**Лабораторная работа №9**

**Инструкция, как получить данные пользователя Facebook через авторизацию OAuth.**

OAuth 2.0 — протокол авторизации, позволяющий выдать одному сервису (приложению) права на доступ к ресурсам пользователя на другом сервисе. Протокол избавляет от необходимости доверять приложению логин и пароль, а также позволяет выдавать ограниченный набор прав, а не все сразу.

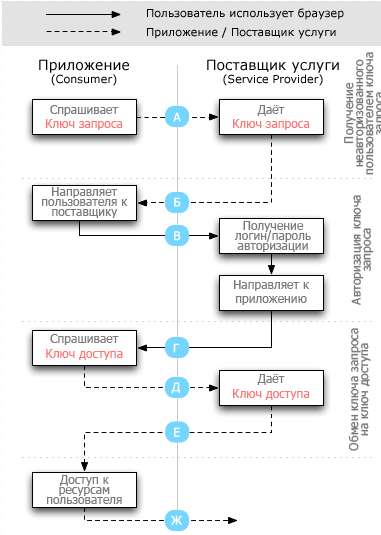
Общая схема работы приложения, использующего OAuth, такова:

* получение авторизации
* обращение к защищенным ресурсам

Результатом авторизации является access token — некий ключ (обычно просто набор символов), предъявление которого является пропуском к защищенным ресурсам. Обращение к ним в самом простом случае происходит по HTTPS с указанием в заголовках или в качестве одного из параметров полученного access token'а.

В протоколе описано несколько вариантов авторизации, подходящих для различных ситуаций:

* авторизация для приложений, имеющих серверную часть (чаще всего, это сайты и веб-приложения)
* авторизация для полностью клиентских приложений (мобильные и desktop-приложения)
* авторизация по логину и паролю
* восстановление предыдущей авторизации

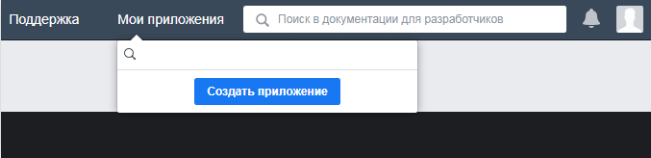


Авторизация для приложений, имеющих серверную часть

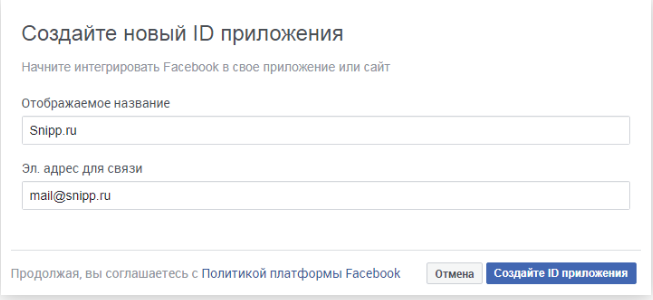
**Авторизация на сайте через Facebook**

**1. Регистрация приложения**

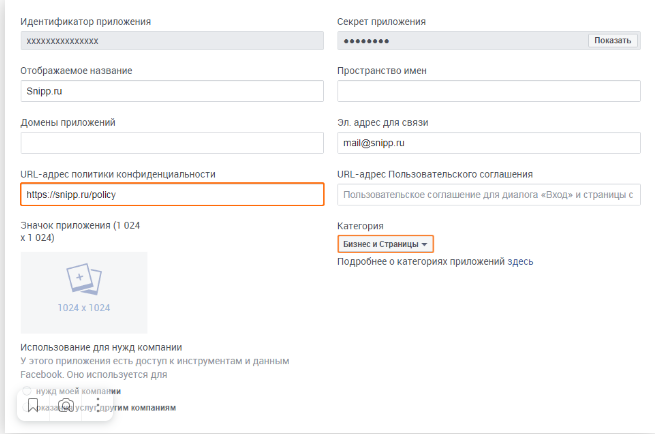
На странице *https://developers.facebook.com/apps/* нужно создать приложение.



Указываем название приложения и контактный e-mail:



Чтобы включить приложение, нужно перейти в основные настройки и указать URL-адрес политики конфиденциальности и категорию.



