

ВИБІР ПРОТОКОЛІВ ДЛЯ ПЕРЕДАЧІ ДАНИХ З ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ ДО ПОСТАЧАЛЬНИКІВ ХМАРНИХ ПОСЛУГ

А. О. Шихмат, З. Є. Верес

Національний університет “Львівська політехніка”,
кафедра комп’ютеризованих систем автоматички
E-mail: anton.o.shykhmat@lpnu.ua, zenovii.y.veres@lpnu.ua

© Шихмат А. О., Верес З. Є., 2024

Інтернет речей (IoT) уможливило побудову мереж між пристроями, людьми, додатками та інтернетом, створюючи нові екосистеми з вищою продуктивністю, енергоефективністю та прибутковістю. Вузли в мережах повинні мати здатність спілкуватися та обмінюватися даними. Для цього використовують протоколи передавання даних, проте вибір певного протоколу для конкретного випадку використання не завжди є очевидним. У статті наведено огляд двох протоколів передавання даних: MQTT та HTTP, що пропонують IoT Core рішення хмарних постачальників AWS і GCP, порівняно кількість тарифікованого трафіку, що виробляє кожен протокол, ефективність витрат на протоколи. Дослідження показало, що порівняно з AWS IoT Core, GCP IoT Core є дорожчим для всіх оцінених сценаріїв і його не рекомендовано використовувати. Для сценаріїв з частим передаванням даних найкращим рішенням є використання MQTT-мосту, що надається AWS IoT Core. Якщо кількість підключених пристроїв перевищує 10 млн з високою частотою передавання даних – кожен 1 хвилину, то доцільно розглянути використання самостійного MQTT-брокера або іншого протоколу на основі TCP, наприклад, CoAP. Якщо дані передають не так часто (кожні 10 хвилин або рідше), HTTP-міст може бути рішенням для кількості до 100 млн пристроїв. У результаті дослідження створено дерево рішень для вибору найкращого протоколу для конкретних випадків використання.

Ключові слова: IoT; протоколи передачі даних; HTTP; MQTT; AWS; GCP; IoT Core.

Вступ

Інтернет речей (IoT) широко використовується в усіх аспектах нашого життя. IoT дає змогу створювати пристрої, що допомагають розпізнавати стан речей, що надає їм перевагу передбачення потреб певної особи або системи на основі зібраної контекстної інформації. COVID-19 збільшив попит на віддалену роботу. Це створює завдання збирання, обробки, зберігання та аналізу отриманих даних. Здатність керувати величезною кількістю пристроїв у системі – складне завдання. Велика кількість пристроїв спричиняє додаткові витрати на створення надійного рішення для отримання телеметрійних даних, перевірки стану пристроїв та прогнозування відключених/несправних пристроїв. Хмарні рішення загалом вважають основою для побудови рішення в галузі IoT. Найпростіший (і водночас найскладніший) підхід – використовувати можливості хмарного обчислення налаштувати всі необхідні компоненти самостійно. Однак хмарні провайдери AWS та GCP надають модулі Cloud IoT Core [1] для налаштування, керування та приймання телеметрії в хмару. Обидва рішення надають сервіс IoT Core, що підтримує передавання даних за допомогою двох широко використовуваних протоколів у галузі IoT: MQTT та HTTP.

1. Аналіз останніх досліджень та публікацій

Дослідження враховують переважно лише характеристики протоколів із їх порівнянням [2] та об'єм трафіку, що передається по мережі [3–5].

У роботі [2] проаналізовано переваги та недоліки протоколів HTTP та MQTT. До переваг MQTT належать: невелика затримка передавання і обробки даних, низьке споживання енергії, підтримка відправлення одним запитом одного повідомлення багатьом отримувачам, висока надійність, можливість використання у мережах з обмеженою пропускну здатністю. Проте складність розроблення клієнтів та унеможливлення обміну повідомлення у разі несправності брокера є недоліками MQTT. Перевагами протоколу HTTP є легкість розроблення клієнтів, можливість кешування відповідей та підтвердження доставки даних. До недоліків зараховують низьку надійність, оскільки не передбачено автоматичне повторне відправлення повідомлення у разі помилки опрацювання цього повідомлення на сервері, обмежену можливість використання у мережах з обмеженою пропускну здатністю та неможливість відправити одним запитом одне повідомлення багатьом отримувачам.

У результаті дослідження [3] автори виявили, що MQTT ефективніший для передавання даних у реальному часі та має меншу кількість трафіку порівняно з HTTP, оскільки кожен запит до інформації у HTTP встановлює нове з'єднання, що збільшує накладні витрати мережевих ресурсів. Зокрема, якщо кількість підключених пристроїв зростає, як зазвичай у типових додатках для Інтернету речей, ці накладні витрати в HTTP стають критичними.

