

JWT: ЭФФЕКТИВНЫЙ МЕХАНИЗМ РЕГИСТРАЦИИ И АУТЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ

Поляков Антон Александрович

студент, Сибирский государственный индустриальный университет, РФ, г. Новокузнецк

Аннотация. Статья посвящена механизму использования JSON Web Token (JWT) для обеспечения безопасности процессов регистрации, аутентификации и авторизации пользователей в веб-приложениях.

Ключевые слова: JWT, регистрация, авторизация, аутентификация, безопасность данных, клиент-серверные приложения.

Зачастую клиенту для доступа к полному функционалу веб-сервиса необходимо обозначить себя, указав требуемые данные как при регистрации, так и при авторизации. В таком случае разработчик должен понимать, что данные клиента необходимо защищать от злоумышленников при работе с сервером. Одним из механизмов безопасной передачи информации между клиентом и сервером является JSON Web Token.

JSON Web Token (JWT) представляет собой открытый стандарт для создания токенов доступа, который позволяет безопасно передавать информацию между клиентом и сервером. Этот механизм широко используется для аутентификации и авторизации пользователей в веб-приложениях.

JWT состоит из трёх основных частей, разделённых точками:

- 1) Header (заголовок). Содержит информацию о типе токена и алгоритме подписи, используемом для его создания. Обычно это HMAC SHA256 или RSA.
- 2) Payload (полезная нагрузка). Содержит данные, которые мы хотим передать. Это могут быть утверждения о пользователе, такие как его идентификатор, роли и срок действия токена.
- 3) Signature (подпись). Генерируется с использованием заголовка и полезной нагрузки, а также секретного ключа. Она обеспечивает целостность токена и подтверждает его подлинность.

Все 3 части, зашифрованные и собранные вместе, непосредственно и являются JWT, представленным в следующем формате: «hhhhh.ppppp.sssss», где h,p,s – header, payload, signature соответственно [1].

Чтобы понять, каким образом JWT позволяет безопасно обмениваться информацией, разберём процессы регистрации или авторизации, они принципиально одинаковы, и аутентификации.

Механизм авторизации и регистрации при помощи JWT аналогичен. Пользователь должен отправить необходимую информацию, которую сервер обрабатывает и генерирует токен, если всё корректно. Затем клиенту в виде ответа отправляется токен и сохраняется в его браузере (Рисунок 1).

