Зачем нужен OAuth2?

Подготовила Анна Петросян, преподаватель – Антон Игоревич Говоров

План

- **Что такое OAyth2 и какие роли**
- **Схема**
- **Сценарии авторизации**
- **Токен доступа для код авторизация**
- **Достоинства и недостатки**

Что такое OAuth2?

OAuth2 - Open Authorization - фреймворк для авторизации, позволяющий приложениям осуществлять ограниченный доступ к пользовательским аккаунтам на HTTP сервисах

Роли

- Клиент (приложение)
- Пользователь (владелец ресурса)
- API
- Сервер ресурсов

Схема

КЛИК

Приложение

запрос на авторизацию

разрешение на авторизацию

Пользователь

запрос авторизационного токена

предоставление информации, авторизация на её использование, проверка подлинности

создание токена доступа

авторизация завершена

Авторизационный сервер (API)

предоставление токена доступа

токен действительный

предоставление ресурса приложению

Сервер ресурсов

Сценарии авторизации

Основные точки авторизации

oauth/authorize - используется для инициирования процедуры авторизации

oauth/authorize/certificate - используется для инициирования авторизации с аутентификацией по сертификату

oauth/token - используется для получения маркера доступа

по ссылкам дока

1. **С использованием кода авторизации**
- Вместо прямого запроса на авторизацию у Владельца Ресурса, Клиентское приложение перенаправляет его на доверенный Центр авторизации.
2. **С аутентификацией по сертификату**
- Стороны проверяют сертификаты друг друга, при этом Центр Идентификации в процессе обработки запроса пытается извлечь клиентский сертификат и, используя его, идентифицировать и аутентифицировать пользователя.
3. **С использованием типа разрешения Implicit**
- Нет механизма маркеров доступа, аутентификации клиентского приложения, проверки сертификатов.
4. **С использованием учётных данных владельца ресурса**
- Учетные данные Владельца ресурса подходит для использования в тех случаях, когда Владелец данных доверяет Клиентскому приложению.
5. **Получение операторского делегирующего маркера доступа к ресурсу**
- Расширение сценария "Авторизация с аутентификацией", подключает DSS, то есть появляется проверка электронной подписи

Токен доступа для код авторизация

- **https://cloud.digitalocean.com/v1/oauth/authorize?response_type=code&client_id=CLIENT_ID&redirect_uri=CALLBACK_URL&scope=read**

- **https://cloud.digitalocean.com/v1/oauth/authorize**: входная точка API авторизации
- client_id=****CLIENT_ID: идентификатор клиента приложения
- redirect_uri=****CALLBACK_URL: URL, на который сервис перенаправит пользовательского агент после выдачи авторизационного кода
- response_type=code: указывает на то, что приложение запрашивает доступ с помощью кода авторизации
- scope=read: задаёт уровень доступа приложения

- **{"access_token":"ACCESS_TOKEN","token_type":"bearer","expires_in":2592000,"refresh_token":"REFRESH_TOKEN","scope":"read","uid":100101,"info":{"name":"Mark E. Mark","email":"mark@thefunkybunch.com"}}**

Достоинства

- Возможность аутентификации пользователей без необходимости создавать новый аккаунт
- Интеграция с API
- Мобильные и десктоп-приложения, сайты

Недостатки

- Отдельная реализация на каждый сервер
- Нужен jwt-токен для устранения дополнительных запросов
- Если токен украли, то есть, хоть и кратковременный, но доступ к данным

Выводы

OAuth2 - фреймворк, предоставляющий приложениям ограниченный доступ к пользовательским аккаунтам других сервисов для аутентификации

Спасибо за внимание!