

ПРОТОКОЛ MQTT ТА ЙОГО ЗАСТОСУВАННЯ У СФЕРІ ІОТ

Вінницький Національний Технічний Університет

Анотація

У роботі розглядається протокол MQTT, який є одним із найефективніших рішень для обміну даними в системах Інтернету речей (IoT). Описано його структуру, принцип роботи та приклади використання в реальних системах.

Ключові слова: MQTT, IoT, брокер, датчики, протоколи зв'язку.

Abstract

The paper considers the MQTT protocol, which is one of the most effective solutions for data exchange in Internet of Things (IoT) systems. Its structure, principle of operation, and examples of use in real systems are described.

Keywords: MQTT, IoT, broker, sensors, communication protocols.

Вступ

Інтернет речей об'єднує мільярди пристроїв, які взаємодіють між собою для передачі даних і виконання різноманітних завдань. Ефективність таких систем залежить від надійності та швидкості обміну інформацією.

Існує безліч різноманітних протоколів передачі даних, серед них протокол MQTT є одним із найпоширеніших і найбільш ефективних. Його популярність пояснюється здатністю працювати з мінімальними вимогами до пропускну здатності та енергоспоживання. MQTT реалізує модель "публікація-підписка" для асинхронного обміну повідомленнями, використовуючи TCP як транспортний рівень. Завдяки відкритості стандарту MQTT підтримує широкий спектр реалізацій і залишається одним із ключових елементів сучасної інфраструктури IoT [1].

Що таке MQTT і як він працює

MQTT (Message Queuing Telemetry Transport) — це легкий протокол обміну даними, спеціально розроблений для пристроїв з обмеженими ресурсами, що функціонують у мережах із низькою пропускну здатністю [2]. Основним принципом роботи MQTT є модель "публікація-підписка", яка відрізняється від традиційної "запит-відповідь", що використовується в HTTP. У цій моделі існує центральний вузол — брокер, який виконує функцію посередника між пристроями. Клієнти можуть виступати в ролі видавців або підписників, взаємодіючи через теми.

Теми — це адреси, які структурують обмін даними в MQTT. Наприклад, датчик температури може публікувати дані у тему `building/room2/temperature`. Будь-який пристрій, який підписаний на цей топик, отримає повідомлення з температурою. Брокер виступає як маршрутизатор, забезпечуючи доставку повідомлень відповідно до тем. Теми мають ієрархічну структуру. Крім того, MQTT підтримує символи підстановки. Наприклад, підписка на `building/#` дозволяє отримувати всі повідомлення з будівлі 1, незалежно від кімнати чи типу даних [3].

MQTT підтримує кілька типів повідомлень, які визначають поведінку системи.

- 1) CONNECT — встановлює з'єднання між клієнтом і брокером;
- 2) PUBLISH — надсилає дані на певний топик;
- 3) SUBSCRIBE і UNSUBSCRIBE — дозволяють клієнту підписуватись або відписуватись від топиків;
- 4) DISCONNECT — завершує з'єднання.

Наприклад, датчик вологості надсилає значення у топик `garden/moisture` за допомогою повідомлення PUBLISH. Розумна система поливу, підписана на цей топик, отримує повідомлення й активує полив, якщо значення вологості низьке.

Quality of Service (QoS) — це принцип, який описує, як саме брокер гарантує доставку повідомлень. Існує три рівні:

- 1) QoS 0 ("Щонайбільше один раз"). Повідомлення надсилається без підтвердження. Це швидкий, але ненадійний метод. Використовується, наприклад, для даних, які оновлюються дуже часто, як показники температури.
- 2) QoS 1 ("Щонайменше один раз"). Забезпечує доставку повідомлення принаймні один раз, але при цьому можливі дублікати. Це означає, що відправник повторює передачу повідомлення, доки не отримає підтвердження доставки від отримувача.
- 3) QoS 2 ("Рівно один раз"). В MQTT гарантує доставку кожного повідомлення без дублювання шляхом чіткої послідовності підтверджень між відправником і брокером. Спочатку відправник публікує повідомлення (PUBLISH), брокер підтверджує його отримання (PUBREC), потім відправник повідомляє про готовність завершити передачу (PUBREL), і брокер підтверджує завершення процесу (PUBCOMP). Цей механізм забезпечує найвищу надійність, але супроводжується збільшеними затримками та навантаженням через додаткові кроки обміну.

Протокол працює на базі TCP/IP, що забезпечує надійне з'єднання між клієнтами та брокером. Пристрій, який відправляє дані, не взаємодіє напряму з пристроєм, що їх отримує [4]. Натомість він надсилає повідомлення брокеру, який аналізує його та надсилає відповідним підписникам. Це знижує вимоги до апаратного забезпечення та спрощує управління з'єднаннями в масштабних системах.

Висновки

Протокол MQTT є важливим елементом сучасних IoT-рішень, оскільки забезпечує гнучкий і надійний обмін даними навіть у слабких мережах. Завдяки моделі "публікація-підписка" та підтримці рівнів QoS, він адаптується до різних пристроїв і умов. Його основними перевагами є низькі вимоги до ресурсів, простота інтеграції та масштабованість, що робить його ідеальним для багатьох IoT-проектів. Водночас недоліками є залежність від центрального брокера та можливі затримки при високих рівнях QoS. Попри це, відкритість і універсальність MQTT роблять його основою для створення ефективних і надійних IoT-екосистем.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Пояснення що таке MQTT [Електронний ресурс]. – Режим доступу: <https://research.rozumni-ideyi.com/poiasnennia-shcho-take-mqtt/>
2. What is MQTT? - MQTT Protocol Explained - AWS [Електронний ресурс]. – Режим доступу: <https://aws.amazon.com/what-is/mqtt/>
3. MQTT Explained: The Complete Guide [Електронний ресурс]. – Режим доступу: <https://cedalo.com/blog/complete-mqtt-protocol-guide/>
4. MQTT: Architecture, Message Structure, Benefits, And Security In IoT [Електронний ресурс]. – Режим доступу: <https://talkingiot.io/understanding-mqtt-architecture-message-structure-benefits-and-security-in-iot/>

Черневський Назар Олександрович — студент групи 2СП-21б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький Національний Технічний Університет, Вінниця, e-mail: chernevskijnazar@gmail.com

Chernevskiy Nazar Oleksandrovich — student of group 2SP-21b, faculty of information technologies and computer engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: chernevskijnazar@gmail.com