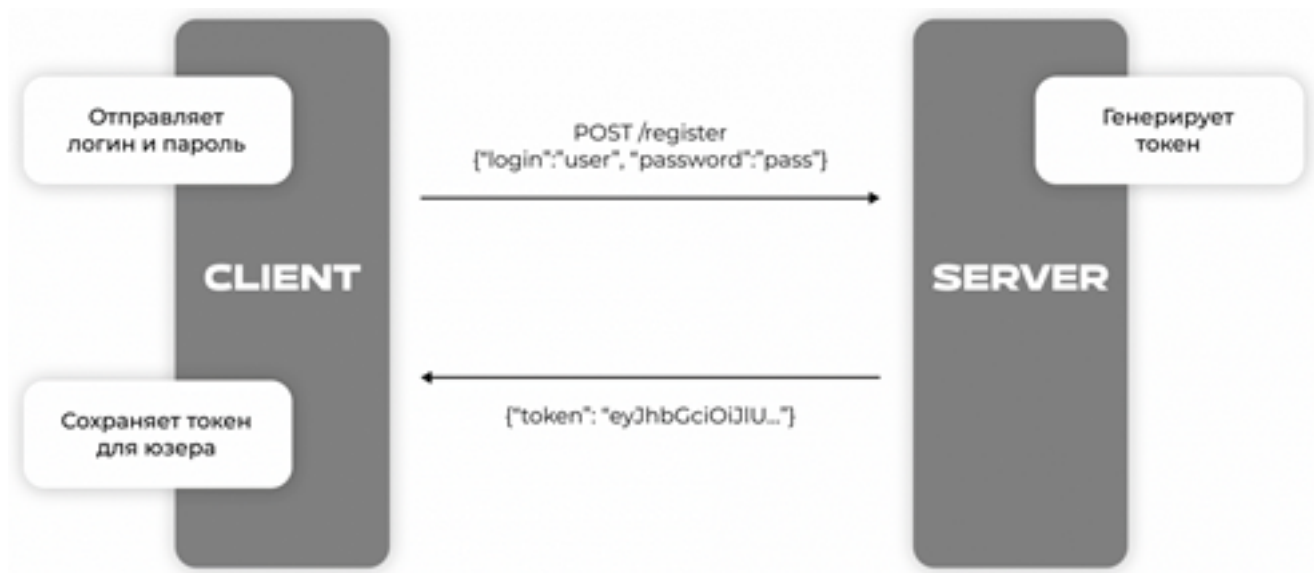


Рисунок 1. Регистрация и авторизация с JWT

Механизм аутентификации пользователя для доступа к защищённым ресурсам описан иначе. Здесь подразумевается, что пользователь уже зарегистрировался, авторизовался и хочет получить доступ к какому-либо ресурсу. Для этого он отправляет свой сохранённый токен в виде заголовка `Authorization` вместе с самим запросом на сервер, где токен расшифровывается, проверяется подпись. На основе расшифрованных данных из токена сервер определяет права доступа пользователя к запрашиваемому ресурсу. Если доступ разрешен, сервер обрабатывает запрос и возвращает ответ клиенту (Рисунок 2) [2].

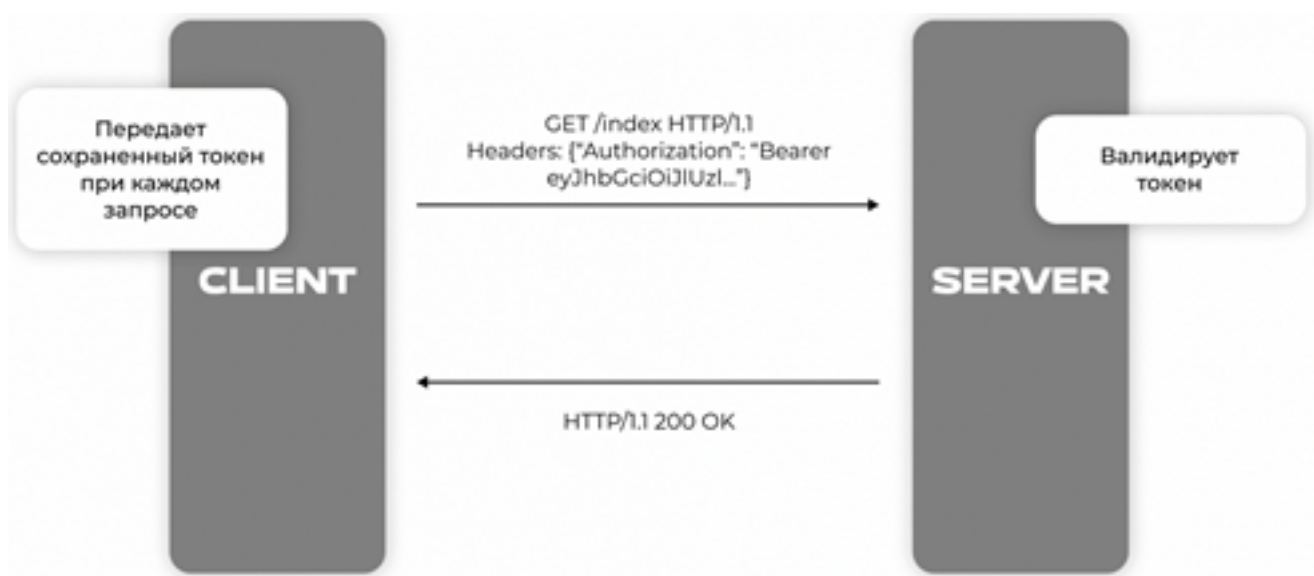


Рисунок 2. Аутентификация с JWT

Существуют и более продвинутые механизмы, где используется несколько токенов, однако схемы, представленные выше, иллюстрируют основные этапы процесса регистрации/авторизации и аутентификации пользователя с использованием JSON Web Token.

Такой подход позволяет эффективно и безопасно управлять процессами пользователей в веб-приложениях. Использование JWT обеспечивает надежную и безопасную передачу информации между клиентом и сервером, минимизируя риски, связанные с подделкой данных и несанкционированным доступом.

Список литературы:

1. Официальная документация JWT (JSON Web Token) – URL: <https://jwt.io/> (дата обращения: 07.01.2025).
2. Полуэктова, Н. Р. Разработка веб-приложений: учебное пособие для вузов / Н. Р. Полуэктова. — 2-е изд. — Москва (Высшее образование). — ISBN 978-5-534-18645-1. — Текст: электронный — URL: <https://urait.ru/bcode/545238> (дата обращения: 07.01.2025).