

Міністерство освіти і науки України  
Державний заклад  
«Луганський національний університет імені Тараса Шевченка»

Навчально-науковий інститут математики та інформаційних технологій

Кафедра інформаційних технологій та систем

**Щебетов Віталій Анатолійович**

**КРИПТОГРАФІЧНИЙ ЗАХИСТ ДАНИХ В БЕЗДРОТОВИХ  
СЕНСОРНИХ МЕРЕЖАХ**

**кваліфікаційна робота**

**здобувача вищої освіти другого (магістерського) рівня**

**освітньої програми «Комп'ютерні мережі»**

**за спеціальністю 123 Комп'ютерна інженерія**

Особистий підпис \_\_\_\_\_ Віталій Щебетов

Науковий керівник \_\_\_\_\_ Микола СЕМЕНОВ,  
кандидат педагогічних наук, доцент  
кафедри інформаційних технологій  
та систем

Завідувач кафедри \_\_\_\_\_ Микола СЕМЕНОВ,  
кандидат педагогічних наук, доцент  
кафедри інформаційних технологій  
та систем

Полтава – 2024

## **АНОТАЦІЯ**

**Тема:** Криптографічний захист даних в бездротових сенсорних мережах.

**Спеціальність:** 123 «Комп'ютерна інженерія».

**Установа:** ЛНУ імені Тараса Шевченка, 2024р.

**Магістерська робота містить:** 71 с., 12 рис., 9 табл., 45 джерел.

**Об'єкт дослідження** – бездротові сенсорні мережі.

**Предмет дослідження** – криптографічний захист даних в бездротових сенсорних мережах.

**Мета роботи** – розробка та впровадження нового криптографічного методу захисту даних в бездротових сенсорних мережах з метою забезпечення високого рівня конфіденційності, цілісності та доступності інформації в умовах обмежених ресурсів вузлів та в умовах розподіленої й динамічної природи мережі.

**Результати роботи** – у ході дослідження проведено огляд існуючих криптографічних методів захисту в бездротових сенсорних мережах; на основі отриманих результатів були визначені вимоги до безпеки в бездротових сенсорних мережах, і розроблені рекомендації щодо їх виконання. Розроблено криптографічний метод та експериментально перевірено його ефективність та надійність для захисту даних у бездротових сенсорних мережах.

**Ключові слова:** БЕЗДРОТОВА СЕНСОРНА МЕРЕЖА, КРИПТОГРАФІЯ, АЛГОРИТМ, ПРОТОКОЛ, АВТЕНТИФІКАЦІЯ, МЕРЕЖЕВА МОДЕЛЬ, КЛЮЧ, ЦИФРОВИЙ ПІДПИС

## ANNOTATION

**Theme:** Cryptographic Data Protection in Wireless Sensor Networks

**Speciality:** 123 "Computer Engineering".

**Institution:** Luhansk Taras Shevchenko National University (LTSNU), 2024 year.

**Master's work of:** 71 p., 12 im., 9 tables, 45 sources.

**Research object** – Wireless Sensor Networks.

**Research subject** – Cryptographic Data Protection in Wireless Sensor Networks.

**Objective of the study** – to develop and implement a new cryptographic method for data protection in wireless sensor networks to ensure a high level of confidentiality, integrity, and availability of information under the constraints of node resources and the distributed and dynamic nature of the network.

**Results of the study** – an overview of existing cryptographic methods for protection in wireless sensor networks was conducted. Based on the obtained results, security requirements for wireless sensor networks were identified, and recommendations for their implementation were developed. A cryptographic method was designed and experimentally verified for its effectiveness and reliability in protecting data in wireless sensor networks.

**Keywords:** WIRELESS SENSOR NETWORK, CRYPTOGRAPHY, ALGORITHM, PROTOCOL, AUTHENTICATION, NETWORK MODEL, KEY, DIGITAL SIGNATURE

## ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. КРИПТОГРАФІЧНІ МЕТОДИ ЗАХИСТУ В БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖАХ .....	9
1.1. Симетричне і асиметричне шифрування .....	9
1.2. Хеш-функції.....	28
1.3. Протоколи аутентифікації та обміну ключами.....	33
Висновки до розділу 1 .....	39
РОЗДІЛ 2. РОЗРОБКА АЛГОРИТМУ КРИПТОГРАФІЧНОГО ЗАХИСТУ ДЛЯ БЕЗДРОТОВОЇ СЕНСОРНОЇ МЕРЕЖІ .....	41
2.1. Визначення вимог до системи захисту даних в бездротових сенсорних мережах .....	41
2.2. Розробка алгоритму та його впровадження .....	46
Висновки до розділу 2 .....	53
РОЗДІЛ 3. АНАЛІЗ І ТЕСТУВАННЯ АЛГОРИТМУ КРИПТОГРАФІЧНОГО ЗАХИСТУ .....	54
3.1. Формування критеріїв аналізу та тестування .....	54
3.2. Валідація розробленої криптографічної методики.....	56
Висновки до розділу 3 .....	63
ВИСНОВКИ.....	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	66

## ВСТУП

В бездротових сенсорних мережах безпека відіграє критичну роль у забезпеченні конфіденційності, цілісності та доступності інформації, що передається. Ці мережі, складені з великої кількості вузлів, що спілкуються через бездротові канали передачі даних, піддаються різноманітним загрозам безпеки, таким як атаки на конфіденційність даних, зловживання трафіком чи навіть фізичний доступ до вузлів.

Безпека в бездротових сенсорних мережах охоплює комплекс заходів, спрямованих на захист інформації від несанкціонованого доступу, модифікації чи втрати. Вона включає в себе розробку та впровадження механізмів шифрування, аутентифікації, контролю доступу, а також заходів з виявлення та усунення вразливостей систем.

Контекст безпеки в бездротових сенсорних мережах розвивається на фоні постійного росту кількості підключених пристроїв та застосування цих мереж у різних сферах життя – від промисловості до медицини та будівництва. Цей контекст вимагає постійного удосконалення методів захисту для забезпечення надійності та безпеки даних у бездротових сенсорних мережах.

Криптографічний захист в бездротових сенсорних мережах є критичним елементом для збереження конфіденційності, цілісності та автентичності даних. Важливість криптографічного захисту в цьому контексті ґрунтується на таких аспектах:

**1. Захист конфіденційності даних.** У бездротових сенсорних мережах, де дані передаються через безпроводний канал, існує велика загроза перехоплення чутливої інформації. Криптографічні методи дозволяють шифрувати дані, забезпечуючи їх конфіденційність і унеможливлюючи зловмисникам зрозуміти чи використовувати перехоплену інформацію.

**2. Забезпечення цілісності даних.** Криптографічні протоколи також забезпечують цілісність даних, що передаються. Вони дозволяють виявляти будь-які зміни або модифікації даних під час передачі, що робить неможливими недозволені зміни в інформації під час трансляції.

**3. Аутентифікація та авторизація.** Криптографічні методи дозволяють перевіряти автентичність вузлів у мережі та контролювати їх доступ до інформації. Це дозволяє запобігати несанкціонованому доступу та забезпечує вірогідність джерела інформації.

**4. Запобігання атакам.** Криптографічний захист ставить перешкоди перед зловмисниками, які намагаються провести атаки, такі як перехоплення, вставка фальшивих даних або відмова в обслуговуванні (DoS). Ефективні криптографічні протоколи можуть ускладнити проведення таких атак та знизити їх успішність.

**5. Дотримання вимог безпеки.** У багатьох сферах, де використовуються бездротові сенсорні мережі (медицина, промисловість, міське управління тощо), існують високі вимоги до захисту даних. Криптографічний захист допомагає відповідати цим вимогам та забезпечує відповідність стандартам безпеки.

Забезпечення безпеки в бездротових сенсорних мережах стикається з низкою проблем і викликів [13, 14], які вимагають уваги та розробки нових стратегій захисту:

- Обмежені ресурси вузлів. Бездротові сенсорні вузли мають обмежені обчислювальні та енергетичні ресурси. Розробка криптографічних протоколів, які б забезпечували високий рівень захисту при обмежених ресурсах, є складною задачею [5].

- Фізична доступність до вузлів. Вузли бездротових сенсорних мереж можуть бути легко доступні для фізичної атаки. Це створює загрозу не лише для самого обладнання, але й для даних, що ними обробляються.

- Нестійкість до відомих атак. Класичні методи криптографічного захисту можуть бути уразливими до певних видів атак, таких як атаки на основі перехоплення трафіку, аналізу витрат енергії та фізичного знищення вузлів [26].

– Розподіленість та динаміка мережі. Сенсорні мережі часто мають розподілену природу та динамічну топологію, що ускладнює виявлення та управління безпекою в таких умовах.

– Збільшення кількості підключених пристроїв. Швидкий розвиток Інтернету речей (IoT) призводить до значного збільшення кількості підключених пристроїв у бездротових мережах, що підвищує загрозу для безпеки даних через розширення атаків векторів.

– Потреба у стандартизації та узгодженості. Наявність багатьох протоколів та підходів до захисту може створювати проблеми у стандартизації та узгодженості в безпеці, що ускладнює взаємодію між різними пристроями та мережами.

Вирішення цих проблем та викликів є ключовим аспектом розробки ефективних та масштабованих методів криптографічного захисту в бездротових сенсорних мережах [32].

**Об’єкт дослідження** – бездротові сенсорні мережі.

**Предмет дослідження** – криптографічний захист даних в бездротових сенсорних мережах.

Метою магістерського дослідження є розробка та впровадження нового криптографічного методу захисту даних в бездротових сенсорних мережах з метою забезпечення високого рівня конфіденційності, цілісності та доступності інформації в умовах обмежених ресурсів вузлів та в умовах розподіленої й динамічної природи мережі.

Для досягнення поставленої мети дослідження необхідно виконати такі завдання:

- 1) провести огляд існуючих криптографічних методів захисту в бездротових сенсорних мережах та визначити їх переваги й обмеження;
- 2) визначити вимоги до безпеки в бездротових сенсорних мережах та розробити рекомендації щодо їх виконання;

- 3) розробити новий криптографічний метод захисту для підвищення ефективності та надійності захисту даних у бездротових сенсорних мережах;
- 4) провести експериментальне тестування розробленого методу для оцінки його ефективності та порівняння з існуючими рішеннями.

Новизна дослідження полягає в розробці гібридного методу, який буде поєднувати переваги різних підходів до криптографічного захисту, забезпечуючи систему, що ефективно захищає дані в бездротових сенсорних мережах. Особливістю цього методу є його адаптивність до різних умов мережі, здатність самостійно вибирати оптимальні параметри шифрування та аутентифікації в залежності від стану мережі та ресурсів, доступних вузлам.

**Методи дослідження.** У роботі використовувалися такі методи: аналітичний – для огляду літературних джерел та існуючих методів захисту в бездротових сенсорних мережах. Методи моделювання та експериментального тестування – для оцінки ефективності розроблених алгоритмів та порівняння їх з існуючими рішеннями. Метод комп'ютерного моделювання – для уточнення параметрів та властивостей розроблених методів захисту.

У першому розділі проведено огляд існуючих криптографічних методів захисту в бездротових сенсорних мережах, виконано аналіз їхніх переваг та недоліків, представлено порівняльний аналіз різних підходів до криптографічного захисту.

У другому розділі було визначено вимоги до системи захисту даних у бездротовій сенсорній мережі та виконано проектування і розробку системи криптографічного захисту.

У третьому розділі було сформовано критерії для аналізу і тестування розробленої методики криптографічного захисту та проведено її комплексну валідацію на відповідність цим критеріям та стійкість до різного роду загроз.



# РОЗДІЛ 1

## КРИПТОГРАФІЧНІ МЕТОДИ ЗАХИСТУ В БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖАХ

У галузі безпеки бездротових сенсорних мереж (WSN) ключовим аспектом є використання ефективних криптографічних методів для захисту конфіденційності, цілісності та доступності даних. Криптографічні методи складають основу захисту інформації, адже пристрої взаємодіють у реальному часі, обмінюючи велику кількість даних. Вони стали основним інструментом для захисту інформації у бездротових сенсорних мережах, забезпечуючи безпеку даних під час їх передачі та зберігання, і використовуються для шифрування, аутентифікації, створення цифрових підписів та інших операцій, що забезпечують захист інформації в мережах IoT [3, 4, 18].

Сучасні технології розвиваються з великою швидкістю, що створює нові виклики для безпеки в таких мережах. Це ставить під сумнів довготривалу ефективність традиційних методів криптографії, які можуть бути недостатньо пристосованими до специфіки даних систем та обмежень ресурсів вузлів. Основним завданням даного розділу є проаналізувати наявні криптографічні методи з огляду на їхню застосовність у бездротових сенсорних мережах. З'ясування, які саме аспекти криптографії є найбільш важливими та ефективними в умовах обмежених ресурсів, є ключовим для забезпечення безпеки даних в цих унікальних та динамічних середовищах.

### 1.1. Симетричне і асиметричне шифрування

Симетричне шифрування – один із фундаментальних підходів до захисту інформації, де використовується один ключ як для шифрування, так і для розшифрування повідомлень [33]. Основний принцип цього типу шифрування полягає в тому, що якщо дані зашифрувати за допомогою ключа **К**, то їх можливо розшифрувати тільки з використанням цього ж ключа **К**.

Розгляд основних алгоритмів симетричного шифрування необхідний, щоб окреслити принципи симетричного шифрування, визначити переваги та

недоліки різних алгоритмів і використати проаналізовану інформацію для виконання завдань цього дослідження.

**AES (Advanced Encryption Standard)** – це алгоритм симетричного блочного шифрування, який у сучасній криптографії є одним з найпоширеніших та надійних і використовується у різних застосуваннях, включаючи захист даних у мережах, криптографічні протоколи та зберігання інформації. AES працює з блоками даних розміром 128 біт. Довжина ключа може бути 128, 192 або 256 біт, що визначає рівень безпеки шифрування.

AES може працювати в різних режимах, таких як ECB, CBC, CTR та ін. Основна мета режимів – забезпечити надійне та ефективне застосування AES для різних типів даних та сценаріїв використання. Режими роботи шифру AES визначають спосіб, яким він застосовується до різних блоків даних:

- **ECB (Electronic Codebook)** – кожен блок даних шифрується незалежно один від одного з використанням того ж ключа.
- **CBC (Cipher Block Chaining)** – перед шифруванням кожен блок комбінується з попереднім зашифрованим блоком даних. Це ускладнює атаки на шифртекст, оскільки зміни в одному блоку поширюються на всі наступні блоки.
- **CTR (Counter)** – використовується для перетворення шифру AES у потоковий шифр. Він застосовує шифрування до послідовності чисел (лічильників), які потім комбінуються з блоками даних за допомогою операції побітового XOR. Це дозволяє паралельно обробляти блоки даних і надає можливість зашифровувати та розшифровувати дані незалежно.
- **OFB (Output Feedback)** – перетворює блоковий шифр в потоковий, дозволяючи шифрувати кожен біт повідомлення окремо. Використовується для схем шифрування потоку даних.
- **CFB (Cipher Feedback)** – шифрувальний блок додається до відкритого тексту, а результат використовується для шифрування наступного

блоку. Це дозволяє шифрувати невеликі порції даних та є варіацією режиму CBC.

- **GCM (Galois/Counter Mode)** – комбінований режим, який поєднує блочний шифр (CTR) з аутентифікацією галуа/повідомленням (GMAC). Використовується для шифрування та перевірки цілісності даних, особливо у мережних протоколах та застосунках, де важлива якість забезпечення послуг (QoS).

