Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

Басистого Віталія Анатолійовича

на здобуття ступеня вищої освіти магістра

Агентний метод моніторингу мережевого трафіку IOT

Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека та захист інформації
Освітня програма Кібербезпека та захист інформації

Шифр КРМКБЗІ.240187.24.01.03 ПЗ

Виконав студент 2 курсу група КБЗІм-24-1 Віталій БАСИСТИЙ
Керівник канд. техн. наук, доцент Віктор ЧЕШУН
Нормоконтролер PhD, старший викладач Наталія ПЕТЛЯК

До захисту допускаю:
Завідувач кафедри кібербезпеки Юрій КЛЬОЦ
10 12 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Рівень вищої освіти Магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформації

Освітня програма Кібербезпека та захист інформації

ЗАТВЕРДЖУЮ

Завідувач кафедри кібербезпеки

Юрій КЛЬОЦ

1 09 2025 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Басистому Віталію Анатолійовичу

1 Тема роботи Агентний метод моніторингу мережевого трафіку IOT
Керівник роботи канд.техн.наук, доцент Віктор ЧЕШУН
Затверджено наказом ректора університету від 25 08 2025 № 65
2 Строк подання студентом кваліфікаційної роботи на кафедру 1.12.2025р.
3 Вихідні дані до роботи Розробити концепцію легковагового агентного методу моніторингу мережевого трафіку IoT, який враховує гетерогенність, обмежені ресурси та децентралізований характер IoT-систем; сформувати архітектуру мультиагентної моделі, визначити її функціональні компоненти, обґрунтувати вибір метрик аналізу трафіку та підготувати основу для подальшої практичної апробації методу.
4 Зміст пояснювальної записки (перелік питань, які потрібно розробити) Дослідження методів аналізу трафіку та вимог до систем моніторингу на пристроях з обмеженими ресурсами. Аналіз правових та регулятивних аспектів реалізації і моніторингу систем IoT. Агентний метод моніторингу мережевого трафіку IoT та засоби його реалізації. Архітектурні вимоги до розподіленої агентної системи моніторингу. Функціональний розподіл агентів для реалізації методу та механізми їх взаємодії. Фізичні топології розгортання системи і функції її компонентів. Проєктування логічної структури програмного агента для одноплатного комп'ютера. Вибір ознак мережевого трафіку для виявлення аномалій в IoT. Методики оцінки ефективності методу. Реалізація та експериментальна валідація методу і системи аналізу трафіку. Аналіз результатів та практичні рекомендації щодо впровадження.
5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень)
–

6 Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7 Дата видачі завдання 1 09 2025 р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
Ґрунтовне ознайомлення та дослідження предметної галузі	15.09.2025	Виконано
Визначення змісту, структури кваліфікаційної роботи	22.09.2025	Виконано
Підготовка першого розділу кваліфікаційної роботи	29.09.2025	Виконано
Підготовка другого розділу кваліфікаційної роботи	10.10.2025	Виконано
Підготовка третього розділу кваліфікаційної роботи	20.10.2025	Виконано
Підготовка статті/тези за темою кваліфікаційної роботи	4.11.2025	Виконано
Підготовка четвертого розділу кваліфікаційної роботи	17.11.2025	Виконано
Підготовка та оформлення ілюстративного матеріалу	24.11.2025	Виконано
Оформлення кваліфікаційної роботи	24.11.2025	Виконано
Попередній захист кваліфікаційної роботи	27.11.2025	Виконано
Захист кваліфікаційної роботи на засіданні ЕК	19.12.2025	Виконано

Студент _____ Віталій БАСИСТИЙ

Керівник кваліфікаційної роботи _____ Віктор ЧЕШУН

АНОТАЦІЯ

Тема кваліфікаційної роботи: Агентний метод моніторингу мережевого трафіку IOT

Автор роботи: Басистий Віталій Анатолійович

Керівник роботи: канд. техн. наук, доц. Чешун Віктор Миколайович

Загальний обсяг роботи: 129 сторінок, 7 рисунків, 6 таблиць, 2 додатки, 79 посилань.

Ключові слова: Інтернет речей, кібербезпека, виявлення вторгнень, MQTT, аналіз трафіку, одноплатні комп'ютери.

Кваліфікаційна робота присвячена розробці ресурсоефективного методу виявлення кіберзагроз у мережах Інтернету речей. В роботі обґрунтовано розподілену архітектуру системи моніторингу, що базується на децентралізованій взаємодії агентів через протокол MQTT, що дозволяє нівелювати апаратні обмеження периферійних пристроїв. Розроблено метод класифікації аномалій, який використовує вектор статистичних ознак та ентропійних показників потоку, відмовляючись від ресурсоємної глибокої інспекції пакетів.

В якості математичного апарату застосовано статистичний ансамблевий алгоритм, адаптований для архітектури ARM. Здійснено програмну реалізацію прототипу та проведено експериментальну валідацію на віртуалізованому полігоні. Результати підтвердили, що запропонований підхід забезпечує високу надійність виявлення загроз при кратному зменшенні споживання системних ресурсів порівняно з індустріальними аналогами, що робить його придатним для автономних систем захисту.

1.12.2025 _____

ANNOTATION

Theme of qualification work: Agent-based method for monitoring IoT network traffic

Author of the work: Basystyi Vitalii Anatoliyovych

Mentor: Ph.D. Cheshun Viktor Mykolaiovych

Total volume of work: 129 pages, 7 figures, 6 tables, 2 appendices, 79 links.

Keywords: Internet of Things, Cybersecurity, Intrusion Detection, MQTT, Statical analysis, Single Board Computers.

The master's thesis is devoted to the development of a resource-efficient method for detecting cyber threats in Internet of Things networks. The paper substantiates a distributed monitoring architecture based on decentralized agent interaction via the MQTT protocol, addressing the hardware limitations of edge devices. An anomaly classification method utilizing a vector of statistical features and flow entropy metrics without Deep Packet Inspection is developed.

A statistical ensemble algorithm adapted for ARM architecture is used as the decision-making core. A software prototype was implemented and validated on a virtualized testbed. Experimental results confirmed that the proposed approach ensures high reliability of threat detection with a multiple reduction in system resource consumption compared to industrial counterparts, making it suitable for autonomous protection systems.

1.12.2025 _____

ЗМІСТ

ВСТУП

Сучасний етап розвитку глобальних інформаційно-телекомунікаційних систем характеризується фундаментальною зміною парадигми взаємодії: переходом від комунікації «людина-людина» до концепції «Всеосяжного Інтернету» (IoE – Internet of Everything) [1], ключовим елементом якого є Інтернет речей (IoT). Експоненціальне зростання кількості підключених пристроїв, що за прогнозами аналітиків сягне десятків мільярдів одиниць вже у найближчі роки, докорінно змінює структуру мережевого трафіку та площину атак.

