Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

на тему «Метод забезпечення функціонування систем з IoT на основі захищених протоколів»

КвРКІ. 2302179.23.02.23 ПЗ

Виконав: студент 2 курсу, група КІ2м-23-2 _____ Владислав КОРШУК
Підпис Ім'я, прізвище

Керівник к.е.н., доцент _____ Світлана САЧЕНКО
Науковий ступінь, вчене звання Підпис Ім'я, прізвище

До захисту допускаю:
Зав. кафедри КІІС, доктор філософії, доцент
Ольга ПАВЛОВА
29 04 2025 р.

Хмельницький, 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЬО-НАУКОВА ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

" 01 " 09 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Владислав КОРШУК

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Керівник проекту (роботи) Світлана САЧЕНКО, к.е.н., доцент

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 08.01.2025 №8

2. Строк подання студентом проекту (роботи) на кафедру 01.05.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз відомих методів та засобів забезпечення функціонування систем з IOT на основі захищених протоколів

Процес захисту комунікацій в системах з IOT на основі бездротового розширення стандарту M-Bus

Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Реалізація захищеного протоколу для пристроїв іот

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

6. Консультанти розділів кваліфікаційної роботи магістра

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Сергій ЛИСЕНКО, професор кафедри КІІС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КІІС		

7. Дата видачі завдання «_01_»_____09_____2024р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи магістра	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики КвРМ з керівником	01.09.2024	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.10.2024	виконано
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	01.11.2024	виконано
4	Робота над розділом 2 – розробка моделей для вирішення поставленої задачі	01.12.2024	виконано
5	Робота над науковою статтею	01.02.2025	виконано
6	Робота над розділом 3 – розробка методів для вирішення поставленої задачі	15.02.2025	виконано
7	Робота над розділом 4 – проектування та розробка ПЗ для вирішення поставленої задачі, експериментальна частина	01.04.2025	виконано
8	Оформлення пояснювальної записки згідно вимог	18.04.2025	виконано
9	Попередній захист ДРМ	29.04.2025	виконано
10	Захист ДРМ на засіданні ЕК	До 15.05.2025	

Студент _____ Владислав КОРШУК
Підпис Ім`я, прізвище

Керівник роботи _____ Світлана САЧЕНКО
Підпис Ім`я, прізвище

РЕФЕРАТ

Тема кваліфікаційної роботи магістра: «Метод забезпечення функціонування систем з IoT на основі захищених протоколів»

Автор роботи: Коршук Владислав Русланович

Керівник роботи: Саченко С.І.

Пояснювальна записка: 73 с., 13 рис., 2 табл., 4 дод., 82 джерела.

ПЕРЕЛІК КЛЮЧОВИХ СЛІВ: протокол, шифр, зловмисник, криптографія, вразливості, пристрої IoT.

Об’єктом дослідження є процес забезпечення функціонування систем з IoT.

Предметом дослідження є методи та протоколи забезпечення функціонування систем з IoT.

Метою кваліфікаційної роботи магістра є покращення ефективності криптографічного захисту від вразливостей в апаратному забезпеченні.

Для розв’язання поставлених задач використовувалися методи криптографії, методи виявлення вразливостей, методи забезпечення функціонування систем з IoT.

Наукова новизна отриманих результатів:

- розроблено новий метод криптографічного захисту від вразливостей в апаратному забезпеченні, в якому на відміну від відомих було розширено його межі застосування для внесених сторонніх знаків в текст.

На основі проведених досліджень розроблено метод та протокол забезпечення функціонування систем з IoT.

Практична значимість отриманих результатів полягає у розроблені захищеного протоколу забезпечення функціонування систем з IoT.

У вступі подано об’єкт та предмет дослідження, мету, наукову новизну та практичну цінність роботи, а також характеристику структури роботи.

У першому розділі проведено аналіз відомих рішень щодо протоколів забезпечення функціонування систем з IoT.

У другому розділі здійснено дослідження предметної області та визначено стратегію захищеного протоколу забезпечення функціонування систем з IoT, а також

розроблено модель системи з пристроями IoT.

У третьому розділі розроблено метод забезпечення функціонування систем з IoT із захищеним протоколом.

У четвертому розділі здійснено розроблення захищеного протоколу забезпечення функціонування систем з IoT та дослідження його стійкості до кіберзагроз та витрат енергії батарей для шифрування повідомлень.

У висновках підведено підсумки досягнення результатів з розв’язання завдань дослідження.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

AES	Advanced Encryption Standard
BLE	Bluetooth Low Energy
BR	Bluetooth Basic Rate
GPS	Global Positioning System
HTTP	Hypertext Transfer Protocol
IDC	International Data Corporation
IEC	International Electrotechnical Commission
IoT	Internet of Things
IoTDs	IoT devices
IPv6	Internet Protocol version 6
IT	Information Technology
LoRa	Long Range
LoRaWAN	Long Range Wide Area Network
LPWAN	Low-Power Wide Area Network
LTE	Long-Term Evolution
MAC	Media Access layer
NFC	Near Field Communication
OSI	Open Systems Interconnection
TCP	Transmission Control Protocol
Wi-Fi	Wireless Fidelity

ВСТУП

Протоколи комунікації для забезпечення функціонування пристроїв в системах з IoT розроблялись без забезпечення заходів безпеки. Вони продовжують поширено використовуватись. Але сьогодення вимагає забезпечувати захист комунікації. Наприклад, при передачі даних з розумних лічильників можуть ці дані ставати доступними зловмисникам і вони впливатимуть на результат. Все це ставить вимоги щодо покращення протоколів комінкації, уведення в них додаткових механізмів захисту.

Перспективним є забезпечення функціонування систем з IoT на основі захищених протоколів, зокрема з використанням Noise Protocol Framework, який захищає від вразливостей і оптимізує енергетичні обмеження пристроїв IoT. Розширення інтелектуальних вимірювань в системі з Інтернету речей (IoT) підкреслює потребу в надійних протоколах безпеки, які захищають передачу даних, оптимізуючи ефективність пристрою. Бездротова шина лічильника, ключовий протокол для дистанційного зчитування показників лічильників у комунальних системах, таких як лічильники газу, води та тепла, стикається зі значними проблемами безпеки.

Актуальність роботи полягає в розробці методу забезпечення функціонування систем з IoT на основі захищених протоколів.

Метою кваліфікаційної роботи магістра є покращення ефективності забезпечення безпеки функціонування систем з IoT на основі захищених протоколів з алгоритмами шифрування.

Поставлена мета досягається розв’язанням таких основних завдань:

- проаналізувати відомі методи забезпечення функціонування систем з IoT та захищені протоколи для зв’язку між пристроями IoT;

- розробити удосконалення методу забезпечення функціонування систем з IoT із захищеним протоколом;

- здійснити реалізацію протоколу згідно розробленого методу забезпечення функціонування систем з IoT із захищеним протоколом;

- здійснити дослідження захищеного протоколу.

Об’єктом дослідження є процес забезпечення функціонування систем з IoT.

Предметом дослідження є методи та протоколи забезпечення функціонування систем з IoT.

Наукова новизна отриманих результатів:

- розроблено новий метод криптографічного захисту від вразливостей в апаратному забезпеченні, в якому на відміну від відомих було розширено його межі застосування для внесених сторонніх знаків в текст.

На основі проведених досліджень розроблено метод та протокол забезпечення функціонування систем з IoT.

Практична значимість отриманих результатів полягає у розроблені захищеного протоколу забезпечення функціонування систем з IoT.

Для розв’язання поставлених задач використовувалися методи криптографії, методи виявлення вразливостей, методи забезпечення функціонування систем з IoT.

За темою кваліфікаційної роботи опубліковано одну публікацію [82] у Збірнику наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп’ютерних наук АПКН-2024». (Хмельницький – 2024. – С. 303-305).

1 АНАЛІЗ ВІДОМИХ МЕТОДІВ ТА ЗАСОБІВ ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНУВАННЯ СИСТЕМ З ІОТ НА ОСНОВІ ЗАХИЩЕНИХ ПРОТОКОЛІВ

1.1 Огляд та поняття протоколів комунікації в IoT

Протоколи комунікації [1, 2] — це набір правил і стандартів, які визначають спосіб обміну даними між пристроями або програмами в мережі. Вони забезпечують узгодженість і правильність передачі інформації, дозволяючи різним системам взаємодіяти одна з одною.

Основні аспекти протоколів комунікації [3, 4]:

1) формат даних, який визначає структуру передаваної інформації;

2) синхронізація забезпечує узгодженість передачі й отримання даних;

3) контроль помилок виявляє і коригує помилки під час передачі;

4) аутентифікація перевіряє дійсність учасників комунікації;

5) шифрування забезпечує безпеку переданих даних.

Розглянемо приклади таких протоколів [5, 6]:

1) протокол HTTP/HTTPS для передачі веб-сторінок у мережі;

2) протокол FTP для передачі файлів;

3) протокол SMTP/IMAP/POP3 для роботи з електронною поштою;

4) протокол TCP/IP для передачі даних у мережі Інтернет;

5) протокол Bluetooth для бездротового з'єднання пристроїв;

6) протокол WebSocket для двосторонньої взаємодії в реальному часі.

Для систем Інтернету речей (IoT, Internet of Things) використовуються спеціалізовані протоколи [7, 8], які враховують обмеження пристроїв (низьке енергоспоживання, обмежена обчислювальна потужність) та специфіку роботи в розподілених мережах. Основні протоколи, які використовуються в IoT:

1) мережеві протоколи IoT забезпечують підключення пристроїв до мережі:

1.1) Wi-Fi (802.11) використовується для високошвидкісного обміну даними, часто в розумних будинках;

1.2) Bluetooth LE (Low Energy) ефективний для пристроїв з низьким

енергоспоживанням;

1.3) Zigbee - протокол для низькошвидкісних, енергоефективних бездротових мереж, популярний у розумних будинках;

1.4) LoRaWAN підходить для передачі даних на великі відстані з низьким енергоспоживанням (у сільському господарстві, смарт-містах);

1.5) NB-IoT (Narrowband IoT) - мобільний стандарт для пристроїв з низькими вимогами до швидкості передачі даних;

1.6) Thread - сучасний протокол для розумних будинків, забезпечує високу сумісність між пристроями.

2) транспортні протоколи IoT забезпечують доставку даних [9]:

2.1) UDP (User Datagram Protocol) - легкий протокол, підходить для систем реального часу;

2.2) TCP (Transmission Control Protocol) забезпечує надійну передачу даних, але може бути ресурсозатратним;

2.3) MQTT (Message Queuing Telemetry Transport) - легкий протокол для обміну повідомленнями, ідеальний для енергоефективних пристроїв і нестабільних мереж;

2.4) CoAP (Constrained Application Protocol) - протокол на основі HTTP, розроблений для пристроїв з обмеженими ресурсами;

2.5) AMQP (Advanced Message Queuing Protocol) для обміну повідомленнями в системах з високими вимогами до надійності.

3) протоколи прикладного рівня IoT [2] використовуються для взаємодії між пристроями та хмарними сервісами:

3.1) HTTP/HTTPS - стандартний протокол для веб-застосунків, використовується в більш потужних IoT-пристроях;

3.2) WebSocket забезпечує двосторонній обмін даними в реальному часі;

3.3) DDS (Data Distribution Service) використовується для складних і масштабованих IoT-систем;

3.4) XMPP (Extensible Messaging and Presence Protocol) застосовується для обміну повідомленнями в реальному часі.

4) протоколи для управління та безпеки:

4.1) DTLS (Datagram Transport Layer Security) забезпечує безпеку на основі UDP;

4.2) TLS/SSL - шифрування та захист даних у мережах;

4.3) OCPP (Open Charge Point Protocol) для управління зарядними станціями електромобілів;

4.4) LWM2M (Lightweight M2M) - протокол управління пристроями IoT з низькими ресурсами.

Ключові особливості протоколів IoT [10]:

1) енергоефективність;

2) мала пропускна здатність;

3) підтримка роботи в мережах з великою кількістю пристроїв;

4) масштабованість та сумісність.

Архітектуру стеку IoT зображена на рис. 1 [2].

Порівняння стеку Z-хвилі зі стеком OSI зображено на рис. 2 [2].

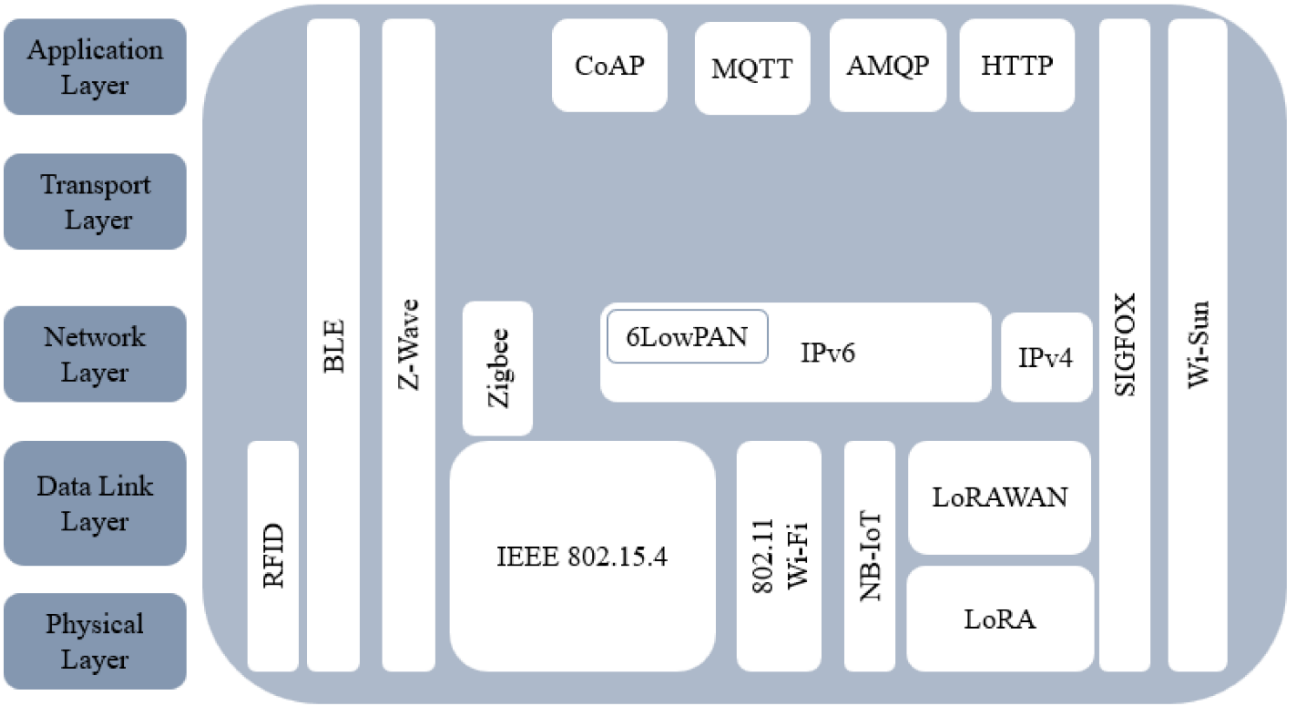


Рисунок 1 - Архітектура стеку IoT [1]

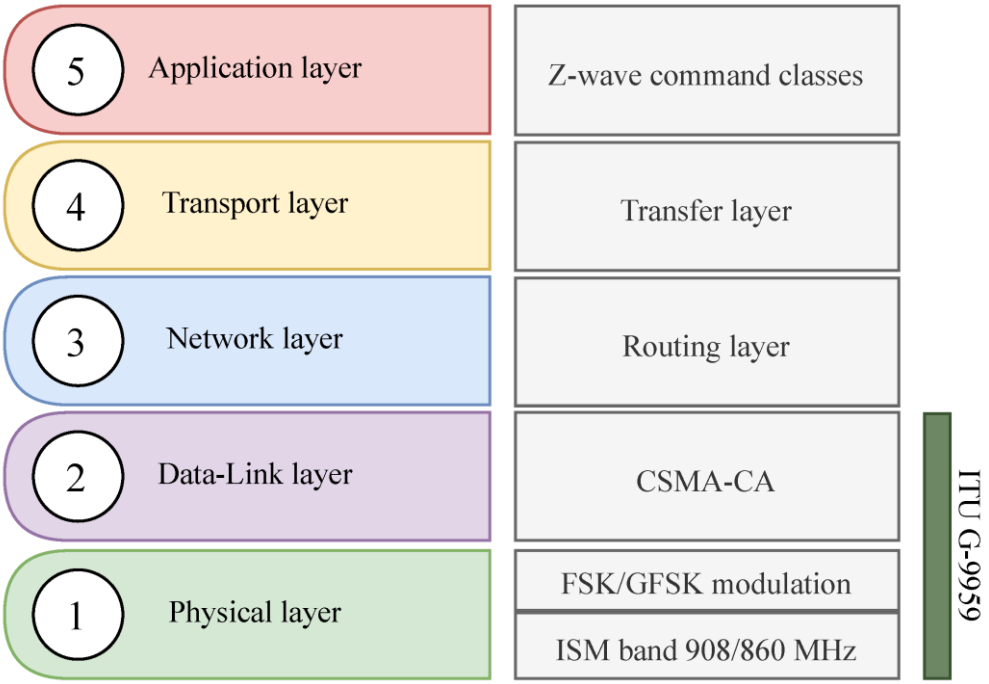


Рисунок 2. Порівняння стека Z-хвилі зі стеком OSI [2]

Якщо потрібно отримати захищений протокол для IoT, що є дуже актуальним для сьогодення, то можна використати спеціальні засоби, які підтримують такі вимоги. Наприклад, Noise Protocol Framework [11, 2] — це криптографічний фреймворк, розроблений для створення безпечних протоколів комунікації. Він забезпечує гнучкий та модульний підхід до реалізації криптографічних протоколів рукостискання, що дозволяє налаштовувати та впроваджувати захищені канали зв’язку для широкого спектра застосувань.

Основні характеристики Noise такі: модульність; простота; визначені властивості безпеки; ефективність; гнучкість.

Модульність Noise базується на базових криптографічних елементах, таких як обмін ключами Диффі-Геллмана, симетричне шифрування та гешування, які можна комбінувати для досягнення конкретних цілей протоколу.

Простота полягає в тому, що фреймворк розроблено з акцентом на зрозумілість і відсутність зайвої складності.

Властивості безпеки. Протоколи Noise аналізуються на предмет таких властивостей, як пряма секретність, взаємна автентифікація та стійкість до повторних

атак.

Ефективність. Протоколи, створені на основі Noise, дуже ефективні, що робить їх придатними для використання в обмежених середовищах, наприклад, у пристроях IoT.

Гнучкість. Noise дозволяє використовувати різні криптографічні алгоритми та конфігурації, які можна адаптувати під конкретні потреби застосунку.

Основні компоненти протоколів Noise [12, 13].

Шаблони рукостискань (Handshake Patterns) визначають послідовність повідомлень між учасниками та криптографічні операції, які виконуються. Наприклад, `NN` (без автентифікації), `XK` (автентифікація клієнта, статичний ключ сервера) тощо.

Криптографічні алгоритми в Noise підтримуються різні, зокрема:

1) обмін ключами Диффі-Геллмана (наприклад, Curve25519, Curve448);

2) гешування (SHA-256, BLAKE2);

3) симетричне шифрування (AES-GCM, ChaCha20-Poly1305);

4) ролі учасників протоколу - це зазвичай "ініціатор" та "відповідач";

5) симетричний стан (Symmetric State) – це структура, яка керує шифруванням та дешифруванням, включаючи ключі та геш-ланцюг для генерації наступних ключів.

Приклад шаблону рукостискання

Шаблон рукостискання Noise — це опис послідовності обміну ключами. Наприклад:

`Noise_XK_25519_AESGCM_SHA256`

`XK`: Шаблон, де ініціатор надсилає ефемерний відкритий ключ, а сервер відповідає статичним ключем.

`25519`: Використовується крива Curve25519 для обміну ключами.

`AESGCM`: Шифрування з використанням AES-GCM.

`SHA256`: Геш-функція SHA-256.

Цей підхід забезпечує високу гнучкість у розробці протоколів безпеки для сучасних систем.

Метод забезпечення функціонування систем з IoT на основі захищених протоколів, зокрема з використанням Noise Protocol Framework, який захищає від вразливостей і оптимізує енергетичні обмеження пристроїв IoT. Розширення інтелектуальних вимірювань в системі з Інтернету речей (IoT) підкреслює потребу в надійних протоколах безпеки [14, 15], які захищають передачу даних, оптимізуючи ефективність пристрою. Бездротова шина лічильника, ключовий протокол для дистанційного зчитування показників лічильників у комунальних системах, таких як лічильники газу, води та тепла, стикається зі значними проблемами безпеки.

Згідно [2, 5-14] можна подати схемою на рис. 3 середовище з IoT.

Рисунок 3 – Схема середовища з IoT [2, 5-14]

Інтернет речей (IoT) зробив трансформаційний зсув у цифровому ландшафті. Він поєднує величезну кількість пристроїв, від повсякденних предметів домашнього вжитку до складних промислових інструментів за допомогою Інтернету. Ця нова мережа має вирішальне значення в сучасних технологічних системах, використовуючи збір та аналіз даних для підвищення ефективності, функціональності та користувацького досвіду в різних доменах. Зокрема, в управлінні комунальними

послугами. Інновації Інтернету речей, такі як розумні лічильники, революціонізують спосіб моніторингу та управління комунальними послугами, такими як газ, вода та електроенергія, сприяючи відстеженню даних у режимі реального часу, оптимізації споживання та прогнозному обслуговуванню. Центральне місце в ефективності Інтернету речей у цих застосуваннях займає роль технологій бездротового зв'язку, серед яких виділяється бездротова вимірювальна шина [15]. Як європейський стандарт для зчитування показників лічильників бездротового зв'язку, вона відіграє вирішальну роль у забезпеченні віддаленого збору та передачі даних про використання комунальних послуг. Це не тільки оптимізує операційні процеси, але й надає споживачам глибоке розуміння своїх моделей споживання, що забезпечує як економічні, так і екологічні вигоди.

У міру того, як розвивається цифрова інтеграція управління комунальними послугами за допомогою таких пристроїв, як розумні лічильники, важливість надійних систем безпеки стає все більш очевидною. Протокол, призначений для дистанційного зчитування показників комунальних послуг, лежить в основі цієї інфраструктури, забезпечуючи ефективну передачу даних через великі мережі. Однак його поточні механізми безпеки демонструють вразливості, які можуть бути використані для порушення конфіденційності даних, цілісності та доступності системи, що призводить до потенційного несанкціонованого доступу, маніпуляції даними та збоїв у обслуговуванні. Ці вразливості не тільки створюють загрозу для конфіденційності та довіри споживачів, але й загрожують відмовостійкості та надійності систем інтелектуальних мереж.

При розгляді переважаючих практик у галузі або управлінні комунальними послугами можна помітити, що заходи безпеки часто ігноруються на користь полегшення розгортання та зниження витрат. Як правило, основна увага приділяється встановленню лічильників з мінімальними початковими інвестиціями та експлуатаційними перешкодами, маючи намір не вимагати обслуговування. Однак, коли впроваджуються протоколи безпеки, вони часто покладаються на єдиний ключ для всіх лічильників, як правило, ключ виробника за замовчуванням, що надається разом із партією лічильників. Такий підхід створює значний ризик. Якщо один ключ

скомпрометований, це робить усі лічильники вразливими, потенційно ставлячи під загрозу всю мережу.

Визнаючи ці виклики, необхідно розробляти і використовувати впровадження надійних, адаптивних рішень безпеки, які захищають окремі лічильники та ширшу систему від таких вразливостей, тим самим підвищуючи загальний стан безпеки систем управління комунальними послугами IoT. Прийняття Noise Protocol Framework стає потенційним рішенням. Він надає можливість значно покращити ландшафт безпеки зв'язку wM-Bus [2]. Мета полягає в тому, щоб створити структуру комплексних заходів безпеки, які забезпечать захист від існуючих загроз, залишаючись при цьому гнучкими для реагування на майбутні виклики.

Таким чином, забезпечення функціонування систем з IoT на основі захищених протоколів може бути вирішено з використанням Noise Protocol Framework.

1.2 Відомі методи та засоби забезпечення функціонування систем з IoT на основі захищених протоколів

Розглянемо відомі методи та існуючі засоби забезпечення функціонування систем з IoT на основі захищених протоколів. Проведемо їх аналіз.

У роботі [14] представлено метод підвищення безпеки шляхом інтеграції Noise Protocol Framework, що полягає в захисті від вразливостей і оптимізації енергетичних обмежень пристроїв IoT. Спочатку розглядалися питання безпеки, особливо в інтелектуальних лічильниках, що працюють від батарейок, і в дослідженні вивчаються легкі, адаптивні рішення безпеки. Аналіз впровадження фокусується на шаблонах рукостискання, оцінюючи їх сумісність за допомогою таких показників, як використання пам'яті, розмір пакету та час рукостискання. Результати показують, що ці моделі значно перевершують традиційні методи, такі як безпека транспортного рівня, у зниженні споживання енергії, тим самим продовжуючи термін служби пристроїв IoT. Дослідження дозволило скоротити час автономної роботи, що підвищило як безпеку, так і ефективність. Запропонований легкий протокол ефективно поєднує покращену безпеку та ефективність, зберігаючи

конфіденційність, цілісність та доступність даних у інтелектуальних вимірюваннях без шкоди для продуктивності. Тестування безпеки проти моделей підміни, несанкціонованого втручання, відмови в розголошенні, відмови в обслуговуванні та підвищення рівня привілеїв підтвердило стійкість цього нового протоколу, тим самим покращивши рамки безпеки.

Технології Smart Grid [16] та Інтернету речей (IoT) представляють трансформаційні підходи для модернізації та підвищення ефективності, надійності та стійкості систем електроенергетики в усьому світі. Інтелектуальні мережі інтегрують цифрові комунікації та передові технології для більш ефективного управління потоком електроенергії, враховуючи відновлювані джерела енергії та забезпечуючи моніторинг і управління в режимі реального часу. З іншої сторони, IoT забезпечує зв'язок з повсякденними об'єктами, перетворюючи їх на інтелектуальні пристрої, здатні спілкуватися та взаємодіяти один з одним та з користувачами, тим самим пропонуючи глибокі наслідки для управління енергією та моделей споживання [17-19]. Інтеграція IoT в Smart Grids має вирішальне значення для досягнення високо інтерактивної, чутливої та автоматизованої енергетичної екосистеми. Ця інтеграція забезпечує розширені функції, такі як реагування на попит, де пристрої можуть регулювати споживання електроенергії у відповідь на сигнали з мережі, сприяючи стабільності та ефективності. Крім того, технології IoT сприяють посиленому моніторингу та прогнозному обслуговуванню компонентів мережі, знижуючи час простою та експлуатаційні витрати [20, 21]. Інтеграція технологій Інтернету речей (IoT) з інтелектуальними мережевими системами поєднує спосіб моніторингу, управління та оптимізації використання енергії. Пристрої IoT, включаючи датчики, маршрутизатори, шлюзи та інтелектуальні лічильники, мають значення для підвищення зв'язку та ефективності в енергетичному секторі [22, 23]. Ці технології полегшують відстеження даних у режимі реального часу, прогнозоване технічне обслуговування та оптимізацію споживання, закладаючи основу для більш стійкої та орієнтованої на споживача енергетичної екосистеми. Перехід до інтелектуальних мереж подає собою критичну еволюцію в управлінні енергією, що втілює перехід від традиційних мереж до динамічних, взаємопов'язаних мереж, здатних адаптуватися до

мінливих вимог сучасного енергоспоживання [24, 25]. Оскільки ці технології продовжують розвиватися та вдосконалюватися, то вони можуть революціонізувати способи виробництва, розподілу та споживання електроенергії.

