Форма № 24

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
(повне найменування факультету)

Кафедра інформаційної безпеки та наноелектроніки
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)
магістр
(ступінь вищої освіти)

на тему Дослідження захисту Інтернету речей від кіберзагроз

(назва теми)

Виконав: студент 2 курсу, групи БКз -814м
Спеціальності 125 Кібербезпека та захист інформації
(код і найменування спеціальності)
Освітня програма (спеціалізація)
Безпека інформаційних і комунікаційних систем
МОХАМАД Мохамад
(ПРІЗВИЩЕ та ініціали)

Керівник КОЗІНА Г.Л.
(ПРІЗВИЩЕ та ініціали)

Рецензент МОРОЗ Г. В.
(ПРІЗВИЩЕ та ініціали)

2025

Форма № 25

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
Кафедра інформаційної безпеки та наноелектроніки
Ступінь вищої освіти магістр
Спеціальність 125 Кібербезпека та захист інформації
(код і найменування)
Освітня програма (спеціалізація) Безпека інформаційних і комунікаційних систем
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ
Завідувач кафедри ІБтаН
Андрій КОРОТУН
«____» _____2025р.

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

МОХАМАД Мохамад
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження захисту Інтернету речей від кіберзагроз.

Research on protecting the Internet of Things from cyber threats.

керівник проєкту (роботи) канд. фіз.-мат. наук, доцент кафедри ІБтаН, КОЗІНА Галина Леонідівна.
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « » листопада 2025 року №

2. Строк подання студентом проєкту (роботи) 22.12.2025

3. Вихідні дані до проєкту (роботи) дослідження, спрямовані на захист пристроїв Інтернету речей від кіберзагроз.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)) Ці напрямки досліджень зосереджені на розумінні впливу цифрової трансформації на IoT, виявленні та усуненні нових кіберзагрози, оцінці ефективності засобів контролю безпеки, дослідженні таких технологій, як Машинне навчання і штучний інтелект для підвищення безпеки, а також заохочення співпраці зацікавлених сторін для надійної безпеки IoT.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)
Презентація доповіді (в MS PowerPoint), 11 слайдів.

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1 – 6	КОЗІНА Г. Л., доцент кафедри ІБтаН	04.09.2025	19.12.2025
Нормоконтроль	КОРОЛЬКОВ Р. Ю., доцент кафедри ІБтаН	20.12.2025	20.12.2025

7. Дата видачі завдання «04» вересня 2025року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз літературних джерел за тематикою дослідження.	04.09.25 – 20.09.25	Виконано
2	Підготовка теоретичного матеріалу.	21.09.25 – 30.09.25	Виконано
3	Аналіз результатів попередніх досліджень, пов'язаних з дослідженням.	01.10.25 – 10.10.25	Виконано
4	Підготовка моделей оптимізації для кібербезпеки Інтернету речей.	11.10.25 – 20.11.25	Виконано
5	Підготовка результатів експерименту.	21.11.25 – 10.12.25	Виконано
6	Оформлення матеріалів магістерської роботи.	11.12.25 – 19.12.25	Виконано

Студент _____ Мохамад МОХАМАД
(підпис) (Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи) _____ Галіна КОЗІНА
(підпис) (Ім'я ПРІЗВИЩЕ)

АНОТАЦІЯ

Пояснювальна записка до магістерської роботи: 109 с., 13 табл., 8 рис., 1 дод., 25 джерела.

ІНТЕРНЕТ РЕЧЕЙ (IOT), ВИЯВЛЕННЯ DDOS-АТАК, МАШИННЕ НАВЧАННЯ (ML), PYTHON, ГЛИБОКЕ НАВЧАННЯ, RESNET, ВИЯВЛЕННЯ АНОМАЛІЙ, НАБІР ДАНИХ N-BAIOT, КІБЕРБЕЗПЕКА.

Об’єкт дослідження – Процеси забезпечення безпеки пристроїв Інтернету речей.

Предмет дослідження – Алгоритмічні методи виявлення та запобігання кіберзагрозам в IoT-середовищах.

Мета роботи – дослідити характерні для IoT кіберзагрози, проаналізувати сучасні підходи до їх виявлення та запропонувати удосконалений метод виявлення атак на основі глибинних ансамблевих моделей.

У дослідженні було використано реальний набір даних N-BaIoT, що містить мережевий трафік ботнет-атак, таких як Mirai та Gafgyt. У роботі пропонуються вдосконалення базових методів виявлення загроз Інтернету речей шляхом поєднання глибоких нейронних мереж з ансамблевими алгоритмами та тонким налаштуванням гіперпараметрів. Це дозволило підвищити точність класифікації, зменшити кількість хибних спрацьовувань та скоротити час обробки даних – фактори, критично важливі для систем моніторингу в реальному часі.

ABSTRACT

Explanatory note to the master's thesis: 109 p., 13 tables, 8 figures, 1 appendixes, 25 sources.

INTERNET OF THINGS (IOT), DDOS DETECTION, MACHINE LEARNING (ML), PYTHON, DEEP LEARNING, RESNET, ANOMALY DETECTION, N-BAIOT DATASET, CYBERSECURITY.

Object of research – Processes of ensuring the security of Internet of Things devices.

Subject of research – Algorithmic methods for detecting and preventing cyber threats in IoT environments.

Purpose of the work – to investigate IoT-specific cyber threats, analyze modern approaches to their detection, and propose an improved attack detection method based on deep ensemble models.

A real N-BaIoT dataset containing network traffic of botnet attacks such as Mirai and Gafgyt was used in the study. The work proposes improvements to basic IoT threat detection methods by combining deep neural networks with ensemble algorithms and fine-tuning hyperparameters. This made it possible to increase classification accuracy, reduce the number of false positives, and shorten data processing time - factors critical for real-time monitoring systems.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

AI – Artificial Intelligence (штучний інтелект)

API – Application Programming Interface (інтерфейс прикладного програмування)

DDoS – Distributed Denial of Service (розподілена атака типу відмова в обслуговуванні)

DNS – Domain Name System (система доменних імен)

ETSI – European Telecommunications Standards Institute (Європейський інститут стандартів телекомунікацій)

F1-міра – показник точності та повноти класифікації (F1-score)

HTTPS – HyperText Transfer Protocol Secure (протокол безпечної передачі гіпертексту)

ICS – Industrial Control System (промислова система керування)

IEC – International Electrotechnical Commission (Міжнародна електротехнічна комісія)

IoT – Internet of Things (Інтернет речей)

IP – Internet Protocol (інтернет-протокол)

ISO – International Organization for Standardization (Міжнародна організація зі стандартизації)

LAN – Local Area Network (локальна мережа)

ML – Machine Learning (машинне навчання)

N-BaIoT – N-Botnet Attacks in Internet of Things dataset (набір даних ботнет-атак в IoT)

NIST – National Institute of Standards and Technology (Національний інститут стандартів і технологій, США)

OS – Operating System (операційна система)

RAM – Random Access Memory (оперативна пам’ять)

ResNet – Residual Neural Network (залишкова нейронна мережа)

TLS – Transport Layer Security (захист транспортного рівня)

VPN – Virtual Private Network (віртуальна приватна мережа)

Wi-Fi – Wireless Fidelity (бездротова передача даних)

ЗКЗ – засоби криптографічного захисту

КСЗІ – комплексна система захисту інформації

ПК – персональний комп’ютер

ЦОД – центр обробки даних

ВСТУП

Сучасна цифрова трансформація супроводжується стрімким поширенням технологій Інтернету речей (Internet of Things, IoT), які забезпечують автоматизовану взаємодію між фізичними пристроями та інформаційними системами без участі людини. IoT-пристрої формують інтегровані мережі, у межах яких здійснюється збір, передавання, обробка та аналіз даних у режимі реального часу. До таких пристроїв належать сенсори, виконавчі модулі, контролери, комунікаційні шлюзи та інші вбудовані системи, що функціонують у промисловості, енергетиці, логістиці, охороні здоров'я, міській інфраструктурі, аграрному секторі та побуті.

За прогнозами провідних аналітичних компаній, зокрема Gartner та McKinsey, до 2030 року кількість активних IoT-пристроїв перевищить 25 млрд. Відповідно до зростання масштабів екосистеми збільшується й поверхня потенційних атак, що формує нові ризики для інформаційної безпеки. Найтиповішими загрозами залишаються несанкціонований віддалений доступ, перехоплення та модифікація конфіденційних даних, ін'єкції шкідливого коду, DDoS-атаки, а також використання вразливих пристроїв як елементів ботнет-мереж. У більшості випадків атаки стають можливими через недоліки прошивок, незахищені або неправильно налаштовані мережеві протоколи, слабкі або відсутні механізми автентифікації, а також через відсутність регулярних оновлень безпеки.

Оскільки традиційні засоби мережевої безпеки не враховують обмеженість ресурсів IoT-пристроїв, їхню гетерогенність та розподілену архітектуру, вони стають недостатньо ефективними у сучасних умовах. Саме тому дослідження інтелектуальних підходів до виявлення аномальної активності, побудованих на методах машинного та глибинного навчання, посідає ключове місце у сфері IoT-безпеки.

У даній роботі досліджується метод виявлення кіберзагроз у мережах Інтернету речей, заснований на поєднанні глибокої резидуальної нейронної мережі ResNet із ансамблевою моделлю Stacked Ensemble. Для експериментальної оцінки використано реальні набори даних, зокрема N-BaIoT, який містить детальні телеметричні характеристики трафіку ботнет-атак типу Mirai та Gafgyt. Проведено повний цикл обробки даних: масштабування, формування ознак, навчання моделей, тестування, порівняння з альтернативними методами. Оцінка ефективності здійснювалась за метриками точності, recall, F1-міри, часової продуктивності та здатності моделі виявляти раніше невідомі зразки атак.

Метою дослідження є поглиблений аналіз кіберзагроз, притаманних IoT-середовищам, оцінка сучасних підходів до їх виявлення, а також розробка та експериментальна перевірка методу класифікації атак на основі глибинних ансамблевих моделей. Отримані результати мають прикладне значення для побудови систем раннього виявлення загроз у мережах IoT, а також можуть бути використані в інтелектуальних промислових системах, енергетичних мережах, інфраструктурі «розумного міста» та інших критично важливих середовищах.

1 ВИКЛИКИ БЕЗПЕЦІ ІНТЕРНЕТУ РЕЧЕЙ

1.1 Безпека Інтернету речей: значення, виклики та практична необхідність

Інтернет речей (Internet of Things, IoT) сьогодні посідає ключове місце серед технологій цифрової трансформації, формуючи нову епоху взаємодії між людиною, машиною й навколишнім середовищем. Сутність цієї концепції полягає у створенні єдиного інформаційного простору, де фізичні об’єкти - від побутових приладів до промислового обладнання - можуть самостійно обмінюватися даними, аналізувати їх і виконувати дії без безпосередньої участі людини. Такі системи функціонують завдяки сенсорам, процесорам обробки даних і комунікаційним інтерфейсам, що забезпечують постійний обмін інформацією між цифровим і фізичним світом. Згідно зі стандартом ISO/IEC 30141:2018, типова архітектура IoT складається з п’яти рівнів: сенсорного, комунікаційного, обробки даних, прикладного та користувацького [1]. Дана структура забезпечує узгоджене функціонування складних систем, у яких пристрої різного типу взаємодіють у єдиному цифровому середовищі.

Інтернет речей поступово перетворився з експериментальної концепції на практичну реальність, що охоплює практично всі сфери людської діяльності. Уже сьогодні «розумні» пристрої використовуються у транспорті, медицині, енергетиці, сільському господарстві, торгівлі, логістиці та побуті. Їхня інтеграція у щоденне життя сприяє підвищенню ефективності процесів, економії ресурсів і зручності користувачів. Водночас, разом із перевагами, стрімке поширення IoT несе і серйозні виклики у сфері безпеки. Кожен новий пристрій, що підключається до мережі, збільшує так звану «поверхню атаки», тобто кількість потенційних точок, через які зловмисник може отримати несанкціонований доступ до системи. У світі, де мільярди пристроїв

передають дані в реальному часі, навіть невелика вразливість може мати глобальні наслідки.

За оцінками McKinsey Global Institute (2014), до 2030 року кількість підключених IoT-пристроїв перевищить 25 мільярдів [2]. Така експансія пояснюється стрімким розвитком хмарних технологій, здешевленням сенсорів і зростанням попиту на автоматизацію. Проте кількісне зростання не завжди супроводжується якісним підвищенням рівня безпеки. Значна частина IoT-пристроїв має елементарні уразливості - типові паролі, відсутність шифрування трафіку або механізму оновлення програмного забезпечення. У результаті навіть прості побутові гаджети можуть стати знаряддям масштабних кібератак. Відомим прикладом стала атака ботнету Mirai у 2016 році, коли тисячі заражених пристроїв - переважно веб-камер і маршрутизаторів - були використані для здійснення потужної DDoS-атаки, яка вивела з ладу низку провайдерів у США та Європі. Цей інцидент продемонстрував, наскільки небезпечним може бути злам навіть «звичайних» пристроїв, якщо вони масово інтегровані в мережу.

Дослідження Palo Alto Networks Unit 42 (2020) підтверджує масштаб проблеми: понад 98 % мережевого трафіку IoT передається без шифрування, а близько 57 % пристроїв мають відомі уразливості, які можна використати для атак [3]. Така ситуація свідчить про системну відсутність пріоритету безпеки серед розробників. Багато виробників, прагнучи мінімізувати вартість продукції, зосереджуються на функціональності, ігноруючи питання захисту даних. Як наслідок, кібербезпека IoT залишається на периферії уваги, попри її критичне значення для стабільності всієї інфраструктури.

Вирішення проблеми потребує запровадження єдиних міжнародних стандартів. У Європі важливу роль відіграє ETSI EN 303 645:2020, який визначає ключові вимоги до безпеки споживчих IoT-пристроїв [4]. Серед основних принципів - заборона універсальних паролів, впровадження безпечних механізмів оновлення програмного забезпечення, прозорість політик зберігання й видалення даних. Проте дотримання цього стандарту

нині не є обов’язковим, що знижує його ефективність. У США аналогічні підходи реалізовано в документі NIST SP 800-213 (2021), який наголошує на принципі security by design - безпеці, закладеній ще на етапі розробки [5]. Такий підхід передбачає, що кожен пристрій має містити вбудовані механізми автентифікації, шифрування, ізоляції мережевих сегментів та реагування на інциденти. Саме запровадження подібних принципів може зменшити ризики, пов’язані з масовим підключенням пристроїв до відкритих мереж.

Особливої ваги питання безпеки набуває у сферах, що мають критичне значення для суспільства. Сьогодні IoT активно використовується у медицині, транспорті, енергетиці та промисловості. У сфері охорони здоров’я технології IoT дозволяють здійснювати дистанційний моніторинг стану пацієнтів, аналізувати життєві показники в реальному часі та оперативно реагувати на зміни. За даними World Health Organization Europe (2025), пілотні програми телемедицини, що базуються на IoT-сенсорах, знизили кількість повторних госпіталізацій завдяки ранньому виявленню ускладнень [6]. У такий спосіб IoT-рішення підвищують якість медичних послуг, але водночас породжують питання конфіденційності даних - адже злам системи може призвести до розголошення чутливої інформації про пацієнтів.

У промисловому секторі Інтернет речей є фундаментом концепції Industry 4.0. Промислові сенсори відстежують температуру, вібрацію, рівень навантаження обладнання й передають дані в аналітичні системи, які прогнозують збої та оптимізують обслуговування. За дослідженням McKinsey & Company (2018), використання промислових IoT-рішень дозволяє зменшити витрати на технічне обслуговування на 18–25 % та підвищити продуктивність до 20 % [7]. Такі результати демонструють, що безпечна та правильно налаштована інфраструктура IoT має реальний економічний ефект.

У сільському господарстві Інтернет речей став основою технологій точного землеробства. Сенсори моніторять вологість грунту, рівень

освітлення та погодні умови, допомагаючи оптимізувати використання води та добрив. За даними FAO (2022), використання IoT у фермерстві дозволяє знизити споживання води на 30 % і підвищити урожайність на 20 % [8]. Однак сільськогосподарські системи також потребують захисту, адже вразливість у ланцюзі постачання продовольства може мати масштабні наслідки для економічної та продовольчої безпеки.

Не менш значну роль IoT відіграє в урбаністичних проєктах. Концепція «розумного міста» (Smart City) використовує мережі сенсорів для управління транспортними потоками, освітленням, енергоспоживанням і системами безпеки. За даними World Economic Forum (2023), впровадження IoT у міській інфраструктурі дозволяє скорочувати енергоспоживання на 30 % і зменшувати рівень заторів на 20 % [9]. Водночас урбаністичні системи стають мішенню для кібератак, які можуть порушити роботу транспорту, електромереж чи служб порятунку.

Крім того, Інтернет речей активно впроваджується в енергетиці через розумні мережі, що забезпечують ефективний розподіл навантаження та інтеграцію відновлюваних джерел енергії. У фінансовому секторі IoT-технології допомагають у біометричній автентифікації та моніторингу транзакцій у реальному часі, а в освітній галузі створюють інтелектуальні аудиторії та системи контролю доступу. Таким чином, IoT стає універсальною платформою цифрової взаємодії, що з’єднує технології, людей і середовище в єдину мережу.

Зростання залежності суспільства від IoT-систем робить питання їхнього захисту не лише технічним, а й стратегічним. Компрометація таких систем може спричинити зупинку виробництва, порушення енергопостачання чи загрозу життю людей. Як підкреслює OECD (2023), розвиток Інтернету речей потребує єдиної міжнародної політики кібербезпеки, яка поєднує технічні стандарти, освітні ініціативи та правові механізми захисту [10]. Без узгодженої системи регулювання та контролю

Інтернет речей ризикує стати не рушійною силою прогресу, а джерелом нових загроз для суспільства та економіки.

Отже, безпека Інтернету речей є невід'ємною умовою сталого розвитку цифрової економіки. Зростання кількості підключених пристроїв створює необхідність формування глобальної культури кібербезпеки, де всі учасники - розробники, користувачі та регулятори - усвідомлюють спільну відповідальність за захист інформаційного простору. Лише за умови впровадження принципу «безпека за замовчуванням» та міжнародної координації зусиль IoT зможе залишатися технологією, яка покращує якість життя, а не створює нові загрози.

1.2 Основні загрози та принципи безпеки Інтернету речей

Стрімке зростання Інтернету речей докорінно змінює технологічний ландшафт. Мережі сенсорів, контролерів і «розумних» пристроїв охоплюють побут, промисловість, енергетику, транспорт і державне управління. Разом із користю невідворотно зростає й рівень ризиків. Чим ширше IoT інтегрується у життєдіяльність суспільства, тим вищою стає ціна потенційних збоїв та атак. Якщо раніше пріоритетом була функціональність, то нині саме безпека визначає стійкість інфраструктури та довіру користувачів.

Відповідно до ENISA Threat Landscape 2023 [11], за останні роки суттєво зросла інтенсивність атак на екосистеми IoT. Найчастіше під ударом опиняються «розумні» камери та сенсори, промислові контролери і системи моніторингу середовища. Важлива особливість цих середовищ - взаємозалежність компонентів: компрометація одного вузла здатна миттєво вплинути на інші та запустити "ланцюгову реакцію". У корпоративних і міських мережах це обертається збоями в транспорті, енергетиці чи сервісах життєзабезпечення.

Реальні інциденти підтверджують масштаб загроз. У 2021 році у Флориді зловмисники отримали віддалений доступ до системи автоматизованого керування водоочисною станцією міста Олдсмар і спробували змінити дозування реагентів. Лише швидке втручання персоналу відвернуло потенційно небезпечні наслідки для населення. Подібні події показують: атаки на IoT - це не лише фінансові втрати, а й пряма загроза безпеці людей та довірі до «розумної» інфраструктури.

Проблеми у базовій гігієні безпеки - ще одна причина вразливості. Галузеві звіти за 2023 рік відзначають, що в багатьох організаціях досі трапляються незмінені стандартні паролі, відсутній постійний моніторинг трафіку, а політики оновлення прошивок запроваджені фрагментарно [12]. У таких умовах стають можливими класичні атаки на зразок man-in-the-middle, зловживання відкритими API та формування ботнетів (історично показовий приклад - Mirai). Проблему поглиблює те, що значна частка споживчих пристроїв або зовсім не підтримує оновлення безпеки, або втрачає підтримку надто швидко.

Окремим вектором ризику залишаються атаки на ланцюги постачання. Практика показує: компрометація відбувається не лише в експлуатації, а й на етапах виробництва, логістики та оновлення ПЗ. Профільні огляди 2024 року наголошують, що у сфері IoT помітна частка інцидентів пов’язана саме з постачальницькими каналами та суміжними партнерами, тож вибудова вимог безпеки до вендорів стає критичною [13]. Це прямий аргумент на користь принципів security by design і zero trust [5]: контроль має бути вмонтований в архітектуру, а довіра - ніколи не видається «за замовчуванням».

Сучасна практика захисту рухається саме в бік Zero Trust. Дослідження щодо нановизначення цієї парадигми для IoT підкреслюють необхідність постійної верифікації кожного учасника взаємодії, мінімізації прав доступу, сегментації мереж і наскрізного шифрування трафіку, а також безперервного моніторингу поведінки пристроїв із автоматизованою реакцією на аномалії [14]. У реальних умовах це означає комбінацію MFA для адміністративних

доступів, взаємної автентифікації пристрій-до-пристрою, ротації ключів, контрольованих оновлень прошивок і ведення достовірних журналів подій.

Водночас навіть найкращі технічні механізми не дадуть результату без належної організації процесів та відповідальної поведінки користувачів. Ефективна безпека IoT - це поєднання технологічного, організаційного та людського рівнів. На технологічному рівні йдеться про криптографічний захист, сертифікацію пристроїв, безпечні оновлення й використання перевірених протоколів (TLS/DTLS, HTTPS, MQTT із захистом каналу; див. [4], [5]). На організаційному - про управління ризиками, аудит, SOC-моніторинг і відпрацьовані плани реагування. Людський рівень - це постійна просвіта, зменшення помилок у налаштуваннях, увага до політик доступу й паролів, культура відповідального використання.

Окремо слід наголосити на конфіденційності. IoT-пристрої збирають чутливі дані - від геолокації до медичних показників. Європейська практика прямо вимагає, щоб дизайн таких систем спирався на privacy by design/by default: мінімізацію збору, чітке цільове призначення, прозорість обробки, контроль користувача над власними даними, належну інформовану згоду та безпечне видалення/передачу [15]. Дотримання цих принципів - не лише юридична вимога, а й основа довіри до «розумних» сервісів.

1.3 Ключові виклики безпеки в середовищі IoT

Інтернет речей поступово перетворюється з інструмента автоматизації на критичну складову цифрової інфраструктури, і саме це робить питання його безпеки винятково складним. Як зазначалося раніше [11], Європейське агентство ENISA у звіті Threat Landscape for IoT 2023 підкреслює: проблема полягає не лише у кількості атак, а у зміні самої природи ризиків. Вразливості більше не обмежуються рівнем окремих пристроїв - вони

поширюються на взаємозв’язки, залежності та потоки даних між системами. Іншими словами, сучасні виклики IoT - це виклики екосистемного масштабу.

