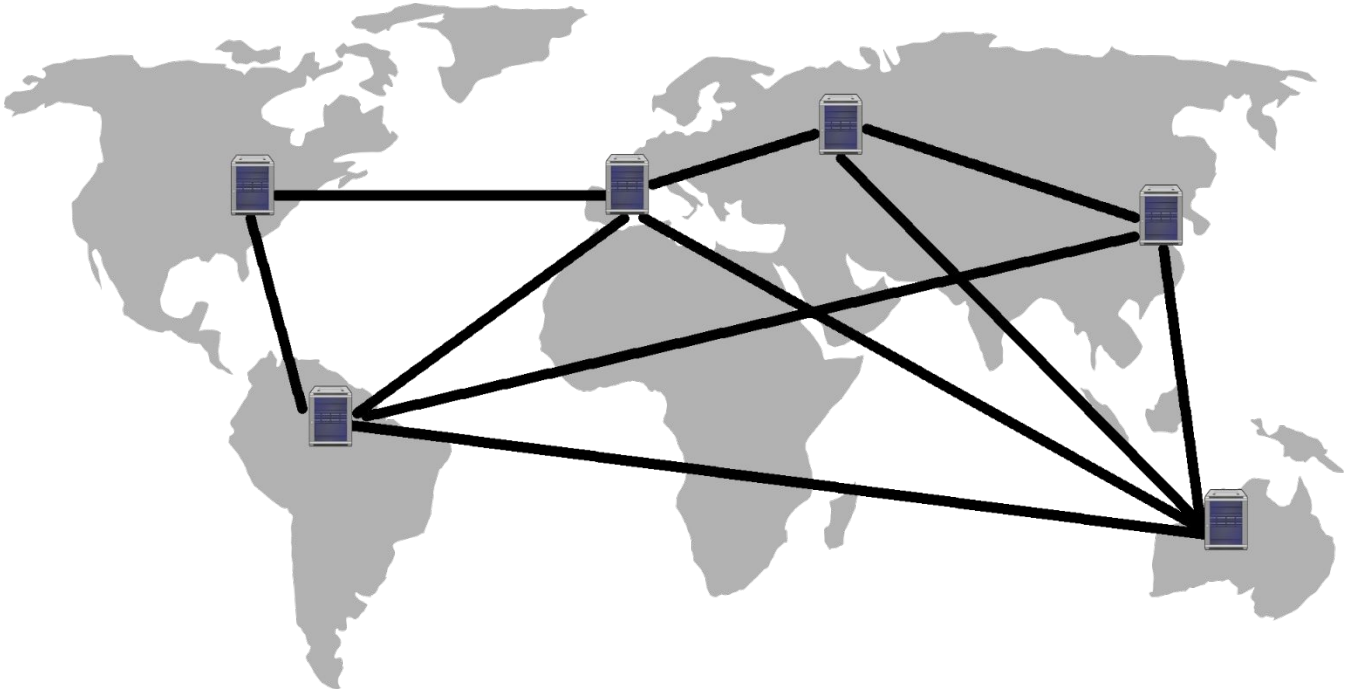
Network types - LAN



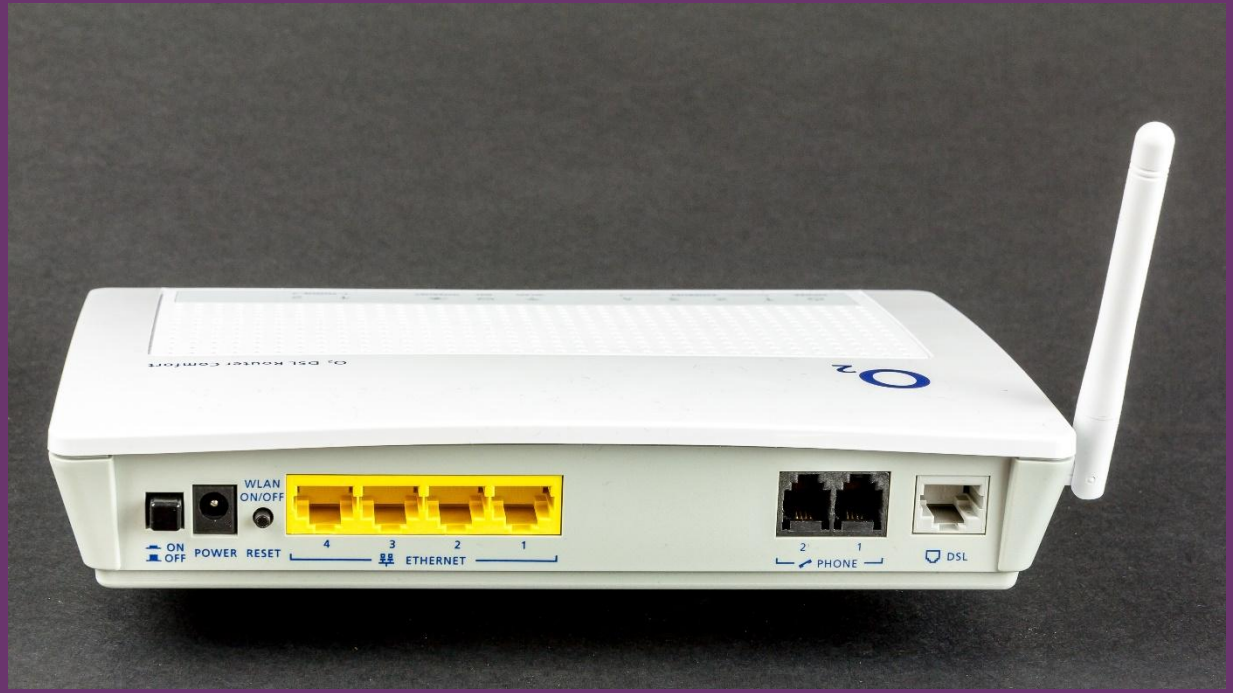
Network types - MAN



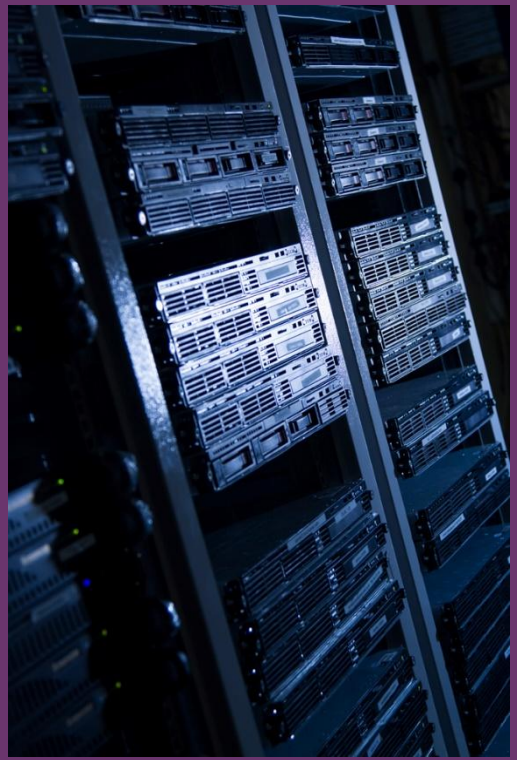
Network types - WAN



Router



Server



- Physical machine
- Powerful processor
- Very weak graphic card
- Several network ports

Application server

Application that waits for requests

Examples of application servers for:

- www pages - Apache, Ngnix, Tomcat
- ftp - ProFTPd
- email - Qmail

After watching this video you know:

- In order to connect to a network, you need a network card.
- That there are there TCP/IP network categories: LAN, MAN, and WAN.
- That networks are created by using routers.

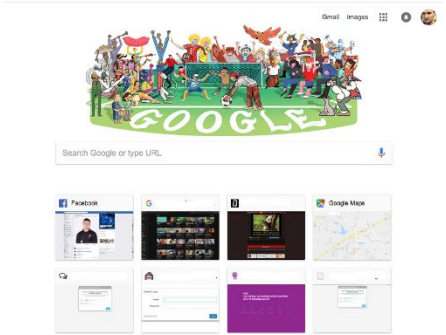
DNS

HTTP Basics

In this video you will learn:

- What is DNS?
- How many servers does DNS consist of?
- Who manage DNS servers

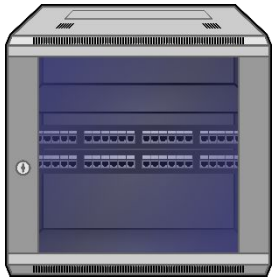
DNS



Visited website

Asks what IP address has a domain called example.com

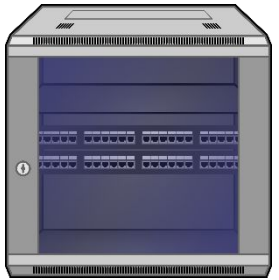
93.184.216.34



DNS Server

Request

Response



Server waits for Request

After watching this video you know:

- That DNS is a dictionary of domains and corresponding IPs.
- That there are 13 root DNS servers.
  - In addition, there are complementary servers.
- Every domain has assigned two DNS servers: main and complementary.
- Every server (main and complementary) has info about the other one.

