УДК 004.45+004.89 DOI: 10.31673/2412-4338.2022.022332

Каплунов А. В., Гайдай А. Р., Гер В. М., Нікольський С. С.

Національний технічний університет України «Київський політехнічний університет імені Ігоря Сікорського»

ЗАСОБИ МОНІТОРИНГУ МЕРЕЖІ В IOT ІНФРАСТРУКТУРІ З ГІБРИДНОЮ АРХІТЕКТУРОЮ

Розглянуто процеси обміну даними в комп'ютерних мережах, які породжують проблему продуктивності та критичного стану мережевого оточення в складних IoT системах з розподіленою інфраструктурою. Запропоновано засоби моніторингу інфраструктури комп'ютерних мереж в системах IoT в реальному часі. Розроблено архітектуру системи моніторингу мережного оточення для реалізації концепції гібридної IoT інфраструктури, що побудована на базі технологій граничних обчислень, та методика інтеграції сучасних інструментів моніторингу мережевої інфраструктури. Розроблені засоби забезпечують збирання, аналіз та планування компонентів продуктивності IoT пристроїв та їх мережного оточення, дозволяють моніторити та запобігати критичному стану мережевої IoT інфраструктури в гібридних архітектурах IoT систем. Запропонований спосіб аналізу продуктивності комп'ютерної мережі на основі зваженого показника продуктивності, дозволив підвищити ефективність моніторингу комп'ютерної мережі в локальному домені мережевої IoT інфраструктури за рахунок балансування навантаженням мережі. Розроблені засоби реалізовані на границі IoT та дозволяють підвищити в цілому ефективність моніторингу мережі на локальному рівні та мають можливість інтеграції в хмарні технології та сервіси.

Ключові слова: *система моніторингу, балансування навантаженням, IoT, Edge Computing, MQTT.*

Kaplunov A.V., Haidai A.R., Ger V.M., Nikolskyi S.S.

National Technical University of Ukraine "Ihor Sikorsky Kyiv Polytechnic University"

NETWORK MONITORING TOOLS IN IOT INFRASTRUCTURE WITH HYBRID ARCHITECTURE

A.

Text of annotation translation. – The processes of data exchange in computer networks have been examined within the context of generating productivity issues and critical network state in complex IoT systems with distributed infrastructure. Monitoring tools for computer network infrastructure in real-time were proposed for IoT systems. The architecture of a network environment monitoring system was developed to implement the concept of hybrid IoT infrastructure which was built with edge computing technologies. Also was proposed the methodology for integrating modern network infrastructure monitoring tools. The developed tools facilitate the collection, analysis and planning of performance components of IoT devices and their network environments. It enables the monitoring and prevention of critical states within the network IoT infrastructure in hybrid IoT system architectures. The proposed method for analysing computer network performance based on a weighted performance metric contributes to enhancing the efficiency of computer network monitoring within the local domain of network IoT infrastructure by load balancing. The developed tools was

realized at the IoT edge, enhancing overall network monitoring efficiency at the local level and possessing the potential for integration with cloud technologies and services.

Keywords: *monitoring system, load balancing, IoT, Edge Computing, MQTT.*

1. Вступ

В рамках парадигми IoT різноманітним IT-компаніям, а також компаніям, що спеціалізуються на зборі даних з різних віддалених пристроїв, підключених до їхніх систем від різних постачальників потрібні системи моніторингу мережного оточення. На такі системи, в першу чергу, покладаються завдання моніторингу стану віддалених серверів та IoT пристроїв шляхом збирання різноманітних метрик та відстеження статусу їх роботи та стану мережної IoT інфраструктури в цілому [1].

В контексті широкого розповсюдження хмарних технологій в інфраструктурі IoT системи моніторингу вирішують завдання інтеграції підприємства з хмарними сервісами. Сучасні хмарні технології пропонують потужні сервіси обробки та аналітики даних [2]. Сучасна технологія хмара речей (CoT, Cloud of Things) забезпечує ефективну інтеграцію IoT із хмарними обчисленнями відома, для специфікації завдань обробки даних та зниження залежності між хмарними сервісами та локальною інфраструктурою IoT [3]. Це безперечно приводить до високої актуальності хмарних сервісі серед компаній, що зацікавлені в тому, щоб контролювати та мати можливість надавати абонентську підтримку користувачам.

Хмарні сервіси пропонують й широке коло систем моніторингу, але як показують сучасні дослідження [4, 5], різноманітність нижніх рівнів IoT інфраструктури, обмежені ресурси та специфікації IoT пристроїв часто робить неефективним як використання уніфікованих систем моніторингу так і традиційних хмарних технологій з централізованою архітектурою загалом. Проблема використання хмарних сервісів широкого призначення обумовлена збитковістю та жорсткою монолітною прив’язкою IoT платформ до універсальних рішень, що в цілому породжує технічні та економічні проблеми для ефективної реалізації інфраструктури IoT.