Стрімке і всеосяжне поширення Інтернету речей (IoT), особливо в критичних і побутових системах, створило принципово нові виклики для забезпечення кібербезпеки, що вимагає розробки інноваційних, гнучких та високоадаптивних механізмів моніторингу мережевого трафіку. Традиційні методи захисту периметра та централізовані системи виявлення вторгнень, що ефективно працюють у класичних корпоративних мережах, виявляються систематично неспроможними в архітектурі IoT. Ця неспроможність зумовлена низкою фундаментальних особливостей IoT-екосистем, зокрема їхньою гострою гетерогенністю, обмеженістю ресурсів кінцевих пристроїв і розподіленим характером функціонування.

Критичною проблемою стає забезпечення кіберстійкості інфраструктури IoT. Гетерогенність апаратних платформ, використання специфічних легковагових протоколів (MQTT, CoAP, ZigBee) та, що найважливіше, відсутність стандартизованих механізмів захисту на рівні прошивок (firmware), створюють унікальний ландшафт загроз. Традиційні методи кіберзахисту, орієнтовані на периметральну оборону корпоративних мереж із використанням високопродуктивних апаратних шлюзів, виявляються неефективними або економічно необґрунтованими в умовах децентралізованої архітектури IoT.

Також набуває актуальності проблема моніторингу мережевого трафіку на периферійному рівні. Обмежені обчислювальні ресурси IoT-шлюзів, які часто базуються на одноплатних комп'ютерах, унеможливлюють розгортання

класичних систем виявлення вторгнень (NIDS), таких як Snort або Suricata. Це вимагає пошуку нових, адаптивних методів аналізу, які б поєднували високу точність детектування аномалій із мінімальною ресурсоємністю.

Ключова проблема моніторингу IoT полягає у неможливості повноцінної централізації. Тисячі чи навіть мільйони пристроїв, від простих датчиків температури до складних камер і контролерів, постійно генерують трафік, який, через свою мінімальну пропускну здатність і високу частоту, може легко перевантажити єдину точку збору та аналізу даних. Крім того, більшість пристроїв IoT є ресурсно-обмеженими (resource-constrained), вони мають мінімальні обсяги оперативної пам'яті та низьку обчислювальну потужність. Це унеможливлює встановлення на них повноцінного антивірусного програмного забезпечення чи складних сигнатурних систем виявлення вторгнень. Такі пристрої, часто нездатні самостійно здійснювати складні криптографічні операції чи підтримувати постійний моніторинг власного стану, стають ідеальними об'єктами для компрометації та формування великомасштабних ботнетів, як це продемонструвала атака Mirai [2,3].

Сучасний ландшафт загроз IoT характеризується переходом від очевидних, високошвидкісних атак до скритих, низькошвидкісних та розподілених аномалій [4,5]. Зловмисники експлуатують слабкість автентифікації та відсутність уніфікованих протоколів, щоб ініціювати поведінкові аномалії в мережевому трафіку [6,7]. Це можуть бути атаки типу Low-Rate DDoS, що використовують легітимні протоколи (наприклад, невеликі, але постійні запити), або ж атаки на цілісність даних, коли пристрій починає надсилати невірні, але правдоподібні показники (наприклад, помилкові дані з датчика температури чи вологості). Традиційні сигнатурні системи є неефективними проти таких поведінкових атак, оскільки тут немає відомого зразка шкідливого коду, а атака виявляється лише шляхом глибокого аналізу контексту та аномалій трафіку. Потрібен метод, здатний здійснювати безперервний, глибокий аналіз інформаційного обміну на рівні кожного вузла.

Актуальність роботи зумовлена існуванням суперечності між необхідністю

забезпечення надійного кіберзахисту децентралізованих мереж IoT та непридатністю існуючих засобів захисту (NIDS) до експлуатації в умовах жорстких апаратних обмежень периферійних обчислювальних пристроїв IoT. Традиційні підходи вимагають надлишкових ресурсів, а існуючі легковагові рішення не забезпечують достатньої точності класифікації загроз.

Мета кваліфікаційної роботи полягає у підвищенні рівня захищеності мереж IoT шляхом розробки та програмної реалізації агентного методу моніторингу мережевого трафіку, який забезпечує високу точність виявлення атак при мінімальному споживанні системних ресурсів.

Об'єктом дослідження є процеси моніторингу та аналізу мережевого трафіку в розподілених системах Інтернету речей.

Предметом дослідження є методи та засоби агентного моніторингу мережевих аномалій в умовах обмежених обчислювальних ресурсів.

Щоб реалізувати програму досліджень необхідно:

– дослідити методи аналізу трафіку IoT та вимог до систем моніторингу на пристроях з обмеженими ресурсами;

– обґрунтувати концепцію агентного методу моніторингу мережевого трафіку IoT;

– визначити принципи організації системи реалізації агентного методу моніторингу мережевого трафіку IOT;

– розробити архітектуру розподіленої мультиагентної системи, що включає функціональні ролі агентів, та визначити протоколи їх ефективної взаємодії.

– обґрунтувати вибір набору статистичних та ентропійних ознак мережевого трафіку, достатніх для ідентифікації аномалій без застосування глибокої інспекції пакетів (DPI).

– розробити та адаптувати алгоритм класифікації, оптимізований для виконання на процесорній архітектурі ARM;

– створити програмний прототип системи та провести експериментальну валідацію його ефективності та продуктивності на платформі Raspberry Pi у порівнянні з існуючими аналогами (Suricata, Tshark).

В основі методів дослідження лежать базові положення інформаційної безпеки, теорії і практики виявлення аномалій мережевого трафіку, теорії реалізації мультиагентних систем.

Наукова новизна отриманих результатів:

Запропоновано комплексну архітектуру розподіленої системи моніторингу безпеки IoT, яка базується на кооперативній взаємодії чотирьох типів спеціалізованих автономних агентів (Сенсор, Аналітик, Координатор, Репортер), що функціонують на одноплатних комп'ютерах. На відміну від існуючих підходів, які намагаються адаптувати централізовані NIDS або розглядають лише окремі алгоритми виявлення аномалій, запропонований метод забезпечує масштабованість, відмовостійкість та зниження обчислювального навантаження шляхом дворівневої обробки: локального аналізу на рівні вузла та кореляції загроз на рівні всієї мережі.