DNS – tracert - demo

HTTP Basics

In this video you will learn:

- How to check which servers in the world your request to google.com server goes through?
- How to show on a map which servers your request goes through?

After watching this video you know:

- How to check which way your request gets to a server. You will know how to use tracert command.
- How will know the page on which you can check what route your request gets to a server.

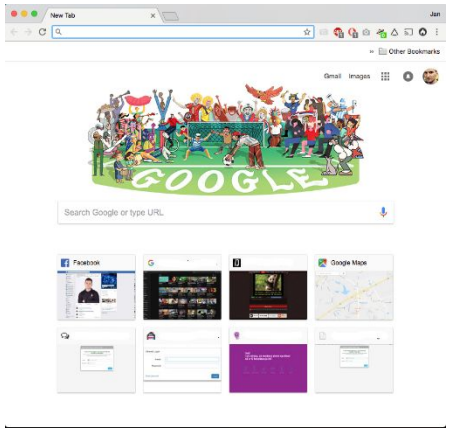
Communication between Client and Server

HTTP Basics

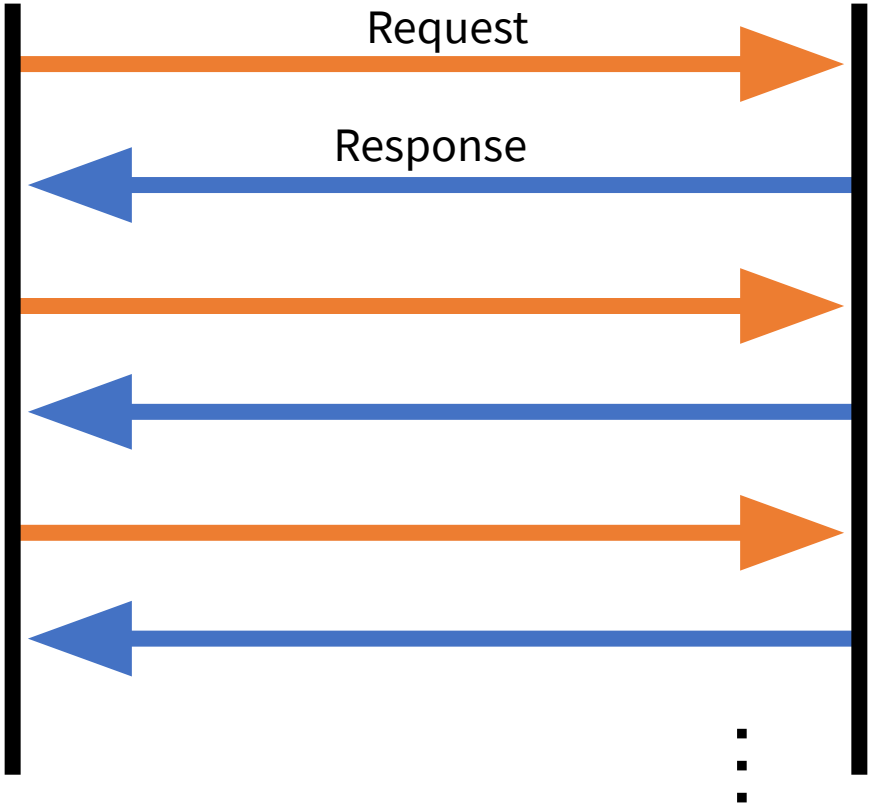
In this video you will learn:

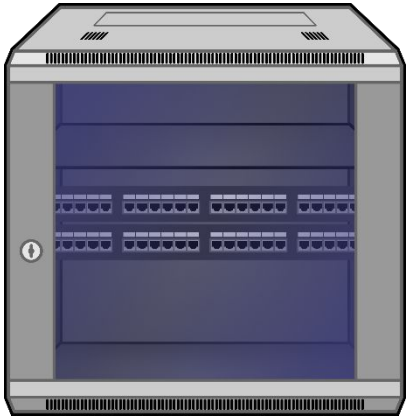
- How the communication looks like between a client and a server?

