

Лекция № 7. Протокол HTTP

УЧЕБНЫЕ ВОПРОСЫ

1. Назначение протокола
2. Структура протокола

Вопрос №1. Назначение протокола

HTTP (HyperText Transfer Protocol — протокол передачи гипертекста) — символно-ориентированный клиент-серверный протокол прикладного уровня без сохранения состояния, используемый сервисом World Wide Web.

Основным объектом манипуляции в HTTP является **ресурс**, на который указывает URI (Uniform Resource Identifier – уникальный идентификатор ресурса) в запросе клиента.

Основными **ресурсами** являются хранящиеся на сервере **файлы**, но ими могут быть и другие логические или абстрактные объекты.

Протокол HTTP позволяет указать **способ представления** (кодирования) одного и того же **ресурса** по различным параметрам. Благодаря этой возможности клиент и веб-сервер могут **обмениваться двоичными данными**, хотя данный протокол является текстовым.

Каждое HTTP-сообщение состоит из трёх частей, которые передаются в следующем порядке:

1. Стартовая строка (англ. Starting line) — определяет тип сообщения;
2. Заголовки (англ. Headers) — характеризуют тело сообщения, параметры передачи и прочие сведения;
3. Тело сообщения (англ. Message Body) — непосредственно данные сообщения. Обязательно должно отделяться от заголовков пустой строкой.

No.	Source	Destination	Protocol	Info
31	192.168.3.1	194.188.210.1	HTTP	HTTP/1.0 302 Moved Temporarily
37	192.168.3.1	194.188.210.1	HTTP	HTTP/1.0 200 OK (text/html)
54	192.168.3.1	194.188.210.1	HTTP	HTTP/1.0 200 OK (text/css)


```

HTTP/1.0 200 OK\r\n
Server: Apache/2.2.3 (CentOS)\r\n
Last-Modified: Wed, 09 Feb 2011 17:13:15 GMT\r\n
Content-Type: text/html; charset=UTF-8\r\n
Accept-Ranges: bytes\r\n
Date: Thu, 03 Mar 2011 04:04:36 GMT\r\n
Content-Length: 2945\r\n
Age: 13165\r\n
X-Cache: HIT from proxy.omgtu\r\n
Via: 1.0 proxy.omgtu (squid/3.1.8)\r\n
Connection: keep-alive\r\n
\r\n
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/T
<html xmlns="http://www.w3.org/1999/xhtml">\n
<head>\n
  <title>IANA &mdash; Example domains</title>\n
  <!-- start common-head -->\n
  <meta http-equiv="Content-type" content="text/html; charset=utf-8" />\n
  <link rel="stylesheet" type="text/css" href="/_css/reset-fonts-grids.css" />\n
  <link rel="stylesheet" type="text/css" media="screen" href="/_css/screen.css" />\n
  <link rel="stylesheet" type="text/css" media="print" href="/_css/print.css" />\n
  <link rel="shortcut icon" type="image/ico" href="/favicon.ico" />\n

```


0000	48 54 54 50 2f 31 2e 30 20 32 30 30 20 4f 4b 0d	HTTP/1.0 200 OK.
0010	0a 53 65 72 76 65 72 3a 20 41 70 61 63 68 65 2f	.Server: Apache/

Стартовая строка
Заголовки
Тело сообщения

Вопрос №2. Структура протокола

Стартовая строка HTTP

Стартовая строка является **обязательным элементом**, так как указывает на тип запроса/ответа, заголовки и тело сообщения могут отсутствовать.

Стартовые строки различаются для **запроса и ответа**.

Стартовая строка HTTP

Строка запроса выглядит так:

Метод *URI* *HTTP/Версия протокола*

Пример запроса:

GET /web-programming/index.html HTTP/1.1

Стартовая строка HTTP

Стартовая строка ответа сервера имеет следующий формат:

HTTP/Версия КодСостояния [Пояснение]

Например, на предыдущий наш запрос клиентом данной страницы сервер ответил строкой:

HTTP/1.1 200 Ok

Стартовая строка HTTP

Метод HTTP (англ. HTTP Method) — последовательность из любых символов, кроме управляющих и разделителей, указывающая на основную **операцию над ресурсом**. Обычно метод представляет собой короткое английское слово, записанное заглавными буквами.

