

More Web

Alexander Evgin

20 марта 2020 г.

Outline

1 Recap

2 Sessions

Cookies

Authentication

3 API

Web Service

SOAP

Web-RPC

4 Three-tier Architecture

Проекты

- Без проекта сдать курс **можно**, но не выше, чем на “3”

- Команды до двух человек

- Баллы за проект общие:

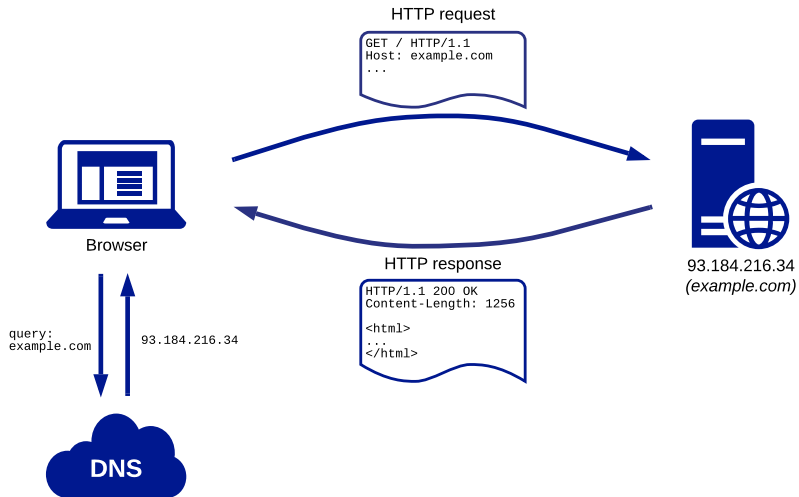
10 баллов Архитектура

10 баллов Качество кода

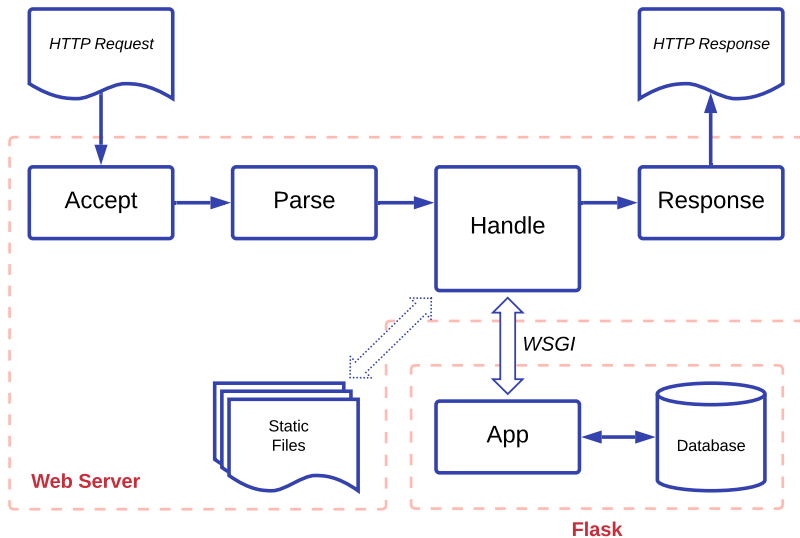
10 баллов Ориентация в проекте

10 баллов Готовность

Recap



Recap



Cookies

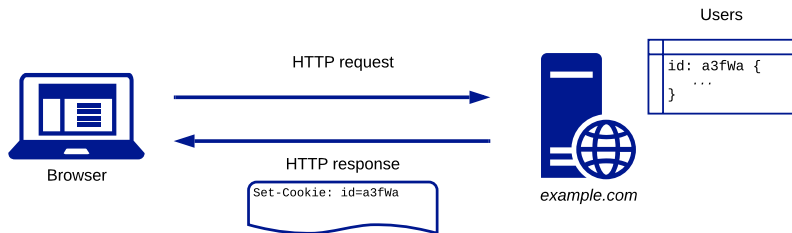
```
HTTP/1.1 200 OK  
Set-Cookie: id=a3fWa; Expires=Wed, 21 Oct 2015 07:28:00 GMT;  
Secure; HttpOnly; Domain=example.com;
```

```
GET / HTTP/1.1  
Cookie: id=a3fWa;
```

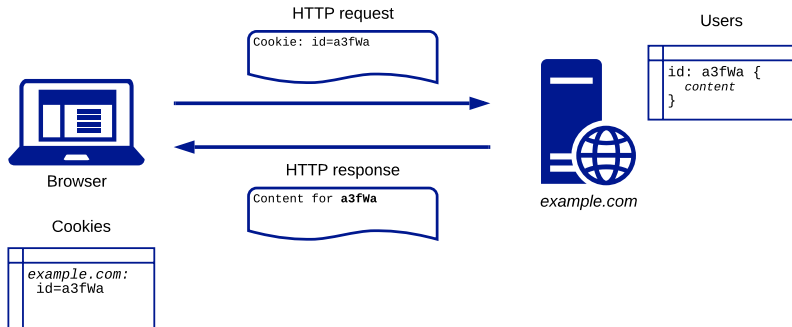
- 50 cookies per domain
- Total 4096 bytes

HTTP is *stateless*, but...