Це приводить до того, що, для більшості компаній, що базуються на використанні IoT інфраструктури, інтеграція з хмарними технологіями не буде абсолютною. Основний концептуальний підхід, який на сьогодні є обґрунтованим в парадигмі IoT це розташування частини інфраструктури на місці, тобто на границі мережі IoT (EC, Edge Computing). Зважаючи на широке використання хмарних технологій, на сьогодні найбільш актуальний гібридний підхід за рахунок інтеграції хмарних та граничних технологій [3].

В рамках реалізації такої гібридної концепції актуальною та доцільною стає задача розробки засобів моніторингу мережевої IoT інфраструктури, яка є синергією хмарного та локального IoT середовища. Постають задачі аналізу продуктивності комп’ютерної мережі та інших показників мережної інфраструктури в системах IoT, та їх аналізу для планування забезпечення параметрів мережевого потоку в режимі реального часу.

2. Аналіз літературних даних та постановка проблеми

Обґрунтування широкого застосування хмарних технологій в парадигмі IoT розглянуто в роботі [2], автори якої узагальнюють такі позитивні риси, як віддалена обчислювальна потужність; підтримка потужних API-інтерфейсів; низька залежність від локальної інфраструктури IoT; централізована безпека й конфіденційність даних; ефективні протоколи автентифікації й шифрування; відсутність вхідного бар'єру для хостинг-провайдерів. Це надає компаніям, що впроваджують системи IoT, можливість ефективного контролю та абонентської підтримки користувачів. Натомість, традиційні хмарні технології характеризуються централізованою та уніфікованою архітектурою, що породжує технічні та економічні проблеми для ефективної реалізації парадигми IoT. В роботі [4, 5] фактори хмарних обчислень оцінені як такі, що мають високий час очікування; високі затримки; віддаленість серверів та відстань між клієнтом та сервером; невизначений рівень безпеки; відсутність сповіщень; обробка на розподілених серверах; часткове технічне обслуговування. Означені фактори негативно впливають на ефективність застосування хмарних технологій та сервісів в

парадигмі IoT. Іншою значною проблемою є необхідність врахування обмежених ресурсів та специфікації IoT пристроїв. В роботі [5] на прикладі архітектури IoT, коли всі дані збираються та аналізуються в хмарі, підсумовано, що масштабні системи IoT з великою кількістю датчиків та значним обсягом даних характеризуються надмірним мережним трафіком та зниженням рівня обслуговування. Комунікаційна конвергенція викликає затримки керування в системі, що критично для чутливих до часу виконання IoT застосунків. Автори робіт [4 – 6] також зазначають високу уніфікацію хмарних рішень, що обумовлює проблеми застосування IoT пристроїв з неоднорідними платформами та специфічних підходів до аналітики й прийняття рішень.

Для підвищення продуктивності застосунків, обмеження витрат та зниження енергоспоживання останнім часом стає актуальним гібридний підхід, коли частина інфраструктури IoT розташовується як можна ближче до джерела даних, тобто на границі мережі IoT і базується на технологіях граничних обчислень (EC, Edge Computing) [7, 8]. Така гібридна концепція передбачає певну синергію хмарного і локального середовища в інфраструктурі IoT. В роботі [9, 3] описані характерні гібридні підходи в IoT, коли розумні програмовані автоматичні засоби на границі локальної мережі реалізують збирання та аналіз даних і негайно надсилають відповіді на пристрої користувачів-відправників. Автоматичні засоби на границі мережі приймають рішення, які дані потрібно направити в хмару, а які необхідно обробляти локально. Характерний приклад гібридної IoT архітектури з системою граничного обчислення (IoT-EC) описаний в роботі [10], де функція збору даних, фільтрації, контролю зворотного зв’язку реалізовані на периферійних серверах базових станцій.

В роботі [3] зазначено актуальність гібридної архітектури IoT для вирішення проблем продуктивності і масштабованості інтелектуальних систем та локальних застосунків. Перенесення функцій збору, обробки та аналізу даних на локальний рівень дозволить уникнути затримки та зменшити витрати на передавання трафіку в глобальній мережі, підвищити стійкість і живучість масштабованих інтелектуальних систем за рахунок розподілення обчислювальних функцій по мережі та усування точок збою, використовувати менш складні та дорогі пристрої IoT за рахунок можливості перенесення процесора та ємність пам’яті на локальні шлюзи та сервери. Хмарні технології та сервіси задіяні для реалізації корпоративних центрів обробки даних.

Для розроблення та підтримки інфраструктури IoT актуальною постає задача моніторингу мережного оточення [1]. Ця задача набуває особливої актуальності в гібридних архітектурах IoT. На такі системи моніторингу покладаються завдання моніторингу як стану локальних і віддалених серверів та пристроїв IoT та стану мережної IoT інфраструктури в цілому.