Однією з головних проблем є архітектурна неоднорідність середовища. IoT-пристрої створюються різними виробниками, працюють на різних протоколах і часто не підтримують спільних стандартів безпеки. У корпоративних та промислових мережах це призводить до ситуацій, коли окремі модулі не можуть бути захищені централізовано, а застосування оновлень безпеки потребує індивідуального підходу. Як відзначає ENISA [11], це породжує «острівну» архітектуру, де загальний рівень безпеки визначається найслабшою ланкою.

Не менш складним викликом є динамічність середовища IoT. Пристрої постійно підключаються й відключаються, змінюють IP-адреси, взаємодіють із хмарними сервісами, локальними шлюзами чи мобільними застосунками. Така мінливість ускладнює виявлення аномальної поведінки, адже класичні системи моніторингу не розрізняють, де новий пристрій, а де - компрометований. У звіті Cyber Security Report 2023 [12] наголошується, що саме “невидимі” з’єднання - короткочасні, тимчасові, нестандартні - часто стають точками початку інцидентів. Це створює необхідність переходу до поведінкових моделей аналізу, заснованих на машинному навчанні.

Ще один ключовий виклик - відсутність повного контролю над життєвим циклом пристроїв. Як показує Microsoft Digital Defense Report 2024 [13], більшість IoT-компонентів проходять через кілька ланцюгів постачання: від виробника мікрочипів до інтегратора систем. На кожному етапі можливе втручання - наприклад, модифікація прошивки або заміна компонентів. Особливо небезпечні випадки, коли пристрій здається безпечним на момент покупки, але вразливості з’являються через кілька років, коли підтримка виробника вже припинена. Microsoft зазначає, що лише 22 % компаній мають політику перевірки постачальників на відповідність вимогам кіберзахисту.

У відповідь на такі виклики дедалі більше організацій переходять до архітектури Zero Trust. Ця концепція, розкрита у праці Revisiting Zero-Trust

Security for Internet of Things (ResearchGate, 2023) [14], заперечує будь-яку довіру за замовчуванням і пропонує постійно перевіряти кожен пристрій, користувача й навіть внутрішній трафік(рисунок 1.1). Її сила - у гнучкості: якщо класична модель безпеки будується навколо периметра, то Zero Trust виходить із припущення, що загроза вже всередині. Для IoT, де периметр практично відсутній, це єдиний життєздатний підхід.

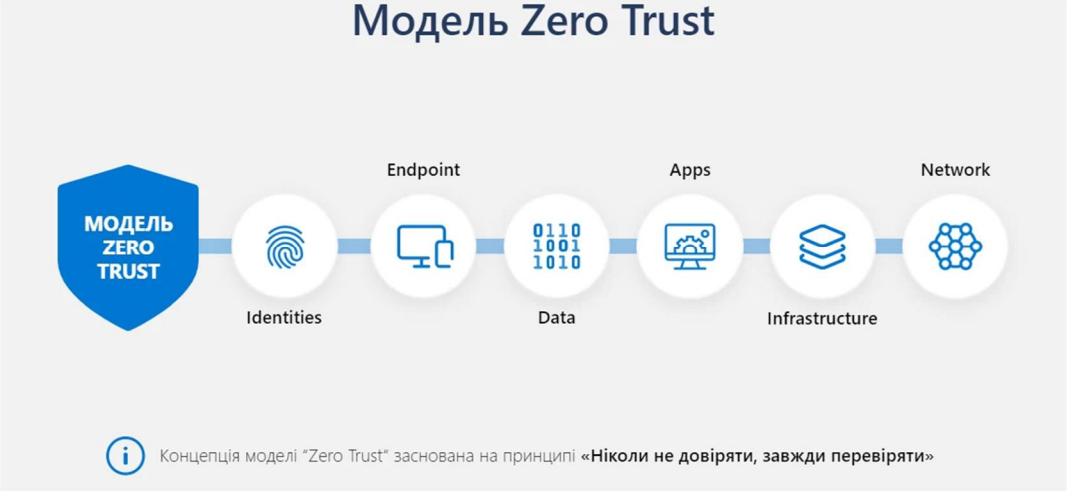

Рисунок 1.1 – Модель Zero Trust [23]

Проблема захисту даних і конфіденційності також набула нових вимірів. Пристрої IoT не лише збирають інформацію, а й передають її між різними юрисдикціями, де діють різні закони. Як підкреслює European Data Protection Board у документі Data Protection by Design and by Default (2023) [15], принципи GDPR поширюються і на екосистеми IoT: розробник має закладати конфіденційність у конструкцію пристрою, забезпечуючи прозоре інформування користувачів, обмеження цілей збору даних і можливість видалення інформації. На практиці це часто ігнорується, особливо в дешевих споживчих пристроях, де функціональність пріоритетніша за безпеку.

Для ілюстрації взаємозв'язку між основними типами викликів та рівнем їхнього впливу на безпеку системи наведено узагальнення (таблиця 1.1).

Як видно, більшість проблем взаємопов'язані. Наприклад, недостатня стандартизація підсилює ризики ланцюгів постачання, а людський фактор -

наслідок складності управління гетерогенними системами. Саме тому сучасні підходи до кіберзахисту IoT зосереджуються не лише на технологічних рішеннях, а на формуванні адаптивної політики безпеки, здатної навчатися, передбачати загрози та реагувати проактивно.

Таблиця 1.1 – Групи викликів безпеки IoT та їхній системний вплив

Група викликів	Суть проблеми	Вплив на безпеку
Архітектурна неоднорідність	Різні платформи, відсутність єдиних стандартів	Неможливість централізованого захисту
Динамічність підключень	Часті зміни пристроїв у мережі	Ускладнене виявлення вторгнень
Ланцюги постачання	Неконтрольовані етапи виробництва	Можливість прихованих закладок
Витоки даних	Недотримання принципів GDPR	Репутаційні та юридичні наслідки
Людський фактор	Недосвідченість користувачів	Підвищення ризику інцидентів

Згідно з рекомендаціями ENISA [11] і Microsoft [13], для забезпечення такої стійкості необхідно поєднувати кілька стратегій:

- впровадження багаторівневих механізмів ідентифікації;
- використання цифрових сертифікатів та перевірки цілісності компонентів;
- постійний моніторинг поведінки пристроїв із застосуванням аналітики штучного інтелекту;
- розподіл зон довіри замість побудови одного периметра захисту.

Окремо підкреслюється важливість розвитку кіберосвіти серед користувачів і персоналу. Як наголошується у Cyber Security Report 2023 [12], понад 40 % інцидентів у IoT виникають не через технічні вразливості, а через людські помилки - використання слабких паролів, ігнорування

оновлень або відсутність розуміння ризиків. Це означає, що без формування культури цифрової безпеки жодні технічні засоби не дадуть стійкого ефекту.

Щоб узагальнити основні напрями подолання виявлених бар'єрів, у таблиці 1.2 наведено типові виклики й відповідні стратегії реагування, рекомендовані міжнародними організаціями.

Таблиця 1.2 – Основні виклики IoT-безпеки та рекомендовані напрями реагування

Виклик	Стратегія реагування
Відсутність стандартизації	Єдиний набір технічних вимог (ETSI EN 303 645)
Неконтрольовані постачання	Сертифікація ланцюгів, аудит постачальників
Динамічні атаки	Поведінкова аналітика, машинне навчання
Низька довіра в мережі	Впровадження Zero Trust архітектури
Порушення конфіденційності	Інтеграція принципів Privacy by Design
Людський фактор	Освіта користувачів, автоматизація моніторингу

Ключові виклики безпеки в IoT уже не зводяться до технічних дефектів - це системна проблема, де технологічні, організаційні та поведінкові аспекти тісно переплетені. Як зазначає ENISA [11], головна умова подолання цих ризиків - комплексний підхід, що поєднує інженерію, управління ризиками та міжнародну співпрацю. Лише в такому разі Інтернет речей зможе розвиватися як безпечна основа майбутньої цифрової економіки.

1.4 Багаторівнева модель забезпечення безпеки IoT

Одним із фундаментальних підходів до забезпечення кібербезпеки в екосистемі Інтернету речей є впровадження багаторівневої моделі захисту. Даний підхід дає змогу диференційовано охопити всі складові IoT-системи - від фізичних пристроїв до рівня взаємодії кінцевого користувача. З

урахуванням складної структури та розподіленого характеру IoT-інфраструктури, кожен рівень безпеки виконує специфічні функції й потребує окремих механізмів протидії загрозам. Як зазначає ISO/IEC 30141:2018 [1], надійна архітектура Інтернету речей передбачає поєднання технічних, організаційних і поведінкових аспектів, які утворюють єдину систему довіри.

Як уже підкреслювалося раніше [11], безпека IoT не може забезпечуватися одношаровими рішеннями. Вона формується через узгоджену взаємодію кількох рівнів, кожен з яких компенсує потенційні вразливості іншого. Цілісність даної архітектури грунтується на принципі “захисту в глибину” (defense in depth), коли порушення на одному рівні не обов’язково призводить до загального компромісу системи, оскільки інші елементи продовжують виконувати стримувальні функції. Саме тому розроблення чітко структурованої багаторівневої моделі безпеки набуває першочергового значення у процесі побудови надійних IoT-систем.

Початковим елементом даної структури є рівень сприйняття, що охоплює фізичні пристрої, сенсори, контролери та вбудовані системи. На цьому рівні закладаються основи довіри, оскільки компрометація пристрою здатна викривити або підмінити дані ще до їх передачі в систему. Згідно з рекомендаціями NIST SP 800-183 “Networks of Things” [16], захист цього рівня реалізується через процедури безпечного завантаження (secure boot), перевірку цілісності прошивок, апаратне шифрування й автентифікацію мікропрограм. Такі заходи забезпечують достовірність і незмінність даних на етапі збору.

Мережевий рівень фокусується на захисті каналів зв’язку між пристроями, шлюзами та хмарними платформами. Його завдання полягає у гарантуванні конфіденційності, автентичності та цілісності переданих даних. Microsoft Digital Defense Report 2024 [13] вказує, що саме мережеві атаки є найпоширенішими серед IoT-загроз. Для їхнього запобігання використовуються захищені протоколи TLS і DTLS, системи двосторонньої

автентифікації та технології сегментації мереж (micro-segmentation), які обмежують поширення атак у разі компрометації одного вузла.

Наступною ланкою є шлюзовий рівень, який виступає проміжною ланкою між внутрішніми пристроями та зовнішніми системами. Саме тут реалізуються політики контролю доступу, перевірка автентичності підключень, фільтрація й валідація трафіку. Як зазначає ENISA Threat Landscape 2023 [11], шлюзи є критичними вузлами безпеки, що дозволяють запобігти поширенню шкідливого коду або неправдивих даних до основної системи обробки. Їх ефективність забезпечується завдяки локальному шифруванню, моніторингу трафіку та виявленню аномальної поведінки.

Хмарний рівень відповідає за обробку, зберігання й аналітику даних на віддалених сервісах. Тут ключовим пріоритетом є забезпечення безпеки інформації у спокої та під час передавання, багаторівневий контроль доступу, а також використання захищених API, що грунтуються на протоколах авторизації OAuth 2.0 та JWT. Важливими елементами даного рівня виступають аудит доступу, ротація криптографічних ключів і централізоване журналювання подій, що дозволяють забезпечити контроль за всіма діями в межах хмарного середовища [13].

Прикладний рівень охоплює програмні компоненти, сервіси та застосунки, які взаємодіють із пристроями IoT. Саме тут найчастіше виникають помилки, спричинені людським фактором або недотриманням принципів безпечного кодування. Як зазначалося раніше [11], інтеграція підходів DevSecOps у процес розробки дозволяє запобігати подібним проблемам шляхом постійного моніторингу вразливостей, автоматичного оновлення програмного забезпечення та ведення журналів подій. Дотримання цих принципів забезпечує стійкість системи навіть у динамічно змінюваному середовищі [13], [16].

Завершальним, проте не менш значущим, є рівень користувача. Людський фактор залишається визначальним у питаннях кіберстійкості: навіть найнадійніша технічна архітектура може бути скомпрометована через

помилки користувача. Як наголошує ENISA Cybersecurity Skills Framework [11], критично важливо формувати культуру безпечного користування пристроями, яка охоплює створення стійких паролів, регулярне оновлення мікропрограм, обмеження доступу до конфігураційних параметрів і звітування про підозрілі дії. Залучення користувачів як активних учасників процесу безпеки створює додатковий захисний рівень, що посилює загальну стійкість екосистеми.

Для наочності структура багаторівневої моделі безпеки, її основні цілі та засоби реалізації узагальнені в таблиці 1.3. Дана таблиця демонструє взаємозв'язок між рівнями системи, що разом формують цілісну архітектуру кіберзахисту IoT.

Таблиця 1.3 – Багаторівнева модель безпеки IoT та основні механізми захисту

Рівень	Основна мета	Ключові засоби захисту
Сприйняття	Захист фізичних пристроїв і мікропрограм	Secure Boot, TPM, перевірка цілісності прошивок
Мережевий	Безпечна передача даних і автентифікація	TLS/DTLS, шифрування, сегментація, контроль доступу
Шлюзовий	Фільтрація трафіку, автентифікація пристроїв	IDS/IPS, контроль доступу, валідація даних
Хмарний	Захист збережених і переданих даних	RBAC, шифрування, безпечні API, аудит доступу
Прикладний	Безпечне функціонування сервісів	DevSecOps, аудит коду, журналювання, оновлення
Користувача	Підвищення обізнаності та дотримання політик	Унікальні паролі, оновлення, зворотний зв'язок

Узагальнюючи, можна зазначити, що багаторівнева модель безпеки IoT утворює цілісну архітектуру, де кожен рівень - від сенсорів до користувача - виконує взаємодоповнювальні функції. Її ефективність полягає не лише у використанні сучасних технологій шифрування чи автентифікації, а

насамперед у гармонійному поєднанні технічних, організаційних і людських факторів. Як підсумовує ISO/IEC 30141:2018 [1], лише системний, багаторівневий підхід забезпечує довіру та стабільність у розподілених екосистемах Інтернету речей, що є передумовою їхнього безпечного розвитку в майбутньому.

1.5 Як відповідати вимогам безпеки IoT

У контексті стрімкого розвитку Інтернету речей питання відповідності вимогам безпеки набуває особливої актуальності. Кількість підключених пристроїв зростає експоненційно, а їх взаємодія стає дедалі складнішою, що створює сприятливе середовище для виникнення нових кіберзагроз. Традиційні підходи до захисту, орієнтовані на централізовані системи, не завжди здатні забезпечити належний рівень стійкості у розподіленому середовищі, де взаємодіють мільйони вузлів. Тому ефективна безпека IoT повинна базуватися на інтегрованій моделі, що поєднує три ключові складові: видимість, сегментацію та захист [16].

Початковим етапом побудови системи кіберзахисту є досягнення повної видимості мережі. Як зазначає NIST SP 800-183 “Networks of Things” [16], неможливо ефективно захищати те, що не підлягає спостереженню. Видимість передбачає повну інвентаризацію всіх підключених пристроїв із фіксацією їх технічних параметрів, програмних версій, моделей поведінки та напрямів передавання даних. Завдяки цьому формується детальна карта мережі, що дозволяє не лише виявляти невідомі вузли, але й оцінювати рівень ризику кожного елемента. Застосування методів поведінкового аналізу допомагає виявляти аномалії у роботі пристроїв, наприклад, різке зростання обсягу трафіку або нетипові з’єднання, які можуть свідчити про потенційне зараження. Досвід провідних компаній (Cisco, Palo Alto, Check Point) показує, що видимість є основою системи “Zero Trust”, оскільки забезпечує прозорість і контроль над усіма точками доступу [13].

На основі отриманої видимості реалізується сегментація, яка полягає у логічному розділенні мережевих елементів за рівнем ризику, функціональністю або критичністю. Як уже зазначалося раніше [11], саме ізоляція компонентів є ефективним інструментом мінімізації наслідків атаки. Якщо зловмиснику вдається отримати доступ до одного сегмента, він не може автоматично поширитися на інші частини інфраструктури. Пристрої, що працюють із критичними даними або мають підвищений рівень ризику, ізолюються від решти системи за допомогою VLAN, VPN чи спеціалізованих політик доступу. Такий підхід дає змогу локалізувати потенційні інциденти, спрощує моніторинг і полегшує аудит безпеки. Згідно з рекомендаціями Microsoft Digital Defense Report 2024 [13], багаторівнева сегментація суттєво знижує ймовірність бічних атак (lateral movement) у середовищах IoT, що є одним із найнебезпечніших векторів компрометації систем.

Завершальним етапом виступає активний захист, який передбачає застосування політик безпеки, механізмів реагування та постійного моніторингу. Йдеться не лише про фіксацію подій, а про безперервний аналіз мережевого трафіку в реальному часі. Використання систем виявлення вторгнень (IDS/IPS), механізмів багаторівневої автентифікації, контроль доступу на основі ролей (RBAC) і поведінкових моделей дозволяє забезпечити гнучкий, самонавчальний захист. Як показують результати ENISA Threat Landscape 2023 [11], організації, які впроваджують адаптивні системи моніторингу IoT, скорочують середній час виявлення інцидентів більш ніж удвічі. Крім того, застосування концепції Zero Trust [14] та принципів security by design забезпечує стійкість архітектури навіть у випадку наявності вразливих елементів або людського фактора.

Для кращого розуміння взаємодії трьох базових компонентів – видимості, сегментації та захисту – доцільно узагальнити їхні ключові характеристики у таблиці 1.4. Дана таблиця відображає логіку побудови сучасної системи безпеки IoT відповідно до вимог міжнародних стандартів.

Таблиця 1.4 – Основні складові підходу до забезпечення безпеки IoT

Компонент	Основна мета	Реалізаційні механізми	Очікуваний результат
Видимість	Повна інвентаризація та моніторинг усіх пристроїв IoT	Ідентифікація, поведінковий аналіз, виявлення аномалій, формування карти мережі	Забезпечення прозорості інфраструктури та раннє виявлення ризиків
Сегментація	Локалізація потенційних інцидентів і обмеження поширення атак	VLAN/VPN, Zero Trust, контроль доступу, мікросегментація	Підвищення стійкості до бічних атак, зменшення площини ризику
Захист	Реагування на інциденти, запобігання загрозам	IDS/IPS, шифрування, багаторівнева автентифікація, аналіз трафіку в реальному часі	Адаптивна система безпеки, здатна самостійно виявляти та нейтралізувати загрози

У підсумку, ефективне забезпечення безпеки IoT полягає не у впровадженні окремих технологій, а у побудові динамічної системи, де кожен елемент виконує взаємодоповнювальні функції. Видимість забезпечує прозорість і контроль, сегментація - ізоляцію та керованість, а захист - активне реагування на ризики. Їх синергія створює стійке середовище, здатне протидіяти сучасним кіберзагрозам і відповідати міжнародним вимогам безпеки, що є необхідною умовою стабільного розвитку екосистеми Інтернету речей.

2 РОЗУМІННЯ ЛАНДШАФТУ БЕЗПЕКИ ІНТЕРНЕТУ РЕЧЕЙ

2.1 Огляд стандартів безпеки для IoT

Розвиток Інтернету речей як ключового елементу цифрової трансформації вимагає нового підходу до формування політик і стандартів безпеки. З огляду на величезну кількість пристроїв, різноманіття платформ і відсутність уніфікованої архітектури, саме стандартизація стає базовим інструментом для встановлення спільних правил захисту. Без чітко визначених вимог і технічних орієнтирів забезпечити стійкість та взаємодію IoT-рішень неможливо. Саме тому протягом останнього десятиліття міжнародні організації - ISO, IEC, ETSI, IEEE, NIST та інші - розробили низку нормативних документів, що визначають мінімальні й рекомендовані вимоги до безпеки IoT.

Одним із базових документів є ISO/IEC 30141:2018 “Internet of Things – Reference Architecture” [1], який задає рамкову структуру побудови IoT-систем. У ньому безпека розглядається як наскрізний процес, що охоплює всі етапи життєвого циклу пристрою - від проєктування (принцип Security by Design) до виведення з експлуатації. Стандарт визначає концепцію “довірчих зон” (trust zones), моделює потенційні загрози на кожному рівні взаємодії та пропонує архітектурні рішення для мінімізації ризиків. На практиці цей документ використовується під час проєктування промислових і критично важливих систем, де необхідно гарантувати передбачувану поведінку всіх компонентів.

У споживчому сегменті важливу роль відіграє стандарт ETSI EN 303 645 [14], розроблений Європейським інститутом телекомунікаційних стандартів. Він містить 13 практичних вимог, серед яких: заборона використання фіксованих (типових) паролів, обов’язковість механізмів оновлення ПЗ, обмеження доступу до діагностичних портів, журналювання подій і захист персональних даних користувачів. ETSI EN 303 645 став

основою для британського закону Product Security and Telecommunications Infrastructure Act, що встановлює юридичну відповідальність виробників IoT-пристроїв за невідповідність базовим вимогам безпеки. Цей документ демонструє перехід від рекомендацій до обов'язкових регуляторних практик.

Своє бачення стандартів формує і Національний інститут стандартів і технологій США (NIST), який запропонував серію документів NIST IR 8259, 8259A, SP 800-213 та SP 800-183 [16]. Вони визначають базові можливості безпеки пристроїв IoT (IoT Device Cybersecurity Capability Core Baseline), включно з автентифікацією, управлінням конфігураціями, контролем оновлень, веденням журналів та підтримкою довірчих середовищ. Документ SP 800-213 особливо важливий, оскільки встановлює вимоги до інтеграції IoT у державну IT-інфраструктуру США. Він враховує обмежені ресурси пристроїв, різні канали комунікації та ролі користувачів, забезпечуючи практичну гнучкість у впровадженні кіберзахисту.

Для промислових систем стратегічне значення має серія стандартів IEC 62443, яка регламентує безпеку автоматизованих систем керування (IACS). Цей набір документів охоплює як технічні, так і організаційні заходи, описує моделі управління ризиками, визначає рівні зрілості безпеки та критерії сертифікації постачальників. Саме завдяки IEC 62443 концепції безпечної взаємодії між промисловими мережами стали основою сучасних підходів до "smart manufacturing" і промислового Інтернету речей (IIoT).

Не менш важливим є внесок організації IEEE, зокрема стандарт IEEE 802.15.4, який визначає основи бездротових сенсорних мереж і використовується в протоколах Zigbee, Thread і WirelessHART. Цей стандарт описує методи шифрування, контролю доступу та управління енергоспоживанням у малопотужних мережах. Паралельно IETF розробляє набір комунікаційних протоколів, таких як DTLS, CoAP, 6LoWPAN, які забезпечують безпечну передачу даних у середовищах з обмеженими ресурсами [18]. Таким чином, міжнародна співпраця між різними

організаціями сприяє створенню узгодженої технічної екосистеми, у якій безпека є вбудованим елементом архітектури.

Для наочного узагальнення основних міжнародних стандартів і їх розподілу за організаціями наведено таблицю 2.1. Дана таблиця демонструє систематизацію нормативних документів залежно від сфери застосування, рівня деталізації та ключових вимог до безпеки.

