УДК 004.428.4

РАЗРАБОТКА СЕРВЕРА ДОСТОВЕРНОСТИ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ

Никоноров Н.В. (ФГБОУ ВО «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича»)

Научный руководитель - Кривоносова Н.В. (ФГБОУ ВО «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича»)

В современное время веб-сервисы требуют все больше внимания к безопасности. Так как все больше инцидентов с информационной безопасностью выявляется у крупных и не очень крупных компаний.

В число аспектов информационной безопасности входит безопасность сессий и токенов.

Есть много решений реализующие аутентификацию и авторизацию с помощью токенов. Но лишь немногие оптимальны для реального проекта.

Существуют уже готовые решения, но не все они отвечают бизнес-требованиям. Чаще всего готовые сервисы не подходят для малого бизнеса. Это связано с тем, что готовые сервисы требуют оплаты ежемесячной или по ресурсам.

Из сторонних провайдеров можно выделить Auth0. Он берет на себя обязанности по авторизации и аутентификации. Также есть и другие дополнительные сервисы у данного провайдера, но их рассматривать не будем. Особенность данного провайдера в том, что он предоставляет надежность, стабильность и интеграцию с большинством сервисов, что не все это делают или не могут себе позволить. Из отрицательных качеств могу выделить стоимость и отсутствие клиентоориентированности на российский рынок.

Также рассмотрим провайдера Firebase. Этот провайдер предназначен для разработчиков мобильных приложений, и не только, как замена собственного сервера приложении. Рассмотрим его сервис Firebase Authentication. Из положительных качеств можно выделить только использование квот вместо фиксированной оплаты и так же бесплатные квоты. Из отрицательных качеств можно выделить малое количество сторонних сервисов, которые подключены к сервису.

Предлагается рассмотреть самописное решение, которое покроет некоторые бизнес-требования. Из технологий было использовано в основном JWT как формат токенов и GraphQL как язык запросов вместо REST. Данная связка позволяет избежать больших затрат на добавление изменений в будущем.

JWT предлагает формат токенов, которые облегчают жизнь в разработке. JWT формат говорит нам о том, что не обязательно использовать монолитную архитектуру и можем распределить сервера по ролям не боясь потерять контроль доступа к данным.

В JWT есть определенная структура, которая позволяет гибко управлять проверкой подлинности данных токена или шифрованием данных. Структура JWT разделяется на заголовок, тело и подпись. В заголовке храниться информация о типе токена и его характеристиках. В теле хранится информация, как и служебная, так и пользовательская. В теле не рекомендуется хранить чувствительную информацию, если не используется шифрование токена. В подписи хранится хэш: заголовка, тела и ключа. Благодаря этому можно гарантировать достоверность данных.

Также стоит добавить, что для сервисов обычно используется схема с access и refresh токенами. В данной схеме токен доступа (access) имеет время актуальности (достоверности) (жизни) намного меньше, чем токен перевыпуска (refresh). Токен перевыпуска токенов обычно имеет свободный формат. Токен перевыпуска используется для получения новой связки токенов доступа и перевыпуска. Токен перевыпуска является одноразовым.

Актуальность данной технологии можно выявить, посмотрев на положительные качества и отрицательные. С явным преимуществом положительных больше, чем отрицательных.

Данное решение можно внедрить практически в любую систему. Также внедрение подразумевает изменение имеющийся системы. Стоимость зависит от сложности системы.

Например, можно внедрить в личный сайт, где пользователи могут регистрироваться и авторизовываться. И в качестве токена доступа использовать JWT.