Хмарні сервіси пропонують для розробників IoT систем широке коло систем моніторингу [11], але як показують сучасні дослідження децентралізована обробка даних та реалізація частини послуг на локальному рівні IoT приводить до ускладнення або унеможливлення використання систем моніторингу широкого застосування , які реалізовані на рівні хмарних сервісів [4, 5]. Окрім цього відомі підходи до моніторингу продуктивності мережі часто орієнтовані на вимірювання таких параметрів, як навантаження на процесори, пропускну здатність, затримки, втрату пакетів, тощо і не пропонують засобів балансування навантаження [12, 13], які актуальні для розподіленої архітектури мікросервісів IoT інфраструктури. В цьому контексті виникає актуальна задача , яка обумовлює актуальність та доцільність подальших досліджень і розробок в області моніторингу мережного оточення, аналізу та планування потоків даних в гібридній інфраструктурі IoT. Цим дослідженням присвячена дана стаття.

3. Мета і задачі дослідження

Метою дослідження є розроблення засобів для моніторингу мережної інфраструктури IoT систем з гібридною архітектурою на базі технології граничних обчислень.

Реалізація мети дослідження спрямована на підвищення ефективності моніторингу мережного оточення на локальному рівні інфраструктури IoT, де виникають задачі моніторингу продуктивності комп’ютерної мережі, аналізу та планування мережного потоку

даних, збирання та аналізу показників інфраструктури в цілому, запобігати критичному стану мережевої інфраструктури в складних IoT системах. Ціллю є реалізація підходів до моніторингу комп'ютерної мережі на локальному рівні інфраструктури IoT в рамках технології граничних обчислень з можливістю інтеграції в хмарні технології та сервіси. Реалізація мети та цілей дослідження дасть можливість вдосконалити архітектуру моніторингових систем на базі технологій граничних обчислень, та підвищити ефективність моніторингу мережного оточення, в першу чергу для застосунків чутливих до часу виконання.

Для досягнення мети були поставлені наступні задачі:

– розробити архітектуру системи моніторингу мережного оточення для реалізації концепції гібридної IoT інфраструктури, що побудована на базі технологій граничних обчислень;

– для реалізації архітектури системи моніторингу розробити методику інтеграції сучасних інструментів моніторингу мережевої інфраструктури, для забезпечення збору, аналізу та планування компонентів продуктивності комп'ютерної мережі і запобігання критичному стану мережевої IoT інфраструктурі в режимі реального часу;

– розробити спосіб аналізу продуктивності комп'ютерної мережі на основі зваженого показника продуктивності, з ціллю підвищення ефективності моніторингу комп'ютерної мережі в локальному домені мережевої IoT інфраструктури.

4. Матеріали та методика інтеграції інструментів моніторингу для реалізації досліджень IoT інфраструктури з гібридною архітектурою

4.1. Програмне забезпечення та засоби мережної комунікації

Серверне програмне забезпечення системи моніторингу мережного оточення для IoT інфраструктури реалізовано на ПК з операційною системою Linux. Комплекс програмного забезпечення розроблений з використанням мови програмування Python.

Для потокової передачі даних між пристроями в реальному часі використаний протокол MQTT. Цей протокол актуальний для мережних рішень в IoT інфраструктурі, яка характеризується низькою пропускною здатністю, непередбачуваною стабільністю, наявністю пристроїв з обмеженою потужністю процесора [12,13]. Розроблення засобів моніторингу мережі з великою кількістю пристроїв базується на здатності протоколу MQTT реалізувати канали або топіки, кожен з яких обробляється як шлях до файлу. MQTT топіки дозволяють обробляти окремо кожну метрику мережі та розділяти її між різними пристроями в мережі. Розроблене програмне забезпечення виконує перехоплення пакетів та пересилання метрик на MQTT- сервер. В якості MQTT-клієнта для роботи топіками використано програмне забезпечення telegraf – це агент з відкритим кодом. Telegraf використовується для збору метрик чи даних із системи моніторингу відповідно до визначених топіків. Telegraf збирає і передає зібрані метрики продуктивності мережі та застосунків до бази даних часових рядів. Бази даних часових рядів, оптимізовані для швидкого зберігання даних з високою ступеню готовності та пошуку часових рядів даних, зокрема у таких галузях, як моніторинг операцій та метрик.

4.2. Засоби для візуалізації та аналізу результатів моніторингу мережного оточення

Для візуалізації, аналізу та моніторингу даних використовується програма з відкритим кодом Grafana [14]. Система Grafana використовується, як функціональна частина системи моніторингу мережі та надає можливість створювати інформаційні панелі та графіки для відображення мережевих та інших показників в реальному часі. Система має відкритий код, що дозволило розробити спеціалізовані інформаційні панелі для відображення статистики, результатів моніторингу та підтримки візуальної інтеграції функціональних частин системи моніторингу. Для перегляду інформаційних параметрів пристроїв мережної інфраструктури та перемикання між ними розроблено комплекс дашбордів. Розроблені правила для реалізації оповіщень електронною поштою на базі вбудованого механізму оповіщень Grafana.