Комунікаційні технології [26, 27] є основою інтелектуальних мереж з підтримкою Інтернету речей, забезпечуючи безперебійну взаємодію між різними пристроями в мережі. Ці технології, які охоплюють бездротові, стільникові та недорогі частотні рішення, мають важливе значення для ефективного впровадження застосунків IoT в інтелектуальних мережах. Вони забезпечують надійну передачу даних, підтримують віддалений моніторинг і управління, а також полегшують інтеграцію відновлюваних джерел енергії в мережу. Вони відіграють важливу роль у досягненні енергоефективності, підвищенні стійкості мережі та наданні споживачам детальної інформації про споживання енергії [28, 29]. Технології малопотужних глобальних мереж стали критично важливими факторами, що сприяють застосуванню IoT у розумних мережах. Технології LPWAN розроблені для забезпечення можливостей зв'язку на великій відстані з мінімальним споживанням енергії, що робить їх ідеальними для пристроїв IoT, яким потрібно працювати протягом тривалого часу на обмежених джерелах живлення. Вони відіграють значну роль у інтелектуальних вимірюваннях, мережах датчиків та системах моніторингу, які сприяють оптимізації роботи мережі та інтеграції розподілених енергетичних ресурсів [30 - 34]. Вони забезпечують [35, 36] необхідну інфраструктуру для надійного, ефективного та безпечного зв'язку між безліччю пристроїв, що складають систему інтелектуальних мереж.

Розглянемо протокол wM-Bus [37, 38]. Він розглядається як критично важливий протокол зв'язку в рамках передової вимірювальної інфраструктури, який в основному використовується для інтелектуальних вимірювальних програм. Його впровадження для обліку комунальних послуг [39, 40] підкреслює його важливість в екосистемі IoT, особливо для сприяння ефективному управлінню комунальними послугами. Протокол wM-Bus розроблений для забезпечення безпечного та надійного зв'язку [41, 42] між лічильниками та збирачами даних, що дозволяє дистанційно зчитувати показники комунальних лічильників. Цей протокол має вирішальне

значення для операційної ефективності [43, 44] інтелектуальних мереж. Однак протокол wM-Bus стикається зі значними проблемами безпеки [45, 46]. Основна проблема полягає не в протоколі або його стандартних специфікаціях, а в реалізації заходів безпеки. Багато компаній з управління нерухомістю та комунальних підприємств вибирають або незашифровану передачу даних лічильників, або використання одного ключа шифрування для всіх лічильників. Цей підхід, який першочергово використовується для спрощення розгортання, зниження обчислювальних витрат і мінімізації складності безпеки, піддає лічильники потенційним атакам [47, 48].

Дослідження, проведене в [49] виділяє ці вразливості безпеки, включаючи короткі розміри ключів, виявлення нульового споживання та передачу відкритого тексту, що ризикує розкриттям конфіденційної інформації, включаючи ключі шифрування, тим самим сприяючи атакам типу "зловмисник посередині". Крім того, для лічильників, що працюють від батарейок, помітна відсутність надійних заходів безпеки [50].

Проблеми безпеки [51, 52] в бездротовому зв'язку, особливо для пристроїв IoT, а точніше для тих, що живляться від батарейок, є критично важливою сферою, що викликає занепокоєння в сучасному взаємопов'язаному світі. Для таких застосунків, як інтелектуальні вимірювання, де дані про споживання енергії в режимі реального часу є життєво важливими, то ці проблеми стають ще більш очевидними через потребу в постійному, надійному зв'язку та цілісності даних. Пристрої IoT часто розгортаються у великих кількостях і в потенційно незахищених середовищах, що робить їх вразливими до різноманітних загроз [53-56]. Вони варіюються від фізичного втручання та несанкціонованого доступу до більш складних кібератак, спрямованих на компрометацію функціональності пристрою або даних, які він передає [57]. Обмежена обчислювальна потужність та енергетичні ресурси багатьох пристроїв IoT ще більше ускладнюють впровадження надійних заходів безпеки [58], роблячи їх слабкою ланкою в ланцюжку безпеки мереж, у яких вони живуть. Розумні лічильники [59] повинні забезпечувати конфіденційність, цілісність і доступність даних, які вони збирають і передають, при цьому працюючи ефективно, щоб підтримувати тривалий

термін служби батареї та мінімізувати технічне обслуговування. Атаки на інтелектуальні лічильники можуть призвести до крадіжки енергії, помилкового виставлення рахунків і навіть нестабільності мережі, якщо скомпрометовані дані використовуються для маніпулювання попитом і пропозицією енергії [60]. Протоколи зв'язку, що використовуються інтелектуальними лічильниками, такі як wM-Bus [61, 62], хоча й ефективні для застосунків із низьким енергоспоживанням у глобальній мережі, можуть бути вразливими до прослуховування та несанкціонованого доступу, якщо вони не захищені належним чином. Впровадження заходів безпеки [63, 64], таких як шифрування та виявлення аномалій, може пом'якшити деякі з цих ризиків, але вони також повинні бути ретельно збалансовані, щоб уникнути значного збільшення споживання енергії пристроєм, що потенційно може скоротити термін його служби. Шина wM-Bus [65-69] розроблена для систем зчитування лічильників і вимагає індивідуальних рішень безпеки, які усувають її унікальні вразливості, зберігаючи при цьому енергоефективність, необхідну для пристроїв, що живляться від акумуляторів. Захист бездротового зв'язку для пристроїв IoT [70, 71], частково в застосунках вимірювання, що використовують wM-Bus, вимагає багатогранного підходу, який вирішує проблеми, пов'язані з обмеженими ресурсами цих пристроїв і вразливостями бездротових протоколів.

Фреймворк NPF [72] постає як перспективне рішення для підвищення безпеки пристроїв IoT та зв'язку в інтелектуальних мережах. У цьому фреймворку використовується передова криптографія методи створення безпечних каналів зв'язку, забезпечення шифрування даних та захисту від різних кіберзагроз. Його застосування в безпеці IoT та за його межами, у тому числі в основних платформах обміну повідомленнями, підкреслює [73, 74] його універсальність та ефективність у вирішенні складних проблем безпеки, притаманних сучасним цифровим інфраструктурам. NPF вирізняється своєю безпекою та полегшеною конструкцією, що робить його високоефективним у обміні повідомленнями, віртуальних приватних мережах та застосунках IoT, особливо в середовищах з обмеженими ресурсами [75]. Фреймворк забезпечує безпечний обмін даними через ненадійні мережі, ефективно захищаючи від прослуховування, несанкціонованого доступу та видавання себе за

іншу особу.

У роботі [76] досліджуємо проблему планування обчислювальних завдань у інтегрованій мережі для послуг, орієнтованій на затримку Інтернету (IoT). У розглянутому сценарії безпілотний повітряний транспортний засіб збирає обчислювальні завдання з пристроїв IoT, а потім приймає в Інтернеті розвантажувальні рішення, в яких завдання можуть бути оброблені безпосередньо у вузлі або вивантажуються на сусідню базову станцію або віддалений супутник.

Поєдннання технології інформаційної комунікації [77] з Grid-орієнтованою потужністю Інтернету речей стало критичним гарантування безпечної та надійної роботи потужності та підвищення енергоефективності системи. Тим не менш, пристрої мають лише обмежені комунікації та обчислювальні ресурси, оскільки вони в основному розгорнуті у віддалених областях, які можуть бути поза висвітленням існуючих наземних мереж. Щоб подолати обмеження ресурсів, використано інтегровані гетерогенні мережі Ця проблема належить до змішаного цілого нелінійного програмування з застосунковими труднощами, де тривала затримка черги та короткострокові обмеження пов'язані.

Інтегровані мережі вважаються ключовою структурою мережі наступного покоління [78]. Космічні супутники та повітряні вузли є потенційними кандидатами, які допоможуть та вивантажують передачу місцевості. Однак, завдяки високій мобільності космічних та повітряних вузлів, а також високої динаміки мережевого трафіку, звичайна стратегія розвантаження трафіку не застосовується для високого динамічного сагіна. У роботі запропоновано підсилення на основі навчання на основі навчання, враховуючи високу рухливість вузлів, а також часті мінливі мережевий трафік та стан посилань.

Концепція фіксованих інфраструктур [79], здатних виконувати вимоги рухомих пристроїв з точки зору зв’язку та надійності, було оптимальним рішенням протягом останніх кількох десятиліть. Сьогодні таке рішення вже неможливо в IoT пов’язаних пристроях. Зараз все пов'язане, і значна кількість з них є мобільними, отже, призводить до зв’язку та надійності. Підключені та автономні транспортні засоби, крім більш сучасних літаючих та рухомих пристроїв, таких як безпілотники та

пристрої IoT, відіграватимуть значну роль у мережах нового покоління. Зв'язок із вузлом до вузла також відіграватиме ключову роль та надаватиме альтернативні рішення щодо зв’язку в багатьох складних середовищах для таких застосунків.

Посильний інтернет речей [80] може задовольнити потреби комунікації для IoT у населених пунктах з недостатнім покриттям наземної базової станції. Враховуючи, що ємність акумулятора пристроїв обмежена і її важко замінити, використано безпілотні літальні транспортні засоби для надання допомоги бездротової передачі електроенергії, щоб забезпечити якість мережевої послуги та безпечну та стабільну роботу.

Таким чином, забезпечення функціонування систем з IoT потребує удосконалення в частині підтримання зв’язків між пристроями і використання захищених протоколів.

1.3 Постановка задачі

Поставлена мета досягається розв’язанням таких основних завдань:

- проаналізувати відомі методи забезпечення функціонування систем з IoT та захищені протоколи для зв’язку між пристроями IoT;
- розробити удосконалення методу забезпечення функціонування систем з IoT із захищеним протоколом;
- здійснити реалізацію протоколу згідно розробленого методу забезпечення функціонування систем з IoT із захищеним протоколом;
- здійснити дослідження захищеного протоколу.

1.4 Висновки до першого розділу

Проаналізовано відомі методи та засоби забезпечення функціонування систем з IoT із захищеним протоколом, а також визначено стратегію для покращення ефективності цього процесу. Для забезпечення захищеним протоколом систем з IoT запропоновано використати фреймворк Noise Protocol Framework.

2 ПРОЦЕС ЗАХИСТУ КОМУНІКАЦІЙ В СИСТЕМАХ З IOT НА ОСНОВІ БЕЗДРОТОВОГО РОЗШИРЕННЯ СТАНДАРТУ M-BUS

2.1 Стратегія захисту пристроїв Інтернету речей з використанням протоколу Wireless M-Bus

Захист пристроїв Інтернету речей (IoT) є критично важливим, оскільки ці пристрої часто працюють у вразливих середовищах і можуть стати об'єктом атак. Основні причини, чому потрібно використовувати протоколи з підвищеними вимогами до безпеки для захисту IoT-пристроїв.

Захист від кіберзагроз. Масові атаки IoT-пристрої часто стають цілями ботнетів (наприклад, Mirai), які використовують незахищені пристрої для масових атак, таких як DDoS. Віддалений контроль. Зловмисники можуть отримати доступ до пристроїв і використовувати їх для шкідливих дій, наприклад, відключення систем або викрадення даних.

Конфіденційність даних. IoT-пристрої часто збирають чутливу інформацію (наприклад, дані про місцезнаходження, поведінку, стан здоров'я). Без належного шифрування та автентифікації, дані можуть бути перехоплені або змінені зловмисниками.

Небезпека фізичного втручання. IoT-пристрої, які керують критичними системами (наприклад, розумними замками, системами управління електромережею або медичними пристроями), можуть становити фізичну небезпеку, якщо їх зламано.

Масштабність загроз. Оскільки кількість IoT-пристроїв швидко зростає, то зловмисники отримують велику кількість потенційних точок входу до мереж. IoT-пристрої часто працюють у глобальній мережі, що розширює можливості атак.

Відповідність законодавству. Багато країн запроваджують регулювання безпеки IoT-пристроїв, вимагаючи дотримання стандартів безпеки. Наприклад, захист персональних даних і недотримання вимог може призвести до великих штрафів.

Запобігання економічним збиткам. Наприклад, хакерські атаки на промислові IoT-пристрої можуть призвести до фінансових втрат. Також, злам IoT-пристроїв компанії підриває довіру споживачів.

Розглянемо механізми безпеки для застосування в протоколі. Шифрування даних гарантує, що дані залишаться захищеними під час передачі (наприклад, TLS, AES). Автентифікація передбачає використання сертифікатів, токенів або багатофакторної автентифікації для перевірки пристроїв та користувачів. Захист протоколів передбачає використання захищених протоколів, таких як MQTT з TLS, CoAP із DTLS, wM-Bus із шифруванням. Регулярне оновлення пристроїв для усунення вразливостей. Використання ролей і політик для обмеження доступу до даних і функцій пристроїв.

Тому, захист IoT-пристроїв протоколами з підвищеними вимогами до безпеки є необхідним для запобігання витоку даних, захисту від кіберзагроз, підтримки стабільності критичних інфраструктур, відповідності стандартам безпеки та законодавству.

Протокол wM-Bus (Wireless M-Bus) є бездротовим розширенням стандарту M-Bus, розробленого для передачі даних з лічильників води, газу, електроенергії та тепла. Він широко застосовується в Інтернеті речей (IoT) для створення енергоефективних та надійних систем збору даних. Розглянемо основні характеристики wM-Bus. Базується на стандарті EN 13757-4, який визначає фізичний і канальний рівні бездротової передачі. Використовує неліцензовані частоти ISM-діапазону (наприклад, 868 МГц у Європі або 915 МГц в інших регіонах). Протокол підтримує кілька режимів передачі даних. Призначений для пристроїв, які працюють у стаціонарному режимі (наприклад, інфраструктурні лічильники), для пристроїв з обмеженим енергоспоживанням. Забезпечує постійну двосторонню передачу даних. Підходить для пристроїв із низькою частотою обміну даними. Використовуються в спеціалізованих сценаріях. Підтримує низьке енергоспоживання, що важливо для пристроїв на батарейному живленні. У режимі T-mode або C-mode передача даних відбувається лише за потреби, зберігаючи заряд акумулятора. Він включає механізми шифрування та автентифікації, що робить протокол безпечним для комерційних застосунків. Залежно від оточення може досягати 500–1000 м в умовах прямої видимості. Використовується для автоматизованого зчитування даних з лічильників (AMR – Automated Meter Reading). Допомагає в управлінні водними, енергетичними

та газовими ресурсами. Може здійснювати віддалений контроль стану обладнання в реальному часі. Застосовується в сенсорах температури, вологості, тиску тощо. Його перевали: низьке енергоспоживання; простота інтеграції; використання стандартних частот ISM; надійна передача даних; масштабованість для великих мереж.

Концепція IoT поставила перед дослідниками чітку мету забезпечення того, щоб розумні пристрої були підключені до загальної платформи і могли взаємодіяти один з одним. Ця мета може бути досягнута шляхом встановлення єдиного стандарту зв'язку. Пристрої IoT, як правило, живляться від акумуляторів, і їх основні вимоги включають: низьке споживання енергії тривалий час автономної роботи; можливість підключення оптимізовано для низького обсягу даних; низька вартість, низьке обслуговування та безпека. Пристрої IoT можуть бути підключені за допомогою різних бездротових технологій і використовувати різний тип покриття підключення. Бездротові мережі, такі як бездротові та домашні мережі — це мережі малого радіусу дії, які використовують бездротові технології,. Бездротові локальні мережі — це мережі середнього радіусу дії з близькістю від 5 до 10 км, які покриваються Wi-Fi, WM-Bus і Wi-SUN. Далекобійні технології можуть розширюватися на понад 100 км і включають неліцензовані технології, які називаються малопотужними глобальними мережами. Існують також ліцензовані технології, такі як стільниковий зв'язок. Так, стільниковий і вузькосмуговий Інтернет речей відомі як технології CIoT. Здійснимо дослідження новітніх технологій бездротового зв'язку, які підтримують безліч діапазонів покриття в бездротовому з'єднанні. Розглянемо та порівняємо архітектуру, безпеку, стандартні функції та стек протоколів кожної технології.

Система IoT цінна тим, що використовує та представляє дані. Ця система повинна представляти архітектуру, яка забезпечує безпеку, стабільність і доступність Інтернету речей. Інфраструктура IoT базується на компонентах, включаючи пристрої, шлюзи, хмару та програми. Апаратні або програмні пристрої часто називають вузлами або кінцевими пристроями. Блок збору даних шлюз дозволяє передавати дані в хмару. Хмара створює механізм обробки потоків даних зі шлюзів або безпосередньо з кінцевих пристроїв. Сервіси IoT, такі як зв'язок, безпека, управління пристроями та послуги даних, є важливими для підтримки загальної архітектури. LPWAN — це тип

WLAN, призначений для підключення кінцевих пристроїв IoT, які живляться від батареї та мають низьку пропускну здатність. Різні бездротові технології з різним покриттям зв'язку конкурують між собою або займають чільне місце в області LPWAN. Однак вибір найбільш підходящої бездротової технології є складним завданням через велику кількість доступних варіантів, починаючи від неліцензованих і закінчуючи ліцензованими технологіями. Промислові, наукові та медичні діапазони – це відкриті частотні діапазони, які відрізняються залежно від регіонів та ліцензій. Бездротові пристрої використовують різні стандарти протоколу для обміну, включаючи власні або відкриті стандарти. Платформа IoT була розроблена для об'єднання мільярдів різнорідних пристроїв, що вимагає адаптованої архітектури з багаторівневими стандартами протоколів. Багато стандартів бездротових протоколів розроблено на основі IEEE 802.11 або IEEE 802.15.4g на різних рівнях структури.

Безпека IoT вимагає захисту пристроїв IoT та їхніх мереж. Однак аспекти безпеки не враховуються при виробництві пристроїв IoT. Установи повинні захищати кінцеві пристрої, впроваджуючи стандарти безпеки на різних рівнях, і захищати протоколі для застосунків IoT за допомогою надійних механізмів аутентифікації користувачів і методів шифрування. Рішення для безпеки IoT повинні розроблятися в рамках кожної технологічної архітектури. Основними вимогами безпеки IoT є цілісність даних та безпека пристрою. Цифрова сертифікація має бути активована для захисту пристроїв, наприклад, перевірка та автентифікація, щоб авторизувати оновлення мікропрограми та захистити зв'язок між пристроями. Ці вимоги безпеки впливають на кілька рівнів протоколу. Існують компроміси між безпекою та продуктивністю, що ставить під загрозу енергоспоживання. Хоча вони підходять для зв'язку на короткій відстані, вони не ідеальні для розумних лічильників через обмежений радіус дії та потенційні перешкоди в густонаселених районах. Крім того, енергоспоживання є високим, і все ще не підтримує тривалий час автономної роботи, необхідний для застсоунків інтелектуального вимірювання. Сприйнятливість до різних атак на безпеку ще більше знижує його життєздатність для критичної інфраструктури, такої як розумні лічильники.

Стандарт бездротової мережі пропонує надійний, безпечний зв'язок із низьким

енергоспоживанням з прийнятною швидкістю передачі даних. Він складається з координатора, маршрутизатора та кінцевих пристроїв. Координатор керує мережею та передачею даних. Він підтримує функції маршрутизації та множинного стрибка для відмовостійкості. Безпека забезпечується на мережевому та прикладному рівнях за допомогою механізмів управління ключами, захисту даних та авторизації. Придатність розумного лічильника підходить для розумних лічильників завдяки низькому енергоспоживанню, надійності та безпечному зв'язку. Однак його обмежений радіус дії та потенційна можливість збоїв доставки через деревоподібну топологію можуть створювати проблеми. Він підключає пристрої через центральний концентратор за допомогою частотного зсуву та багатошляхової маршрутизації. Чотирирівневий протокол включає прикладний, транспортний, мережевий і фізичний рівні. Функції безпеки включають аутентифікацію, конфіденційність та захист від повторних атак за допомогою шифрування. Придатність розумного лічильника не є ідеальним для розумних лічильників через обмежений радіус дії та неможливість підключення до Інтернету. Однак він забезпечує безпечний зв'язок, що підходить для домашньої автоматизації. Сумісність розумних лічильників не є ідеальним для розумних лічильників через високе енергоспоживання та вразливість до бездротових атак, хоча він забезпечує надійне з'єднання. Він забезпечує надійну автентифікацію та шифрування, але стикається з новими проблемами безпеки через різні впровадження виробниками. Застарілі пристрої IoT не підтримують IP, що обмежує мережеву інтеграцію. Він підходить для розумних лічильників завдяки своєму далекому покриттю та ефективному проникненню через перешкоди. Однак перешкоди від інших пристроїв і відсутність комерційно доступних чіпів є проблемними. Він інтегрує IEEE 802.15.4g для вдосконалених комунальних мереж з використанням вузькосмугової бездротової технології. Це відкритий стандарт для безпечного, сумісного зв'язку IoT, який підходить для комунальних служб розумних мереж. Він підтримує понад тисячу вузлів з низькошвидкісним зв'язком і працює в топологіях «зірка» або «сітка». Він забезпечує максимальну швидкість передачі даних і мінімальну затримку. Безпека включає розширюваний протокол автентифікації, стандарти та шифрування. Він дуже підходить для розумних

лічильників завдяки своїй безпечній, малопотужній і масштабованій природі, що робить його ідеальним для великомасштабного розгортання.

Шина бездротового лічильника WM-Bus в основному використовується для дистанційного зчитування лічильників газу, води, тепла або електроенергії, а також підходить для промислових бездротових сенсорних мереж. WM-Bus працює, підтримуючи зв'язок на великій відстані з низьким споживанням енергії. Він використовує мережу з топологією «зірка» з головними та підлеглими пристроями, пропонуючи прийнятну швидкість передачі даних. WM-Bus має обмежені функції безпеки, які підтримують лише шифрування, але не підтримують інші механізми, такі як аутентифікація пристрою та розподіл ключів. WM-Bus добре підходить для розумних лічильників завдяки низькому енергоспоживанню, тривалому терміну служби батареї і надійному діапазону зв'язку. Однак він має обмежені функції безпеки.

Розглянемо бездротові технології для застосувань Echo-систем. Інтернет речей включає в себе широкий спектр бездротових технологій у нескінченному діапазоні застосувань. Мільярди пристроїв і незліченна кількість продуктів і послуг можуть бути підключені до Інтернету, однак мало згадується про технології, що лежать в їх основі, і про те, чому одна з них краща за іншу для будь-якої конкретної програми IoT. Пояснення того, чому існує так багато варіантів IoT, полягає в тому, що існує багато галузевих програм, оскільки діапазон вимог до застосування варіюється від однієї доменної області до іншої. Для того, щоб вибрати правильну технологію, необхідно враховувати певні вимоги. Енергоспоживання, дальність і близькість, кількість пристроїв, розмір повідомлення, чутливість до часу, безпека і, звичайно ж, вартість. Існує ряд бездротових технологій. Вимога до особливостей може стати відповіддю на вибір правильної бездротової технології для бажаного застосування IoT. Відповідна технологія для кожного застосування Echo-системи базується на їх технічних специфікаціях і вимогах до IoT. Наприклад, якщо для програми потрібна вища швидкість передачі даних, можна використовувати IoT. Якщо вартість є пріоритетом, а програма не вимагає високої швидкості передачі даних, то добре підходить інша технологія, оскільки пропонує приватну мережу без необхідності

залежати від провайдера. Інші технології обслужать нижчу вартість пристрою, забезпечать дуже великий радіус дії, нечасту швидкість зв'язку і дуже тривалий час автономної роботи. Одна з відмінностей між різними технолгіями полягає в тому, що що вони базуються на використанні різних діапазонів частот. Широкосмугове рішення схильне до перешкод з боку інших технологій. Односторонній зв'язок і низька швидкість передачі даних обмежують сферу можливих застосувань. І навпаки, він має дуже низьке енергоспоживання з низькою швидкістю передачі даних, але вартість пристроїв може становити проблему. Деякі технології вимагають щомісячної плати залежно від використання даних і кількості пристроїв. З іншого боку, для технологій малого радіусу дії, вони націлені на зовсім інший набір застосунків. Технологія підтримує блискавичне з'єднання, передачу коротких серій крихітних пакетів даних з подальшим блискавичним відключенням. Багато застосунків IoT стали дуже популярними. Ідея мережі була дуже популярна протягом останніх років, хоча багато компаній взяли її на озброєння, тому що не було іншого вибору. Багато радіотехнологій малої потужності, що виходять сьогодні, розроблені спеціально для IoT, і для багатьох застосувань сітчаста топологія не є найкращим вибором. Однак він вимагає менше енергії для покриття того ж діапазону. Стільниковий зв'язок може бути хорошим потенціалом для інтелектуальних вимірювань, основна перевага стільникової технології полягає в тому, що інфраструктура вже є, вона може бути дуже енергоефективною та має швидку швидкість передачі даних. Проте занепокоєння викликає доступність технології. Деякі з них є багатообіцяючими, але реальних розгортань немає. Він також має великий час автономної роботи, глибоке проникнення в будівлі та підземне поширення. Крім того, пристрої коштують недорого і не можуть мати виділену мережеву інфраструктуру. Безпека також є важливим елементом, який необхідно враховувати, оскільки бездротові технології може отримати будь-хто. Вимоги безпеки ґрунтуються на відповідях на наступні питання. Наскільки конфіденційними є дані, що передаються Наскільки він повинен бути безпечним. Які авторизації та механізми аутентифікації існують. І, нарешті, які механізми шифрування використовуються під час передачі даних, що гарантує цілісність даних. Бездротові технології вразливі до різних типів атак, таких як

прослуховування, відмова в обслуговуванні, зловмисник посередині, спуфінг тощо. Для кожної технології існує сертифікація, щоб гарантувати, що їхні пристрої/рішення сумісні з іншими пристроями в системі Echo.