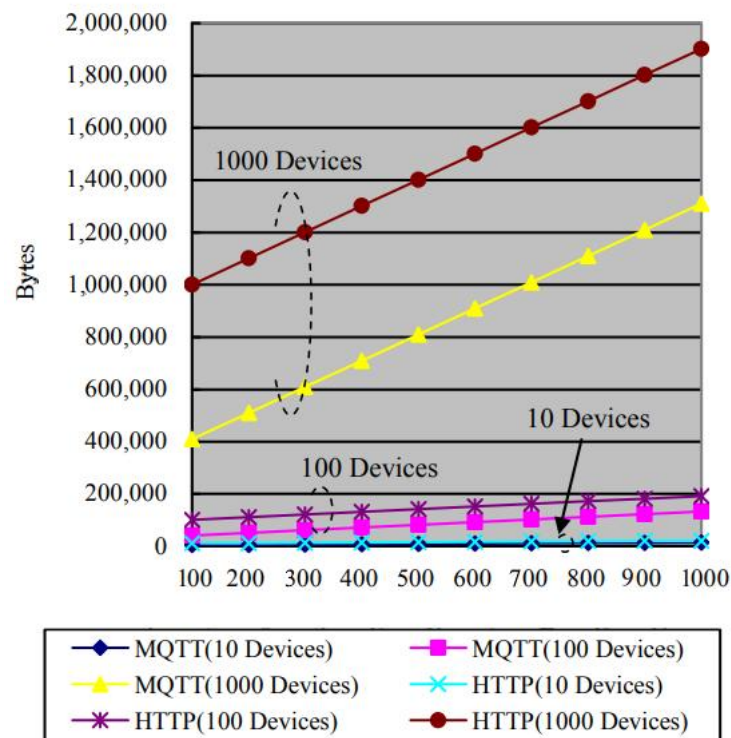


Рис. 1. Характеристики розміру корисного навантаження у HTTP та MQTT [3]

Ці дослідження також виявили, що для сценаріїв з великою кількістю пристроїв MQTT має перевагу порівняно з HTTP, оскільки може працювати з більшою кількістю підключених пристроїв. Однак для передавання великих об'ємів даних HTTP є ефективнішим, оскільки може передавати дані у великих блоках.

У праці [4] наведено аналіз додаткового часу, необхідного на доставку повідомлення із використанням безпечного з'єднання, а саме шифрування даних алгоритмом RSA у HTTP протоколі та

використання WSS у MQTT. Результати виявили, що в середньому час доставки повідомлення збільшився на 1400 мілісекунд у випадку HTTP та на 300 мілісекунд для MQTT. На додачу дослідження [5] вказує, що MQTT має менший час затримки та споживає менше енергії порівняно з HTTP, а саме те, що споживання енергії протоколом HTTP у разі синхронної комунікації вище на 92 % порівняно з MQTT, та на 20 % за асинхронної комунікації.

Проте жодне із досліджень не враховує об'єми саме тарифікованого трафіку та грошові витрати, необхідні для використання протоколів залежно від різних сценаріїв використання, адже швидкодія чи легкість розроблення не є єдиними важливими факторами під час вибору технологій. У них також не проаналізовано доцільності використання IoT Core рішень від хмарних постачальників AWS та GCP порівняно із самостійним розгортанням усієї необхідної інфраструктури для роботи з MQTT та HTTP протоколами.

2. Постановка задачі

Дослідити вартість використання протоколів MQTT та HTTP із використанням IoT Core рішень від постачальників хмарних послуг AWS та GCP для різних сценаріїв використання, залежно від частоти передавання повідомлень, кількості пристроїв, що передають дані, проміжку часу, впродовж якого пристрій передає дані, та створити дерево рішень, яке дасть змогу вибрати найоптимальніші за грошовими витратами протокол та хмарне рішення.

3. Огляд протоколів передавання даних, що надаються постачальниками хмарних послуг

Обидва постачальник хмарних послуг AWS та GCP у своїх IoT Core рішеннях передбачають використання лише протоколів MQTT та HTTP.

MQTT є протоколом обміну повідомленнями для Інтернету речей (IoT). Його розробив Комітет технічного стандартизування MQTT у межах OASIS. Це відкритий, простий та розроблений з урахуванням легкості реалізації протокол, що робить його ідеальним для використання в різних ситуаціях, зокрема в обмежених середовищах, наприклад, для комунікації у контексті “Машини-до-Машини” (M2M) та Інтернету речей (IoT), де потрібний невеликий розмір коду та/або обмежена пропускна здатність мережі [6].