Для відстежування кодів помилок, отриманих від сервера, та дослідження поведінки системи в систему моніторингу інтегровано розподілену систему пошуку і аналітики Elasticsearch [15]. Для відображення подій у дашбордах розроблено інструкції, анотації до яких налаштовано як запити до пошукового рушія Elasticsearch. Результати виконання розроблених інструкцій відображаються на дашбордах вертикальними червоними лініями, під час наведення курсору на які відображується опис події.

Для реалізації розроблюваних засобів моніторингу, система Grafana використовується для відлагодження та моніторингу функціональних складових системи та налагодження застосунків. Реалізовано ідентифікація помилок на стороні клієнта, сервера або помилок у логіці системних повідомлень. Всі веб-запити, ініційовані клієнтом відстежуються як адміністраторами застосунків так і самими застосунками в автоматичному режимі в реальному часі, що дозволяє визначити причину помилок. Налагодження та виправлення помилок також здійснюється при наявності певних аномалій на графіках результатів моніторингу.

В роботі [14] описані характерні налаштування інформаційних панелей для налагодження системи моніторингу. Інформаційна панель, яка зображує час відповіді на веб-запити протягом певного періоду часу надає можливість відстеження максимального, мінімального та середнього часу відгуку. Якщо виявляється запит, на обробку якого витрачено багато часу, частина графіка масштабується, що дозволяє детально вивчити проблему. Інформаційні панель для відстеження трафіку дозволяє відстежувати завантаження системи протягом певного періоду часу і відслідковувати непередбачувані сплески активності. Такі сплески можуть допомогти виявити, наприклад, використання сканерів Google, які індексують контент веб-сайту, або шкідливих ботів, що сканують систему на наявність уразливостей.

Для реалізації структурованого пошуку записів в системних журналах до розробленого програмного забезпечення інтегровано інструмент Graylog, який використовує Elasticsearch [15]. Цей засіб використовуються для низькорівневого аналізу проблем та аномалій, які виявлені на інструментальних панелях Grafana.

4.3. Засоби для забезпечення функцій моніторингу та аналітики даних для IoT інфраструктури з гібридною архітектурою

Для рішення задач моніторингу та аналітики даних використані технології для збирання та обробки даних часових рядів. Часові ряди це послідовність точок даних, проіндексованих у часі. Прикладами часових рядів є саме дані з датчиків, показники продуктивності серверів, тощо. Інструменти та методи спостереження, аналізу та прогнозування часових рядів лежать в основі технологій та систем сучасної аналітики даних. Моделі даних організовані в часові ряди є вихідними для програм та сервісів інтелектуальної аналітики даних, зокрема й в хмарних середовищах.

Технологія Prometheus з відкритим вихідним кодом, є фактично базою даних для роботи з часовими рядами, використовується для забезпечення функцій моніторингу та оповіщення для хмарних середовищ. Ця система забезпечує функціональну частину інтеграції до хмарних сервісів в системі моніторингу для гетерогенної системи. Задача Prometheus складається в автоматичному збиранні показників – метрик продуктивності мережі, у вигляді даних часових рядів, записуючи інформацію з міткою часу. Особливості технології Prometheus, а саме здатність самостійно збирати метрики із системи моніторингу обґрунтовує доцільність використання цієї технології для реалізації IoT інфраструктури пов’язаної з використанням хмарного середовища. Ця системи є частиною загальнодоступних хмарних сервісів, тому розвертання цієї технології на локальному рівні інфраструктури IoT не розглядається.

Для забезпечення функцій функцій моніторингу та оповіщення для локального IoT середовища використана технологія Influx DB. Influx DB — база даних часових рядів з відкритим кодом, яка оптимізована для отримання даних із високою доступністю, швидшого зберігання даних часових рядів, зокрема для моніторинг операцій, показників застосунків, даних IoT датчиків та аналітика в реальному часі. Використання InfluxDB для системи моніторингу забезпечує високу продуктивність та функціональність оброблення та аналізу

даних в контексті вирішення задач в реальному часі на границі IoT інфраструктури. Технологія запити через уніфікований API та інтеграцію інтерфейсу для візуалізації даних Grafana.

На відміну від технології Prometheus, який автоматично сканує всі цільові об'єкти для отримання метрик продуктивності мережі, Influx DB працює з метриками параметрів мережі, які йому безпосередньо надсилає системне програмне забезпечення. Перевагою цього в першу чергу є здатність працювати з подіями, на основі яких реалізований функціонал системи моніторингу мережі на границі IoT інфраструктури.