Ці режими розширюють функціональні можливості AES, надаючи більш гнучкі та спеціалізовані методи застосування шифрування для різних типів даних та сценаріїв використання [29]. Вибір режиму залежить від конкретних вимог до безпеки та функціональних потреб системи.

Порядок, в якому відбувається шифрування за алгоритмом AES показано на рис. 1.1.

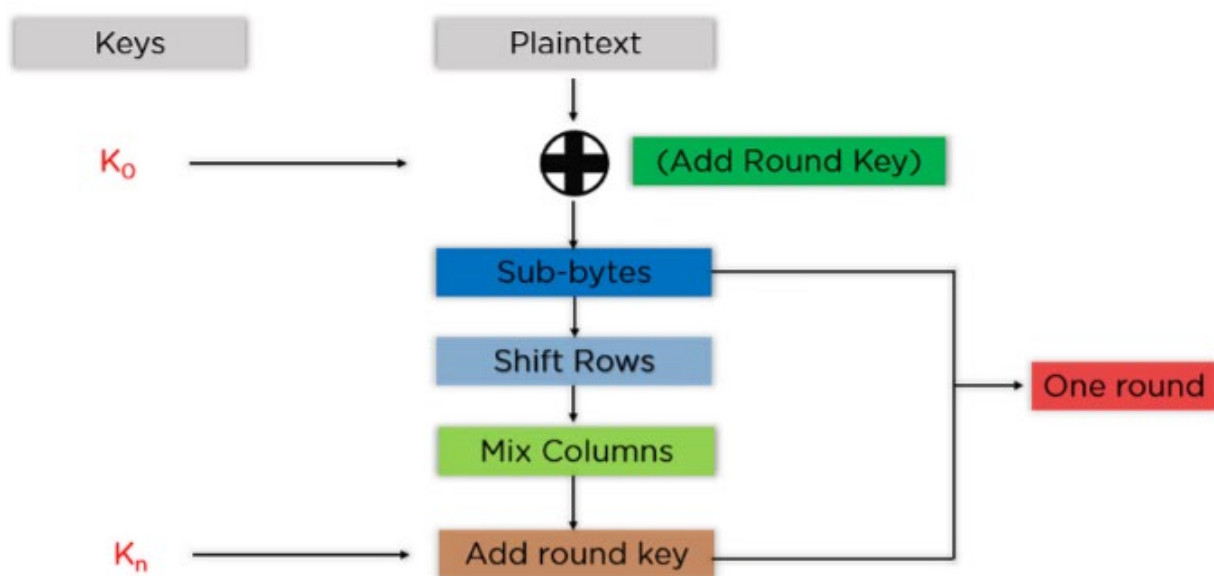


Рис. 1.1. – Етапи шифрування за алгоритмом AES [25]

Шифрування даних в AES починається з додавання ключа, що включає змішування початкового блоку даних з відповідним початковим ключем у внутрішньому стані шифру. Це змішування формує базовий етап для подальших операцій шифрування.

Раунди шифрування є ітеративним процесом, який складається з певної кількості раундів в залежності від довжини ключа (10 раундів для 128-бітного ключа, 12 для 192-бітного та 14 для 256-бітного). Кожен раунд включає певні операції, такі як підстановка байтів, змішування стовпців, зсуви рядків та додавання ключа, що забезпечують необхідний рівень шифрування.

У завершальному раунді відбувається виключно підстановка байтів, змішування стовпців та додавання ключа, без зсувів рядків. Цей останній етап шифрування завершує виконання операцій над блоком даних, призначених для забезпечення високого рівня захисту інформації.

AES відомий своєю високою безпекою, навіть при використанні ключів меншої довжини, та швидкістю роботи, що робить його привабливим для застосування у різних системах та протоколах. Саме тому AES став стандартом для багатьох криптографічних застосувань та продовжує залишатися одним з найефективніших та надійних методів шифрування для захисту інформації.

**DES (Data Encryption Standard).** Це один з перших симетричних алгоритмів шифрування, який використовувався для шифрування та розшифрування даних, і вперше був стандартизований у 1977 році. DES працює з блоками даних розміром 64 біти. Довжина ключа складає 56 біт, проте фактично використовується 64-бітний ключ, оскільки включає в себе 8 бітів для контролю парності. DES використовує режим роботи CBC (Cipher Block Chaining), де кожен блок шифрується залежно від попереднього блоку, утворюючи ланцюг шифрованих блоків.

Шифрування в DES відбувається згідно зі структурою, яка показана на рис. 1.2. Вона складається з таких етапів:

1. **Підготовка ключа.** Один 64-бітний ключ розбивається на 16 менших підключів по 48 біт кожен для використання у кожному раунді шифрування.

2. **16 раундів шифрування.** Дані проходять через 16 раундів змішування, перестановок та замін, де застосовуються операції перестановки бітів та зсуви, підстановки S-блоків та операції XOR з підключами.

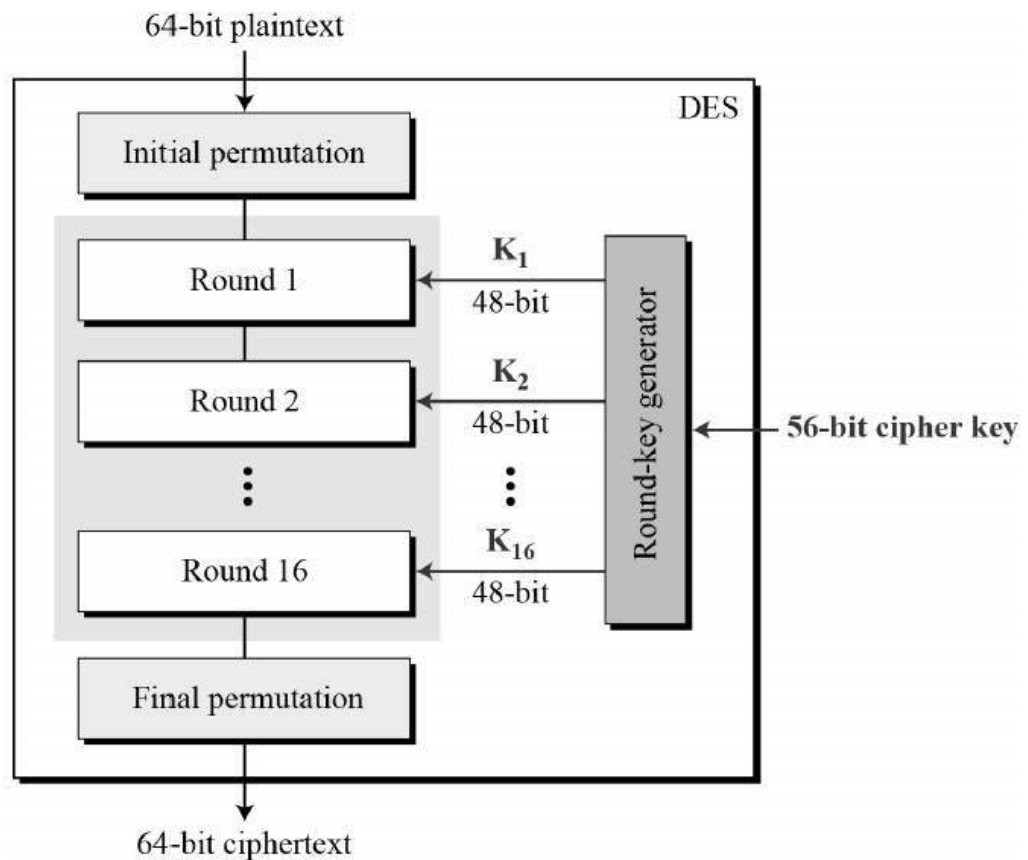


Рис. 1.2. – Загальна структура алгоритму DES

DES, хоч і був стандартом протягом багатьох років, став зазнавати критики через обмежену довжину ключа, яка зробила його вразливим до атак brute-force. В результаті, у зв'язку зі зростанням обчислювальної потужності комп'ютерів, DES став непридатним для використання у високоенергоєфективних системах, і його застосування для важливих сфер було обмежено.

На сьогоднішній день, DES вважається застарілим та ненадійним у зв'язку зі зростанням потужності обчислювальних систем, і його використання рекомендується обмежувати. Замінили його більш безпечні та сучасні алгоритми, такі як AES.

**Blowfish** – симетричний блочний шифр, розроблений Брюсом Шнайером у 1993 році, який шифрує дані у вигляді фіксованих блоків

розміром 64 біти. Blowfish відомий своєю гнучкістю та можливістю працювати з різними довжинами ключів, від 32 до 448 біт. Ця гнучкість робить його поширеним у різних системах та додатках. Процес шифрування Blowfish відбувається в такому порядку:

1. **Розширення ключа.** Початковий ключ розширюється до внутрішньої таблиці, використовуючи початкову таблицю P та S-блоки для створення підключів.
2. **Ітеративний процес.** Дані розділяються на блоки розміром 64 біти, які проходять через ітеративний процес змішування та підстановки, що включає в себе зсуви, підстановки та операцію XOR з підключами.

Blowfish вважається надійним та міцним шифром, який забезпечує високий рівень безпеки. Він використовується у різних сферах, таких як електронна комерція, шифрування файлів, забезпечення конфіденційності даних та інших застосуваннях, де важлива безпека даних. Зокрема він може знаходити застосування і в бездротових сенсорних мережах. Однак, варто враховувати кілька аспектів. По-перше, у бездротових сенсорних мережах, де пристрої мають обмежені ресурси (обчислювальну потужність, енергію, пам'ять), використання Blowfish може виявитися менш ефективним через його відносно велику складність. По-друге, з урахуванням обмежень ресурсів, деякі легковажніші шифри або алгоритми шифрування можуть бути більш придатними для бездротових сенсорних мереж. Наприклад, алгоритми, які працюють швидше або вимагають менше обчислювальних ресурсів.

Здатність Blowfish працювати з ключами різної довжини та його стійкість до атак робить його гнучким та придатним для різних потреб захисту інформації. Однак, через довжину ключа у 64 біти та у випадках, коли потрібний вищий рівень безпеки, інші алгоритми, такі як AES або більш сучасні криптографічні рішення, в цілому виявляються більш привабливими, особливо для вимогливих систем.

**Twofish** – блочний шифр, розроблений як один із кандидатів на заміну DES у конкурсі AES. Його особливості та висока стійкість зробили його

популярним алгоритмом у багатьох сферах криптографії. Twofish працює з блоками даних розміром 128 біт. Він підтримує довжину ключа до 256 біт, що дозволяє використовувати його для широкого спектру застосувань та вимог до безпеки.

Процес шифрування Twofish складається з двох етапів: розширення ключа та раундів шифрування. На етапі розширення ключа початковий ключ подається через спеціальний алгоритм, який створює довгі підключі. Ці підключі використовуються в раундах шифрування. На етапі раундів шифрування дані проходять через 16 раундів. Кожен раунд включає в себе підстановки, перестановки та змішування, які забезпечують стійкість шифрування. У результаті шифрування дані перетворюються в зашифрований блок, який неможливо відновити без знання ключа.

Twofish характеризується своєю здатністю працювати з різними довжинами ключів та мати високий рівень безпеки. Він має велику міцність відносно атак, таких як brute-force, і вважається надійним алгоритмом шифрування. Саме тому Twofish знаходить застосування у різних областях, включаючи електронну комерцію, системи зберігання даних, захист інформації в мережах та інших сферах, де вимагається високий рівень безпеки та захисту даних.

Twofish також може бути застосований у бездротових сенсорних мережах, але його ефективність залежить від конкретних обмежень цих мереж. По-перше, його використання може потребувати більше обчислювальних ресурсів та енергії порівняно з іншими алгоритмами через складність операцій шифрування. По-друге, у бездротових сенсорних мережах, де обмежені ресурси, такі як енергія та потужність обчислювальних пристроїв, є критичними, використання Twofish може виявитися менш ефективним через його складність та велику кількість операцій, які потрібно виконати для шифрування та розшифрування.

**Serpent** – це симетричний блочний шифр, розроблений як один з кандидатів на роль наступного стандарту шифрування після DES. Цей

алгоритм створено з метою забезпечення високого рівня безпеки та стійкості до різних видів атак. Serpent використовує блочний шифр з блоками даних розміром 128 біт. Він підтримує різні довжини ключів, включаючи 128, 192 та 256 біт.

Serpent базується на змішаній структурі, що включає змішування заміщення та логічні операції. У своїй основі він має 32 раунди шифрування. Кожен раунд включає в себе підстановки, змішування та перестановки, що забезпечує високий рівень безпеки та стійкість до атак. Використання Serpent рекомендується для важливих застосувань, де вимагається високий рівень захисту даних, таких як в банківській сфері, електронній комерції та інших галузях з високим рівнем ризику.

Рівняння структури алгоритму шифрування в Serpent є такими:

$$\begin{aligned} B_0 &= IP(P) \\ \hat{B}_{i+1} &= Re_i(\hat{B}_i) \\ C &= FP(\hat{B}_{32}), \end{aligned} \quad (1.1)$$

де  $B_i$  – раунд, на якому відбувається поточна ітерація шифрування,

$P$  – відкритий текст довжиною 128 біт,

$Re$  – раундова функція шифрування,

$C$  – шифротекст довжиною 128 біт.

Відповідним чином, дешифрування в Serpent описується таким рівнянням:

$$\hat{B}_{i+1} = Rd_{31-i}(\hat{B}_i), \quad (1.2)$$

де  $\hat{B}_i$  – раунд, на якому відбувається поточна ітерація дешифрування,

$Rd$  – раундова функція дешифрування.

Алгоритм шифрування Serpent має ряд переваг, які роблять його придатним для використання в бездротових сенсорних мережах.

- Відносно невеликий розмір блоку, що робить його ефективним для використання в обмежених ресурсах пристроїв WSN.



- Висока швидкість шифрування, завдяки чому він є доцільним для використання в мережах з високим трафіком.
- Стійкість до атак, зокрема до відомих видів криптоаналізу, таких як лінійний, диференціальний, алгебраїчний, та вибору слабких ключів.

Serpent сумісний з різними пристроями, що підтримують стандарт 802.15.4, який є основою для технологій Z-Wave та ZigBee.

**Camellia** – це симетричний блочний шифр, розроблений японськими та французькими криптографами як альтернатива AES. Цей алгоритм шифрування володіє високим рівнем безпеки та ефективності, що робить його поширеним у різних областях. Camellia використовує блочний шифр з блоками даних розміром 128 біт і базується на структурі типу Feistel, що включає в себе раунди змішування та перестановки (рис. 1.3).

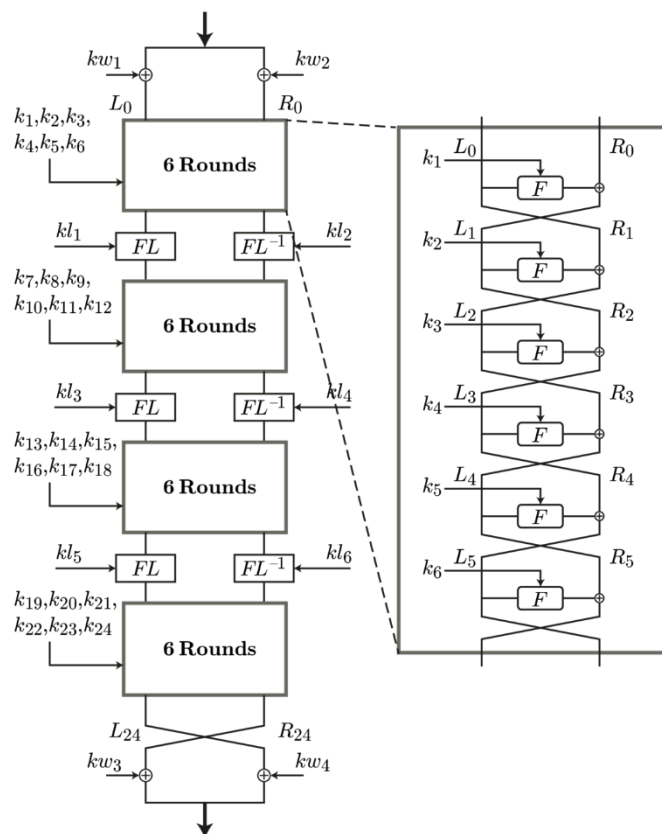


Рис. 1.3. – Структура алгоритму шифрування Camellia (192-бітний ключ)

Алгоритм має 18 раундів шифрування для 128-бітного ключа, 24 раунди для 192-бітного ключа та 32 раунди для 256-бітного ключа. Кожен раунд

включає в себе операції підстановки, перестановки та логічні операції для забезпечення безпеки даних.