Проаналізуємо бездротові технології для інтелектуального вимірювання. Технології прокладають шлях для формування ініціатив у сфері чистої енергії, включаючи розумні мережі. У міру того, як ця ефективна інтелектуальна доставка енергії буде розвиватися, користувачі та комунальні служби увійдуть у модель двостороннього зв'язку, яка дозволить розумним лічильникам надавати дані про споживання енергії в режимі реального часу безпосередньо користувачеві для актуального моніторингу. Наступним кроком у цій ініціативі щодо чистої енергії є виявлення важливості та збільшення використання бездротових технологій, які дозволять користувачам віддалено контролювати та контролювати використання енергії. Для досягнення наступного кроку багато технологій спрямовані на розумне управління енергією. Деякі з них все ще знаходяться на стадії розробки, інші знаходяться в процесі розгортання. Що допомагає, так це переваги енергоефективності, пропускної здатності та затримки. Інтелектуальні вимірювання спрямовані на зменшення споживання енергії та витрат. Виробники оригінального обладнання беруть участь у розгортанні інфраструктури телеметрії по всьому світу. Це використовується комунальними службами в житлових, комерційних і промислових сценаріях. Існує багато різних підходів до підключення інтелектуальних лічильників енергії для створення «розумних» мереж, і бездротова технологія є ключовим варіантом дизайну. Використання бездротового каналу зв'язку може спростити встановлення та зіставлення даних з інтелектуального лічильника назад до концентратора вдома або в вулиці, але такі посилання повинні бути економічно ефективними, а також безпечними. Це має значний вплив на вибір з безлічі різних частот, протоколів і топологій, доступних для забезпечення зв'язку з інтелектуальним лічильником. Комунальним підприємствам потрібні надійні та гнучкі системи зв'язку, щоб відповідати різноманітним вимогам до мережі на всій території обслуговування. Зв'язок є центральним елементом базової концепції розумної мережі, але не існує «універсальної» технології зв'язку, варіанти включають приватні радіочастотні

сітчасті рішення, які підключають лічильники через концентратор; зв'язок «точка-точка» з окремими операторами за допомогою стільникових мереж загального користування, які також забезпечують зворотний зв'язок для мереж. Фактично, більшість комунальних підприємств сьогодні використовують загальнодоступні бездротові мережі, тобто мережі для передачі інформації про вимірювання, зібраної на рівні мережі району, до центрального місця. Сьогодні уявлення про використання стільникових технологій аж до лічильників починають змінюватися, оскільки автомобілісти тісно співпрацюють з операторами та постачальниками, щоб запровадити більш привабливі тарифні плани та продемонструвати, що вони можуть відповідати суворим вимогам комунальних служб. Використання технології розширеного спектру дозволяє уникнути проблем у шумному електромагнітному середовищі. Сітчасті роботи можуть надати значні переваги для проектування та впровадження бездротових вузлів для інтелектуальних лічильників. Зменшення споживання енергії для подовження циклу заміни акумуляторів якомога довше або навіть повна відмова від нього за рахунок використання збору енергії може забезпечити економічні вигоди, оскільки оператори впроваджують системи інтелектуальних мереж. Вибір частоти залежить від вимог до діапазону, але багато трансиверів забезпечують значну гнучкість для підтримки широкого діапазону діапазонів в одній конструкції без потреби в застосункових зовнішніх компонентах. Ще одним важливим аспектом конструкції розумного лічильника є безпека.

Безпека в IoT – це простір завдань, що швидко розвивається, і складні IT-мережі, які розгортають комунальні підприємства, повинні працювати протягом дуже тривалого часу. Тому безпека вимагатиме постійної уваги протягом усього терміну служби мережі.

Розумна мережа в даний час вважається важливим етапом для енергетичних систем, які були вдосконалені завдяки використанню технологій зв'язку та різних діапазонів підключення для використання розумних ресурсів. Технології розумних мереж включають безліч пристроїв IoT, таких як датчики, маршрутизатори, шлюзи та розумні лічильники. Всі ці пристрої полегшують підключення і комунікації та дають можливість споживачам оптимізувати споживання енергії. Розумну електромережу

часто визначають як самодостатню розподілену систему. Промислові організації останнім часом вивчають численні можливості, які пропонує розгортання Інтернету речей у сфері комунальних послуг. IoT має величезний потенціал для підвищення ефективності, покращення використання енергії та забезпечення кращого обслуговування клієнтів. Крім того, технологія зв'язку зробила бездротові, стільникові та бездротові частоти недорогими та простими для впровадження в будь-яку програму інтелектуальних мереж. Розумні лічильники, такі як лічильники води, газу та електроенергії, є одними з основних компонентів, що використовуються в розумних мережах. Розумні лічильники дають чітке розуміння звичок споживання енергії. В цілому, розумні лічильники дозволяють кінцевим споживачам і комунальним підприємствам співпрацювати і мати можливість контролювати щоденне споживання електроенергії, тим самим зменшуючи свої рахунки. Інтелектуальні лічильники забезпечують двосторонній зв'язок між лічильником і збирачем одиниць даних. При впровадженні мережі для системи вимірювання використовуються численні технології, такі як бездротовий зв'язок малого та далекого радіусу дії. Розумні лічильники, такі як лічильники газу та води, живляться від батарейок у бездротовій мережі, що робить моніторинг лічильників складним завданням, оскільки він вимагає роботи протягом багатьох років з обмеженим джерелом енергії. Останніми роками технологія бездротового зв'язку wM-Bus стала одним із найвигідніших бездротових протоколів.

Введемо модель процесу з пристроями та технологіями IoT так:

$$M = \langle P, G, L, S, Pr, Z \rangle, \tag{1.1}$$

де P – множина пристроїв, які задано їх моделями, тобто елементами множини є моделі різних пристроїв, що підтримуються в технологіях IoT; G – граф зв’язків між пристроями, який відображає певну топологію; L – множина логічних функцій в кожному з пристроїв, які задано моделями в множині пристроїв P; S – множина значень, які отримано від пристроїв; Pr – множина протоколів для зв’язку між

пристроями з IoT; Z – тип зв’язку між пристроями.

Пристрої з IoT характеризуються зв’язками між собою, множиною логічних функцій в них для підготовки значення з пристрою, топологією, множиною протоколів для зв’язку між пристроями з IoT, типом зв’язку між пристроями. Окреме місце в цій моделі процесу займає множина значень, які отримуються з пристроїв. Модель процесу M з пристроями та технологіями IoT задамо схемою, яку зображено на рис. 2.1.

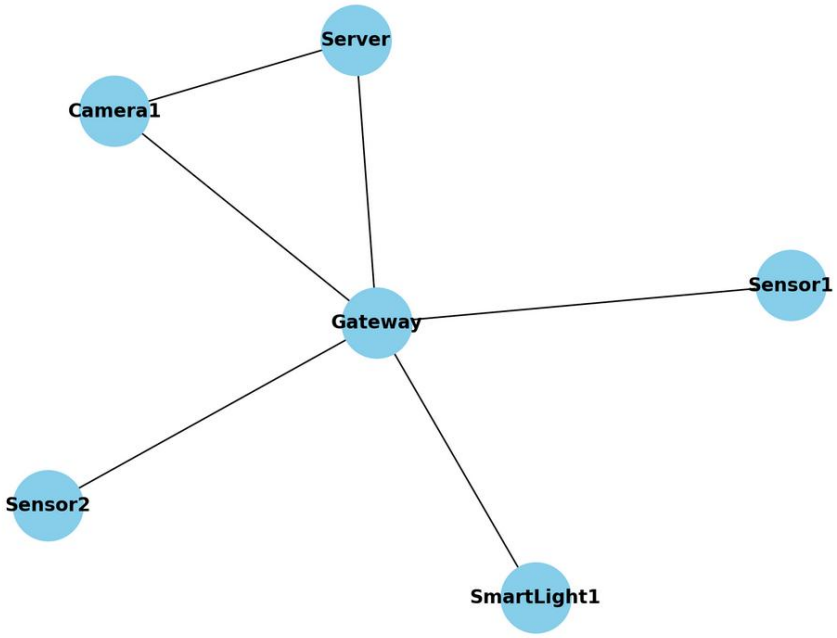


Рисунок 2.1 – Модель процесу з пристроями та технологіями IoT

Таким чином, отримано модель процесу з пристроями та технологіями IoT, яка враховує топологію та протоколи зв’язку, закрема і тип зв’язку. Тоді, ця модель буде потребувати удосконалення в частині розроблення безпеки для протоколу, яким підтримується комунікація між пристроями.

2.2 Модель протоколу обміну повідомленнями між пристроями з підвищеними характеристиками безпеки

Розглянемо опис моделі протоколу обміну повідомленнями між IoT-пристроями з підвищеними характеристиками безпеки. Основні компоненти моделі визначають

особливості процесу з підвищеними характеристиками безпеки в протоколах обміну повідомленнями.

IoT-пристрій (сенсор/актуатор) збирає дані або виконує дії (наприклад, вимірює температуру, керує освітленням). Шифрує дані перед передачею. Використовує сертифікати або токени для автентифікації. Шлюз приймає зашифровані дані від IoT-пристроїв. Проводить перевірку автентичності пристроїв. Перенаправляє дані до хмарної платформи або сервера. Хмарна платформа виконує обробку та зберігання даних. Використовує шифрування для зберігання даних у базах даних. Контролює доступ до пристроїв та даних через політики. Клієнт (наприклад, мобільний додаток) надає інтерфейс користувачеві для перегляду даних та управління пристроями. Підключається до хмари через захищений канал (наприклад, HTTPS).

Основні етапи обміну повідомленнями. Ініціалізація (автентифікація та встановлення сесії). IoT-пристрій генерує запит на підключення до шлюзу. Шлюз запитує сертифікат пристрою або автентифікує його через токен. Встановлюється захищений канал зв'язку (наприклад, через протоколи TLS/DTLS).

Передача даних. IoT-пристрій шифрує дані (наприклад, за допомогою AES-128). Дані передаються через шлюз до хмарної платформи. Хмара зберігає дані в зашифрованій формі.

Команди управління (від клієнта до пристрою). Користувач у мобільному застосунку відправляє команду до хмари. Хмара шифрує команду та передає її шлюзу. Шлюз передає команду IoT-пристрою після перевірки цілісності.

Використані протоколи для безпеки. Захищені транспортні протоколи TLS/DTLS для шифрування трафіку. Протоколи автентифікації OAuth 2.0 або протоколи сертифікації на основі PKI. Протоколи обміну повідомленнями MQTT з TLS для передачі даних у малопотужних мережах. Для низькошвидкісних пристроїв. Протокол wM-Bus для збору даних із лічильників.

Застосункові заходи безпеки включають регулярне оновлення прошивки для усунення вразливостей, виявлення аномалій у трафіку (IDS/IPS). Політики контролю доступу для мінімізації впливу компрометованих пристроїв.

Тому, розглянемо протокол Wireless Meter Bus (wM-Bus) для вирішення завдання

покращення безпеки в комунікаціях між пристроями з IoT. Це відкритий стандарт, створений для інтелектуальних вимірювань і застосунків і сформований групою Open Metering Systems. Його використання стрімко поширюється для обліку електроенергії, газу, води та тепла. Мережа wM-Bus базується на мережі топології «зірка» з ведучими та підлеглими пристроями. wM-Bus можна віднести до категорії коротких або середніх у діапазоні технологій бездротового зв'язку. Протокол wM-Bus стандартизований (EN 13757-4) і працює в неліцензованих діапазонах частот 169, 433 і 868 МГц. Стек протоколів wM-Bus складається з фізичного рівня, рівня каналу передачі даних, рівня розширених каналів передачі даних, транспортного рівня та прикладного рівня. На прикладному рівні третя частина описує стандартизований протокол застосування для забезпечення співпраці з кількома постачальниками.

Таким чином, пристрої різних виробників можуть бути об'єднані в єдину систему. Стандарт включає фізичний рівень, а також канальний рівень для використання бездротових пристроїв, і він відповідає спеціальним режимам зв'язку wM-Bus. Стаціонарний режим (S) - передача між лічильником і збирачем даних, в якому лічильник надсилає дані кілька разів на день. У режимі S1 збирач даних економить енергію перед передачею даних. У S2 передавач вимагає підтвердження. S1-m - це те ж саме, що і S1, але збирачем даних є мобільний приймач. Режим частої передачі (T) – це коли пристрої лічильника надсилають дані колекторам через налаштований інтервал часу, наприклад, невелику кількість передач за секунду або хвилину. Передавач вимагає АКК після від колекторів. Режим частого прийому (R) – це коли лічильник чекає запиту від колектора, перш ніж передати будь-які дані. Зазвичай лічильник знаходиться в режимі енергозбереження і прокидається через задані проміжки часу. R2 регулярно прослуховує повідомлення про пробудження від приймача-передавача. У них є кілька секунд на спілкування після отримання повідомлення про пробудження. Вузькосмуговий режим (N) розглядаються як ретранслятори з декількома стрибками використовуються для зв'язку на великих відстанях. Він призначений для передачі в низькочастотному вузькому діапазоні, а також для зв'язку на далеких відстанях, включаючи односторонній, двосторонній і прямий зв'язок. N1 - це одностороння передача. Вузол регулярно передає сигнал на

нерухоме місце прийому, що дозволяє використовувати однострибкові ретранслятори. N2 - двостороння передача. Вузол передає те ж, що і N1. Протягом короткого періоду приймач залишається активним після кожної передачі, а потім деактивується, коли виявляється правильна преамбула та стан синхронізації. Режим частої передачі та прийому (F) призначений для зв'язку на великих відстанях і поділяється на односторонній та двосторонній підрежими. wM-Bus складається з двох режимів зв'язку: односпрямованого і двонаправленого. Односпрямований режим підтримує тільки передачу даних від лічильника до збирача даних. Поодинокі пристрої реалізують цей режим з низькими накладними витратами, що робить його вигідним. Лічильник передає тільки дані, а збирачі даних отримують тільки дані. При цьому двонаправлений режим підтримує зв'язок від збирача даних до приладу обліку. Колектори даних підтримують тільки двонаправлені режими, які можуть запитувати дані з двонаправлених вимірювальних пристроїв. Збирач даних надсилає колектору запит на отримання даних другого користувача. Прилад обліку отримує запит і відповідає повідомленням «Відповідь на дані другого користувача», що містить інформацію, пов'язану зі збирачем даних. Лічильник буде повторювати спрацьовування до тих пір, поки колектор не припинить зв'язок або не виникне тайм-аут. Дані wM-Bus-повідомлення в пакеті wM-Bus складаються з декількох блоків. Він резервує байти для першого блоку, а наступні блоки вміщують задану кількість байт. Для кожного переданого блоку існує контрольна сума. Багатобайтові поля передають спочатку найменш значущі байти, за винятком полів CRC, які передають найбільш значущі байти першими.

Графічна схема комунікації між пристроями IoT за протоколом Wireless M-Bus зображена на рис. 2.2.

Базова архітектура протоколу Wireless M-Bus (wM-Bus) включає ключові компоненти та функціональні модулі, які забезпечують збір, передачу та обробку даних у системах IoT. Розглянемо детальний опис загальної архітектури з рис. 2.2.

Основні компоненти архітектури wM-Bus. Кінцеві пристрої (End Devices) – це лічильники води, газу, електроенергії або інші сенсори. обладнані радіомодулями, які підтримують протокол wM-Bus. Основні функції: збір даних (наприклад, споживання

ресурсів); періодична передача даних у зашифрованому вигляді; шлюз або концентратор (Gateway/Data Concentrator) приймає дані від кінцевих пристроїв. Він виконує роль проміжної ланки між кінцевими пристроями та хмарними сервісами.

Рисунок 2.2 - Графічна схема комунікації між пристроями IoT за протоколом Wireless M-Bus

Його основні функції: декодування повідомлень wM-Bus; агрегація даних від кількох пристроїв; захищене передавання даних у хмару або сервер. Сервер обробки даних (Data Processing Server) зазвичай розташований у хмарі або на локальному сервері. Його основні функції: обробка, аналіз і зберігання даних; надання доступу до даних через API або інтерфейси. Користувацькі інтерфейси (User Interfaces) – це веб-застосунки або мобільні програми для доступу до даних. Вони дозволяють переглядати інформацію в реальному часі, налаштовувати пристрої та отримувати звіти. Моделі зв'язку в wM-Bus: точка-точка (Point-to-Point), тобто пряма передача даних між пристроєм і концентратором, яка використовується для невеликих систем; мережа типу "зірка" (Star Network), тобто кілька пристроїв під’єднані до одного

концентратору і є типовою архітектурою для великих розподілених систем; транзитна передача (Repeater Mode), коли в окремих сценаріях використовуються повторювачі для розширення радіусів дії. Протокольні рівні wM-Bus: фізичний рівень (Physical Layer) використовує бездротові частоти ISM-діапазону (868/915 МГц) і підтримує кілька типів модуляції (FSK, PSK); канальний рівень (Data Link Layer) відповідає за виявлення та виправлення помилок і забезпечує шифрування та автентифікацію; транспортний рівень (Transport Layer) забезпечує доставку повідомлень між пристроями; прикладний рівень (Application Layer) обробляє дані, передані з пристроїв (наприклад, значення лічильників), який підтримує формати, зручні для хмарних сервісів. Переваги архітектури wM-Bus: енергоефективність (низьке споживання енергії підходить для батарейних пристроїв); безпека (вбудоване шифрування та автентифікація); масштабованість (підтримка тисяч пристроїв в одній системі); сумісність (підтримка стандартів EN 13757 для роботи з різними типами лічильників).

Базова загальна архітектура wM-Bus складається з боку датчика та шлюзу. Архітектура моделі охоплює будівлі в мікрорайоні, створюючи бездротову мережу, в якій вимірювальні пристрої вважаються вузлами мережевих датчиків. Зв'язок між лічильниками та шлюзом/збирачем даних здійснюється за протоколом wM-Bus. Потім зв'язок між шлюзом і внутрішніми службами, такими як постачальники комунальних послуг, здійснюється через Інтернет. Загальна архітектура комунікації між пристроями IoT за протоколом Wireless M-Bus зображена на рис. 2.3.

Головна слабкість wM-Bus полягає не в протоколі чи стандартних специфікаціях. Замість цього - це реалізація безпеки, якщо така взагалі існує. Багато компаній з управління нерухомістю/комунальних підприємств схиляються до використання незашифрованих даних лічильників або одного ключа шифрування для всіх своїх лічильників, щоб заощадити зусилля під час розгортання та зменшити вартість обчислень, і знизити трудомісткість безпеки. Таким чином, це зробить лічильник вразливим до атак. Загальні проблеми включають короткий розмір ключа, який становить 64 біти, виявлення нульового споживання, відкритий текст, який показує розкриття інформації, включаючи ключ, і атаки типу "зловмисник посередині".

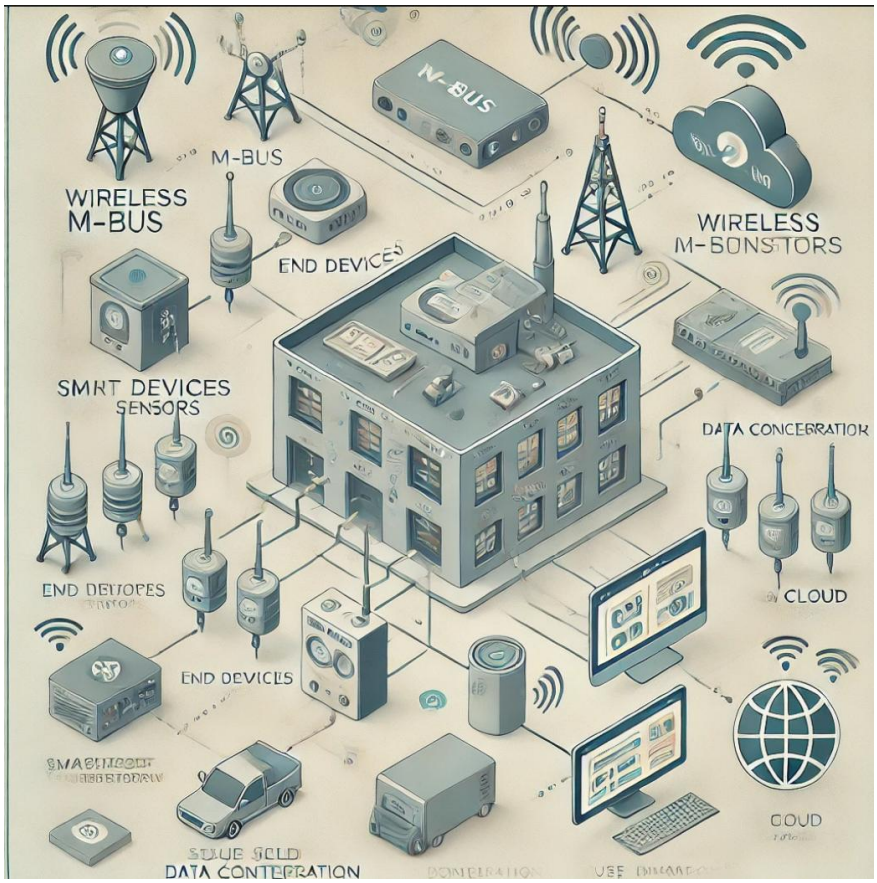


Рисунок 2.3 - Графічна схема комунікації між пристроями IoT за протоколом

Однак остання специфікація представила розширення безпеки, яке дозволяє використовувати застосункові режими шифрування 5, 7, 8, 9 і 10, покладаючись на AES-128 з використанням динамічних ключів, збереженням цілісності та отриманням ключа з рівня аутентифікації та фрагментації. Схема протоколу взаємодії між пристроями в IoT в частині забезпечення безпеки зображена на рис. 2.4.

Запропонована структура безпеки wM-Bus побудована на основі стандартів безпеки. Крім того, це має бути легкий протокол, щоб враховувати час автономної роботи лічильника. Фреймворк буде розглядати різні рівні стеку wM-Bus. Архітектура запропонованого рішення складається з двох компонентів: Meter і DUC/Gateway. Він повинен забезпечити зв'язок між місцевим лічильником і шлюзом у межах місцевої мережі, яка може бути, наприклад, як житловий будинок. Кожен блок має лічильник і один шлюз для зв'язку з кожним лічильником для збору інформації про споживання лічильників. Шлюз може бути збоку від лічильників або

краю хмари. Основними цілями запропонованої моделі безпеки є забезпечення цілісності та приватності/конфіденційності. Цілісність передбачає збереження точності даних, щоб гарантувати, що дані не можуть бути змінені несанкціонованим доступом під час передачі. У той же час, приватність/конфіденційність запобігає несанкціонованому доступу до конфіденційної інформації.

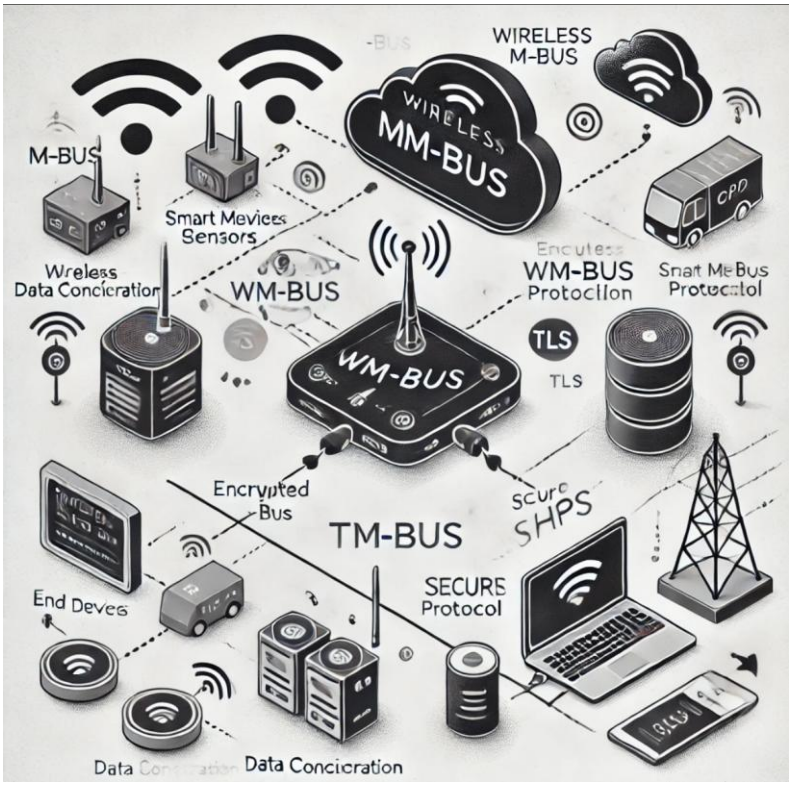


Рисунок 2.4 - Схема комунікації між пристроями IoT за протоколом в частині забезпечення безпеки

Цілісність та приватність/конфіденційність часто досягаються за допомогою аутентифікації, шифрування даних та використання динамічних ключів (публічних та приватних). Тому для досягнення всіх безпекових аспектів у межах запропонованої моделі потрібно переконатися, що всі конфігурації безпеки встановлені з кожного боку лічильнику та шлюзу, як показано на рис. 2.4. Він обчислює алгоритм аутентифікації повідомлень на основі блокового шифру. Він визначає кілька профілів безпеки, які в даний час використовують. Крім того, аутентифікація повідомлень виконується на зашифрованому тексті. Перевірка та дешифрування автентифікації відбуваються паралельно з міркувань продуктивності в більшості реалізацій. Залежно

від режиму зв'язку, встановленого між лічильником і шлюзом, дані передаватимуться через односторонній зв'язок. Як тільки почнеться початковий зв'язок, лічильник ініціює стан пробудження на основі попередньо налаштованого інтервалу. Лічильник надішле параметри, необхідні для безпечного зв'язку зі шлюзом. Під час початкового зв'язку обидва повинні мати безпечну передачу. Їм не потрібно узгоджувати набір шифрів або алгоритм шифрування/дешифрування, оскільки він підтримує передові стандарти шифрування. Кожне повідомлення може бути зашифроване за допомогою застосункового спільного секретного ключа та алгоритму асиметричного шифрування. Той, що знаходиться над стороною лічильника, обчислює ключ майстер-сеансу та виконує взаємну автентифікацію зі шлюзом. Під час обміну ключами може бути виконана взаємна аутентифікація, щоб переконатися, що запропонована структура є легкою. Щоб побудувати легкий протокол, потрібно знизити вартість електроенергії на стороні лічильника. Оскільки лічильник і шлюз налаштовані на пакет безпеки, узгодження параметрів безпеки буде виключено. Рукостискання буде складатися всього з двох основних завдань: встановлення секретного ключа для симетричного шифрування та аутентифікації за допомогою асиметричного ключа і спільне використання секретного ключа за допомогою еліптичної кривої криптографії. Процес передаватиме зашифровані повідомлення, оскільки коли вони приймаються шлюзом, то він може розшифруватися за допомогою симетричного секретного ключа та перевірити цілісність повідомлення. Повідомлення міститиме незашифровані дані, які додає модель лічильника, такі як ідентифікатор виробника, адресу та ідентифікаційний номер. Ці додані дані надходитимуть до алгоритму як частина процесу шифрування та дешифрування, який називається автентифікованим шифруванням із пов'язаними даними. Що стосується функції виведення ключів (KDF), шифрування покладається на ефемерний ключ, який використовує лише одне повідомлення. KDF базується на CMAC.

