

АДАПТИВНА МУЛЬТИПРОТОКОЛЬНА АРХІТЕКТУРА ІОТ СИСТЕМ

Самсонов С.С., Глоба Л.С.

*Навчально-науковий інститут телекомунікаційних
систем КПІ ім. Ігоря Сікорського, Україна
E-mail: s.samsonov@kpi.ua*

ADAPTIVE MULTIPROTOCOL IOT SYSTEM ARCHITECTURE (AMISA)

This paper presents a brief review of application-layer protocols for resource-constrained IoT devices (MQTT, CoAP, LwM2M) and proposes an adaptive multiprotocol architecture (AMISA). The architecture allows smart switching between different protocols to optimize energy usage, reliability, and delivery guarantees in unstable network conditions.

Дослідження наукової літератури демонструє наявність великої кількості досліджень комунікаційних протоколів IoT систем. Такі протоколи прикладного рівня як Message Queuing Telemetry Transport (MQTT), Constrained Application Protocol (CoAP), Lightweight M2M (LwM2M) вважаються фундаментальними для використання для IoT систем з обмеженими ресурсами [1] та забезпечують енергоефективність, надійність та безпечну передачу даних.

Статті що проводять аналіз та порівняння різних протоколів прикладного рівня [2] відзначають що CoAP має мінімальну надлишкову сервісну інформацію пакетів, втім не може гарантувати 100% доставку пакетів в поганих умовах передачі, а MQTT зарекомендував себе дуже гарно для пристроїв з обмеженим ресурсом батареї в певних сценаріях. В дослідженні що глибоко порівнює MQTT, CoAP та HTTP підкреслюють сильні та слабкі сторони цих протоколів, зважаючи розмір повідомлень та хедерів, енергоспоживання, затримок та використання різних рівнів QoS (або їх аналогів) [3] [4].

Проте, у наявній літературі бракує описів IoT-систем зі «смарт» передавачами, які здатні динамічно оцінювати стан мережі та перемикатися між кількома протоколами. У такому підході важливо зважати на практичні обмеження. Наприклад, оскільки IoT пристрої є обмеженими у ресурсах, перемикання між більше ніж 2 протоколами було б непрактичним. Крім того, такий підхід потребує відповідної бекенд-інфраструктури. На практиці це означає наявність «інтерконектора протоколів» для одного з протоколів [5] і «шлюзу протоколу» для іншого.

На рис. 1 представлено адаптивну мультипротокольную архітектуру IoT системи (Adaptive Multiprotocol IoT System Architecture, AMISA), що передбачає належну бекенд-інфраструктуру для ефективного використання багатопрокольних IoT-передавачів у різних сценаріях. На відміну від існуючих типових багатопрокольних рішень, AMISA додає ще два вузли для координації роботи таких пристроїв.



Рис.1. Адаптивна мультипротокольна архітектура IoT системи (AMISA).

Нижче розглянуто призначення кожного вузла:

IoT передавач — це передавальний пристрій, який збирає та/або агрегує дані і надсилає їх у хмару. Він обирає один із двох прикладних протоколів для передачі чергового повідомлення залежно від поточного стану мережі та конфігурації, що надходить від бекенд системи.

Інтерконектор протоколів одночасно є шлюзом для повідомлень, надісланих за допомогою Протоколу 2, і перетворювачем цих повідомлень у повідомлення Протоколу 1. Після «перекладу» повідомлення передаються далі через шлюз Протоколу 1.

Шлюз протоколу 1 виступає «стандартною» точкою входу всіх повідомлень у систему — як безпосередньо від IoT-відправника, так і від вузла протокольного інтерконектора. За потреби цей вузол додає метадані та перенаправляє повідомлення до брокера IoT-повідомлень.

Брокер IoT-повідомлень зберігає отримані повідомлення від IoT-пристроїв. Оскільки повідомлень може бути дуже багато, наявність брокера підвищує надійність, відмовостійкість і масштабованість бекенд-системи та дозволяє різним сервісам читати лише потрібні їм дані. Такий вузол є досить типовим для систем із великою кількістю подій.

Споживач даних бізнес-рівня і сервіси бізнес-логіки відображають базову бізнес-логіку системи та не є основним предметом цього дослідження.

Споживач телеметрії сигналу та метаданих отримує релевантні дані, які допомагають визначити стан мережевого середовища (наприклад, відношення сигнал-шум), рівень заряду батареї пристрою, тип повідомлення (наприклад, критичний звіт про збій чи звичайне «хартбіт»-повідомлення), а також

ефективність доставки (наприклад, затримка, кількість повторних спроб, рівень QoS, використаний протокол).

Аналізатор метаданих обробляє дані, отримані від попереднього вузла. На базі великих вибірок статистичної інформації цей вузол приймає рішення, який протокол бажано використовувати за певних умов, і передає це рішення далі до модуля зміни конфігурації. Логіка прийняття рішення може бути реалізована за допомогою машинного навчання або ж у вигляді попередньо запрограмованих алгоритмів.

Модуль зміни конфігурації приймає рішення від попереднього вузла, трансформує його у формат придатний для оновлення конфігурації на IoT-передавачі, і надсилає таке оновлення.

В цій архітектурі важливе значення має саме оновлення конфігурації, а не прошивки передавача. Якщо ПЗ IoT передавача буде достатньо гнучким — бажаний ефект оновлення логіки вибору протоколу для відправки повідомлення може бути досягнутий за допомогою оновлення параметрів конфігурації. В порівнянні з оновленням прошивки це дозволить менше навантажувати мережу великою кількістю даних та уникнути ризику виходу IoT пристрою з ладу під час оновлення прошивки. При цьому слід пам'ятати що такі оновлення конфігурації не повинні відбуватися занадто часто: це дозволить аналізатору метаданих накопичити достатньо даних для аналізу та не перевантажувати мережу частими пакетами з оновленням конфігурації.

Література

1. Sharma C., Gondhi N.K. Communication protocol stack for constrained IoT systems : Proc. 2018 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU) (25–27 Feb. 2018; Bhimtal, India). – 2018. – P. 1–6.
2. Al-Masri E., Mahmoud Q.H., Pham D., Ezeobi C., Abdelrahman S.M., Yan C., Ali A.K. Investigating messaging protocols for the internet of things (IoT) // IEEE Access. – 2020. – Vol. 8. – P. 94880–94911.
3. Naik N. Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP: Proc. 2017 IEEE International Systems Engineering Symposium (ISSE) (11–13 Oct. 2017; Vienna, Austria). – 2017. – P. 1–7.
4. Sreeraj S., Kumar G.S. Performance of IoT protocols under constrained network: a use case based approach : Proc. 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT) (15–17 Feb. 2018; Chennai, India). – 2018. – P. 495–498.
5. Dave M., Doshi J., Arolkar H. MQTT-CoAP interconnector: IoT interoperability solution for application layer protocols : Proc. 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) (7–9 Oct. 2020; Palladam, India). – 2020. – P. 122–127.