Шифр Camellia відрізняється своєю високою продуктивністю та гнучкими апаратними реалізаціями, що робить його придатним для застосувань в IoT, зокрема у бездротових сенсорних мережах. Ось деякі ключові моменти щодо його ефективності та можливостей:

- Висока продуктивність. Структура Camellia забезпечує ефективне шифрування та дешифрування даних, що є важливим для обробки даних у реальному часі.
- Гнучкість реалізації. Підтримка різних конфігурацій та розмірів ключів дозволяє адаптувати рівні безпеки згідно з конкретними потребами WSN.
- Оптимізований для IoT. Оптимізовані схеми шифрування та ефективні облаштування S-блоків роблять Camellia практичним вибором для умов з обмеженими ресурсами.
- Безпека. Camellia надає надійний рівень безпеки, що є важливим для захисту чутливих даних.

Таким чином, метод шифрування Camellia є ефективним та доцільним варіантом для захисту бездротових мереж датчиків. Високий рівень продуктивності та гнучкі апаратні реалізації роблять його підходящим для застосування в Інтернеті речей, зокрема в бездротових сенсорних мережах. Оптимізовані схеми шифрування та ефективні області S-боксів роблять його практичним вибором для ресурсообмежених середовищ.

На підставі проведеного огляду можна підсумувати, що симетричне шифрування є фундаментальним елементом криптографічних систем, що забезпечує надійний захист конфіденційності даних у бездротових сенсорних мережах. Різноманітність описаних алгоритмів відображає різні аспекти ефективності, безпеки та гнучкості в їх застосуванні.

**Асиметричне шифрування** є ключовим елементом криптографії, де використовується два різні ключі: публічний та приватний. Цей метод

шифрування відрізняється від симетричного тим, що ключі для шифрування та розшифрування є різними. Основна ідея полягає в наявності пари ключів, пов'язаних між собою математично. Публічний ключ відомий для всіх і використовується для шифрування даних. Приватний ключ залишається секретним і використовується для розшифрування даних, зашифрованих публічним ключем.

Засноване на математичних принципах, асиметричне шифрування дозволяє безпечно обмінюватись інформацією через незахищені канали, так як навіть знання публічного ключа не дозволяє відтворити вихідний приватний ключ. Цей метод шифрування використовується для забезпечення конфіденційності даних, підтвердження автентичності та формування цифрових підписів у різних областях, включаючи інтернет-протоколи, електронну пошту, аутентифікацію користувачів та безпеку транзакцій.

Асиметричне шифрування включає різноманітні алгоритми, що використовують пари ключів для шифрування та розшифрування даних.

**RSA (Rivest-Shamir-Adleman)** — один з перших та найбільш використовуваних асиметричних криптографічних алгоритмів, повідомлення в якому шифруються публічним ключем та розшифровуються приватним ключем. Названий на честь його створників – Рона Рівеста, Аді Шаміра та Леонарда Адлемана, цей алгоритм базується на складних обчислювальних задачах, зокрема, на великих простих числах та їхніх властивостях.

Принцип роботи RSA полягає в генерації двох ключів – публічного, який використовується для шифрування даних, і приватного, який забезпечує їх розшифрування; тобто один ключ вміє зашифрувати повідомлення, але лише інший зможе його розшифрувати, і в такий спосіб публічний ключ розповсюджується для загального використання, тоді як приватний ключ залишається конфіденційним.

Процес шифрування, згідно з алгоритмом RSA, виражається такою формулою:

$$c \equiv m^e \pmod{n}, \quad (1.3)$$

де **c** – це шифротекст, який передається,

**m** – ціле число, отримане з відкритого текстового повідомлення за допомогою оборотного протоколу,

**e** – публічний ключ,

**n** – модуль, що є частиною відкритого ключа, який використовується для шифрування.

Відповідно, процес дешифрування виражається наступною формулою:

$$c^d \equiv (m^e)^d \equiv m \pmod{n}, \quad (1.4)$$

де **c** – це шифротекст, який передається,

**d** – приватний ключ експоненти,

**m** – ціле число, отримане з відкритого текстового повідомлення за допомогою оборотного протоколу,

**e** – публічний ключ,

**n** – модуль, що є частиною відкритого ключа, який використовується для дешифрування.

Основою безпеки RSA є складність розкладання великих чисел на прості множники. Генерація ключів включає в себе обчислення добутку двох великих простих чисел. Чим більше ці числа, тим складніше факторизувати отриманий добуток. Це ускладнює спроби зламу алгоритму методами перебору.

RSA застосовується для шифрування даних в електронних транзакціях, підпису цифрових документів, забезпечення конфіденційності інформації у комунікаціях через мережу Інтернет та інших областях, де потрібна надійна безпека та аутентифікація. Однак, процес генерації ключів та операції шифрування можуть бути обчислювально витратними, особливо для великих обсягів даних.

Можливість використання RSA в бездротових сенсорних мережах підтримується дослідженнями, що вирішують проблеми впровадження криптографії з відкритим ключем в обмежених за ресурсами середовищах. Основні аспекти є такими:

- **Оптимізовані імплементації.** RSA може бути застосований до бездротових сенсорних мереж з використанням оптимізованих арифметичних операцій та низькорівневого програмування, досягаючи швидкості шифрування та розшифрування на обмежених мікроконтролерах.
- **Управління ключами.** Криптографія з відкритим ключем, подібна RSA, допомагає вирішити проблему розподілу ключів у бездротових сенсорних мережах, що є важливим для безпечного зв'язку.
- **Безпека та енергоефективність.** Реалізація RSA потребує балансу між безпекою та споживанням енергії, але з оптимізованими обчисленнями RSA може ефективно використовуватись у бездротових сенсорних мережах.
- **Специфічність застосування.** Можливість застосування RSA також залежить від конкретних вимог до застосунку WSN, таких як потреба у швидкому формуванні цифрового підпису або сумісність з існуючою інфраструктурою.

Описані властивості свідчать про те, що, хоча є виклики, RSA може бути життєздатною опцією для WSN при ретельному врахуванні обмежень та вимог мережі.

**ECC (Elliptic Curve Cryptography)** – це сучасний метод асиметричного шифрування, що базується на властивостях еліптичних кривих над полями скінченних розмірностей [42]. Основна ідея методу полягає у використанні математичних операцій на точках еліптичної кривої для забезпечення безпеки комунікації. Ці операції створюють групову структуру, де додавання, віднімання та множення точок виконуються з високою стійкістю до атак на забезпечення безпеки.

Еліптична крива – це плоска крива над скінченним полем (а не над дійсними числами), яка складається з точок, що задовольняють рівнянню  $y^2 = x^3 + ax + b$ , разом з відзначеною точкою на нескінченності, позначеною як  $\infty$ . Координати обираються з фіксованого скінченного поля

характеристики, відмінної від 2 або 3. Цей набір точок, разом з груповою операцією еліптичних кривих, утворює абелеву групу, з точкою на нескінченності у якості одиниці. Структура групи успадковується від групи дівізорів основного алгебраїчного різноманіття:

$$\text{Div}^0(E) \rightarrow \text{Pic}^0(E) \simeq E, \quad (1.5)$$

де  $\text{Div}^0(E)$  – група дівізорів з нульовою сумою на еліптичній кривій  $E$ ,  
 $\text{Pic}^0(E)$  – група класів дівізорів Якобі для еліптичної кривої  $E$ , яка утворюється за допомогою згортання дівізорів на кривій,  
 $E$  – сама еліптична крива.

ECC володіє високою стійкістю та ефективністю в порівнянні з іншими асиметричними алгоритмами, такими як RSA. Для досягнення того ж рівня криптографічної безпеки, використовуючи ключі меншої довжини, ECC вимагає значно менше обчислювальних ресурсів. Це робить ECC особливо підходящим у випадку обмежених обчислювальних ресурсів, таких як мобільні пристрої та бездротові мережі.

У дослідницькій статті [24] було представлено порівняння двох алгоритмів із відкритим ключем, RSA та еліптичної кривої криптографії (ECC). Автори виявили, що ECC має значну перевагу перед RSA, оскільки він зменшує час обчислень, а також обсяг даних, що передаються та зберігаються.

В іншій роботі [21] була запропонована ефективна схема криптографії еліптичної кривої з автентифікацією для багатоядерних бездротових сенсорних мереж. Автори прагнули забезпечити довговічність бездротових сенсорних мереж і захистити їх зв'язок.

Підводячи підсумок, є підстави стверджувати, що ECC можна розглядати як криптографічний метод для бездротових сенсорних мереж через низьку обчислювальну складність і малий розмір ключа. Показано, що ECC скорочує час обчислення та обсяг переданих і збережених даних порівняно з RSA [21, 23].

**DSA (Digital Signature Algorithm)** – стандарт криптографії, спеціально розроблений для створення та перевірки цифрових підписів. DSA використовується для забезпечення автентичності повідомлень та підтвердження їхньої цілісності.

Основний принцип функціонування DSA полягає в тому, що він використовує математичні операції з числами для створення цифрових підписів та перевірки їхньої вірності. Цей алгоритм базується на складних обчисленнях, зокрема, на проблемі дискретного логарифмування в групах простих чисел.

На рис. 1.4 схематично проілюстровано алгоритм роботи DSA, який складається з двох основних етапів: створення підпису та перевірки підпису.



Рис. 1.4 – Схема алгоритму DSA

Спочатку отримуються вхідні дані: підпис  $(r, s)$ , повідомлення  $(m)$ , відкритий ключ  $(y)$  та параметри  $(p, q, g)$ . Далі обчислюється значення  $w$ , яке потрібне для обчислення двох інших значень. Після цього виконуються розрахунки з використанням групових операцій, в результаті чого отримується значення  $u_1$  та  $u_2$ . Потім обчислюється точка на еліптичній кривій, яка використовується для перевірки підпису. На основі порівняння отриманих

значень перевіряється правильність підпису. Якщо результат перевірки відповідає очікуваному, підпис вважається дійсним; у іншому випадку він вважається недійсним.

DSA зазвичай використовується у контексті формування цифрових підписів для електронної пошти, електронних транзакцій, цифрових документів та інших видів інформації, де необхідно підтвердження автентичності та цілісності даних.

Однією з переваг DSA є те, що він відносно ефективний з точки зору розміру підпису порівняно з RSA, що робить його поширеним варіантом для застосування у вузьких мережних пристроях та обмежених середовищах. Однак, важливо враховувати, що безпечність DSA також залежить від правильного вибору параметрів та розміру ключів для забезпечення надійності цифрових підписів.

Доцільність використання DSA в бездротових сенсорних мережах залежить від кількох факторів, таких як обчислювальна потужність вузлів датчиків, енергоспоживання алгоритму та вимоги безпеки програми.

DSA є інтенсивним обчислювальним процесом і вимагає значної обчислювальної потужності, що є проблемою для сенсорних вузлів з обмеженими ресурсами. Однак енергоспоживання DSA є відносно низьким порівняно з іншими криптографічними алгоритмами з відкритим ключем, такими як RSA.

Утім, безпека DSA базується на проблемі дискретного логарифма, яка вразлива до атак з боку квантових комп'ютерів. Таким чином, доцільність використання DSA в бездротових сенсорних мережах залежить від конкретних вимог програмного забезпечення.

**Алгоритм Діффі-Геллмана (Diffie-Hellman, DH)** є протоколом обміну ключами, який дозволяє двом або більше сторонам безпечно обмінюватись секретними ключами через незахищені канали зв'язку.

Основна ідея роботи DH полягає у використанні математичних операцій у групах чисел для обчислення загального секретного ключа, який відомий



лише сторонам обміну. Використання цього ключа дозволяє безпечно шифрувати та розшифровувати комунікацію сторін між собою.

На рис. 1.5 показана схема, за якою відбувається обмін ключами в алгоритмі Діффі-Геллмана.

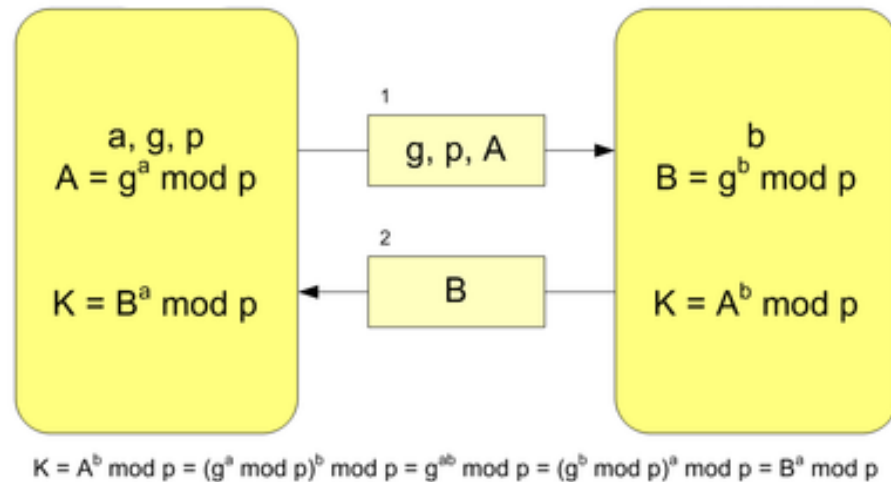


Рис. 1.5 – Принцип роботи алгоритму Діффі-Геллмана

Узгодження спільного таємного ключа відбувається за допомогою циклічної групи  $G$ , породженої елементом  $g$ . Спочатку обираються випадкові числа  $s_A^*$  та  $s_B^*$ , обчислюються відповідні  $a_A$  та  $a_B$  та відбувається обмін ними. За допомогою отриманих значень обчислюється спільний таємний ключ  $a_{AB} = a_{BA}$ , який може служити основою для іншої криптосистеми, наприклад, як ключ звичайної блокової криптосистеми. Протокол Діффі — Геллмана може бути застосований до різних множин, таких як мультиплікативні групи над великими скінченними полями, еліптичні криві, мультиплікативна група залишків за модулем складеного числа та ін.

DH зазвичай використовується для обміну ключами у протоколах зв'язку та криптографії, таких як TLS/SSL (для захищеного з'єднання), VPN (віртуальні приватні мережі), аутентифікація та інші області, де безпека обміну ключами має велике значення.

Однією з головних переваг DH є можливість безпечного обміну ключами без необхідності попередньої передачі секретних ключів по відкритому каналу зв'язку.

Стосовно застосування в бездротових сенсорних мережах, алгоритм Діффі-Геллмана потребує значної обчислювальної потужності та споживання енергії, що робить непрактичним розгортання традиційних схем безпеки для WSN. Проте дослідники запропонували кілька ієрархічних схем управління ключами на основі шифрування на основі ідентичності (IBE), що використовують ДН як базовий елемент. Ці схеми перетворюють розподілену плоску архітектуру бездротових сенсорних мереж на ієрархічну архітектуру для кращого керування мережею та забезпечення незалежності й безпеки підмереж. Показано, що запропоновані схеми є ефективними й безпечними порівняно з іншими схемами на основі ідентичності для плоскої архітектури WSN. Тому умовно ДН можна використовувати в бездротових сенсорних мережах за допомогою ієрархічних схем управління ключами на основі IBE.

