

ВВЕДЕНИЕ В WEB

**KEYWORDS: HYPERTEXT, HTTP, COOKIES,
STATUS CODES, FTP, EMAIL**

СОДЕРЖАНИЕ

1. Ключевые понятия web

1. Гипертекст, HTML
2. URI/URL

2. Протокол HTTP

1. Методы
2. Заголовки
3. Параметры и cookies

3. Другие протоколы сети

1. FTP, SMTP, IMAP, POP3

КЛЮЧЕВЫЕ ПОНЯТИЯ

Internet – это сеть передачи информации между устройствами

WWW (Web) – система документов (гипертекста), размещённых на web-серверах, подключенных к Интернету

Гипертекст (hypertext) – текст, содержащий ссылки для перехода к другим текстам

HTML (hypertext markup language) – стандарт представления гипертекста в Интернете

Web 2.0 – совокупность всего, что находится поверх статичных HTML-документов. (Социальные сети, блоги, видео-хостинги, *интернет-маркетинг*, web-приложения и сервисы). Сервис-ориентированная сеть

Web 3.0 (*semantic web*) – сеть машинно-интерпретируемых, семантических документов. Контент-ориентированная сеть

ЭВОЛЮЦИЯ ГИПЕРТЕКСТА

Web 1.0	Web 2.0	Web 2.0, 3.0
HTML – подмножество языка разметки SGML	xHTML – совмещение тегов HTML со стандартом XML	[x]HTML5 – серьёзные изменения в вёрстке, добавление семантики
SGML parsers	XML parsers	XML parsers
Предназначен для отображения в браузере	Предназначен для отображения интерактивного и мультимедиа контента	Предназначен для создания web- приложений, поддерживает сематическую разметку

URI И URL

Unifier resource identifier (URI) – составленный по правилам (машино-читаемый) текстовый идентификатор некоторого ресурса

URI common syntax: **scheme:scheme-specific-part**

Unifier resource locator (URL) – подвид URI, описывающий *расположение и способ доступа* объекта в сети Интернет

`http://www.mail.ru/`

Unifier resource name (URN) – подвид URI, идентифицирующий объект, но не дающий информацию о местонахождении

`urn:isbn:0451450523`

СИНТАКСИС URL

протокол

http:// пара логин/пароль
user:pass@

порт

www.example.com:80

имя или IP-адрес сервера

/Path/to/Res

путь к ресурсу а сервере

параметры

?k1=v1&k2=v2#hash

идентификатор внутри документа

Примеры:

- `https://mail.google.com/`
- `ftp://root:qwerty@ftp.example.com/`
- `wss://server.name:443/method/name`
- `http://sprotasov.ru/index.html#author:Aleksandr%20Buyanov`

ПРОТОКОЛ HTTP

HTTP (hypertext transfer protocol) – протокол прикладного уровня доставки текстовых данных, разработанный для передачи гипертекста. Обеспечивает общение *клиента* (обычно браузера) и *сервера* (web-сервера) через запросы клиента (request) и ответы сервера (response)

HTTP v1.0. Не поддерживает использование одной TCP-сессии для нескольких запросов. Поддерживает методы (типы запросов) от клиента:

- **GET** – получение контента с сервера
- **HEAD** – получение заголовка без тела ответа
- **POST** – отправка данных на сервер

HTTP v1.1. Поддерживает также методы **PUT**, **DELETE**, **TRACE**, **OPTIONS**, **CONNECT**, **PATCH**

HTTP HEADER

HTTP request =

Request URI + **HOST** + **[[headers]]** + `<empty_line>` + **body**

```
POST /index.html HTTP/1.1
HOST: example.org:8080
Cache-Control: max-age=0
Accept: text/html,application/xhtml+xml
...
```

```
param1=value1&param2=value2
```

HTTP response=

Response code + **headers** + `<empty_line>` + **body**

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=UTF-8
Content-Length: 3123
...
```