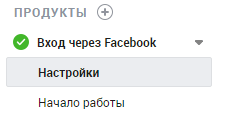
Включаем приложение:

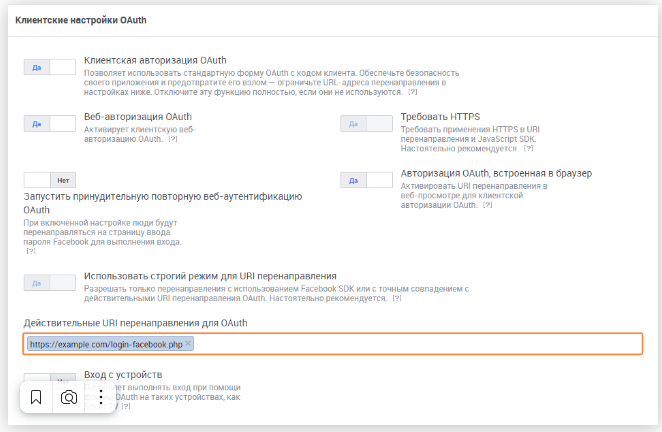


**2. Ссылка для входа**

Создадим PHP-скрипт, который будет обрабатывать ответы от Facebook, к примеру *https://example.com/login-facebook.php*

Этот URL нужно добавить в список «Действительных URI перенаправления для OAuth» в настройках входа через Facebook.





После этого можно сформировать ссылку для входа:

https://www.facebook.com/dialog/oauth?client\_id=Идентификатор\_приложения&redirect\_uri=https://example.com/login-facebook.php&scope=email&response\_type=code

В URL можно добавить параметр «state» для передачи своих данных.

$params = array(

'client\_id' => 'Идентификатор приложения',

'redirect\_uri' => 'https://example.com/login-facebook.php',

'scope' => 'email',

'response\_type' => 'code',

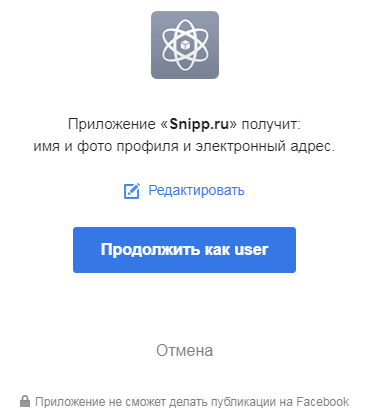
'state' => '123'

);

$url = 'https://www.facebook.com/dialog/oauth?' . urldecode(http\_build\_query($params));

echo '<a href="' . $url . '">Войти через Facebook</a>';

Кликнув по ссылке, пользователь переходит на страницу с диалогом:



После того, как пользователь подтвердил запрос, он будет перенаправлен на redirect\_uri с добавлением кода в GET-параметре.

**3. Скрипт получения данных пользователя**

В скрипте, полученный код отправляется запросом получения access\_token и по нему *метод https://graph.facebook.com/me* возвращает id, имя, фамилию, e-mail и фото пользователя (полный список полей на *https://developers.facebook.com/docs/graph-api/reference/user*).

<?php

if (!empty($\_GET['code'])) {

// var\_dump($\_GET['state']);

$params = array(

'client\_id' => 'Идентификатор приложения',

'client\_secret' => 'Секрет приложения',

'redirect\_uri' => 'https://example.com/login-facebook.php',

'code' => $\_GET['code']

);

// Получение access\_token

$data = file\_get\_contents('https://graph.facebook.com/oauth/access\_token?' . urldecode(http\_build\_query($params)));

$data = json\_decode($data, true);

if (!empty($data['access\_token'])) {

$params = array(

'access\_token' => $data['access\_token'],

'fields' => 'id,email,first\_name,last\_name,picture'

);

// Получение данных пользователя

$info = file\_get\_contents('https://graph.facebook.com/me?' . urldecode(http\_build\_query($params)));

$info = json\_decode($info, true);

var\_dump($info);

}

}

Результат:

array(

'id' => '123456789',

'email' => 'mail@example.com',

'first\_name' => 'Иван',

'last\_name' => 'Иванов',

'picture' => array(

'data' => (

'height' => 50,

'is\_silhouette' => true,

'url' => 'https://platform-lookaside.fbsbx.com/platform/profilepic/?asid=123456',

'width' => 50

}

}

}

**4. «#\_=\_» в URL**

После успешной авторизации Facebook добавляет #\_=\_ к адресу возврата.



Сделано это специально, поскольку некоторые браузеры, при редиректе переносят хеш из URL-адреса в новый URL. Такой хеш можно удалить:

if (window.location.hash === "#\_=\_"){

history.replaceState

? history.replaceState(null, null, window.location.href.split("#")[0])

: window.location.hash = "";

}