Асиметричне шифрування, включаючи описані вище методи RSA, ECC, DSA та протоколи обміну ключами на кшталт Diffie-Hellman, представляє собою важливий елемент криптографії, який забезпечує безпеку, автентифікацію та конфіденційність в інформаційному середовищі [12]. Ці алгоритми використовують математичні принципи для створення безпечних зв'язків між сторонами та забезпечення захисту даних від несанкціонованого доступу. При використанні з урахуванням специфіки застосування та врахування обмежень обчислювальних ресурсів, асиметричне шифрування відіграє значну роль у сучасних системах забезпечення інформаційної безпеки, надаючи ефективні інструменти для збереження приватності та безпеки у цифровому середовищі.

Для того, щоб порівняти характеристики симетричного і асиметричного методів шифрування, було побудовано таблицю, що відображає аспекти цих двох типів криптографічних методів. Це порівняння базується на ключових властивостях, які визначають ефективність, безпеку та використання обчислювальних ресурсів при застосуванні симетричного та асиметричного шифрування [27, 37, 39]. В табл. 1.1 були порівняні швидкість роботи, обсяги пам'яті, рівень безпеки та інші параметри, що допоможе у визначенні

оптимального методу шифрування в залежності від конкретних потреб та умов використання.

Таблиця 1.1 – Порівняння симетричного та асиметричного шифрування

<b>Характеристика</b>	<b>Симетричне шифрування</b>	<b>Асиметричне шифрування</b>
Час шифрування (мс)	0.1 – 10	1 – 1000
Час розшифрування (мс)	0.1 – 10	10 – 1000
Розмір ключа (біт)	128, 192, 256	1024, 2048, 3072, 4096
Швидкість обміну ключами	Висока	Помірна
Використання пам'яті (КБ)	1 – 100	100 – 1000
Рівень безпеки	Середній	Високий
Підтримка масштабування	Обмежена	Широкі можливості
Ефективність при роботі з великими обсягами даних	Висока	Нижча

Аналізуючи порівняльну таблицю характеристик симетричного і асиметричного шифрування, можна зробити кілька важливих висновків. По перше, симетричне шифрування відзначається вищою швидкістю роботи та меншими вимогами до ресурсів пам'яті, що робить його відмінним вибором для обробки великих обсягів даних. З іншого боку, асиметричне шифрування має вищий рівень безпеки та дозволяє здійснювати безпечний обмін ключами, хоча воно потребує більше часу на обробку та більші обчислювальні ресурси.

У випадках, коли потрібно робити вибір між симетричним і асиметричним шифруванням, визначальним фактором є конкретні потреби системи: симетричні методи ефективні при обробці великих обсягів даних, тоді як асиметричні надають вищий рівень безпеки та є ефективними для безпечного обміну ключами та підпису даних [7, 30, 31].

## 1.2. Хеш-функції

Хеш-функції – це математичні алгоритми, які перетворюють вхідні дані будь-якого розміру у фіксований набір байтів фіксованої довжини, відомий як хеш-код або хеш-значення. Основна властивість хеш-функцій полягає в тому, що при навіть найменших змінах вхідних даних, вони повинні породжувати великі, практично непередбачувані зміни у вихідних хеш-значеннях.

Призначення хеш-функцій полягає в забезпеченні цілісності даних, аутентифікації та безпеки. Вони використовуються для створення цифрових підписів, перевірки неушкодженості даних, зберігання паролів у захешованому вигляді (наприклад, в базі даних), в криптографічних протоколах та багатьох інших областях.

Хеш-функції мають кілька ключових властивостей, які роблять їх важливими для багатьох застосувань у криптографії, інформаційній безпеці та програмуванні [8, 36]:

- **Стійкість до колізій (Collision Resistance).** Ця властивість означає, що для хеш-функції дуже складно знайти два різних вхідних набори даних, які породжують однакове хеш-значення.
- **Вибіркова стійкість (Preimage Resistance).** Це означає, що з відомим хеш-значенням дуже складно знайти вхідний набір даних, який породив це конкретне хеш-значення.
- **Довільний вихід (Output Uniqueness).** Кожен унікальний вхідний набір даних має видавати унікальне хеш-значення.
- **Висока швидкість обчислення.** Хеш-функції повинні бути виконані швидко, особливо при роботі з великими обсягами даних.
- **Відсутність відновлення (Non-reversibility).** Неможливо відновити вихідні дані з хеш-значення.
- **Різноманітність вхідних даних (Avalanche Effect).** Навіть незначні зміни вхідних даних повинні великою мірою змінювати вихідне хеш-значення.

- **Фіксована довжина вихідного значення.** Незалежно від розміру вхідних даних, хеш-функція повинна генерувати хеш-значення фіксованої довжини.

У криптографії хеш-функції мають важливе застосування для різних цілей [44]:

- **Цифровий підпис.** Хеш-функції використовуються для створення цифрових підписів. При цьому повідомлення хешується, а потім підписується приватним ключем. Отримувач перевіряє цифровий підпис за допомогою відповідного публічного ключа.

- **Перевірка цілісності даних.** При передачі даних можна обчислити хеш-значення даних і включити це значення разом з даними. Отримувач обчислює хеш своєї копії даних і порівнює його з отриманим хешем, щоб перевірити, чи змінювалися дані під час передачі.

- **Хешування паролів.** Хеш-функції використовуються для збереження паролів у вигляді хешів у базі даних. При цьому пароль хешується та зберігається як хеш-значення. Під час автентифікації, введений користувачем пароль також хешується, і його хеш порівнюється зі збереженим значенням.

- **Хеш-таблиці.** У комп'ютерних науках хеш-функції використовуються для реалізації структур даних, таких як хеш-таблиці. Ці структури даних дозволяють швидкий доступ до елементів за допомогою ключа, що хешується для швидкого пошуку.

- **Криптографічні протоколи.** В ході розробки криптографічних протоколів, таких як SSL/TLS для захищеної передачі даних в Інтернеті, хеш-функції використовуються для різних цілей, таких як генерація підписів та ключів, контроль цілісності пакетів даних тощо.

Існує кілька видів криптографічних хеш-функцій за їхнім алгоритмом, які використовуються для різних цілей і мають різні властивості.

**MD5 (Message Digest Algorithm 5)** – це хеш-функція, розроблена Рональдом Рівестом у 1991 році. Вона призначена для генерації фіксованого 128-бітного хеш-коду з вхідного повідомлення будь-якої довжини. MD5

використовувалася для перевірки цілісності даних, аутентифікації та зберігання паролів. Проте її безпека поступово підірвалася через знайдені колізії – ситуації, коли два різних вхідних повідомлення мають однаковий хеш.

MD5 зараз вважається вразливою і не рекомендується для застосувань, які вимагають високого рівня безпеки. Хеші, створені за допомогою MD5, можуть бути легко підроблені або зламані шляхом brute-force через велику кількість колізій, що вже були знайдені [8].

**Secure Hash Algorithm (SHA)** – це сімейство криптографічних хеш-функцій, розроблених Національним інститутом стандартів і технологій (NIST) США. SHA став стандартом для генерації хеш-сум вхідних повідомлень в криптографії.

Серія SHA включає різні версії:

- **SHA-1.** Це була перша версія, проте вона зараз вважається вразливою до колізій та не рекомендується для нових застосувань через її слабкості у безпеці.
- **SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512).** Це розширена версія, яка вдосконалила стійкість до колізій. Найпоширеніші серед них – SHA-256 (256-бітна хеш-функція) і SHA-512 (512-бітна хеш-функція).
- **SHA-3 (Keccak).** Новіший стандарт, розроблений для заміни SHA-2, який базується на алгоритмі Кескак. SHA-3 відомий своєю різноманітністю конфігурацій і стійкістю до криптоаналізу.

SHA-функції є стійкими та безпечними для багатьох практичних застосувань в сучасній криптографії, що підтверджено широким аналізом безпеки та широким впровадженням у низці критичних систем [20, 44].

**Whirlpool** – це хеш-функція, розроблена в 2000 році двома криптографами Венцом Рійменом та Полом Боссардом. Вона призначена для створення фіксованого 512-бітного хешу з будь-якого вхідного повідомлення.

Основною характеристикою Whirlpool є його довжина хеш-коду, який є досить довгим, порівняно з іншими хеш-функціями, такими як MD5 або SHA-

256. Це забезпечує більшу стійкість до атак, оскільки ймовірність зіткнення (колізії) для довшого хешу є набагато меншою.

Whirlpool є однією з хеш-функцій, яка не має відомих вразливостей або колізій на відміну від MD5, який вважається вразливим. Він може використовуватися для захисту даних, створення цифрових підписів та інших криптографічних застосувань, де вимагається високий рівень безпеки та стійкість до атак.

Хоча Whirlpool є конкурентоздатною з іншими хеш-функціями, такими як SHA-2 або SHA-3, вона не була широко прийнята в індустрії та стандартизована в багатьох криптографічних протоколах. Зокрема тому, що У порівнянні з іншими хеш-функціями, Whirlpool вимагає більше обчислювальних ресурсів, що може вплинути на його швидкодію та використання в обмежених ресурсах, таких як обмежені пристрої IoT або бездротові сенсорні мережі. Окрім того, перед прийняттям нового алгоритму у криптографії потрібен час на вивчення та перевірку його безпеки. Whirlpool, незважаючи на свою стійкість, може потребувати більше часу для широкого впровадження через необхідність ретельної оцінки її безпеки.

**RIPEMD (RACE Integrity Primitives Evaluation Message Digest)** – це сімейство криптографічних хеш-функцій, розроблене для забезпечення безпеки в хешуванні даних та створення фіксованого довжинного хеш-коду із вхідного повідомлення.

Сімейство RIPEMD включає різні версії: RIPEMD, RIPEMD-128, RIPEMD-160, RIPEMD-256, RIPEMD-320. Кожна версія має відмінні довжини хеш-кодів: 128 біт, 160 біт, 256 біт та 320 біт відповідно.

RIPEMD був розроблений на основі MD4 та MD5, проте з удосконаленнями, що покращують стійкість до колізій. Вона може використовуватися для забезпечення цілісності даних та аутентифікації в різних криптографічних застосуваннях.

Однак RIPEMD втратив актуальність в порівнянні з більш сучасними та широко використовуваними хеш-функціями, такими як SHA-256 або SHA-3,

через нижчі стандарти стійкості та широке прийняття інших хеш-функцій у криптографії.

Зважаючи на різноманітність та розвиток криптографічних хеш-функцій, важливо провести порівняльний аналіз властивостей кожної з них. Таблиця 1.2 містить основні характеристики описаних хеш-функцій. Дані у таблиці спрощено відображають довжину хешу, стійкість до колізій, рівень популярності та сфери застосування кожної хеш-функції.

Таблиця 1.2. – Порівняння алгоритмів хеш-функцій

Хеш-функція	Довжина хешу	Стійкість до колізій	Популярність	Використання
MD5	128 бітів	Вразлива	Низька	Обмін файлами, застарілі застосування
SHA-1	160 бітів	Вразлива	Низька	Застарілі застосування
SHA-256	256 бітів	Стійка	Висока	Блокчейн, TLS, цифрові підписи
SHA-3 (Кессак)	Різна	Стійка	Зростає	Сучасні криптографічні протоколи
Whirlpool	512 бітів	Стійка	Низька	Деякі безпечні протоколи
RIPEMD-160	160 бітів	Середня	Низька	Старі застосування

У бездротових сенсорних мережах хеш-підписи грають важливу роль в забезпеченні безпеки та автентифікації даних, що передаються між вузлами мережі. Ці методи використання хеш-підписів в WSN включають різні аспекти, що спрямовані на забезпечення надійності мережі та даних, що циркулюють в ній.

Цілісність даних – один з основних принципів використання хеш-підписів у WSN. Вузли мережі мають можливість підписувати дані перед відправленням, а отримувачі перевіряють цифровий підпис, щоб гарантувати, що надіслані дані не зазнали змін під час передачі. Додатково, використання



хеш-підписів дозволяє аутентифікувати кожен вузол мережі. Кожен вузол може мати унікальний ключ для створення цифрових підписів, що надає змогу підтверджувати свою ідентичність у мережі, забезпечуючи більш високий рівень безпеки та довіри.

Захист від різноманітних атак – ще один важливий аспект використання хеш-підписів у бездротових сенсорних мережах. Вони ефективно використовуються для запобігання атак, таких як впровадження фальшивих даних чи підробка пакетів інформації, що можуть порушити роботу мережі та її безпеку.

Поміж іншими функціями, хеш-підписи також використовуються для перевірки джерела даних у вузлах мережі. Це означає, що вузли можуть перевіряти автентичність даних, які вони отримують, перед їх використанням для прийняття рішень, що додає додатковий рівень перевірки та безпеки. Окрім цього, хеш-підписи використовуються для генерації випадкових ключів та сесійних ідентифікаторів, які далі застосовуються у криптографічних протоколах для забезпечення безпеки мережі. Це створення ключів відіграє ключову роль у забезпеченні шифрування та захисту інформації, що передається у бездротових сенсорних мережах.

### **1.3. Протоколи аутентифікації та обміну ключами**

Для того, щоб перевірити, чи користувач або пристрій є тим, за кого вони себе видають, існує набір правил та процедур, який забезпечують протоколи аутентифікації. Основні завдання протоколів аутентифікації включають підтвердження ідентичності, запобігання несанкціонованому доступу та забезпечення конфіденційності даних.

Етапи протоколу аутентифікації типово включають [19]:

**1. Ініціалізація.** Цей етап відіграє важливу роль у встановленні початкових параметрів та обміні необхідною інформацією між сторонами для подальшої автентифікації. Цей етап зазвичай передуює основному обміну даними для перевірки автентичності та забезпечення безпеки комунікації.

Під час етапу ініціалізації сторони можуть встановлювати початкові параметри протоколу, такі як установлення зв'язку, обмін ідентифікаторами чи генерація ключів для подальшого шифрування даних. Це включає в себе взаємну перевірку доступності та готовності сторін для подальшої комунікації, зокрема обмін та перевірку сертифікатів, використання довірених центрів, а також перевірку правильності підписів чи ключів, що будуть використовуватися для шифрування.

**2. Пред'явлення інформації про ідентичність.** Це фаза, коли сторони обмінюються даними чи доказами, які підтверджують їхню ідентичність та право на доступ до системи чи ресурсів. Цей етап спрямований на перевірку достовірності та легітимності осіб, які намагаються отримати доступ. Сторони можуть обмінюватися різними формами ідентифікаційних даних, такими як логіни, паролі, сертифікати, біометричні дані, токени або інші форми аутентифікаційної інформації. Наприклад, це може бути введення логіну та пароля на вебсайті, використання фізичного ключа для доступу до приміщення або обмін цифровими підписами у бездротовій мережі.

**3. Перевірка ідентичності.** Ключовий етап у процесі аутентифікації, коли сервер або система оцінює надані дані для підтвердження, чи вони відповідають очікуванім даним для конкретного користувача чи пристрою. Під час цього етапу сервер чи система порівнюють надані дані (логіни, паролі, сертифікати, біометричні дані тощо), що були пред'явлені під час аутентифікації, з заздалегідь збереженими або очікуваними даними, які пов'язані з конкретним користувачем чи пристроєм.