Практична значимість отриманих результатів полягає у розробці рішень, що дозволяють створювати гнучкі, масштабовані та економічно ефективні системи захисту для широкого спектра IoT-застосувань, від "розумного будинку" до промислових мереж. Запропонована архітектура може бути реалізована на доступних апаратних платформах (наприклад, Raspberry Pi), що значно знижує поріг входу для впровадження комплексних рішень з кібербезпеки для малого та середнього бізнесу, а також для індивідуальних користувачів.

Результати роботи можуть бути використані при проєктуванні захищених IoT-систем та як основа для подальших наукових досліджень.

Публікації. За темою магістерської роботи опубліковано 1 статтю у фаховому науковому виданні, тези доповідей на 5-и Всеукраїнських та 2-х міжнародних науково-технічних і науково-практичних конференціях.

1 ДОСЛІДЖЕННЯ МЕТОДІВ АНАЛІЗУ ТРАФІКУ ТА ВИМОГ ДО СИСТЕМ МОНІТОРИНГУ НА ПРИСТРОЯХ З ОБМЕЖЕНИМИ РЕСУРСАМИ

1.1 Мережі інтернету речей як об’єкт атак і захисту

У сучасну епоху стрімкої цифровізації та технологічної конвергенції, Інтернет речей (IoT) перетворився з перспективної, науково-фантастичної концепції на один із ключових технологічних чинників, що формують майбутнє промисловості, міської інфраструктури та побутового середовища. Сучасні IoT-технології утворюють складну екосистему, що об’єднує мільярди фізичних об’єктів у єдину мережу через сенсори, мережеві протоколи, вбудовані обчислювальні можливості та аналітичні платформи. Ця інтеграція створює принципово нові, безпрецедентні можливості для збору, обробки й використання даних у реальному часі, відкриваючи нові горизонти для автоматизації та інтелектуального управління. Експоненційне зростання кількості підключених пристроїв, розширення можливостей обробки даних на периферії мережі та поглиблена інтеграція IoT з технологіями штучного інтелекту (ШІ) та машинного навчання (МН) створюють передумови для найглибших трансформацій у багатьох галузях економіки й суспільного життя (рисунок 1.1).

В економічних реаліях підприємства по всьому світу все частіше інтегрують IoT-рішення у свої операційні моделі. Це робиться для досягнення цілей оптимізації витрат, радикального покращення контролю якості продукції, впровадження віддаленого моніторингу стану обладнання та, що найважливіше, для реалізації прогнозного технічного обслуговування [8]. Такий проактивний підхід дозволяє суттєво зменшувати непередбачені простої виробництва, запобігати дороговартісним аваріям і будувати більш гнучкі та адаптивні бізнес-моделі, які можуть швидко реагувати на динамічні зміни ринку. IoT виступає невіддільною та ключовою складовою розвитку Індустрії 4.0, де відбувається повна інтеграція фізичних виробничих процесів з інформаційними технологіями, що веде до створення повністю автоматизованих, кібер-фізичних та самоадаптивних виробничих систем [9].

Рисунок 1.1 – Сфери застосування інтернету речей [10]

Концепція розумного дому (Smart Home), що є одним із найбільш відчутних проявів IoT у побутовій сфері, втілюється у складних екосистемах, які інтегрують численні взаємопов'язані пристрої. Сюди входять різноманітні сенсори, відеокамери, інтелектуальні контролери, побутова техніка та актуатори, що колективно забезпечують користувачам підвищений комфорт, енергоефективність та фізичну безпеку житла. Однак, саме ця гетерогенність та мережева зв'язаність створюють велику і потенційно вразливу поверхню атаки, яку можуть використовувати зловмисники для порушення не лише конфіденційності, але й фізичної безпеки самої системи. Критичною проблемою, що лежить в основі цих вразливостей, є архітектурна фрагментація.

Не менш важливим напрямом залишається глибоке проникнення IoT у побутову сферу. Сучасні смарт-пристрої для будинків, включаючи інтелектуальні термостати, системи освітлення, охоронні комплекси та побутову техніку, забезпечують не лише високий комфорт, але й значно підвищують енергоефективність житла, оптимізуючи споживання ресурсів. У межах концепції

Smart Cities (розумні міста) IoT-технології застосовуються для організації інтелектуального управління міським транспортом, оптимізації енергомереж, забезпечення екологічного моніторингу (вимірювання якості повітря та води) і підвищення громадської безпеки [11]. Ці міські рішення є життєво необхідними в контексті глобальних вимог до сталого розвитку та підвищення якості життя населення. Особливо значущу роль IoT відіграє у сфері охорони здоров'я (HealthCare IoT, IoMT), де системи дистанційного моніторингу пацієнтів, новітні носимі медичні пристрої та автоматизовані платформи збору даних допомагають медичному персоналу оперативно реагувати на найменші зміни стану здоров'я. Це особливо критично в умовах глобальних викликів, як-от пандемії, зростання хронічних захворювань та демографічне старіння населення [12, 13].

Наразі у сфері IoT доступна велика кількість різноманітних протоколів зв'язку, які функціонують на різних рівнях мережевої моделі. До них належать Wi-Fi, Bluetooth, ZigBee, Z-Wave, Thread та багато інших. Проте, на жаль, відсутні загальноприйняті, універсальні стандарти для мережевого об'єднання цих пристроїв [14, 15]. Це призводить до ситуації, коли пристрої від різних виробників часто не можуть коректно і безпечно взаємодіяти між собою без використання проміжних шлюзів (hubs) або пропрієтарних хмарних сервісів. Така фрагментація значно ускладнює централізоване управління безпекою та створення єдиної довірчої зони.

Крім того, технічні вимоги до мереж IoT суттєво відрізняються від традиційних IT-мереж. Використання технологій локальних мереж, таких як Ethernet або навіть високошвидкісний Wi-Fi, часто є неефективним через їхню властиву надмірність у сенсі пропускної здатності та високе енергоспоживання, що є неприйнятним для більшості IoT-сенсорів, які живляться від батарей. Технології, що застосовуються в системах IoT, мають відповідати суворим критеріям, серед яких низьке енергоспоживання, висока надійність передачі даних, забезпечення безпеки каналу, низька вартість самих пристроїв та простота фізичного розгортання (наприклад, без складного прокладання кабелів). Важливо, що більшість побутових застосувань IoT, таких як датчики температури, руху чи

стану вікон, не потребують високої швидкості передачі даних; їм достатньо надсилати невеликі порції інформації лише кілька разів на годину. Ця необхідність балансу між низьким енергоспоживанням та надійністю обмежує можливість впровадження складних криптографічних протоколів та механізмів безпеки безпосередньо на кінцевому пристрої.