Централізований і автоматизований характер технології Prometheus не заснований на подіях. Він орієнтований на використання в централізованих та уніфікованих хмарних сервісах, зберігає заздалегідь агреговані метрики для певного сервісу і не фіксує збір даних з прив'язкою до певних подій.

5. Розроблення засобів для моніторингу мережної інфраструктури IoT систем з гібридною архітектурою

5.1. Архітектура системи моніторингу мережної IoT інфраструктури та методика інтеграції інструментів моніторингу

Архітектура запропонованої системи моніторингу складається з комплексу інструментів для моніторингу, які інтегровані в єдину систему для рішення задачі моніторингу продуктивності комп'ютерної мережі, аналізу та планування мережного потоку даних, збирання та аналізу показників інфраструктури в цілому, запобігання критичного стану мережевої IoT інфраструктури.

До складу системи моніторингу входять комплекс IoT пристроїв, сервер MQTT, сервер Telegraf, сервер InfluxDB, сервер Grafana (рис. 1).

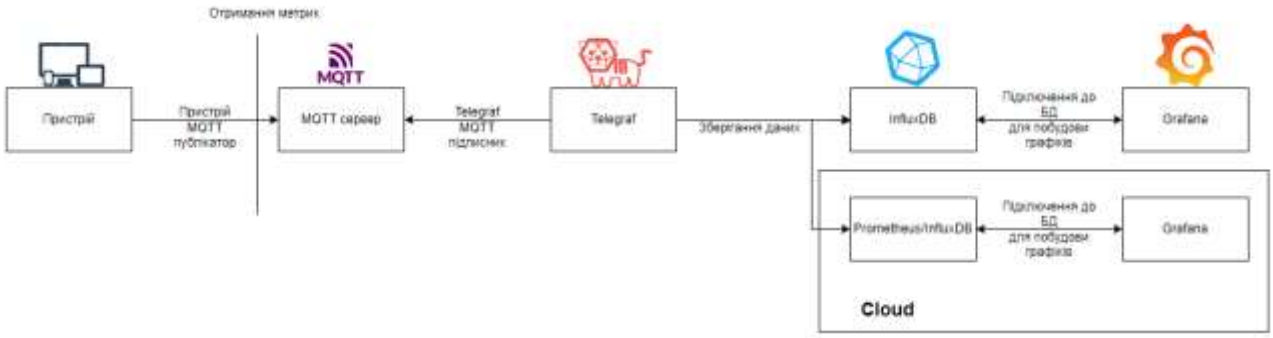

Рис. 1. Схема архітектури системи моніторингу мережі

Важливу функцію виконує є протокол MQTT, який працює за традиційною технологією клієнт-сервер. Пристрої-клієнти є публікаторами, які генерують інформацію, позначають її певною темою і надають пристрою-підписнику інформацію за певною темою. Загальний принцип функціонування можна розглянути на наступному прикладі. Датчик температури (публікатор) передає показники з темою «Температура», а пристрій який отримує інформацію про температуру в офіс підписаний на тему «Температура» отримує дані від усіх датчиків температури. Сервера MQTT – пристрої брокери, отримують дані від публікаторів, зберігають їх та передають підписникам.

Для перехоплення пакетів за допомогою спеціальних бібліотек з подальшою пересилкою метрик на сервер MQTT розроблено спеціальне програмне забезпечення. Для кожної метрики створена окрема тема MQTT, яка додатково розділюється для різних пристроїв. Крім того, клієнти MQTT використовуються для підписки на ці теми. Підписка на тему реалізується через сервер Telegraf.

Telegraf автоматично збирає і передає зібрані метрики продуктивності мережі та застосунків до бази даних InfluxDB. Для налаштування InfluxDB нам потрібно отримати токен доступу який знаходиться в вкладці Load Data -> API Tokens -> admin's Token

Для візуалізації результатів моніторингу та аналітики розроблено комплекс інструментальних панелей та дашбордів в системі візуалізації даних Grafana, які інтегровані в базою даних InfluxDB.

Розроблена архітектура спрямована на моніторинг комп'ютерної мережі на локальному рівні інфраструктури IoT в рамках технології граничних обчислень. В контексті інтеграції в хмарні технології та сервіси, як вже було зазначено в п. 4.3, сервер збирає дані для передачі в базу даних Prometheus, яка налаштовується за методологією хмарних сервісів, або необхідно розгорнути InfluxDB і Grafana віддалено на хмарному сервері.

5.2. Спосіб аналізу продуктивності комп'ютерної мережі на основі зваженого показника продуктивності

Система моніторингу обробляє наступні метрики продуктивності мережі: затримки в мережі або відставання – час який потрібен пакету даних для проходження через мережу та декодування; пропускна здатність мережі – максимальна швидкість передачі даних у мережі в певний час; кількість втрачених пакетів даних; завантаженість процесора.