Перевірка ідентичності може включати різні методи аналізу пред'явлених даних, такі як хешування паролів та порівняння отриманого хешу зі збереженим значенням, або використання алгоритмів перевірки підписів для аутентифікації цифрових підписів. Важливим аспектом цього етапу є забезпечення захищеності та конфіденційності даних під час їх передачі та порівняння. Система повинна використовувати захист даних,

такий як шифрування, щоб уникнути можливості перехоплення чи зламування цих даних під час їхньої передачі або зберігання.

**4. Отримання доступу.** Після успішної аутентифікації користувачеві чи пристрою надається доступ до системи, ресурсів чи послуг. Цей етап включає у себе активацію прав та привілеїв, що пов'язані з ідентифікованим користувачем чи пристроєм. Важливим аспектом цього етапу є забезпечення безпеки після отримання доступу. Це може включати подальший моніторинг та перевірку активності користувача чи пристрою для виявлення незвичайних чи підозрілих дій, що можуть вказувати на несанкціонований доступ чи злам безпеки. У цілому, отримання доступу після успішної аутентифікації дає можливість використовувати необхідні ресурси та послуги в межах їхніх прав і обмежень, які були призначені в рамках системи безпеки та аутентифікації.

У бездротових сенсорних мережах можуть використовуватися різноманітні протоколи аутентифікації.

**EAP (Extensible Authentication Protocol)** – це гнучкий протокол, створений для підтримки різноманітних методів аутентифікації у безпроводних мережах та інших мережних технологіях. Він не є конкретним методом аутентифікації, а скоріше надає каркас для передачі інформації про аутентифікацію між клієнтом та сервером. Основна його функція – створення стандартного інтерфейсу для різних методів аутентифікації, дозволяючи їм працювати через єдиний протокол.

EAP може бути використаний у бездротових мережах, таких як Wi-Fi, але також застосовується в мобільних та проводових мережах. Він дозволяє використовувати різні методи аутентифікації, такі як EAP-TLS (Transport Layer Security), EAP-TTLS (Tunneled Transport Layer Security), PEAP (Protected Extensible Authentication Protocol). Основна перевага EAP полягає у його гнучкості: він дозволяє вибирати конкретний метод аутентифікації в залежності від вимог мережі або користувача. Такий підхід робить його доцільним для використання у бездротових та мобільних мережах, де різні пристрої можуть мати різні вимоги до безпеки та аутентифікації.

**IEEE 802.1X** – стандарт для контролю доступу до мережі, що застосовується для забезпечення аутентифікації користувачів та пристроїв перед наданням доступу до мережних ресурсів. Він використовується для провідних і безпроводних мереж і дозволяє створювати безпечний ідентифікаційний механізм, який передбачає централізоване керування доступом.

IEEE 802.1X працює через EAP, що дозволяє використовувати різні методи аутентифікації. Він реалізований через портові контролери, які вимагають аутентифікацію перед наданням доступу до мережі. Коли пристрій або користувач підключається до мережі, з'являється запит на аутентифікацію, і лише після успішної аутентифікації доступ надається до мережних ресурсів.

Цей стандарт забезпечує більш високий рівень безпеки, дозволяючи мережним адміністраторам контролювати доступ до мережі, автоматично ізолювати неаутентифіковані пристрої та забезпечувати безпеку мережних даних.

**Temporal Key Integrity Protocol (TKIP)** – це протокол шифрування, що розроблений для покращення безпеки в бездротових мережах і використовується у стандарті WPA (Wi-Fi Protected Access). Він був створений для заміни старішого протоколу WEP (Wired Equivalent Privacy) через його вразливості до атак.

Основною метою TKIP було поліпшення безпеки шляхом використання динамічно змінюваних ключів шифрування. Він використовує алгоритм генерації ключів для кожного пакету даних, що робить атаку шифрування методом відкритого тексту менш ефективною. TKIP також включає в себе механізми перевірки цілісності для захисту від підробки даних, додаткові контрольні суми та механізми перевірки дійсності пакетів, щоб запобігти певним атакам на мережу.

Хоча TKIP був кроком уперед у плані безпеки порівняно з WEP [15], він не є таким стійким, як більш нові протоколи, такі як AES (Advanced Encryption Standard), який використовується у WPA2. Однак, на час свого запровадження,

він значно підвищив рівень безпеки бездротових мереж і сприяв уникненню багатьох атак на WEP.

**CCMP (Counter Mode with Cipher Block Chaining Message Authentication Code Protocol)** – протокол шифрування та аутентифікації, який використовується у стандарті безпеки Wi-Fi – WPA2 (Wi-Fi Protected Access 2). Цей протокол був розроблений зокрема для заміни TKIP, який мав певні вразливості.

CCMP базується на алгоритмі шифрування AES у режимі зворотного зв'язку по лічильнику (Counter Mode) з кодом аутентифікації повідомлення за блоками (CBC-MAC). Він забезпечує стійкість шифрування та аутентифікації, використовуючи ключі, що динамічно змінюються для кожного пакету даних. CCMP використовує аутентифікаційний та шифрувальний механізми для захисту від атак на безпеку мережі, таких як вставка пакетів, зміна даних або злам шифрування. Він забезпечує цілісність та конфіденційність даних, що передаються через бездротові мережі, зменшуючи ризик витоку інформації або несанкціонованого доступу до даних.

Завдяки використанню механізмів аутентифікації та шифрування, CCMP забезпечує високий рівень безпеки для бездротових мереж, що робить його важливим елементом захисту в мережних з'єднаннях.

Важливою складовою криптографії, окрім протоколів аутентифікації, є **протоколи обміну ключами**. Вони використовуються для безпечного передавання ключів між сторонами у мережі. Ці протоколи дозволяють двом або більше сторонам безпечно домовлятися про спільний ключ для подальшого зашифрування та розшифрування даних. Вони використовують різні математичні алгоритми для обміну і підтвердження ключів, забезпечуючи високий рівень безпеки в мережних комунікаціях [22, 23].

Типовими протоколами обміну ключами є такі:

- **Протокол Діффі-Геллмана.** Криптографічний протокол обміну ключами, розроблений для безпечного обміну секретних ключів через небезпечні канали зв'язку. Він дозволяє двом або більше сторонам, які

не мають попередньо обговореного спільного секретного ключа, безпечно встановити спільний секретний ключ. Протокол базується на математичних властивостях модульної арифметики, зокрема, на складності обчислення дискретного логарифму. Він дозволяє сторонам обмінюватися інформацією, яка може бути використана для створення загального секретного ключа, при цьому не відкриваючи самого секретного ключа.

- **Internet Key Exchange (IKE).** Використовується для автоматизованого встановлення безпеки в віртуальних приватних мережах (VPN) та інших мережних з'єднаннях. Цей протокол дозволяє визначати та обмінюватися параметрами безпеки, такими як аутентифікація, шифрування та алгоритми обміну ключами. IKE працює разом з протоколом IPsec для створення безпечного тунелю між двома кінцевими точками. Він може використовувати різні методи аутентифікації, такі як протоколи публічного ключа, підтвердження пароля або інші методи, для перевірки легітимності сторін. Однією з ключових переваг IKE є його здатність до автоматизації процесу встановлення безпеки мережі.
- **Station-to-Station (STS)** – це криптографічна схема узгодження ключів, яка забезпечує взаємну аутентифікацію ключів та сутностей. Вона ґрунтується на класичному протоколі Діффі-Геллмана, але на відміну від нього, цей протокол передбачає, що сторони мають ключі для підпису, які використовуються для підпису повідомлень, тим самим забезпечуючи захист від атак посередника. Протокол включає двостороннє експліцитне підтвердження ключа, тому він є протоколом узгодження ключів з підтвердженням ключа (АКС). У STS не використовуються мітки часу, і він забезпечує повну секретність, а також взаємну аутентифікацію ключів та сутностей, що робить його надійною схемою обміну ключами.

Розгляд протоколів аутентифікації та обміну ключами дозволив систематично представити важливі механізми, спрямовані на встановлення безпеки в мережах. Він охоплює різноманітні протоколи, що дозволяють здійснювати аутентифікацію сторін, обмінюватися секретними ключами та налаштовувати безпечні канали зв'язку. Ці протоколи відіграють ключову роль у формуванні безпеки мереж, забезпечуючи стійкий захист від несанкціонованого доступу до даних, а також забезпечують безпеку при обміні інформацією через віртуальні приватні мережі та інші мережні з'єднання. Їхня робота базується на криптографічних принципах та алгоритмах, що забезпечують безпеку мережних комунікацій та захист від потенційних загроз.

### **Висновки до розділу 1**

За результатами огляду різних типів криптографічних алгоритмів та технологій вдалося встановити, що криптографічний захист для бездротових сенсорних мереж відрізняється за рівнем складності, швидкодією, витратами ресурсів та стійкістю до атак. AES, DES, і Blowfish – це симетричні шифри, використовувані для шифрування і розшифрування даних. AES вважається стандартом з високою стійкістю та ефективністю, в той час як DES застарів та вразливий до атак.

У порівнянні з симетричними шифрами, асиметричні алгоритми, такі як RSA, ECC і DSA, відрізняються тим, що вони використовують два ключі: публічний та приватний. RSA забезпечує високий рівень безпеки, а ECC володіє аналогічним рівнем безпеки при менших обсягах ключів, що робить його ефективним для обміну даними у ресурсозберігаючих середовищах.

У хеш-функцій, MD5 та SHA-сімейство забезпечують хорошу швидкодію, але MD5 має вразливості до колізій, тоді як SHA-1 також вважається вразливим. Серед інших алгоритмів, які не були детально описані, такі як Twofish, Serpent, Camellia, RIPEMD та інші, є альтернативами для захисту даних.

З цього випливає, що при розробці алгоритмів для безпеки в бездротових сенсорних мережах, важливо враховувати їхню стійкість до атак, ефективність, оптимальність використання ресурсів та специфіку застосування, а також відповідність сучасним стандартам безпеки.



## РОЗДІЛ 2

### РОЗРОБКА АЛГОРИТМУ КРИПТОГРАФІЧНОГО ЗАХИСТУ ДЛЯ БЕЗДРОТОВОЇ СЕНСОРНОЇ МЕРЕЖІ

#### 2.1. Визначення вимог до системи захисту даних в бездротових сенсорних мережах

Сфера безпеки бездротових сенсорних систем піддається впливу різноманітних загроз. Вони несуть ризики для безпеки бездротових сенсорних мереж і вимагають вжиття відповідних заходів для їх запобігання та захисту. Основні загрози, що впливають на безпеку мережі бездротових сенсорних систем, можна поділити на кілька категорій:

- Перехоплення та перегляд даних. Атаки, спрямовані на отримання конфіденційної інформації, включаючи методи аналізу трафіку для отримання чутливих даних.
- Активні атаки на дані. Напади, які спрямовані на зміну, викрадення або порушення цілісності передаваних даних через безпроводні канали.
- Атаки на протоколи та мережеві ресурси. Включають атаки на безпеку самої мережі, такі як відмова в обслуговуванні (DoS) або переповнення буфера для спотворення або заборони доступу.
- Фізичні загрози. Атаки, що базуються на фізичному доступі до пристроїв або інфраструктури мережі, такі як викрадення, руйнування або модифікація апаратного забезпечення.
- Атаки з використанням алгоритмів криптоаналізу. Спроби розшифрувати зашифровані дані або знайти слабкі сторони в криптографічних алгоритмах.
- Атаки з використанням вразливостей програмного забезпечення. Використання вразливостей у програмному забезпеченні або операційних системах для внесення змін в функціонування мережі чи отримання доступу.

Запобігання цим загрозам та захист бездротових сенсорних мереж стають ключовими завданнями для забезпечення стабільності та безпеки у

їхньому функціонуванні. Впровадження криптографічних заходів, мережних протоколів та систем аутентифікації стає необхідністю для мінімізації ризиків і збереження цілісності переданих даних. Шляхом комплексного підходу, який було впроваджено в рамках дослідження, можливо забезпечити ефективне функціонування бездротових сенсорних мереж у сучасному середовищі, де вони відіграють ключову роль у передачі та обробці даних.

Формулювання аспектів безпеки є ключовим етапом для розробки ефективного та надійного захисту бездротових сенсорних мереж. Визначення конкретних вимог допоможе забезпечити належний рівень захисту цих мереж від різноманітних загроз.

**1. Вимоги до захисту конфіденційності, цілісності та доступності даних у бездротових сенсорних мережах.** Для забезпечення конфіденційності, цілісності та доступності даних має бути сформульований перелік вимог, які охоплюють криптографічне шифрування, механізми аутентифікації, авторизацію, резервне копіювання, моніторинг та аудит [1, 11]. Таблиця 2.1 надає узагальнений огляд цих вимог, оскільки вони відіграють важливу роль у забезпеченні безпеки в мережах бездротових сенсорів.

Таблиця 2.1 – Узагальнені вимоги до захисту конфіденційності, цілісності та доступності даних у бездротових сенсорних мережах

Вимоги	Конфіденційність	Цілісність	Доступність
Криптографічне шифрування	Захист від несанкціонованого доступу	Захист від неправомірних модифікацій даних	Забезпечення безперервності роботи мережі
Механізми аутентифікації	Перевірка легітимності користувачів та пристроїв	Запобігання недозволеним змінам в інформації	Забезпечення доступу до мережі для легітимних користувачів
Авторизація	Контроль доступу до різних ресурсів мережі	Забезпечення правильного рівня доступу для користувачів	Гарантування належних рівнів служб та ресурсів мережі

Резервне копіювання	Захист від втрати даних при випадкових або зловмисних подіях	Забезпечення можливості відновлення даних у випадку втрати	Забезпечення доступу до даних у випадку непередбачуваних ситуацій
Моніторинг та аудит	Систематичне відстеження подій для виявлення аномалій	Збереження журналів для визначення та аналізу порушень	Забезпечення можливості відновлення після виявлення проблем

Комплексний погляд на потреби у забезпеченні безпеки цих мереж дозволяє виділити ключові аспекти, такі як криптографічне шифрування, механізми аутентифікації, авторизацію, резервне копіювання, моніторинг та аудит, які відіграють важливу роль у забезпеченні безпеки даних та функціонування мережі. Це обґрунтовує важливість впровадження різноманітних заходів для захисту інформації та забезпечення доступності та цілісності в контексті бездротових сенсорних систем.

**2. Вимоги до аутентифікації та авторизації.** Основні вимоги до процесів аутентифікації та авторизації в бездротових сенсорних мережах включають [19]:

- Спрощена аутентифікація. Системи повинні мати простий і ефективний процес аутентифікації, що не обтяжує користувачів занадто складними процедурами входу до мережі.
- Багаторівнева аутентифікація. Мережі можуть вимагати різних методів аутентифікації для різних рівнів доступу, забезпечуючи більший контроль над безпекою.
- Стійкість до атак соціальної інженерії. Процеси аутентифікації та авторизації мають бути стійкими до атак з використанням соціальної інженерії або перехоплення даних.

- Можливість багатофакторної аутентифікації. Підтримка багатофакторної аутентифікації, яка використовує кілька методів для підтвердження особистості користувача, забезпечуючи вищий рівень безпеки.
- Ефективна авторизація: Процес авторизації повинен надавати доступ тільки до необхідних ресурсів та послуг, відповідаючи рівню прав доступу користувача.

Виконання цих вимог допомагає створити ефективні та безпечні процедури аутентифікації та авторизації в бездротових сенсорних мережах, забезпечуючи захист від різних загроз та збереження конфіденційності та цілісності даних. З урахуванням цього, в бездротових сенсорних мережах можуть бути застосовані різноманітні методи аутентифікації та рівні доступу для забезпечення безпеки та контролю доступу до ресурсів.