Системи IoT також стикаються з широким спектром загроз безпеці, які є спільними для багатьох комп'ютерних мереж, але набувають особливої гостроти через специфіку IoT-пристроїв [16]. Недоліки в архітектурі та експлуатації створюють кілька критичних векторів атаки.

Першим і найпоширенішим вектором є використання спільних каналів зв'язку. Багато пристроїв можуть підключатися до мережі через загальнодоступні канали, такі як Wi-Fi або Bluetooth, які можуть бути скомпрометовані. Існуючі методи безпеки, зокрема, традиційні антивірусні програми або брандмауери, часто є недостатньо ефективними для повноцінного сканування або моніторингу підключених IoT-пристроїв на наявність вірусів або шкідливого програмного забезпечення [17, 18]. Це дозволяє зараженим пристроям, які можуть бути носіями шкідливого коду, проникати в систему, стаючи плацдармом для подальшого поширення загрози та компрометації інших, можливо, більш захищених компонентів IoT.

Друга група загроз стосується внутрішньої безпеки мережі та конфігурації. Внутрішні мережі IoT часто виявляються вразливими до атак через слабкі, стандартні або незмінені паролі та відсутність регулярних оновлень безпеки (прошивок). Для багатьох бюджетних або застарілих IoT-пристроїв виробники взагалі не випускають патчі. Традиційні засоби захисту, такі як мережеві брандмауери на периметрі, виявляються неефективними проти складних атак на горизонтальному рівні, особливо коли в атаці використовується взаємодія кількох пристроїв для створення каскадного ефекту [19, 20]. Атака, спрямована на один невеликий датчик, може стати точкою входу для контролю над усією системою опалення чи охорони.

Особливою категорією загроз, специфічних для IoT-систем, є проблеми,

пов'язані з надійністю даних та управлінням збоями.

По-перше, існуючі системи часто використовують прості, нестійкі алгоритми обробки даних, які не завжди здатні адекватно враховувати можливі збої або аномальні показники датчиків [21, 22]. Наприклад, якщо датчик температури через апаратну несправність або зовнішній вплив починає видавати неправильні (аномально високі чи низькі) дані, система може не розпізнати цю проблему як збій датчика. Натомість вона продовжує виконувати команди, ґрунтуючись на помилкових вхідних даних, що може призвести до небажаних або навіть небезпечних наслідків (наприклад, неконтрольованого перегріву чи охолодження приміщення, що становить загрозу майну або здоров'ю).

По-друге, коли окремі пристрої виходять з ладу або їхня прошивка пошкоджується, команди можуть надсилатися до системи в неправильному форматі або містити непередбачені помилки. Сучасні методи безпеки та системи діагностики часто не здатні ефективно діагностувати такі протокольні помилки та запобігти їхньому шкідливому впливу на загальну систему, що може призвести до нестабільної або навіть небезпечної роботи критичних підсистем, таких як блокування дверей або спрацювання пожежної сигналізації [23, 24].

По-третє, сам процес налаштування та конфігурації пристроїв Розумного дому, який часто залежить від попередньо визначених налаштувань виробника та індивідуальних уподобань користувача, є джерелом вразливостей. Помилки конфігурації, що виникають через недосвідченість користувача або недоліки інтерфейсу, можуть призвести до неправильної роботи системи, спричиняючи незручності або навіть пряму загрозу безпеці (наприклад, некоректне налаштування зон охорони). Сучасні системи безпеки не завжди здатні адекватно керувати такими складними, динамічними та мінливими конфігураціями [25, 26].

Нарешті, помилки в налаштуванні режимів роботи пристроїв (наприклад, встановлення конфліктуючих або нелогічних параметрів для систем опалення, вентиляції чи безпеки) створюють додаткові ризики. Оскільки поведінку системи у разі помилок конфігурації неможливо передбачити або контролювати традиційними засобами, це відкриває можливості для експлуатації цих

непередбачуваних станів зловмисниками [19, 27].

Для ефективного підвищення безпеки та надійності IoT-середовищ застосовується широкий спектр інноваційних та адаптованих до специфіки периферії рішень. Одним із перших кроків у забезпеченні безпеки будь-якої мережі є повне усвідомлення її складу. Інструменти уніфікованого виявлення активів відіграють тут вирішальну роль, допомагаючи виявляти та контролювати керовані та некеровані пристрої системи IoT у режимі реального часу [28]. Їхня основна функція полягає в автоматичній ідентифікації всіх підключених пристроїв у гетерогенній мережі, незалежно від їхнього типу, виробника чи операційної системи. Це дає IT та безпековим командам повну та актуальну інформацію про стан активів, що є критично важливим для управління ризиками. Ці інструменти інтегрують активне сканування, пасивний моніторинг мережевого трафіку, аналіз протоколів та інтеграцію з іншими системами, що дозволяє формувати детальні профілі пристроїв: визначати модель, версію прошивки, відкриті порти та рівень ризику.

Паралельно з інвентаризацією критичне значення набуває постійний моніторинг. Системи виявлення вторгнень (Intrusion Detection Systems – IDS) для IoT-середовищ, особливо адаптовані до промислових систем, забезпечують моніторинг, орієнтований на специфіку промислових протоколів (таких як Modbus, DNP3, OPC UA, BACnet та інші, що використовуються в ICS/SCADA системах) [28, 29]. На відміну від традиційних IT IDS, які аналізують загальні IP-протоколи, IoT-IDS враховують особливості форматів команд, дозволених операцій та типових сценаріїв взаємодії пристроїв у виробничих мережах. Ключова вимога до промислових IDS – пасивний моніторинг, оскільки будь-яке активне втручання чи навіть додаткове навантаження на критичне обладнання може порушити технологічні процеси. Тому такі системи зазвичай використовують дзеркальне копіювання трафіку (port mirroring) або TAP-пристрої для непомітного спостереження за мережею [30, 31].

В умовах, коли велика кількість IoT та операційних технологій пристроїв не підтримує регулярних оновлень програмного забезпечення або не має технічної

можливості для їх впровадження через вимоги сертифікації чи фізичні обмеження, застосовується віртуальне патчування [28]. Цей практичний метод захисту реалізується на мережевому рівні, де спеціалізовані пристрої (наприклад, міжмережеві екрани або IPS-системи) перехоплюють і фільтрують трафік, блокуючи спроби експлуатації відомих вразливостей ще до того, як вони досягнуть незахищеного пристрою. Це створює віртуальний захисний бар'єр. Рішення для віддаленого управління та моніторингу (Remote Management and Monitoring – RMM) додатково полегшують централізоване управління патчами та моніторинг стану цих розподілених пристроїв, що є критичним для довгострокової підтримки [14]. Цей підхід дозволяє суттєво продовжити безпечний життєвий цикл застарілого обладнання, мінімізуючи його вразливість перед новими експлойтами без необхідності фізичної заміни чи виведення з експлуатації, що особливо актуально для об’єктів критичної інфраструктури.

