

PROTÓCOLOS FUNDAMENTAIS DA INTERNET

AULA 12

HTTP - Hyper Text Transfer Protocol *

HTTP is an ASCII protocol (like SMTP). Assim é fácil para o utilizador falar directamente com o servidor web.

Basta fazer uma ligação (TCP) com o programa telnet para a porta 80

HTTP transacções

1. cliente conecta servidor na porta 80
cliente emite comando (method) HTTP com endereço do documento e HTTP versão como argumentos

exemplo SET /index.html HTTP/1.1

2. Cliente envia informações adicionais (opcional) sobre as propriedades do browser e documentos que aceita

exemplo

User-Agent: Mozilla/4.05 (WinNT; I)

Accept: image/gif, image/jpeg, */*

3. cliente envia linha em branco para terminar o cabeçalho

4. cliente envia dados (opcional)

exemplo POST /cgi-bin/birthday.pl HTTP/1.0

⋮
(cabeçalho)

linha em branco

month = august & date = 29

* J. Spohnour, P. Eusekin: WEBMASTER IN A NUTSHELL O'REILLY 1999

O servidor responde:

1. linha de status com 3 campos: versão HTTP, status código, das condições
exemplo HTTP/1.1 200 OK

2. cabeçalho

Date: Fri, 20 Sep 1996 08:17:58 GMT

Server: MESA/1.5.2

Last-modified: Mon, 17 Jun 1996 21:53:08 GMT

Content-type: text/html

Content-length: 2482

3. linha em branco

4. dados

COMANDOS (MÉTODOS) HTTP

GET pedido de dados ao servidor. É o comando mais utilizado

exemplo

GET /index.html HTTP/1.0

Connection: keep-alive

← pedido ao servidor para não fechar a ligação

User-Agent:

Host:

Accept: image/gif, image/jpeg, */*

linha em branco

O método GET também serve para enviar dados ao servidor (quando a HTML <form> tag especifica method="GET")

exemplo

GET /cgi-bin/birthday.pl?month=august & date=29 HTTP/1.0

HEAD - semelhante a GET mas não são enviados dados
estamos apenas interessados:

- data da última atualização
- tamanho do documento
- tipo de dados
- tipo de servidor

POST - envio de dados (para um programa que os vai processar)

exemplo

```
POST /cgi-bin/birthday.pl HTTP/1.0
```

{ case:thw

birth en 6 rows

month = august & date = 24

OUTROS COMANDOS

LINK

UNLINK

PUT } pede para o documento no campo de dados ser colocado
DELETE } no ficheiro indicado como argumento

RESPOSTAS DO SERVIDOR

Códigos

- | | |
|-----------|-------------------------------------|
| 100 - 199 | informação |
| 200 - 299 | pedido com sucesso |
| 300 - 399 | redireção, outras ações necessárias |
| 400 - 499 | pedido do cliente incompleto |
| 500 - 599 | erro |

exemplos

200 OK

301 Moved Permanently
Location: indica o novo URL

305 Use Proxy
Location: indica o URL do proxy

400 Bad request (syntax)

404 Not Found

408 Request Time-out

500 Internal Server Error (CGI program crashed)

505 HTTP Version not supported

CABECALHOS HTTP

Há 4 categorias de cabeçalhos

- geral (info não relacionada com cliente, server ou HTTP)
- Pedidos (formatos preferidos e parâmetros do server)
- Respostas (info do servidor)
- Entidade (info sobre os dados)

Cabeçalho pode estender-se por mais de 1 linha; cada linha extra deve começar por 1 espaço ou 1 tab

exemplos:

Cache-Control: no-cache

Date: Mon, 06 May 1999 05:57:00 GMT

HTTP-Version: 1.0

Accept: text/html

Accept-Language: en

Host: hostname[:port]

If-Modified-Since: date

Cascollo HTTP (cont)

Cookie: name = value

Set-Cookie: name = value [; options]

value; expires = date

value; secure

User-Agent: Mozilla 2.1

Content-Encoding: x-gzip

Content-Language: en

Content-Length: 40238

Content-Transfer-Encoding: base64

Last-modified: date

Expires: date