```
<html>...
```


ВАЖНЫЕ ЗАГОЛОВКИ

REQUEST

Accept, Accept-Charset, Accept-Encoding – понятные браузеру форматы (text/plain, application/xml), кодировки (utf-8) и форматы сжатия (gzip, deflate)

Authorization – заголовок, несущий информацию о типе авторизации и содержащий пароли, ключи и т.п.

Authorization: Basic QWxhZGRpbjpvYVUHNlc2FtZQ==

Content-Length – длина тела запроса (и ответа тоже)

User-Agent – браузер и ОС

Upgrade – запрос на смену протокола (Upgrade: websocket)

RESPONSE

Cache-Control – продолжительность жизни документа в кэше

Content-Encoding, Content-Language, Content-Type – характеристики содержимого

КОДЫ ОТВЕТА

- **1xx – information**
 - 101 - Switching Protocols
- **2xx – success**
 - 200 – OK
 - 201 – created (new resource)
 - 206 – partial content
- **3xx – redirection (заголовок Location: addr)**
 - 301 – moved permanently
 - 304 – not modified
- **4xx – client-side error**
 - 401 – unauthorized
 - 404 – not found
- **5xx – server-side error**
 - 503 – server unavailable

ВЛИЯНИЕ ЗАПРОСОВ НА СОСТОЯНИЕ СИСТЕМЫ: БЕЗОПАСНОСТЬ И ИДЕМПОТЕНТНОСТЬ ЗАПРОСОВ

Безопасный запрос не изменяет состояние сервера и предназначен для получения информации

Идемпотентный запрос при повторном и последующих выполнениях приводит к одному приводит к тому же состоянию сервера, что и однократный запрос ($F(x) == F(F(x))$)

	SAFE	NOT SAFE
IDEMPOTENT	HEAD, GET, OPTIONS, TRACE	PUT, DELETE
NON-IDEMPOTENT	--	POST, PATCH

PARAMETERS

GET params:

- `http://server.name/path?param1=value1¶m2=value2`

POST params:

- `POST /path HTTP/1.1`

`param1=value1¶m2=value2`

COOKIES

Cookies – небольшой объём данных, присланный сервером браузеру и хранимый на диске. Предельно – 4К

Нужны для поддержки stateful-сервисов (корзина интернет-магазина, и т.п.) или для хранения ключа сессии

Cookies имеют время жизни и **отправляются на сервер при каждом запросе**

COOKIES EXAMPLES

```
GET /index.html HTTP/1.1  
Host: www.example.org
```

browser -----> **server**

```
HTTP/1.0 200 OK  
Content-type: text/html  
Set-Cookie: name=value  
Set-Cookie: name2=value2; Expires=Wed, 09 Jun 2021 10:18:14 GMT  
  
