

Практика 15: OAuth

Юрий Литвинов

y.litvinov@spbu.ru

Задачей этой практики станет поработать с клиентской стороны с OAuth на примере Google Drive API. Google Drive поддержан во всех нормальных языках программирования, естественно, с удобным готовым механизмом авторизации. Поэтому, чтобы было интереснее, есть дополнительное условие — не использовать клиентские библиотеки, реализуя работу с API на «голом» HTTPS. Требуется написать консольный клиент, умеющий показывать список файлов и папок в корне Google Drive аккаунта пользователя.

При этом:

1. Придётся зарегистрировать приложение на <https://console.developers.google.com> и включить нужный API. После этого надо создать OAuth Credentials, Google Developers Console отдаст файл credentials.json с клиентскими секретами, их надо сохранить к себе, но в систему контроля версий не выкладывать.
2. Справка по OAuth, как он реализован в Google Drive: <https://developers.google.com/drive/api/v3/about-auth>. Тут поддерживаются все три «каноничных» режима OAuth: для веб-приложений, для серверных приложений и для настольных/мобильных клиентов, нам интересен последний случай. Вот документация конкретно про него: <https://developers.google.com/identity/protocols/oauth2/native-app>.

Некоторые рекомендации:

1. Запрос access token должен быть обязательно POST-запросом, GET вернёт 404.
2. Копипастить access token из адресной строки браузера для этой задачи вполне ок. Вообще, по-хорошему надо поднять локальный сервер и сказать странице авторизации сделать на него (на localhost) редирект с access token в URL. Сервер парсит URL и вычитывает из него access token, который потом используется в приложении. Обратите внимание, что хоть у нас и консольный клиент, в любом случае требуется запустить браузер и отправить пользователя на страницу авторизации.
3. Некоторые значения возвращаются в BASE64-кодировке, а ожидаются в plain text.
4. Для этой задачи не надо продлять Access Token, правильно хранить Refresh Token и всё такое. Если будет время, можно попробовать, потому что «настоящее» приложение должно это делать, но если у вас каждый раз при запуске приложение будет просить авторизоваться, это вполне допустимо.
5. Значения Client Id и Client Secret (или путь до credentials.json) можно принимать параметрами командной строки.