Інтегровані платформи безпеки IoT надають комплексний захист для пристроїв та систем на різних рівнях, а шлюзи безпеки IoT відіграють роль фортеці, забезпечуючи безпечне підключення пристроїв до хмарних сервісів та захищену обробку даних на вході в мережу [32]. Для забезпечення Root of Trust (кореня довіри) та виконання чутливих криптографічних операцій на найнижчому рівні використовуються апаратні механізми. Апаратні модулі безпеки (Hardware Security Modules – HSM) гарантують надійну криптографічну обробку та захищене зберігання ключів, роблячи їх недоступними навіть у разі фізичного зламу [32]. Подібну функцію виконують і безпечні елементи (Secure Elements – SE) та довірені середовища виконання (Trusted Execution Environments – TEEs), які створюють ізольовані, захищені області на процесорах для виконання критичних операцій з даними.

Новітні технології також вносять свій вклад у безпеку IoT. Технологія блокчейн пропонує децентралізований та захищений від підробок спосіб управління даними IoT, зокрема, для ведення незмінного журналу подій (immutable ledger), що значно підвищує цілісність та безпеку ланцюгів постачання даних [6,33,34]. Багатооператорські eSIM підвищують надійність та гнучкість

підключення IoT-пристроїв, дозволяючи автоматичне перемикання між операторами, що критично для забезпечення безперервності зв'язку в умовах мінливого географічного середовища [35]. Нарешті, Штучний інтелект та машинне навчання використовуються для проактивного виявлення шаблонів атак та захисту пристроїв IoT [34-38]. Ці системи здатні аналізувати величезні обсяги трафіку, виявляти аномалії, які не можуть бути ідентифіковані традиційними сигнатурними методами, та автоматично адаптувати захисні механізми до нових, раніше невідомих загроз.

Поряд із великим різноманіттям потужних технологічних рішень для забезпечення безпеки IoT, характерною рисою сучасного етапу їх розвитку є відсутність єдиних, універсальних підходів в їх реалізації та впровадженні. Різні галузі (промисловість, медицина, побут) використовують власні, часто несумісні стандарти. Для вирішення існуючих проблем та недоліків, подолання фрагментації ринку та забезпечення глобальної стійкості IoT-екосистеми необхідні комплексні, інтегровані рішення, які одночасно враховують аспекти безпеки, конфіденційності та операційної ефективності IoT. Ці рішення повинні бути уніфіковані та базуватися на узгоджених міжнародних нормативно-регулюючих документах, створюючи обов'язкові рамки для всіх виробників та операторів. Це забезпечить не лише технічну захищеність, але й суспільну довіру до технології, яка вже стала невід’ємною частиною нашого цифрового світу. З огляду на всі багатогранні загрози, від архітектурної фрагментації та неефективності традиційних антивірусних рішень до критичних вразливостей, пов'язаних із помилками в даних та конфігурації, є очевидною нагальна потреба в розробці та впровадженні більш комплексних, гнучких та адаптивних систем безпеки. Ці нові системи повинні не лише ефективно запобігати потенційним загрозам, але й вміти проактивно реагувати на непередбачувані збої, аномалії даних та помилки конфігурації, забезпечуючи справжню кіберстійкість систем IoT [39].

1.2 Дослідження методів виявлення аномалій у мережевому трафіку

Задача ідентифікації несанкціонованих впливів у комп'ютерних мережах є класом задач розпізнавання образів [40-42]. У контексті мережевої безпеки, "образом" є сукупність характеристик мережевого пакету або потоку даних [41].

Існуючі методи аналізу традиційно класифікують залежно від глибини інспекції даних та механізму прийняття рішень на три основні групи: сигнатурний аналіз, статистичний аналіз та методи алгоритмічної класифікації [43, 44].

Сигнатурний метод [45, 46] є історично першим і найбільш поширеним підходом у комерційних системах безпеки.

Метод базується на детермінованому порівнянні вмісту мережевих пакетів із попередньо сформованою базою даних відомих патернів атак (сигнатур). Процес аналізу передбачає повну реконструкцію мережевих сесій та глибоку інспекцію пакетів (DPI – Deep Packet Inspection). DPI передбачає аналіз не лише заголовків пакетів (L2-L4 моделі OSI), але й корисного навантаження (Payload, L7).

Для пошуку відповідностей використовуються алгоритми точного пошуку підрядка, такі як Ахо-Корасік (Aho-Corasick) [47] або Боєра-Мура (Boyer-Moore) [48].

Алгоритм Ахо-Корасік будує в пам'яті скінченний автомат (trie). Розмір цієї структури зростає лінійно з кількістю сигнатур. Сучасні бази правил (наприклад, Emerging Threats) містять десятки тисяч записів, що вимагає сотень мегабайт оперативної пам'яті лише для зберігання автомата.

Інша проблема - складність обчислень. Необхідність перевіряти кожен байт корисного навантаження створює значне навантаження на центральний процесор (CPU). На архітектурі ARM, яка є типовою для IoT-шлюзів (Raspberry Pi, Orange Pi), відсутність векторних інструкцій для прискорення обробки рядків призводить до критичного падіння пропускної здатності.

Критичні недоліки сигнатурного методу в контексті IoT:

– метод здатний виявляти лише відомі загрози. Він безсилий проти атак нульового дня (Zero-day) та поліморфних вірусів;

– зростання частки шифрованого трафіку робить DPI неефективним,

оскільки корисне навантаження є недоступним для аналізу без ресурсоємного дешифрування "на льоту"

Статистичні методи [49-51] базуються на гіпотезі, що трафік під час атаки має статистичні відхилення від профілю "нормальної" поведінки.

Система оперує метаданими мережевих потоків (Flow-based analysis), ігноруючи вміст пакетів. Аналізуються кількісні показники: інтенсивність пакетів (pps), обсяг байтів (bps), співвідношення вхідного/вихідного трафіку, розподіл протоколів.

Ключові метрики для виявлення аномалій в IoT:

– ентропія Шеннона;

– аналіз інтервалів.

Ентропія Шеннона використовується для кількісної оцінки невизначеності (хаотичності) розподілу атрибутів трафіку, таких як IP-адреси джерела або порти призначення.

В нормальному стані IoT-пристрої зазвичай комунікують з обмеженим набором серверів. Ентропія IP-адрес є низькою.