MQTT забезпечує можливість підключення пристрою незалежно від передавання будь-яких даних. Брокер відстежує підключені пристрої за допомогою функції Keep-Alive. Keep-Alive – це інтервал часу, вимірюваний у секундах. Він виражається як 16-бітне слово; це максимальний часовий інтервал, який дозволено від моменту завершення передавання клієнтом одного контрольного пакета до моменту, коли він починає відправляти наступний. Клієнт зобов'язаний забезпечити, щоб інтервал між відправкою контрольних пакетів не перевищував значення Keep-Alive. За відсутності відправки інших контрольних пакетів клієнт ОБОВ'ЯЗКОВО повинен відправляти пакет PINGREQ.

Коли втрачається з'єднання, брокер може відправити останнє повідомлення Last Will and Testament (LWT) клієнта. Повідомлення може бути використане як тригерний механізм для сповіщення користувача про проблему та передбачення причини відключення. Однак відключення може відбуватись через проблеми зі зв'язком і може бути хибним сигналом. Внаслідок цього для визначення відмови пристрою потрібний надійніший підхід.

Стандарт MQTT визначає три рівні якості обслуговування (QoS) [6]:

1. QoS 0, повідомлення доставлено не більше від одного разу.
2. QoS 1, повідомлення доставлено принаймні один раз.
3. QoS 2, повідомлення доставлено точно один раз.

Варто зауважити, що Cloud IoT Core не підтримує QoS 2. Публікація повідомлень з QoS 2 призводить до закриття з'єднання. Згідно з документацією GCP, Cloud IoT Core обмежує максимальний період бездіяльності, встановивши час очікування на 20 хвилин: “З'єднання клієнта автоматично буде припинено, якщо клієнт не надсилає жодних повідомлень протягом 20 хвилин, навіть якщо інтервал тримання зв'язку є довшим. Якщо значення підтримання зв'язку не встановлено, тайм-аут очікування на 20 хвилин діє за замовчуванням”.

HTTP створений як складова частина Всесвітньої павутини для передавання документів. Користувачі інтернету знають його передусім як технологію, що дає змогу працювати веббраузерам. Сервери містять ресурси, ідентифіковані URL-адресами, до яких HTTP-клієнти зазвичай можуть надсилати запити. HTTP є “безз’єднувальним” протоколом: пристрої не підтримують з’єднання з Cloud IoT Core із використанням моста HTTP. Замість цього вони надсилають запити та отримують відповіді. Cloud IoT Core обох хмарних постачальників AWS і GCP підтримує лише HTTP 1.1 (не 2.0). Міст можна використовувати для регулярного відправлення стану пристрою до IoT Core.

REST – архітектурний стиль для створення вебсервісів на основі протоколу HTTP. Сервіси, які підтримують цей стиль, називають RESTful-сервісами. Такі сервіси не зберігають стан клієнта, що робить їх використання швидким, надійним та масштабованим. У відповідь на запити, надіслані до URI ресурсу, RESTful-сервіси часто відповідають у форматах HTML, JSON або XML (але не обмежуються ними). RESTful-сервіси найчастіше використовують чотири методи HTTP [7]:

1. GET: використовується для отримання інформації про ресурс, але не для його зміни.
2. POST – для створення нових ресурсів.
3. PUT: використовується для оновлення наявних ресурсів.
4. DELETE – для видалення поточного ресурсу.

4. Аналіз об’єму тарифікованого трафіку із використанням протоколів HTTP та MQTT у Cloud IoT Core сервісах

Згідно з тестами, виконаними для доставки 1К повідомлень за протоколами MQTT та HTTP, MQTT у шість разів швидше виконує завдання публікації стійких даних і є ефективнішим з погляду споживання енергії [8]. Для порівняння обсягу тарифікованого трафіку, який стягується GCP та AWS, було оцінено набір сценаріїв, що представляють найпоширеніші шаблони [9] для передавання даних в галузі Інтернету речей:

1. Розмір повідомлення 1Кб, дані передаються щохвилини.
2. Розмір повідомлення 1Кб, дані передаються кожні 5 хвилин.
3. Розмір повідомлення 1Кб, дані передаються кожні 10 хвилин.
4. Розмір повідомлення 1Кб, дані передаються кожні 15 хвилин.
5. Розмір повідомлення 1Кб, дані передаються кожні 20 хвилин.
6. Розмір повідомлення 1Кб, дані передаються кожні 30 хвилин.
7. Розмір повідомлення 1Кб, дані передаються щогодини.
8. Розмір повідомлення 1Кб, дані передаються кожні 2 години.
9. Розмір повідомлення 1Кб, дані передаються кожні 3 години.
10. Розмір повідомлення 1Кб, дані передаються кожні 6 годин.