How the communication looks like between a client and a server?



Visited webpage





Server waits for Request

After watching this video you know:

- That a client sends requests and a server sends responses for that requests.

Request

HTTP Basics

In this video you will learn:

- From what parts a request consists of.

Request

GET /?query=London HTTP/1.1

Host: example.com

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:60.0) Gecko/20100101 Firefox/60.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Language: en-US;q=0.7,en;q=0.3

Accept-Encoding: gzip, deflate

Connection: close

[Empty line]

After watching this video you know:

- That a request consists of:
  - http method
  - urn
  - http version
  - host
  - headers
  - body

Response

HTTP Basics

In this video you will learn:

- From what parts a response consists of.

Response

HTTP/1.1 200 OK

Host: example.com

Connection: close

X-Powered-By: PHP/5.6.33

Access-Control-Allow-Origin: *

Content-type: text/html; charset=UTF-8

Body

After watching this video you know:

- That a request consists of:
  - http version
  - http code with description
  - host
  - response headers
  - body

HTTP methods

HTTP Basics

In this video you will learn:

- What are HTTP methods?
- How many HTTP methods can be distinguished?
- With what HTTP methods the data can be sent to a server?

Methods of communication - HTTP

Method	Request Body	Response Body	Application
GET	**no**	yes	Download / display resource
POST	yes	yes	Send data to a server as a key-value form **or a file**
PUT	yes	yes	Update data on a server as a key-value form.
DELETE	yes	yes	Remove resource from a server
HEAD	**no**	**no**	It is like GET, but returns only headers - without body

After watching this video you know:

- That the example http methods were GET, POST, PUT, DELETE, HEAD.
- That the http methods that can send data to a server are:
  - POST
  - PUT
  - DELETE

Headers

HTTP Basics

In this video you will learn:

- What are headers?
- What are examples of headers?

Request Headers

Header name	Explanation	Example
Accept	Sets data type which is accepted by application	Accept: text/plain
Authorization	Data used for authorization on the server site	Authorization: Basic QWxhZffghyssaGVuIHRRtgtZQ==
Cookie	Cookie content that client received from a server	Cookie: name=value
Content-type	Determines data type in a request body	Content-Type: application/xhtml+xml; charset=utf-8
Host	Domain consistent with URI	Host: sda.jszewczak.com
Content-Length	Determines size of a request(in bytes)	Content-Length: 12

Response Headers

Header name	Explanation	Example
Access-Control-Allow-Origin	Determines which domains can send requests to a server	Access-Control-Allow-Origin: *
Allow	HTTP methods which are supported by a server	Allow: GET, HEAD
Expires	Determines time when the response is considered outdated by the browser	Expires: Fri, 02 Dec 1999 17:00:00 GMT
Set-Cookie	Sets cookie value on the browser site	Set-Cookie: User=Max; sessio_id=3601;
Content-Encoding	Encoding used by a server to send the response	Content-Encoding: gzip

After watching this video you know:

- That headers are an extra information send to and from a server.
- A few examples of headers and you will know that there is an unlimited number of header names.

URL, URI

HTTP Basics

In this video you will learn:

- Of what parts does the URI consist of?
- How a very expanded version of the URI looks like.

URI structure

<table>
<tr><td colspan="2">URI</td></tr>
<tr><td>https://github.com/</td><td>login</td></tr>
<tr><td>URL</td><td>URN</td></tr>
</table>

http://login:password@sda.jszewczak.com:80/res/2/second?p1=1&p2=2#paragraph1

<protocol>://<login>:<password>@<server name>:<port>/<path to resource>?<optional parameter>#<anchor to the html fragment>

After watching this video you know:

- That the URI in the basic form consists of a protocol and a server name.
- That the most expanded version of the URI consists of parts like:
  - Protocol
  - Login, Password
  - Server name, port
  - Path to resource, optional parameters
  - Anchor to the html fragment

HTTP Codes

HTTP Basics

In this video you will learn:

- What are HTTP codes?
- How many HTTP codes there are?