Під час DDoS-атаки типу IP Spoofing (підробка адреси відправника) спостерігається поява тисяч унікальних IP-адрес у короткий проміжок часу, що призводить до різкого зростання значення ентропії. Це дозволяє детектувати атаку без аналізу вмісту пакетів.

Аналіз інтервалів дозволяє виявляти автоматизовані інструменти атак. Легітимний трафік часто має стохастичний характер, тоді як ботнети генерують пакети з фіксованою або аномально малою затримкою для максимізації навантаження на жертву.

Перевагою статистичних методів є висока швидкість обробки, стійкість до шифрування, низькі вимоги до ресурсів.

Поведінковий аналіз на основі алгоритмічної класифікації [52-54] є еволюційним розвитком статистичного методу. Замість використання простих статичних порогів (thresholds), які часто дають хибні спрацювання, застосовуються складніші алгоритмічні структури прийняття рішень.

Система класифікує поточний стан мережі як "Норма" або "Атака" на основі набору вирішальних правил. Ці правила формуються на етапі попереднього налаштування (калібрування) системи на основі аналізу історичних даних.

Одним з найефективніших алгоритмічних підходів для систем реального часу є використання композицій (ансамблів) дерев рішень.

Дерево рішень – це ієрархічна структура, де кожен вузол представляє перевірку певного атрибута (наприклад, "Чи перевищує ентропія значення 3.5?"), а кожна гілка – результат перевірки. Листя дерева представляють кінцеве рішення (клас).

Композиційний підхід передбачає, що для підвищення точності використовується не одне дерево, а їх множина. Остаточне рішення приймається шляхом агрегації результатів (наприклад, мажоритарним голосуванням).

На відміну від складних математичних методів (як-от нейронні мережі або метод опорних векторів), які вимагають тисяч операцій множення матриць з плаваючою комою, дерева рішень трансформуються у прості логічні інструкції процесора CMP (порівняння) та JMP (перехід). Це забезпечує надзвичайно високу швидкість класифікації, що є критично важливим для архітектури ARM.

1.3 Огляд та аналіз існуючих рішень та їх обмежень при роботі на мікрокомп’ютерах

Ринок Open-Source рішень для моніторингу мережі є доволі насиченим. Проте більшість інструментів розроблялися для серверних платформ x86_64 з великим обсягом ресурсів. Проведемо аналіз їх застосовності на одноплатних комп'ютерах (Raspberry Pi 4/5).

Suricata [55] – це сучасна високопродуктивна система виявлення вторгнень, що є стандартом індустрії.

Suricata використовує механізм сигнатурного аналізу. Архітектура Suricata багатопотокова.

Suricata має такі обмеження для застосування, як значні обсяги оперативної пам'яті і надлишковість.

Для відслідковування стану TCP-з'єднань Suricata резервує значні обсяги оперативної пам'яті. На практиці, запуск Suricata з повним набором правил на пристрої з 1 ГБ RAM призводить до використання Swap-файлу та деградації продуктивності, або ж до примусового завершення процесу системою.

Більшість правил Suricata орієнтовані на корпоративні протоколи (SMB, FTP, SQL), які не використовуються в IoT-сегменті, але створюють навантаження на систему.

Zeek [56] – це фреймворк для аналізу мережевої активності, що фокусується на семантичному розборі протоколів.

Архітектура подієво-орієнтована. Користувач пише скрипти для обробки подій мережі.

Zeek має обмеження для застосування через аналіз стану та інтерпретацію скриптів.

Zeek зберігає стан кожного з'єднання в пам'яті. IoT-мережі часто генерують велику кількість короткоживучих з'єднань (наприклад, періодична відправка телеметрії через MQTT). Це призводить до швидкого переповнення таблиць станів в пам'яті. Обробка подій скриптовою мовою створює додаткове навантаження на CPU порівняно з компільованим кодом.

Tshark – консольна версія популярного аналізатора пакетів Wireshark.

Призначенням Tshark є глибока діагностика та форензіка (розслідування інцидентів), а не моніторинг в реальному часі.

Zeek має обмеження для застосування через архітектуру дисекторів та втрати пакетів.

Архітектура дисекторів є складною. Tshark будує детальне дерево розбору для кожного пакету. Це вкрай повільна операція.

Експериментальні дані свідчать, що Tshark починає втрачати пакети (packet drops) вже при навантаженні 50-100 Мбіт/с на процесорах класу Cortex-A72, що робить його непридатним для ролі постійного сенсора безпеки.

Таким чином, жодне з розглянутих рішень не є оптимальним для IoT-пристроїв. Необхідна розробка спеціалізованого, "легкого" програмного агента.

1.4 Аналіз правових та регулятивних аспектів реалізації і моніторингу систем IoT

Хоча кібербезпека IoT сьогодні закономірно розглядається як один із ключових викликів цифрового суспільства, що безпосередньо впливає на економічну стабільність, конфіденційність користувачів та національну безпеку, широке та системне нормативно-правове осмислення цієї тематики на міжнародному рівні сформувалося відносно недавно. Із вибуховим темпом зростання кількості мережевих сенсорів, побутових гаджетів, роботизованих систем, елементів критичної інфраструктури та автономних модулів, ризики, пов’язані з мінімальним рівнем захисту таких пристроїв, лише загострилися. Через це питання IoT-безпеки швидко перейшло з виключно технічної площини у стратегічну, політичну та навіть геополітичну. Держави швидко усвідомили, що слабка безпека мільйонів домашніх камер, дешевих датчиків або мережевих реєстраторів може слугувати плацдармом для масштабних кібератак і впливати на загальний рівень національної кіберстійкості. Тому виникла гостра потреба у розробленні цілісної нормативної та методичної бази, яка б установлювала зрозумілі, уніфіковані й обов’язкові вимоги до безпеки IoT на всіх етапах життєвого циклу – від початкового проєктування й виробництва до повноцінної експлуатації та остаточного списання.

Сьогодні на глобальному рівні вже існує масштабний масив регламентів, рекомендацій, технічних фреймворків і стандартів, що створюють міцний фундамент для безпечного функціонування мережевих пристроїв. Ці документи є результатом інтенсивної співпраці урядових структур, наукової спільноти, комерційних компаній і міжнародних організацій. Вони спрямовані на підвищення кіберстійкості систем, які інтегрують IoT-компоненти, і забезпечують

поступове впровадження принципів “безпеки за замовчуванням” (security by design). Такі стандарти також сприяють необхідній уніфікації підходів на світовому ринку: виробники в різних країнах змушені орієнтуватися на спільні вимоги, а це, своєю чергою, зменшує ризики фрагментації та появи “слабких ланок” у глобальних ланцюгах постачання. В умовах транснаціонального характеру IoT-екосистеми, де пристрої збираються з компонентів, виготовлених у десятках країн, саме такі узгоджені підходи стають визначальними, оскільки жодна держава не здатна окремо забезпечити надійний захист від загроз, які швидко поширюються через відсутність кордонів у кіберпросторі.