Для моста MQTT повідомлення PINGREQ або повідомлення з даними мають надсилатися як мінімум один раз на 20 хвилин для збереження відкритого з’єднання [10]. В GCP повідомлення PINGREQ та повідомлення з даними тарифікуються однаково. AWS не тарифікує повідомлення PINGREQ, але тарифікує час з’єднання. Мінімальний розмір тарифікованого повідомлення – 1 Кб, навіть якщо повідомлення само по собі містить лише кілька байтів. Для HTTP моста як GCP, так і AWS тарифікують кожен запит та відповідь з передаванням даних. Мінімальний розмір тарифікованого повідомлення також становить 1 Кб. Мости HTTP та MQTT проаналізовані з такими налаштуваннями, коли повідомлення PINGREQ надсилається з кожного пристрою кожні 20 хвилин для 10 тис., 100 тис., 1 млн, 10 млн та 100 млн пристроїв. Розрахунки передбачають, що кожен пристрій підключається/перепідключається до моста MQTT тільки один раз на день (підключення тарифікуються як повідомлення 1 Кб у GCP, для AWS тарифікуються як розмір повідомлення, отже, припустимо, що вони становлять 1 Кб). Детальніші розрахунки наведеної нижче для сценаріїв 1, 5, 6 та 7.

Сценарій 1: повідомлення розміром 1 Кб надсилається щохвилини

Як видно з табл. 1, частота передавання даних додає значну кількість трафіку. Повідомлення PINGREQ становлять приблизно 5 % від загального трафіку і є порівняно малими, і їхній внесок можна ігнорувати під час розрахунків. HTTP-міст використовує майже вдвічі більше повідомлень, оскільки кожен запит і відповідь окремо оплачуються. Тому HTTP-міст непридатний для сценарію з високою частотою передавання повідомлень.

Таблиця 1

Тарифікований трафік для MQTT та HTTP мостів для сценарію 1

Кількість пристроїв	Трафік повідомлень PINGREQ, Мб/місяць	Трафік з'єднання, Мб/місяць	Трафік телеметрійних повідомлень, Мб/місяць	Сумарний тарифікований трафік AWS MQTT, Мб/місяць	Сумарний тарифікований трафік GCP MQTT, Мб/місяць	Сумарний тарифікований трафік HTTP, Мб/місяць
10К	21,6К	300	432,3К	432,6К	453,9К	864К
100К	216К	3К	4,323М	4,326М	4,539М	8,64М
1М	2,16М	30К	43,23М	43,26М	45,39М	86,4М
10М	21,6М	300К	432,3М	432,6М	453,9М	864М
100М	216М	3М	4,323В	4,326В	4,539В	8,64В

Сценарій 5: повідомлення розміром 1 Кб надсилається кожні 20 хвилин

Як видно з табл. 2, обсяг трафіку, який тарифікується у випадку використання HTTP мосту, майже ідентичний обсягу тарифікованого трафіку в разі використання MQTT в GCP (для AWS обсяг трафіку, що тарифікується для MQTT, все ще вдвічі менший). Відмінність між мостами зумовлена припущенням про частоту підключення/перепідключення MQTT. Це припущення додає приблизно 0,7 % додаткового трафіку. Основний варіант зниження вартості трафіку, який тарифікується, – збільшення інтервалу відправки повідомлення PINGREQ від 20 хв до максимально можливого значення для MQTT згідно зі специфікацією – 18 год [7]. Це зменшить трафік PINGREQ на пристрій у 54 рази, як зазначено в табл. 3.

Таблиця 2

Тарифікований трафік для MQTT та HTTP мостів для сценарію 5

Кількість пристроїв	Трафік повідомлень PINGREQ, Мб/місяць	Трафік з'єднання, Мб/місяць	Трафік телеметрійних повідомлень, Мб/місяць	Сумарний тарифікований трафік AWS MQTT, Мб/місяць	Сумарний тарифікований трафік GCP MQTT, Мб/місяць	Сумарний тарифікований трафік HTTP, Мб/місяць
10К	21,6К	300	21,6К	21,9К	43,5К	43,2К
100К	216К	3К	216К	219К	435К	432К
1М	2,16М	30К	2,16М	2,19М	4,35М	4,32М
10М	21,6М	300К	21,6М	21,9М	43,5М	43,2М
100М	216М	3М	216М	219М	435М	432М

Збільшення інтервалу відправки PINGREQ повідомлень зменшує кількість тарифікованого трафіку практично на 49 % для MQTT-моста у GCP і не впливає на AWS. Оскільки за кожне HTTP-передавання стягують як за два 1 Кб повідомлення порівняно з 1 Кб для MQTT, MQTT-міст є дешевшим і бажанішим підходом для передавання даних для сценаріїв з частими передаваннями даних (менше ніж 20 хв) і стабільними мережевими з'єднаннями. Трафік з'єднання показує, що MQTT є кращим варіантом порівняно з HTTP-мостом, навіть якщо повторне підключення відбувається кожні 20 хв – трафік PINGREQ дорівнюватиме трафіку телеметрійних повідомлень.

