

ЭФФЕКТИВНЫЙ ОБМЕН ДАННЫМИ В ИНТЕРНЕТЕ ВЕЩЕЙ: СРАВНЕНИЕ ПРОТОКОЛОВ И ИСПОЛЬЗОВАНИЕ ОТКРЫТЫХ MQTT-БРОКЕРОВ

Гафаров Булат Рустемович

Московский государственный технический
университет им. Н.Э. Баумана
gafarovbr@student.bmstu.ru

EFFICIENT DATA EXCHANGE IN THE INTERNET OF THINGS: COMPARISON OF PROTOCOLS AND THE USE OF OPEN MQTT BROKERS

B. Gafarov

Summary. When implementing systems related to the Internet of Things, designers and developers of information systems are faced with the task of choosing technologies for data exchange between subsystems. The article highlights the problem of wireless data transmission between physical devices equipped with sensors, actuators, and the main application server. A comparison of the main protocols used in the Internet of Things was carried out, the concept of one of them was described, an option for its implementation using an open source broker was proposed, and testing was carried out. As part of the testing, the results of the study made it possible to reduce costs at the stage of system operation and increase security. The knowledge gained as a result of the study can be used both in small projects related to home automation and in large projects, for example, in the field of industrial Internet of Things.

Keywords: automation, Internet of Things, data transfer protocol, information system, mqtt, mqtt broker, open source solutions, microcontroller, sensors.

Аннотация. При реализации систем, связанных с интернетом вещей, проектировщики и разработчики информационных систем сталкиваются с задачей выбора технологий для обмена данными между подсистемами. В статье освещается проблема беспроводной передачи данных между физическими устройствами, оснащенных датчиками, исполнительными устройствами, и основным сервером приложения. Было проведено сравнение основных протоколов, использующихся в интернете вещей, описана концепция одного из них, предложен вариант его реализации с помощью брокера с открытым исходным кодом, проведена апробация. В рамках апробации результаты исследования позволили сократить расходы на этапе эксплуатации системы, повысили безопасность. Знания, полученные в результате исследования, могут быть использованы как в небольших проектах, связанных с домашней автоматизацией, так и в крупных проектах, например, в области промышленного интернета вещей.

Ключевые слова: автоматизация, интернет вещей, протокол передачи данных, информационная система, mqtt, mqtt-брокер, решения с открытым исходным кодом, микроконтроллер, датчики.

Введение

В последнее время широкое распространение во всем мире получают цифровые, информационные и телекоммуникационные ресурсы, происходит активная цифровизация процессов деятельности различных сфер жизни общества [1]. Одним из примеров является отрасль сельского хозяйства. В настоящее время идёт активное внедрение информационных технологий в данной сфере. Растущее население Земли — а к 2050 году оно может увеличиться на два миллиарда человек — требует все больше продовольствия. Справиться с производством больших объемов продукции, пользуясь исключительно устаревшими разработками, довольно сложно. Последнее является одним из важнейших аргументов для введения IT-технологий во все сферы агропромышленного комплекса [2].

По мнению многих экспертов, те компании, которые активно используют инновационные технологии в сво-

ем бизнесе, могут добиться высокого роста доходности и стать лидерами на рынке агропродукции в современных экономических условиях. При этом существует мнение, что через 10 лет при прочих равных условиях более 80 % российских сельхозпредприятий будут использовать в своей работе цифровые решения [3]. Однако, реализация таких технологий зачастую невозможна без применения средств автоматизации и облачных решений.

При реализации подобных систем возникают проблемы управления и передачи данных от датчиков к пользователям системы. Цель данной статьи — предложить решение, которое обеспечит стабильную и полную передачу данных между пользователями и объектами автоматизации.

Анализ технологий беспроводной передачи данных, используемых в интернете вещей

Передача данных в Интернете Вещей является самой важной функцией этой концепции. Именно поэтому для

неё были разработаны различные протоколы передачи данных, среди которых можно выделить основные: HTTP, WebSocket, MQTT [4]. Рассмотрим эти протоколы в контексте передачи данных между клиентским приложением и микроконтроллером, поскольку в архитектуре интернета вещей микроконтроллеры используются для взаимодействия с физическими объектами [5].

WebSocket — это постоянное соединение между клиентом и сервером, которое обеспечивает двунаправленную связь между обеими сторонами с использованием TCP-соединения. Это означает, что протокол позволяет отправлять данные от клиента к серверу и от сервера к клиенту в любой момент времени [6].

Технология WebSocket позволяет реализовать передачу данных между микроконтроллером и веб-приложением, однако основными сценариями применения являются приложения с интерактивным и динамичным контентом, такими как онлайн-чат, игры и другие приложения с интенсивным обменом данными в реальном времени. Кроме того, она требует дополнительной обработки сообщений, и надежность соединения зависит от реализации на уровне приложения [6]. Что подразумевает под собой высокие требования к знаниям языков программирования от разработчика систем, которые будут использовать данный протокол.