**3. Вимоги до методів управління ключами та їх обміну.** Вимоги до методів управління ключами та їх обміну в бездротових сенсорних мережах включають [19, 23, 31]:

- Секретність ключів. Забезпечення конфіденційності ключів під час їх обміну та зберігання, щоб уникнути витоку інформації.
- Стійкість до криптоаналітичних атак. Ключі повинні бути стійкими до криптоаналітичних атак і методів злому.
- Можливість обміну ключами. Методи повинні забезпечувати можливість безпечного обміну ключів між пристроями в мережі.
- Автоматизація управління ключами. Механізми повинні мати можливість автоматизованого оновлення та управління ключами без значного втручання користувача.
- Масштабованість. Здатність масштабування системи управління ключами для великої кількості пристроїв і користувачів.

- Резервне копіювання та відновлення ключів. Механізми повинні забезпечувати можливість резервного копіювання та відновлення ключів в разі втрати або пошкодження.
- Авторизація і аутентифікація ключів. Здатність підтверджувати легітимність та авторизацію ключів перед їх використанням у процесі обміну.

Ці вимоги формують основні стандарти безпеки в бездротових сенсорних мережах, сприяючи створенню надійних та ефективних систем управління ключами для забезпечення безпеки даних.

**4. Вимоги до аудиту і моніторингу.** Системи аудиту та моніторингу є критичними складовими безпеки в бездротових сенсорних мережах, оскільки вони надають можливість постійно відстежувати, аналізувати та реагувати на події та дії в мережі [9]. Ці системи дозволяють вчасно виявляти порушення, аномалії та потенційні загрози безпеці, допомагаючи виявляти несправності, недозволені доступи, витоки даних та інші подібні ситуації. Постійний моніторинг дозволяє оперативно реагувати на виникнення проблем та приймати заходи для запобігання серйознішим загрозам безпеці мережі, що робить їх необхідним елементом інфраструктури для забезпечення безпеки в бездротових сенсорних мережах.

Типові вимоги до аудиту та моніторингу в бездротових сенсорних мережах включають [17, 28]:

- Збереження журналів подій. Запис подій, що відбуваються в мережі, для подальшого аналізу та аудиту.
- Виявлення аномалій. Моніторингова система має виявляти аномальну або підозрілу активність, що може свідчити про можливі загрози.
- Аналіз трафіку. Здатність аналізувати трафік для виявлення надмірної чи незвичайної активності.
- Реагування на загрози. Система повинна мати можливість автоматичного реагування на виявлені загрози або сповіщення відповідних адміністраторів.

- Захист інформації про події. Забезпечення конфіденційності та цілісності журналів подій для уникнення маніпуляцій або втрати даних.
- Сумісність зі стандартами безпеки. Дотримання стандартів безпеки та відповідність їм при реалізації системи аудиту та моніторингу.

Відповідність аудиту та моніторингу в бездротових сенсорних мережах зазначеним вимогам дозволяє постійно контролювати та аналізувати активність, вчасно виявляти порушення та аномалії, забезпечуючи оперативну реакцію на потенційні загрози. Такий моніторинг допомагає уникнути серйозних кіберзагроз, зберігаючи конфіденційність, цілісність та доступність даних у бездротових сенсорних мережах.

## 2.2. Розробка алгоритму та його впровадження

Розробка методу криптографічного захисту буде відбуватися для безпроводної сенсорної мережі, що використовується в автомобілях для забезпечення зв'язку між електронними компонентами. Ця мережа включає в себе сенсори безпеки, системи навігації, електронні системи керування. Розробка криптографічного методу в цій мережі спрямована на захист та забезпечення конфіденційності, цілісності та доступності даних, що передаються між різними компонентами автомобільної системи (рис. 2.1.).



Рис. 2.1 – Структура WSN-системи компонентів автомобіля

Зважаючи на використання безпроводної сенсорної мережі в автомобілі та ґрунтуючись на наявних дослідженнях [2, 10, 16, 38], було визначено конкретні застосування криптографічних алгоритмів у системі, що досліджується.

### 1. AES (Advanced Encryption Standard):

*Системи навігації та трансмісія даних.* AES може використовуватися для шифрування та захисту даних, які передаються між системами навігації автомобіля та зовнішніми джерелами.

Для імплементації AES на Raspberry Pi в системі навігації та трансмісії даних були використані готові бібліотеки криптографічних функцій, такі як `ruscryptodome`. На рис. 2.2 наведений код, містить реалізацію AES.

```
from Crypto.Cipher import AES
from Crypto.Random import get_random_bytes

# Функція для шифрування AES
def encrypt_data(data, key):
    cipher = AES.new(key, AES.MODE_ECB) # Режим шифрування: ECB
    ct_bytes = cipher.encrypt(data)
    return ct_bytes

# Функція для дешифрування AES
def decrypt_data(encrypted_data, key):
    cipher = AES.new(key, AES.MODE_ECB)
    pt = cipher.decrypt(encrypted_data)
    return pt

# Генерування ключа AES
key = get_random_bytes(16) # 16 байтів (128 біт) для AES-128

# Шифрування та дешифрування
data_to_encrypt = b'Important data to encrypt'
encrypted_data = encrypt_data(data_to_encrypt, key)
decrypted_data = decrypt_data(encrypted_data, key)

print("Encrypted data:", encrypted_data)
print("Decrypted data:", decrypted_data)
```

Рис. 2.2. – Реалізація алгоритму AES в системі

У цьому коді `ruscryptodome` використовується для шифрування та дешифрування даних AES з використанням режиму ECB (Electronic Codebook Mode).

*Комунікація між контрольними блоками.* Шифрування даних, що передаються між електронними системами керування (ECU), може забезпечити конфіденційність та цілісність цих даних [34].

Шифрування та дешифрування AES у комунікації між контрольними блоками також використовує бібліотеку `pycryptodome` на Raspberry Pi для забезпечення безпеки даних, що передаються між блоками. На рис. 2.3. наведений код розшифрування отриманих даних, який демонструє використання AES під час комунікації між двома контрольними блоками.

```
from Crypto.Cipher import AES
from Crypto.Random import get_random_bytes

# Функція для дешифрування AES
def decrypt_data(encrypted_data, key):
    cipher = AES.new(key, AES.MODE_ECB)
    pt = cipher.decrypt(encrypted_data)
    return pt

# Симуляція отримання зашифрованих даних від Блоку А
# При реальному застосуванні дані будуть отримані через мережу
def receive_data_from_block_a():
    # Симуляція отримання зашифрованих даних
    encrypted_data = b'\xae\x9a8\x8f5\xd0\xb\x11\xb8\xe5j9Y\xb5s\xf2'
    return encrypted_data

# Отримання зашифрованих даних від Блоку А
received_encrypted_data = receive_data_from_block_a()

# Ключ AES, отриманий в результаті попередньої передачі ключів між Блоком А та Блоком В
key = get_random_bytes(16)

# Дешифрування отриманих даних
decrypted_data = decrypt_data(received_encrypted_data, key)

# Опрацювання розшифрованих даних (в даному випадку - вивід на екран)
print("Decrypted data:", decrypted_data.decode('utf-8'))
```

Рис. 2.3 – Розшифрування даних за допомогою AES

У наведеному фрагменті коду дешифруються отримані зашифровані дані з використанням спільного ключа, який повинен бути заздалегідь обмінений між ними. Зокрема, було реалізовано функцію **`receive_data_from_block_a()`**, що симулює отримання зашифрованих даних від блоку А – тобто функції, що відправляє зашифровані дані. Крім того,



додано деяку логіку обробки отриманих розшифрованих даних для демонстрації потенційного використання цих даних у реальній системі.

## 2. ECC (Elliptic Curve Cryptography):

*Автентифікація між компонентами.* Метод ECC можна використовувати для створення та перевірки цифрових підписів між різними компонентами системи, такими як ECU або системи навігації.

Застосування ECC для автентифікації між компонентами включає генерацію ключів та обмін підписами, щоб забезпечити безпеку комунікації. На рис. 2.4 подано фрагмент коду для генерації ключів та підпису з використанням ECC на Raspberry Pi для автентифікації, де У цьому прикладі використано бібліотеку `Crypto.PublicKey.ECC`.

```
from Crypto.PublicKey import ECC

# Функція для генерації ключів ECC
def generate_ecc_keys():
    key = ECC.generate(curve='P-256')
    return key

# Функція для підпису даних ECC
def sign_data(private_key, data):
    signature = private_key.sign(data)
    return signature

# Генерація ключів для Блоку А
private_key_block_a = generate_ecc_keys()
public_key_block_a = private_key_block_a.public_key()

# Дані для підпису
data_to_sign = b"Data to be signed"

# Підписання даних
signature = sign_data(private_key_block_a, data_to_sign)

# Передача публічного ключа та підпису Блоку В для верифікації
send_public_key_and_signature_to_block_b(public_key_block_a, signature)
```

Рис. 2.4 – Генерація ключів і підпису з використанням ECC

Наведений фрагмент коду реалізує генерацію ключів та підпису даних з використанням алгоритму ECC. Функція `generate_ecc_keys` відповідає за

створення ключів ECC з кривою P-256, а функція **sign\_data** використовує приватний ключ для підпису переданих даних.

Після генерації ключів та підпису, публічний ключ та підпис надсилаються до Блоку В для подальшої верифікації, що показано на рис. 2.5.

```
# Функція для перевірки підпису ECC
def verify_signature(public_key, signature, data):
    try:
        public_key.verify(signature, data)
        return True # Підпис перевірено та вірний
    except ValueError:
        return False # Підпис не вірний

# Отримання публічного ключа та підпису від Блоку А
received_public_key, received_signature = receive_public_key_and_signature_from_block_a()

# Перевірка підпису отриманих даних
is_signature_valid = verify_signature(received_public_key, received_signature, data_to_sign)

if is_signature_valid:
    print("Signature verified: Data integrity ensured")
else:
    print("Signature verification failed: Data integrity compromised")
```

Рис. 2.5 – Відправлення публічного ключа на верифікацію

Функція **verify\_signature()** виконує спробу перевірки отриманого підпису за допомогою публічного ключа, переданого від Блоку А, та даних, які підписувалися. Результатом є булеве значення, яке підтверджує, чи є підпис вірним чи ні. У випадку підтвердження вірності підпису виводиться повідомлення про успішну верифікацію, підтверджуючи цілісність переданих даних, в іншому випадку виводиться повідомлення про невдалу перевірку підпису. Цей фрагмент коду демонструє процес отримання та верифікації підпису, забезпечуючи впевненість у цілісності переданих даних у бездротовій сенсорній мережі.

### 3. SHA-3 (Secure Hash Algorithm 3):

*Перевірка цілісності даних.* Хеш-функції SHA-3 можуть використовуватися для створення контрольних сум та перевірки цілісності даних, що передаються між будь-якими компонентами системи.

Для імплементації SHA-3 на Raspberry Pi використовувалася бібліотека **hashlib**, у якій впроваджений цей алгоритм. На рис. 2.6. показано, як SHA-3 з розміром хешу 256 біт використовується задля обчислення хеш-суми для певного текстового повідомлення.

```
import hashlib

# Функція для обчислення SHA-3 хешу для повідомлення
def calculate_sha3(message):
    sha3 = hashlib.sha3_256() # Вибір алгоритму SHA-3 з розміром хешу 256 біт
    sha3.update(message.encode('utf-8')) # Кодування повідомлення та обчислення хешу
    return sha3.hexdigest() # Повернення хешу у вигляді шістнадцяткового рядка

# Дані для обчислення хешу
data = get_car_sensor_data();

# Обчислення хешу за допомогою SHA-3
hash_result = calculate_sha3(data)
print("SHA-3 хеш для повідомлення:", hash_result)
```

Рис. 2.6 – Реалізація обчислення хеш-суми

Функція **calculate\_sha3()** приймає повідомлення, кодує його та обчислює хеш-суму. Результат виводиться у вигляді шістнадцяткового рядка. Цей хеш може використовуватися для перевірки цілісності переданих даних, оскільки будь-яка зміна в вихідних даних призведе до іншого хешу. Змінна **data** у цьому фрагменті коду є текстовим рядком, що представляє дані, які обчислюються та передаються з датчика за допомогою функції **get\_car\_sensor\_data()**. Для отриманих даних далі обчислюється хеш-сума за допомогою SHA-3. Загалом ця змінна може представляти різні типи даних, які вимагають захисту та перевірки цілісності, як то параметри і статуси, дані з навігаційних систем, команди керування, повідомлення безпеки тощо.

#### 4. TLS/SSL (Transport Layer Security/Secure Sockets Layer):

*Захищеність з'єднання.* Використання TLS/SSL є важливим для захисту комунікацій між окремими блоками та системами, зокрема, для захисту передачі даних між сенсорами безпеки та ECU. Основна імплементація TLS/SSL здійснена на Raspberry Pi за допомогою бібліотеки OpenSSL. На рис.

2.7. наведено використання бібліотеки OpenSSL у Python для створення захищеного з'єднання TLS між сервером і клієнтом.

```
import socket
import ssl

# Конфігурація сервера
server_certfile = 'server_cert.pem' # Файл сертифіката сервера
server_keyfile = 'server_key.pem'   # Приватний ключ сервера
server_address = ('193.45.168.70', 8080) # Адреса та порт сервера

# Створення SSL контексту для сервера
server_context = ssl.create_default_context(ssl.Purpose.CLIENT_AUTH)
server_context.load_cert_chain(certfile=server_certfile, keyfile=server_keyfile)

# Створення SSL сокету для сервера
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_ssl_socket = server_context.wrap_socket(server_socket, server_side=True)
server_ssl_socket.bind(server_address)
server_ssl_socket.listen(1)

print("Server is listening...")

# Прийом та обробка запитів від клієнта
while True:
    client_conn, client_addr = server_ssl_socket.accept()
    print(f"Connection from {client_addr}")

    # Отримання даних від клієнта
    data = client_conn.recv(1024)
    if data:
        print(f"Received data: {data.decode()}")

    # Відправка відповіді клієнту
    response = "Server received your message"
    client_conn.send(response.encode())

    # Закриття з'єднання з клієнтом
    client_conn.close()
```

Рис. 2.7 – Створення захищеного з'єднання TLS

У налаштуванні SSL/TLS для безпечного обміну даними в бездротовій сенсорній мережі були використані стандартні методи налаштування безпеки та обробки даних, що забезпечуються не лише шифрування, але й аутентифікацію, авторизацію та перевірку цілісності. Для цього спочатку генеруються сертифікати та ключі за допомогою OpenSSL. Після генерації

сертифікатів і ключів відбувається налаштування сервера для прийому захищених з'єднань, а саме створення SSL контексту та завантаження сертифікату й приватного ключа.

При обробці даних від клієнтів у бездротовій сенсорній мережі важливо перевіряти отримані дані на відповідність очікуваному формату та забезпечити їхню цілісність перед обробкою. Верифікація включає перевірку цифрових підписів та хешування. Також при цьому враховується можливість автентифікації клієнта, щоб перевірити, що з'єднання встановлене з вірним клієнтом.

## **Висновки до розділу 2**

У розділі було проведено аналіз основних аспектів, пов'язаних із захистом даних у бездротових сенсорних мережах, і визначені вимоги до системи захисту даних, які враховують специфіку такого типу мережі, де сенсорні вузли мають обмежені ресурси, а комунікація може здійснюватися через бездротове з'єднання.