Важливо зазначити, що формування нормативної бази IoT-безпеки не було одноразовим проактивним процесом. Воно розвивалося поступово й нерідко реактивно після масштабних і резонансних інцидентів. Найбільш значущим із них стала атака ботнету Mirai у 2016 році [2], яка скомпрометувала сотні тисяч погано захищених IoT-пристроїв. Ботнет Mirai (з японської «майбутнє») був реалізований як самопоширюване шкідливе програмне забезпечення, яке постійно сканувало інтернет у пошуках IoT-пристроїв із відкритими портами та заводськими паролями за замовчуванням (наприклад, admin/123456). Здобувши доступ, він перетворював пристрій на зомбі-вузол, не даючи жодних візуальних ознак зламу. Цей інцидент, який призвів до проведення безпрецедентної за масштабом DDoS-атаки на ключові DNS-провайдери, став справжнім «дзвінком пробудження» для урядів. Саме Mirai наочно продемонстрував, що слабка безпека домашнього гаджета може створювати системний ризик для критичної інфраструктури. Цей інцидент став одним із рушійних факторів для багатьох держав, які зрозуміли, що без чітких мінімальних стандартів ринок продовжуватиме продукувати пристрої з критичними вразливостями. Згодом до цього додалися інші інциденти: масові злами медичних пристроїв, проникнення у транспортні системи, компрометація промислових сенсорів, які продемонстрували, що слабкі IoT-компоненти можуть створити каскадні ефекти для всієї критичної інфраструктури. Загроза ботнету Mirai не усунута до сьогодні, у 2023 році зявилась інформація про появу нової потужнішої версії Mirai [3] і це тільки один і численних інструментів атаки на

системи IoT.

Одним із найвпливовіших міжнародних документів, який фактично визначив старт нової епохи у формуванні загальнодержавних підходів до захисту побутових IoT-пристроїв, стали Рекомендації щодо безпеки споживчих IoT-пристроїв – Code of Practice for Consumer IoT Security [57]. Їх було вперше опубліковано Міністерством цифрових технологій, медіа, культури та спорту Великої Британії у 2018 році. Документ став серйозним політичним сигналом, орієнтованим на перехід від добровільного саморегулювання ринку до формування чітких мінімальних вимог щодо безпеки для всіх виробників та постачальників споживчих IoT-рішень. Саме британська ініціатива створила основу для появи подальших європейських стандартів і надала чітку структуру підходам до безпечного життєвого циклу пристрою.

Початкова редакція кодексу складалася з 13 основних принципів безпеки, які охоплювали весь життєвий цикл пристрою – від етапу ідеї та технічного проєктування до утилізації. У документі наголошувалося на важливості унікальних облікових даних, недопустимості використання стандартних заводських паролів, забезпеченні прозорості у питаннях оновлення програмного забезпечення (ПЗ), наявності процедур повідомлення про вразливості, необхідності підтримки криптографічних протоколів, а також забезпеченні захисту конфіденційних даних відповідно до вимог GDPR. Крім того, підкреслювалася важливість мінімізації поверхні атаки, належного журналювання, контролю фізичного доступу та гарантування можливості безпечного відновлення системи після збоїв.

Серед ключових вимог кодексу виділялися такі:

– посилена автентифікація – заборона універсальних стандартних паролів та обов’язкове їхнє оновлення користувачем під час першої конфігурації, що є найпростішим, але найбільш критичним заходом проти атак, подібних до Mirai;

– управління вразливостями – необхідність мати відкритий канал для відповідальних дослідників безпеки та чітку, публічну політику реагування на виявлені вразливості (Vulnerability Disclosure Policy), що сприяє добровільній

взаємодії між виробниками та спільнотою безпеки;

– безпечні оновлення – підтримка механізмів безпечного завантаження ПЗ, перевірка цифрового підпису оновлень та гарантування певного періоду технічної підтримки;

– захист персональних даних відповідно до основ GDPR;

– захист цілісності системи, включно з контролем конфігурації та шифруванням даних під час передачі;

– безпечне виведення з експлуатації, що передбачає надійне видалення персональної інформації;

– операційна стійкість, журналювання діяльності та здатність до відновлення;

– користувацька інформованість – надання прозорої технічної документації й опису політик безпеки.

Ці параметри стали одним із перших глобальних прецедентів формування стандартів безпечного IoT і значною мірою вплинули на наступні регуляції у Європі та поза її межами. Зокрема, саме вони лягли в основу специфікації ETSI EN 303 645 [58], яка згодом стала фактичним галузевим стандартом для побутових IoT пристроїв.

Водночас у США важливим кроком стало ухвалення федерального закону “IoT Cybersecurity Improvement Act of 2020” [59]. Цей закон є першою в історії США спробою встановити обов’язкові вимоги до IoT-пристроїв, які використовуються у федеральних установах. Він став прямою відповіддю на численні інциденти, пов’язані з використанням у державному секторі дешевих і слабко захищених модулів, які зловмисники нерідко використовували як точки доступу до внутрішніх мереж. Законодавці визначили, що будь-які пристрої, які закуповуються державними структурами, повинні відповідати рекомендаціям та стандартам Національного інституту стандартів і технологій (NIST). Це запровадило принцип “безпеки за контрактом” (Security by Contract). Тепер вимоги до безпеки інтегруються безпосередньо у контракти на державні закупівлі, що створює потужний економічний стимул для виробників підвищувати рівень

захисту своєї продукції.

У межах цього закону NIST розробив ряд нормативних документів, що складають основу для побудови безпечних IoT-систем, серед яких ключове місце займає серія спеціальних публікацій NIST SP 800-213 [60]. Вона деталізує вимоги до інтеграції IoT у державні інформаційні системи, визначає категорії ризиків, моделі оцінки, методи перевірки безпеки та перелік контрольних механізмів. Цей комплекс документів суттєво вплинув на стандартизацію IoT у критично важливих секторах, оскільки застосовується не лише в урядових структурах, а й у підрядних організаціях, які є частиною федерального ланцюга постачання.