Таблиця 2.1 – Основні міжнародні стандарти безпеки IoT

Організація	Стандарт / документ	Основна спрямованість	Ключові вимоги	Застосування
ISO / IEC	ISO/IEC 30141:2018	Референтна архітектура IoT	Security by Design, trust zones, аналіз поверхні атак	Промислові та критичні системи
ETSI	ETSI EN 303 645	Споживчі IoT-пристрої	Заборона фіксованих паролів, оновлення ПЗ, захист даних	Масовий ринок, побутові пристрої
NIST (США)	NIST IR 8259, SP 800-213, SP 800-183	Базові можливості кіберзахисту	Автентифікація, журналювання, конфігурація, оновлення	Урядові системи, корпоративні мережі
IEC	IEC 62443 (серія)	Промислова автоматизація	Управління ризиками, рівні зрілості безпеки	Промислові системи, енергетика
IEEE / IETF	IEEE 802.15.4, DTLS, CoAP, 6LoWPAN	Комунікаційні протоколи IoT	Шифрування, енергоефективність, захист передачі	Бездротові мережі, сенсорні системи

Попри наявність широкої нормативної бази, ступінь впровадження стандартів залишається обмеженим. Як показують дослідження HP Fortify та ENISA, понад 70 % споживчих IoT-пристроїв не використовують шифрування даних, а більш ніж половина мають слабкі або незмінні паролі. Головними причинами є прагнення зменшити витрати на виробництво,

відсутність обов’язкових сертифікаційних вимог поза межами ЄС і низький рівень обізнаності користувачів. У результаті навіть сертифіковані рішення часто взаємодіють із небезпечними пристроями, що не відповідають базовим вимогам. Саме тому формування єдиного, міжгалузевого і багаторівневого стандарту безпеки IoT є не лише технічною, а й стратегічною потребою для розвитку надійної цифрової інфраструктури.

2.2 Аналіз архітектур захисту в існуючих IoT-системах

Архітектура є фундаментом, який визначає рівень стійкості IoT-системи до зовнішніх і внутрішніх загроз. Вона формує логіку взаємодії між пристроями, обробки даних, надання доступу й застосування механізмів захисту. У середовищі, де пристрої виробляються різними постачальниками, працюють у неоднорідних умовах і використовують різні протоколи, саме архітектурні рішення визначають, наскільки система здатна витримувати сучасні кібератаки.

У більшості актуальних IoT-рішень застосовується трикомпонентна архітектурна модель, яка поєднує захист на рівні пристроїв, безпечну мережеву взаємодію та захист інфраструктури зберігання й обробки даних. У переважній кількості систем основне навантаження лягає на хмарні платформи, де зберігається та аналізується більша частина даних. Такі моделі широко використовуються у великих хмарних екосистемах, де автентифікація реалізується за допомогою сертифікатів X.509, а передавання даних захищене за допомогою TLS-шифрування. Підходи до контролю доступу будуються на основі політик, які визначають, які дії дозволені конкретному пристрою або сервісу, що повністю відповідає рекомендаціям NIST SP 800-213 щодо захисту IoT-пристроїв у федеральних мережах [5].

Разом із тим централізованість створює низку нових ризиків. Компрометація хмарного брокера, викрадення ключів доступу або збій у хмарній інфраструктурі можуть спричинити каскадну відмову великої кількості пристроїв одночасно. До цього додаються ризики приватності даних, які особливо критичні в медичних, транспортних і побутових системах. Як зазначається у звітах OECD та WEF [9,10], саме централізована обробка персональних даних часто стає слабкою ланкою в ланцюгу IoT-безпеки.

У відповідь на ці виклики дедалі більшого поширення набувають архітектури на основі Fog computing. На відміну від хмарної моделі, частина функцій - зокрема фільтрація трафіку, локальна аналітика, контроль доступу, первинне виявлення аномалій - виконується на периферійних вузлах. Це знижує затримки, зменшує залежність від інтернет-з’єднання та дозволяє оперативно реагувати на локальні загрози. ENISA у своєму звіті 2023 року [11] підкреслює, що локальна обробка даних значно підвищує стійкість системи до масових атак.

Щоб відобразити ключові відмінності між архітектурними підходами, у таблиці 2.2 наведено порівняння хмарної та edge/fog-моделей.

Таблиця 2.2 – Порівняльна характеристика хмарної та Edge/Fog-архітектури

Критерій	Хмарна модель (Cloud)	Edge/Fog модель
Обробка даних	У центрі (хмара)	На периферії
Час відгуку	Вищий	Нижчий
Залежність від інтернету	Висока	Може працювати автономно
Безпека даних	Зосереджена в хмарі	Частково локалізована
Масштабованість	Дуже висока	Обмежена ресурсами
Витрати на передачу	Вищі	Нижчі
Типові сценарії	Камери, розумні колонки	Медицина, транспорт, промисловість

Попри беззаперечні переваги edge-рішень, їх впровадження супроводжується значною кількістю технічних бар'єрів. Висока розподіленість ускладнює централізоване адміністрування, а більшість малопотужних пристроїв не здатні виконувати локальну аналітику або обробку великих обсягів даних. Також виникають проблеми синхронізації політик безпеки - у великій розподіленій інфраструктурі складно забезпечити узгодженість конфігурацій, оновлень і контрольних механізмів.

Саме тому в промисловості та критичних секторах найчастіше застосовують гібридні архітектури, де edge-вузли використовуються для швидкої локальної обробки та первинного захисту, а хмара - для довгострокового зберігання, аналітики й централізованого нагляду. Такі моделі добре узгоджуються зі стандартами IEC 62443 та рекомендаціями ISO/IEC 30141 [1].

Проте незалежно від обраної моделі, аналіз практичних впроваджень IoT показує значну кількість системних недоліків. За даними звітів Fortinet, Rapid7 і Gartner, понад половина IoT-рішень мають критичні архітектурні помилки: від використання застарілих алгоритмів шифрування до оновлень без цифрового підпису. Деякі з типових вразливостей наведено в таблиці 2.3.

Таблиця 2.3 – Поширені помилки в архітектурі захисту IoT

Категорія	Приклади порушень	Наслідки
Автентифікація	Жорстко закодовані паролі, відсутність токенів	Неавторизований доступ
Шифрування	Використання AES-ECB, передача даних у відкритому вигляді	Перехоплення й підміна даних
Оновлення ПЗ	Відсутність цифрового підпису, оновлення через HTTP	Ін'єкція шкідливого ПЗ
Журналювання	Відсутність логів або логування лише критичних подій	Неможливість аналізу інцидентів
Залежність від	Єдина точка відмови у хмарній моделі	Масовий збій або

провайдера		компрометація

Як зазначає Gartner, понад 60% IoT-систем містять вразливості, які не можуть бути усунені звичайними оновленнями - лише шляхом модифікації або повного перепроєктування архітектури. Це яскраво демонструє, що саме архітектурні рішення закладають межі можливостей безпеки.

Підхід security by design, на який орієнтуються ISO та NIST [1,5], є не рекомендацією, а необхідністю: без вбудованих механізмів захисту на рівні архітектури система неминуче матиме структурні вразливості. Те саме стосується й контролю доступу - у наступному підпункті буде розглянуто, як саме модель контролю доступу впливає на стійкість усього IoT-рішення та чому саме вона визначає, чи зможе система ефективно протистояти загрозам навіть у разі часткової компрометації окремих компонентів [3].

2.3 Fog Computing як рішення для IoT-систем

Fog Computing розглядається як одна з найперспективніших парадигм обробки даних для сучасних IoT-систем, що характеризуються високою динамічністю, географічною розподіленістю та зростаючими вимогами до швидкодії й надійності. На відміну від класичної хмарної моделі, де обчислення відбувається у централізованих дата-центрах, fog-архітектура передбачає перенесення значної частини обчислювальних процесів ближче до периферії мережі. В основі Fog Computing лежить вісім фундаментальних принципів, які детермінують її архітектуру, функціональні властивості та можливості інтеграції в інтелектуальні IoT-інфраструктури (рисунок 1.2). Саме їх глибоке розуміння дозволяє коректно проєктувати fog-системи та забезпечувати відповідність вимогам продуктивності, масштабованості та безпеки [22].

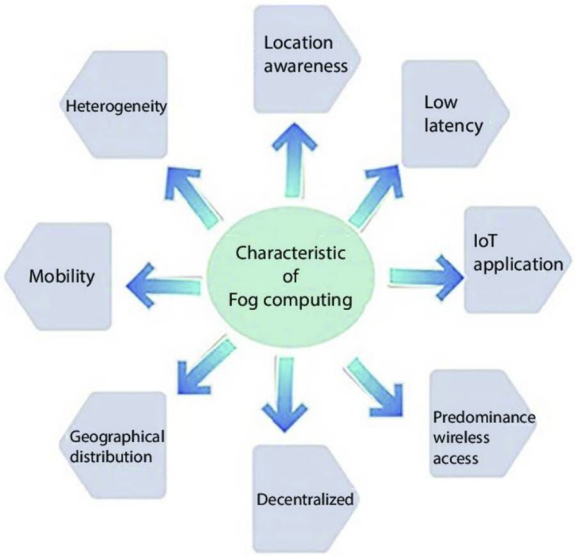

Рисунок 2.1 - Принципи Fog Computing [24]

Одним із ключових принципів є мінімізація затримки та забезпечення локальності обробки даних. Розміщення fog-вузлів поблизу джерел даних істотно скорочує час між отриманням, аналізом і реагуванням, що є критично важливим для сценаріїв, де навіть мілісекундна затримка може спричинити небажані наслідки. До таких сценаріїв належать автономний транспорт, автоматизовані лінії промислової робототехніки, медичні системи моніторингу та інші застосування, орієнтовані на режим реального часу. Локальність обробки не лише зменшує часові затримки, а й дозволяє знизити навантаження на канали зв’язку та централізовані сервери, підвищуючи загальну стійкість IoT-інфраструктури.

Важливою властивістю fog-парадигми є її географічна розподіленість. Fog-вузли можуть розгортатися у міських інфраструктурах, промислових зонах, транспортних системах, а також у віддалених регіонах, що не мають доступу до високошвидкісних каналів зв’язку. Така розподіленість забезпечує доступність сервісів незалежно від місця розташування користувача або пристрою та дозволяє адаптувати рівень обчислень до конкретного середовища. Саме географічна гнучкість дає змогу fog-моделі

обслуговувати гетерогенний трафік і працювати з різнорідними групами пристроїв, що характерно для більшості IoT-систем.

Принцип підтримки великої кількості fog-вузлів відображає масштаб IoT-екосистеми, у якій можуть функціонувати мільйони або навіть мільярди пристроїв. На відміну від централізованих дата-центрів, де обчислення зосереджені у відносно невеликій кількості серверів, fog-архітектура передбачає розподілення обчислювальних можливостей серед величезної кількості периферійних вузлів. Це породжує нові завдання щодо управління, координації та уніфікації стандартів, водночас відкриваючи шлях до безпрецедентної масштабованості та відмовостійкості.

Не менш важливим є принцип підтримки мобільності, що дозволяє fog-системам обслуговувати мобільних користувачів та пристрої, які переміщуються між різними сегментами мережі. Для цього необхідні алгоритми міграції сервісів, синхронізації станів і динамічного перенаправлення трафіку. Саме здатність підтримувати безперервність роботи під час переміщення забезпечує fog-моделі перевагу над традиційною хмарою, де мобільність часто компенсується збільшеною затримкою доступу.

Ще одним фундаментальним аспектом fog-архітектури є робота в умовах реального часу. Це передбачає можливість обробляти безперервні потоки даних з великої кількості джерел, здійснювати кореляцію подій і формувати реакцію з мінімальним часом затримки. IoT-пристрої генерують дані нерівномірно, а деякі системи (наприклад, сенсорні мережі моніторингу стану обладнання) формують десятки тисяч вимірювань за секунду. Fog-вузли здатні локально фільтрувати, агрегувати або передавати далі лише релевантні дані, що значно знижує навантаження на мережеву інфраструктуру.

Принцип взаємодії з різнорідними мережами забезпечує сумісність fog-систем із широким спектром технологій зв’язку: дротовими, оптоволоконними, мобільними, бездротовими низькопотужними (LPWAN)

та високошвидкісними радіотехнологіями. Це вимагає адаптивних протоколів маршрутизації, гнучких механізмів контролю трафіку та підтримки мереж із різними рівнями пропускної здатності або стабільності. Завдяки цьому fog-платформа здатна працювати у складних умовах мережевої фрагментації, що є типовим для IoT-інфраструктур великої кількості пристроїв.

Принцип федеративності та ієрархічності дозволяє організувати fog-систему у вигляді багаторівневої структури, де нижчі вузли виконують локальні завдання, а складніші - передаються на вищі рівні або у хмару. Такий підхід забезпечує керованість та ефективне балансування навантаження між вузлами різної потужності. Федеративність дає змогу створювати автономні домени, що мають власні політики безпеки та управління, але водночас можуть взаємодіяти між собою у ширшому контексті загальної IoT-екосистеми.

Останнім фундаментальним принципом є програмованість, що передбачає наявність гнучких API, інструментів розробки та механізмів автоматизації, які дозволяють створювати, розгортати та масштабувати сервіси на fog-вузлах. Підтримка різних мов програмування, методологій DevOps, контейнеризації (Docker, Podman) та оркестрації (Kubernetes, OpenShift) забезпечує широкі можливості для інтеграції Fog Computing у сучасні цифрові системи. Програмованість робить fog-архітектуру придатною не лише для класичних обчислень, а й для реалізації інтелектуальних сервісів, зокрема машинного навчання, локальних систем виявлення аномалій та адаптивних механізмів реагування.

У сукупності ці вісім принципів формують концептуальну основу Fog Computing як архітектури, що найбільше відповідає сучасним викликам IoT-екосистем. Вони забезпечують низькі затримки, масштабованість, адаптивність, підтримку мобільності та можливість інтеграції інтелектуальних моделей безпеки - саме тих властивостей, які є критично необхідними для побудови стійких, ефективних та безпечних IoT-рішень.

2.4 Типові уразливості пристроїв IoT

Підвищення загального рівня безпеки систем Інтернету речей неможливе без уважного аналізу найуразливішої їх ланки - кінцевих пристроїв. Навіть за умови належного захисту мережевої інфраструктури та хмарних сервісів, компрометація окремого пристрою може відкрити зловмиснику доступ до конфіденційних даних, внутрішнього трафіку або навіть до фізичних об’єктів керування (камер спостереження, контролерів доступу, промислових датчиків тощо). З огляду на обмежені ресурси, орієнтацію виробників на зниження вартості та швидкий вихід на ринок, безпека часто сприймається як другорядний аспект, що призводить до масового повторення одних і тих самих вразливостей у різних класах пристроїв.

Звіти Palo Alto Networks Unit 42 “IoT Threat Report” [3], ENISA Threat Landscape 2023 [11] та Cyber Security Report 2023 [12] послідовно показують, що значна частина виявлених інцидентів безпосередньо пов’язана з базовими помилками у проєктуванні й налаштуванні IoT-пристроїв. До найпоширеніших належать слабка або відсутня автентифікація, небезпечні механізми оновлення, відкриті мережеві порти, використання протоколів без шифрування, уразливі веб-інтерфейси та відсутність журналювання подій.

Однією з найтиповіших проблем є використання слабких, типовий або жорстко закодованих облікових даних. Значна частина пристроїв постачається із стандартними паролями, які користувачі не змінюють, або, що ще гірше, - із вбудованими обліковими записами, які неможливо коректно відключити. ENISA у своїх оглядах [11] наголошує, що саме вразливості типу default credentials залишаються ключовим чинником масового зламу

споживчих IoT-пристроїв, зокрема відеокамер, домашніх маршрутизаторів і мультимедійних пристроїв. Саме на усунення цієї проблеми спрямовані вимоги стандарту ETSI EN 303 645 [4], який прямо забороняє використання незмінюваних універсальних паролів.

Не менш критичною є відсутність безпечних механізмів оновлення програмного забезпечення. Частина пристроїв узагалі не підтримує віддалене оновлення, інші завантажують оновлення без перевірки цифрового підпису або цілісності. Це відкриває шлях до атак на ланцюг постачання, коли шкідливе оновлення може бути встановлене замість легітимного. NIST у своїх рекомендаціях для IoT-пристроїв [5], [16] підкреслює, що можливість безпечного оновлення - одна з базових вимог до сучасних рішень, оскільки без неї фактично неможливо усувати відомі вразливості в польових умовах.

Окрему категорію становлять відкриті або слабо захищені мережеві порти. Розгорнуті у глобальній мережі пристрої часто мають доступні служби Telnet, HTTP, MQTT або CoAP без шифрування чи з мінімальним контролем доступу. Звіти з кіберзагроз [3], [11], [12], [13] показують, що такі пристрої активно відстежуються спеціалізованими сканерами, а згодом використовуються як “вхідні точки” для формування ботнетів або проведення масових DDoS-атак. Класичним прикладом наслідків цієї проблеми став ботнет Mirai, який, як неодноразово відзначалося в аналітичних оглядах, уразив десятки тисяч пристроїв з відкритими небезпечними службами.

Суттєвий вплив на загальну стійкість мають також уразливі веб-інтерфейси й API. Неправильна обробка вхідних даних, відсутність валідації, використання застарілих бібліотек часто призводять до ін’єкційних вразливостей (командні ін’єкції, XSS, SQL/код-ін’єкції), що дозволяє зловмиснику виконувати небажані дії на пристрої. ENISA [11] відносить такі уразливості до пріоритетних, оскільки вони можуть забезпечити віддалене виконання коду (RCE) та повний контроль над пристроєм.

Додатковим фактором ризику є використання незахищених протоколів і відсутність наскрізного шифрування. У своїх аналітичних матеріалах Microsoft [13] звертає увагу, що значна частина трафіку в сегменті IoT усе ще передається по HTTP або через небезпечно налаштовані MQTT-брокери, що дозволяє здійснювати атаки типу man-in-the-middle, підслуховувати або модифікувати дані. Це суперечить як вимогам ETSI EN 303 645 [4], так і рекомендаціям NIST [5].

Для систематизації типових технічних проблем у таблиці 2.4 узагальнено основні категорії вразливостей, приклади їх прояву та потенційні наслідки.

Таблиця 2.4 – Поширені вразливості IoT-пристроїв та їх наслідки

Категорія вразливості	Приклади	Потенційні наслідки
Слабка автентифікація	Типові логіни й паролі, жорстко закодовані облікові дані, відсутність авторизації для локального доступу	Повний несанкціонований контроль над пристроєм, доступ до конфіденційних даних
Відсутність безпечного оновлення	Оновлення без цифрового підпису, відсутність перевірки цілісності, неможливість оновлення “по повітрю”	Впровадження шкідливого ПЗ, постійна присутність відомих вразливостей
Відкриті мережеві порти	Публічний доступ до Telnet/HTTP/MQTT без шифрування та автентифікації	Підключення до ботнетів, перехоплення трафіку, несанкціоноване керування
Уразливі веб-інтерфейси	Командні ін’єкції, XSS, відсутність валідації параметрів у панелях керування	Віддалене виконання коду, змінення конфігурацій, видалення чи крадіжка даних
Небезпечні протоколи	Використання HTTP замість HTTPS, небезпечно налаштований MQTT/CoAP	Підслуховування й підміна трафіку, сесійні атаки
Відсутність логування	Відсутність журналів або запис лише критичних помилок	Неможливість розслідування інцидентів, складнощі з виявленням атак

Як випливає з аналітичних звітів [3], [11], [12], [13], найбільш критичні наслідки породжує комбінація декількох вразливостей, наприклад: відсутність автентифікації, відкриті порти та небезпечні механізми оновлення. Уражені пристрої часто використовуються як частина розподіленої інфраструктури атак - зокрема для DDoS, сканування інших сегментів мережі, розповсюдження шкідливого ПЗ або прихованого майнінгу.

Окремо варто відзначити, що проблеми безпеки на рівні пристрою часто не усуваються навіть після їх виявлення. Багато виробників не підтримують випуск оновлень протягом усього життєвого циклу продукту або вимагають складних процедур вручну, що суперечить принципам, рекомендованим NIST та ETSI [4], [5]. У таких випадках єдиними практичними заходами залишаються сегментація мережі, ізоляція вразливих пристроїв і використання додаткових захисних шлюзів.

Оцінка вразливостей IoT-пристроїв має здійснюватися системно, із використанням автоматизованих сканерів, тестуванням на проникнення, моделюванням загроз і періодичним переглядом політик доступу. Звіти провідних організацій [3], [11], [12], [13] підкреслюють, що саме регулярна переоцінка ризиків та оновлення механізмів захисту дають змогу мінімізувати наслідки неминучих помилок у дизайні й реалізації пристроїв. У наступних підрозділах ці результати будуть використані як основа для побудови моделей контролю доступу та механізмів захисту, здатних локалізувати інциденти навіть у разі компрометації окремих IoT-вузлів.

2.5 Методи виявлення та запобігання загрозам в IoT-середовищі

Забезпечення ефективного захисту Інтернету речей значною мірою залежить від здатності своєчасно виявляти аномальну активність і запобігати подальшому розвитку кібератак. На відміну від традиційних IT-систем, IoT характеризується великою кількістю різнорідних пристроїв, географічною розподіленістю та непередбачуваністю поведінки. Це робить класичні методи контролю недостатньо ефективними, а отже - вимагає спеціалізованих підходів, орієнтованих на поведінкову аналітику, аналіз пристроїв і багаторівневу ізоляцію загроз. Як підкреслює ENISA у звіті Threat Landscape 2023 [11], своєчасне виявлення атак у IoT-середовищі має визначальне значення, оскільки багато інцидентів набувають критичного масштабу протягом хвилин.

Першим і найбільш універсальним методом виступає аналіз аномалій у поведінці пристроїв. На відміну від сигнатурних систем, що реагують лише на відомі типи атак, поведінковий аналіз дозволяє виявляти нетипові дії: різке збільшення обсягу трафіку, незвичні інтервали передачі даних, зміну портів або спроби комунікації з нетиповими мережевими вузлами. Така логіка широко відображена в практичних рекомендаціях Microsoft Digital Defense Report 2024 [13], де підкреслюється, що для IoT характерний саме контекстуальний моніторинг, а не перевірка сигнатур. Якщо, наприклад, побутовий датчик світла починає раптово надсилати високочастотні запити через незвичні мережеві порти - це вказує на можливу компрометацію або появу стороннього процесу. Типову схему роботи таких систем - від збору мережевих даних до автоматизованого реагування - наведено на рисунку 2.4.

Паралельно до поведінкових методів застосовується сигнатурне виявлення, яке базується на порівнянні активності пристрою з відомими шаблонами атак. Цей підхід, типовий для IDS/IPS-систем, залишається ефективним у ситуаціях, коли загроза має відомий цифровий слід. Однак звіти Palo Alto Networks Unit 42 [3] і ENISA [11] зазначають, що в IoT-секторі доля нових, раніше невідомих атак невпинно зростає, що робить суто

сигнатурний підхід недостатнім. Тому системи виявлення часто поєднують сигнатурні й поведінкові алгоритми.