HTTP Codes

CODE	EXAMPLE	MEANING	DESCRIPTION
1xx	101	Information Code	Switching Protocols – requester has asked the server to switch protocols
2xx	200	Confirmation Code	OK – server accepted request and sent the response
3xx	304	Redirect Code	Not modified – what is in cache is still valid
4xx	404	Client application error Code	Not found – requested resource does not exist
5xx	500	HTTP server error code	Internal server error - there was a server error during processing that request

After watching this video you know:

- That an HTTP code is a number and a short description.
- That an HTTP code is returned by the server to inform what happened with a request.
- That there are 62 official HTTP codes.

REST

HTTP Basics

In this video you will learn:

- What are:
  - REST
  - JSON
  - HATEOAS
- What is the function of REST, JSON, and HATEOAS.

REST

URL / Method	api.test.com/cars	api.test.com/cars/car/1
GET	Download all cars from a server	Download one car form a server
PUT	Change all cars on a server	Change one car on a server
POST	Add one car to existing list on a server	You shouldn’t use that
DELETE	Remove all cars from a server	Remove one car from a server

JSON

Car

```
{
  "name": "Ford",
  "doors": 4,
  …
}
```

Car collection:

[{car1}, {car2}]

HATEOAS

GET /cars/car/1 HTTP

```
{
   "id": 1,
   "name": "Ford",
   "links": [
   {"href": ”/cars/car/1/status", "rel": "status", "type": "GET"},
   ...
   ]
}
```

After watching this video you know:

- That REST is a way of creating an urn part of the URI. The same as HTTP, REST is also stateless.
- That JSON is a text format used in HTTP communication.
- That HATEOAS is an additional link in JSON response body from a server.
- That HATEOAS is the most mature level of API levels.

Redirections

HTTP Basics

In this video you will learn:

- How to make a redirection on a user side?
- How to make a redirection on a server side?

Redirections

Link in html

Please click on that link.

Header named Location

HTTP/1.1 301 Moved Permanently
Location: https://mypage.com/me.html
Content-Type: text/html

Please click on that link.

Redirections - apache rewrite

Redirect permanent /oldpage.html http://www.example.com/newpage.html

Redirect 301 /oldpage.html http://www.example.com/newpage.html

After watching this video you know:

- How to make a redirection on a user’s browser side using Location header.
- How to make a redirection on a server side using a correct configuration.

curl - installation

HTTP Basics

In this video you will learn:

- How to install curl in Windows system step by step.
- Watch this move and repeat all steps of the installation by yourself.

Important!

- If you have installed Git for Windows you can omit this move.

After watching this video you know:

- How to correctly install a curl program.

curl - demo

HTTP Basics

In this video you will learn:

- How to sent requests using curl.

After watching this video you know:

- How to send a request by curl using HTTP methods: GET, POST.
- How to sent HTTP headers via curl.
- How HTTP headers and HTTP codes look like.

wget – installation, demo

HTTP Basics

In this video you will learn:

- How to install and use a wget program?

Important!

- This program isn’t installed with Git for Windows. That is why you should install it with this instruction.

After watching this video you know:

- How install a wget program.
- How to download a response and save it to the file using wget.

ping - demo

HTTP Basics

In this video you will learn:

- How to check if computers are connected to the same network?

After watching this video you know:

- How to check whether two computers are in the same network.
- Known flags:
  - -t - unlimited number of repetitions
  - -n - specified number of repetitions
  - -l - package size
- That you can use a few flags in one command in a terminal and in a a cmd.

ssh - installation

HTTP Basics

In this video you will learn:

- How to install the ssh tool.
- How to create a basic configuration.

Important!

- If you have installed Git for Windows already, you can skip this video.

After watching this video you know:

- How to install and configure the ssh tool.