Таблиця 3

**Тарифікований трафік для MQTT та HTTP мостів для сценарію 5
при частоті PINGREQ 18 годин**

Кількість пристроїв	Трафік повідомлень PINGREQ, Mb/місяць	Трафік з'єднання, Mb/місяць	Трафік телеметрійних повідомлень, Mb/місяць	Сумарний тарифікований трафік AWS MQTT, Mb/місяць	Сумарний тарифікований трафік GCP MQTT, Mb/місяць	Сумарний тарифікований трафік HTTP, Mb/місяць
10К	400	300	21,6К	21,9К	22,3К	43,2К
100К	4К	3К	216К	219К	223К	432К
1М	40К	30К	2,16М	2,19М	2,23М	4,32М
10М	400К	300К	21,6М	21,9М	22,3М	43,2М
100М	4М	3М	216М	219М	223М	432М

Сценарій 6: повідомлення розміром 1 Кб надсилається кожні 30 хвилин

Цей сценарій оцінюється з такими припущеннями:

1. Повідомлення PINGREQ доставляється кожні 20 хв для MQTT-моста.
2. Підключення/перепідключення виконується лише один раз на день для MQTT-моста.

Як видно з табл. 4, 30-хвилинний інтервал передавання телеметрії потребує від MQTT-моста в GCP на 26 % більше трафіку, ніж HTTP міст, але MQTT-міст AWS потребує на 48,95 % менше трафіку, ніж HTTP міст.

Таблиця 4

Тарифікований трафік для MQTT та HTTP мостів для сценарію 6

Кількість пристроїв	Трафік повідомлень PINGREQ, Mb/місяць	Трафік з'єднання, Mb/місяць	Трафік телеметрійних повідомлень, Mb/місяць	Сумарний тарифікований трафік AWS MQTT, Mb/місяць	Сумарний тарифікований трафік GCP MQTT, Mb/місяць	Сумарний тарифікований трафік HTTP, Mb/місяць
10К	21,6К	300	14,4К	14,7К	36,3К	28,8К
100К	216К	3К	144К	147К	363К	288К
1М	2,16М	30К	1,44М	1,47М	3,63М	2,88М
10М	21,6М	300К	14,4М	14,7М	36,3М	28,8М
100М	216М	3М	144М	147М	363М	288М

Сценарій 7: повідомлення розміром 1 Кб надсилається щогодини

Цей сценарій оцінюється з такими припущеннями:

1. Повідомлення PINGREQ надсилається кожні 20 хв для моста MQTT.
2. Підключення/перепідключення виконується лише один раз на день для моста MQTT.

Таблиця 5

Тарифікований трафік для MQTT та HTTP мостів для сценарію 7

Кількість пристроїв	Трафік повідомлень PINGREQ, Mb/місяць	Трафік з'єднання, Mb/місяць	Трафік телеметрійних повідомлень, Mb/місяць	Сумарний тарифікований трафік AWS MQTT, Mb/місяць	Сумарний тарифікований трафік GCP MQTT, Mb/місяць	Сумарний тарифікований трафік HTTP, Mb/місяць
10К	21,6К	21,6К	300	7,2К	7,5К	29,1К
100К	216К	216К	3К	72К	75К	291К
1М	2,16М	2,16М	30К	720М	750К	2,91М
10М	21,6М	21,6М	300К	7,2М	7,5М	29,1М
100М	216М	216М	3М	72М	75М	291М

Як показано в табл. 5, повідомлення PINGREQ додають більше ніж 74 % тарифікованого трафіку до моста MQTT GCP. Цей відсоток зростає, коли доставка повідомлень відбувається рідше: повідомлення PINGREQ додають 83 % рахункового трафіку, якщо телеметрійне повідомлення доставляється кожні дві години, то воно додає 90 % трафіку для сценарію 10 (телеметрійне повідомлення доставляється кожні 6 год).

5. Аналіз грошових витрат на використання протоколів HTTP та MQTT у Cloud IoT Core сервісах

Об'єм тарифікованого трафіку – не єдиний критерій для вибору рішення для підключення IoT-пристроїв до хмарного постачальника. Грошові витрати також є ключовим фактором для бізнесу, який обмежує проектування архітектури. Як ми вже зазначали, AWS не стягує плату за повідомлення PINGREQ, але стягує за час підключення пристрою. У табл. 6 подано розрахунок витрат для різних сценаріїв передавання з використанням GCP MQTT моста із частотою PINGREQ 20 хв; у табл. 7 – для AWS MQTT моста, коли пристрій підключений до моста цілодобово; у табл. 8 – для GCP HTTP моста, а у табл. 9 – для AWS HTTP моста (сценарії 6–10). Розрахунок виконано за допомогою калькуляторів вартості, що надають хмарні постачальники.