В контексті реалізації підходу балансування навантаженням для підвищення ефективності моніторингу показників продуктивності в розподіленій системі запропоновано надавати вагу кожному вимірюваному компоненту шляхом використання коефіцієнтів для кожного показника продуктивності. Використовуючи такі коефіцієнти формується зважений показник продуктивності мережі. Використовується підхід описаний в роботі [16] – відомий протокол визначення найкращого маршруту, з точки зору забезпечення якості обслуговування QoS, базується на визначенні композитарної метрики, яка розраховується як сума показників продуктивності з певним ваговим коефіцієнтом. Можливість використання вагових коефіцієнтів дозволяє адаптувати алгоритм балансування навантаженням до задач моніторингу продуктивності мережі за зваженим показником. Це надає можливість певного балансування навантаженням у випадку, якщо окремий показник навантаження визначений як високий, а вплив його на систему мінімальний.

Розглянемо таке балансування на наступному прикладі моніторингу мережі. На рис. 2 зображено інформаційна панель з результатами моніторингу завантаження процесора, на якій видно високе завантаження центрального процесора (ЦП) на певному пристрої мережі. Графік на інформаційній панелі (рис. 2) не надає комплексної інформації окрім завантаження процесора в певний період часу. Одначе часто сильне завантаження процесору може мати мінімальний вплив на продуктивність системи в цілому або певного компонента системи.

Для комплексної оцінки продуктивності мережі та балансування навантаженням запропоновано ввести зважений показник продуктивності мережі. В експериментальній моделі системи обрано наступні коефіцієнти впливу параметрів продуктивності на загальну оцінку продуктивності мережі: пропускна здатність мережі (0,4), затримки в мережі (0,2), кількість втрачених пакетів даних (0,2), завантаженість процесора (0,4). Зважений показник дозволяє врахувати вплив кожного показника продуктивності відповідно його впливу на загальну продуктивність мережі відповідно поставленим задачам. Зважений показник дозволяє оцінити продуктивність мережі за десятибальною шкалою.

Рис. 2. Результати моніторингу показника завантаження процесора (мГц)

Інформаційна панель з результатами моніторингу з врахуванням зваженого показника зображена на рис. 3. Введення вагових коефіцієнтів для показників продуктивності надає комплексну інформацію і показує, що система працює коректно. На рис. 3 видно, що продуктивність система коливається в межах від семи до дев'яти балів за десятибальною шкалою. Десять балів означає що система має максимальну продуктивність, зменшення значень показує, що система має певні проблеми зі зниженням продуктивності, які потребують подальшого аналізу з використанням детальних інформаційних панелей моніторингової системи і подальшому низькорівневому дослідженню.

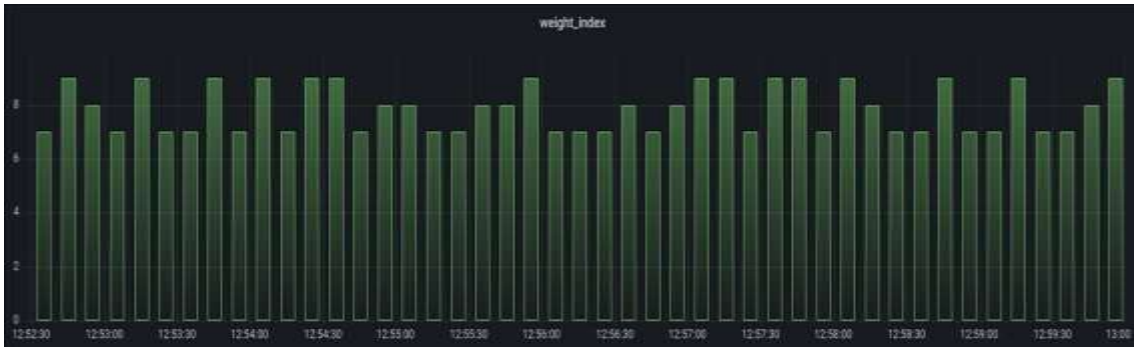

Рис. 3. Результати моніторингу за зваженим показником продуктивності

6. Обговорення результатів дослідження засобів моніторингу мережної IoT інфраструктури

Дашборд панель для демонстрації моніторингу продуктивності для одного пристрою мережі зображена на рис. 4. На цій панелі відображено графіки окремих показників, таких як пропускна здатність, завантаження ЦП, затримка та втрата пакетів при передачі. Пропускна здатність вимірюється в кількості пакетів в секунду, завантаження ЦП в МГц, затримки – в секундах, кількість втрачених пакетів вимірюються за їх кількістю в секунду.

Рис. 4. Дашборд панель для одного пристрою мережі

Графіки для кожного окремого показника да дашборд панелі дають змогу деталізувати критичні місця під час аналізу кожного окремого показника продуктивності мережі і дослідити їх засобами низькорівневого аналізу описаними в п. 4.2.