Запропонована структура передбачає впровадження нового профілю безпеки зі збереженням полегшеного протоколу для економії часу автономної роботи лічильника. Запропонована структура враховує такі фактори безпеки, як цілісність та конфіденційність/приватність. Лічильники та шлюзи з протоколом wM-Bus повинні

бути налаштовані для забезпечення функцій безпеки. Безпека визначається тим, який режим охорони використовується. Її слід використовувати для підтримки аутентифікації, оскільки він допомагає підтримувати довжину кадру повідомлення на короткому кадровому каналі передачі даних. Аутентифікація виконується шляхом додавання кодів автентифікації хешованих повідомлень за допомогою відповідного алгоритму. Для цього потрібно використовувати функцію виведення ключів, що означає, що ніколи не використовуємо один і той же ключ двічі. Лічильник повідомлень отримує унікальне значення лічильника для початкового вектору шифрування Під час рукостискання слід виключити частину переговорів щодо підтримки вартості лічильника при споживанні електроенергії. Це пов'язано з попередньою конфігурацією безпеки.

Таким чином, нова структура передбачає інтеграцію сучасного профілю безпеки, який зберігає спрощений протокол для зменшення енергоспоживання та продовження автономної роботи лічильників. У цій структурі враховуються ключові аспекти безпеки: забезпечення цілісності та конфіденційності даних. Для коректної роботи функцій захисту лічильники та шлюзи, що працюють за протоколом wM-Bus, мають бути належним чином налаштовані. Рівень безпеки системи визначається обраним режимом захисту, який також відповідає за аутентифікацію. Аутентифікація дозволяє підтримувати оптимальну довжину кадру на каналі передачі даних завдяки додаванню кодів хешованих повідомлень, створених відповідним алгоритмом. Ключі шифрування генеруються унікально для кожної сесії, завдяки чому один і той самий ключ не використовується повторно. Лічильник повідомлень генерує унікальний початковий вектор шифрування для кожної передачі даних. Процес встановлення з’єднання («рукостискання») оптимізовано за рахунок виключення частини переговорів. Це стало можливим завдяки попередній конфігурації безпеки, що дозволяє знизити витрати енергії лічильника при обміні інформацією.

2.3 Висновки до другого розділу

Таким чином, отримано модель процесу з пристроями та технологіями IoT, яка

враховує топологію та протоколи зв’язку, закрема і тип зв’язку. Тоді, ця модель буде потребувати удосконалення в частині розроблення безпеки для протоколу, яким підтримується комунікація між пристроями. Також, розроблена нова структура безпеки протоколу передбачає інтеграцію сучасного профілю безпеки, який зберігає спрощений протокол для зменшення енергоспоживання та продовження автономної роботи лічильників. У цій структурі враховуються ключові аспекти безпеки: забезпечення цілісності та конфіденційності даних. Для коректної роботи функцій захисту лічильники та шлюзи, що працюють за протоколом wM-Bus, мають бути належним чином налаштовані. Рівень безпеки системи визначається обраним режимом захисту, який також відповідає за аутентифікацію. Аутентифікація дозволяє підтримувати оптимальну довжину кадру на каналі передачі даних завдяки додаванню кодів хешованих повідомлень, створених відповідним алгоритмом. Ключі шифрування генеруються унікально для кожної сесії, завдяки чому один і той самий ключ не використовується повторно. Лічильник повідомлень генерує унікальний початковий вектор шифрування для кожної передачі даних. Процес встановлення з’єднання («рукостискання») оптимізовано за рахунок виключення частини переговорів. Такого результату досягнуто завдяки попередній конфігурації безпеки, що дозволяє знизити витрати енергії лічильника при обміні інформацією.

Потребують деталізації кроки з безпосередньої деталізації процесів забезпечення безпеки в комунікації пристроїв з IoT.

3 МЕТОД ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНУВАННЯ СИСТЕМ З ІОТ НА ОСНОВІ ЗАХИЩЕНИХ ПРОТОКОЛІВ

3.1 Основи методу забезпечення функціонування систем з IoT на основі захищених протоколів

Метод забезпечення функціонування систем з Інтернету речей (IoT) на основі захищених протоколів є ключовим для підтримки безпеки, конфіденційності та стійкості системи. Основи цього методу включають такі кроки:

1. Аналіз загроз і вимог до безпеки.
2. Використання захищених протоколів зв'язку.
3. Аутентифікація та управління ідентифікацією.
4. Контроль доступу.
5. Захищене зберігання даних.
6. Моніторинг та оновлення.
7. Використання криптографії.
8. Підтримка стійкості до атак.
9. Використання стандартів безпеки.
10. Навчання та інформування користувачів.

Розглянемо їх детальніше.

Аналіз загроз і вимог до безпеки включає: ідентифікацію активів, тобто визначення пристроїв, даних і взаємодій, які потребують захисту; оцінювання загроз, тобто аналіз можливих атак, таких як перехоплення даних, підміна ідентичності, DDoS-атаки; визначення рівня ризиків, тобто розподіл пріоритетів між ризиками на основі їх потенційного впливу.

Використання захищених протоколів зв'язку включає: забезпечення шифрування даних у каналі зв'язку між пристроями IoT з використанням, наприклад, TLS/SSL (Transport Layer Security)**;** шифрування для UDP-протоколів, часто використовуваних у системах IoT з обмеженими ресурсами, зокрема з використанням, наприклад, DTLS (Datagram Transport Layer Security)**;** забезпечення захищеного обміну даними між брокерами і пристроями з використанням, наприклад, легкий протокол для пристроїв з низькою потужністю, наприклад, MQTT з TLS**,** CoAP (Constrained Application

Protocol) з DTLS; для веб-запитів у системах IoT, наприклад, використання HTTPS**.**

Аутентифікація та управління ідентифікацією включає такі завдання: використання для автентифікації пристроїв, наприклад, сертифікатів X.509**;** для абезпечення контрольованого доступу, наприклад, OAuth 2.0; легкий механізм для передачі даних аутентифікації JWT (JSON Web Token).

Контроль доступу передбачає такі кроки: доступ на основі ролей користувачів і пристроїв RBAC (Role-Based Access Control); доступ на основі атрибутів (наприклад, час доби, геолокація) ABAC (Attribute-Based Access Control).

Захищене зберігання даних може бути здійснене так: локальне шифрування даних на пристроях IoT; використання захищених баз даних і хмарних сховищ.

Моніторинг та оновлення включають такі завдання: безпечне оновлення програмного забезпечення OTA (Over-the-Air Updates); системи моніторингу для виявлення аномалій у трафіку або поведінці пристроїв.

Використання криптографії для таких завдсань: иметричне шифрування (наприклад, AES) для швидкої передачі даних; асиметричне шифрування (наприклад, RSA, ECC) для аутентифікації; гешування (наприклад, SHA-256) для перевірки цілісності даних.

Підтримка стійкості до атак може бути забезпечена такими засобами: вбудовані механізми виявлення та запобігання DoS/DDoS-атак; резервні копії даних та аварійне відновлення.

Використання стандартів безпеки включає такі кроки: управління інформаційною безпекою за стандартом ISO/IEC 27001; рекомендації для побудови безпечних систем IoT з використанням NIST Cybersecurity Framework; практичні рекомендації для розробників і впроваджувачів IoT з використанням IoT Security Foundation Guidelines.

Навчання та інформування користувачів включає: навчання користувачів правильному управлінню паролями та оновленням; інформування про можливі загрози та дії у разі інцидентів.

Ці основи забезпечують всебічний підхід до створення безпечних IoT-систем і захищають їх від основних кіберзагроз.

Розглянемо безпосередньо протокол із захищеною спроможністю. Необхідно розробити нову архітектуру моделі, в якій можливе включення протоколу wM-Bus, зокрема через використання режиму, який служить основним каналом для обміну даними між лічильником і шлюзом. Ця структура ще більше зміцнюється, що підвищує як ефективність, так і безпеку передачі даних від потенційних загроз. Сторона лічильнику з датчиком води використовує радіочастотний модуль, що підкреслює його здатність до широкого діапазону та проникнення і є критично важливим атрибутом для вимірювання комунальних послуг у різних умовах. Ця передова інтеграція становить основу безпечної та ефективної системи зв'язку. Модель ефективності відповідає специфікаціям пакетних кадрів протоколу wM-Bus. Конструкція використовує 10-байтовий заголовок для інкапсуляції важливої інформації wM-Bus разом із деталями вимірювального пристрою. Слідуючи за заголовком, протокол виділяє наступний сегмент спеціально для полегшення обміну повідомленнями в рамках NPF, тим самим забезпечуючи безпечний зв'язок рукостискання.

Передача даних відповідно до моделі OSI зображена на рис. 3.1.

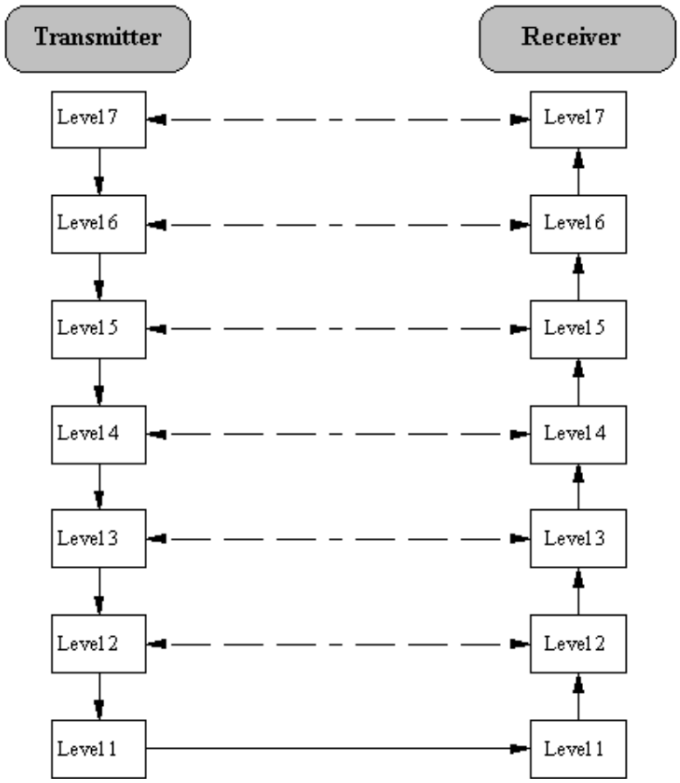


Рисунок 3.1 - Передача даних відповідно до моделі OSI

Цей протокол використовує асинхронну серійну передачу бітів, в якій синхронізація реалізована за допомогою шматочків запуску та зупинки для кожного символу. Не повинно бути пауз у телеграмі, навіть після зупинки. Оскільки спокою на лінії відповідає 1 (позначці), біт старту повинен бути простором, а зупинка біт - позначкою. Між восьми бітами даних та рівним біт паритету передаються, гарантуючи, що принаймні кожен одинадцятий біт є позначкою. Біти даних передаються у порядку висхідного порядку, тобто біт з найменшим значенням (LSB = найменший значущий біт) є першим, який можна знайти на лінії. Передача відбувається в половині дуплексу та зі швидкістю передачі даних не менше 300 од. На рис. 3.2 зображено передачу байту в напрямку відповіді.

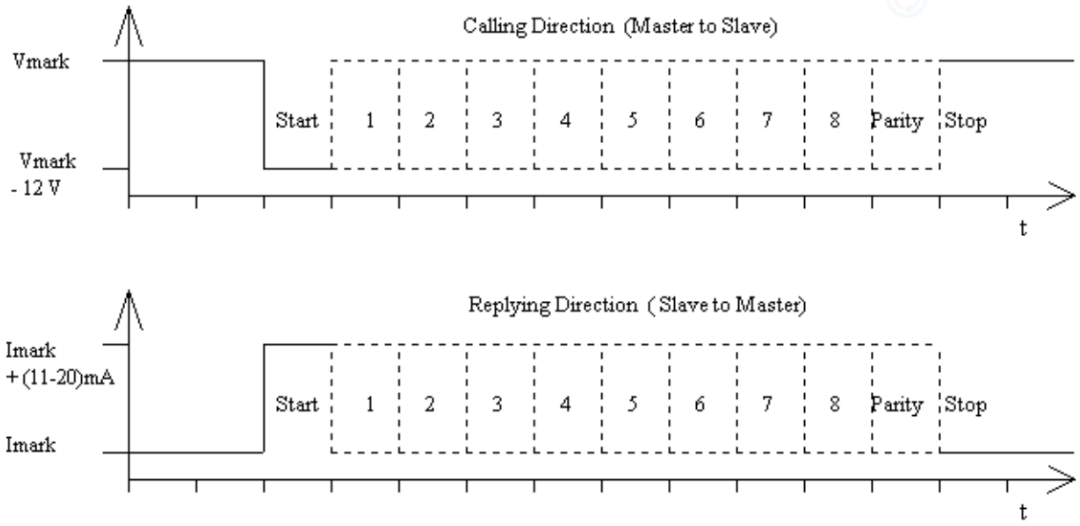


Рисунок 3.2 - Передача даних в напрямку відповіді

Лічильник, який бере на себе роль ініціатора, і шлюз, що діє як відповідач, інтегровані з відповідними моделями wM-Bus. Ця схема підкреслює подвійний акцент на дотриманні встановлених специфікацій протоколу для бездротового зв'язку та вдосконаленні протоколів безпеки за допомогою NPF. Ця інтеграція забезпечує ефективний і безпечний потік даних, дотримуючись специфікацій пакетного кадру wM-Bus, таким чином відповідаючи як стандартам безпеки, так і сумісності, необхідним для застосунків інтелектуального вимірювання в середовищах IoT.

Запропонована модель зв'язку рукостискання, що інтегрує протоколи wM-Bus з

НПФ чітко окреслює процес комунікації за допомогою послідовності чітких кроків, демонструючи безпечний обмін даними. Підготовка до рукостискання. Усі пристрої підтримують NPF на додаток до протоколу wM-Bus. Кожен пристрій генерує статичну пару ключів для використання в рукостисканні NPF XX. Ініціація рукостискання. Ініціатор генерує ефемерну пару ключів і надсилає відкритий ключ відповідачу. Реакція на рукостискання. Отримавши відкритий ключ, відповідач генерує свою власну ефемерну пару ключів. Відповідач шифрує свій ефемерний відкритий ключ за допомогою ефемерного відкритого ключа ініціатора та надсилає його назад разом із зашифрованим статичним відкритим ключем. Завершення рукостискання. Ініціатор розшифровує ефемерні та статичні відкриті ключі відповідача. Потім він шифрує свій статичний відкритий ключ за допомогою ефемерного відкритого ключа відповідача та надсилає його відповідачу. Обидві сторони тепер поділилися своїми ефемерними та статичними публічними ключами, зашифрованими за допомогою отриманих спільних секретів. Безпечне встановлення сеансу. Обидві сторони використовують функції NPF для отримання спільного секрету з обмінюваних ключів. Ця спільна секретна інформація використовується для шифрування та розшифровки подальших повідомлень. Безпечний зв'язок. Встановлена спільна секретна інформація сприяє шифруванню та розшифровці повідомлень відповідно до симетричних схем шифрування NPF. Це гарантує безпечну передачу повідомлень відповідно до специфікацій протоколу wM-Bus, а застосунковий рівень шифрування підвищує безпеку протоколу wM-Bus.

Таким чином, інтегруючи XX шаблон NPF, протокол wM-Bus може досягти більш високого рівня безпеки, забезпечуючи конфіденційність, цілісність і автентичність зв'язку в програмах інтелектуального вимірювання.

Архітектура пристроїв і моделі довіри, також, формують обмеження цього дослідження. Ідентичність, тип і версія кожного пристрою, а також унікальні ключі для лічильників і шлюзів, а також підтримка NPF лежать в основі механізмів безпеки протоколу. Однак ці припущення можуть не справедливі у всіх потенційних сценаріях застосування, що обмежує застосовність. Припущення щодо довіри, пов'язані з обміном ключами, а також можливістю використання довіри при першому

використанні у середовищах, де відсутня інфраструктура для цифрових сертифікатів або попередньо спільних ключів, є значною. Хоча ці припущення підходять для контексту wM-Bus, але вони можуть не бути універсально застосовними або прийнятними, особливо в сценаріях з підвищеними вимогами до безпеки або коли початковий зв'язок може бути під загрозою перехоплення. Крім того, необхідно вирішити занепокоєння щодо сумісності протоколу wM-Bus. Протокол wM-Bus доступний у кількох версіях (наприклад, S, T, C, N), кожна з яких має унікальні специфікації та вимоги. Він має бути адаптований для роботи зі спеціальною версією протоколу wM-Bus, що використовується. Вибір шаблону NPF (наприклад, XX, NK, NX, IK) повинен ґрунтуватися на конкретних вимогах зв'язку wM-Bus, враховуючи відмінні риси та можливості кожного шаблону. Міркування продуктивності мають першорядне значення, оскільки NPF повинен працювати в умовах обмеженої обчислювальної потужності, пам'яті та часу автономної роботи пристроїв wM-Bus, що вимагає оптимізації для низького енергоспоживання та ефективного використання ресурсів.

Розглянемо методологію розроблення протоколу. Використаємо структурований підхід до проектування протоколів, зосереджений на практичній реалізації та оцінці протоколів зв'язку в безпечних середовищах. Це спеціально застосовується до NPF і протоколу wM-Bus в налаштуваннях IoT. Методологія розробки протоколу побудована на окремі етапи: впровадження фреймворку та попередній аналіз, тобто етап встановлення орієнтирів для продуктивності NPF та отримання детального аналізу протоколу wM-Bus, зосереджуючись на тривалості зв'язку, розмірі пакету та використанні пам'яті, що розширює функціональність NPF з різними конфігураціями безпеки, оцінюючи вплив шифрів шифрування, методів обміну ключами та хеш-функцій, оцінює ефективність протоколу wM-Bus у пристроях IoT у змодельованих сценаріях вимірювання; моделювання та інтеграція, що на цьому етапі оновлюється протокол wM-Bus за допомогою TLS для кращої безпеки та оцінюється його продуктивність за стандартними показниками, також динамічно аналізуються допустимі моделі, щоб визначити ті, які оптимізують безпеку, розмір пакету та тривалість рукостискання, прагнучи зрозуміти вплив різних моделей безпеки на

ключові показники ефективності; комплексне тестування та оптимізація, що передбачає завершення повною оцінкою, об'єднуючи всі властивості безпеки, щоб запропонувати функціональний, легкий протокол і цей етап синтезує ідеї всіх попередніх етапів, пропонуючи цілісне уявлення про характеристики продуктивності, сильні та слабкі сторони, що спостерігаються і призводить до пропозиції вдосконаленого, захищеного протоколу.

Розглядатимемо пристрої IoT у мережах wM-Bus, використовуючи такі інструменти, як набір розробників для збору даних про використання пам'яті, розмір пакета, час рукостискання та функції безпеки. Він використовує Noise Explorer для перевірки безпеки NPF для інтеграції. Оцінюватимемо безпеку за моделлю та ефективність і вплив батареї, при цьому оцінка показала, що заходи безпеки скорочують термін служби батареї як компроміс порівняно з незахищеними операціями. Щоб забезпечити глибоку оцінку, використовуватимемо розроблену систему впровадження, що має вирішальне значення для оцінки ефективності та актуальності обраних моделей. Налаштування включатиме в себе збірку апаратної та програмної складових. Використовуватимемо бітову архітектуру. Цей обчислювальний пристрій відігравав важливу роль, забезпечуючи необхідну обчислювальну потужність та інтерфейс для моделювання ролей сервера та клієнта. Щодо програмного забезпечення, то пакет став основою для початкової конфігурації та діагностичної оцінки зв'язку карток wM-Bus. Цей пакет, спеціально розроблений для сімейства wM-Bus, включає в себе такі інструменти, що спрощують етапи розробки і розгортання модулів. Процес інсталяції, що характеризується своєю простотою завдяки інтеграції компонентів, сприяє безперебійному налаштуванню. Конфігурації були встановлені з використанням портів для карт, що оптимізувало налаштування з'єднання і процес обміну даними. Рішення про передачу даних у відкритому вигляді, позбавлене будь-яких заходів безпеки, було прийнято для того, щоб встановити основу для оцінювання зв'язку wM-Bus, заклавши основу для подальших удосконалень з підвищення безпеки.

Таким чином, розроблено основні кроки методу забезпечення безпечного протоколу для комунікації пристроїв з IoT. Встановлено до розгляду IoT-пристрої в

мережах wM-Bus, використовуючи інструменти, такі як набір розробників для аналізу використання пам'яті, розміру пакетів, часу встановлення з'єднання та функцій безпеки. Для перевірки безпеки NPF у контексті інтеграції застосовуватимемо Noise Explorer. Оцінювання безпеки здійснюватиметься на основі моделі, а також аналізуватимемо ефективність і вплив на ресурс батареї.

3.2 Організація протоколів для захисту бездротового зв'язку між пристроями IoT

Розглянемо протоколи у контексті захисту зв'язку з wM-Bus, зосереджуючись на критичних показниках продуктивності, таких як використання пам'яті, розмір пакету та час процесу рукостискання. Діяльність та налаштування фази включає детальний аналіз моделей, кожна з яких має конкретні параметри безпеки з точки зору обміну ключами, механізму аутентифікації, методу шифрування та алгоритму хешування. Ключовими показниками продуктивності є час, витрачений на зв'язок, розмір пакета та споживання меморії, що працює на серверних та клієнтських терміналах. Результатом цього етапу має бути створений базовий рівень показників продуктивності, що дає відомості про те, як кожна модель працює окремо. Цей базовий рівень є основою для наступних фаз проекту, де ці шаблони можуть бути застосунково протестовані та адаптовані в поєднанні з протоколом wM-Bus і порівняні з іншими протоколами безпеки. На другому кроці розглянемо базові показники продуктивності протоколу wM-Bus для встановлення базового орієнтиру для продуктивності протоколу wM-Bus ізольовано. Цей етап має вирішальне значення для розуміння невід'ємних можливостей та обмежень протоколу wM-Bus перед його інтеграцією. Основні заходи та результати цього етапу – це налаштування. Використовуються ті ж ключові метрики: час, витрачений на зв'язок між клієнтом і сервером під час рукостискання; розмір переданих пакетів; загальне споживання пам'яті під час рукостискання. Результатом цього етапу має стати набір базових показників продуктивності для протоколу wM-Bus. Ці показники будуть орієнтиром для наступних етапів проекту, особливо при оцінці необхідності інтеграції і оцінці компромісів у продуктивності, пов'язаних із забезпеченням зв'язку wM-Bus.

Наступним етапом є етап щодо підвищення безпеки протоколу зв'язку wM-Bus шляхом включення нового рівня. Цей етап ґрунтувався на початковій розробці зв'язку wM-Bus, якому раніше не вистачало заходів безпеки. Основними цілями та результатами цього етапу були використані сокети та криптографія. На цьому етапі потрібно реалізувати важливі криптографічні функції, операції шифру для шифрування та дешифрування, хешування та функція виведення ключів. Ці інструменти важливі у створенні надійного каналу зв'язку між клієнтом wM-Bus (лічильником) і сервером (шлюзом). Крім того, потрібно застосувати комплексний підхід шляхом інтеграції різних механізмів обміну ключами з кожним набором шифрів для ретельного вивчення їх властивостей безпеки. На етапі аналізу результатів потрібно зафіксувати та проаналізувати результати впровадження захищеного рівня у протоколі зв'язку wM-Bus. Журнали виконання для кожного набору шифрів нададуть детальну інформацію про процес рукостискання, включаючи такі всі кроки. Ці журнали мають вирішальне значення для оцінки властивостей безпеки, використання пам'яті та витраченого часу для кожної фази зв'язку, що дозволило б детально вивчити вплив кожного набору шифрів на безпеку та продуктивність зв'язку wM-Bus. Фінальною стадією чи етапом має бути інтеграція з технологією wM-Bus для підвищення безпеки зв'язку між лічильниками та шлюзами. Цей етап можна узагальнити в трьох основних аспектах: розробка технології wM-Bus; налаштування; використання.

Розглянемо концептуальні підходи до реалізації. Ця реалізація повинна бути ретельно протестована з використанням послідовних метрик протягом усього процесу, зосереджуючись на використанні пам'яті та часі, необхідному для процесу рукостискання. Це можна дослідити із використанням задокументоваих відомостей для шлюзу та лічильника, а також для всього процесу рукостискання та передачі даних зашифрованого застосунку. Результатом цього етапу стала б детальна оцінка продуктивності та потреб у ресурсах інтегрованої системи, що сприяло точному налаштуванню та оптимізації для розгортання в реальному середовищі. Зокрема, патерн мав значно нижчий загальний обсяг використання пам'яті порівняно з іншими моделями, що робить його потенційно більш придатним для пристроїв з обмеженими

ресурсами пам'яті. З точки зору ефективності часу, патерн може бути визнаний найбільш ефективним у часі, з найкоротшим загальним часом, необхідним для процесу рукостискання, незважаючи на низьке використання пам'яті, зайняв найдовший загальний час, можливо, через застосункові кроки, пов'язані з процесом рукостискання. Оптимізація протоколу для отримання його полегшеної версії. Цей етап спрямований на оптимізацію у бік легкого протоколу, що використовує багато шаблонів і включає всі властивості безпеки, щоб запропонувати найкращий протокол для пристроїв, що живляться від батарейок. Цей етап можна розбити на три основні компоненти: налаштування; використання; реалізація; результат. Функціонування підтримувало послідовне налаштування апаратного забезпечення, використовуючи карти для ретельного моделювання функціональності як лічильника, так і шлюзу. Методологія реалізації включала спеціалізовані бібліотеки для зв'язку wM-Bus. Криптографічні елементи, а також алгоритми хешування, були обрані за їх ефективність і безпеку. Протягом усього експерименту необхідно виконати дослідження з багатьма чіткими шаблонами у поєднанні з конфігураціями безпеки, щоб оцінити їх продуктивність та сумісність. Кожен цикл виконання необхідно методично фіксувати з систематичним документуванням результатів, що забезпечить детальний і доступний запис результатів. Це сприятиме глибокому аналізу та порівнянню різних протестованих конфігурацій безпеки, надаючи всебічний огляд результатів експерименту та висвітлюючи наслідки для безпеки кожного протестованого шаблону та конфігурації. Аналіз результатів має на меті визначити найбільш ефективний і безпечний метод бездротового вимірювання зв'язку з урахуванням таких факторів, як використання пам'яті, розмір пакета та час рукостискання. Аналіз включав порівняння різних моделей, зосереджуючись на використанні ними пам'яті, розмірі пакета та часі рукостискання. Шаблон для категорії трьох повідомлень і шаблон для категорії двох повідомлень можуть виявитись особливо ефективними, пропонуючи низьке використання пам'яті та менші розміри пакетів, що матиме вирішальне значення для пристроїв IoT, які працюють від акумуляторів. Тому, їх необхідно зафіксувати і по ним отримати багатократні результати.