HTTP — Протокол передачи гипертекста Hypertext Transfer Protocol (HTTP) представляет собой наиболее распространенный способ передачи ресурсов в Интернете [7]. Он определяет правила и стандарты для передачи данных между клиентами и серверами. Он поддерживает передачу веб-страниц и данных между различными приложениями и системами. Простота и широкая поддержка делают его одним из наиболее распространенных протоколов для обмена данными в сети.

MQTT (Message Queuing Telemetry Transport) — это простой протокол обмена сообщениями, разработанный для ограниченных устройств с низкой пропускной способностью.

В ходе анализа перечисленных протоколов передачи данных между микроконтроллерами и клиентскими приложениями, технология WebSocket была исключена из выбора. Поскольку она требует высоких знаний языков программирования, что зачастую является препятствием для инженеров по автоматизации. Кроме того, технология не является целевой для реализаций подобных решений.

Протокол MQTT показывает наилучшие результаты: меньшую скорость передачи данных при схожем объеме данных с HTTP, а также имеет большую пропускную способность, что определяет его как наиболее релевант-

ный выбор при передаче данных от шлюза Интернета Вещей на сервер, при условии беспроводной передачи данных [8].

Описание протокола MQTT

В MQTT есть несколько основных концепций:

- Опубликовать/Подписаться (Publish/Subscribe);
- Сообщения (Messages);
- Топики (Topics);
- MQTT-брокер (Broker).

Первая концепция — это идея публикации и подписки. Она подразумевает, что устройства могут публиковать сообщения на топики или подписываться на определенные топики для получения сообщений.

Например, устройство 1 публикует информацию в топик. Устройство 2 подписано на тот же топик, в которой публикуется устройство 1. Таким образом, согласно правилам протокола, устройство 2 сразу должно получить сообщение.

Вторая концепция протокола — это концепция сообщений. Сообщения — это информация для обмена между устройствами. Например, показания датчиков или команды управления.

Следующая концепция — это способ, которым устройство регистрирует интерес к входящим сообщениям или указывает, куда необходимо опубликовать сообщение, концепция топиков. Топики представлены строками, разделенными косой чертой. Каждая косая черта указывает уровень топика.

Сообщения публикуются в топики. Устройства подписываются на топики. Устройства получают только те сообщения, на топики которых оно подписано.

Протокол оснащен функцией LWT (Last Will and Testament — «последняя воля и завещание»). Она уведомляет все стороны об аварийном отключении сети [9].

За прием всех сообщений, их фильтрацию, определение тех, кто в них заинтересован, а затем публикацию сообщения для всех подписанных клиентов отвечает MQTT-брокер.

Недостатки популярных коммерческих решений mqtt-брокеров

На рынке представлено множество клиентов MQTT брокеров. Например, платформа WQTT.RU, Yandex IoT Core и другие. Они обладают рядом преимуществ, таких как: удобный пользовательский интерфейс, низкие требования к разработчикам системы.

Однако подобные решения обладают существенными ограничениями. Первое ограничение — это обеспечение обмена данными с низким уровнем защиты информации, поскольку доступ к данным получают разработчики сторонних сервисов. Это может не соответствовать требованиям к информационной безопасности разрабатываемых систем. Кроме того, все эти системы являются платными и подразумевают ежемесячную оплату.

Исходя из этого при реализации передачи данных по протоколу mqtt предлагается не использовать сторонние платформы, а выбрать mqtt-брокер с открытым исходным кодом. Его можно развернуть на том же сервере, где будут развернуты другие компоненты системы, например, web-приложение, либо на отдельном облачном сервере. Далее доработать его функционал под свой проект. Это позволяет не платить ежемесячно разработчикам сторонних платформ, а также проектировать систему более гибко, реализуя все необходимые функции без каких-либо ограничений.

Апробация

На текущий момент доступно множество mqtt-брокеров, с открытым исходным кодом. Для апробации был выбран mqtt-брокер aedes, поскольку он имеет публичный репозиторий с хорошо описанной документацией и примерами.

Aedes обеспечивает базовый механизм обмена сообщениями, позволяя клиентам публиковать и подписываться на темы. Aedes автоматически обрабатывает перенаправление сообщений для подписчиков по соответствующим темам, и это не требуется делать явно в коде. Библиотека заботится об этом, основываясь на внутренней логике. Кроме того, Aedes поддерживает механизмы аутентификации и авторизации, что позволяет реализовать различные политики безопасности. Имеется возможность настраивать пользовательские функции проверки подлинности и авторизации доступа к темам.