HTTP міст може використовуватися для передавання даних від 10 тисяч до 1 мільйона пристроїв, якщо це відбувається кожні 30 хв або рідше. Порівняння цін показує величезне збільшення між 1 мільйоном і 10 мільйонів пристроїв у GCP. Хоча кількість пристроїв зросла в 10 разів, витрати на MQTT GCP збільшилися в 18 разів для передавання даних з частотою 6 годин і майже в 20 разів для частоти передавання кожні 30 хв. Вартість моста HTTP також зростає в 10 разів між 1 мільйоном і 10 мільйонів пристроїв. Для AWS ситуація інша. Вартість повідомлення зменшується зі збільшенням кількості трафіку, і, як бачимо, протокол HTTP дешевший, навіть якщо виробляє більше трафіку для розрахунку. Також можна помітити, що міст MQTT AWS є утричі дешевший, ніж міст MQTT GCP, а міст HTTP AWS є – у шість разів дешевший за міст HTTP GCP.

Таблиця 6

Грошові витрати для трафіку GCP MQTT моста

Кількість пристроїв	Частота відправлення даних із пристрою				
	30 хвилин	1 година	2 години	3 години	6 годин
10К	\$162,23	\$129,83	\$113,63	\$108,23	\$102,82
100К	\$1 350,50	\$1 206,5	\$1 134,50	\$1 092,39	\$1 038,38
1М	\$7 884,50	\$6 444,5	\$5 724,50	\$5 484,50	\$5 244,5
10М	\$150 848,88	\$118 448,88	\$102 248,88	\$96 848,88	\$91 448,88
100М	\$1 620 998,88	\$1 296 998,88	\$1 134 998,88	\$1 080 998,88	\$1 026 998,87

Таблиця 7

Грошові витрати для трафіку AWS MQTT моста

Кількість пристроїв	Частота відправлення даних із пристрою				
	30 хвилин	1 година	2 години	3 години	6 годин
10К	\$49,64	\$42,34	\$38,69	\$37,47	\$36,26
100К	\$496,40	\$423,4	\$386,9	\$374,7	\$362,60
1М	\$4 872	\$4 234	\$3 869	\$3 747	\$3 622,6
10М	\$45 960	\$40 850	\$38 160	\$37 184	\$36 216
100М	\$453 300	\$402 200	\$376 650	\$368 110	\$359 640

Таблиця 8

Грошові витрати для трафіку GCP HTTP моста

Кількість пристроїв	Частота відправлення даних із пристрою				
	30 хвилин	1 година	2 години	3 години	6 годин
10К	\$128,48	\$63,68	\$31,28	\$20,47	\$9,67
100К	\$1 200,50	\$646,88	\$322,88	\$214,87	\$106,87
1М	\$6 384,50	\$3 504,50	\$2 064,50	\$1 584,50	\$1 078,88
10М	\$117 098,88	\$52 298,88	\$19 898,88	\$10 224,50	\$5 424,50
100М	\$1 283 498,88	\$635 498,88	\$311 498,88	\$203 498,88	\$95 498,88

Таблиця 9

Грошові витрати для трафіку AWS HTTP моста

Кількість пристроїв	Частота відправлення даних з пристрою				
	30 хвилин	1 година	2 години	3 години	6 годин
10К	\$29,2	\$14,6	\$7,3	\$4,87	\$2,43
100К	\$292	\$146	\$73	\$48,7	\$24,3
1М	\$2 536	\$1 368	\$730	\$487	\$243
10М	\$21 140	\$10 920	\$5 810	\$4 096	\$2 144
100М	\$205 100	\$102 900	\$51 800	\$34 790	\$17 710

Як задувалося в розділі “Сценарій 5”, MQTT GCP може бути налаштований для доставки повідомлення PINGREQ кожні 18 год, щоб зекономити трафік. Витрати для таких умов (сценарії 1–5) представлені нижче у табл. 10.

Таблиця 10

Грошові витрати для трафіку GCP MQTT моста за частоти PINGREQ 18 годин

Кількість пристроїв	Частота відправлення даних із пристрою				
	1 хвилина	5 хвилин	10 хвилин	15 хвилин	20 хвилин
10К	\$1 489,90	\$390,83	\$196,42	\$131,63	\$99,23
100К	\$9 278,50	\$2 366,50	\$1 502,50	\$1 214,50	\$1 002,37
1М	\$182 213,88	\$26 693,88	\$9 404,50	\$6 524,50	\$5 084,50
10М	\$1 934 648,87	\$379 448,88	\$185 048,88	\$120 248,87	\$87 848,88
100М	\$19 458 998,88	\$3 906 998,87	\$1 962 998,87	\$1 314 998,88	\$990 998,87

Щодо AWS можна зменшити витрати, якщо не потрібно підключати пристрій до MQTT місця на увесь день, а лише на певні робочі години. Витрати на 8 год підключення на день (сценарії 1–5) наведено в табл. 11 (для MQTT) і в табл. 12 (для HTTP).