Модель потребує визначення в контексті варіанту, який підходить найкраще. І таку модель для зв’язку між пристроями IoT потрібно підібрати серед розглядуваних. Протокол з такою моделлю забезпечує ефективний баланс між безпекою та ефективністю ресурсів, що зробить його ідеальним для пристроїв IoT, які працюють від акумуляторів. Час рукостискання та передачі буде відносно однаковим у всіх комбінаціях, що вказуватимуть на стабільну продуктивність. Протокол потрібно розгорнути ізольовано для встановлення фундаментальних орієнтирів ефективності, що мають вирішальне значення для подальшого аналізу. Цей етап стане ключовим у закладенні основи для всього вибору варіанту моделі. Кожен шаблон потрібно обрати з унікальною комбінацією функцій безпеки, включаючи механізми обміну ключами, методів шифрування та алгоритмів гешування, тим самим пропонуючи різноманітний спектр конфігурацій безпеки. Основними показниками, які потрібно оцінювати на цьому етапі - це час, витрачений на зв'язок, розмір пакетів і споживання пам'яті. Цей етап повинен бути присвячений не лише збору даних, а й розумінню складного балансу між безпекою та ефективністю, який представляли всі розглядувані моделі. Ретельно аналізуючи ці моделі в контрольованому середовищі, дослідження їх матиме на меті виявити невід'ємні компроміси та синергії, які пропонувала кожна модель. Цей фундаментальний аналіз відіграватиме важливу роль у підготовці основи для наступних етапів, де ці шаблони були б застосунково перевірені в поєднанні з протоколом wM-Bus, таким чином доповнивши дослідження емпіричними даними та висновками, які допоможуть розробити оптимізований протокол безпеки для систем зв'язку з низьким енергоспоживанням.

Проведемо аналіз попередньо визначених шаблонів. Всі патерне ретельно аналізуватимемо, щоб оцінити їх вплив та ефективність. Оцінка має бути організована таким чином, щоб дати уявлення про те, наскільки ефективною та практичною є кожна модель по відношенню до цілей дослідження. Вона повинна представляти основні результати глибокого аналізу шаблонів, тим самим пропонуючи чіткий огляд їх переваг, обмежень і потенційної ролі в підвищенні безпеки та ефективності протоколу.

Відсутність аутентифікації, шифрування та використання для обміну ключами в

бездротовому зв'язку між лічильником і шлюзом через wM-Bus викликає значні зауваження щодо безпеки. Без автентифікації неможливо забезпечити особистість сторін, що спілкуються. Це відкриває спроможності для потенційного приєднання несанкціонованих пристроїв до мережі, видачі себе за законні пристрої або перехоплення та маніпулювання даними. Відсутність шифрування означає, що дані, що передаються між пристроєм і шлюзом, вразливі до прослуховування та фальсифікації. Зловмисники можуть непомітно перехоплювати конфіденційну інформацію, маніпулювати показаннями або впроваджувати шкідливі дані в потік зв'язку. Використання протоколу для обміну ключами є широко прийнятим з алгоритмом обміну ключами на основі еліптичної кривої. Хоча це сильний вибір для встановлення спільної секретності між лічильником і шлюзом, але він не вирішує проблему відсутності аутентифікації та шифрування в загальному обміні даними.

Наслідками для безпеки тоді стане відсутність автентифікації і це допускає потенційну загрозу всієї системи зв'язку. Несанкціонований доступ до мережі може призвести до несанкціонованого доступу до даних, збою в обслуговуванні або ін'єкції неправдивих даних. Відсутність шифрування наражає конфіденційні дані, такі як показання лічильників та інформацію про клієнтів, несанкціонованому доступу або маніпуляціям. Це серйозна проблема конфіденційності, яка може призвести до фінансових або репутаційних збитків. У шаблоні обмін ключами здійснюється за допомогою симетричного спільного ключа за зразком, подібним до базового, без аутентифікації, шифрування, і відсутність гешування викликає деякі конкретні проблемні моменти безпеки. Використання симетричного спільного ключа для обміну ключами є перспективним підходом, коли є потреба в гнучкості і ефективності. Однак безпека значною мірою залежить від захисту та розповсюдження цього спільного ключа. Будь-яка компрометація попередньо наданого ключа може призвести до повного порушення безпеки. Відсутність автентифікації означає відсутність взаємної перевірки сторін, що спілкуються. Без належної автентифікації неавторизовані суб'єкти можуть отримати доступ до мережі, виконувати атаки типу "людина посередині" або видавати себе за законні пристрої. Алгоритм шифрування забезпечує як конфіденційність, так і цілісність, захищаючи дані від прослуховування

та фальсифікації. Це вибір для забезпечення безпеки каналу зв'язку. Відсутність гешування може викликати занепокоєння, особливо якщо існує потреба в перевірці цілісності даних. Хешування зазвичай використовується для того, щоб гарантувати, що отримані дані не були змінені під час передачі. Без гешування немає вбудованого механізму перевірки цілісності обмінюваних повідомлень. Хоча використання цього протоколу забезпечує надійне шифрування, відсутність автентифікації становить значний ризик. Зловмисник може отримати несанкціонований доступ, впровадити шкідливі дані або маніпулювати комунікацією непомітно. Залежність від попередньо спільного ключа для обміну ключами вимагає ретельного управління ключами. Будь-який компроміс щодо ключа може мати серйозні наслідки. Комбінація забезпечує те, що отримані дані не були підроблені та надходять із законного джерела. Відсутність шифрування означає, що дані передаються у вигляді відкритого тексту. Це піддає інформацію прослуховуванню, і зловмисники потенційно можуть отримати доступ до конфіденційних даних. Важливо враховувати чутливість інформації, що передається, і те, чи потрібне шифрування для захисту конфіденційності. Метою гешування в цьому контексті є перевірка цілісності. Гешування гарантує, що дані не були змінені під час передачі. Однак важливо зазначити, що гешування саме по собі не забезпечує конфіденційності. Залежність від попереднього спільного ключа для виведення ключів означає, що безпека системи така ж сильна, як і захист цього ключа. Належні методи керування ключами, включаючи безпечне розповсюдження та періодичне оновлення ключів, є важливими. Використання протоклу забезпечує надійну автентифікацію, але відсутність шифрування означає, що дані розкриваються під час передачі. Це може бути прийнятним для певних випадків використання, коли конфіденційність не є першочерговою проблемою. Вибір протоколу для хешування сприяє цілісності даних, але важливо переконатися, що геш належним чином перевірений з обох сторін, щоб виявити будь-яке втручання.

Схема обміну ключами передбачає, що обидві сторони вносять публічні ключі, і протокол обчислює спільні секрети на основі цих публічних ключів. Зазвичай він не включає підтвердження вбудованого ключа. Ключова відмінність полягає в наявності підтвердження ключа. Він включає спеціальне корисне навантаження для

підтвердження ключів, яке гарантує, що обидві сторони надійно домовляються про одні й ті ж симетричні ключі. З іншого боку, він зазвичай зосереджується на обміні ключами без вбудованого підтвердження. Шаблон призначений для явної обробки підтвердження ключів як частини потоку зв'язку, забезпечуючи застосунковий рівень гарантії безпеки.

Під час інтеграції в архітектуру клієнт/сервер потрібно використати шість різних шаблонів рукостискання, щоб закласти основу для подальшого емпіричного аналізу. Цей аналіз передбачатиме ретельне виконання кожного шаблону для оцінки властивих їм атрибутів безпеки, одночасно відстежуючи споживання пам'яті та розміри пакетів на кожному етапі процесу рукостискання. Для цього потрібно розрізняти дві основні парадигми обміну повідомленнями: послідовність з трьох повідомлень; послідовність з двох повідомлень. Для всього процесу рукостискання використання пам'яті для всіх шаблонів є відносно близьким за діапазоном, при цьому найменше використання пам'яті для шаблону може становити менше порогового значення, а найбільше може бути більше від використовуваної одиниці. Тому, цей шаблон має найвище використання пам'яті. Немає чіткої висхідної або низхідної тенденції у використанні пам'яті за всіма шаблонами. Так чи інакше, моделі виділяються як такі, що мають найвищі вимоги до пам'яті, в той же час виявляються найбільш ефективними з точки зору оперативної пам'яті. Якщо використання пам'яті є критичним фактором для системи, то шаблон може бути кращим. І навпаки, якщо шаблони забезпечують основні функції безпеки, які переважують міркування щодо пам'яті, то їх можна вибрати, незважаючи на більш високе використання пам'яті.

Розглянемо можливість встановлення базового рівня продуктивнсті. На цьому етапі основна мета полягатиме в тому, щоб встановити базовий рівень продуктивності для протоколу wM-Bus відповідно до стандарту протоколу. Ця оцінка охоплюватиме як апаратну, так і програмну складову. Початковий крок вимагатиме налаштування апаратного забезпечення за допомогою двох карт бездротового зв'язку для емуляції ролей шлюзу (серверу) та лічильника (клієнту). Для цього експерименту потрібно буде використати набір розробників. Цей процес буде оптимізований завдяки використанню програми, важливого інструменту в наборі, призначеного для

підтримки, розробки та розгортання модулів. Інструменти wM-Bus охоплюють інструменти, адаптовані для сімейства Wireless MBUS. Безшовна інсталяція, включаючи компоненти .NET і USB Driver, досягається виконанням наданого файлу інсталяції. Реалізована розробка протоколу зв'язку wM-Bus надає уявлення про взаємодію між двома картами, імітуючи ролі лічильника (клієнту) та шлюзу (серверу) у системі. Використовуючи спеціальні бібліотеки wM-Bus, а також інші бібліотеки, для розрахунку використання пам'яті та для підключення двох карт і використовуваних для запису та читання з послідовних пристроїв конфігурація для розробки встановить дані для кожної карти. Таке впровадження спрощує процес встановлення з'єднання та передачі даних. Передача корисного навантаження відбувається у відкритому тексті без будь-яких заходів безпеки. Це має за мету створити основу для зв'язку wM-Bus на стороні шлюзу без рівнів безпеки, прокладаючи шлях для майбутніх порівнянь після інтеграції вимірювань безпеки. Процес передачі даних відбувається за встановленими полями wM-Bus. Протокол wM-Bus охоплює різні типи повідомлень, з конкретними прикладами. Цей комплексний підхід забезпечує ретельне вивчення динаміки зв'язку wM-Bus на стороні шлюзу та лічильника, закладаючи основу для подальших оцінок, посилених заходами безпеки. Під час кожної ітерації система динамічно генеруватиме повідомлення з заданою довжиною. Результати експерименту на стороні шлюзу відображатимуться на трьох діаграмах, кожна з яких ілюструватиме використання пам'яті для різних типів повідомлень. Початкова точка для кожного типу повідомлень демонструє підвищені показники використання пам'яті. Однак після ініціалізації всіх програмних змінних використання пам'яті знижується і стабілізується після того, як довжина повідомлення перевищує невелику кількість байт. Примітно, що для певної довжини повідомлення ці конкретні типи повідомлень демонструють значно вищі вимоги до часу, вносячи цікаву варіацію в загальну динаміку зв'язку. Для забезпечення більш точного зображення представлено вигляд терміналу, який конкретно демонструє довжину останнього повідомлення з типом повідомлення. Цей дисплей містить необроблене повідомлення, а також обчислені показники, такі як використання пам'яті та час. Крім того, видно поля заголовка, такі як версія,

ідентифікатор виробника та довжина корисного навантаження, що дає більш детальне уявлення про зв'язок на стороні шлюзу. Що стосується сторони лічильника, то ілюстровані результати представлені і числа будуть демонструвати варіації використання пам'яті для різних типів повідомлень у протоколі wM-Bus на різній довжині. Очевидно, що тип повідомлень вимагає більшого споживання пам'яті порівняно з двома іншими типами повідомлень.

Здійснимо аналіз шлюзу. Абсолютний час вищий для всіх записів порівняно з лічильником, що може бути пов'язано з тим, що сервер виконує застосункові завдання обробки. Що стосується використання пам'яті, то використання пам'яті здається однаковим для всіх алгоритмів з невеликими відмінностями між записами. Алгоритми можуть показати однакове використання пам'яті, що вказуватиме на те, що з точки зору пам'яті будь-який алгоритм може підійти для серверної частини. Це передбачає різні конфігурації або вимоги до процесу обміну ключів на стороні сервера порівняно з лічильником. Розмір не корелює безпосередньо з часом або використанням пам'яті, що означає, що розмір може не бути визначальним фактором у показниках продуктивності. Перераховані шифри мають ті ж типи, що і раніше, але продуктивність на шлюзі може відрізнятися через різне операційне середовище та завдання, які він обробляє. На стороні серверу, що розглядає використання пам'яті, конфігурації для різних наборів шифрів показують різне використання пам'яті, але в цілому знаходяться в середньому діапазоні спектру використання пам'яті. Конфігурації показують діапазон використання пам'яті, але не завжди займають найвище або найменше використання пам'яті. Конфігурації, як правило, мають менше використання пам'яті в усіх наборах шифрів. Крім того, витрачений час на стороні шлюзу (серверу) з використанням різних наборів шифрів і алгоритмів обміну ключами. Майже у всіх випадках алгоритм показуватиме трохи менший витрачений час порівняно з нетиповим підходом, що свідчить про те, що він швидший для більшості операцій. Він демонструє кращу продуктивність (менший витрачений час) для різних наборів шифрів і типів повідомлень на стороні шлюзу. Час проміжку часу для повідомлень зазвичай вищий, ніж для нетипових повідомлень. Виходячи з аналізу сторони шлюзу зв'язку wM-Bus, алгоритм обміну ключами пропонує дещо кращу

продуктивність з точки зору витраченого часу для різних наборів шифрів і типів повідомлень. Ця стабільна перевага в продуктивності може бути фактором при виборі для цілей ефективності в цьому конкретному контексті.

Розглянемо загальний огляд процесу рукостискання між лічильником і шлюзом разом узятих, а також цифри загального використання пам'яті. Крім того, використовуваний шифр використовує трохи менше пам'яті у більшості записів, але знову ж таки, різниця не суттєва. Він працює швидше, але відмінності між ним і рештою шифрів мінімальні, що свідчить про схожі характеристики продуктивності для ключових блоків. Розмір пакета здається однаковим для кожного набору шифрів для обох алгоритмів. Однак розміри пакетів знаходяться в близькому діапазоні. Обмін ключами має тенденцію до дещо більших розмірів пакетів у всіх наборах шифрів і типах. Набір шифрів зазвичай призводить до найбільших розмірів пакетів, що очікується, оскільки він використовує більший ключ, що потенційно може призвести до більших повідомлень про рукостискання. В даний час не існує значної дисперсії в розмірі пакета тільки на основі типу повідомлення M-Bus, оскільки на варіацію в більшій мірі впливає алгоритм обміну ключами і набір шифрів. Тип повідомлень незмінно має найменший розмір пакету серед різних обмінів ключами та наборами шифрів. З цього випливає, що якщо мінімізація розміру пакета має вирішальне значення, наприклад, в середовищах з обмеженою пропускною здатністю, то обмін ключами може бути кращим. Більші розміри пакетів можуть бути виправдані підвищеною безпекою, що забезпечується шифром або обміном ключами, але конкретні потреби в безпеці доведеться зважити з вартістю збільшення розміру передачі даних.

Розглянемо розроблення технології wM-Bus інтегрувавши її з NPF. Ця інтеграція дозволить впровадити та оцінити шість різних попередньо визначених моделей зв'язку. Щоб досягти цього, використаємо потужність і гнучкість середовища програмування, розробляючи код, який використовував би як бездротові, так і NPF-бібліотеки. Однією з головних цілей цього етапу буде визначення пріоритетності безпеки зв'язку між пристроєм і шлюзом. З цією метою використовуватимемо NPF для посилення конфіденційності та цілісності даних, що передаються. Це для забезпечення того, щоб

чутливі формування залишалися захищеними від потенційних загроз. Для того, щоб оцінити ефективність та результативність впровадження, використовуватимемо одні й ті самі показники протягом усього процесу тестування. Модель має значно нижчий загальний обсяг використання пам'яті порівняно з іншими моделями, що може зробити її більш придатною для пристроїв з обмеженими ресурсами пам'яті. Патерн демонструє найменше використання пам'яті серед інших патернів. Сторона лічильника представляє дані про шість різних шаблонів рукостискання в системі wM-Bus, захищеної NPF, з детальним описом як використання пам'яті, так і витраченого часу для всього процесу рукостискання на стороні лічильника. Модель має найвище загальне використання пам'яті, що потенційно робить її менш придатною для пристроїв з обмеженими ресурсами пам'яті. Патерн показує найменше використання пам'яті серед патернів. Модель виявляється найбільш ефективною за часом, з найкоротшим загальним часом, необхідним для процесу рукостискання.

Об'єднання обох сторін шлюзу і лічильника рукостискання включає зв'язок до відправки даних по захищених каналах. Використання пам'яті порівняно з ефективністю часу виділяється значно нижчим загальним використанням пам'яті, що робить її привабливим варіантом для систем із жорсткими обмеженнями пам'яті. Однак це відбувається за рахунок найвищого загального часу, що вказує на компроміс між ефективністю пам'яті та ефективністю часу.

Шаблони з високим рівнем пам'яті демонструють завжди найвище використання пам'яті, що може бути непридатним для середовищ з обмеженим обсягом пам'яті, незважаючи на його відносно високий загальний час. Вибір між цими шаблонами може залежати від конкретних вимог системи wM-Bus. Системи, які віддають перевагу меншому використанню пам'яті, можуть схилятися до моделі, незважаючи на її довший час. На противагу цьому, системи, які вимагають балансу між використанням пам'яті та ефективністю часу, можуть віддати перевагу типовому підходу.

Проаналізуємо використання та вплив пам'яті лічильника. Категорії мають однаковий час, що свідчить про те, що їхні процеси, ймовірно, можна порівняти з точки зору часової складності. Категорія виділяється найкоротшим часом, що минув,

який значно менший, ніж інші, що означає, що модель може бути більш ефективною або спрощеною в термінах фази. Оптимізація NPF для протоколу. На цьому етапі потрібно послідовно використовувати ідентичне апаратне налаштування для карт, зокрема розроблену модель (формула (2.1)), щоб точно відтворити функціональність як лічильника, так і шлюзу. Щоб підвищити надійність NPF, не потрібно обмежуватися простим виконанням заздалегідь встановлених шаблонів. Натомість потрібно застосувати більш динамічний підхід, інтегрувавши комплексні властивості безпеки в розроблений код, тим самим оптимізувавши загальну продуктивність системи та забезпечивши вищий рівень безпеки та надійності передачі даних.

Таким чином, запропонований процедурний підхід відобразив усталені практики, адаптуючи їх у контексті NPF. Це передбачає ініціювання рукостискання з подальшим обміном повідомленнями. Залежно від варіанту протоколу, це може завершитися повідомленням у сценарії з трьох повідомлень або продовжитися без нього за сценарієм із двох повідомлень. Подальше шифрування та розшифровка комунікацій залежать від заздалегідь узгодженої комбінації набору шифрів, механізму Діффі-Хеллмана та алгоритму хешування, що забезпечує безпечне та ефективне криптографічне рукостискання.

Кожен цикл виконання експерименту потрібно фіксувати, а результати систематично документувати в спеціальному файлі журналу. Такий підхід забезпечить детальний і доступний запис результатів, сприяючи глибокому аналізу та порівнянню різних протестованих конфігурацій безпеки. Накопичені дані, поряд з ілюстративними цифрами, нададуть всебічний огляд результатів експерименту, підкреслюючи ефективність, надійність і наслідки для безпеки кожного протестованого зразка і конфігурації.

В результаті застосований процедурний підхід ґрунтується на усталених практиках, які адаптовано до специфіки NPF. Це передбачає ініціацію рукостискання, після чого відбувається обмін повідомленнями між пристроями. Залежно від обраного протоколу, процес може завершитися після трьох обмінів повідомленнями (сценарій з трьома повідомленнями) або продовжуватися без цього етапу (сценарій з двома повідомленнями). Подальше шифрування і розшифровка переданих даних

залежать від заздалегідь узгодженої комбінації криптографічних алгоритмів: набору шифрів, механізму обміну ключами Діффі-Хеллмана та алгоритму хешування, що гарантує безпечний і ефективний процес криптографічного рукостискання. Кожен цикл експерименту має бути задокументований, а результати необхідно ретельно фіксувати в спеціалізованому журналі. Такий підхід забезпечить створення детальних та доступних записів результатів, що сприятиме глибокому аналізу та порівнянню різних конфігурацій безпеки, які були протестовані. Накопичені дані, доповнені ілюстративними графіками, дозволять отримати всебічний огляд результатів дослідження, підкреслюючи ефективність, надійність та вплив на безпеку кожної протестованої конфігурації.

3.3 Висновки до третього розділу

Розроблено метод забезпечення безпечного протоколу для комунікації пристроїв з IoT. Встановлено необхідність включати до розгляду IoT-пристрої в мережах wM-Bus такі інструменти, як набір розробників для аналізу використання пам'яті, розміру пакетів, часу встановлення з'єднання та функцій безпеки. Для перевірки безпеки NPF у контексті інтеграції застосовуватимемо Noise Explorer. Оцінювання безпеки здійснюватиметься на основі моделі, а також аналізуватимемо ефективність і вплив на ресурс батареї. В результаті застосований процедурний підхід ґрунтується на усталених практиках, які адаптовано до специфіки NPF. Це передбачає ініціацію рукостискання, після чого відбувається обмін повідомленнями між пристроями. Залежно від обраного протоколу, процес може завершитися після трьох обмінів повідомленнями (сценарій з трьома повідомленнями) або продовжуватися без цього етапу (сценарій з двома повідомленнями). Подальше шифрування і розшифровка переданих даних залежать від заздалегідь узгодженої комбінації криптографічних алгоритмів: набору шифрів, механізму обміну ключами Діффі-Хеллмана та алгоритму хешування, що гарантує безпечний і ефективний процес криптографічного рукостискання.

4 РЕАЛІЗАЦІЯ ЗАХИЩЕНОГО ПРОТОКОЛУ ДЛЯ ПРИСТРОЇВ ІОТ

4.1 Захищений протокол для пристроїв IoT

В епоху цифрової трансформації пристрої Інтернету речей (IoT) стали невід'ємною частиною повсякденного життя, впливаючи на різні аспекти, такі як домашня автоматизація, моніторинг охорони здоров'я, промислові системи управління та розумні міста. Повсюдний характер цих пристроїв підкреслює критичну важливість безпеки для захисту особистих даних, конфіденційності та безперебійної функціональності систем. Однак, незважаючи на незаперечну потребу в надійних заходах безпеки, у ландшафті IoT зберігається помітна прогалина. Деякі програми, які надають пріоритет зниженню витрат, простоті розгортання та енергоефективності, часто не враховують впровадження адекватних протоколів безпеки. Цей недолік частково виражений у пристроях IoT, що живляться від батарейок, таких як розумні лічильники води та газу, де застосункове споживання енергії, пов'язане із заходами безпеки, викликає серйозні занепокоєння. Розумні лічильники, які мають ключове значення для модернізації управління комунальними послугами та підвищення енергоефективності, уособлюють цю проблему. Ці пристрої, особливо ті, що відстежують споживання води та газу, зазвичай працюють від батарейок і, як очікується, працюватимуть протягом тривалого часу без обслуговування. Впровадження будь-якого механізму безпеки, який може потенційно розрядити батарею або ускладнити процес розгортання, часто зустрічається з реакцією, що підкреслює критичний компроміс між безпекою та оперативною ефективністю. Тому, представлено нове застосування NPF для захисту зв'язку wM-Bus, які є ключовими для інфраструктури інтелектуальних вимірювань.

Згідно методу забезпечення безпеки пристроїв IoT захищеним протоколом для комунікації між ними розроблено новий протокол, який має більшу захищеність при передачі даних і стійкий до розрялдення батарей пристроїв. Він базується на протоколі wM-Bus, який є його основою і в який інтегровано елементи захисту. Структура повідомлення протоколу wM-Bus зображена на рис. 4.1.

Рисунок 4.1 - Структура повідомлення протоколу wM-Bus [81]

Процедуру пошуку за цим протоколом можна розширити за допомогою пошуку виробника, покоління, які мають однаковий ідентифікаційний номер [81]. Також можна шукати всі версії певного виробника або всі версії певного носія, встановивши відповідне значення. Компанія [81] моделювала такий пошук, щоб знайти мінімум, середню та максимальну кількість вибору як функцію кількості версій. Для мінімальної кількості спроб було обрано оптимальний розподіл ідентифікаційних номерів, для максимальної кількості найбільш несприятливого та для середньої кількості спроб випадкового розподілу. Комплексне дослідження технологій бездротового зв'язку було проведено з огляду протоколів бездротового зв'язку з метою визначення найбільш підходящої технології з урахуванням як експлуатаційних, так і безпекових вимог. В основному, зосереджуючись на ключових архітектурних елементах, таких як дальність, швидкість передачі даних, частота, структура протоколу та безпека. Вибір wM-Bus та вирішення його проблем безпеки виявилось стратегічно доцільним при поєднанні його з певними технологіями бездротового зв’язку. Протокол wM-Bus був обраний через його актуальність та широке впровадження на практиці. Злиття протоколу Noise Protocol з операційною структурою wM-Bus дало змогу встановити підвищені стандарти безпеки в системі IoT. Комплексне впровадження для забезпечення зв'язку wM-Bus систематично проводилося з етапів. Крім того, це рішення не є квантово-готовим. Хоча поточні криптографічні механізми забезпечують надійну безпеку для традиційних обчислювальних середовищ, але вони можуть не витримувати передових обчислювальних можливостей квантових комп'ютерів. Запропоноване рішення, яке спирається на традиційні криптографічні методи, може стати вразливим, як тільки квантові обчислення стануть більш поширеними. Майбутні дослідження повинні вивчати інтеграцію квантово-стійких алгоритмів для забезпечення довгострокової безпеки в технологічному ландшафті, що розвивається. Нарешті, усунення обмежень,

виявлених особливо щодо обмежень розміру повідомлень, пропонує критичний шлях для майбутньої роботи. Потрібно удосконалювати способи подолання обмежень, пов'язаних з розмірами повідомлень, тим самим підвищуючи універсальність і застосовність фреймворку в більш складних сценаріях. Графік залежності величини повідомлення від кількості пристроїв IoT, який зображено на рис. 4.2, показує лінійну залежність, яка несуттєво зростає при ускладненні протоколу з включенням до нього елементів захисту при бездротовій передачі повідомлень.