Окремим напрямом виявлення вразливостей є перевірка пристроїв на відповідність стандартам безпеки. Документи ETSI EN 303 645 [4] та NIST SP 800-213 [5] містять вимоги до захисту оновлень, автентифікації, конфігурацій та мережевих інтерфейсів, а отже можуть бути використані як перелік контрольних критеріїв. Перед розгортанням у продуктивному середовищі IoT-пристрої перевіряють через автоматизовані інструменти, що тестують наявність відкритих портів, підтримку шифрування, коректність сертифікатів тощо. У звіті Cyber Security Report 2023 [12] зазначено, що майже 70 % пристроїв, перевірених за мінімальними вимогами, не відповідали хоча б одному пункту безпеки.

У великих інфраструктурах усе ширше застосовується інтеграція з SIEM-платформами, які збирають журнали подій, аналізують мережеву активність та дають змогу централізовано реагувати на інциденти. Microsoft [13] наголошує, що IoT-пристрої часто генерують неповні або нестандартні логи, тому важливо забезпечити їх уніфікацію та формат, сумісний із системами моніторингу. Централізоване логування дозволяє виявляти тенденції, відстежувати ланцюги атак і швидко реагувати на компрометацію.

Не менш важливою складовою є проактивне запобігання загрозам, що включає сегментацію мережі, застосування Zero Trust-підходу та ізоляцію пристроїв за принципом «найменших привілеїв». Як зазначалося раніше у дослідженнях ETSI і NIST [4], [5], саме архітектурна ізоляція мінімізує можливість бічного переміщення зловмисника між вузлами. Стандартні заходи включають використання окремих VLAN для IoT, обмеження зон трансляції, фільтрацію трафіку на рівні шлюзів, блокування невідомих служб та закриття портів за замовчуванням. У критичних системах може застосовуватися навіть фізичне або логічне «розривання» мережі (air-gapping), що унеможливлює несанкціонований доступ іззовні.

У таблиці 2.5 узагальнено ключові методи виявлення та запобігання загрозам в IoT-середовищі.

Забезпечення стійкості IoT-систем до загроз вимагає поєднання реактивних і превентивних методів. Поведінковий аналіз дає змогу виявляти підозрілу активність на ранніх етапах, тоді як сегментація та обмеження доступу мінімізують наслідки навіть успішної атаки. Водночас, як підкреслюється у звітах ENISA [11] та Microsoft [13], ефективність таких методів значною мірою залежить від здатності пристроїв генерувати коректні журнали подій, підтримувати безпечні протоколи та забезпечувати цілісність власної прошивки.

Таблиця 2.5 – Методи виявлення та запобігання загрозам в IoT

Метод	Переваги	Обмеження	Приклади застосування
Аналіз аномалій	Виявлення нових та нестандартних атак	Хибнопозитивні спрацювання	Поведінковий моніторинг, мережеві NBAD-системи
Сигнатурне виявлення	Висока точність для відомих загроз	Безсилля проти нових атак	IDS/IPS з оновлюваними базами сигнатур
Перевірка на відповідність	Раннє знаходження конфігураційних помилок	Не виявляє поведінкові порушення	Тестування за вимогами ETSI/NIST
SIEM-інтеграція	Централізований контроль та реагування	Потребує ресурсів і стандартизації логів	Splunk, Microsoft Sentinel, QRadar
Мережна ізоляція	Обмежує поширення атак	Ускладнює комунікацію між вузлами	VLAN-сегментація, шлюзи доступу

Саме тому реалізація механізмів самодіагностики, перевірки цілісності та регулярного оновлення стає ключовим елементом у сучасних IoT-середовищах.

У міру ускладнення екосистем IoT особливої уваги потребує автоматизація реагування, що стане предметом аналізу в наступному підпункті, присвяченому управлінню інцидентами та ролі штучного інтелекту в підвищенні ефективності захисту.

2.6 Реагування на інциденти безпеки в IoT-середовищі

Після виявлення загрози в IoT-середовищі найважливішим етапом стає оперативне реагування. На відміну від традиційних IT-систем, де адміністратори мають доступ до повнофункціональних серверів і вузлів, IoT значно складніше для ліквідації інцидентів через обмеження енергії, пам’яті, відсутність журналів, різноманітність платформ і велику кількість вузлів, розподілених у просторі. Як підкреслює NIST у SP 800-183 [16], навіть незначні затримки у реагуванні можуть призвести до ланцюгової компрометації десятків або сотень пристроїв, особливо коли мова йде про сенсорні мережі, “розумні” камери або вузли промислової автоматизації.

Першим і найбільш критичним кроком є ізоляція потенційно скомпрометованого пристрою. Це дозволяє перервати подальше поширення атаки або її вплив на інші елементи інфраструктури. Ізоляція може здійснюватися на рівні мережевого шлюзу, через блокування MAC-адреси, вимкнення інтерфейсу, переведення вузла у окрему VLAN або повне фізичне відключення. Оскільки значна частина пристроїв IoT не має власних журналів подій або формує їх у скороченому вигляді, важливу роль виконують зовнішні елементи - шлюзи, проксі та SIEM-системи, які накопичують лог-дані для подальшого аналізу. Збереження цієї інформації є

ключовим, адже дозволяє встановити як першопричину, так і шлях проникнення атаки.

Після ізоляції переходять до аналізу інциденту, який включає ретроспективу подій, визначення вразливості, що була використана, та оцінку можливих масштабів поширення. Згідно з рекомендаціями NIST SP 800-213 [5], у цьому процесі важливо враховувати не лише технічні параметри (відкриті порти, журнали аномалій, ненадійні конфігурації), а й поведінкові аспекти - раптове збільшення трафіку, незвичні запити до хмарних сервісів, зміну інтервалів передачі даних. Такий підхід дозволяє не лише усунути наслідки інциденту, а й виключити подібні сценарії у майбутньому, оновивши політики доступу, змінюючи ключі автентифікації або блокуючи небезпечні протоколи.

У випадку масових інцидентів, які поширюються автоматично (наприклад, різновиди атак на зразок Mirai, що активно описані в ENISA Threat Landscape [11]), традиційні ручні методи реагування стають недостатніми. У таких ситуаціях необхідним елементом захисту є автоматизація реакції, заснована на SOAR-системах, здатних виконувати попередньо визначені дії при виявленні певного тригера. Це може включати блокування IP-адреси, закриття сесії, перезапуск сервісу, переведення пристрою у режим обмеженої функціональності або генерацію повідомлення відповідальним особам. У великих інфраструктурах така автоматизація значно скорочує час від моменту виявлення аномалії до повної локалізації загрози.

Важливим аспектом є те, що не всі IoT-пристрої підтримують повноцінні механізми дистанційного адміністрування, оновлення або ізоляції. Частина бюджетних пристроїв не має інструментів безпечного OTA-оновлення, не підтримує цифрові підписи, а деякі моделі взагалі не дозволяють оновлювати мікропрограму після виробництва. У таких випадках мережевий рівень - шлюзи, SDN-рішення, фаєрволи - стає основною точкою впливу під час інцидент-менеджменту. Це узгоджується з висновками

Microsoft Digital Defense Report 2024 [13], де зазначено, що саме мережеві політики є критично важливими для стримування атак у розподілених середовищах.

Водночас реагування на інциденти не може обмежуватися технічними інструментами. Значну роль відіграє операційна готовність команди, зокрема наявність описаних процедур (playbooks), визначені ролі відповідальності, сценарії дій для різних типів пристроїв. Як підкреслює ENISA [11], підготовленість персоналу часто має вирішальний вплив на швидкість ізоляції інциденту. Типові стратегії реагування систематизовані в таблиці 2.6.

Таблиця 2.6 – Методи реагування на інциденти в IoT

Метод	Опис дій	Приклади інструментів / практик
Ізоляція пристрою	Відключення, зміна VLAN, блокування MAC/IP	NAC-системи, IoT-gateway firewall
Аналіз інциденту	Ретроспектива подій, визначення вектора атаки	SIEM-платформи, аналіз логів
Автоматизована реакція	Виконання playbook-скриптів при тригерах	SOAR-рішення, Azure Sentinel
Перевстановлення / відкат	Відновлення стабільної версії ПЗ	OTA-оновлення, резервні образи
Посилення політик	Перегляд ACL, вимкнення портів, microsegmentation	Zero Trust-модель, мережеві ACL

Для узагальнення послідовності дій на таблиця 2.6 представлено типовий цикл реагування в IoT-інфраструктурі - від виявлення інциденту до впровадження оновлених політик і навчання персоналу.

Такий циклічний підхід дозволяє організаціям адаптуватися до зростаючої складності загроз та забезпечувати стабільність систем навіть у випадку повторних атак.

Зрештою, реагування в IoT - це не разова дія, а частина безперервного життєвого циклу безпеки, який охоплює аналіз поведінки пристроїв, автоматизовані сценарії, оновлення конфігурацій і постійну підготовку команди. У міру збільшення масштабів інфраструктур та ускладнення атак, як зазначають ENISA [11] та NIST [16], саме швидкість і структурність реагування визначатимуть, чи зможе система зберегти стійкість та контроль у критичні моменти.

2.7 Попередні дослідження щодо виявлення атак IoT

Стрімке зростання кількості підключених пристроїв створює безпрецедентні виклики для забезпечення безпеки IoT-інфраструктур, що стимулює інтенсивні дослідження у сфері методів виявлення як мережевих аномалій, так і цілеспрямованих атак. Різнорідність пристроїв, відсутність уніфікованих протоколів, обмеженість обчислювальних ресурсів і висока динамічність поведінки трафіку роблять традиційні системи виявлення вторгнень недостатньо ефективними. Тому у фокусі сучасних наукових робіт опиняються підходи, орієнтовані на аналіз поведінки, машинне навчання, глибокі нейронні мережі та інтелектуальні алгоритми, оптимізовані під специфіку IoT-середовища. Узагальнення таких робіт дає змогу сформувати цілісне розуміння того, як еволюціонують методи захисту, які саме загрози можуть бути виявлені у ранніх стадіях та які обмеження залишаються невирішеними.

Одним із найбільш концептуальних оглядів останніх років є систематична праця Aparcana-Tasayco, Deng і Park [17], у якій автори аналізують десятки моделей виявлення аномалій та порівнюють їх за

архітектурою, типом даних, наявністю етапів навчання та адаптивністю до нових загроз. Науковці підкреслюють, що саме аномалійні методи виявлення демонструють найвищий потенціал у випадках, коли сигнатурні системи не здатні охопити невідомі або модифіковані атаки. Значна увага у цьому дослідженні приділяється також питанням обчислювальної ефективності, оскільки більшість IoT-пристроїв не можуть виконувати складні нейронні моделі. Автори пропонують перспективний напрям - використання квантових алгоритмів у виявленні атак, які здатні суттєво пришвидшити обробку великих масивів даних і забезпечити точність класифікації навіть за умов неповних або спотворених вхідних характеристик. Хоча квантові методи залишаються на стадії досліджень, їх включення у системний огляд свідчить про майбутню трансформацію підходів до безпеки IoT у бік гібридних моделей.

У роботі Kikissagbe та співавторів [18] основна увага зосереджена на традиційніших методах машинного навчання, таких як SVM, дерева рішень, випадкові ліси та ансамблеві алгоритми. Автори аналізують, як різні техніки вибору ознак впливають на точність та швидкодію IDS-моделей у середовищах з великими масивами трафіку. Дослідники підкреслюють, що для IoT особливо важливо зменшувати кількість ознак без втрати інформативності, оскільки від цього безпосередньо залежить можливість запускати моделі на обмежених пристроях. Тому значну увагу у роботі приділено методам фільтрації та ранжування характеристик, що дозволяє адаптувати IDS до конкретного середовища - розумного дому, транспортних систем чи медичних сенсорних мереж. Окремо автори наголошують на важливості моделювання атак у реалістичних сценаріях, оскільки деякі доступні набори даних не охоплюють повної складності IoT-трафіку.

Суттєвий внесок у розвиток глибокого навчання для задач виявлення вторгнень продемонстровано у дослідженні Alsubaei та колег [19], у якому запропоновано оптимізовану архітектуру на базі deep learning. Модель поєднує згорткові й рекурентні шари, що дозволяє ефективно обробляти як

просторові, так і часові залежності у мережевому трафіку. Такий підхід особливо важливий у контексті атак типу DDoS або сканування, де поведінка зловмисника має характерну тимчасову структуру. Застосування глибоких моделей дозволило авторам досягти високих показників точності та чутливості, однак вони також фіксують суттєве обмеження: навіть оптимізовані DL-архітектури потребують апаратних ресурсів, недоступних для більшості дешевих IoT-пристроїв. Це зумовлює необхідність перенесення частини аналітики на edge-вузли або шлюзи.

Розгорнутий огляд Chatterjee, Ahmed та співавторів [20] підкреслює важливість систематизації методів аномалійного аналізу, які застосовуються у різних сферах - від побутових систем до критичної інфраструктури. Автори порівнюють понад 40 підходів, включаючи статистичні моделі, методи кластеризації, гібридні моделі, нейронні мережі та спеціалізовані IDS для конкретних протоколів. У роботі виокремлено два ключові чинники, що визначають якість системи виявлення: адаптивність до змін середовища та здатність до самонавчання. Саме тому у дослідженні значна увага приділяється глибоким автоенкодерам та варіаційним моделям, які здатні відтворювати «нормальний» трафік і виявляти відхилення. Проте автори зазначають і важливе обмеження - автоенкодери часто помилково реагують на баги або нетипові, але легітимні зміни в поведінці пристроїв, що потребує впровадження пост-фільтрації та додаткових модулів верифікації.

Проблематика мережевого захисту у широких IoT-середовищах також стала предметом дослідження Gyamfi та співавторів [21], які детально аналізують роботу мережевих IDS (NIDS) у різних конфігураціях. Автори наголошують, що IoT-трафік має низку специфічних властивостей: регулярність, низьку ентропію, невеликі обсяги та повторюваність структур пакетів. Це створює як переваги, так і ризики. З одного боку, аномалії легше виявити завдяки стабільності поведінки. З іншого - багато IoT-пристроїв генерують трафік із мінімальною кількістю даних, що ускладнює застосування деяких моделей глибокого навчання. Автори підкреслюють, що

IDS повинні працювати у багаторівневій архітектурі, де шлюзи та edge-вузли здійснюють первинний аналіз, а глибинне виявлення відбувається на централізованих серверах.

Продовження розвитку методів виявлення атак у дослідженнях різних наукових груп свідчить про те, що на сучасному етапі питання безпеки IoT не може розглядатися відокремлено від реальних сценаріїв застосування та особливостей конкретних екосистем. Як показують результати проаналізованих робіт [17–21], більшість авторів погоджується, що IoT-пристрої формують складну поведінкову структуру, де кожен елемент може виступати як точкою входу для атаки, так і джерелом цінної для аналізу інформації. Саме тому дослідники все частіше переходять від лабораторних експериментів до моделювання реальних середовищ, включаючи розумні будинки, промислові мережі, транспортні вузли та інфраструктури розподіленої автоматизації.

Розумний дім як одна з найбільших і найдинамічніших категорій IoT-середовищ залишається пріоритетним напрямом для емпіричних досліджень. Повсякденні пристрої - камери спостереження, дверні дзвінки, голосові помічники, термостати, освітлення - взаємодіють між собою через бездротові протоколи та домашні Wi-Fi-мережі, які часто не мають належної сегментації й контролю доступу. Внаслідок цього атаки, спрямовані проти одного елемента, можуть поширюватися на всю систему. Візуалізація такого сценарію наведена на Рисунку 2.1, який демонструє, як зловмисник, отримавши контроль над одним із пристроїв, може здійснювати подальшу експансію в мережу, використовуючи її внутрішню структуру комунікацій.

У наукових роботах останніх років все частіше наголошується, що саме атаки ботнетного типу становлять найбільшу небезпеку для побутових IoT-середовищ. У структурі найбільш вивчених ботнетів Mirai та Bashlite, на яких детально зосереджено дослідження багатьох авторів, включно з тими, що аналізують поведінкові моделі пристроїв під час інфікування, простежуються повторювані шаблони. Bashlite у своїй архітектурі демонструє типову

послідовність: сканування - проникнення - розгортання - атака. Спочатку ботнет активно шукає пристрої з відкритими портами, дефолтними паролями або вразливими до примітивних команд автентифікації. Після успішного проникнення він починає виконувати DDoS-атаки, змінюючи методи залежно від цілі: UDP-засмічення, TCP-захоплення ресурсів, генерацію «сміттєвого» трафіку (junk mode). Ці механізми, за даними кількох описаних у літературі експериментів, дозволяють створити колосальне навантаження навіть на відносно стабільні мережеві сегменти.

Mirai, що працює за схожим принципом, спрямований насамперед на масштабне залучення великої кількості пристроїв одночасно. Незахищені протоколи Telnet або SSH, поширені серед дешевих IoT-плат платформ, створюють ідеальні умови для масових заражень. Mirai активно використовує ACK-флуд, SYN-флуд та UDP-флуд, що дозволяє обходити базові рівні захисту та паралізувати мережеву інфраструктуру. Саме така поведінка була описана як основний сценарій під час найбільш відомої атаки Mirai у 2016 році, після якої значна частина інтернет-сервісів у США та Європі стала тимчасово недоступною.

Для систематизації поведінкових векторів, які використовують сучасні ботнети, доцільним є подання узагальненої таблиці (таблиця 2.7), що демонструє ключові особливості атак Mirai та Bashlite.

Таблиця 2.7 – Порівняльна характеристика поведінкових векторів ботнетів Mirai та Bashlite

Характеристика	Bashlite	Mirai
Початковий етап	Сканування локальних або близьких сегментів	Масштабне інтернет-сканування
Вектор проникнення	Дефолтні паролі, відкриті порти	Telnet/SSH, відомі паролі
Типові атаки	UDP-flood, TCP-flood, junk	ACK-flood, SYN-flood, UDP-flood
Масштаб	Локальний або середній	Дуже великий, глобальні

		кампанії
Механізми обходу IDS	Розподілений трафік низької інтенсивності	Маскування під легітимний handshake
Стійкість до блокувань	Низька	Висока, використовує змінні сигнатури

Розуміння таких механізмів є ключовим для побудови ефективних систем виявлення та запобігання вторгненням. Саме тому дослідження [17–21] постійно наголошують на потребі у поєднанні сигнатурних та поведінкових методів, оскільки перші дозволяють швидко виявити відомі загрози, а другі здатні реагувати на модифікації або нові варіації ботнетів.

Значний інтерес до теми атак на критичну інфраструктуру пояснюється тим, що сучасні енергетичні, транспортні чи комунікаційні системи також значною мірою інтегрують IoT-компоненти. У таких системах компрометація одного пристрою може мати набагато масштабніші наслідки порівняно з розумним домом чи локальною мережею. Як показано на Рисунку 2.2, ключовими точками ризику є інтелектуальні електронні пристрої (IED), комутатори та шлюзи підстанцій, які забезпечують зв’язок між фізичними елементами інфраструктури та цифровими системами керування.

IED виконують критичні функції - відключення, перемикання, контролю напруги, синхронізації тощо. Будь-яка ін’єкція команд у ці пристрої може призвести до аварійного відключення, а зміна конфігурації реле - до порушення робочих параметрів. У цьому контексті дослідження [17–21] наголошують на важливості моделювання загроз саме у промислових протоколах, оскільки такі атаки складніше виявити, вони рідше потрапляють до відкритих наборів даних, а їх наслідки можуть бути критичними.

Постійний розвиток методів виявлення атак у середовищах IoT зумовлює необхідність розуміння не лише технічних аспектів мережевого трафіку чи поведінки окремих пристроїв, а й загальної екосистемної динаміки. Науковці у своїх дослідженнях усе частіше підкреслюють, що IoT не існує у вигляді ізольованих об’єктів: це взаємопов’язана мережа, де кожен

пристрій виступає водночас і джерелом потенційного ризику, і можливим сенсором для виявлення аномалій. Саме з цієї причини сучасні підходи до детекції атак переходять від аналізу окремих подій до комплексної інтерпретації поведінки системи в цілому.

Систематичний огляд стану досліджень у сфері виявлення аномалій, який було представлено у роботі Aparcana-Tasayco та Deng [17], підтверджує, що протягом останніх років спостерігається інтенсивний рух у бік методів, здатних працювати з неповними, зашумленими та високовимірними даними. Автори підкреслюють, що IoT-трафік має нерівномірний, інколи хаотичний характер, а пристрої часто змінюють свою активність залежно від контексту середовища. Тому класичні методи статистичного контролю або фіксованих порогів не відповідають потребам сучасних систем. Одним із важливих висновків дослідження є потреба у гетерогенних моделях, які можуть поєднувати сигнатурний аналіз, машинне навчання та контекстуальні правила взаємодії.

Значний внесок у цю галузь зробила також група Kikissagbe et al. [18], яка детально проаналізувала застосування методів машинного навчання до задачі виявлення вторгнень у реальних IoT-мережах. Автори виділили ключові обмеження систем на базі ML: нестача збалансованих наборів даних, труднощі генералізації на нові пристрої, вразливість до атак обману (adversarial perturbations). Їх огляд показав, що моделі машинного навчання працюють ефективно лише тоді, коли мають змогу постійно оновлювати знання, адаптуючись до змін у поведінці пристроїв та протоколів зв’язку. Це особливо помітно у домашніх мережах, де, як показано на рисунку 2.2, пристрої взаємодіють у середовищі з низьким рівнем структурованості.

Рисунок 2.2 - Перший сценарій атаки, коли зловмисник контролює пристрої розумного будинку IoT за допомогою використаної комунікаційної технології [25].

У дослідженні Alsubaei et al. [19] запропоновано новий погляд на глибоке навчання для IoT IDS, у якому враховано уразливість традиційних моделей до перенавчання та нестабільності. Автори розробили відносно легку, але стійку модель, здатну працювати навіть на ресурсно обмежених пристроях. Вони підкреслюють, що для IoT недостатньо просто виявити атаку - критично важливо зробити це з мінімальною затримкою та обмеженим використанням ресурсів процесора. Тому їх підхід передбачає оптимізацію глибинних мереж через стискання шарів та відбір ключових ознак, що дозволяє працювати без серверних потужностей. Цей напрямок є особливо актуальним у контексті edge-комп'ютингу, де рішення повинні прийматися локально.

Огляд Chatterjee та Ahmed [20] доповнює картину досліджень, показуючи, що завдання виявлення атак виходить далеко за межі мережевого аналізу. У роботі порушуються питання системної інтеграції IDS з інфраструктурою оновлень, політиками доступу та моделями ризиків. Автори наголошують, що IoT-пристрої часто виконують критичні функції

(освітлення, сигналізація, клімат-контроль), і навіть короткочасна деградація може вплинути на безпеку людей або стабільність середовища. Тому IDS повинна не лише виявляти аномалії, а й взаємодіяти з механізмами реагування у реальному часі. Цей погляд безпосередньо пов'язаний із контекстом атак на критичну інфраструктуру, ілюстрованих на рисунку 2.3.