Праворуч на дашборд панелі (рис. 4) зображено максимальні числові показники деяких параметрів пристрою за певний час вимірювання, такі як максимальна кількість втрачених пакетів, максимальне завантаження ЦП. Ці показники використовуються для відображення загальної статистики моніторингу:

– перший графік зображує поточне навантаження на процесор в МГц та діаграму значення завантаження у відсотках від максимального значення (максимальне значення на пристрої встановлене 3100 МГц);

– інші вимірювальні діаграми показують максимальні значення кожного показника за певний період часу, в тих же одиницях виміру що й відповідні їм графіки.

В нижній частині дашборд панелі розміщені графіки комплексної оцінки продуктивності мережі. Так на рис. 5 показано графік продуктивності виміряної за зваженим індексом, де продуктивність системи знаходиться в межах від нуля до десяти. Значення на графіку формуються з врахуванням вагових коефіцієнтів впливу кожного показника на продуктивність пристрою та його мережного оточення. В цілому, при збільшенні завантаження ЦП показники на графіку будуть знижуватися, а при звільненні ресурсів - зростати. Наприклад, на графіку (рис. 5) комплексні значення продуктивності не знижуються нижче 7 балів, що викликано значним навантаженням на один або два вимірювані параметри або незначним навантаженням на всі вимірювані параметри. Проте це не становить проблеми для роботи даного пристрою.

Зниження зваженого показника нижче шести балів розглядається як значний спад продуктивності, при подальшому зниженні якої система може видавати некоректні дані. В цьому випадку необхідна деталізація проблеми з використанням інформаційних панелей по кожному показнику.

Останній графік (рис. 6) відображає порівняльну продуктивність декількох пристроїв та їх мережевого оточення за зваженим індексом. Що в свою чергу дає можливість виконати порівняльний аналіз продуктивності і обрати більш ефективний пристрій в контексті виконуваної задачі.

Рис. 5. Графік продуктивності мережного оточення пристрою за зваженим показником продуктивності

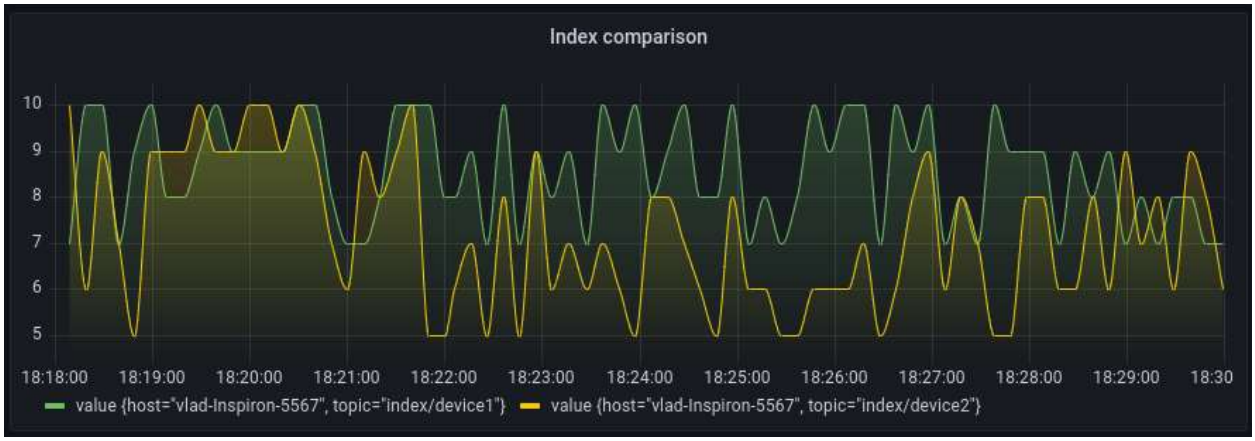

Рис. 6. Порівняльний графік продуктивності мережевого оточення декількох пристроїв за зваженим індексом

7. Висновки

Розроблено засоби для підвищення ефективності моніторингу мережевого оточення на локальному рівні інфраструктури IoT, де виникають задачі моніторингу продуктивності комп'ютерної мережі, аналізу та планування мережного потоку даних, збирання та аналізу показників інфраструктури в цілому, запобігати критичному стану мережевої інфраструктури в складних IoT системах. Реалізовано підходи до моніторингу комп'ютерної мережі на локальному рівні інфраструктури IoT в рамках технології граничних обчислень з можливістю інтеграції в хмарні технології та сервіси.

Розроблено архітектуру системи моніторингу мережевого оточення, яка реалізує концепцію гібридної IoT інфраструктури на базі технологій граничних обчислень. Розроблена архітектура забезпечила можливість вдосконалення широкодоступних хмарних сервісів IoT систем за рахунок використання технологій граничних обчислень, що дозволило підвищити ефективність моніторингу мережного оточення для застосунків чутливих до часу виконання.