Стартовая строка HTTP

Основные Методы протокола

OPTIONS - Используется для определения возможностей веб-сервера или параметров соединения для конкретного ресурса.

GET - Используется для запроса содержимого указанного ресурса.

HEAD - Аналогичен методу GET, за исключением того, что в ответе сервера отсутствует тело.

POST - Применяется для передачи пользовательских данных заданному ресурсу.

PUT - Применяется для загрузки содержимого запроса на указанный в запросе URI.

Стартовая строка HTTP

Коды состояния

Код состояния **информирует** клиента о результатах выполнения запроса и **определяет** его дальнейшее поведение. Набор кодов состояния является стандартом.

Каждый код представляется **целым трехзначным числом**. Первая цифра указывает на **класс состояния**, последующие - **порядковый номер состояния**. За кодом ответа обычно следует краткое описание на английском языке.



Стартовая строка HTTP

Коды состояния

1xx Informational (Информационный) - В этот класс выделены коды, информирующие о процессе передачи.

- 100 Continue (Продолжать)
- 101 Switching Protocols (Переключение протоколов)
- 102 Processing (Идёт обработка)

Стартовая строка HTTP

Коды состояния

2xx Success (Успешно) - Сообщения данного класса информируют о случаях успешного принятия и обработки запроса клиента.

- 200 OK (Успешно).
- 201 Created (Создано)
- 202 Accepted (Принято)
- 204 No Content (Нет содержимого)
- 206 Partial Content (Частичное содержимое)

Стартовая строка HTTP

Коды состояния

3xx Redirection (Перенаправление) - Коды статуса класса 3xx сообщают клиенту, что для успешного выполнения операции нужно произвести следующий запрос к другому URI.

- 300 Multiple Choices (Множественный выбор)
- 301 Moved Permanently (Перемещено навсегда)
- 304 Not Modified (Не изменялось)

Стартовая строка HTTP

Коды состояния

4xx Client Error (Ошибка клиента) - Класс кодов 4xx предназначен для указания ошибок со стороны клиента. .

- 401 Unauthorized (Неавторизован)
- 402 Payment Required (Требуется оплата)
- 403 Forbidden (Запрещено)
- 404 Not Found (Не найдено)
- 405 Method Not Allowed (Метод не поддерживается)
- 406 Not Acceptable (Не приемлемо)
- 407 Proxy Authentication Required (Требуется аутентификация прокси)

Стартовая строка HTTP

Коды состояния

5xx Server Error (Ошибка сервера) - Коды 5xx выделены под случаи неудачного выполнения операции по вине сервера.

- 500 Internal Server Error (Внутренняя ошибка сервера)
- 502 Bad Gateway (Плохой шлюз)
- 503 Service Unavailable (Сервис недоступен)
- 504 Gateway Timeout (Шлюз не отвечает)

Заголовки HTTP

Заголовок HTTP (HTTP Header) — это строка в HTTP-сообщении, содержащая разделённую двоеточием пару вида «параметр-значение».

```
Server: Apache/2.2.3 (CentOS)
Last-Modified: Wed, 09 Feb 2011 17:13:15 GMT
Content-Type: text/html; charset=UTF-8
Accept-Ranges: bytes
Date: Thu, 03 Mar 2011 04:04:36 GMT
Content-Length: 2945
Age: 51
X-Cache: HIT from proxy.omgtu
Via: 1.0 proxy.omgtu (squid/3.1.8)
Connection: keep-alive
```

Тело сообщения

Тело HTTP сообщения (message-body), если оно присутствует, используется для передачи сущности, связанной с запросом или ответом.

Сущности (entity, в переводах также встречается название “объект”) — это полезная информация, передаваемая в запросе или ответе. Сущность состоит из метайнформации и непосредственно содержания.