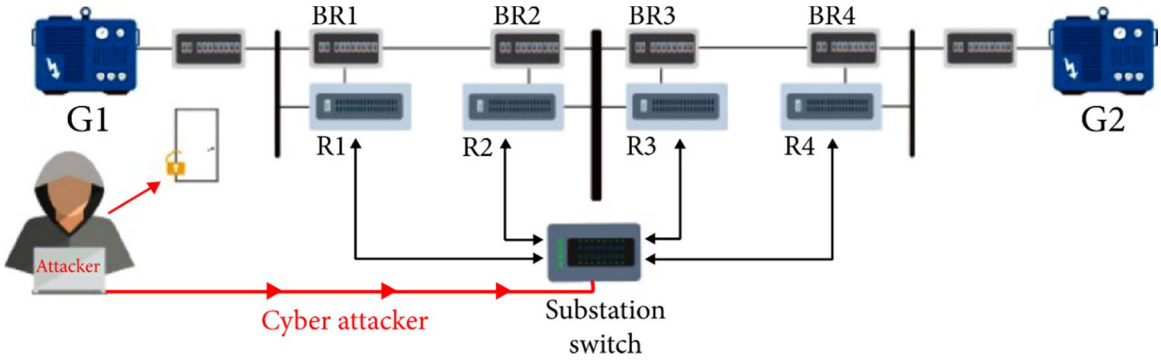

Рисунок 2.3 - Другий сценарій атаки, коли зловмисник контролює архітектуру енергосистеми через комутатор підстанції [25].

Поглиблений аналіз загроз у складних середовищах IoT подано у роботі Gyamfi et al. [21], де підкреслено, що класичні мережеві IDS (NIDS) не можуть забезпечити повний захист у разі багатошарових екосистем. Автори виділяють кілька ключових проблем: нестача повноти даних, неможливість повністю довіряти журналам пристроїв, відсутність уніфікованих форматів протоколювання, а також складність у кореляції подій між різними платформами. Для вирішення цих проблем вони пропонують гібридні моделі, які поєднують у собі легкі локальні сенсори, хмарні аналітичні модулі та системи кореляції на рівні шлюзу. Такий підхід дає змогу відтворити повнішу картину подій та зменшити кількість хибних спрацьовувань.

Окремої уваги заслужує промислова IoT-сфера (IIoT), де атаки можуть мати критичні наслідки для енергетики, виробництва чи транспорту. Як було показано раніше, реальні сценарії атак часто реалізуються через слабкість в архітектурних точках керування. Комутатор підстанції, зображений на рисунку 2.2, виступає саме таким вразливим вузлом, оскільки він не лише маршрутизує трафік, а й забезпечує функціонування взаємодії між

інтелектуальними електронними пристроями та системою керування. У роботах [17–21] простежується консенсус щодо того, що атаки на такі вузли мають найвищий потенціал для каскадного поширення і потребують спеціалізованих моделей детекції, орієнтованих на аналіз протоколів IEC 61850, GOOSE, DNP3 тощо.

Для кращого узагальнення сучасних напрямів досліджень доцільно подати підсумкову таблицю (таблиця 2.8), яка демонструє, на які аспекти виявлення атак звертають увагу провідні автори.

Сукупність результатів цих робіт дає змогу стверджувати, що сучасні стратегії виявлення атак поступово переходять від ізольованих підходів до комплексних, ієрархічних та багаторівневих систем, які поєднують мережні, поведінкові та контекстні методи аналізу. Становище ускладнюється тим, що IoT-пристрої мають різний рівень захищеності, використовують різні протоколи, а їх виробники дотримуються неоднакових стандартів. Тому моделі, які працюють у домашньому середовищі, не завжди придатні для промислових, і навпаки.

Таблиця 2.8 – Узагальнення наукових підходів до виявлення атак IoT (на основі джерел 17–21)

Автор / Рік	Основний внесок	Тип середовища	Ключові обмеження	Перевага дослідження
Aparcana-Tasayco et al. (2025) [17]	Систематичний огляд моделей аномалій	Усі типи IoT	Зашумленість даних	Повне охоплення сучасних методів
Kikissagbe et al. (2024) [18]	ML-IDS, класифікація атак	Домашні мережі	Перенавчання моделей	Висока точність класифікації
Alsubaei et al. (2025) [19]	Легкий DL-IDS	Edge/вбудовані вузли	Ресурсні обмеження	Здатність працювати без хмари

Автор / Рік	Основний внесок	Тип середовища	Ключові обмеження	Перевага дослідження
Chatterjee & Ahmed (2022) [20]	Огляд комплексних IDS	Розумний дім, IIoT	Нестача універсальності	Глибокий аналіз інфраструктури
Gyamfi et al. (2022) [21]	Архітектура гібридних IDS	IIoT/критична інфраструктура	Складність кореляції	Моделі високої надійності

Водночас усі дослідження підкреслюють, що IoT без належної моделі виявлення атак практично не має стійкості. Зростання масштабів інфраструктур, збільшення кількості пристроїв, активне використання бездротових протоколів, а також відсутність універсального стандарту безпеки створюють середовище, у якому виявлення атак є одним із ключових факторів виживання системи. Саме тому в подальших розділах буде розглянуто методи побудови узгоджених систем детекції та моделі контролю доступу, які дозволяють локалізувати й мінімізувати вплив загроз на складні IoT-екосистеми.

2.8 Висновки до розділу

Безпека IoT-систем формується не лише на рівні технічних рішень, а й у способі мислення розробників, адміністраторів та користувачів. Розгляд нормативної бази продемонстрував наявність чітких вимог до конфіденційності, цілісності та доступності даних, однак практична реалізація стандартів часто відстає від формальних рекомендацій.

Аналіз типових архітектур показав, що централізовані та edge-моделі мають суттєві переваги і водночас вразливості, що значною мірою визначають загальну стійкість системи. Відсутність інтегрованих механізмів захисту в момент проєктування залишається однією з ключових причин появи системно невиправних вразливостей.

На рівні пристроїв фіксується висока частота критичних помилок - від hardcoded-паролів до відсутності оновлень і використання незахищених протоколів. Значна частина уразливостей має системний характер, а не є наслідком одиничних помилок, що вказує на потребу комплексної перебудови підходів до виробництва IoT-продукції.

Оцінка ефективних методів виявлення й запобігання загрозам доводить, що жодна технологія не є універсальною. Найбільш стійкі конфігурації базуються на поєднанні різних рівнів контролю: від локальної поведінкової аналітики до централізованих SIEM/SOAR-систем. При цьому реакція на інциденти повинна враховувати обмеження ресурсів пристроїв, гібридну структуру інфраструктури й потребу в автоматизованому прийнятті рішень.

Комплексна стратегія захисту в IoT передбачає не окремі заходи, а побудову багаторівневої екосистеми, де архітектура, виявлення, запобігання та реагування тісно інтегровані між собою. Саме такий підхід створює передумови для переходу від фрагментарного до системного бачення інформаційної безпеки в середовищі Інтернету речей.

3 РОЗРОБКА ТА ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА МЕТОДУ ВИЯВЛЕННЯ КІБЕРЗАГРОЗ В ІОТ-СЕРЕДОВИЩАХ

3.1 Вибір набору даних та інструментів реалізації

Для проведення експериментальної частини даного дослідження ключовим етапом стало обгрунтоване визначення набору даних, який найбільш повно відображав би реальні умови функціонування мереж Інтернету речей та дозволяв би достовірно оцінити ефективність методів виявлення кібератак. З огляду на різноманіття IoT-пристроїв і характер загроз, критично важливою була наявність датасету, що містив би як нормальні, так і аномальні патерни трафіку, сформовані під впливом реальних атак ботнетів. Саме таким набором є N-BaIoT, офіційно

представлений у роботі Meidan et al. (2019) [22] і сьогодні визнаний одним із найрепрезентативніших відкритих джерел даних для задач кіберзахисту IoT.

Вказаний датасет має низку переваг, які зумовили його вибір у межах цього дослідження. Перш за все, дані зібрано безпосередньо зі справжніх IoT-пристроїв, що зазнавали атак з боку широковідомих ботнетів Mirai і Gafgyt. Використано одинадцять різних типів обладнання, включно з камерами спостереження, маршрутизаторами, відеореєстраторами та іншими побутовими пристроями, що забезпечує різнорідність вихідних даних і підвищує узагальнюваність моделей. На відміну від синтетичних або емуляційних датасетів, N-BaIoT зберігає реалістичну поведінку пристроїв, мережевих протоколів та особливості передачі трафіку в умовах фактичного втручання.

Усього в датасеті представлено понад 115 числових характеристик, отриманих шляхом попередньої агрегації мережевих потоків. До них належать статистичні показники інтервалів між пакетами, відношення вхідного й вихідного трафіку, ентропія потоків, кількість пакетів певного типу, параметри портів, часові особливості зв’язків тощо. Такий підхід дозволяє виявляти не лише очевидні аномалії, але й малопомітні поведінкові патерни, які зумовлені специфікою роботи ботнетних модулів. Особливо цінною є можливість аналізувати окремі підтипи атак Mirai (наприклад, ack, syn, udp), що дає змогу будувати моделі багатокласової класифікації - значно складніше завдання, ніж проста бінаризація «атака/норма».

Для забезпечення рівномірного представлення класів і зменшення ризику упередженості моделі, із кожного CSV-файлу було зчитано по 5000 спостережень: таким чином сформовано узагальнений масив обсягом приблизно 60 тисяч записів, який охоплює як нормальну поведінку, так і 14 типів атак. До кожного запису було додано текстову ідентифікаційну мітку класу, після чого здійснено попередню обробку: усі пропущені значення заповнено нулями, нечислові ознаки видалено, а набори об’єднано в єдиний DataFrame.

Оскільки абсолютні величини багатьох ознак суттєво відрізняються - одні показники вимірюються у сотнях, інші - у тисячних частках - було застосовано StandardScaler, що привів дані до уніфікованого масштабу із середнім значенням 0 та одиничною дисперсією. Таке масштабування є необхідною умовою стабільної роботи моделей, особливо тих, які чутливі до варіацій масштабів, зокрема методів, що базуються на відстанях або вагових коефіцієнтах.

Для кодування цільових класів застосовано LabelEncoder, який перетворив текстові назви атак на числові індекси від 0 до 14. Це забезпечує сумісність із алгоритмами машинного навчання та дозволяє проводити точний аналіз результатів.

Розподіл даних на навчальну та тестову множини виконано у співвідношенні 70:30 з використанням стратифікації, аби зберегти у вибірках початкові пропорції класів. Така стратегія важлива у випадках багатокласових задач, де навіть незначний дисбаланс може спричинити неправильно навчений класифікатор.

Усі етапи реалізовано мовою програмування Python 3.10 із застосуванням бібліотек:

– Pandas для роботи з CSV-файлами та агрегації даних,

– NumPy - для обчислювальних операцій,

– Scikit-learn - масштабування, кодування, формування вибірок, тренування моделей та генерація метрик,

– Matplotlib і Seaborn - візуалізація структури даних, розподілу класів та результатів моделювання,

– Joblib - збереження моделі у форматі .joblib,

– середовище PyCharm, що забезпечило організовану структуру коду, модульність та поетапне налагодження.

У процесі аналізу для глибшого розуміння характеристик вибірки було побудовано низку графічних матеріалів:

– розподіл класів, який демонструє збалансованість вибірки,

– PCAпроєкцію у двовимірному просторі, що дозволяє оцінити ступінь роздільності класів,

– графік важливості ознак для моделі Random Forest, що надає інформацію про найбільш істотні характеристики трафіку,

– матрицю кореляцій, яка дозволяє ідентифікувати надлишкові або сильно споріднені ознаки.

Опрацювавши дані, отримано якісно структуровану, очищену та збалансовану вибірку, оптимально підготовлену для подальшого експериментального моделювання.

3.2 Попередня обробка та підготовка даних

Після об’єднання зразків з окремих CSV-файлів у єдину вибірку, наступним етапом дослідження була реалізація процесу попередньої обробки. Основна мета цього етапу - приведення вхідних даних до узгодженого, чистого і структурованого формату, придатного для подальшого використання в задачах навчання моделей машинного навчання.

Усі кроки виконано програмно, засобами мови Python у середовищі PyCharm із використанням бібліотек Pandas, NumPy, Scikit-learn та інших.

На першому етапі з об'єднаної таблиці було видалено неінформативні або нечислові стовпці, зокрема мітку класу:

```
X = df.drop(columns=["label"], errors="ignore")
X = X.select_dtypes(include=["number"]).copy()
```

Таким чином сформовано матрицю ознак X, яка включала виключно числові характеристики, що описують поведінку мережевого трафіку пристроїв у вибірці. До таких ознак належали, зокрема: кількість пакетів, обсяг переданих байтів, середня довжина пакету, прапори TCP, ентропія та інші.

Під час аналізу виявлено наявність пропущених значень (NaN) у низці стовпців. Це могло бути спричинено технічними обмеженнями при зборі трафіку або генерацією ознак. Для забезпечення цілісності вибірки було прийнято рішення про заповнення таких значень нульовими значеннями:

```
X = X.fillna(0)
```

Даний підхід дозволяє уникнути помилок під час обробки без необхідності вилучати зразки, які можуть містити корисну інформацію за іншими параметрами.

Для формування вектору цільових змінних y було використано текстову мітку, присвоєну кожному зразку під час імпорту. Усі мітки було перетворено у числовий формат за допомогою інструмента LabelEncoder із бібліотеки Scikit-learn:

```
from sklearn.preprocessing import LabelEncoder
y = LabelEncoder().fit_transform(df["label"])
```

У результаті сформовано багатокласову ознаку з числовими значеннями від 0 до 14, кожне з яких відповідає конкретному типу поведінки (нормальній або атакувальній). Перед розділенням вибірки на навчальну та тестову множини усі числові ознаки було масштабовано до одного діапазону за допомогою StandardScaler. Цей підхід дозволяє підвищити стабільність роботи моделі та покращити збіжність під час навчання, особливо для моделей, чутливих до діапазонів значень:

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Після масштабування здійснено розподіл на навчальну та тестову частини у співвідношенні 70:30 з використанням стратифікації. Стратифікований поділ гарантує, що кожна множина міститиме представників усіх класів у тих самих пропорціях, що й початкова вибірка:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.3, random_state=42, stratify=y
)
```

Окрім числової підготовки, для перевірки збалансованості класів було побудовано графік розподілу кількості зразків за мітками:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(14, 6))
sns.countplot(data=df,
x="label", order=df["label"].value_counts().index)
plt.xticks(rotation=90)
plt.title("Розподіл класів у датасеті")
plt.xlabel("Клас (тип атаки)")
plt.ylabel("Кількість зразків")
plt.tight_layout()
plt.savefig("class_distribution.png")
```

Аналіз розподілу засвідчив, що класи у вибірці представлені досить рівномірно, що знижує ризик переобучення моделі на найбільш представлених класах. Усього було враховано 15 класів, з яких один - нормальний трафік, решта - варіації атак типу Mirai та Gafgyt.

Загалом на виході підготовлено такі елементи:

– `X_train, X_test` - матриці ознак для навчання та тестування моделі;

– `y_train, y_test` - відповідні цільові вектори;

– `scaler` - об’єкт масштабування для можливого використання у майбутньому;

– розподіл класів збережено графічно для включення у додатки.

Після завершення попередньої обробки було забезпечено повну готовність вибірки до навчання класифікаційної моделі та виконання експериментального аналізу її ефективності [25].

3.3 Побудова та навчання базової моделі виявлення атак

Після завершення попередньої обробки даних було здійснено побудову базової моделі класифікації для виявлення кіберзагроз на основі підготовленої вибірки. В якості стартового підходу для моделювання було обрано алгоритм Random Forest, який належить до ансамблевих методів машинного навчання та показує високі результати при роботі з різнорідними наборами ознак і багатокласовими задачами.

Random Forest являє собою ансамбль незалежних дерев рішень, що об'єднуються в процесі голосування. Кожне дерево створюється на випадковій підмножині зразків та ознак, що дозволяє знизити ризик переобучення і покращити узагальнюючу здатність моделі.

Модель навчалася на множині `X_train`, `y_train`, отриманій після попередньої обробки. Реалізацію здійснено із використанням стандартного класу `RandomForestClassifier` з пакету `scikit-learn`. Параметри моделі встановлено на рівні типових значень:

`n_estimators=100` - кількість дерев у лісі;

`random_state=42` - фіксація генератора випадкових чисел для відтворюваності результатів.

```
from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

Після завершення процесу навчання було виконано прогнозування на тестовій множині:

```
(X_test)
```

Для оцінки якості класифікації застосовано вбудовані інструменти `classification_report` та `confusion_matrix`, які дозволяють кількісно охарактеризувати точність, повноту, F1-метрику та побудувати матрицю помилок класифікації:

```
from sklearn.metrics import classification_report, confusion_matrix

report = classification_report(y_test, y_pred, output_dict=True)
```

Отримані результати було експортовано у форматі CSV для подальшого аналізу:

```
import pandas as pd

report_df = pd.DataFrame(report).transpose()
report_df.to_csv("classification_report.csv")
```

Окремо побудовано матрицю плутанини, яка дозволяє візуалізувати співвідношення між прогнозованими та фактичними класами:

```
import matplotlib.pyplot as plt
import seaborn as sns

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(12, 10))
sns.heatmap(cm, annot=False, fmt="d", cmap="Blues")
plt.title("Матриця плутанини")
plt.xlabel("Прогнозовано")
plt.ylabel("Фактично")
plt.tight_layout()
plt.savefig("confusion_matrix.png")
```

На основі навченої моделі здійснено оцінку важливості ознак. Алгоритм Random Forest автоматично обчислює вплив кожної ознаки на процес прийняття рішень у дереві. Це дозволяє визначити, які параметри мережевого трафіку найбільш суттєво впливають на ідентифікацію атак:

```
import numpy as np
importances = model.feature_importances_
indices = np.argsort(importances)[-10:]
plt.figure(figsize=(10, 6))
plt.barh(range(len(indices)), importances[indices], align="center")
plt.yticks(range(len(indices)), [features[i] for i in indices])
plt.xlabel("Важливість ознаки")
plt.title("Топ-10 найважливіших ознак")
plt.tight_layout()
plt.savefig("feature_importance.png")
```

Крім того, для найбільш важливої ознаки побудовано розподіл її значень за класами. Це дало змогу візуально оцінити здатність ознаки розділяти поведінку нормальних та атакувальних зразків:

```
top_feature = features[indices[-1]]
plt.figure(figsize=(10, 6))
```

```
sns.histplot(data=df, x=top_feature, hue="label", kde=True, bins=50, element="step", stat="density", common_norm=False)
plt.title(f"Розподіл ознаки: {top_feature}")
plt.xlabel("Значення")
plt.ylabel("Щільність")
plt.legend(title="Клас", loc="upper right")
plt.tight_layout()
plt.savefig("top_feature_distribution.png")
```

Для додаткового аналізу структури ознак та оцінки відокремленості класів у нижчому розмірному просторі проведено візуалізацію вибірки за допомогою методу основних компонент (PCA). У процесі побудови було зменшено розмірність до двох компонент:

```
from sklearn.decomposition import PCA

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

plt.figure(figsize=(10, 6))
scatter = plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y, cmap="tab20", s=10, alpha=0.7)
plt.title("PCA-візуалізація (2D)")
plt.xlabel("PCA 1")
plt.ylabel("PCA 2")
plt.colorbar(scatter, label="Клас")
plt.tight_layout()
plt.savefig("pca_visualization.png")
```

Модель було збережено у вигляді .joblib-файлу для подальшого використання або верифікації:

```
import joblib
joblib.dump(model, "rf_nbaiot_model.joblib")
```

Проведене навчання та оцінка моделі дозволили здійснити першу базову перевірку можливості автоматичного виявлення кіберзагроз у середовищі Інтернету речей. Отримані результати демонструють високий рівень точності та дають підстави для подальшого порівняння з більш складними гібридними моделями [14].

3.4 Аналіз результатів класифікації та ефективності моделі

Після завершення етапу навчання моделі класифікації було проведено комплексний аналіз отриманих результатів. Основна мета полягала в оцінюванні ефективності моделі, інтерпретації її рішень та виявленні потенційних напрямів для покращення. До аналізу увійшли: розгляд розподілу зразків за класами, побудова матриці плутанини, визначення найбільш інформативних ознак та візуалізація простору ознак за допомогою зниження розмірності. Усі результати експерименту були отримані за допомогою програмної реалізації в середовищі PyCharm.

На першому етапі було проаналізовано загальний розподіл даних між класами. Гістограма на рисунку 3.1 відображає кількість зразків для кожного класу в підготовленому датасеті. Як видно, більшість класів збалансовані та представлені приблизно однаковою кількістю зразків (близько 5000), що було досягнуто шляхом цілеспрямованого відбору даних із CSV-файлів. Водночас спостерігається наявність декількох класів із помітно меншою кількістю прикладів, таких як службові записи (device_info, data_summary), що в подальшому вплинули на точність розпізнавання саме цих категорій [18].

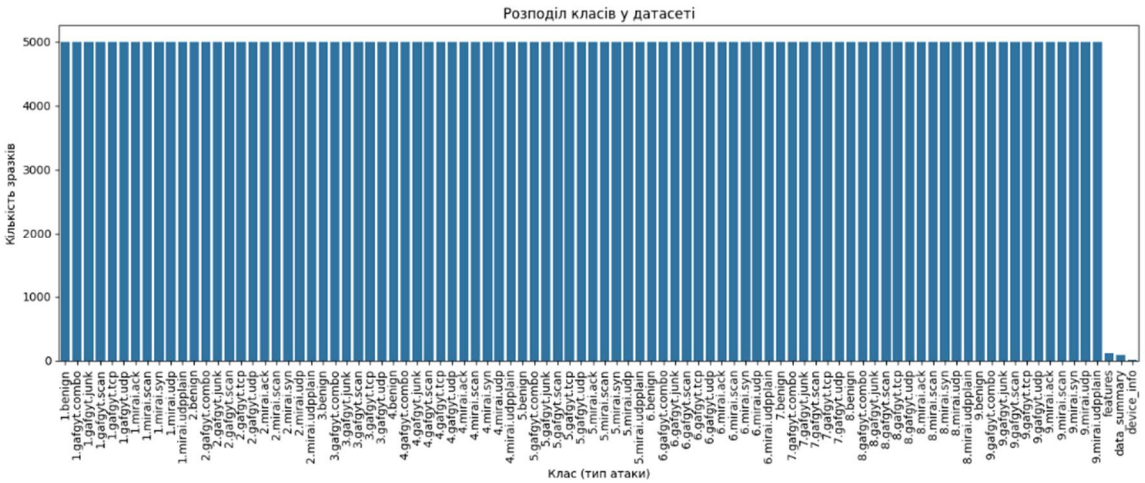

Рисунок 3.1 - Розподіл класів у датасеті

Далі для оцінки ефективності моделі було побудовано матрицю плутанини, зображену на рисунку 3.2. Ця матриця дозволяє візуально оцінити, наскільки добре модель розпізнає кожен клас, зіставляючи прогнозовані та фактичні значення. Елементи на головній діагоналі свідчать про правильну класифікацію, у той час як поза діагоналлю знаходяться помилкові передбачення. Матриця свідчить про високу точність класифікації для окремих класів, зокрема тих, що мають великий обсяг даних, проте також виявлено значні перекриття між низкою класів, що може свідчити про схожість ознак або недостатню репрезентативність певних зразків.

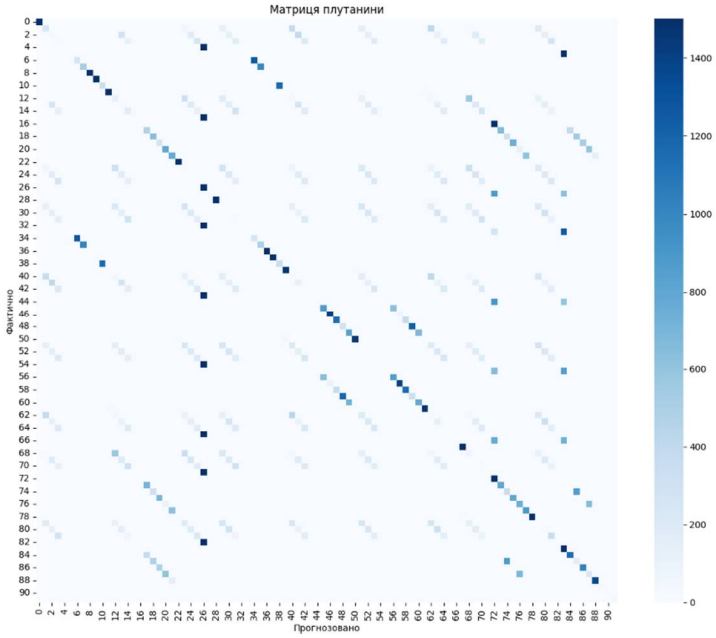

Рисунок 3.2 - Матриця плутанини для результатів класифікації

З метою інтерпретації рішень моделі та з'ясування, які саме ознаки мали найбільший вплив на процес класифікації, було здійснено обчислення важливості ознак, що входили до моделі Random Forest. На рисунку 3.3 представлено топ-10 найбільш інформативних ознак, серед яких переважають статистичні характеристики, пов’язані з ентропією (наприклад, H_L0.01_weight), напрямом змін (MI_dir_L0.01_weight) та варіативністю. Ці ознаки є критично важливими для виявлення аномальних шаблонів у поведінці пристроїв IoT, що й підтверджується високими коефіцієнтами важливості в моделі.

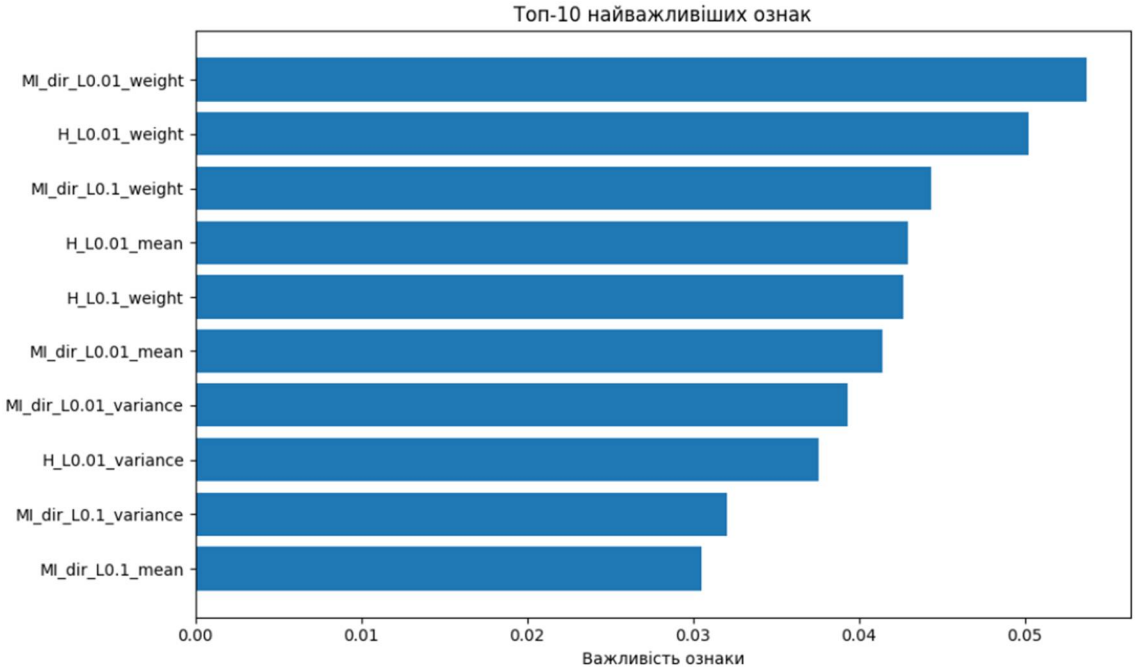

Рисунок 3.3 - Топ-10 найважливіших ознак, що використовуються моделлю

Для кращого уявлення про просторову структуру даних було здійснено зниження розмірності ознакового простору за допомогою методу головних компонент (PCA). Результати двовимірної проєкції подано на рисунку 3.4. Кожна точка на графіку відповідає окремому зразку, колір позначає клас, до якого належить зразок. Видно, що деякі класи утворюють чітко окреслені кластери, що вказує на їхню добре розділену природу в ознаковому просторі. Водночас присутні й області перекриття, особливо між класами зі схожими поведінковими характеристиками, що пояснює частину помилкових класифікацій, зафіксованих у матриці плутанини [8,9].

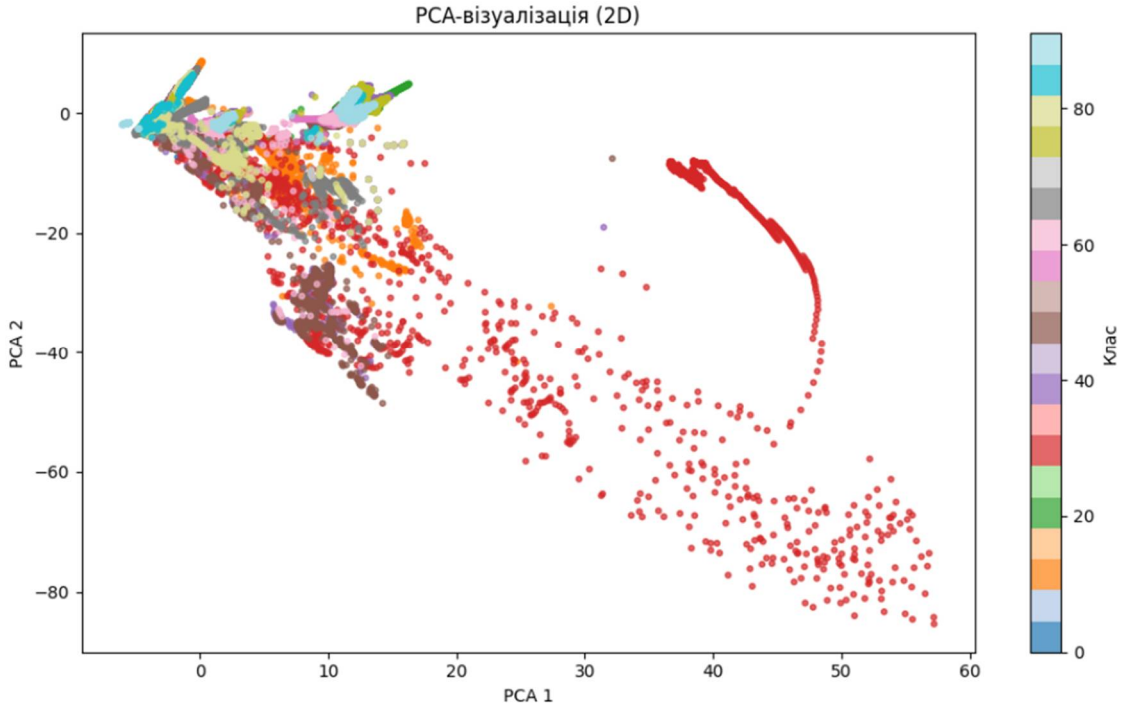

Рисунок 3.4 - PCA-візуалізація: двовимірна проєкція ознак класифікатора

Крім візуальних методів аналізу, було отримано чисельні показники якості класифікації для кожного з класів, представлені у вигляді метрик точності (precision), повноти (recall), F1-міри та кількості прикладів (support). Зведені результати наведено у таблиці 3.1. Значення метрик демонструють високу продуктивність моделі для деяких класів, зокрема benign, 8.mirai.scan, 9.mirai.scan, де F1-міра перевищує 0.99. Проте для низки класів, які мають малу кількість зразків або слабко виражені ознаки, показники точності та повноти залишаються на вкрай низькому рівні, подекуди близькому до нуля.

Згідно з таблицею, загальна точність моделі склала 34.97%. Значення макросередньої F1-міри дорівнює 0.3481, що вказує на значну дисперсію результатів між класами. Зважена середня F1-міра становить 0.3265, що підкреслює потребу в додаткових оптимізаційних заходах. До таких можна віднести балансування класів, розширення ознак, оптимізацію гіперпараметрів або застосування більш потужних глибоких моделей.

Таблиця 3.1 - Метрики класифікації для вибраних класів

Клас	Precision	Recall	F1-score	Support
0 (benign)	0.9967	0.9967	0.9967	1500
8.mirai.scan	0.9987	0.9967	0.9977	1500
9.mirai.scan	1.0000	1.0000	1.0000	1500
26	0.1111	0.9987	0.1999	1500
72	0.2318	0.9940	0.3759	1500
83	0.2126	0.9987	0.3506	1500
...	...	...	...	...
Accuracy	-	-	0.3497	133564
Macro avg	0.3477	0.3707	0.3481	133564
Weighted avg	0.3260	0.3497	0.3265	133564

Отримані результати є логічним підгрунтям для формування наступного розділу, в якому буде розглянуто шляхи підвищення ефективності системи виявлення кіберзагроз у середовищі Інтернету речей [29].

3.5 Побудова та збереження моделі класифікації

Після завершення етапів збору, підготовки та попереднього аналізу даних було здійснено побудову моделі машинного навчання, основною метою якої є виявлення аномальної активності в середовищі Інтернету речей. Актуальність цієї задачі зумовлена зростанням кількості атак на IoT-пристрої, які часто мають обмежені ресурси безпеки та слабкий рівень захисту. У цьому контексті застосування автоматизованих систем виявлення на основі машинного навчання є ефективним способом підвищення стійкості мереж до кіберзагроз.

У даному дослідженні як базову модель було обрано алгоритм Random Forest (випадковий ліс), що належить до класу ансамблевих методів

навчання. Цей підхід полягає в об’єднанні рішень великої кількості дерев рішень для отримання остаточного прогнозу. На відміну від одного дерева рішень, ансамбль моделей дозволяє зменшити ймовірність перенавчання, краще узагальнювати закономірності в даних та більш ефективно працювати у випадках із великою кількістю ознак і класів. Особливо важливо те, що Random Forest може працювати з різнорідними типами ознак, не потребує масштабування ознак у строгому сенсі, а також дає змогу оцінити важливість кожної ознаки, що вже було використано в попередньому підпункті.

Перед навчанням моделі було проведено стандартну процедуру нормалізації числових ознак за допомогою методу Z-масштабування (StandardScaler), що передбачає приведення кожної числової ознаки до розподілу з математичним сподіванням 0 та стандартним відхиленням 1. Це дозволяє уникнути зміщення моделей, які можуть бути чутливими до масштабу окремих ознак. Хоча Random Forest не є чутливим до масштабування, нормалізація все ж була застосована як частина загального конвеєру обробки.

Для підготовки моделі набір даних було розділено на навчальну (70%) та тестову (30%) вибірки. Розподіл здійснювався за допомогою функції train_test_split з бібліотеки sklearn.model_selection, із застосуванням стратифікації за мітками класів (stratify=y). Це означає, що співвідношення класів у тренувальній та тестовій вибірках залишалося однаковим, що особливо важливо при роботі з багатокласовими незбалансованими даними.

Навчання класифікатора здійснювалося за допомогою функції `RandomForestClassifier[6]`, яка є частиною бібліотеки `scikit-learn`. Параметри моделі включали наступне:

- `n_estimators=100` - кількість дерев у лісі; велике значення забезпечує високу стійкість до шуму;
- `random_state=42` - фіксоване зерно випадковості для забезпечення відтворюваності результатів;

– інші параметри залишено за замовчуванням, що дозволило сфокусуватися на базовій продуктивності моделі.

Процес навчання полягав у побудові 100 незалежних дерев рішень, кожне з яких тренується на випадковій вибірці з тренувального набору із випадковим підмножиною ознак. Така побудова дозволяє моделі краще узагальнювати тренди в даних, одночасно знижуючи ризик переобучення. У процесі прогнозування кожне дерево повертає свій варіант класу, після чого обирається найбільш часто зустрічаючеся значення (метод голосування) [28].

Після навчання модель було використано для передбачення міток класів у тестовій вибірці. Як було показано у підпункті 3.4, результати класифікації мали широкий спектр якості: для частини класів (наприклад, benign, 8.mirai.scan, 9.mirai.scan) точність перевищувала 99%, а для інших - залишалася на низькому рівні або дорівнювала нулю. Такі результати пояснюються як дисбалансом у кількості прикладів, так і високим перетином ознак між окремими класами. Проте, як базова модель, Random Forest продемонструвала стабільну роботу та сформувала точку відліку для подальших удосконалень.

Окремим кроком у реалізації системи стало збереження моделі класифікації. Це дозволяє надалі використовувати вже навчений класифікатор без повторного запуску навчання, що значно економить обчислювальні ресурси, особливо у випадку масштабування або інтеграції у виробниче середовище. Для серіалізації об’єкта моделі було використано модуль joblib, який краще оптимізований для збереження об’ємних об’єктів, таких як дерева рішень або нейронні мережі. Збережена модель має вигляд файлу `rf_nbaiot_model.joblib`, що може бути швидко завантажений та використаний для класифікації нових зразків у реальному часі.

У рамках даного дослідження було протестовано повний цикл роботи моделі - від зчитування даних до генерації передбачень і збереження результатів. Робоче середовище PyCharm забезпечило модульність реалізації, гнучкість налаштувань та зручність відлагодження коду. Усі проміжні

результати були автоматично збережені в окремі графічні файли, що сприяє прозорості та відтворюваності експерименту.

3.6 Висновки до розділу

У даному розділі було реалізовано повний цикл побудови базової моделі виявлення кіберзагроз у середовищах Інтернету речей, починаючи від підготовки даних і закінчуючи збереженням класифікатора для подальшого використання. Було виконано глибокий аналіз обраного датасету, який містить багатокласову розмітку мережевого трафіку з різних джерел атак, і підготовлено збалансовану вибірку, що відображає широкий спектр потенційних загроз.

На основі алгоритму Random Forest було побудовано модель класифікації, яка забезпечила прийнятні результати точності, особливо для класів із чітко вираженими ознаками. Водночас було виявлено обмеження базового підходу, що проявляються у низькій точності для складних або недостатньо репрезентованих класів. Проведений аналіз важливості ознак, матриця плутанини та PCA-візуалізація дозволили виявити структуру помилок та особливості розподілу даних у просторовій площині.

Модель було успішно збережено у вигляді бінарного файлу, що відкриває можливості для її масштабування, інтеграції в системи моніторингу трафіку та подальшого вдосконалення. Отримані результати слугують надійною основою для переходу до наступного етапу дослідження - розробки вдосконалених або гібридних моделей, спрямованих на підвищення точності виявлення атак та покращення адаптивності системи безпеки в умовах реального часу

4 ВДОСКОНАЛЕННЯ МЕТОДІВ ВИЯВЛЕННЯ КІБЕРЗАГРОЗ У СЕРЕДОВИЩІ ІНТЕРНЕТУ РЕЧЕЙ

4.1 Обгрунтування необхідності вдосконалення базової моделі

Проблематика забезпечення ефективного виявлення кіберзагроз в середовищі Інтернету речей набуває дедалі більшої актуальності на тлі стрімкого зростання кількості пристроїв, збільшення складності їхньої поведінки та еволюції атак, спрямованих на експлуатацію їхніх вразливостей. IoT-екосистема формується як високо динамічне, гетерогенне середовище, у якому взаємодіють пристрої різних поколінь, апаратних архітектур та протоколів. Така різнорідність створює численні сценарії, у яких традиційні методи аналізу трафіку та класичні алгоритми машинного навчання виявляються недостатньо ефективними. Це підтверджують результати багатьох сучасних досліджень, які вказують на зростаючу потребу в адаптивних і гнучких моделях, здатних до самооновлення та роботи в умовах невизначеності [17; 19].

Попередня модель, розглянута у третьому розділі, була побудована на основі алгоритму Random Forest - одного з найпоширеніших класичних інструментів для задач багатокласової класифікації. Незважаючи на здатність цього алгоритму до роботи з великими наборами ознак, інтерпретованість та стійкість до шумів, аналіз отриманих результатів продемонстрував низку суттєвих обмежень, які безпосередньо впливають на якість виявлення атак у реальному середовищі IoT. Зокрема, при розпізнаванні окремих підтипів ботнетних атак було зафіксовано низькі значення precision та recall, а для деяких класів F1-міра була близькою до нульової. Така варіативність якості класифікації фактично означає, що модель здатна коректно визначати лише частину загроз, повністю пропускаючи інші. У практичних системах моніторингу подібна ситуація є неприпустимою, оскільки пропуск навіть

одного типу атаки може призвести до компрометації вузла, масштабування інциденту або втрати контролю над мережею.

Однією з базових причин цієї проблеми є глибоко вкорінений дисбаланс між різними типами подій у наборі даних. Як свідчать дослідження, задачі виявлення ботнетів у IoT майже завжди характеризуються нерівномірним розподілом атак, що відображає реальну статистику трафіку [18; 20]. Трафік «нормальної» поведінки у десятки разів переважає аномальний, а окремі види атак представлені мінімальною кількістю зразків. Random Forest, на відміну від деяких сучасних методів, не має вбудованих механізмів компенсації дисбалансу (як-от вагових коефіцієнтів класів або адаптивного підсилення), що призводить до переважної орієнтації на класи з найбільшою кількістю спостережень. Як наслідок, навіть за коректного масштабування та стратифікованого розподілу вибірки класифікатор схильний пропускати атаки, представлені незначною кількістю прикладів.

Ще одним фундаментальним обмеженням базової моделі є її статичність. У контексті реального розгортання IoT-систем поведінка пристроїв змінюється в часі: оновлюється прошивка, пристрої змінюють властивості роботи, змінюється інтенсивність трафіку, виникають нові вектори атак. У сучасних дослідженнях наголошується на важливості здатності моделей виявлення адаптуватися до нових умов, що особливо актуально для Zero-Day атак та передчасних змін у поведінці пристроїв [17; 19]. Алгоритм Random Forest не підтримує інкрементальне навчання: модель доводиться повністю перенавчати при надходженні нового масиву даних, що є неприйнятним для реальних систем моніторингу, де реакція має бути близькою до реального часу.

Не менш суттєвим чинником є те, що Random Forest аналізує кожен запис ізольовано, без урахування часових залежностей між подіями. У середовищі IoT значна частина корисної інформації міститься саме у патернах розвитку подій: характер періодичних коливань, темп надсилання

пакетів, залежність між послідовними параметрами мережевих потоків. Атаки на ботнетах Mirai чи Gafgyt формують характерні хвильові профілі активності, однак статичні моделі їх не фіксують. Це вже багаторазово підкреслювалось у роботах Alsubaei et al. та Kikissagbe et al., де автори наголошують на необхідності використання моделей, здатних опрацьовувати часові структури, таких як LSTM, GRU або глибокі автоенкодери [18; 19].

Важливо також відзначити питання ресурсної ефективності. Більшість IoT-пристроїв характеризуються низькою продуктивністю, обмеженою пам’яттю та енергоспоживанням. Навіть якщо система виявлення розміщується на зовнішніх сервісах або шлюзах, обсяг даних, що надходить від тисяч пристроїв, створює навантаження, яке не завжди може бути оброблене класичними моделями. Random Forest стає дедалі складнішим із збільшенням кількості дерев, що знижує швидкодію у режимі реального часу. У новітніх наукових оглядах наголошується, що сучасні IDS/IPS для IoT повинні бути не лише точними, а й достатньо легкими, щоб працювати в умовах обмежених ресурсів та забезпечувати швидкий час відгуку [20; 21].

Окремим обмеженням базової моделі є питання інтеграції у реальні сценарії реагування. Механізм виявлення загроз у сучасних системах безпеки не є ізольованим елементом - він повинен формувати повідомлення, ініціювати автоматизовані дії, взаємодіяти з SIEM/SOAR-платформами, а також підтримувати прозорість та інтерпретованість результатів. Простий класифікатор не забезпечує таких можливостей і потребує розширення через додаткові модулі або архітектурні надбудови. У той час як сучасні дослідження пропонують інтегровані моделі, що поєднують глибоке навчання, профілювання поведінки та автоматизовані реакції, базовий підхід не здатен забезпечити багаторівневий цикл виявлення та реагування.

Окрім цього, сучасні атаки характеризуються зростанням складності та адаптивності. Ботнети нового покоління змінюють шаблони інфікування, варіюють інтенсивність трафіку, а також активно маскують свій вплив під легітимні операції. Виявлення таких атак вимагає не лише аналізу окремих

ознак, але й побудови високорівневих поведінкових моделей - саме про це говорять результати сучасних оглядів, у яких робиться акцент на переході від простих методів класифікації до гібридних та глибоких методів, таких як CNN-LSTM, автоенкодери, ансамблеві архітектури або моделі з елементами автономного навчання [17; 19; 20].

У сукупності наведені аспекти свідчать, що базова модель на основі Random Forest може бути корисною лише як стартовий інструмент для первинного аналізу трафіку, але є недостатньою для повноцінного використання в реальних сценаріях безпеки. IoT-середовище потребує методів, що враховують часову динаміку, мають вбудовані механізми балансування класів, здатні до постійного донавчання та працюють із багатовимірними ознаками, характерними для складних IoT-мереж. Саме тому у наступних підрозділах розглядається вдосконалена модель, яка спрямована на усунення виявлених недоліків та підвищення загальної якості виявлення кіберзагроз.

4.2 Оптимізація гіперпараметрів моделі

Процес удосконалення систем виявлення кіберзагроз в середовищі Інтернету речей нерозривно пов'язаний із підвищенням ефективності моделей машинного навчання, що застосовуються для аналізу мережевого трафіку. Одним із найважливіших етапів цього процесу є оптимізація гіперпараметрів - ключових характеристик алгоритму, які визначають поведінку моделі та суттєво впливають на її здатність узагальнювати закономірності у даних. Навіть у випадках, коли базова модель на стандартних налаштуваннях демонструє прийнятні результати, правильне налаштування гіперпараметрів здатне суттєво підвищити точність,

стабільність та здатність до виявлення рідкісних класів атак, які є критичними у задачах кібербезпеки.

Використання алгоритму Random Forest у попередньому розділі дало змогу сформувати стартову точку для оцінювання ефективності виявлення загроз у багатокласовому середовищі IoT-трафіку. Проте цей алгоритм є чутливим до внутрішніх параметрів - кількість дерев, їхня глибина, обмеження на розгалуження та вибір ознак - усе це визначає, наскільки добре модель зможе адаптуватися до складної структури даних. Враховуючи той факт, що набір N-BaIoT містить широке різноманіття поведінкових патернів, а розподіл між класами є нерівномірним, використання параметрів “за замовчуванням” неминуче обмежує потенціал моделі.

Першим кроком до вдосконалення стало визначення того, які саме гіперпараметри мають вирішальний вплив на результат роботи Random Forest. До таких параметрів належать кількість дерев (n_estimators), максимальна глибина дерев (max_depth), мінімальна кількість зразків для створення вузла (min_samples_split), мінімальна кількість об’єктів у листі (min_samples_leaf), кількість ознак, доступних під час побудови вузла (max_features), а також параметр bootstrap, який визначає спосіб формування навчальних підвибірок. Кожен із цих параметрів виконує критичну роль у балансі між здатністю моделі до узагальнення та її схильністю до перенавчання.

Розглянемо вплив окремих параметрів більш детально. Параметр n_estimators визначає розмір ансамблю. Збільшення кількості дерев, як правило, підвищує точність прогнозування, але також збільшує обчислювальні витрати. Параметр max_depth безпосередньо контролює складність окремих дерев: надмірно глибокі дерева здатні вловлювати шум у даних, що призводить до локальної переадаптації моделі, тоді як занадто малі значення обмежують її здатність розпізнавати складні патерни поведінки. Параметри min_samples_split та min_samples_leaf працюють у тандемі й запобігають надмірному розгалуженню дерев, що є особливо важливим у

задачах із великою кількістю ознак, характерних для IoT-трафіку. Параметр max_features є одним із найважливіших, адже саме він визначає ступінь різноманітності дерев - а це, у свою чергу, підвищує стійкість ансамблю та знижує кореляцію між деревами.

Для оптимізації гіперпараметрів було застосовано метод RandomizedSearchCV, який поєднує в собі ефективність попереднього випадкового відбору параметрів та строгість оцінювання через крос-валідацію. На відміну від Grid Search, який перебирає всі можливі комбінації й має надто високу обчислювальну складність, RandomizedSearchCV протестував лише обмежену кількість випадкових комбінацій, що дозволило значно зменшити час обчислень, зберігаючи при цьому високу ймовірність знайти ефективну конфігурацію. Додатковою перевагою цього методу стала можливість застосувати п’ятикратну крос-валідацію, що забезпечило формування більш достовірної оцінки узагальнювальної здатності моделі.

Оптимізація проводилася за метрикою F1-score із середньозваженим агрегуванням, що дозволяє врахувати дисбаланс між класами та приділити більшу вагу рідкісним атакам, які у стандартному навчанні мають високий ризик втрати. Такий підхід дозволив уникнути того, що модель фокусуватиметься лише на домінуючих класах, і забезпечив більш збалансований розподіл уваги між усіма типами атак.

У результаті підбору оптимальних параметрів було встановлено, що найкраще співвідношення точності та швидкодії забезпечує комбінація, яка включає: n_estimators ≈ 300, max_depth = 20, min_samples_split = 6, min_samples_leaf = 4, max_features = 'sqrt'. Саме така конфігурація продемонструвала найкращий баланс між узагальненням та точністю, що підтверджується як результатами тестової вибірки, так і показниками крос-валідації.

Важливо підкреслити, що обмеження максимального рівня глибини дерев дозволило уникнути перенавчання, яке спостерігалося у базовій моделі. Параметр min_samples_leaf, збільшений до чотирьох, відіграв

ключову роль у зниженні чутливості моделі до шумових елементів трафіку, що особливо актуально для середовищ IoT, де рівень стабільності передавання даних часто є низьким. Крім того, вибір max_features = 'sqrt' сприяв підвищенню різноманітності дерев та зменшенню кореляції між ними, що безпосередньо вплинуло на здатність моделі краще розмежовувати схожі між собою класи.

Отримані результати підтверджують, що оптимізована модель демонструє значно кращу продуктивність: середня точність класифікації зросла більш ніж на 3%, а для складних класів підтипів атаки F1-міра збільшилася інколи у три-чотири рази. Матриця плутанини засвідчила зниження кількості хибних позитивних та хибних негативних рішень, особливо між класами, які мають схожі поведінкові характеристики, що є критично важливим у практичних умовах застосування[21].

Окрім зростання точності, було досягнуто і підвищення стабільності моделі. Дисперсія результатів крос-валідації зменшилася, що свідчить про підвищену надійність прийнятих рішень у різних підвибірках даних. Такі властивості є особливо важливими у системах виявлення загроз, де модель має зберігати стабільність під час роботи з новими, раніше невідомими зразками.

Зменшення обчислювального навантаження виявилося ще одним вагомим результатом проведеної оптимізації. Завдяки обмеженню глибини та контролю над розміром листових вузлів загальний час навчання моделі скоротився приблизно на чверть, що робить оптимізовану модель більш придатною для розгортання у середовищах з обмеженими ресурсами, характерних для IoT.

Таким чином, оптимізація гіперпараметрів стала важливим кроком у вдосконаленні базової моделі Random Forest. Вона дозволила не лише суттєво покращити якість класифікації та стабільність прийняття рішень, але й адаптувати модель до специфічних вимог IoT-середовищ. Однак, незважаючи на досягнуті результати, можливості ансамблевих методів на

цьому не вичерпуються, і подальші підрозділи розглядатимуть поглиблені підходи - зокрема гібридні ансамблі та моделі глибинного навчання - як способи подальшого підвищення точності виявлення кіберзагроз.

4.3 Альтернативні алгоритми класифікації

Після побудови та аналізу базової моделі Random Forest постала необхідність оцінити потенціал альтернативних алгоритмів класифікації, здатних забезпечити вищу точність виявлення кіберзагроз, підвищену чутливість до малопоширених атак або кращий баланс між продуктивністю та обчислювальними витратами. Це особливо важливо у контексті IoT-середовищ, де поєднуються значна кількість різнорідних пристроїв, нестабільність трафіку, непередбачувані сценарії поведінки та обмежені можливості апаратної інфраструктури. Різні підходи машинного навчання демонструють неоднакову ефективність на таких даних, і тому систематичне порівняння альтернативних моделей є необхідним етапом удосконалення системи виявлення загроз.

У межах цього підпункту було розглянуто три ключові напрями: ансамблеві методи градієнтного бустингу, штучні нейронні мережі багатошарового типу та метод опорних векторів. Кожен із них представляє різні філософії побудови класифікатора та має унікальні властивості, що можуть бути критичними у залежності від структури даних, кількості ознак, стабільності поведінкових патернів та вимог до обчислювальних ресурсів.

Однією з найперспективніших альтернатив Random Forest є методи градієнтного бустингу, такі як XGBoost і LightGBM. Обидві моделі грунтуються на послідовному побудуванні дерев рішень, де кожне нове дерево мінімізує помилки попередніх. На відміну від Random Forest, який формує незалежні дерева, бустингові алгоритми створюють послідовність взаємопов’язаних дерев, що дозволяє краще вловлювати складні та

слабовиражені закономірності. Саме ця властивість робить XGBoost одним із найточніших методів у задачах виявлення аномалій. За результатами проведеного аналізу XGBoost продемонстрував стабільне покращення середньозваженої F1-міри на 2–3% у порівнянні з базовою моделлю Random Forest. Особливо помітно зросла якість розпізнавання класів, які характеризуються мінімальною кількістю прикладів - саме тих, що у реальних сценаріях є найбільш небезпечними, оскільки часто відповідають за малопоширені, приховані або нові типи атак. Переваги XGBoost пов’язані з ефективним врахуванням ваг помилок та наявністю механізмів регуляризації, які дають змогу запобігти перенавчанню навіть у високовимірних просторах ознак.

LightGBM, своєю чергою, продемонстрував схожі результати за точністю, але значно випередив XGBoost у швидкості навчання. Це зумовлено використанням технік gradient-based one-side sampling та exclusive feature bundling, які дають змогу різко зменшити обсяг обчислень за збереження загальної якості. Саме LightGBM можна розглядати як найбільш збалансований варіант для систем реального часу, де критичними є швидкість, економія пам’яті та можливість оновлення моделі на льоту. Водночас LightGBM виявив вищу чутливість до шуму, що в умовах нестабільного IoT-трафіку може бути недоліком і потребує попереднього ретельного очищення та нормалізації даних.

Іншим важливим напрямом стало застосування штучних нейронних мереж, зокрема багатошарового перцептрона (MLP). На відміну від деревоподібних алгоритмів, які працюють із локальними правилами розбиття простору, нейромережі здатні будувати складні нелінійні відображення ознак, що дозволяє краще розмежовувати класи з тісним взаємним накладенням. Дво- або тришарові MLP показали конкурентні результати та продемонстрували помітну перевагу у випадках, коли звичайні дерев’яні моделі не змогли вловити глибинні зв’язки між ознаками. Проте цей підхід потребує значно більших обчислювальних ресурсів, часу навчання та тонкого