В рамках апробации в соответствии с правилами протокола MQTT было отправлено сообщение на топик /sensors/1 в формате json, которое содержит значение показания датчика. Сообщение было отправлено с микроконтроллера esp32 на mqtt-брокер с открытым исходным кодом, а после на mqtt-брокер WQTT.RU. Далее был разработан прототип web-приложения, которое подписывается на соответствующий топик и отображает значение датчика.

Для того, чтобы решить поставленную задачу в первую очередь был доработан mqtt-брокер aedes: под-

ключены соответствующие библиотеки, настроено логирование таких событий как подключение клиентов к брокеру, получение сообщений, отправка сообщений. А также прописаны топики, которые будет обрабатывать брокер.

Далее MQTT-брокер был развернут на удаленном сервере. Для этого были проделаны следующие действия:

1. Арендвали сервер в российской компании, предоставляющей облачные инфраструктурные сервисы и услуги дата-центров, Selectel.
2. Подключились к этому серверу через протокол SSH.
3. Установили на сервер git, чтобы загрузить доработанный код mqtt-брокера с платформы github.
4. Установили node.js для возможности запуска разработанного брокера.
5. Установили менеджер процессов pm2, который позволяет постоянно держать приложения запущенными, и запустили с помощью него mqtt-брокер.

После проделанных действий, в программное обеспечение микроконтроллера и веб-приложения был прописан ip-адрес и порт брокера, и значение датчика стало доступно для пользователей.

Для реализации такой же задачи с помощью сторонней платформы WQTT.RU потребовалось совершить меньше действий. Достаточно было оформить подписку, и платформа выдала ip-адрес и порт брокера, к которому мы можем подключаться. Логирование автоматически осуществлялось на платформе.

Заключение

Таким образом, при решении задачи беспроводного обмена данными между физическими устройствами и клиентскими подсистемами в условиях наличия сети Интернет, рекомендуется использовать протокол MQTT. Его реализация подразумевает наличие серверной системы, которая координирует сообщения между разными клиентами, — MQTT-брокера. В зависимости от требований к системе Интернета вещей и компетенций разработчиков можно использовать как сторонние MQTT-брокеры, так и открытые. Однако, открытые обладают рядом существенных преимуществ, описанных в данной статье. Но предполагают наличие дополнительных знаний при их развертывании и доработке, что является преодолимой задачей, если выбирать mqtt-брокеры с наличием качественной документации в их репозиториях.

ЛИТЕРАТУРА

1. Ли Цзюнь, Юй Шуанюань. Актуальность внедрения процесса цифровизации в деятельность предприятий // *Universum: экономика и юриспруденция*. 2021. №11 (86) С. 13–18.
2. Волобуева Т.А. IT-технологии в сельском хозяйстве: перспективы и проблемы использования // *Евразийское Научное Объединение*. 2020. № 8-4 (66). С. 193–196.
3. Ниточкин М. Цифровизация АПК. Модный «хайп» или реальный бизнесинструмент для отрасли // URL: <https://news.myseldon.com/ru/news/index/229059266> (дата обращения 14.08.2020).
4. Носов Андрей Владимирович. Анализ прикладных протоколов передачи данных для систем интернета вещей // *Научные междисциплинарные исследования*. 2021. №2. С. 48–57.
5. Сабуткевич Артем Михайлович. БЕСКОНТАКТНЫЙ КОНТРОЛЬ ПОСЕЩАЕМОСТИ ПЕРСОНАЛА С ИСПОЛЬЗОВАНИЕМ МИКРОКОНТРОЛЛЕРА ESP32 // *Инновационные аспекты развития науки и техники*. 2021. №9. С. 69–73.
6. Шестаков В.С., Сагидуллин А.С. Применение технологии websocket в web-приложениях технологического назначения // *Приборостроение*. 2015. №4. С. 328–330.
7. Benjamin D. HTTP 2.0 Specification / D. Benjamin // W3 — 1995 — [Electronic resource]. URL: <https://www.w3.org/MarkUp/html-spec/> (accessed: 05.12.2021).
8. Курмаев Т.И. СРАВНЕНИЕ ПРОТОКОЛОВ ПЕРЕДАЧИ ДАННЫХ В ИНТЕРНЕТЕ ВЕЩЕЙ // *МНИЖ*. 2022. №1-1 (115) С. 45–47.
9. Якупов Д.Р. ОБЗОР И СРАВНЕНИЕ ПРОТОКОЛОВ ИНТЕРНЕТА ВЕЩЕЙ: MQTT И AMQP // *International Journal of Open Information Technologies*. 2022. №9. С. 90–98.

© Гафаров Булат Рустемович (gafarovbr@student.bmstu.ru)

Журнал «Современная наука: актуальные проблемы теории и практики»