(content of page)
```

browser

←-----

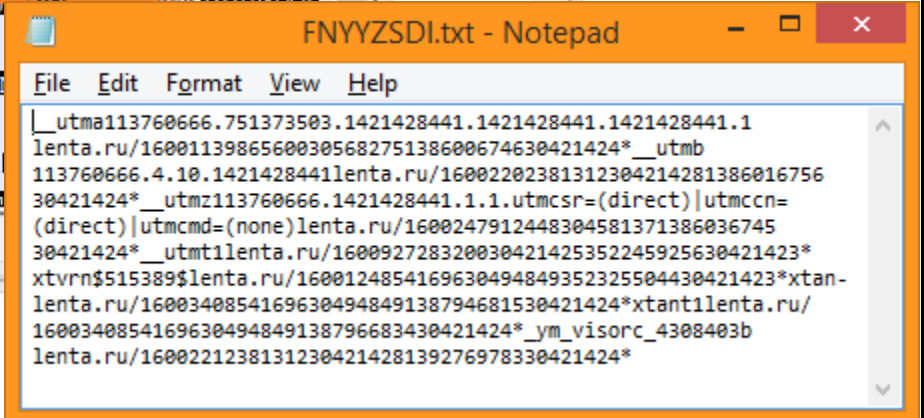
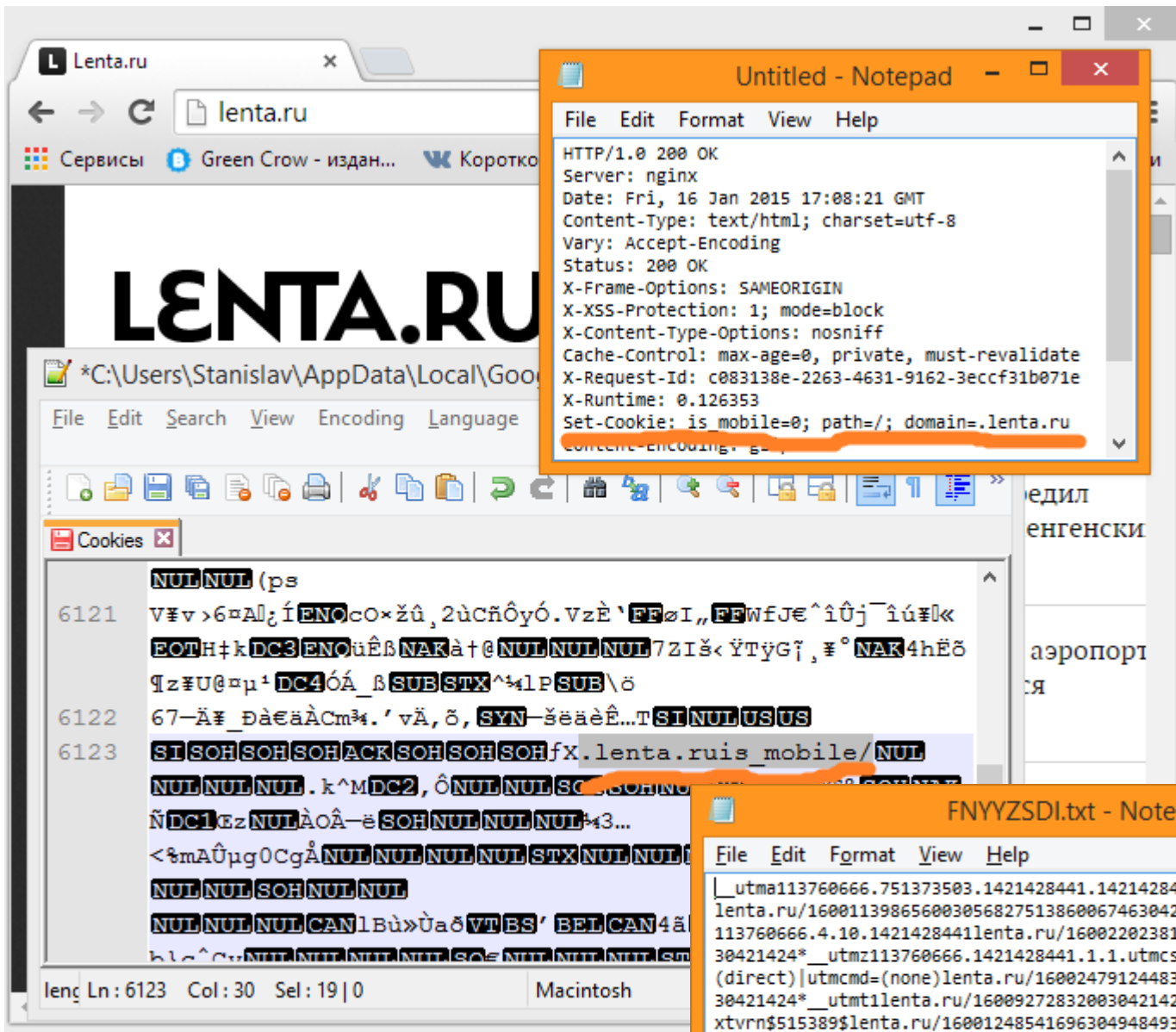
server

```
GET /spec.html HTTP/1.1  
Host: www.example.org  
Cookie: name=value; name2=value2  
Accept: */*
```

browser

----->

server



FTP

FTP (file transfer protocol) – протокол передачи файлов, 1971. Состоит из greeting и сессии запросов (VERB param [params]\015\012) и ответов сервера.

Client:		Connects @server:21
Server:	220 Hello, ...	
Client:	USER MB1234	
Server:	331 Password required to access user account MB1234.	
Client:	PASS QXJ4Z2AF	PLAIN TEXT
Server:	230 Logged in.	
Client:	CWD Bills	Change directory to "Bills."
Server:	250 "/home/MB1234/Bills" is new working directory.	
Client:	PORT 192,168,1,2,7,138	accepts data @client:1930 [=7*256 + 138]
Server:	200 PORT command successful.	
Client:	LIST	Send the list of files in "Bills."
Server:	150 Opening ASCII mode data connection for /bin/ls.	server connects out from its port 20 to port client:1930
Server:	226 Listing completed.	succeeded
Client:	PORT 192,168,1,2,7,139	
Server:	200 PORT command successful.	
Client:	RETR Yoyodyne.TXT	Download "Yoyodyne.TXT."
Server:	150 Opening ASCII mode data connection for Yoyodyne.TXT	
Server:	226 Transfer completed.	succeeded
Client:	QUIT	
Server:	221 Goodbye.	

EMAIL: SMTP, IMAP, POP3

SMTP (simple mail transfer protocol, @:25) – для передачи сообщений между серверами и для отправки с клиентов. Очень опохож на FTP

POP3 (post office protocol v3, @:110) – стандартный протокол получения почты клиентом с сервера

IMAP (internet message access protocol, @:143) – стандартный протокол получения почты клиентом с сервера, имеет возможность отправки (но считается неудачной), не разрывает сессию, поддерживают множественные клиенты для одного ящика

ОБЗОР