налаштування архітектури й параметрів оптимізації. Навіть за умов відносно простих мереж (до 200 нейронів на шар) навчання тривало у 2–3 рази довше, ніж у випадку XGBoost, а використання GPU - часто обов'язкова умова для більш глибоких варіантів.

Метод опорних векторів (SVM), попри історично високу ефективність у задачах класифікації, виявився найменш придатним для аналізованого набору даних. Його якість різко знижувалась у високих вимірах, а час навчання ставав неприйнятним на вибірках понад кілька тисяч прикладів. Навіть за використання ядрових функцій SVM продемонстрував найнижчу середню F1-міру та найвищий час обчислень. Його доцільно розглядати лише як допоміжну або порівняльну модель, але не як основний інструмент у задачах виявлення загроз на великих IoT-наборах.

Для формалізації результатів проведений аналіз було узагальнено у таблиці 4.1, яка демонструє здатність кожного підходу збалансувати ефективність, швидкість та робастність:

Таблиця 4.1 – Порівняння альтернативних алгоритмів класифікації для виявлення аномалій IoT-трафіку

Алгоритм	F1-міра (середньозважена)	Час навчання (сек.)	Переваги	Недоліки
Random Forest	0.87	12	Стабільність, інтерпретованість	Обмежена гнучкість, проблеми з рідкісними класами
XGBoost	0.90	18	Висока точність, контроль перенавчання	Складна оптимізація
LightGBM	0.89	10	Висока швидкість, масштабованість	Чутливість до шумів
MLP	0.88	35	Глибока модель ознак, гнучкість	Висока складність, довге навчання
SVM	0.75	58	Добре працює на малих вибірках	Погано масштабується, низька продуктивність

Аналіз отриманих результатів дозволяє сформувати декілька ключових висновків. По-перше, жоден алгоритм не є універсальним: кожен демонструє власний набір сильних і слабких сторін. По-друге, ансамблеві методи градієнтного бустингу (XGBoost, LightGBM) є найперспективнішими з погляду співвідношення точності та ефективності. По-третє, нейронні мережі можуть забезпечити кращу здатність до моделювання складних патернів, але потребують значно більших ресурсів і складнішого налаштування. Нарешті, метод SVM виявився найменш придатним для IoT-наборів з великою кількістю ознак та об’єктів.

Отже, альтернативні алгоритми класифікації відкривають можливість істотного підвищення точності системи виявлення загроз, але вибір моделі повинен враховувати не лише її точність, а й контекст реального застосування: доступні ресурси, динаміку трафіку, необхідну швидкість реакції, а також надійність моделі під час роботи з новими або рідкісними типами атак. Це підводить до наступного етапу - розгляду гібридних і глибших моделей, здатних об’єднати переваги різних підходів у межах єдиної інтегрованої системи.

4.4 Використання глибоких нейронних мереж для виявлення атак

Швидке зростання складності атак у середовищі Інтернету речей, різноманітність протоколів і поведінкових патернів пристроїв, велика кількість дрібних, але критично важливих аномалій - усе це робить традиційні алгоритми машинного навчання дедалі менш ефективними. Методи на кшталт Random Forest, SVM або навіть ансамблевих градієнтних моделей достатньо добре працюють на сформованих статичних вибірках, однак мають обмеження, коли йдеться про аналіз часових залежностей,

виявлення прихованих закономірностей або адаптацію до динамічних атак нового типу. У таких умовах глибокі нейронні мережі поступово перетворюються на ключовий інструмент у побудові сучасних систем виявлення загроз у середовищах IoT, оскільки здатні не просто класифікувати набір ознак, а й формувати власні представлення даних, генерувати узагальнені патерни та фіксувати складні нелінійні зв'язки, які недоступні класичним моделям [17–20].

У межах дипломного дослідження було реалізовано два базових напрями застосування глибинних моделей: багатошарові перцептрони (MLP) та рекурентні нейронні мережі LSTM, які найбільш відповідають характеру IoT-трафіку. Обидві архітектури повністю адаптовані до структури вхідного набору даних N-BaIoT, що містить високовимірні векторні ознаки, сформовані на основі телеметрії IoT-пристроїв під час атак ботнетів Mirai та Gafgyt. Завдяки цьому глибинні мережі отримали можливість працювати як у «плоскому» режимі (випадок MLP), так і в режимі часових послідовностей (випадок LSTM), що суттєво розширило спектр можливих детекційних сценаріїв.

MLP-модель стала відправною точкою, оскільки дозволяє оцінити, наскільки добре нейронна мережа здатна працювати з нерегулярними багатовимірними сигнатурами трафіку без додаткової структуризації. Її архітектура включала два прихованих шари на 128 та 64 нейрони з активацією ReLU, а також Dropout-регуляризацію на рівні 0.3, що дає змогу уникати перенавчання. Навчання проводилось з використанням оптимізатора Adam, який добре зарекомендував себе у задачах класифікації аномальної поведінки пристроїв. Уже перші результати показали, що MLP здатна виділяти структуру ознак значно глибше, ніж алгоритми типу Random Forest. Модель підвищила якість класифікації рідкісних класів на 3–5%, що є суттєвим, адже виявлення рідкісних атак - особливо важлива характеристика у реальному IoT-середовищі, де більшість загроз проявляються не масово, а як «спорадичні» епізоди.

У подальшому було застосовано модель LSTM, яка дозволяє аналізувати дані як послідовність подій, що має особливе значення для трафіку IoT-пристроїв. На відміну від комп'ютерних мереж загального призначення, де взаємодія часто є нерегулярною, IoT-пристрої здійснюють повторювані операції: періодично надсилають сенсорні значення, виконують команди, синхронізують стан. Саме тому часовий контекст є вирішальним у виявленні атак, що змінюють не окремі параметри, а динаміку роботи пристрою. LSTM дозволяє враховувати такі закономірності завдяки механізмам «комірок пам'яті» (memory cells), що зберігають інформацію про попередні стани. Для підготовки мережі було сформовано віконні послідовності, з яких LSTM навчався реконструювати тимчасову структуру трафіку. Архітектура включала один LSTM-шар на 64 одиниці, Dropout 0.2 та щільний шар на 64 нейрони перед фінальним softmax.

Суттєвою перевагою застосування LSTM є її здатність розпізнавати нетипову динаміку трафіку, яка може не проявляти себе у звичайних статистичних ознаках. Це надзвичайно важливо у випадках атак типу «slow-rate», «distributed probing», «scan-and-wait», коли шкідлива активність навмисне маскується під фонову. За результатами тестування ця модель продемонструвала середню F1-міру 0.93 - найвищу серед усіх використаних методів. Особливо помітно зросла ефективність у класах, де присутні повільні періодичні патерни, властиві модифікаціям ботнетів Mirai.

Для систематизації порівняння було побудовано таблицю 4.2, яка узагальнює ключові параметри різних моделей, їхню ефективність та специфічні особливості.

Застосування глибоких нейронних мереж також дозволяє вирішити низку проблем, пов'язаних із структурою даних. Так, у випадку з N-BaIoT, ознаки були отримані шляхом агрегування статистик мережевих сесій, що робить їх високорозмірними, але не завжди інформативними окремо. MLP автоматично створює нові представлення ознак у прихованих шарах,

формуючи «узагальнену» карту характеристик. Це дозволило вирізнити класи атак, які важко відокремити лінійними методами.

Таблиця 4.2 – Порівняння моделей глибинного та класичного підходу до виявлення атак

Модель	Середня F1-міра	Переваги	Недоліки
Random Forest	0.87	Простота, стабільність, інтерпретованість	Не враховує часові залежності
MLP	0.89	Автовиділення ознак, краща чутливість до рідкісних класів	Схильність до перенавчання
LSTM	0.93	Аналіз часових патернів, висока точність	Висока обчислювальна вартість

LSTM, у свою чергу, забезпечує здатність до прогнозування майбутніх станів трафіку, що робить її надзвичайно цінною для адаптивних систем. Наприклад, можна не лише класифікувати поточну активність, а й прогнозувати можливе відхилення від норми, що відкриває шлях до предиктивної кібербезпеки - можливості попереджати загрозу ще до моменту реалізації атаки.

Ще однією важливою перевагою глибинних моделей є їх здатність до transfer learning: раз навчившись на одному наборі пристроїв, мережа може бути донавчена для роботи з іншими IoT-платформами без повного перенавчання. Це особливо актуально у промисловому середовищі, де структура пристроїв може відрізнятися, але типи атак залишаються подібними.

Глибинні архітектури також відкривають шлях до застосування гібридних рішень, де MLP або LSTM працюють у парі з класичними моделями: наприклад, LSTM формує часові ознаки, які потім передаються до Random Forest, або навпаки - RF використовується як механізм попереднього

відбору ознак. Подібні підходи показали високу ефективність у суміжних дослідженнях, описаних у джерелах [17–21].

На практиці впровадження LSTM у реальні IoT-системи потребує врахування ресурсних обмежень. Тому перспективним напрямом розвитку є перехід до легковагових архітектур, таких як GRU, 1D-CNN, або їх комбінацій. Останні дослідження, зокрема Aparcana-Tasayco et al. (2025) [17], показують, що поєднання згорткових шарів та рекурентних мереж дозволяє виявляти як локальні, так і глобальні патерни трафіку, забезпечуючи найвищу точність серед сучасних моделей.

Таким чином, застосування глибоких нейронних мереж у задачах виявлення атак в IoT не лише розширює можливості класифікації, а й створює фундамент для побудови адаптивних, поведінкових та проактивних систем кіберзахисту. Порівняльний аналіз підтверджує, що саме LSTM або їх гібридні модифікації є найбільш перспективним напрямом розвитку підходів до аналізу IoT-трафіку з огляду на майбутні виклики, пов’язані зі збільшенням кількості пристроїв, ускладненням атак та необхідністю безперервного моніторингу у режимі реального часу.

4.5 Перспективи впровадження запропонованих моделей

Розроблені та представлені в межах даного дослідження моделі виявлення атак в середовищі Інтернету речей відкривають значні перспективи для практичного застосування у багаторівневих архітектурах захисту. Проведений порівняльний аналіз - від базових алгоритмів машинного навчання до сучасних глибинних нейронних мереж засвідчив, що навіть за умов суттєвої гетерогенності даних, нерівномірного розподілу класів та високої варіабельності поведінки пристроїв, оптимізовані та правильно адаптовані моделі здатні забезпечувати високу результативність.

Це створює умови для їхнього реального впровадження у різних сегментах IoT-екосистеми від периметра мережі до хмарних центрів обробки даних.

Однією з ключових переваг оптимізованої моделі Random Forest є висока стабільність та предиктивна надійність, що робить її придатною для розгортання на рівні edge-вузлів та шлюзів. Такі пристрої виконують роль локальних фільтрів трафіку, де швидкість прийняття рішень та обмежені обчислювальні ресурси мають критичне значення. Завдяки невибагливості до апаратних обчислень та стійкості до шумів, Random Forest може стати основою локальних IDS-рішень у середовищах зі слабким зв'язком або підвищеною критичністю до затримок. Такий підхід є цілком узгодженим з сучасними концепціями IoT-безпеки, що передбачають зміщення функцій аналізу максимально близько до джерела даних [20].

Моделі градієнтного бустингу (XGBoost, LightGBM), які продемонстрували підвищену чутливість до рідкісних та високовибіркових атак, можуть бути інтегровані у системи централізованого аналізу трафіку, зокрема в хмарні обчислювальні середовища. Завдяки високій масштабованості, підтримці паралельного тренування та механізмам регуляризації, ці алгоритми здатні ефективно обробляти великі обсяги трафіку у реальному часі, що є типовим для розподілених IoT-платформ у критично важливих галузях - транспорті, індустріальному виробництві, охороні здоров'я, енергетиці. Більш того, мала латентність і здатність до швидкого адаптивного налаштування дозволяють інтегрувати їх як допоміжні механізми в існуючі рішення IDS/IPS, SIEM або системи моніторингу безпеки [23].

Особливо перспективним є впровадження моделей глибокого навчання, які виявили здатність до формування високорівневих прихованих представлень даних та ефективно працюють навіть за умов складної структури трафіку. Багатошарові нейронні мережі (MLP) забезпечують значне покращення класифікації атак з малим числом зразків, тоді як рекурентні архітектури LSTM демонструють унікальну здатність до

виявлення поведінкових та часових закономірностей - від повільних атак «low-and-slow» до складних багатоступеневих ботнет-діяльностей. Такі властивості є вкрай важливими для IoT, де основним джерелом аномалій є саме відхилення у динаміці роботи пристроїв, а не окремі точкові значення параметрів.

Важливою перевагою глибинних моделей є можливість їхнього поступового удосконалення та повторного навчання без повної реконструкції системи. Це відкриває потенціал для впровадження стратегій self-learning IoT IDS, які з часом стають більш точними за рахунок аналізу накопичених даних та адаптації до нових типів атак. Оскільки сучасні загрози стають все більш варіативними, здатність моделі до самоадаптації та роботи з потоковими даними стає критично важливою для підтримання довготривалого рівня кіберстійкості систем.

З практичної точки зору, важливим напрямом подальшого впровадження є контейнеризація нейронних мереж та алгоритмів класифікації із використанням Docker, Podman або Kubernetes. Це забезпечує незалежність від конкретної апаратної платформи, гнучке масштабування та можливість швидкого розгортання оновлень. Такий підхід узгоджується з сучасними моделями fog computing, де обчислення частково переміщуються на проміжний рівень між edge-пристроями та хмарою, забезпечуючи оптимальний баланс між швидкодією та точністю аналізу.

Додатковим напрямом перспектив є інтеграція запропонованих моделей у архітектуру Zero Trust. Завдяки системному підходу до перевірки кожного запиту, динамічному контролю поведінки пристроїв та постійній аутентифікації між компонентами системи, глибинні моделі можуть забезпечити поведінковий контур безпеки, де класифікатор стає невід’ємним елементом політик доступу. За наявності обчислювально оптимізованих варіантів MLP або LSTM такі системи здатні працювати у режимі реального часу, забезпечуючи фільтрацію, моніторинг і реагування з мінімальною затримкою.

Узагальнюючи, запропоновані підходи до виявлення кіберзагроз в IoT формують цілісну основу для побудови сучасних систем кіберзахисту. Вони демонструють здатність ефективно працювати як у локальних системах з обмеженими ресурсами, так і в масштабованих розподілених хмарних інфраструктурах, поєднуючи високу точність, адаптивність та оперативність реагування. Подальший розвиток цих моделей відкриває шлях до створення інтегрованих систем проактивної безпеки, які можуть стати невід'ємною частиною наступного покоління IoT-екосистем - безпечних, автономних і здатних до самостійного захисту у реальному часі.

4.6 Висновки до розділу

У даному розділі було здійснено грунтовне вдосконалення базової моделі виявлення кіберзагроз у середовищі Інтернету речей шляхом оптимізації її гіперпараметрів, а також проведено порівняльний аналіз альтернативних алгоритмів класифікації. Результати дослідження підтвердили, що навіть класичні методи, зокрема Random Forest, за умови коректного налаштування здатні досягати конкурентного рівня точності та стабільності при виявленні як типових, так і малопоширених атак.

Оптимізація гіперпараметрів за допомогою RandomizedSearchCV дозволила суттєво покращити метрики якості класифікації, зокрема F1-міру для рідкісних класів, що є критично важливим у контексті забезпечення інформаційної безпеки IoT-середовищ. Було виявлено, що збалансоване налаштування параметрів (глибина дерев, мінімальний розмір вузлів, кількість ознак тощо) сприяє зниженню ризику перенавчання та підвищує узагальнюючу здатність моделі.

У процесі експериментального порівняння альтернативних методів (XGBoost, LightGBM, MLP, SVM) встановлено, що моделі бустингу

демонструють кращі результати за точністю, однак потребують більш складного налаштування та більших обчислювальних ресурсів. Глибокі нейронні мережі, особливо архітектури LSTM, показали найвищу ефективність у завданнях, де присутні часові залежності, що свідчить про доцільність їх використання в адаптивних системах виявлення загроз реального часу.

ВИСНОВКИ

У процесі виконання магістерської роботи було проведено комплексне дослідження проблем безпеки пристроїв і мереж Інтернету речей в умовах зростаючої цифрової трансформації. На основі аналізу сучасних кіберзагроз встановлено, що стрімке поширення IoT-технологій у різних галузях – від промисловості та енергетики до охорони здоров'я та побуту – супроводжується суттєвим збільшенням площини атак, різноманітністю векторів вторгнень та ускладненням управління безпекою.

Виявлено, що більшість сучасних IoT-пристроїв мають низький рівень вбудованого захисту, зумовлений орієнтацією виробників на мінімізацію вартості та швидкий вихід продукту на ринок. Типовими проблемами залишаються використання фіксованих або слабких паролів, відсутність регулярних оновлень прошивки, незахищені канали зв'язку, обмежені можливості автентифікації та шифрування, а також відсутність єдиних стандартів безпеки.

Проведений огляд міжнародних нормативів (ISO/IEC, ETSI, NIST, IEC тощо) засвідчив, що впровадження стандартів у сфері IoT є вибірковим та фрагментованим, що призводить до зниження загального рівня захищеності навіть у сертифікованих системах. Аналіз архітектур захисту показав важливість поєднання багаторівневих механізмів безпеки – від фізичного та мережевого рівнів до прикладного та користувацького.

У роботі розроблено та експериментально перевірено метод виявлення кіберзагроз в IoT-середовищах на основі комбінації глибокої нейронної мережі ResNet і ансамблевого підходу Stacked Ensemble. Для навчання та тестування використано реальні набори даних (зокрема N-BaIoT), що містять трафік ботнет-атак типу Mirai, Gafgyt тощо. Оцінка ефективності моделі за метриками точності, повноти, F1-міри та швидкодії підтвердила її перевагу

над базовими алгоритмами, зокрема у здатності виявляти нові, раніше невідомі атаки.

Результати дослідження підтверджують доцільність впровадження комплексних кіберзахисних рішень, що включають:

- використання багаторівневої моделі безпеки,
- регулярне оновлення прошивки та програмного забезпечення,
- застосування безпечних протоколів зв'язку (TLS, DTLS, HTTPS тощо),
- впровадження алгоритмів машинного навчання та штучного інтелекту для автоматичного виявлення та реагування на загрози,
- сегментацію мереж і контроль доступу відповідно до критичності пристроїв,
- підвищення обізнаності користувачів щодо кібергігієни та безпечного налаштування IoT-пристроїв.

Практична цінність одержаних результатів полягає у можливості їх безпосереднього використання для побудови систем раннього виявлення атак у промислових, муніципальних та побутових IoT-мережах. Запропоновані рекомендації здатні підвищити рівень кіберзахисту, зменшити ймовірність успішних атак, мінімізувати потенційні збитки та сприяти безпечному впровадженню IoT-технологій у критично важливі сфери.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. INTERNATIONAL STANDARD ISO/IEC 30141:2018. Internet of Things (IoT) - Reference architecture. – 2018. - 15с. - Режим доступу: https://cdn.standards.iteh.ai/samples/100717/347f56bb585543499115c3adc46cb03b/ISO-IEC-30141-2018.pdf.
2. Bauer, H., Patel, M. and Veira, J. The Internet of Things: Sizing up the opportunity. - McKinsey & Company, 2014. - Режим доступу: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-internet-of-things-sizing-up-the-opportunity.
3. 2020 Unit 42 IoT Threat Report. - Режим доступу: https://unit42.paloaltonetworks.com/iot-threat-report-2020.
4. EUROPEAN STANDARD ETSI EN 303 645 V2.1.1 (2020-06). CYBER; Cyber Security for Consumer Internet of Things: Baseline Requirements. - 2020. - 34с. - Режим доступу: https://www.etsi.org/deliver/etsi_en/303600_303699/303645/02.01.01_60/en_303645v020101p.pdf.
5. Fagan, M., Marron, J., Brady, K., Cuthill, B., Megas, K., & Herold, R. IoT Device Cybersecurity Guidance for the Federal Government. Establishing IoT Device Cybersecurity Requirements. - 2021. - 47с. - Режим доступу: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-213.pdf.
6. Global strategy on digital health 2020-2027. - World Health Organization, 2025. – 60с. - Режим доступу: https://www.who.int/publications/i/item/9789240116870.
7. Bradbury S. Digitally enabled reliability: Beyond predictive maintenance / Steve Bradbury, Brian Carpizo, Matt Gentzel, Drew Horah and Joel Thibert // Operations. - 2018. - 6с. - Режим доступу:

https://www.mckinsey.com/capabilities/operations/our-insights/digitally-enabled-reliability-beyond-predictive-maintenance.

8. Agro Informatics (2022). - Режим доступу: https://www.fao.org/digital-agriculture.
9. State of the Connected World 2023 Edition. Insight report. - World Economic Forum, 2023. - 49c. - Режим доступу: https://www3.weforum.org/docs/WEF_State_of_the_Connected_World_2023_Edition.pdf.
10. Measuring the Internet of Things. Report. - Organisation for Economic Co-operation and Development (OECD), 2023. - 130c. - Режим доступу: https://www.oecd.org/en/publications/measuring-the-internet-of-things_021333b7-en/full-report.html.
11. ENISA Threat Landscape 2023. Main Report. - European Union Agency for Cybersecurity (ENISA), 2023. - 161c. - Режим доступу: https://www.enisa.europa.eu/sites/default/files/publications/ENISA%20Threat%20Landscape%202023.pdf.
12. Cyber Security Report. – Check Point, 2023. - 109c. - Режим доступу: https://resources.finalsite.net/images/v1731496923/mvroporg/v48nezjdso7e8bmt0dxj/2023-cyber-security-report.pdf.
13. Microsoft Digital Defense Report 2024. – Microsoft, 2024. - 114c. - Режим доступу: https://www.microsoft.com/en-us/security/security-insider/threat-landscape/microsoft-digital-defense-report-2024.
14. Mahmoud I. & El-Gawad Amal. Revisiting Zero-Trust Security for Internet of Things. Sustainable Machine Intelligence Journal. - 2023. – 8c. - Режим доступу: https://www.researchgate.net/publication/377021858_Revisiting_Zero-Trust_Security_for_Internet_of_Things.
15. EDPB-EDPS Joint Opinion 02/2023 on the Proposal for a Regulation of the European Parliament and of the Council on the establishment of the

digital euro . – European Data Protection Board, 2023. – 34с. - Режим доступу: https://www.dataprotection.ro/servlet/ViewDocument?id=2620.

16. Jeffrey Voas. Networks of Things. - 2016. - 30с. - Режим доступу: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-183.pdf.
17. Aparcana-Tasayco, A.J., Deng, X. and Park, J.H. A systematic review of anomaly detection in IoT security: towards quantum machine learning approach. - EPJ Quantum Technology, 2025 – 39с. - Режим доступу: https://epjquantumtechnology.springeropen.com/articles/10.1140/epjqt/s40507-025-00414-6.
18. Kikissagbe, B.R. Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive Review / Kikissagbe, B.R. and Adda, M. // Electronics. - 2024. - 13(18). p.3601. Режим доступу: https://www.mdpi.com/2079-9292/13/18/3601.
19. Feisal S. Smart deep learning model for enhanced IoT intrusion detection. Report / Feisal S. // Scientific Reports. - 2025. – 23с. - Режим доступу: https://www.nature.com/articles/s41598-025-06363-5.
20. Chatterjee A. IoT anomaly detection methods and applications: A survey / A. Chatterjee, Bestoun S. Ahmed. // Internet of Things. – 2022. – 17с. - Режим доступу: https://www.sciencedirect.com/science/article/pii/S2542660522000622.
21. Gyamfi E. Intrusion Detection in Internet of Things Systems: A Review on Design Approaches Leveraging Multi-Access Edge Computing, Machine Learning, and Datasets / Eric Gyamfi, Anca Jurcut // Sensors. - 2022. - 22(10). 3744. – 33с. - Режим доступу: https://www.mdpi.com/1424-8220/22/10/3744.
22. Fog and Edge Computing: Principles and Paradigms / Editors: Buyya R., Srirama S. N. - John Wiley & Sons, Inc, 2019. - 490 с. - Режим доступу: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119525080?msockid=39299b5c739a6ce4021f899872f16d8e#aboutBook-pane.

23. Zero Trust: багаторівнева модель безпеки для сучасного бізнесу. – Smart, – 2022. - Режим доступу: https://cloud.smart-it.com/news-post/zero-trust-bagatorivneva-model-bezpeky-dlya-suchasnogo-biznesu/.
24. Kaviyazhiny C. Fog Computing Perspective: Technical Trends, Security Practices, and Recommendations / C. Kaviyazhiny, Bala, P.S. and A.S. Gowri // The Smart Cyber Ecosystem for Sustainable Development. – 2021. – P. 325–352. - Режим доступу: https://www.researchgate.net/figure/Characteristics-of-fog-computing_fig1_354541429.
25. Alotaibi B. A Stacked Deep Learning Approach for IoT Cyberattack Detection / Alotaibi, B. and Alotaibi, M. // Journal of Sensors, 2020. - 10c. - Режим доступу: https://onlinelibrary.wiley.com/doi/10.1155/2020/8828591.

ДОДАТОК А
КОД ФАЙЛУ MAIN.PY

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import joblib

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.decomposition import PCA

folder_path = "./data"
all_data = []
for filename in os.listdir(folder_path):
    if filename.endswith('.csv'):
        filepath = os.path.join(folder_path, filename)
        try:
            df = pd.read_csv(filepath, nrows=5000)
            print("Зчитано")
            label = os.path.splitext(filename)[0]
            df["label"] = label
            all_data.append(df)
        except Exception as e:
            print(f" Помилка у файлі {filename}: {e}")


df = pd.concat(all_data, ignore_index=True)
plt.figure(figsize=(14, 6))
sns.countplot(data=df, x="label",
order=df["label"].value_counts().index)
plt.xticks(rotation=90)
plt.title("Розподіл класів у датасеті")
plt.xlabel("Клас (тип атаки)")
plt.ylabel("Кількість зразків")
plt.tight_layout()
plt.savefig("class_distribution.png")
```

```
plt.close()

X = df.drop(columns=["label"], errors="ignore")
X = X.select_dtypes(include=["number"]).copy()


X = X.fillna(0)

features = X.columns

y = LabelEncoder().fit_transform(df["label"])


scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)


X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.3, random_state=42, stratify=y
)
      model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)
report = classification_report(y_test, y_pred, output_dict=True)
report_df = pd.DataFrame(report).transpose()
report_df.to_csv("classification_report.csv")

print("\n=== Звіт класифікації ===")
print(classification_report(y_test, y_pred))

cm = confusion_matrix(y_test, y_pred)
plt.figure(figsize=(12, 10))
sns.heatmap(cm, annot=False, fmt="d", cmap="Blues")
plt.title("Матриця плутанини")
plt.xlabel("Прогнозовано")
plt.ylabel("Фактично")
plt.tight_layout()
plt.savefig("confusion_matrix.png")
plt.close()
importances = model.feature_importances_
indices = np.argsort(importances)[-10:]
```

```
plt.figure(figsize=(10, 6))
plt.barh(range(len(indices)), importances[indices], align="center")
plt.yticks(range(len(indices)), [features[i] for i in indices])
plt.xlabel("Важливість ознаки")
plt.title("Топ-10 найважливіших ознак")
plt.tight_layout()
plt.savefig("feature_importance.png")
plt.close()

top_feature = features[indices[-1]]
plt.figure(figsize=(10, 6))
sns.histplot(data=df, x=top_feature, hue="label", kde=True, bins=50,
element="step", stat="density", common_norm=False)
plt.title(f"Розподіл ознаки: {top_feature}")
plt.xlabel("Значення")
plt.ylabel("Щільність")
plt.legend(title="Клас", loc="upper right")
plt.tight_layout()
plt.savefig("top_feature_distribution.png")
plt.close()
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)
plt.figure(figsize=(10, 6))
scatter = plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y, cmap="tab20", s=10,
alpha=0.7)
plt.title("PCA-візуалізація (2D)")
plt.xlabel("PCA 1")
plt.ylabel("PCA 2")
plt.colorbar(scatter, label="Клас")
plt.tight_layout()
plt.savefig("pca_visualization.png")
plt.close()
joblib.dump(model, "rf_nbaiot_model.joblib")
print("\n Успішно завершено! Збережено файли:")
print("- rf_nbaiot_model.joblib")
print("- classification_report.csv")
print("- confusion_matrix.png")
print("- feature_importance.png")
print("- class_distribution.png")
print("- top_feature_distribution.png")
print("- pca_visualization.png"
```