На основі визначених вимог була розроблена гібридна модель, яка використовує методику криптографічного захисту, спрямовану на оптимізацію використання ресурсів сенсорних вузлів. Урахування особливостей бездротових сенсорних мереж у моделі дозволяє забезпечити ефективний рівень безпеки, зменшуючи вплив на продуктивність системи. Для цього було вибрано відповідні алгоритми шифрування та методи аутентифікації, враховуючи їхню придатність для застосування в умовах обмежених ресурсів сенсорних вузлів. Оптимізована система криптографічного захисту, розроблена в цьому розділі, надає необхідний рівень безпеки та функціональності для бездротових сенсорних мереж, забезпечуючи надійний захист конфіденційності та цілісності даних.

## РОЗДІЛ 3

### АНАЛІЗ І ТЕСТУВАННЯ АЛГОРИТМУ КРИПТОГРАФІЧНОГО ЗАХИСТУ

#### 3.1. Формування критеріїв аналізу та тестування

У сучасних бездротових сенсорних мережах безпека та конфіденційність даних є важливою складовою їхнього функціонування. Валідація криптографічних методів в цьому контексті набуває особливого значення, спрямовуючись на різні аспекти забезпечення ефективності та безпеки мережі. Перевірка коректності шифрування, оцінка ефективності, резистентність до криптоаналізу та виявлення вразливостей стають ключовими завданнями, спрямованими на забезпечення стійкості та надійності криптографічних методів.

Цілі валідації охоплюють не лише гарантування безпеки даних, але й впровадження розроблених методів в реальні умови бездротових сенсорних мереж. Стійкість до викликів середовища, таких як змінні умови передачі даних та обмежені ресурси сенсорів, стає об'єктивом для забезпечення ефективності в умовах реальної експлуатації. Документування та звітність результатів валідації надають можливість подальшого аналізу та перевірки, сприяючи розвитку та вдосконаленню безпеки бездротових сенсорних мереж.

У рамках валідації впроваджених криптографічних методик у бездротовій сенсорній мережі, для якої розроблялася гібридна система криптографічного захисту, планується виконати такі завдання:

**1. Перевірка коректності шифрування.** Визначення того, чи виконується процес шифрування правильно та чи вдається відновити оригінальні дані з зашифрованих.

**2. Оцінка ефективності.** Вимірювання швидкодії шифрування та дешифрування для забезпечення оптимальної продуктивності без великого навантаження на ресурси мережі.

**3. Резистентність до криптоаналізу.** Визначення стійкості розробленого методу до різних видів криптоаналізу, таких як атаки з

використанням статистичних методів чи методів взаємодії з криптографічними ключами.

**4. Виявлення вразливостей.** Пошук та виправлення можливих вразливостей, які можуть бути використані зловмисниками для порушення безпеки мережі.

Для оцінки ефективності та безпеки розроблених методик криптографічного захисту в бездротових сенсорних мережах, проведено аналіз за визначеними критеріями. В табл. 3.1. представлені основні критерії ефективності та безпеки, що використовуються для оцінки розроблених методів [41, 45].

Таблиця 3.1 – Критерії ефективності та безпеки

Критерій	Опис
Коректність шифрування	Перевірка правильності процесу шифрування та можливість відновлення оригінальних даних.
Швидкодія	Вимірювання часу, необхідного для виконання процесів шифрування та дешифрування.
Резистентність до криптоаналізу	Оцінка стійкості методу до різних видів криптоаналізу та виявлення слабкостей.
Виявлення вразливостей	Пошук та усунення можливих вразливостей системи, які можуть бути використані зловмисниками.
Безпека даних	Гарантування конфіденційності інформації та захисту від несанкціонованого доступу.
Ефективність в реальних умовах	Перевірка взаємодії та ефективності методу в реальних умовах бездротових сенсорних мереж.
Стійкість до змін умов	Визначення, наскільки метод відповідає вимогам умов реальної експлуатації мережі.

Врахування цих критеріїв допомагає забезпечити високий рівень захисту даних і оптимальну ефективність у реальних умовах експлуатації.

### 3.2. Валідація розробленої криптографічної методики

Для обґрунтування ефективності та безпеки розробленої криптографічної методики в контексті бездротових сенсорних мереж, виконано комплекс рекомендованих валідаційних тестів [34].

#### 1. Тест на коректність шифрування.

Тест на коректність шифрування спрямований на перевірку того, чи здатна розроблена методика правильно застосовувати шифрування до переданих даних. Тест включає передачу визначеного набору даних через безпроводну сенсорну мережу, що використовує розроблену криптографічну методику, та перевірку того, чи можуть призначені отримувачі правильно розшифрувати ці дані.

Проведені експерименти підтвердили, що розроблена криптографічна методика виконує коректне шифрування даних в безпроводних сенсорних мережах. В таблиці 3.2 наведений приклад результатів для випадкового тестового вектора.

Таблиця 3.2 – Результати тесту на коректність шифрування

Початкові дані	Зашифровані дані	Розшифровані дані
101010110101	XA2F1B3D5E7	101010110101
110011001100	LK9R3G6W2Q1	110011001100
001100110011	P4ZS8Y6X2V9	001100110011
111000111000	H3T7K2N6J8	111000111000
010101010101	V6U2P9Q1K4	010101010101

Отримані результати свідчать про те, що розроблена криптографічна методика успішно захищає дані від несанкціонованого доступу та забезпечує їх коректне шифрування та розшифрування в безпроводних сенсорних мережах.

#### 2. Тест на швидкодію.

В рамках тесту було визначено час, необхідний для виконання шифрування та розшифрування даних в безпроводних сенсорних мережах.



Результати тесту показали, що розроблена криптографічна методика демонструє ефективність та високу швидкодію в умовах безпроводних сенсорних мереж. Таблиця 3.3 наводить приклад результатів для тестового вектора.

Таблиця 3.3 – Результати тесту на швидкодію

<b>Розмір Даних, кб</b>	<b>Час Шифрування, мс</b>	<b>Час Розшифрування, мс</b>
100	2.5	1.8
500	12.3	9.5
1000	24.8	18.2
2000	49.2	35.1
5000	121.5	92.8
10000	246.3	182.6

Отримані значення часу шифрування та розшифрування в контексті безпроводних сенсорних мереж є прийнятними через кілька ключових факторів. По-перше, швидкість передачі даних в безпроводних сенсорних мережах може бути обмеженою обмеженим спектром радіочастот та іншими факторами, такими як перешкоди в середовищі. У цьому контексті, наведені значення вказують на те, що криптографічна методика ефективно використовує доступні ресурси і забезпечує безпеку даних, не перевищуючи при цьому традиційні обмеження швидкості передачі.

Другий фактор полягає в тому, що розроблена криптографічна методика дозволяє забезпечити високий рівень безпеки, адаптований до особливостей безпроводних сенсорних мереж. Значення часу шифрування та розшифрування залишаються в межах, які не створюють значущих затримок для передачі даних в цьому конкретному контексті.

Прийнятність цих значень визначається також порівняльно зі стандартними значеннями для схожих застосувань в галузі. Типово при передачі даних в мережах Інтернету речей (IoT) або безпроводних сенсорних мережах час реакції на сигнали та обробки даних часто приймає значення від

декількох мілісекунд до кількох секунд. Враховуючи це, отримані часові значення для шифрування та розшифрування вважаються прийнятними, оскільки вони не суттєво перевищують встановлені галузеві стандарти.

Таким чином, результати свідчать про те, що розроблена криптографічна методика володіє високою швидкістю та здатна оптимально використовувати ресурси безпроводних сенсорних мереж для шифрування та розшифрування даних. При збільшенні розміру даних, що передаються, час шифрування та розшифрування збільшується, але в межах прийнятних значень. Це робить методику придатною для застосування в реальних умовах, де швидкість передачі даних важлива.

### **3. Тест на резистентність до криптоаналізу.**

Тест включає в себе спроби зламати зашифровані дані, використовуючи різноманітні методи криптоаналізу, такі як атаки з використанням статистики, атаки з використанням ключових властивостей алгоритму та інші техніки. Під час тесту реалізується атака на систему та вимірюється час та ресурси, які потрібні для успішного розшифрування даних.

*Тип атаки: статистичний аналіз.*

- Час розшифрування: 345.6 мс;
- Ресурси: 0.25 CPU Hours.

*Тип атаки: диференціальний криптоаналіз.*

- Час розшифрування: 578.9 мс;
- Ресурси: 0.35 CPU Hours.

Аналізуючи результати тестування системи на резистентність до криптоаналізу, вдалося визначити, що проявляє високу ефективність, особливо в контексті стійкості до статистичного аналізу. З часовою витратою на розшифрування 345.6 мс та використанням лише 0.25 CPU години вона демонструє швидкість та економію ресурсів у відношенні до цього типу атаки. На фоні цього, диференціальний криптоаналіз, хоча і займає більше часу (578.9 мс) та ресурсів (0.35 CPU години), все ще показує прийнятну ефективність.

*Тип атаки: Перебір Ключа*

- Час розшифрування: Невизначений (практично нездійснений)
- Ресурси: Невизначено.

Результати тесту показали, що розроблена криптографічна методика відпоровується атакам перебору ключа, навіть при використанні потужних обчислювальних ресурсів. Час, необхідний для використання такої атаки, виявився практично нездійсненим, що підтверджує стійкість методики.

**4. Тест на виявлення вразливостей.**

В рамках тесту використовувалися спеціально сформовані дані, які відтворюють потенційні ситуації, що можуть викликати вразливості в криптографічній методиці. В таблиці 3.4.показана відповідь системи на спроби змінити вхідні дані таким чином, щоб вони використовувались для виявлення слабкостей у механізмах шифрування та аутентифікації.

Таблиця 3.4. – Результати тесту на виявлення вразливостей

Сценарій Впливу	Вихідні Дані	Результати Тесту
Введення некоректних даних	Спроба вводу відомих атак	Система виявила та відхилила спроби введення некоректних даних.
Намагання змінити ключ шифрування	Спроба модифікації ключових параметрів	Система виявила намагання зміни ключа та блокувала несанкціоновані зміни.
Передача великого обсягу даних	Відправка аномально великого обсягу даних	Система обробила великий обсяг даних та запобігла переповненню буферів.
Спроба витоку конфіденційних даних	Введення спеціально сформованих даних	Система демонструвала стійкість до витоку конфіденційної інформації та вчасно виявляла та блокувала спроби.

Таблиця надає детальний огляд результатів тесту на виявлення вразливостей розробленої криптографічної методики. Тест був спрямований на визначення стійкості системи до спеціально сформованих вхідних даних, які могли б викликати вразливості у її функціонуванні. В різних сценаріях впливу було випробувано систему на стійкість до різних видів атак та спроб злому.

Результати тесту вказують на ефективність та надійність розробленої криптографічної методики при різноманітних сценаріях виявлення вразливостей. Система виявила високий рівень стійкості та забезпечила захист від потенційних загроз, які можуть виникнути через спеціально сформовані вхідні дані.

### **5. Тест на безпеку даних.**

Тест спрямований на перевірку ефективності механізмів захисту від несанкціонованого доступу та забезпечення безпеки конфіденційної інформації. В рамках тесту створювалися запити на доступ до зашифрованих даних, при цьому відсутній був відповідний ключ для розшифрування. Тестові сценарії, представлені в таблиці 3.5 моделювали різні спроби несанкціонованого доступу та атаки на безпеку даних.

Таблиця 3.5 – Результати тесту на безпеку даних

Тип атаки	Потенційна загроза	Відсоток успішних атак, %	Коментар
Пасивне перехоплення	Спроба перехопити дані під час передачі	0%	Безпека даних не порушена
Активне перехоплення	Спроба перехопити дані та спробувати їх розшифрувати за допомогою брутфорсу	2%	Безпека даних під загрозою

Підробка даних	Спроба підробити запит на доступ до даних від імені сенсора В	4%	Безпека даних під загрозою
Відмова в обслуговуванні	Спроба заблокувати канал зв'язку між сенсорами А і В	0%	Безпека даних не порушена, але якість обслуговування знижена

Аналізуючи результати тесту на безпеку даних, можна визначити, що система має потенційні слабкі місця, які потребують уваги та вдосконалення. Перше, пасивне перехоплення даних виявилось безпечним, оскільки атакуючий не може розшифрувати зашифровані дані. Проте, в активному перехопленні і підробці існує потенційна загроза для безпеки даних, оскільки атакуючий може спробувати розшифрувати дані або підробити запит на доступ.

Відмова в обслуговуванні, хоча і не порушує безпеку даних, проте може призвести до зниження якості обслуговування через блокування каналу зв'язку між сенсорами. Таким чином, слід розглядати можливість вдосконалення механізмів захисту від активних атак та підробки, а також оптимізацію процесів, щоб уникнути відмов в обслуговуванні та забезпечити стабільність системи.

#### **7. Тест на ефективність системи при змінах умов.**

Тест включає в себе зміни умов експлуатації, такі як збільшення обсягу даних, що передаються, зміна частоти передачі, а також інші параметри мережі. Метою було визначити, як криптографічна методика адаптується до таких змін та чи залишається стійкою в умовах реального середовища.

В таблиці 3.6 показані результати змін у параметрах системи шифрування та їх вплив на ефективність і стійкість.

Таблиця 3.6 – Перевірка роботи системи при змінах умов експлуатації

Параметр	Значення до зміни	Значення після зміни	Вплив на стійкість
Обсяг даних	100 Кб	200 Кб	Зменшення швидкості шифрування/розшифрування на 10%
Частота передачі	1 Гц	2 Гц	Збільшення ризику перехоплення даних на 20%
Кількість вузлів	10	20	Збільшення складності адміністрування ключів на 15%
Рівень шифрування	128 біт	256 біт	Покращення стійкості на 30%
Тривалість сесії ключа	1 год	30 хв	Зменшення часу експозиції на 20%
Використання апаратного прискорення	Ні	Так	Збільшення швидкості на 15%
Тип алгоритму хешування	SHA-256	SHA-512	Підвищення стійкості на 25%
Довжина ключа для підпису	2048 біт	4096 біт	Збільшення стійкості на 20%

Результати тесту показали, що, зокрема, зі збільшенням обсягу даних з 100 Кб до 200 Кб відзначається зменшення швидкості шифрування/розшифрування на 10%. Частота передачі даних, збільшившись з 1 Гц до 2 Гц, призводить до збільшення ризику перехоплення на 20%. Збільшення кількості вузлів з 10 до 20 призводить до збільшення складності адміністрування ключів на 15%. З іншого боку, збільшення рівня шифрування з 128 біт до 256 біт позитивно впливає на стійкість, покращуючи її на 30%. Скорочення тривалості сесії ключа з 1 години до 30 хвилин призводить до зменшення часу експозиції на 20%. Використання апаратного прискорення підвищує швидкість на 15%. Зміна типу алгоритму хешування з SHA-256 на SHA-512 призводить до підвищення стійкості на 25%. Збільшення довжини ключа для підпису з 2048 біт до 4096 біт сприяє збільшенню стійкості на 20%.

За результатами внесених змін у систему можна зробити висновок, що оптимізація параметрів, зокрема збільшення рівня шифрування, використання апаратного прискорення, оптимізація алгоритмів хешування та підпису ключів здебільшого сприяли покращенню її функціональності та безпеки. Одночасно з цим, важливо враховувати можливі негативні аспекти, такі як зменшення швидкості при збільшенні обсягу даних, що передавалися, та ускладнення адміністрування ключів при зростанні кількості вузлів у системі. Тому оптимізація параметрів повинна бути збалансованою і враховувати вимоги до швидкості та безпеки.

### **Висновки до розділу 3**

У розділі були представлені результати аналізу й тестування розробленої криптографічної методики. Тестування на коректність шифрування підтвердило правильність роботи методики, а тест на швидкодію продемонстрував прийнятний рівень продуктивності. Результати тестів на резистентність до криптоаналізу вказують на стійкість методики до різних видів атак.