Розроблено методику інтеграції сучасних інструментів моніторингу мережевої інфраструктури, яка забезпечує можливість збирання, аналізу та планування компонентів продуктивності комп'ютерної мережі на локальному рівні IoT інфраструктури та має можливість інтеграції в хмарні сервіси;

Розроблено спосіб аналізу продуктивності комп'ютерної мережі на основі зваженого показника продуктивності, використання якого дозволило підвищити ефективність моніторингу комп'ютерної мережі в локальному домені мережевої IoT інфраструктури за рахунок балансування навантаженням комп'ютерної мережі.

Запропонована архітектура системи моніторингу, а також використання зваженого показника продуктивності комп'ютерної мережі з урахуванням вагових коефіцієнтів параметрів комп'ютерної мережі дозволяє підвищити ефективність аналізу продуктивності комп'ютерної мережі і мережеву інфраструктуру в цілому.

Список використаної літератури

1. Gupta A. Network Monitoring: Network Monitoring Basics. [Електронний ресурс] https://www.motadata.com/blog/network-monitoring-basics/

2. Mishra S. Cloud of Things (CoT): Security, Privacy & Adoption / S. Mishra // International Journal of Security and Its Applications. – Vol. 14, No. 3 – IJSIA Copyright, 2020. – pp.1-14. http://doi.org/10.33832/ijsia.2020.14.3.01

3. Klymenko I., Gaidai A., Nikolskyi S., Tkachenko V. The Architectural Concept Of The Monitoring System On The Basis On A Neuron Module IoT Data Analytics. - Vol. 2, No. 41 - 2022. – pp. 111-123. https://doi.org/10.20535/1560-8956.41.2022.271355

4. Ramachandra G., Iftikhar M., Khan F.A. A Comprehensive Survey on Security in Cloud Computing / G. Ramachandra, M. Iftikhar, F.A. Khan // Procedia Computer Science. – Vol 3, Issue 11. – 2017. – pp. 465–472. https://doi.org/10.1016/j.procs.2017.06.124

5. Kaur C. The Cloud Computing and Internet of Things (IoT)/ C. Kaur // International Journal of Scientific Research in Science, Engineering and Technology. – 2020 – Vol. 7. Issue 1. http://dx.doi.org/10.32628/IJSRSET196657

6. Atlam H., Walters R., Wills G. Fog Computing and the Internet of Things: A Review. Big Data and Cognitive Computing. – 2018. – Vol. 2, No. 2. - pp. 10. https://doi.org/10.3390/bdcc2020010.

7. All one needs to know about fog computing and related edge computing paradigms: A complete survey / A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala and all // Journal of Systems Architecture. – Vol. 98 – 2019. – p. 289–330. https://doi.org/10.1016/j.sysarc.2019.02.009

8. PunithaIlayarani P., Dominic M. Anatomization of Fog Computing and Edge Computing / P. PunithaIlayarani, M. Dominic. // 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), (Coimbatore, India, 20-22 February 2019). – 2019. https://doi.org/10.1109/ICECCT.2019.8869125

9. Hamdan S., Ayyash M., Almajali S. Edge-Computing Architectures for Internet of Things Applications: A Survey / S. Hamdan, M. Ayyash, S. Almajali // Sensors (Basel). – 2020 – No 20(22). https://doi.org/10.3390%2Fs20226441

10. Ogino T., Kitagami S., Shiratori N. A Multi-agent Based Flexible IoT Edge Computing Architecture and Application to ITS. Journal of Communications. – 2019. – pp. 47–52. https://doi.org/10.12720/jcm.14.1.47-52

11. Wilson M. Best Monitoring Software & Tools for Bandwidth & Traffic Analysis [Електронний ресурс] – PCWDLD.com, 2023 - Режим доступу: https://www.pcwdld.com/router-monitoring-software/#wbounce-modal

12. Rogier B. How to measure network performance metrics via passive traffic analysis? [Електронний ресурс] – accedian.com, 2023 - Режим доступу: https://accedian.com/blog/measure-network-performance-metrics-passive-traffic-analysis/

13. Lamberti A. 19 Network Metrics: How to Measure Network Performance. [Електронний ресурс] – obkio.com, 2023 - Режим доступу: https://obkio.com/blog/how-to-measure-network-performance-metrics/

14. Tkachenko D. How to Use Grafana for Technical Monitoring in Software Products. Tutorial – 2018. [Електронний ресурс]. Режим доступу: https://dzone.com/articles/how-to-use-grafana-for-technical-monitoring-in-sof.

15. Andhavarapu, A. Learning Elasticsearch.Andhavarapu, A. - 2017, Packt Publishing. - p. 404. http://surl.li/labiv

16. Hedrick C. RFC 1058 - Routing Information Protocol – 1998. [Електронний ресурс]. Режим доступу: https://datatracker.ietf.org/doc/html/rfc1058.