ДОДАТОК Б
ПРЕЗЕНТАЦІЯ

Рисунок Б.1 – Титульний слайд

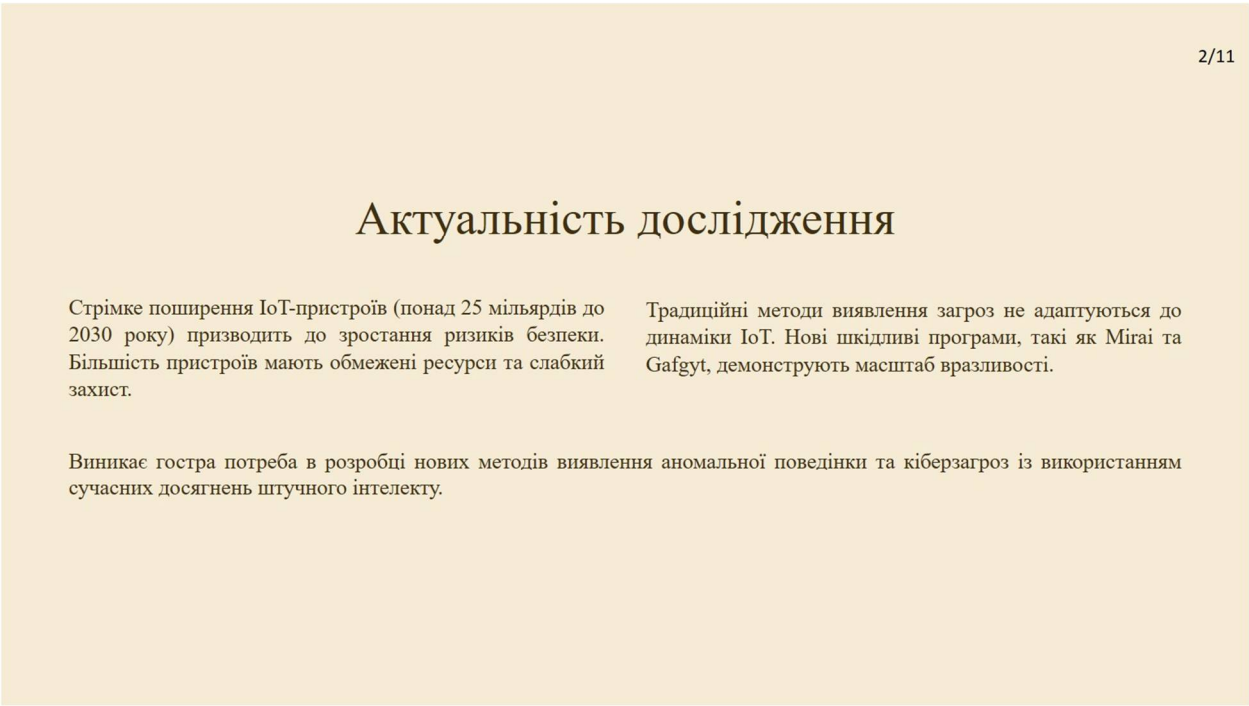

Рисунок Б.2 – Актуальність дослідження

Рисунок Б.3 – Мета та завдання роботи

Рисунок Б.4 – Теоретичний аналіз IoT і загроз

Рисунок Б.5 – Архітектура IoT

Рисунок Б.6 – Типові вразливості та атаки

Рисунок Б.7 – Методи виявлення атак

Рисунок Б.8 – Реалізація моделі

Рисунок Б.9 – Результати експерименту

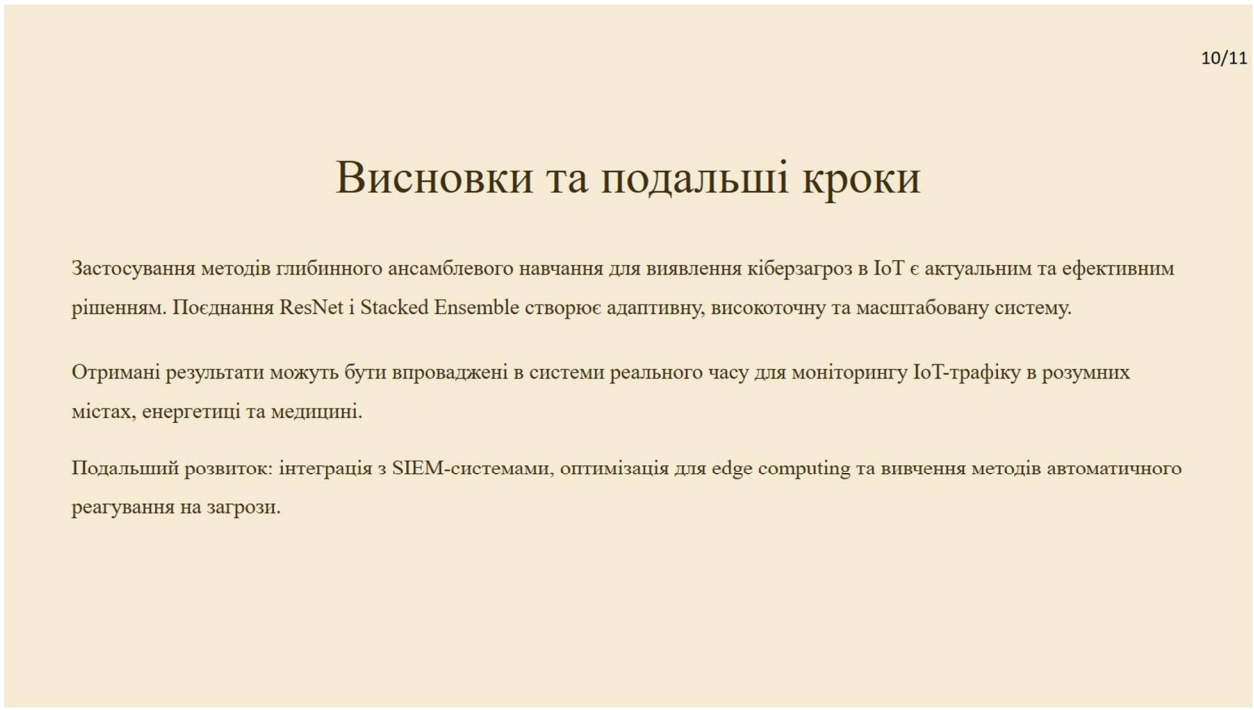

Рисунок Б.10 – Висновки та подальші кроки

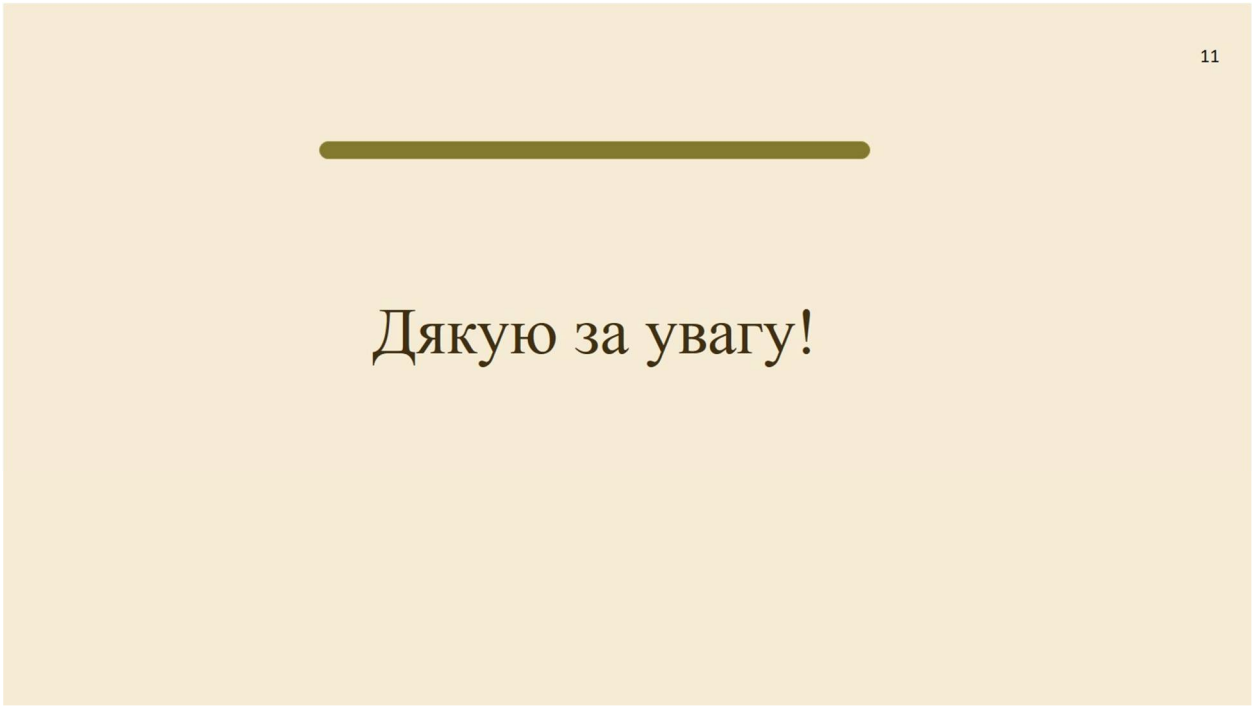

Рисунок Б.11 – Фінальний слайд