Таблиця 11

Грошові витрати для трафіку AWS MQTT моста за під'єднання 8 годин на день

Кількість пристроїв	Частота відправлення даних із пристрою				
	1 хвилина	5 хвилин	10 хвилин	15 хвилин	20 хвилин
10К	\$157,68	\$40,88	\$26,28	\$21,41	\$18,98
100К	\$1 484,8	\$408,8	\$262,8	\$214,1	\$189,8
1М	\$12 088	\$3 704	\$2 536	\$2 141	\$1 898
10М	\$114 580	\$32 820	\$22 600	\$19 191	\$17 490
100М	\$1 139 500	\$321 900	\$219 700	\$185 610	\$168 600

Загалом AWS IoT Core в п'ять разів дешевший за GCP IoT Core як для MQTT, так і для HTTP протоколів, а MQTT місток в AWS на 45 % дешевший, ніж HTTP місток в AWS для сценаріїв частотного передавання даних (<5 хв).

Таблиця 12

Грошові витрати для трафіку AWS HTTP моста за під'єднання 8 годин на день

Кількість пристроїв	Частота відправлення даних з пристрою				
	1 хвилина	5 хвилин	10 хвилин	15 хвилин	20 хвилин
10К	\$292	\$58,4	\$29,2	\$19,47	\$14,6
100К	\$2 536	\$584	\$292	\$194,7	\$146
1М	\$21 140	\$4 788	\$2 536	\$1 757	\$1 368
10М	\$205 100	\$41 580	\$21 140	\$14 329	\$10 920
100М	\$2 044 700	\$409 500	\$205 100	\$136 990	\$102 900

6. Дерево вибору протоколів передавання даних та хмарних постачальників

Проаналізувавши грошові витрати для набору сценаріїв, що представляють найпоширеніші шаблони для передавання даних у галузі Інтернету речей, ми побудували дерево вибору оптимальних протоколу передавання даних та постачальника хмарних послуг для доставки телеметрії залежно від конкретних факторів, а саме від кількості пристроїв, що передають дані, та від частоти, з якою передаються ці дані (табл. 13).

Таблиця 13

Дерево вибору протоколів передавання даних та хмарних постачальників

Кількість пристроїв	Тривалість передавання даних		Частота відправлення повідомлень		Рішення
	24 години на день	8 годин на день	<10 хвилин	>10 хвилин	
<10К	+	-	+	-	AWS MQTT
<10К	+	-	-	+	AWS HTTP
<10К	-	+	+	-	AWS MQTT
<10К	-	+	-	+	AWS HTTP
<100К	+	-	+	-	AWS MQTT
<100К	+	-	-	+	AWS HTTP
<100К	-	+	+	-	AWS MQTT
<100К	-	+	-	+	AWS HTTP
<1М	+	-	+	-	AWS MQTT
<1М	+	-	-	+	AWS HTTP
<1М	-	+	+	-	AWS MQTT
<1М	-	+	-	+	AWS HTTP
<10М	+	-	+	-	AWS MQTT
<10М	+	-	-	+	AWS HTTP
<10М	-	+	+	-	AWS MQTT
<10М	-	+	-	+	AWS HTTP
<100М	+	-	+	-	Custom MQTT broker або CoAP
<100М	+	-	-	+	AWS HTTP
<100М	-	+	+	-	Custom MQTT broker або CoAP
<100М	-	+	-	+	AWS HTTP
>100М	+	-	+	-	Custom MQTT broker або CoAP
>100М	+	-	-	+	AWS HTTP
>100М	-	+	+	-	Custom MQTT broker або CoAP
>100М	-	+	-	+	AWS HTTP

Висновки

Вибір протоколу залежить від вимог бізнесу, від частоти доставки телеметрії та кількості підключених пристроїв. У роботі наведено порівняння вартості використання протоколів MQTT та HTTP, що надають постачальники хмарних послуг AWS та GCP. Як зазначено у статті, GCP IoT Core є дорожчим порівняно з AWS IoT Core для всіх оцінених сценаріїв, тому не рекомендований для використання. Майже для усіх випадків частого передавання даних MQTT-міст, що надається AWS IoT Core, є найкращим рішенням. Збільшення кількості підключених пристроїв (>10 млн) з високою частотою доставки даних (приблизно кожену 1 хв) змушує розглянути використання самостійного MQTT-брокера або іншого протоколу на основі TCP, наприклад, CoAP [11]. Якщо передавання даних відбуватиметься рідше, ніж кожні 10 хв, HTTP-міст може бути рішенням для кількості до 100 млн пристроїв. Згідно зі звітом про випадки використання IoT 2021 р., MQTT є хорошим вибором для віддаленого моніторингу активів, управління автопарком, відстеження місця розташування та внутрішньозаводських випадків використання. HTTP – доцільний вибір для використання IoT на основі автоматизації процесів та передбачувального обслуговування. Результатом цього дослідження є дерево рішень, яке надає рекомендації стосовно вибору протоколу та постачальника хмарних послуг залежно від кількості пристроїв, що надсилають дані, та від частоти надсилання даних.