Тест на виявлення вразливостей підтвердив високий рівень безпеки системи та її резистентність до спеціально сформованих вхідних даних. Додатково проведені тести на безпеку даних підтвердили успішне відстоювання системою спроб несанкціонованого доступу та збереження конфіденційності інформації, утім, існує необхідність у вдосконаленні механізмів захисту від активного перехоплення та підробки даних.

Тест на стійкість до змін умов підтвердив адаптивність методики до змінних параметрів експлуатації. Зміни в обсязі потоку даних, частоті передачі та інших параметрах мали несуттєвий вплив, що свідчить про її придатність до реальних умов застосування.

Таким чином, розроблена криптографічна методика виявилася ефективною та готовою до застосування в реальних сценаріях, що підтверджує її потенціал у забезпеченні безпеки та захисту конфіденційної інформації.

## ВИСНОВКИ

В розвитку безпеки даних та захисту інформації, особливу увагу слід приділяти застосуванню криптографічних методів у бездротових сенсорних мережах. З урахуванням обмежених ресурсів сенсорних вузлів та специфіки безпроводного зв'язку, було поставлена мета розробити ефективне та стійке рішення, що забезпечує надійний захист конфіденційності та цілісності даних.

У контексті проведеного дослідження увага була спрямована на вивчення різних типів криптографічних алгоритмів, аналіз вимог до захисту даних в бездротових сенсорних мережах та розробку оптимізованої методики криптографічного захисту. Результати проведених тестів та аналізу висвітлюють важливі аспекти ефективності та стійкості розробленої системи.

Отримані результати дослідження криптографічного захисту даних в бездротових сенсорних мережах вказують на значний прогрес у забезпеченні ефективності та стійкості використовуваної методики. Огляд різних типів криптографічних алгоритмів виявив їхню різноманітність та специфіку застосування в контексті бездротових сенсорних мереж.

Аналіз основних аспектів захисту даних у бездротових сенсорних мережах дозволив визначити вимоги до системи захисту, що враховують обмежені ресурси сенсорних вузлів та специфіку комунікації через бездротове з'єднання. Розроблена гібридна модель, враховуючи ці вимоги, забезпечує оптимальний баланс між безпекою та продуктивністю системи.

Результати аналізу та тестування розробленої методики свідчать про її високий рівень ефективності та стійкості в умовах реального застосування. Тестування на коректність шифрування, резистентність до криптоаналізу та виявлення вразливостей підтвердили надійність системи. Високий рівень безпеки та резистентність до змін умов експлуатації дозволяють вважати методику ефективною у забезпеченні безпеки конфіденційної інформації в бездротових сенсорних мережах.

За результатами дослідження вдалося зробити такі висновки:



– різноманітність та специфіка криптографічних алгоритмів, що використовуються для захисту даних в бездротових сенсорних мережах, вимагають вибору оптимального алгоритму або комбінації алгоритмів з урахуванням конкретних умов застосування;

– система захисту даних у бездротових сенсорних мережах повинна враховувати обмежені ресурси сенсорних вузлів та специфіку комунікації через бездротове з'єднання, що впливають на безпеку та продуктивність системи;

– розроблена гібридна модель захисту даних, що поєднує симетричні та асиметричні криптографічні алгоритми, забезпечує оптимальний баланс між безпекою та продуктивністю системи, а також має гнучкість та адаптивність до змін умов експлуатації;

– розроблена методика захисту даних виявилася ефективною та стійкою в умовах реального застосування, що підтверджено результатами аналізу та тестування на коректність шифрування, резистентність до криптоаналізу та виявлення вразливостей;

– розроблена методика захисту даних є готовою до практичного впровадження в реальні сценарії використання в бездротових сенсорних мережах, що вимагають високого рівня безпеки конфіденційної інформації.

У подальших дослідженнях рекомендується розглядати можливості оптимізації методики з урахуванням специфіки різних застосувань в бездротових сенсорних мережах. Також важливим напрямком є дослідження нових алгоритмів та технологій криптографічного захисту, що враховують зростаючі вимоги до безпеки в сучасних інформаційних системах.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Базилевич В. М. Аналіз методів захисту від кіберзагроз в бездротових мережах стандарту IEEE 802.11 // Захист інформації. – 2017. – № 19 (3). – С. 222-227.
2. Безрук В. М., Власова В. О., Колтун Ю. М., Костромицький А. І. Вибір алгоритму самоорганізації бездротової сенсорної мережі методом аналізу ієрархій // Вісник Національного університету “Львівська політехніка”. Радіoeлектроніка та телекомунікації. – 2016. – № 849. – С. 179-184.
3. Бойко Ю. М. Концептуальні особливості реалізації безпроводних сенсорних мереж / Ю.М. Бойко, В.М. Локазюк, В.В. Мішан // Вісник Хмельницького національного університету. – 2010. – № 2. – С. 94–97.
4. Волошко С. В. Інформаційна безпека в безпроводових сенсорних мережах [Електронний ресурс] / С.В. Волошко, Д.О. Курца // Новітні інформаційні системи і технології. – 2018. – Випуск 9. – Режим доступу: <http://journals.pntu.edu.ua/mist/article/view/1039/869>.
5. Гнатушенко В. В. Алгоритм мінімізації енергоспоживання активними вузлами в бездротовій сенсорній мережі // Радіoeлектронні і комп'ютерні системи. – 2015. – № 2. – С. 88-92.
6. Інформаційна безпека в середовищі безпроводових сенсорних мереж: монографія / М.Б. Александер, С.М. Балабан, М.П. Карпінський, С.А. Райба, В.М. Чиж. – Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2016. – 160 с.
7. Кветний Р.Н. Метод та алгоритм обміну ключами серед груп користувачів на основі асиметричних шифрів ECC та RSA / Р.Н. Кветний, Є.О. Титарчук, А.А. Гуржій // Інформаційні технології та комп'ютерна інженерія. – 2016. – № 3. – С. 38-43.
8. Матов О. Я. Хеш-функції та цілісність інформаційних об'єктів / О.Я. Матов, В. С. Василенко // Реєстрація, зберігання і обробка даних. – 2014. – Т.16, № 4. – С. 12-17.

9. Носенко К., Півторак О., Ліхоузова Т. Огляд систем виявлення атак в мережевому трафіку // Адаптивні системи автоматичного управління. – К : НТУУ КПІ. – 2014. – № 1 (24). – С. 67-75.

10. Пороло Є. Удосконалена архітектура мережі для хмарного Інтернету речей / Є. Пороло, В. Курдеча // Перспективи телекомунікацій / Є. Пороло, В. Курдеча. – м. Київ, Україна: ISSN(print) 2663-502X, ISSN (online) 2664-3057, 2020. – С. 219–221.

11. Романов В.О. Вимоги до забезпечення функціональної та інформаційної безпеки бездротових сенсорних мереж / В.О. Романов, І.Б. Галелюка, В.О. Остапенко // Комп'ютерні засоби, мережі та системи. – 2017. – № 16. – С. 106-117.

12. Сич, К. В. Сучасна криптографія. симетричне та асиметричне шифрування / К. В. Сич ; наук. кер. С. М. Семедяй // Новітні технології у науковій діяльності і навчальному процесі : зб. тез Всеукр. наук.-практ. конф. студентів, аспірантів та молодих учених (м. Чернігів, 18-19 берез. 2021 р.) : збірник тез доп. – Чернігів : НУ «Чернігівська політехніка», 2021. – С. 166-167.

13. Шовкута В. А., Флоров С. В. Аналіз механізмів захисту та вразливостей бездротових WI-FI мереж. ДВНЗ «Національний гірничий університет», 2016. 10 с.

14. Abdulwahid, Ali, and Muwaffaq Salih. "Wireless Sensor Networks Applications, Challenges, and Security Requirements." Proceedings of 2nd International Multi-Disciplinary Conference Theme: Integrated Sciences and Technologies, IMDC-IST 2021, 7-9 September 2021, Sakarya, Turkey. 2022.

15. A. Faquih, P. Kadam and Z. Saquib, "Cryptographic techniques for wireless sensor networks: A survey," 2015 IEEE Bombay Section Symposium (IBSS), Mumbai, India, 2015, pp. 1-6, doi: 10.1109/IBSS.2015.7456652.

16. Balan, A., Balan, T., Cirstea, M. et al. A PUF-based cryptographic security solution for IoT systems on chip. J Wireless Com Network 2020, 231 (2020). <https://doi.org/10.1186/s13638-020-01839-6>

17. Belei, Oleksandr & Svatiuk, Oksana. (2020). Development of algorithm for encryption of messages in the wireless sensor network. *Cybersecurity: Education, Science, Technique*. 1. 69-84. 10.28925/2663-4023.2020.9.6984.
18. Bhavani, A., and V. Nithya. "Cryptographic algorithm for enhancing data security in wireless IoT sensor networks." *Intelligent Automation & Soft Computing* 36.2 (2023): 1381-1393.
19. Canetti, R., Shahaf, D., Vald, M. (2016). Universally Composable Authentication and Key-Exchange with Global PKI. In: Cheng, CM., Chung, KM., Persiano, G., Yang, BY. (eds) *Public-Key Cryptography – PKC 2016*. PKC 2016. *Lecture Notes in Computer Science()*, vol 9615. Springer, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-662-49387-8\\_11](https://doi.org/10.1007/978-3-662-49387-8_11)
20. Chung, Kai-Min, Michael Mitzenmacher, and Salil Vadhan. "Why simple hash functions work: Exploiting the entropy in a data stream." *Theory of Computing* 9.1 (2013): 897-945.
21. E. T. Oladipupo et al., "An Efficient Authenticated Elliptic Curve Cryptography Scheme for Multicore Wireless Sensor Networks," in *IEEE Access*, vol. 11, pp. 1306-1323, 2023, doi: 10.1109/ACCESS.2022.3233632
22. Gautam, A.K., Kumar, R. A comprehensive study on key management, authentication and trust management techniques in wireless sensor networks. *SN Appl. Sci.* 3, 50 (2021). <https://doi.org/10.1007/s42452-020-04089-9>
23. Günther, Felix. "Modeling advanced security aspects of key exchange and secure channel protocols" *it – Information Technology*, vol. 62, no. 5-6, 2020, pp. 287-293. <https://doi.org/10.1515/itit-2020-0029>
24. H. Modares, A. Moravejosharieh and R. Salleh, "Wireless Network Security Using Elliptic Curve Cryptography," 2011 First International Conference on Informatics and Computational Intelligence, Bandung, Indonesia, 2011, pp. 348-351, doi: 10.1109/ICI.2011.63.
25. Jena B. K. What Is AES Encryption and How Does It Work? [Electronic resource] / Baivab Kumar Jena // [Simplilearn.com](https://simplilearn.com). – Mode of access:

<https://www.simplilearn.com/tutorials/cryptography-tutorial/aes-encryption> (date of access: 10.02.2023).

26. K. Alanezi and S. Mishra, "A privacy negotiation mechanism for the internet of things," in IEEE 16th International Conference on Dependable, Autonomic and Secure Computing, 16th International Conference on Pervasive Intelligence and Computing, 4th International Conference on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2018, pp. 512–519.

27. Kapoor, J., Thakur, D. (2022). Analysis of Symmetric and Asymmetric Key Algorithms. In: Fong, S., Dey, N., Joshi, A. (eds) ICT Analysis and Applications. Lecture Notes in Networks and Systems, vol 314. Springer, Singapore, doi: 10.1007/978-981-16-5655-2\_13

28. Lee, Cheng-Chi. 2020. "Security and Privacy in Wireless Sensor Networks: Advances and Challenges" *Sensors* 20, no. 3: 744. <https://doi.org/10.3390/s20030744>

29. Li, M. Automatic Encryption Method of Sensor Network Capture Data Based on Symmetric Algorithm. *Wireless Pers Commun* 127, 353–367 (2022). <https://doi.org/10.1007/s11277-021-08267-9>

30. M. AlRoubiei, T. AlYarubi and B. Kumar, "Critical Analysis of Cryptographic Algorithms," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, 2020, pp. 1-7, doi: 10.1109/ISDFS49300.2020.9116213.

31. M. Althamir, A. Alabdulhay and M. M. Yasin, "A Systematic Literature Review on Symmetric and Asymmetric Encryption Comparison Key Size," 2023 3rd International Conference on Smart Data Intelligence (ICSMDI), Trichy, India, 2023, pp. 110-117, doi: 10.1109/ICSMDI57622.2023.00027

32. Martin, Keith M., 'Data Integrity', *Everyday Cryptography: Fundamental Principles and Applications*, 2nd edn (Oxford, 2017; online edn, Oxford Academic, 20 July 2017), <https://doi.org/10.1093/oso/9780198788003.003.0006>

33. M. B. Yassein, S. Aljawarneh, E. Qawasmeh, W. Mardini and Y. Khamayseh, "Comprehensive study of symmetric key and asymmetric key encryption algorithms," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-7, doi: 10.1109/ICEngTechnol.2017.8308215.
34. N. Mouha, M. Dworkin et al., "Review of the advanced encryption standard," Technical report, National Institute of Standards and Technology, 2021.
35. Oleiwi, Zahraa Ch, et al. "Overview and Performance Analysis of Encryption Algorithms." Journal of physics: conference series. Vol. 1664. No. 1. IOP Publishing, 2020.
36. P. -Y. Ting, J. -L. Tsai and T. -S. Wu, "Signcryption Method Suitable for Low-Power IoT Devices in a Wireless Sensor Network," in IEEE Systems Journal, vol. 12, no. 3, pp. 2385-2394, Sept. 2018, doi: 10.1109/JSYST.2017.2730580.
37. Saleh, I. "Comparative analysis of modern methods and algorithms of cryptographic protection of information." International Journal of Computer Science and Information Security (IJCSIS) 15.12 (2017).
38. Silva, C., Cunha, V.A., Barraca, J.P. et al. Analysis of the Cryptographic Algorithms in IoT Communications. Inf Syst Front (2023). <https://doi.org/10.1007/s10796-023-10383-9>
39. Soni, Ankit Kumar, et al. Comparative Analysis of Cryptographic Algorithms in Computer Network. No. 10218. EasyChair, 2023.
40. S. Sinha and S. Aggarwal, "Cryptographic Algorithms for Security in Wireless Sensor Networks," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2022, pp. 111-117, doi: 10.1109/ICIEM54221.2022.9853139.
41. Toms L. 5 Common Cyber Attacks in the IoT – Threat Alert on a GrandScale [Electronic resource] / L.Toms // Global Sign. – 2016. – Access: <https://www.globalsign.com/en/blog/five-common-cyber-attacks-in-the-iot/>.

42. Qazi, R., Qureshi, K.N., Bashir, F. et al. Security protocol using elliptic curve cryptography algorithm for wireless sensor networks. *J Ambient Intell Human Comput* 12, 547–566 (2021). <https://doi.org/10.1007/s12652-020-02020-z>
43. Wireless Sensor Network Security for Cyber-Physical Systems / Saqib Ali, Taiseera Al, Balushi Zia, Nadir Omar, Khadeer Hussain // *Cyber Security for Cyber Physical Systems. Studies in Computational Intelligence.* – 2018. – Vol. 768. – P. 35-63.
44. Wireless Sensor Network (WSN) [Electronic resource] // *GeeksforGeeks.* – Mode of access: <https://www.geeksforgeeks.org/wireless-sensor-network-wsn/> (date of access: 16.05.2023).
45. X. Du and H. -h. Chen, "Security in wireless sensor networks," in *IEEE Wireless Communications*, vol. 15, no. 4, pp. 60-66, Aug. 2008, doi: 10.1109/MWC.2008.4599222.