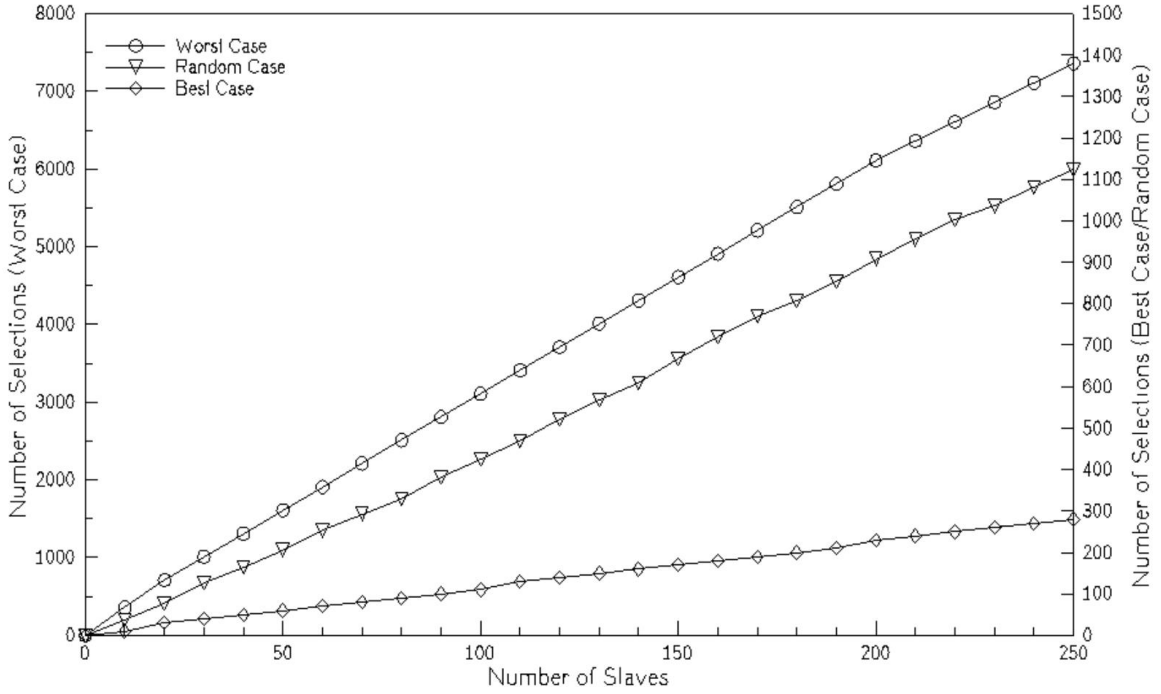


Рисунок 4.2 - Графік залежності величини повідомлення від кількості пристроїв IoT [81]

Майстер вивільняє розширене скидання даних застосунків для всіх повідомлень. Усі повідомлення з типами даних користувача вимагаються від усіх пристроїв. Інтерфейс для повідомлень зображено на рис. 4.3.

```
Master to Slave: 68 04 04 68 | 53 FE 50 | 10 | B1 16
Slave to Master: E5
```

Рисунок 4.3 - Інтерфейс для повідомлень

У напрямку відповіді з довгим кадром використовуються дві різні структури даних. Фіксована структура даних, крім фіксованої довжини, обмежена передачею лише двох лічильників заздалегідь визначеної довжини, які мають двійкове або ускладнене кодування. На відміну від структури змінних даних дозволяє передавати більше лічильників у різних кодах та подальшу корисну інформацію про дані. Кількість байтів переданих лічильників стану також є змінною з цією структурою даних. На відміну від фіксованої структури, змінна структура, також, може використовуватися в напрямку виклику. З цієї причини фіксована структура даних не рекомендується для майбутніх розробок. На рис. 4.4 зображено структуру повідомлення з полями і їх розміром.

Identification No.	Access No.	Status	Medium/Unit	Counter 1	Counter 2
4 Byte	1 Byte	1 Byte	2 Byte	4 Byte	4 Byte

Рисунок 4.4 - Фіксована структура даних у повідомлені (послідовність передачі зліва направо) [81]

Для ідентифікації фіксованої структури даних використовуються цифри 73H/77H для поля управління контролем. Таким чином, головне програмне забезпечення може побачити, як воно повинно інтерпретувати дані.

Wireless Meter-Bus (M-Bus) - це європейський стандарт (EN 13757-4), призначений для дистанційного зняття показань лічильників газу або електроенергії. Зі зростаючим впровадженням інтелектуальних лічильників та Інтернету речей (IoT) безпека зв'язку в цих пристроях стала першочерговою. Традиційний підхід спирається на складні криптографічні алгоритми та повнофункціональний стек протоколу, який може потребувати пам'яті. Це включає зберігання сертифікатів, приватних ключів та інформації про стан сеансу. Обсяг пам'яті може бути значним, особливо для пристроїв з обмеженими ресурсами, через додаткові витрати криптографічних бібліотек і самого протоколу. NPF розроблений таким чином, щоб

бути легким та ефективним. Зазвичай він вимагає менше пам'яті, оскільки використовує оптимізовані криптографічні операції та не має великих накладних витрат протоколу. Використання патерну оптимізовано для мінімального використання пам'яті, що робить його добре придатним для пристроїв з обмеженими ресурсами. Традиційний протокол додає значну кількість додаткових витрат до кожного пакету, включаючи шифрування, MAC для цілісності та, можливо, заголовки записів. Це збільшує розмір пакету, що може викликати занепокоєння в середовищах з обмеженою пропускною здатністю. Початкове рукостискання також включає кілька надсилань повідомлень туди і назад та обмін даними сертифікату, що може призвести до великих розмірів пакетів на ранніх стадіях зв'язку. Базова версія протоколу, як правило, має нижчі додаткові витрати на повідомлення, оскільки він не має підпису для простоти та ефективності. Шаблон протоколу зосереджений на криптографічних примітивах композицій, зменшуючи загальний розмір пакету. Процес рукостискання оптимізований, включає меншу кількість повідомлень і менший обмін даними, що призводить до менших початкових розмірів пакетів.

Рукостискання – це багатоетапний процес, який включає кілька переходів між клієнтом і сервером туди і в зворотньому напрямі. Це може спричинити значну затримку, особливо в мережах із високим часом проходження сигналу туди і в зворотньому напрямі. Процес узгодження криптографічних параметрів, обміну сертифікатами та їх перевірки може зайняти багато часу, особливо на пристроях з обмеженою обчислювальною потужністю. Рукостискання Noise розроблено таким чином, щоб бути швидким та ефективним. Патерн з його вибором криптографічних алгоритмів оптимізований для швидкого рукостискання з мінімальними обчислювальними витратами. Зменшена кількість надсилань повідомлень туди і в зворотньому напрямі та відсутність обміну сертифікатами в процесі рукостискання Noise сприяють скороченню часу завершення. Вибір між традиційним підходом до забезпечення бездротового зв’язку і NPF з використанням патерну для захисту зв'язку wM-Bus в середовищах IoT передбачає балансування потреб безпеки з обмеженнями пристрою. Незважаючи на те, що традиційний підхід добре зарекомендував себе та в ньому універсально підтримуються системи безпеки, але його вимоги до пам'яті та

пропускної здатності, а також трудомісткий процес рукостискання можуть бути не ідеальними для всіх пристроїв IoT, особливо для тих, що мають жорсткі обмеження потужності та повторного джерела. З іншої сторони, NPF з оновленим патерном надає легку та ефективну альтернативу, потенційно краще підходить для обмежених середовищ. Компромісом є необхідність ретельного впровадження та потенційно менш універсальної підтримки протоколів.

Таким чином, розроблено протокол з покращеними характеристиками безпеки для пристроїв IoT за рахунок внесення в базову схему wM-Bus додаткового механізму від фреймворку NPF, який забезпечив захист бездротового зв’язку.

4.2 Дослідження впливу кіберзагроз на захищений протокол для пристроїв IoT

Визначення кіберзагроз – це процес виявлення, аналізу та класифікації потенційних загроз, які можуть вплинути на безпеку інформаційних систем, мереж та даних. У контексті захищених протоколів для пристроїв Інтернету речей (IoT) кіберзагрози охоплюють широкий спектр атак, які можуть порушити конфіденційність, цілісність або доступність переданих даних.

Ключові аспекти визначення кіберзагроз:

1. Ідентифікація можливих загроз. Визначення типів атак, які можуть бути спрямовані на протокол. Наприклад: MitM-атаки (зловмисник посередині), тобто перехоплення даних між пристроями; DDoS-атаки, тобто перевантаження пристроїв або серверів через велику кількість запитів; атаки на автентифікацію, тобто зламування облікових даних за допомогою грубої сили; експлойти у вразливостях програмного забезпечення, тобто використання помилок в реалізації протоколу.

2. Класифікація загроз. Використання моделей і стандартів для систематизації загроз: STRIDE-модель, в якій класифікуються загрози на шість категорій (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege); OWASP IoT Top Ten, в якій аналізується перелік основних ризиків для пристроїв IoT, таких як ненадійна аутентифікація, недостатній контроль доступу тощо.

3. Вивчення можливих сценаріїв атак. Аналіз реальних або гіпотетичних

сценаріїв, за яких загроза може бути реалізована. Наприклад: сценарій 1, коли зловмисник перехоплює незашифрований трафік між двома пристроями; сценарій 2, коли відбувається атака з використанням вразливості в механізмах оновлення прошивки пристрою.

4. Оцінка ймовірності та впливу. Оцінка ризиків для кожної загрози, враховуючи ймовірність її реалізації та потенційний збиток: аналіз даних, які можуть бути втрачені або скомпрометовані; дослідження та аналіз рівня доступу, який може отримати зловмисник.

5. Використання інструментів для ідентифікації загроз: автоматизовані сканери, такі як Nmap, Nikto чи Burp Suite, для виявлення вразливостей; аналіз журналів подій для пошуку аномалій у мережевому трафіку.

6. Документування загроз. Оформлення звіту, який включає: перелік виявлених загроз; можливі наслідки загроз; рекомендації для усунення чи мінімізації загроз.

Розумні лічильники вразливі до різних кіберзагроз, які можуть спричинити значні збої або витік даних. Ці загрози для системи інтелектуального обліку можна класифікувати як активні або пасивні. У той час як пасивні загрози передбачають просте спостереження та аналіз даних без будь-яких змін, активні загрози включають маніпуляції та зміну даних. Виходячи з аналізу типів кіберзагроз та конкретних вимог до побудови моделі загрози STRIDE для wM-Bus з використанням NPF розглянемо їх безпосередньо. Щоб побудувати модель загроз STRIDE для запропонованого протоколу з NPF, потрібно систематично аналізувати кожен компонент STRIDE (Spoofing, Tampering, Reclosure, Information Disclosure, Denial of Service та Elevation of Privilege) у контексті властивостей безпеки шаблонів NPF.

Розглянемо детальніше особливості моделей загроз STRIDE.

Підробка особистості - це загроза, яка передбачає, що зловмисник видає себе за іншого користувача або пристрій. У контексті NPF використання безпечних механізмів обміну ключами для встановлення спільної таємниці може пом'якшити цю загрозу. Впровадження механізмів автентифікації за допомогою попереднього спільного ключа може допомогти подолати цю загрозу.

Фальсифікація - це загроза, яка полягає в тому, що зловмисник змінює дані під

час передачі. NPF використовує алгоритми шифрування, щоб захистити цілісність і конфіденційність даних. Забезпечення шифрування та автентифікації всіх даних перед передачею може зменшити загрози несанкціонованого доступу.

Відмова – це загроза, яка означає, що суб'єкт заперечує дію, яку він виконав. Щоб протистояти цій загрозі, може бути ефективним впровадження заходів невідмови, таких як цифрові підписи або комплексні механізми ведення журналу, які записують обмін ключами та транзакції даних у межах роботи протоколу.

Розкриття інформації - це загроза, що стосується несанкціонованого доступу до даних. Використання надійних алгоритмів шифрування гарантує шифрування даних, зменшуючи ризик розголошення інформації. Дуже важливо переконатися, що всі конфіденційні дані зашифровані, а ключі надійно керовані.

Відмова в обслуговуванні (DoS) - це загроза, яка полягає в тому, що зловмисник перешкоджає законним користувачам отримати доступ до послуг. У той час як NPF в першу чергу зосереджений на конфіденційності та цілісності, пом'якшення DoS може включати впровадження обмеження швидкості, аутентифікацію перед розподілом ресурсів і моніторинг аномальних моделей трафіку.

Підвищення привілеїв - це загроза, яка полягає в тому, що зловмисник отримує вищий рівень захисту, ніж планувалося. Належний контроль доступу на основі ролей і забезпечення того, що протокол ненавмисно не допускає ескалації привілеїв через недоліки в його дизайні або реалізації, можуть пом'якшити цю загрозу. Потрібно визначити компоненти лічильника та пристрій, який збирає дані про використання комунальних послуг.

Модель загроз для патернів характеризується тим, що ініціатор надсилає ефемерний відкритий ключ і зашифровані корисні навантаження без аутентифікації в першому повідомленні. Автентифікація відповідача перед ініціатором у наступному повідомленні, забезпечується інтеграцією NPF за допомогою шаблону у зв'язок wM-Bus, що в свою чергу забезпечує конфіденційність, цілісність і автентичність повідомлень, якими обмінюються між лічильником і шлюзом.

Тому, використання шаблону NPF у комунікаціях wM-Bus значно підвищує безпеку, забезпечуючи шифрування, автентифікацію та захист повідомлень від атак

повторного відтворення.

Патерн — це шаблон рукостискання в NPF, який забезпечує взаємну аутентифікацію Цей патерн передбачає: двосторонню аутентифікацію, коли обидві сторони доводять один одному свою особистість; обмін ключами, що дозволяє обом сторонам отримати спільний секретний ключ для шифрування та розшифровки свого зв'язку. Безпека зв'язку wM-Bus, посилена NPF за допомогою шаблону гарантує: конфіденційність; цілісність; автентичність повідомлень між сторонами, що спілкуються.

У дослідженні інтеграції технології wM-Bus з NPF було використано інструмент верифікації для оцінки ефективності безпеки двох попередньо визначених моделей зв'язку в рамках цієї структури. Він глибоко аналізує криптографічні протоколи, зосереджуючись на їхній здатності захищати зв'язок за допомогою серії запитів на аутентифікацію та конфіденційність. Ці запити призначені для оцінки того, наскільки добре протокол може автентифікувати сторони один перед одним і зберігати конфіденційність їхніх повідомлень. У цьому контексті інструмент інтеграції технології wM-Bus з NPF є критично важливим компонентом в оцінці стану безпеки моделей зв'язку. Систематично оцінюючи реакцію шаблонів на серію запитів безпеки, інструмент надає цінну інформацію про їхні сильні та потенційні вразливості, дозволяючи приймати обґрунтовані рішення щодо їх розгортання в захищених каналах зв'язку.

Проаналізуємо час автономної роботи та фактори, які слід враховувати. Вони включатимуть споживання енергії, пов'язане з криптографічними операціями, вплив збільшення розмірів пакетів на використання радіозв'язку та загальну ефективність роботи протоколу в середовищі з низьким енергоспоживанням. В оцінці часу автономної роботи пристроїв IoT за трьома протоколами зв'язку очевидно, що вибір протоколу значно впливає на довговічність роботи. Розрахунок споживаної потужності для конкретного фреймворку або програмного пакета типу NPF за допомогою вимірювального пристрою вимагає більш конкретної інформації. Споживання енергії залежить від різних факторів, зокрема від обладнання, на якому він працює, шаблонів використання та робочого навантаження. Однак покладання на

дані мікросхеми допоможе оцінити енергоспоживання та час автономної роботи за цими двома закономірностями. Тобто, потрібно визначитись щодо збільшення витрат електроенергії при ускладненні базової схеми в зв’язку із доповненням протоколу захисними елементами. В зв’язку з тим, що вимірювати кожен пристрій IoT в момент передачі повідомлення неможливо, бо для різних пристроїв час надсилань відомостей різний, тоді збільшене використання електроенергії можна лише оцінити. Оскільки в протокол додано не тільки додаткові повідомлення для рукостискання, але й шифрування, то витрати на їх реалізацію впливатимуть на використання батарей. Тому, розглянемо спочатку шифри, які можуть бути використані в пристроях IoT та обсяг завдань на їх реалізацію. Це необхідно для оцінювання їх впливу на витрати енергії батарей порівняно з впливом без них щодо тривалості цього процесу. Тобто, за основу методології оцінювання візьмемо час, як критерій. Вважатимемо чи допускатимемо таку модель споживання енергії батарей, при якій витрати енергії батарей пристроїв IoT розподілено рівномірно за часом споживання. Тоді, витрати на шифрування будуть мати аналогічні показники до витрат на опрацювання, підготовку та надсилання повідомлень з лічильників. Передбачається, що процес рукостискання споживає енергію зі швидкістю, що включає зв'язок, але не повну передачу даних. Енергія, витрачена в процесі рукостискання, розраховується як на основі наданих припущень і детальної розбивки того, як з передачами двічі на день, включаючи час рукостискання.

Шифрування у пристроях Інтернету речей (IoT), які використовуються в лічильниках комунальних послуг (наприклад, електроенергії, води, газу), є ключовим елементом для забезпечення конфіденційності, цілісності та автентичності даних. Дані, що передаються з таких лічильників, мають бути захищені від несанкціонованого доступу, маніпуляцій і перехоплення.

Розглянемо особливості шифрування в IoT-лічильниках.

Лічильники мають обмежену потужність процесорів, пам’ять і енергоспоживання. Вони використовують легковагові алгоритми шифрування, оптимізовані для пристроїв з низькими ресурсами. Для них передбачено тривале використання. Пристрої можуть працювати десятиліттями, тому важливо

використовувати криптографічні алгоритми, які залишатимуться надійними протягом усього життєвого циклу. Лічильники часто працюють у мережах із низькою пропускною здатністю (наприклад, LoRaWAN, Zigbee) або мобільному зв’язку.

Алгоритми шифрування для лічильників IoT.

Симетричне шифрування (AES – Advanced Encryption Standard) - широко використовується завдяки високій швидкості та ефективності. Використовуються ключі довжиною 128, 192 або 256 біт. Наприклад, у протоколах Zigbee та LoRaWAN застосовується AES-128 для захисту даних.

Асиметричне шифрування (ECC – Elliptic Curve Cryptography) забезпечує високу безпеку за меншого розміру ключів порівняно з RSA. Використовується для встановлення безпечного з'єднання або обміну симетричними ключами.

Гешування (SHA – Secure Hash Algorithm) забезпечує цілісність даних шляхом створення хеш-функцій. Наприклад, SHA-256 може використовуватися для перевірки автентичності повідомлень.

Для шифрування в IoT-лічильниках використовуються різні протоколи шифрування. Розглянемо основні з них. Протокол DLMS/COSEM (Device Language Message Specification / Companion Specification for Energy Metering) **в**икористовується для комунікації в інтелектуальних лічильниках електроенергії і підтримує шифрування даних на основі AES. Протокол LoRaWAN для бездротових пристроїв із низьким енергоспоживанням та включає AES-128 для захисту даних на мережевому та додатковому рівнях. Протокол Zigbee **в**икористовується в лічильниках для коротких бездротових з’єднань і підтримує шифрування AES-128 для забезпечення безпеки передачі. Протокол TLS/DTLS (Transport Layer Security / Datagram TLS) **з**абезпечує шифрування в мережах IP для передачі даних лічильників і DTLS є адаптацією TLS для роботи з UDP.

Виклики шифрування в IoT-лічильниках.

Управління ключами. Безпечна генерація, зберігання та оновлення ключів є критично важливими. Використання вбудованих модулів безпеки, таких як TPM (Trusted Platform Module).

Вразливості в реалізації**.** Неправильна реалізація криптографії може стати

джерелом вразливостей. Потрібне регулярне оновлення прошивки для усунення помилок.

Фізичний доступ до пристроїв**.** Лічильники можуть бути доступними фізично, тому слід захистити внутрішні компоненти та запобігти крадіжці ключів.

Енергоспоживання. Шифрування повинно бути ефективним, щоб не зменшувати час роботи пристрою від батареї.

Рекомендації для впровадження шифрування. Використовувати стандартизовані протоколи безпеки (наприклад, LoRaWAN, DLMS/COSEM). Інтегрувати механізми оновлення прошивки для підтримки сучасних криптографічних алгоритмів. Використовувати апаратні модулі безпеки для зберігання ключів. Регулярно тестувати безпеку пристроїв на вразливості.

Шифрування забезпечує захист даних лічильників, запобігаючи їх несанкціонованому використанню та маніпуляціям, що є критичним для сучасних IoT-систем у комунальних послугах.

Витрати енергії на шифрування в пристроях Інтернету речей (IoT), зокрема в комунальних лічильниках, є важливим аспектом, оскільки ці пристрої зазвичай працюють від батарей протягом тривалого часу. Енергоефективність шифрування залежить від обраного алгоритму, апаратної реалізації та обсягу оброблюваних даних.

Чинники, що впливають на витрати енергії.

Тип алгоритму шифрування: симетричні алгоритми (наприклад, AES; швидші та менш енерговитратні; використовуються для великих обсягів даних); асиметричні алгоритми (наприклад, ECC, RSA; енергозатратні через складні математичні операції; застосовуються лише на етапі встановлення з’єднання або обміну ключами); геш-функції (наприклад, SHA-256; використовуються для перевірки цілісності даних, мають середній рівень енерговитрат).

Апаратна реалізація**.** Спеціалізовані апаратні модулі (наприклад, Cryptographic Accelerators) значно знижують витрати енергії. Програмна реалізація шифрування на стандартному мікроконтролері споживає більше енергії.

Обсяг даних для шифрування**.** Менші повідомлення (наприклад, телеметрія лічильників) потребують менше енергії. Великі обсяги даних збільшують

енергоспоживання.

Частота операцій**.** Пристрої, які надсилають дані рідше (наприклад, раз на добу), витрачають менше енергії на шифрування, ніж ті, що працюють у режимі реального часу.

Мережевий протокол**.** Протоколи, що оптимізовані для енергоефективності (наприклад, LoRaWAN, Zigbee), мінімізують частоту та обсяг передачі даних, зменшуючи витрати на шифрування.

Типові витрати енергії на шифрування подано в табл. 4.1.

Таблиця 4.1

Орієнтовні енергетичні витрати для поширених алгоритмів шифрування:

Алгоритм	Енергоспоживання (на 1 блок/операцію)	Примітка
AES-128	~20-60 мкДж	Ефективний для симетричного шифрування
ECC (ключ 256 біт)	~5-10 мДж	Висока енерговитратність, застосовується рідше
SHA-256	~50-150 мкДж	Використовується для хешування
RSA (ключ 2048 біт)	~30-50 мДж	Дуже енерговитратний, не рекомендується для IoT

Способи зменшення енерговитрат на шифрування.

Використання легковагових алгоритмів. Алгоритми, як-от AES-128, є оптимальним вибором для IoT-пристроїв. Полегшені версії криптографічних алгоритмів, наприклад, TinyCrypt, розроблені для обмежених ресурсів.

Оптимізація роботи. Зменшення частоти передачі даних. Пакетування даних для одноразового шифрування великих обсягів.

Апаратні прискорювачі. Використання мікроконтролерів із вбудованими модулями шифрування, які споживають менше енергії.

Вибір енергоефективних протоколів. Протоколи, такі як LoRaWAN і Zigbee, мінімізують витрати енергії завдяки оптимізації комунікації.

Компроміс між безпекою та ефективністю. Для лічильників, що надсилають дані рідко, можна використовувати менш енерговитратні алгоритми з достатнім рівнем безпеки.

Приклад реального сценарію. Лічильник, який використовує AES-128 для шифрування даних та LoRaWAN для передачі, витрачає приблизно 20-60 мкДж на один блок шифрування (16 байт)**.** При передачі одного повідомлення обсягом 64 байти шифрування споживає близько 240 мкДж, що є невеликим внеском у загальне енергоспоживання пристрою.

Приклад 1. Енергоспоживання при використанні AES-128 у LoRaWAN. Лічильник комунальних послуг передає дані про споживання електроенергії через мережу LoRaWAN кожні 15 хвилин. Повідомлення складається з 64 байтів. Використовується шифрування AES-128 (16-байтові блоки). Витрати енергії. шифрування одного блоку (16 байтів) з AES-128 споживає приблизно 30 мкДж. Для 64-байтового повідомлення потрібні 4 блоки, тому:

30 мкДж×4=120 мкДж30\,

\text{мкДж}\times 4 = 120\,

\text{мкДж}30мкДж×4=120мкДж

Якщо пристрій передає дані 96 разів на добу (кожні 15 хвилин), то енергія, витрачена на шифрування:

120 мкДж×96=11.52 мДж/добу120 \,

\text{мкДж} \times 96 = 11.52 \,

\text{мДж/добу}120мкДж×96=11.52мДж/добу

Приклад 2. Використання ECC для обміну ключами. Лічильник газу встановлює з’єднання з сервером для передачі даних. Під час встановлення з’єднання використовують ECC (еліптичну криптографію) для обміну ключами. Витрати енергії. ECC-256 споживає близько 6 мДж на один обмін ключами. Якщо пристрій оновлює сесію раз на день, це додає:

6 мДж/день6 \, \text{мДж/день}6мДж/день.

Це значно більше, ніж симетричне шифрування, але обмін ключами виконується рідше.

Приклад 3. Шифрування телеметрії з використанням RSA. Лічильник води передає дані через мережу 3G раз на день, використовуючи RSA-2048 для автентифікації. Витрати енергії. RSA-2048 потребує близько 40 мДж для одного циклу шифрування/дешифрування. Якщо передача відбувається щодня, це споживання становитиме:

40 мДж/день40 \, \text{мДж/день}40мДж/день.

У порівнянні з AES або ECC, RSA має набагато більші енергетичні витрати і не підходить для частого використання.

Приклад 4. Хешування даних для перевірки цілісності. Лічильник електроенергії передає дані зі SHA-256 для створення хешу повідомлення. Дані складаються з 32 байтів. Витрати енергії.SHA-256 споживає близько 100 мкДж для хешування 32 байтів. Якщо пристрій передає дані раз на годину:

100 мкДж×24 разів на добу=2.4 мДж/добу100 \,

\text{мкДж} \times 24 \, \text{разів на добу} = 2.4 \,

\text{мДж/добу}100мкДж×24разів на добу=2.4мДж/добу

Порівняння витрат енергії для різних алгоритмів шифрування подано в табл. 4.2.