ssh - demo

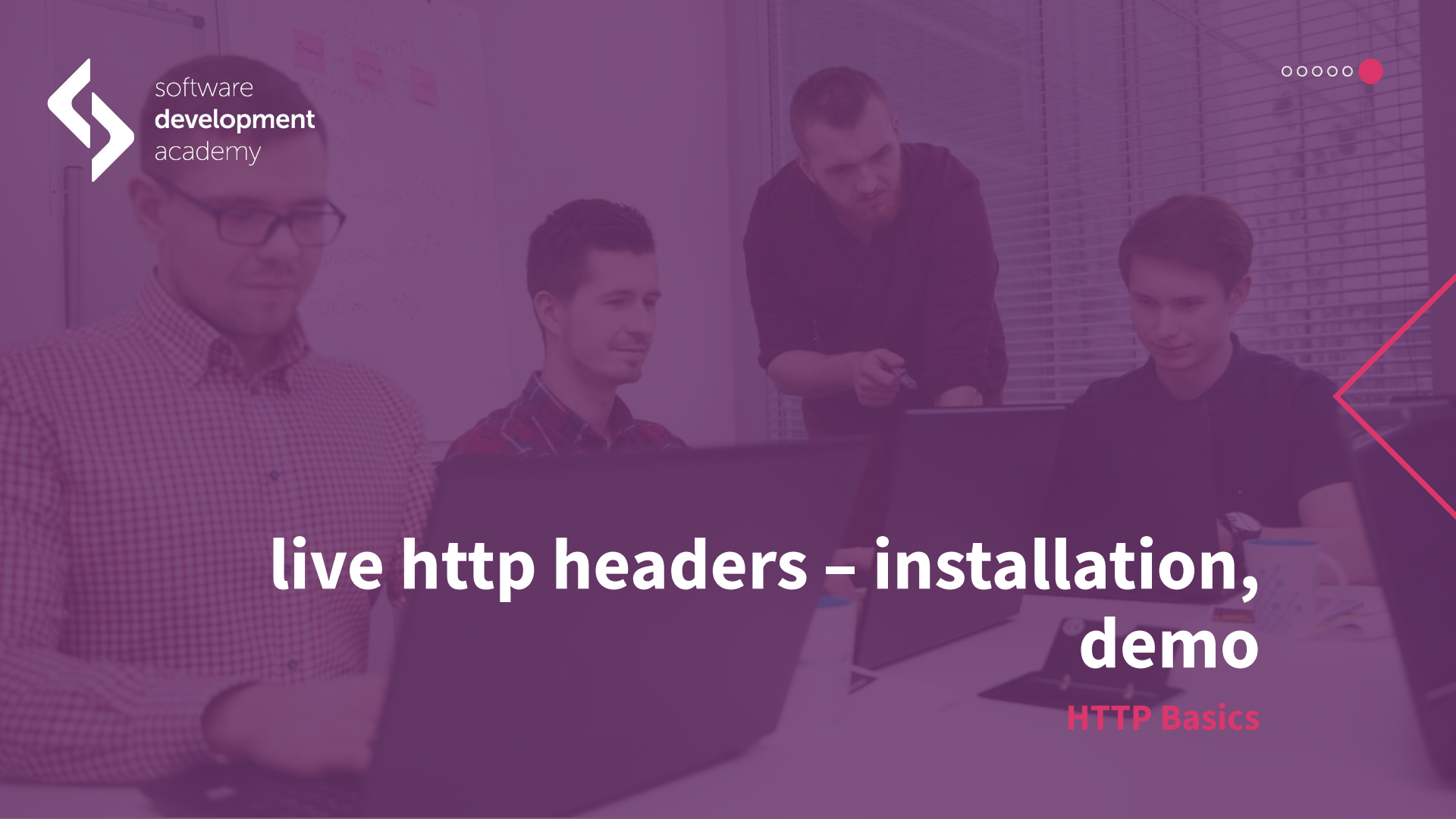
HTTP Basics

In this video you will learn:

- What the ssh can do.
- How to log in to a server using the ssh.
- How you can run a command on a server.
- How to log in to a server without a password.
- How to configure the ssh command.

After watching this video you know:

- That the ssh can be used to log in to a server.
- That in order to log in to a server without a password, you need a public key.
- How to configure the ssh to use:
  - port
  - username
  - any server name that you like



live http headers – installation, demo

HTTP Basics

In this video you will learn:

- How to install the Live Http Headers addon to Google Chrome and Mozilla Firefox browsers.
- How to collect all requests and responses that are in your browser using the Live Http Headers addon .

After watching this video you know:

- Have installed the live http headers addon in Google Chrome or Mozilla Firefox browser.
- Know how to collect and browse all requests and responses in your browser using the live http headers addon.

postman - installation

HTTP Basics

In this video you will learn:

- How to install Postman.