Sessions



Sessions



Sessions

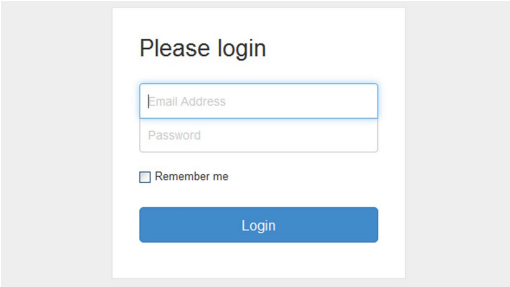
Server session

- Server sends *unique id*
- Server stores personal content mapped to id

Client session

- Server sends *encrypted personal data*
- Server stores secret key

Authentication



Please login

Email Address

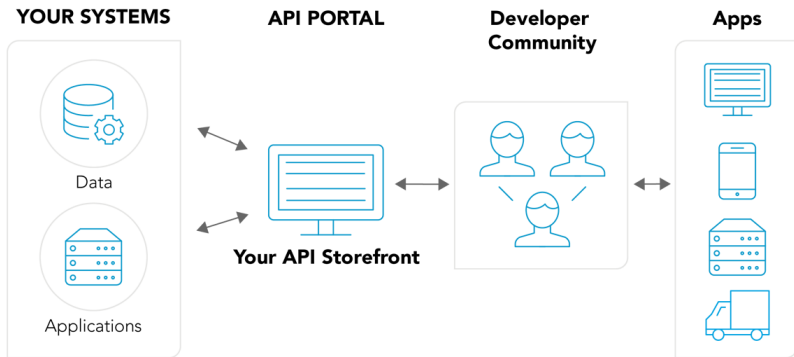
Password

☐ Remember me

Login

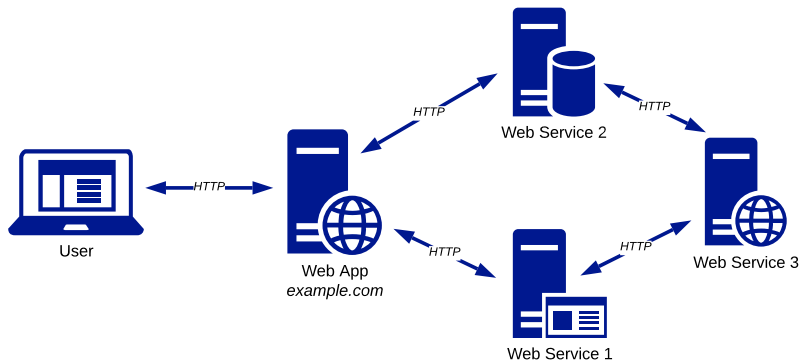
1. Authentication (*confirm login and password*)
2. Generate **unique id**, send via cookie
3. Every further request accompanies with this cookie
 - Cookie hijacking
 - No passwords in cookies!

API (Application Programming Interface)



Web Service (Web API)

SOA (Service-oriented Architecture)



Web App Provides service directly to user

Web Service Provides service to other apps or services

JSON API

```
GET /articles HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{  
  "data": [{  
    "type": "articles",  
    "id": "1",  
    "attributes": {  
      "title": "One great article",  
      ...  
    }  
  }  
}
```

XML API

```
GET /?prefix=/europe/france/ HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/xml
```

```
<?xml version='1.0' encoding='utf-8'?>
```

```
<ListBucketResult xmlns=
```

```
    'http://doc.storage.googleapis.com/2010-03-01'>
```

```
  <Name>travel-maps</Name>
```

```
  <Prefix>/europe/france/</Prefix>
```

```
  <Contents>
```

```
    <Key>europe/france/cannes.jpg</Key>
```

```
    <LastModified>2010-02-17T22:11:12.487Z</LastModified>
```

```
  ...
```

SOAP (Simple Object Access Protocol)

- Encapsulated in HTTP
- Content type: XML

```
POST /InStock HTTP/1.1
Content-Type: application/soap+xml; charset=utf-8
SOAPAction: "http://www.w3.org/2003/05/soap-envelope"

<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:m="http://www.example.org">
  <soap:Header>
  </soap:Header>
  <soap:Body>
    <m:GetStockPrice>
      <m:StockName>T</m:StockName>
    </m:GetStockPrice>
  </soap:Body>
</soap:Envelope>
```

Web-RPC (Remote Procedure Call)

- Encapsulated in HTTP
- Content type: XML

```
<?xml version="1.0"?>
<methodCall>
  <methodName>examples.getStateName</methodName>
  <params>
    <param>
      <value><i4>40</i4></value>
    </param>
  </params>
</methodCall>
```

```
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value><string>South Dakota</string></value>
    </param>
  </params>
</methodResponse>
```


Three-tier Architecture

Tier 1

Presentation

- *HTML*
- *JavaScript*
- *CSS*

Tier 2

Application

- *Flask*
- *Django*

Tier 3

Data

- *SQLite*
- *Mongo*
- *MySQL*

Now let's go back to Python

Сегодня нам нужно:

Flask & env

<https://flask.palletsprojects.com/en/1.1.x/installation/>

git

<https://git-scm.com/download/win>

Heroku

Account: <https://www.heroku.com/>

Heroku-CLI: <https://devcenter.heroku.com/articles/heroku-cli>