Список літератури

1. P. Pierleoni, R. Concetti, A. Belli and L. Palma, "Amazon, Google and Microsoft Solutions for IoT: Architectures and a Performance Comparison", in *IEEE Access*, Vol. 8, 5455–5470, 2020. DOI: 10.1109/ACCESS.2019.2961511.
2. B. Wukkadada, K. Wankhede, R. Nambiar and A. Nair, "Comparison with HTTP and MQTT In Internet of Things (IoT)", 2018 *International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, 2018, 249–253. DOI: 10.1109/ICIRCA.2018.8597401.
3. Yokotani, Tetsuya & Sasaki, Yuya. (2016). *Comparison with HTTP and MQTT on required network resources for IoT*, 1–6. DOI: 10.1109/ICCEREC.2016.7814989.
- 4 Ahmed, Muneeb & Akhtar, Mohd. (2021). *Smart Home: Application using HTTP and MQTT as Communication Protocols*. DOI: 10.48550/arXiv.2112.10339
5. Canek, Rodrigo & Borges, Pedro & Taconet, Chantal (2022). *Analysis of the Impact of Interaction Patterns and IoT Protocols on Energy Consumption of IoT Consumer Applications*. DOI: 10.1007/978-3-031-16092-9_9.
6. S. Misra, A. Mukherjee, A. Roy. "Introduction to IoT". Cambridge: Cambridge University Press, 2021, 1–424. DOI: 10.1017/9781108913560
7. N.M. Shaikh, Y. Ingle. "Application of Restful APIs in IOT": A Review. Haryana: iJRASET, 2021, 145–151. DOI: 10.22214/ijraset.2021.33013
8. Atmoko, Rachmad & Riantini, Rona & Hasin, M. (2017). *IoT real time data acquisition using MQTT protocol*. *Journal of Physics: Conference Series*, 853, 1–7. DOI: 10.1088/1742-6596/853/1/012003.
9. M. Amadeo, C. Campolo, A. Iera and A. Molinaro, "Information Centric Networking in IoT scenarios: The case of a smart home", 2015 *IEEE International Conference on Communications (ICC)*, London, UK, 2015, 648–653. DOI: 10.1109/ICC.2015.7248395.
10. E. Longo, A. E. C. Redondi, M. Cesana, A. Arcia-Moret and P. Manzoni, "MQTT-ST: a Spanning Tree Protocol for Distributed MQTT Brokers", *ICC 2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, 2020, 1–6. DOI: 10.1109/ICC40277.2020.9149046.
11. labbas Alhaj, "Constraint application protocol (CoAP) for the IoT". *Frankf. Univ. Appl. Sci.*, 2018, 1–5. DOI: 10.13140/RG.2.2.33265.17766

**SELECTION OF PROTOCOLS FOR DATA TRANSMISSION
FROM INTERNET OF THINGS DEVICES TO CLOUD PROVIDERS**

A. Shykhmat, Z. Veres

Lviv Polytechnic National University,
Department of Computerized Automatic Systems

© Shykhmat A., Veres Z., 2023

The Internet of Things (IoT) enables the creation of networks between devices, people, applications, and the Internet, creating new ecosystems with higher productivity, better energy efficiency, and higher profitability. Nodes in these networks must have the ability to communicate and exchange data, which requires the use of data transfer protocols. However, choosing the right protocol for a specific use case is not always straightforward. This paper provides an overview of two existing data transfer protocols, MQTT and HTTP, compares the amount of tariffed traffic generated by each protocol, and evaluates the efficiency of protocol costs. The research showed that in comparison to AWS IoT Core, GCP IoT Core is more expensive for all evaluated scenarios and is not recommended for use. For scenarios with frequent data transfer, the best solution is to use the MQTT bridge provided by AWS IoT Core. If the number of connected devices exceeds 10 million with a high frequency of data transmission every 1 minute, it is advisable to consider using a standalone MQTT broker or another TCP-based protocol such as CoAP. In the case of less frequent data transmission (every 10 minutes or less), an HTTP bridge may be a solution for up to 100 million devices. As a result of the research, a decision tree was created to select the best protocol for specific use cases.

Key words: IoT; data transfer protocols; HTTP; MQTT; AWS; GCP; IoT Core.