After watching this video you know:

- Have Postman installed.
- Have an account created in Postman.
- Have the basic settings in Postman configured.

postman – demo

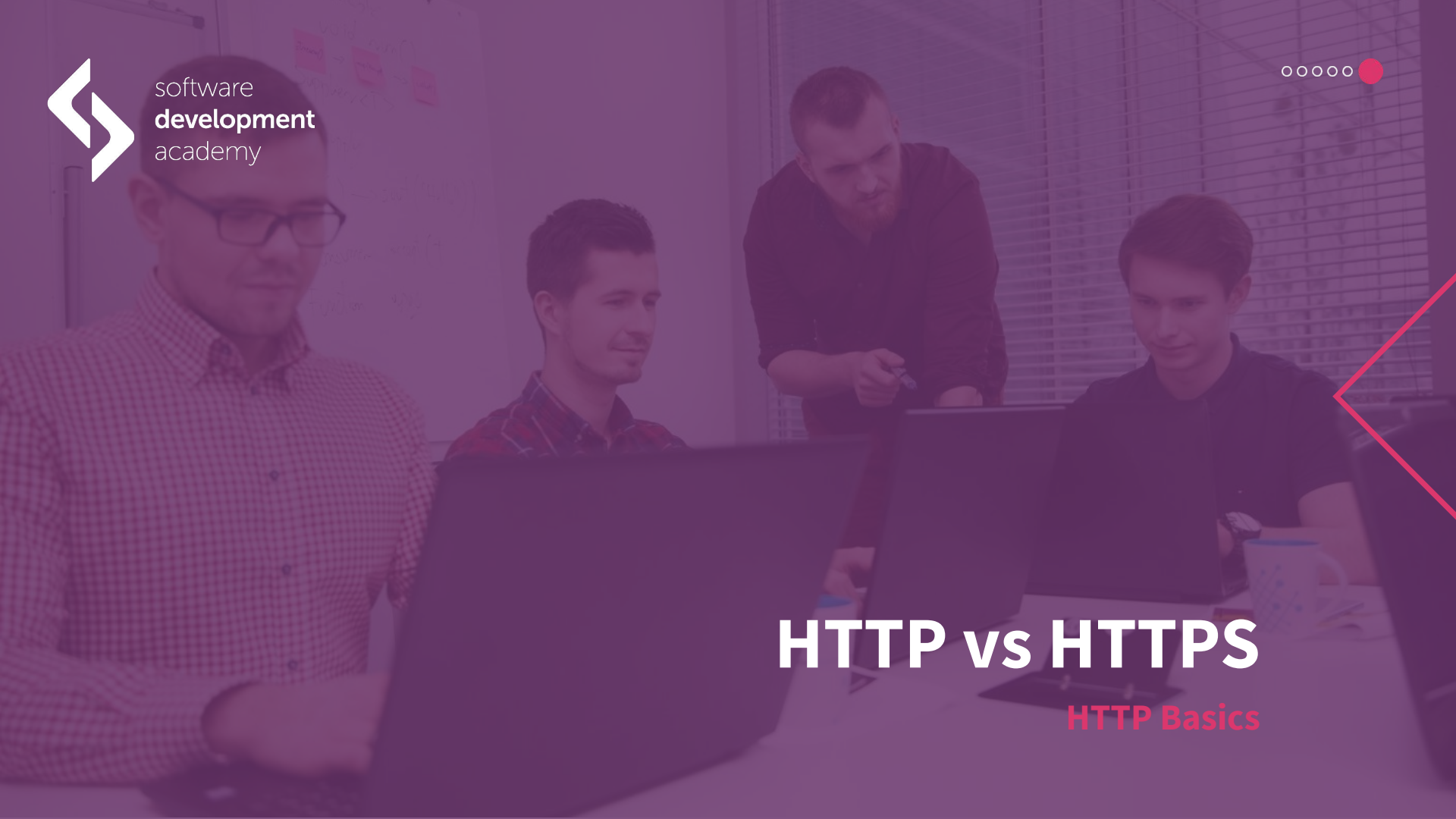
HTTP Basics

In this video you will learn:

- What Postman can be used for.
- What are the basic concepts in Postman.
- How to send a request and store it in Postman.

After watching this video you know:

- What Postman can be used for.
- What are:
  - History
  - Collection
  - Environments
  - Workspaces
- How response headers and HTTP codes look like.
- How to send POST and GET requests with or without headers.

