УДК 004.85

Макаров Д.А.
студент
факультет «Информатика и системы управления»
Московский государственный технический университет имени Н.Э. Баумана
Россия, г. Москва
Шибанова А.Д.
студент
факультет «Робототехника и комплексная автоматизация»
Московский государственный технический университет имени Н.Э. Баумана
Россия, г. Москва

МЕХАНИЗМ АВТОРИЗАЦИИ OAUTH

Аннотация: в данной статье рассмотрен механизм авторизации OAuth. Проведен анализ технологии, описана необходимость ее применения. Показан принцип работы OAuth. Описаны плюсы и минусы применения данного механизма.

Ключевые слова: пароль, OAuth, токен, приложение, авторизация.

Makarov D.A.
student
Faculty of Informatics and Management Systems
Moscow State Technical University named after N.E. Bauman
Russia, Moscow
Shibanova A.D.
student

Faculty of Robotics and complex automation
Moscow State Technical University named after N.E.
Bauman
Russia, Moscow

OAUTH AUTHORIZATION MECHANISM

Abstract: This article discusses the OAuth authorization mechanism. The analysis of the technology is carried out, the necessity of its application is described. Shows how OAuth works. The advantages and disadvantages of using this mechanism are described.

Keywords: password, OAuth, token, application, authorization.

OAuth (Open Authorization) - стандарт авторизации ресурсов. Он позволяет пользователям в организации входить в систему с помощью провайдеров подключения OAuth / OpenID, таких как Microsoft Azure AD, AWS Cognito, приложения Google, Facebook и т. д., и делиться своей информацией с корпоративными приложениями. Он использует механизм авторизации на основе токенов для предоставления пользователям доступа к корпоративным приложениям.

Приложения, которые поддерживают вход с использованием сторонних сервисов, обычно предлагают пользователю пройти аутентификацию, предоставляя такие опции, как «Войти через Facebook» или «Войти через Google» и т. д., таким образом, позволяя пользователю использовать свои учетные данные для входа в сторонний сервис. В ответ служба отправляет маркер доступа запрашивающему приложению, который доказывает подлинность пользователя, запрашивающего доступ. Затем токен используется для выполнения запросов к ресурсам, необходимым конечному пользователю. [1]

Почему OAuth?

До OAuth стандартом базовой аутентификации был HTTP, при котором пользователю предлагалось ввести имя пользователя и пароль для доступа к каждому приложению. Веб-сайты будут предлагать вам ввести свое имя пользователя и пароль непосредственно в форму, и они будут входить в ваши данные (например, вашу учетную запись Gmail) как вы.

Базовая аутентификация по-прежнему используется в качестве примитивной формы аутентификации API для серверных приложений, в которой вместо отправки имени пользователя и пароля на сервер с каждым запросом пользователь отправляет идентификатор ключа API и секрет. [2]

В отличие от вышеизложенного, OAuth позволяет выполнять аутентификацию с использованием токенов доступа, что является более безопасным, поскольку не используется обмен паролями.

Как работает OAuth?

Рабочий процесс системы единого входа (SSO) OAuth:

1. Неизвестный пользователь пытается получить доступ к ресурсам.
2. Веб-приложение отправляет запрос авторизации провайдеру OAuth.
3. Сервер OAuth предлагает пользователю войти в систему и авторизует приложение.
4. Пользователь перенаправляется на страницу входа, где он входит в систему.
5. Провайдер OAuth аутентифицирует пользователя и отправляет код авторизации в веб-приложение miniOrange.
6. Веб-приложение отправляет свой собственный client_id, client_secret с кодом авторизации, полученным от сервера OAuth.
7. Затем сервер аутентифицирует запрос и отправляет токен доступа клиенту.

8. Ваше веб-приложение использует токен доступа для доступа к ресурсам на сервере ресурсов.
9. Использование токена доступа, токена идентификатора и информации о пользователе позволяет пользователю получить доступ к защищенным функциям.
10. Теперь пользователь прошел аутентификацию и вошел в систему. Таким образом, приложение предоставляет доступ к ресурсам.

Достоинства

1. Предотвращает необходимость поддерживать службу аутентификации:

Поскольку OAuth позволяет приложениям аутентифицировать пользователей, устанавливая их личность из сторонних сервисов, это избавляет приложение от необходимости реализовывать собственную систему аутентификации.

2. Простой обмен пользовательскими данными:

Поскольку для единого входа требуется единый набор учетных данных для аутентификации и предоставляется доступ к нескольким приложениям / веб-сайтам, пользователю необходимо запомнить только один пароль, в отличие от запоминания нескольких паролей для каждого отдельного приложения / веб-сайта.

3.Более надежный:

Обычно приложениям аутентификации (Google, Facebook) и т. Д. Пользователи доверяют больше, чем вашему собственному приложению. Следовательно, они легко аутентифицируются с помощью этих сервисов, не беспокоясь о том, чтобы делиться своей информацией напрямую.

4. Простота реализации:

OAuth использует исключительно протокол HTTP для отправки и получения информации и токенов доступа. Это упрощает реализацию службы OAuth, а также клиента, поскольку не требует использования сложных стандартов или определений.

Ограничения

Уязвимость системы безопасности: если вы используете одну службу для подключения ко всем другим вашим любимым сайтам, и одна учетная запись будет взломана, последствия будут ощущаться на нескольких сайтах, а не только на одном. Например, если вы вошли в систему через Facebook для всех приложений, а затем, если ваша учетная запись Facebook была взломана, ваша личность и информация будут скомпрометированы на нескольких сайтах, а не только на одном. [3]

Неправильное использование данных: несмотря на то, что OAuth ограничивает новые веб-сайты и приложения от получения всех ваших пользовательских данных и продажи их против вашей воли, вам все равно нужно доверять службе аутентификации с вашими данными. Это означает, что вашими данными владеет служба, и вы больше не можете контролировать, могут ли они быть использованы без вашего разрешения.

Список литературы

1. OAuth 2.0 простым и понятным языком. URL: https://habr.com/ru/company/mailru/blog/115163/ (дата обращения: 04.01.2021).
2. HTTP Authentication: Basic and Digest Access Authentication. URL: https://tools.ietf.org/html/rfc2617 (дата обращения: 04.01.2021).
3. The OAuth 2.0 Authorization Framework. URL: https://tools.ietf.org/html/rfc6749 (дата обращения: 04.01.2021).