Таблиця 4.2

Порівняння витрат енергії для різних алгоритмів шифрування

Алгоритм	Тип використання	Енергоспоживання за одну операцію	Частота	Денне енергоспоживання
AES-128	Шифрування даних	~30 мкДж	96 передач	11.52 мДж
ECC-256	Обмін ключами	~6 мДж	1 раз	6 мДж
RSA-2048	Автентифікація	~40 мДж	1 раз	40 мДж
SHA-256	Перевірка цілісності	~100 мкДж	24 передачі	2.4 мДж

Для IoT-лічильників найчастіше використовують AES-128 для шифрування

даних через його низькі енергетичні витрати, а також ECC для обміну ключами, якщо потрібен високий рівень безпеки. Використання енергоефективних алгоритмів є критично важливим для забезпечення довготривалої роботи пристроїв.

Таким чином, досліджено особливості кіберзагроз для протоколів комунікації між пристроями IoT. Встановлено кіберзагрози та проаналізовано їх потенційний вплив при використанні інтеграції технології wM-Bus з фреймворком NPF. В результаті такого поєднання значна частина кіберзагроз втрачає суттєві перспективи щодо подальшого розвитку і реалізації. Також, було досліджено загрози, які пов’язані з додатково уведеним шифруванням, що потребує витрат енергії батарей і, відповідно, неможливості функціонування пристроїв IoT. При цьому, досліджено в цьому контексті алгоритми шифрування щодо витрат енергії при їх використанні, проведено порівняння витрат та обрано варіанти для використання.

4.3 Висновки до четвертого розділу

В результаті розроблено захищений протокол для комунікації пристроїв IoT згідно методу забезпечення безпеки комунікацій згідно інтеграції технології wM-Bus з фреймворком NPF. Досліджено особливості кіберзагроз для протоколів комунікації між пристроями IoT. Встановлено кіберзагрози та проаналізовано їх потенційний вплив при використанні інтеграції технології wM-Bus з фреймворком NPF. В результаті такого поєднання значна частина кіберзагроз втрачає суттєві перспективи щодо подальшого розвитку і реалізації. Також, досліджено в цьому контексті алгоритми шифрування щодо витрат енергії при їх використанні, проведено порівняння витрат та обрано варіанти для використання.

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень розроблено новий метод забезпечення функціонування систем з IoT на основі захищених протоколів та отримано такі результати.

1. Проаналізовано відомі методи та засоби забезпечення функціонування систем з IoT на основі захищених протоколів.

2. Розроблено забезпечення функціонування систем з IoT на основі захищених протоколів.

3. Здійснено реалізацію протоколу згідно розробленого методу забезпечення функціонування систем з IoT на основі захищених протоколів.

4. Досліджено розроблений захищений протокол функціонування пристроїв IoT в контексті кіберзагроз та додаткових витрат енергії на здійснення шифрування повідомлень.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Michalski A., Watral Z. Problems of Powering End Devices in Wireless Networks of the Internet of Things. *Energies* 2021, 14. P. 2417. https://doi.org/10.3390/en14092417

2. Mansour M., Gamal A., Ahmed A.I., Said L.A., Elbaz A., Herencsar N., Soltan A. Internet of Things: A Comprehensive Overview on Protocols, Architectures, Technologies, Simulation Tools, and Future Directions. *Energies*. 2023; 16(8). P. 3465. https://doi.org/10.3390/en16083465

3. Darabkh K.A., Alfawares M.G., Althunibat S. MDRMA: Multi-data rate mobility-aware AODV-based protocol for flying ad-hoc networks. *Veh. Commun*. 2019, 18. P. 100163.

4. Alhasanat M., Althunibat S., Darabkh K.A., Alhasanat A., Alsafasfeh M. A physical-layer key distribution mechanism for IoT networks. *Mob. Netw. Appl*. 2020, 25. Pp. 173–178.

5. Hendriks S. Internet of Things: How the World Will Be Connected in 2025. *Master's Thesis*, Utrecht University, Utrecht, The Netherlands, 2016.

6. Milić D.C., Tolić I.H., Peko M. Internet of Things (IoT) solutions in smart transportation management. *In Proceedings of the Business Logistics in Modern Management*, Osijek, Croatia, 5–6 October 2020.

7. Wytrębowicz J., Cabaj K., Krawiec J. Messaging Protocols for IoT Systems—A Pragmatic Comparison. Sensors 2021, 21. P. 6904.

8. Sadeghi-Niaraki A. Internet of Thing (IoT) review of review: Bibliometric overview since its foundation. Future Gener. Comput. Syst. 2023, 143. Pp. 361–377.

9. Miorandi D., Sicari S., De Pellegrini F., Chlamtac I. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks* 2012, 10. Pp. 1497–1516.

10. Said O., Masud M. Towards internet of things: Survey and future vision. *Int. J. Comput. Networks* 2013, 5. Pp. 1–17.

11. Gerodimos A., Maglaras L., Ferrag M.A., Ayres N., Kantzavelou I. IoT: Communication protocols and security threats. *Internet Things Cyber-Phys. Syst.* 2023, 3. Pp. 1–13.

12. Domínguez-Bolaño T., Campos O., Barral V., Escudero C.J., García-Naya J.A. An overview of IoT architectures, technologies, and existing open-source projects. *Internet Things*. 2022, 20. P. 100626.

13. Bayılmış C., Ebleme M.A., Çavuşoğlu Ü., Küçük K., Sevin A. A survey on communication protocols and performance evaluations for Internet of Things. *Digit. Commun. Networks.* 2022, 8. Pp. 1094–1104.

14. Goulart A., Chennamaneni A., Torre D., Hur B., Al-Aboosi F.Y. On Wide-Area IoT Networks, Lightweight Security and Their Applications—A Practical Review. *Electronics*. 2022, 11. P. 1762.

15. Verma D., Singh K.R., Yadav A.K., Nayak V., Singh J., Solanki P.R., Singh R.P. Internet of things (IoT) in nano-integrated wearable biosensor devices for healthcare applications. *Biosens. Bioelectron.* X 2022, 11. P. 100153.

16. Oliveira L., Rodrigues J.J., Kozlov S.A., Rabêlo R.A., de Albuquerque V.H.C. MAC layer protocols for internet of things: A survey. *Future Internet* 2019, 11. P. 16.

17. Gupta S., Gupta A., Shankar G. Cloud Computing: Services, Deployment Models and Security Challenges. *In Proceedings of the 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 7–9 October 2021. Pp. 414–418.

18. Alotaibi A., Barnawi A. Securing massive IoT in 6G: Recent solutions, architectures, future directions. *Internet Things.* 2023, 22, P. 100715.

19. Singh R., Kovacs J., Kiss T. To offload or not? an analysis of big data offloading strategies from edge to cloud. *In Proceedings of the 2022 IEEE World AI IoT Congress (AIIoT)*, Seattle, WA, USA, 6–9 June 2022. Pp. 46–52.

20. Wang X., Han Y., Leung V.C.M., Niyato D., Yan X., Chen X. Convergence of Edge Computing and Deep Learning: A Comprehensive Survey. *IEEE Commun. Surv. Tutorials.* 2020, 22. Pp. 869–904.

21. Pujol V.C., Dustdar S. Fog robotics—Understanding the research challenges. *IEEE Internet Comput*. 2021, 25. Pp. 10–17.

22. Kumar P., Gupta G.P., Tripathi R. A distributed ensemble design based intrusion detection system using fog computing to protect the internet of things networks. *J. Ambient. Intell. Humaniz. Comput*. 2021, 12. Pp. 9555–9572.

23. Abouaomar A., Cherkaoui S., Mlika Z., Kobbane A. Resource Provisioning in Edge Computing for Latency-Sensitive Applications. *IEEE Internet Things J*. 2021, 8. Pp. 11088–11099.

24. Laroui M., Nour B., Moungla H., Cherif M.A., Afifi H., Guizani M. Edge and fog computing for IoT: A survey on current research activities & future directions. *Comput. Commun*. 2021, 180. P. 210–231.

25. Iftikhar S., Gill S.S., Song C., Xu M., Aslanpour M.S., Toosi A.N., Du J., Wu H., Ghosh S., Chowdhury D. et al. AI-based fog and edge computing: A systematic review, taxonomy and future directions. *Internet Things.* 2023, 21. P. 100674.

26. Shakarami A., Shakarami H., Ghobaei-Arani M., Nikougoftar E. Faraji-Mehmandar, M. Resource provisioning in edge/fog computing: A Comprehensive and Systematic Review. *J. Syst. Archit*. 2022, 122. Pp. 102362.

27. Zhang T., Shen Z., Jin J., Zheng X., Tagami A., Cao X. Achieving Democracy in Edge Intelligence: A Fog-Based Collaborative Learning Scheme. *IEEE Internet Things J*. 2021, 8. Pp. 2751–2761.

28. McEnroe P., Wang S., Liyanage M. A survey on the convergence of edge computing and AI for UAVs: Opportunities and challenges. *IEEE Internet Things J.* 2022, 9. Pp. 15435–15459.

29. Zhang Y., Yu H., Zhou W., Man M. Application and Research of IoT Architecture for End-Net-Cloud Edge Computing. *Electronics*. 2023, 12. P. 1.

30. Singh R., Gill S.S. Edge AI: A survey. *Internet Things-Cyber-Phys. Syst.* 2023, 3. Pp. 71–92.

31. Manowska A., Wycisk A., Nowrot A., Pielot J. The Use of the MQTT Protocol in Measurement, Monitoring and Control Systems as Part of the Implementation of Energy Management Systems. *Electronics.* 2023, 12. P. 17.

32. Arvind S., Narayanan V.A. An overview of security in CoAP: Attack and analysis. In Proceedings of the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019. Pp. 655–660.

33. Nikolov N. Research of MQTT, CoAP, HTTP and XMPP IoT Communication protocols for Embedded Systems. *In Proceedings of the 2020 XXIX International Scientific*

Conference Electronics (ET), Sozopol, Bulgaria, 16–18 September 2020. Pp. 1–4.

34. Sun, L.; Chen, Y.; Cheng, Q.; Zhu, B.; Chen, C.; Hou, X. Communication Application of Distributed Energy Resources Monitoring System Based on XMPP. In Proceedings of the 2021 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), Shanghai, China, 21–29 August 2021; pp. 66–70.

35. Hofer-Schmitz, K.; Stojanović, B. Towards formal verification of IoT protocols: A Review. Comput. Networks 2020, 174, 107233.

36. Adi P.D.P., Sihombing V., Siregar V.M.M., Yanris G.J., Sianturi F.A., Purba W., Tamba S.P., Simatupang J., Arifuddin R., Subairi et al. A Performance Evaluation of ZigBee Mesh Communication on the Internet of Things (IoT). *In Proceedings of the 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, Surabaya, Indonesia, 9–11 April 2021. Pp. 7–13.

37. Gavra V.D., Pop O.A. Usage of ZigBee and LoRa wireless technologies in IoT systems. *In Proceedings of the 2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, Pitesti, Romania, 21–24 October 2020. Pp. 221–224.

38. Cheruvu S., Kumar A., Smith N., Wheeler D.M. Demystifying Internet of Things SECURITY: Successful Iot Device/Edge and Platform Security Deployment; *Springer*: *Berlin/Heidelberg*, Germany. 2020.

39. Fatihah S.N., Dewa G.R.R., Park C., Sohn I. Self-Optimizing Bluetooth Low Energy Networks for Industrial IoT Applications. *IEEE Commun. Lett*. 2023, 27. Pp. 386–390.

40. Ortiz J.C.G., Silvestre-Blanes J., Sempere-Payá V.M., Frau D.C. Evaluation of improvements in BLE Mesh through CODED PHY. *In Proceedings of the 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA),* Vasteras, Sweden, 7–10 September 2021. Pp. 1–4.

41. Ndolo A., Çavdar İ.H. Current state of communication systems based on electrical power transmission lines. *J. Electr. Syst. Inf. Technol*. 2021, 8. Pp. 1–10.

42. Noura H.N., Melki R., Chehab A. Fernandez J.H. Efficient and robust data availability solution for hybrid PLC/RF systems. *Comput. Netw*. 2021, 185. P. 107675.

43. Saleem M.S. Development of PLC based communication architecture for battery management system. *In Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, Antwerp, Belgium, 25–28 May 2020. Pp. 1–5.

44. Conti M., Donadel D., Turrin F. A survey on industrial control system testbeds and datasets for security research. *IEEE Commun. Surv. Tutorials*. 2021, 23. Pp. 2248–2294.

45. Min J., Park Y. Performance Enhancement of In-Vehicle 10BASE-T1S Ethernet Using Node Prioritization and Packet Segmentation. *IEEE Access 2022*, 10. Pp. 103286–103295.

46. Sanz A., Ibar J.C., Lacasa L. PLC-RF hybrid communication systems, model and simulation. *In Proceedings of the 2021 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Aachen, Germany, 25–28 October 2021. Pp. 158–163.

47. Ishaq M., Afzal M.H., Tahir S., Ullah K. A Compact Study of Recent Trends of Challenges and Opportunities in Integrating Internet of Things (IoT) and Cloud Computing. *In Proceedings of the 2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, Quetta, Pakistan, 11–12 April 2021. Pp. 1–4.

48. Djonov M., Galabov M., Georgieva-Trifonova T. Solving IoT Security and Scalability Challenges with Blockchain. *In Proceedings of the 2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Ankara, Turkey, 21–23 October 2021. Pp. 52–56.

49. Razzaq A. Microservices Architecture for IoT Applications in the Ocean: Microservices Architecture based Framework for Reducing the Complexity and Increasing the Scalability of IoT Applications in the Ocean. *In Proceedings of the 2020 20th International Conference on Computational Science and Its Applications (ICCSA)*, Cagliari, Italy, 1–4 July 2020. Pp. 87–90.

50. Bansal S., Tomar V. Challenges & Security Threats in IoT with Solution Architectures. *In Proceedings of the 2022 2nd International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, Mathura, India, 21–22 January 2022. Pp. 1–5.

51. Yassein M.B., Hmeidi I., Meqdadi O., Alghazo F., Odat B., AlZoubi O., Smairat

A. Challenges and Techniques of Constrained Application Protocol (CoAP) for Efficient Energy Consumption. *In Proceedings of the 2020 11th International Conference on Information and Communication Systems (ICICS),* Irbid, Jordan, 7–9 April 2020. Pp. 373–377.

52. Foukalas F., Tziouvaras A. Edge Artificial Intelligence for Industrial Internet of Things Applications: An Industrial Edge Intelligence Solution. *IEEE Ind. Electron. Mag*. 2021, 15, Pp. 28–36.

53. Georgiana Dorobantu O., Halunga S. Security threats in IoT. *In Proceedings of the 2020 International Symposium on Electronics and Telecommunications (ISETC),* Timisoara, Romania, 5–6 November 2020. Pp. 1–4.

54. Bonkra A., Dhiman P. IoT Security Challenges in Cloud Environment. *In Proceedings of the 2021 2nd International Conference on Computational Methods in Science & Technology (ICCMST)*, Mohali, India, 17–18 December 2021. Pp. 30–34.

55. Abdul Sattar K., Al-Omary A. A survey: Security issues in IoT environment and IoT architecture. *In Proceedings of the 3rd Smart Cities Symposium (SCS 2020),* Virtual, 21–23 September 2020; Volume 2020. Pp. 96–102.

56. Landum M., Moura M., Reis L. ICT Good Practices in alignment with Green IT. *In Proceedings of the 2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*, Sevilla, Spain, 17–20 June 2020. Pp. 1–6.

57. Ye N., Yu J., Wang A., Zhang R. Help from space: Grant-free massive access for satellite-based IoT in the 6G era. Digit. *Commun. Networks 2022*, 8, Pp. 215–224.

58. Niu H., Lin Z., Chu Z., Zhu Z., Xiao P., Nguyen H.X., Lee I., Al-Dhahir N. Joint Beamforming Design for Secure RIS-Assisted IoT Networks. *IEEE Internet Things J.* 2023, 10, Pp. 1628–1641.

59. Giordani M., Polese M., Mezzavilla M., Rangan S., Zorzi M. Toward 6G Networks: Use Cases and Technologies. *IEEE Commun. Mag*. 2020, 58, Pp. 55–61.

60. Qadir Z., Le K.N., Saeed N., Munawar H.S. Towards 6G Internet of Things: Recent advances, use cases, and open challenges. *ICT Express 2022*.

61. Kök İ., Okay F.Y., Özdemir S. FogAI: An AI-supported fog controller for Next Generation IoT. *Internet Things 2022*, 19, P. 100572.

62. Tegos S.A., Diamantoulakis P.D., Lioumpas A.S., Sarigiannidis P.G., Karagiannidis G.K. Slotted ALOHA with NOMA for the next generation IoT. *IEEE Trans. Commun*. 2020, 68, Pp. 6289–6301.

63. Lin Z., Lin M., de Cola T., Wang J.B., Zhu W.P., Cheng J. Supporting IoT With Rate-Splitting Multiple Access in Satellite and Aerial-Integrated Networks. *IEEE Internet Things J*. 2021, 8, Pp. 11123–11134.

64. Kumar A., Li F.Y., Martinez-Bauset J. Revealing the Benefits of Rate-Splitting Multiple Access for Uplink IoT Traffic. In Proceedings of the GLOBECOM 2022—2022 *IEEE Global Communications Conference,* Rio de Janeiro, Brazil, 4–8 December 2022. Pp. 111–116.

65. Liu H., Tsiftsis T.A., Kim K.J., Kwak K.S., Poor H.V. Rate splitting for uplink NOMA with enhanced fairness and outage performance. *IEEE Trans. Wirel. Commun*. 2020, 19, Pp. 4657–4670.

66. Agrawal N., Bansal A., Singh K., Li C.P., Mumtaz S. Finite Block Length Analysis of RIS-Assisted UAV-Based Multiuser IoT Communication System With Non-Linear EH. *IEEE Trans. Commun*. 2022, 70, Pp. 3542–3557.

67. Bansal A., Singh K., Li C.P. Analysis of hierarchical rate splitting for intelligent reflecting surfaces-aided downlink multiuser MISO communications. *IEEE Open J. Commun. Soc*. 2021, 2, Pp. 785–798.

68. Ruan Y., Li Y., Zhang R., Cheng W., Liu C. Cooperative Resource Management for Cognitive Satellite-Aerial-Terrestrial Integrated Networks Towards IoT. *IEEE Access*. 2020, 8, Pp. 35759–35769.

69. Zhou D., Gao S., Liu R., Gao F., Guizani M. Overview of development and regulatory aspects of high altitude platform system. *Intell. Converg. Networks*. 2020, 1, Pp. 58–78.

70. Qin P., Zhu Y., Zhao X., Feng X., Liu J., Zhou Z. Joint 3D-Location Planning and Resource Allocation for XAPS-Enabled C-NOMA in 6G Heterogeneous Internet of Things. *IEEE Trans. Veh. Technol*. 2021, 70, Pp. 10594–10609.

71. Zare M., Elmi Sola Y. Hasanpour H. Towards distributed and autonomous IoT service placement in fog computing using asynchronous advantage actor-critic algorithm. J.

King Saud Univ. *Comput. Inf. Sci*. 2023, 35, Pp. 368–381.

72. Gomes E., Costa F., De Rolt C., Plentz P., Dantas M. A Survey from Real-Time to Near Real-Time Applications in Fog Computing Environments. Telecom 2021, 2, Pp. 489–517.

73. Alghamdi I., Anagnostopoulos C., Pezaros D.P. Data quality-aware task offloading in Mobile Edge Computing: An Optimal Stopping Theory approach. *Future Gener. Comput. Syst.* 2021, 117, Pp. 462–479.

74. Baek J., Kaddoum G. Online partial offloading and task scheduling in SDN-Fog networks with deep recurrent reinforcement learning. *IEEE Internet Things J*. 2021, 9, Pp. 11578–11589.

75. Chen J., Yang Y., Wang C., Zhang H., Qiu C., Wang X. Multitask offloading strategy optimization based on directed acyclic graphs for edge computing. *IEEE Internet Things J*. 2021, 9, Pp. 9367–9378.

76. Zhou C., Wu W., He H., Yang P., Lyu F., Cheng N., Shen X. Deep Reinforcement Learning for Delay-Oriented IoT Task Scheduling in SAGIN. *IEEE Trans. Wirel. Commun*. 2021, 20, 911–925. https://doi.org/10.48550/arXiv.2010.01471

77. Qin P., Fu Y., Zhao X., Wu K., Liu J., Wang M. Optimal Task Offloading and Resource Allocation for C-NOMA Heterogeneous Air-Ground Integrated Power Internet of Things Networks. *IEEE Trans. Wirel. Commun*. 2022, 21, Pp. 9276–9292. http://dx.doi.org/10.1109/TWC.2022.3175472

78. Tang F., Hofner H., Kato N., Kaneko K., Yamashita Y., Hangai M. A. Deep Reinforcement Learning-Based Dynamic Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN). *IEEE J. Sel. Areas Commun*. 2022, 40, Pp. 276–289. http://dx.doi.org/10.1109/JSAC.2021.3126073

79. Al Ridhawi I., Otoum S. Supporting Next-Generation Network Management with Intelligent Moving Devices. *IEEE Network* 2022, 36, Pp. 8–15. https://doi.org/10.1109/MNET.009.2100585

80. Liu J., Zhao X., Qin P., Geng S., Meng S. Joint Dynamic Task Offloading and Resource Scheduling for WPT Enabled Space-Air-Ground Power Internet of Things. *IEEE Trans. Netw. Sci. Eng*. 2022, 9, Pp. 660–677.

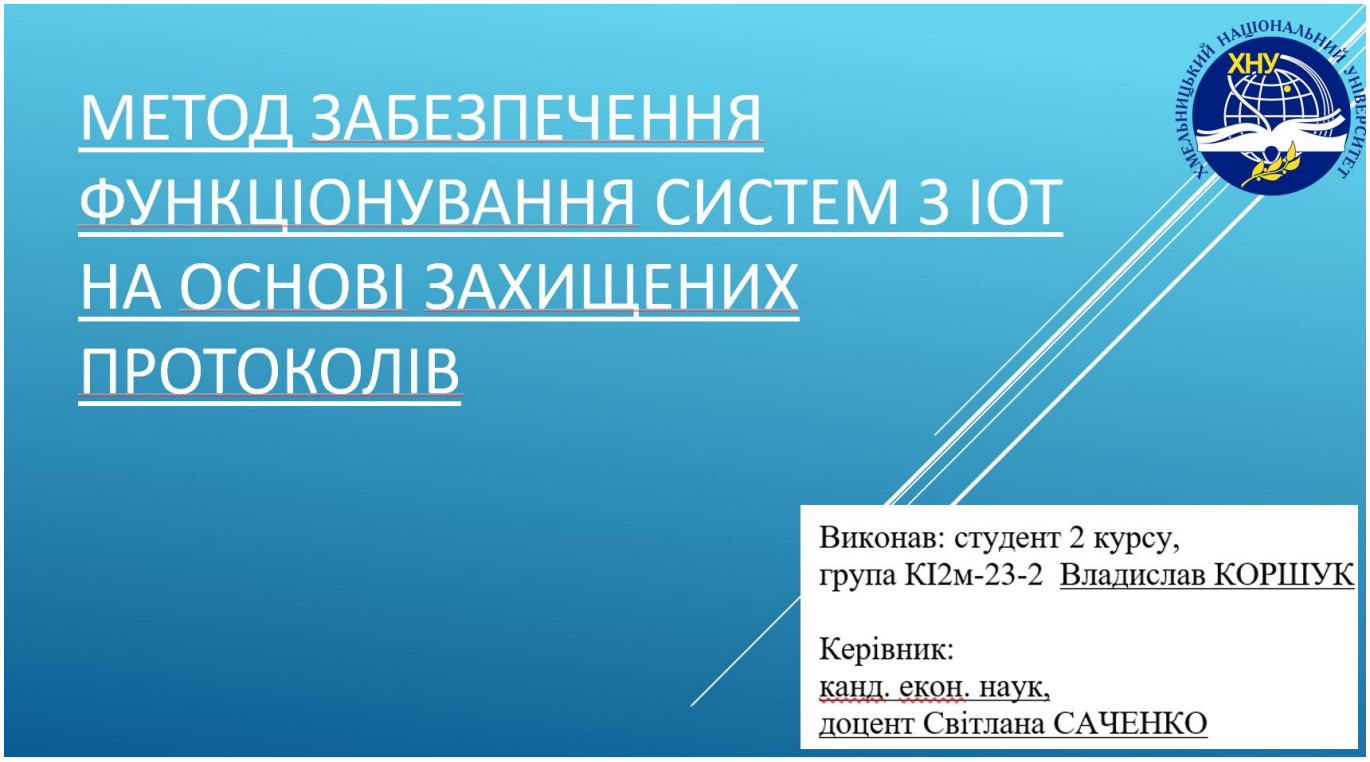
http://dx.doi.org/10.1109/TNSE.2021.3130251

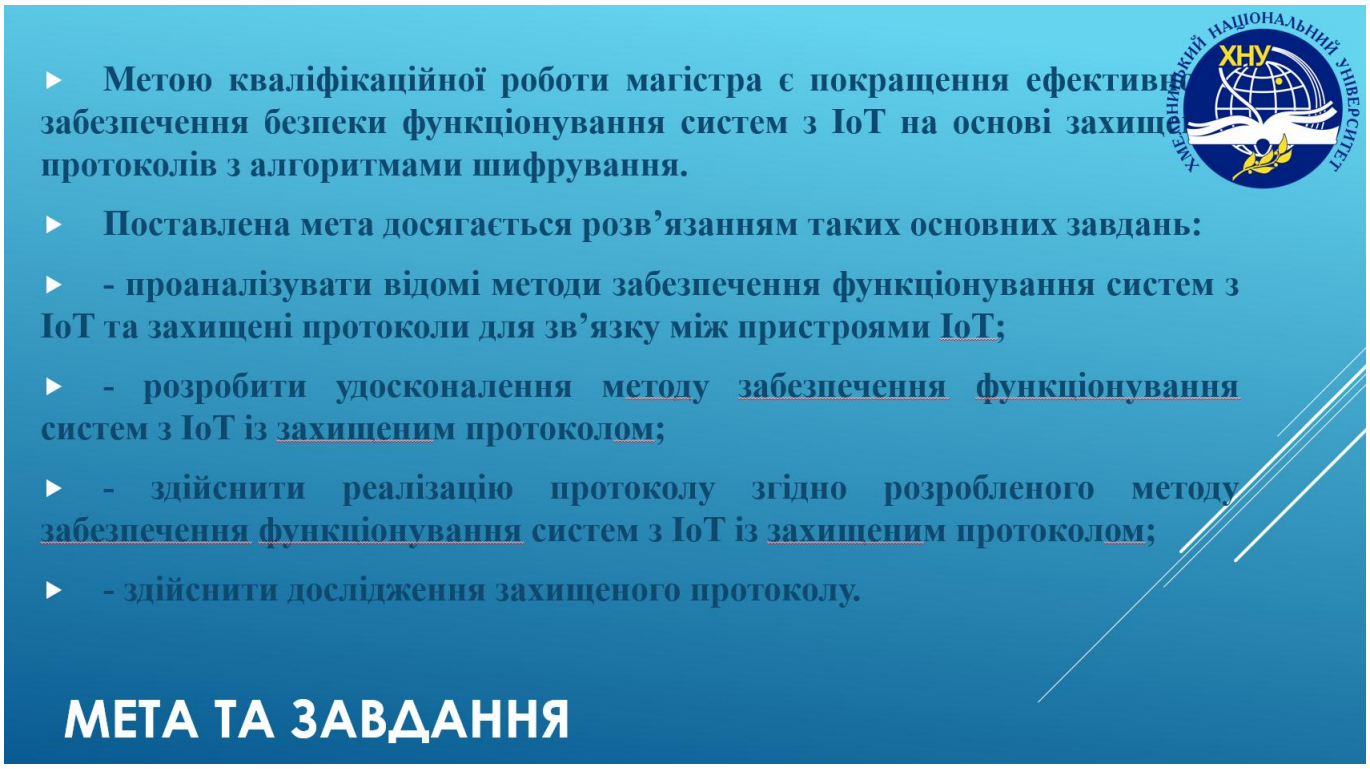
81. M-Bus OMS-Group: https://m-bus.com/. Дата звернення 13.01.2025.