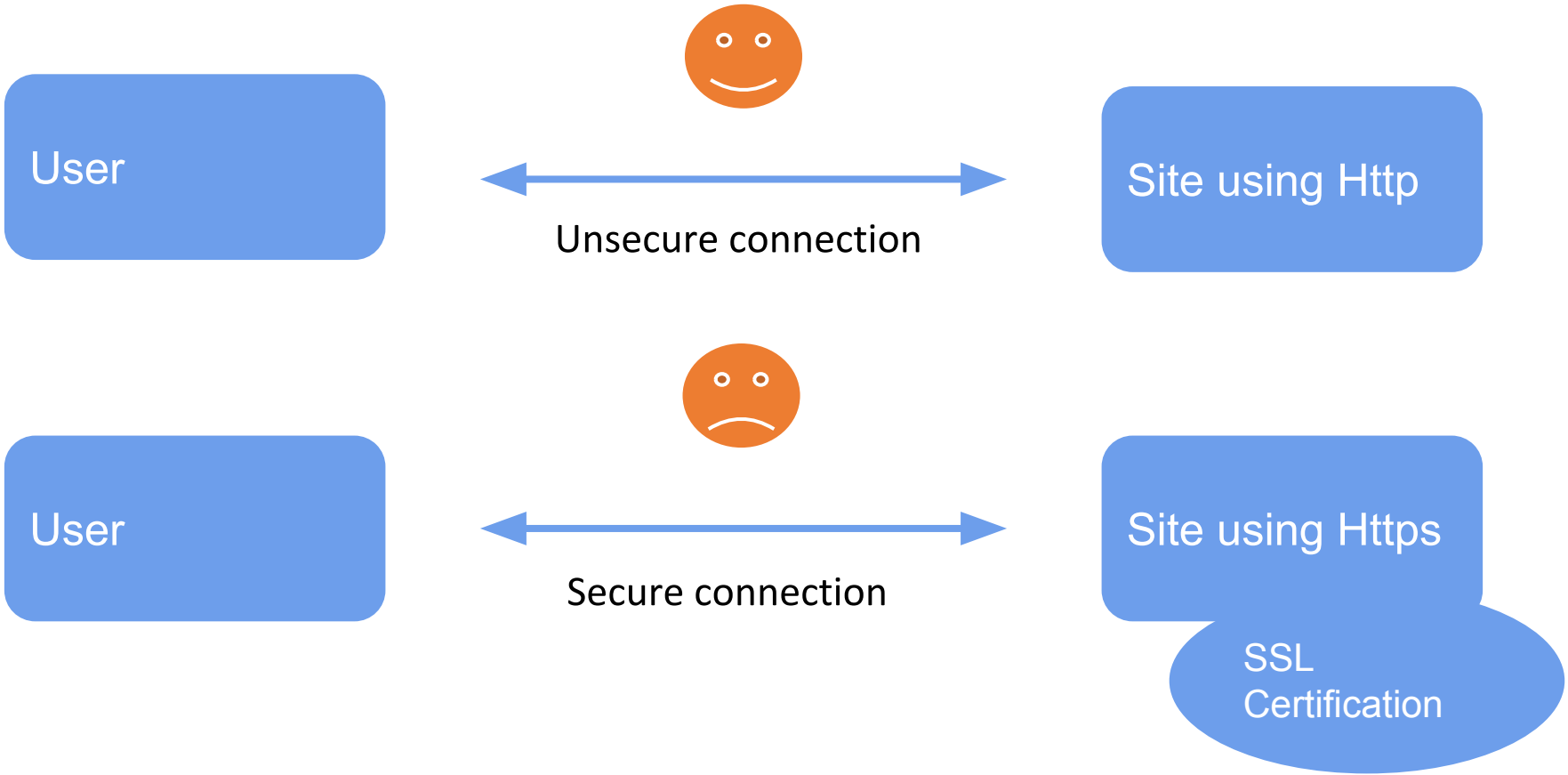


HTTP vs HTTPS

HTTP Basics

In this video you will learn:

- What is HTTPS, what benefit you can get from it, and when you should use it.
- What the server should have in order to be available over the HTTPS.


User
Unsecure connection
Site using Http
User
Secure connection
Site using Https
SSL
Certification

After watching this video you know:

- That HTTPS makes it difficult to eavesdrop on requests and responses to and from a server.
- That a server must have a SSL certificate.


software
development
academy
Congratulations!
You've finished the course!