82. Коршук В.Р., Віжевський П.В., Сорочинський О.Ю. Метод забезпечення функціонування систем з IoT на основі захищених протоколів / Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024». Хмельницький, 2024, С. 303-306. https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2024-corpuspaper.pdf

Додаток А

Презентація роботи



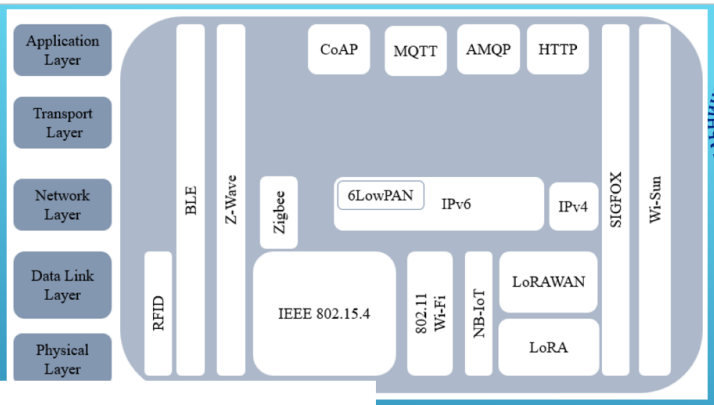


- **Об’єктом дослідження є процес забезпечення функціонування систем з IoT.**
- **Предметом дослідження є методи та протоколи забезпечення функціонування систем з IoT.**
- **Наукова новизна отриманих результатів:**
- **- розроблено новий метод криптографічного захисту від вразливостей в апаратному забезпеченні, в якому на відміну від відомих було розширено його межі застосування для внесених сторонніх знаків в текст.**
- **На основі проведених досліджень розроблено метод та протокол забезпечення функціонування систем з IoT.**
- **Практична значимість отриманих результатів полягає у розроблені захищеного протоколу забезпечення функціонування систем з IoT.**
- **Для розв’язання поставлених задач використовувалися методи криптографії, методи виявлення вразливостей, методи забезпечення функціонування систем з IoT.**

ОБ’ЄКТ, ПРЕДМЕТ

- Протоколи комунікації — це набір правил і стандартів, які визначають спосіб обміну даними між пристроями або програмами в мережі. Вони забезпечують узгодженість і правильність передачі інформації, дозволяючи різним системам взаємодіяти одна з одною.
- Основні аспекти протоколів комунікації:
- **1) формат даних, який** визначає структуру передаваної інформації;
- **2) синхронізація** забезпечує узгодженість передачі й отримання даних;
- **3) контроль помилок** виявляє і коригує помилки під час передачі;
- **4) аутентифікація** перевіряє дійсність учасників комунікації;
- **5) шифрування** забезпечує безпеку переданих даних.

АКТУАЛЬНІСТЬ РОБОТИ



Ключові особливості протоколів IoT:

1) енергоефективність;

2) мала пропускна здатність;

3) підтримка роботи в мережах з великою кількістю пристроїв;

4) масштабованість та сумісність.

Архітектуру стеку IoT зображена на рисунку.

ПЕРЕВАГИ ТА НЕДОЛІКИ РІШЕНЬ

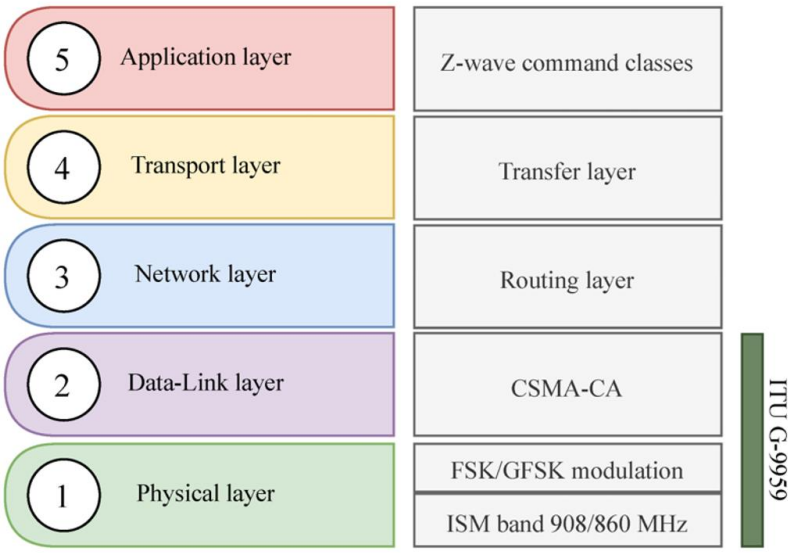


Рисунок 2. Порівняння стека Z-хвилі зі стеком OSI [2]

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Метод забезпечення функціонування систем з Інтернету речей (IoT) на основі захищених протоколів є ключовим для підтримки безпеки, конфіденційності та стійкості системи. Основи цього методу включають такі кроки:

1. Аналіз загроз і вимог до безпеки.
2. Використання захищених протоколів зв'язку.
3. Аутентифікація та управління ідентифікацією.
4. Контроль доступу.
5. Захищене зберігання даних.
6. Моніторинг та оновлення.
7. Використання криптографії.
8. Підтримка стійкості до атак.
9. Використання стандартів безпеки.
10. Навчання та інформування користувачів.

МЕТОД

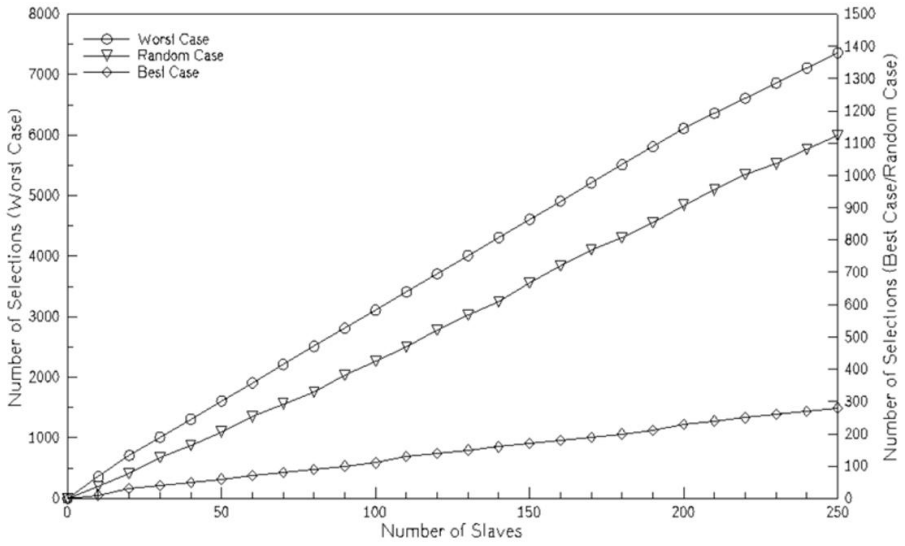


Рисунок 4.2 - Графік залежності величини повідомлення від кількості пристроїв IoT [81]

Орієнтовні енергетичні витрати для поширених алгоритмів шифрування:

Алгоритм	Енергоспоживання (на 1 блок/операцію)	Примітка
AES-128	~20-60 мкДж	Ефективний для симетричного шифрування
ECC (ключ 256 біт)	~5-10 мДж	Висока енерговитратність, застосовується рідше
SHA-256	~50-150 мкДж	Використовується для хешування
RSA (ключ 2048 біт)	~30-50 мДж	Дуже енерговитратний, не рекомендується для IoT

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТУ

Порівняння витрат енергії для різних алгоритмів шифрування

Алгоритм	Тип використання	Енергоспоживання за одну операцію	Частота	Денне енергоспоживання
AES-128	Шифрування даних	~30 мкДж	96 передач	11.52 мДж
ECC-256	Обмін ключами	~6 мДж	1 раз	6 мДж
RSA-2048	Автентифікація	~40 мДж	1 раз	40 мДж
SHA-256	Перевірка цілісності	~100 мкДж	24 передачі	2.4 мДж

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТУ

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень розроблено новий метод забезпечення функціонування систем з IoT на основі захищених протоколів та отримано такі результати.

- 1. Проаналізовано відомі методи та засоби забезпечення функціонування систем з IoT на основі захищених протоколів.
- 2. Розроблено забезпечення функціонування систем з IoT на основі захищених протоколів.
- 3. Здійснено реалізацію протоколу згідно розробленого методу забезпечення функціонування систем з IoT на основі захищених протоколів.
- 4. Досліджено розроблений захищений протокол функціонування пристроїв IoT в контексті кіберзагроз та додаткових витрат енергії на здійснення шифрування повідомлень.

ПУБЛІКАЦІЯ

- ✓ Коршук В.Р., Віжевський П.В., Сорочинський О.Ю. Метод забезпечення функціонування систем з IoT на основі захищених протоколів / Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024». Хмельницький, 2024, С. 303-306. https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2024-corpuspaper.pdf

Додаток Б

Публікація за результатами роботи

Сертифікат № 2024-031-1

Міністерство освіти і науки України
Хмельницький національний університет

АПКН 2024
АКТУАЛЬНІ ПРОБЛЕМИ КОМП'ЮТЕРНИХ НАУК

СЕРТИФІКАТ

Коршук Владислав Русланович

учасник XVI Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2024»
24 години участі (0,8 ECTS credits)

Голова оргкомітету АПКН-2024

Олег СИНЮК
проректор Хмельницького національного університету з наукової роботи, доктор технічних наук, професор

м. Хмельницький
15-16 листопада 2024

E-mail: apkt.khnu@gmail.com

УДК 004.45

Коршук В.Р., Віжевський П.В., Сорочинський О.Ю.

Хмельницький національний університет

МЕТОД ЗАБЕЗПЕЧЕННЯ ФУНКЦІОНУВАННЯ СИСТЕМ З IOT НА ОСНОВІ ЗАХИЩЕНИХ ПРОТОКОЛІВ

Розширення інтелектуального вимірювання в екосистемі Інтернету речей (IoT) недостатньо оцінює потребу в надійних протоколах безпеки, які захищають передачу даних, підвищення ефективності пристрою. Протокол пристроїв IoT є ключовим протоколом для віддаленого лічильника зчитування в комунальних системах, таких як лічильники газу, води та тепла, стикається із значною безпекою виклики. Ця робота представляє метод підвищення безпеки шляхом інтеграції створення протоколу, який захищає пристрої від уразливостей і оптимізує енергетичні обмеження пристроїв IoT.

Expanding the intellectual measurement in the Internet Ecosystem (IOT) does not sufficiently evaluate the need for reliable security protocols that protect data transmission, and increase the efficiency of the device. The IOT device protocol is a key protocol for a remote reading meter in communal systems, such as gas, water and heat meters, faced with considerable safety of the call. This work represents a safety improvement method by integrating a protocol that protects devices from vulnerability and optimizes the energy constraints of IOT devices.

Захищені протоколи для пристроїв Інтернету речей (IoT) є критично важливими для забезпечення безпеки даних [1] і приватності, оскільки пристрої IoT можуть бути вразливими до кібератак. Розглянемо основні протоколи, які використовуються для захисту IoT-пристроїв:

1. TLS/SSL (Transport Layer Security / Secure Sockets Layer). TLS (та його попередник SSL) забезпечує шифрування даних, що передаються між IoT-пристроєм і сервером, захищаючи від перехоплення даних або атак типу "людина посередині" (MITM). Використовується для забезпечення безпечного з'єднання між пристроями, такими як смарт-лічильники чи датчики, та хмарними серверами або іншими пристроями.

2. DTLS (Datagram Transport Layer Security). Версія TLS, оптимізована для UDP, що дозволяє забезпечувати захищене з'єднання для додатків реального часу, де важливіше швидкість і ефективність передачі, ніж надійність TCP. Використовується для IoT-пристроїв, які працюють з реальним часом або малопотужними мережами, як-от смарт-освітлення чи відеоспостереження.

3. MQTT (Message Queuing Telemetry Transport) з TLS. Протокол для обміну повідомленнями між IoT-пристроями через посередницький сервер (брокер), який підтримує TLS для забезпечення безпеки даних. Використовується в системах, де необхідно передавати дані від численних пристроїв у хмару або сервер. Наприклад, у смарт-будинках для передачі даних про стан систем.

4. CoAP (Constrained Application Protocol) з DTLS. Легковаговий протокол, який розроблений для пристроїв з обмеженими ресурсами, таких як сенсори, та працює поверх UDP. DTLS додає шифрування та автентифікацію. Використовується для невеликих IoT-пристроїв у сітках датчиків [2], зокрема для збору даних про екологію або стан здоров'я.

5. HTTPS (Hypertext Transfer Protocol Secure). HTTPS – це розширення HTTP з використанням шифрування TLS. Хоча він не є спеціалізованим для IoT, його можна використовувати для захисту передачі даних між пристроями IoT і веб-серверами. Використовується в додатках, де передача даних з пристроїв IoT здійснюється через веб-інтерфейс, наприклад, для розумних термостатів або домашніх камер безпеки.

6. ZigBee з безпекою. Протокол для низькопотужних бездротових мереж, який підтримує AES-128 шифрування для захисту переданих даних. Використовується в системах домашньої автоматизації (розумне освітлення, опалення), а також у промислових IoT рішеннях.

7. LoRaWAN (Long Range Wide Area Network). Протокол для передачі даних на великі відстані між пристроями IoT з мінімальним енергоспоживанням. LoRaWAN підтримує шифрування AES для забезпечення безпеки даних. Використовується [3] для розгортання сенсорних мереж на великих площах, наприклад, у сільському господарстві або логістиці.

8. OTrP (Open Trust Protocol). Протокол для забезпечення захищеного дистанційного управління пристроями IoT, зокрема їх налаштуванням та оновленням. Використовується для забезпечення безпеки при управлінні великими кількостями IoT-пристроїв у мережі, зокрема для корпоративних або індустріальних систем.

9. IPsec (Internet Protocol Security). Протокол, який забезпечує шифрування і автентифікацію на рівні IP. Використовується для захисту пакетів даних, що передаються по мережі. Використовується для захисту IoT-пристроїв, що працюють в мережах IP, наприклад, у промислових IoT системах або інфраструктурних мережах.

10. Lightweight M2M (LwM2M). Протокол управління для IoT, розроблений для пристроїв з обмеженими ресурсами. Він забезпечує шифрування та автентифікацію через протоколи типу DTLS. Використовується для управління та

моніторингу пристроїв IoT, таких як датчики та контрольні пристрої в розумних містах або в системах моніторингу здоров'я.

Метою роботи є розробка захищеного протоколу для IoT-пристроїв.

Вибір захищеного протоколу [4] для IoT-пристроїв залежить від конкретних вимог системи, таких як обмеженість ресурсів, необхідний рівень безпеки, топологія мережі та інші фактори.

Запропоновано протокол, який ефективно балансує покращену безпеку та ефективність, збереження конфіденційності, цілісності та доступності даних в інтелектуальному обліку без шкоди для продуктивності. Тестування безпеки проти спуфінгу підтвердило стійкість цього нового протоколу, тим самим підвищивши рамки безпеки. Це дослідження не лише створює більш надійну основу для інтелектуального вимірювання, але також створює прецедент для майбутніх досліджень інтеграції шифрування в середовищах IoT

Перелік посилань

1. Song, S., Gao, N., Zhang, Y. et al. BRITD: behavior rhythm insider threat detection with time awareness and user adaptation. Cybersecurity 7, 2 (2024). https://doi.org/10.1186/s42400-023-00190-9
2. Xu, K., Cheng, G. F3l: an automated and secure function-level low-overhead labeled encrypted traffic dataset construction method for IM in Android. Cybersecurity 7, 1 (2024). https://doi.org/10.1186/s42400-023-00185-6
3. Habiba Sultana, A.H.M. Kamal, Tasnim Sakib Apon, Md. Golam Rabiul Alam, Increasing embedding capacity of stego images by exploiting edge pixels in prediction error space, Cyber Security and Applications, Volume 2, 2024, 100028, ISSN 2772-9184, https://doi.org/10.1016/j.csa.2023.100028. (https://www.sciencedirect.com/science/article/pii/S2772918423000164)
4. Cremer F, Sheehan B, Fortmann M, Kia AN, Mullins M, Murphy F, Materne S. Cyber risk and cybersecurity: a systematic review of data availability. Geneva Pap Risk Insur Issues Pract. 2022;47(3):698-736. doi: 10.1057/s41288-022-00266-6. Epub 2022 Feb 17. PMID: 35194352; PMCID: PMC8853293.

Додаток В

Результати перевірки на плагіат

Anti-Plagiarism v-15.260 Educational

Максимальне співпадіння з одним документом 7.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 9%**

	Документ		Сумарний збіг по Базі Даних	
ID: 229473 Назва: МКР Метод забезпечення функціонування систем з IoT на основі захищених протоколів Додано в БД: 2025-04-15 Автора: Владислав КОРШУК Керівники: Світлана САЧЕНКО Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	147100	1142	11869 (8%)	100 (9%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

StrikePlagiarism.com

Дата звіту 4/15/2025

Дата редагування 4/15/2025

Документ прийнятий

Звіт подібності

метадані

Назва організації

Khmelnytskyi National University

Заголовок

КОРШУК_Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Автор

Владислав КОРШУК Науковий керівник / Експерт

підрозділ

Кафедра комп'ютерної інженерії та інформаційних систем

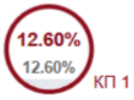
Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		1
Інтервали		0
Мікропробіли		163
Білі знаки		1
Парафрази (SmartMarks)	a	83

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



25	20877	159711
Довжина фрази для коефіцієнта подібності 2	Кількість слів	Кількість символів

Додаток Г

Програмний код побудови графа зв’язків між пристроями IoT

```
import networkx as nx
import matplotlib.pyplot as plt

# Створення графа
graph = nx.Graph()

# Додавання пристроїв (вершин)
devices = ["Sensor1", "Sensor2", "Camera1", "SmartLight1", "Gateway", "Server"]
graph.add_nodes_from(devices)

# Додавання зв'язків (ребер)
connections = [
   ("Sensor1", "Gateway"),
   ("Sensor2", "Gateway"),
   ("Camera1", "Gateway"),
   ("Gateway", "Server"),
   ("SmartLight1", "Gateway"),
   ("Server", "Camera1")  # Приклад двостороннього зв'язку
]
graph.add_edges_from(connections)

# Візуалізація графа
plt.figure(figsize=(8, 6))
nx.draw(graph, with_labels=True, node_color='skyblue', node_size=2000, font_size=12,
font_weight='bold')
plt.title("Граф IoT-пристроїв")
plt.show()
```

Додаток В

Програмний код для комунікації між пристроями IoT на Python

```
import time
import random
import paho.mqtt.client as mqtt

# MQTT брокер
BROKER = "mqtt.example.com"
PORT = 1883
TOPIC_PREFIX = "iot/meters"

# Дані пристроїв
DEVICES = [
   {"id": "water_meter_1", "type": "water"},
   {"id": "gas_meter_1", "type": "gas"},
   {"id": "electricity_meter_1", "type": "electricity"},
]

# Функція для створення випадкових показів
```

```
def generate_reading(device_type):
    if device_type == "water":
        return round(random.uniform(0.1, 5.0), 2)  # м³
    elif device_type == "gas":
        return round(random.uniform(0.1, 3.0), 2)  # м³
    elif device_type == "electricity":
        return round(random.uniform(0.1, 15.0), 2)  # кВт·год
    return 0

# Callback для підключення
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT Broker!")
    else:
        print(f"Failed to connect, return code {rc}")

# Налаштування MQTT клієнта
client = mqtt.Client()
client.on_connect = on_connect

try:
    client.connect(BROKER, PORT, 60)
except Exception as e:
    print(f"Error connecting to broker: {e}")
    exit(1)

client.loop_start()

# Цикл для передачі даних
try:
    while True:
        for device in DEVICES:
            reading = generate_reading(device["type"])
            payload = {
                "device_id": device["id"],
                "type": device["type"],
                "reading": reading,
                "timestamp": int(time.time())
            }
            topic = f"{TOPIC_PREFIX}/{device['type']}/{device['id']}"
            client.publish(topic, str(payload))
            print(f"Published to {topic}: {payload}")
        time.sleep(86400)  # Передача показів раз на добу
except KeyboardInterrupt:
    print("Stopping the script...")
finally:
    client.loop_stop()
    client.disconnect()
```

Захищений протокол

Код з шифруванням та безпровідною передачею повідомлень на Python

```
import time
```

```
import random
import json
from cryptography.fernet import Fernet
import paho.mqtt.client as mqtt

# MQTT брокер
BROKER = "mqtt.example.com"
PORT = 1883
TOPIC_PREFIX = "iot/meters"

# Генерація або використання існуючого ключа шифрування
# Збережіть цей ключ для всіх пристроїв, щоб вони могли декодувати повідомлення
ENCRYPTION_KEY = Fernet.generate_key()
cipher_suite = Fernet(ENCRYPTION_KEY)

# Дані пристроїв
DEVICES = [
    {"id": "water_meter_1", "type": "water"},
    {"id": "gas_meter_1", "type": "gas"},
    {"id": "electricity_meter_1", "type": "electricity"},
]

# Функція для створення випадкових показів
def generate_reading(device_type):
    if device_type == "water":
        return round(random.uniform(0.1, 5.0), 2)  # м³
    elif device_type == "gas":
        return round(random.uniform(0.1, 3.0), 2)  # м³
    elif device_type == "electricity":
        return round(random.uniform(0.1, 15.0), 2)  # кВт·год
    return 0

# Функція для шифрування повідомлення
def encrypt_message(message):
    return cipher_suite.encrypt(message.encode()).decode()

# Функція для розшифрування повідомлення (для отримувача)
def decrypt_message(encrypted_message):
    return cipher_suite.decrypt(encrypted_message.encode()).decode()

# Callback для підключення
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT Broker!")
    else:
        print(f"Failed to connect, return code {rc}")

# Налаштування MQTT клієнта
client = mqtt.Client()
client.on_connect = on_connect

try:
```

```
    client.connect(BROKER, PORT, 60)
except Exception as e:
    print(f"Error connecting to broker: {e}")
    exit(1)

client.loop_start()

# Цикл для передачі даних
try:
    while True:
        for device in DEVICES:
            reading = generate_reading(device["type"])
            payload = {
                "device_id": device["id"],
                "type": device["type"],
                "reading": reading,
                "timestamp": int(time.time())
            }
            # Серіалізація і шифрування
            serialized_payload = json.dumps(payload)
            encrypted_payload = encrypt_message(serialized_payload)

            topic = f"{TOPIC_PREFIX}/{device['type']}/{device['id']}"
            client.publish(topic, encrypted_payload)
            print(f"Published to {topic}: {encrypted_payload}")
        time.sleep(86400)  # Передача показів раз на добу
except KeyboardInterrupt:
    print("Stopping the script...")
finally:
    client.loop_stop()
    client.disconnect()
```

Завідувачу кафедри КІІС,
доктору філософії, доц. Ользі ПАВЛОВІЙ

Владислав КОРШУК

ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2м-23-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

25 квітня 2025 року

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Владислав КОРШУК

Тема: Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень -; кількість сторінок записки 108

1. Короткий зміст роботи та прийнятих рішень У роботі розроблено метод та засоби координації точок доступу в мережах Wi-Fi

2. Висновок про відповідність роботи дипломному завданню

Кваліфікаційна робота відповідає виданому завданню

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі подано об'єкт та предмет дослідження, мету, наукову новизну та практичну цінність роботи, а також характеристику структури роботи.

У першому розділі проведено аналіз відомих рішень щодо протоколів забезпечення функціонування систем з IoT.

У другому розділі здійснено дослідження предметної області та визначено стратегію захищеного протоколу забезпечення функціонування систем з IoT, а також розроблено модель системи з пристроями IoT.

У третьому розділі розроблено метод забезпечення функціонування систем з IoT із захищеним протоколом.

У четвертому розділі здійснено розроблення захищеного протоколу забезпечення функціонування систем з IoT та дослідження його стійкості до кіберзагроз та витрат енергії батарей для шифрування повідомлень.У висновках підведено підсумки досягнення результатів з розв'язання завдань дослідження.

4. Позитивні сторони роботи: ___

5. Негативні сторони роботи: немає.

6. Оцінка графічного оформлення та пояснювальної записки роботи: -

7. Відгук про роботу в цілому: __Робота виконана на належному рівні.

8. Інші зауваження: ___–

9. Оцінка дипломної роботи:

Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи вважаю, що робота заслуговує оцінки «добре» 4,00 (С)

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) __Корецька Людмила Олександрівна, к.т.н., доцент кафедри АКІТР ХНУ

"_1_" __травня__2025р. _____

Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Владислав КОРШУК

Співавтор:

Назва: КОРШУК_Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:12.6%

Коефіцієнт подібності 2:11.7%

Мікропробіли: 163

Заміна букв: 1

Інтервали: 0

Білі знаки: 1

Дата створення звіту: 2025-04-15 13:03:53.0

Після аналізу Звіту подібності констатую наступне:

☒ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

2025-04-15 Доцент Андрій Нічепорук

Дата експерт

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод забезпечення функціонування систем з IoT на основі захищених протоколів

Автор: Владислав КОРШУК

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Світлана САЧЕНКО, к.е.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) окремі виявлені збіги є загальновживаними фразами або виразами, про що свідчить посилання системи на збіг з 10-40 джерелами на один фрагмент речення;
2) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості, складає менше 12,6% і адресується до джерел з інтернету та бібліотеки списку використаних джерел (переліку), що, з урахуванням наведених обґрунтувань, відповідає характеру завдання і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ Світлана САЧЕНКО

Гарант ОП _____ Олег САВЕНКО

Завідувач кафедри КІІС _____ Ольга ПАВЛОВА