Суттєвим елементом нормативної бази NIST є фреймворк "IoT Device Cybersecurity Guidance for the Federal Government" [61]. Документ визначає мінімальний набір вимог, що має бути виконаний як виробниками, так і державними агентствами, які експлуатують IoT. Його поява сприяла створенню більш прозорої та прогнозованої системи закупівель, де критерії безпеки відіграють ключову роль, вимагаючи від постачальників чіткого документування безпекових можливостей їхніх пристроїв.

Особливу роль у формуванні підходів до IoT-кібербезпеки відіграла серія NISTIR 8259 [62]. Це фундаментальний комплекс рекомендацій, призначений безпосередньо для виробників IoT-пристроїв, який містить базові технічні та організаційні вимоги. Серія спрямована на усунення прогалин у ланцюгах постачання (Supply Chain Security), що є однією з найбільших загроз IoT. Шляхом встановлення мінімальних вимог до проєктування та виробництва, NIST прагне гарантувати, що вразливості не будуть інтегровані в пристрої ще на етапі фабричного виробництва. Серія складається з трьох взаємодоповнювальних документів, що покривають архітектурні, технічні та нефункціональні аспекти:

– NISTIR 8259 – окреслює базові безпекові процеси, яких мають дотримуватися виробники [63];

– NISTIR 8259A – формує технічний мінімум функцій кібербезпеки (керування ідентифікацією, шифруванням, оновленнями, вимоги до апаратних модулів безпеки, таких як Trusted Platform Modules – TPM, та механізмів Secure Boot) [64];

– NISTIR 8259B (проєкт) – описує нефункціональні, але важливі умови експлуатації: створення повної технічної документації, інструкцій для користувачів, політик сервісної підтримки та управління життєвим циклом пристрою, включаючи політику end-of-life [65].

Цей тріадичний підхід гарантує, що безпека IoT розглядається не лише як набір технічних функцій, але й як повноцінна програма управління безпекою, охоплюючи всі зацікавлені сторони.

Паралельно з урядовими ініціативами різні міжнародні організації створили власні, деталізовані фреймворки. Наприклад, Cloud Security Alliance розробила CSA IoT Security Controls Framework [66] як структуровану систему безпекових контролів, яка охоплює мультидоменну архітектуру IoT. Інструмент охоплює хмарні сервіси, шлюзи, мережі, сенсори та додатки і містить вимоги щодо сегментації трафіку, ізоляції критичних компонентів, управління ключами, захисту API, безпечного оновлення та постійного моніторингу. Його перевага полягає у можливості використання для комплексного аудиту та оцінки ризиків у великих, корпоративних IoT-екосистемах.

Ще один вагомий інструмент – Compliance Framework від IoT Security Foundation (IoTSF) [67]. Він орієнтований на виробників, інтеграторів та постачальників послуг, які прагнуть упровадити принципи “secure by design” на практиці. Рамка охоплює сотні контрольних пунктів, включаючи захист ПЗ, безпечний ланцюг постачання, тестування на проникнення, аудит конфігурацій та вимоги до комунікаційних протоколів. На відміну від CSA, IoTSF більше фокусується на процесах розробки та відповідності.

На європейському рівні центральним стандартом для споживчих IoT стала специфікація ETSI EN 303 645 [58]. Вона встановлює вимоги до мінімального рівня кіберзахисту для широкого спектра побутових пристроїв – від дитячих smart-іграшок до домашніх асистентів. Стандарт також передбачає оцінку ризиків, детальні правила щодо оновлень, вимоги до журналювання та обмеження доступу.

Однак найбільш значущим кроком у європейському регулюванні є Cyber Resilience Act (CRA). Цей законодавчий акт, ухвалений ЄС, знаменує перехід від

добровільних рекомендацій до обов'язкових правових норм для всіх виробників апаратного та програмного забезпечення, що продається на європейському ринку. CRA вимагатиме від виробників IoT:

– декларації відповідності – документального підтвердження відповідності пристрою встановленим базовим вимогам безпеки;

– усунення вразливостей – встановлення жорстких часових рамок для усунення виявлених вразливостей після випуску пристрою;

– маркування CE – наявність обов'язкового маркування CE, яке включатиме відповідність вимогам кібербезпеки;

– термін підтримки – чітке визначення та дотримання мінімального терміну технічної підтримки та надання оновлень безпеки.

Порушення CRA передбачає значні штрафи, що можуть сягати мільйонів євро або відсотка від світового річного обороту компанії, що робить цей закон ключовим драйвером глобальних змін у підходах до IoT-безпеки. CRA закріплює вимогу, що безпека повинна бути інтегрована у продукт на етапі проєктування (security by design), а не додана пізніше.

Вагому роль у міжнародних підходах до кібербезпеки відіграють стандарти сімейства ISO/IEC, які забезпечують методологічну основу для побудови надійних систем управління безпекою та конфіденційністю.

ISO/IEC 27001 [68] – міжнародний стандарт для систем управління інформаційною безпекою (СУІБ). У сфері IoT він є критично важливим для промислового IoT та операційних технологій, оскільки вимагає формальної оцінки ризиків (Risk Assessment). СУІБ вимагає ідентифікації всіх IoT-компонентів як активів і впровадження організаційних та технічних контролів (наприклад, сувора сегментація мережі IoT від корпоративної, використання промислових брандмауерів) для забезпечення конфіденційності, цілісності та доступності даних. Саме ISO 27001 надає структурний підхід до управління інцидентами безпеки, що є життєво необхідним для IoT-систем.

ISO/IEC 27701 [69] – розширення до 27001, що фокусується на конфіденційності та захисті персональних даних. Цей стандарт набуває особливої

важливості у медичному IoT та системах Розумного дому, де обробляються чутливі дані користувачів, і його використання допомагає організаціям дотримуватися таких регуляцій, як GDPR. Він вимагає реалізації принципів privacy by default та privacy by design.

Особливо важливими ці стандарти є для IoT у критичних секторах – енергетиці, транспорті, охороні здоров'я, промисловості, оскільки порушення їхніх вимог може призвести до катастрофічних наслідків. Імплементація ISO/IEC 27001/27701 дозволяє організаціям формалізувати підхід до оцінки ризиків і гарантувати, що технічні та організаційні заходи захисту відповідають міжнародним вимогам, а також забезпечити можливість сертифікації цих систем.

Загалом нормативна база кібербезпеки IoT продовжує розвиватися комплексно та системно, постійно адаптуючись до нових технологічних викликів. Кожного року з'являються нові уточнення, методики тестування, механізми сертифікації та додаткові рекомендації, які деталізують вимоги до безпеки як на рівні пристроїв, так і на рівні хмарних платформ та мобільних застосунків. Інтеграція принципів «security by design» і «privacy by default» стає глобальною нормою. Глобальний ринок IoT поступово рухається до того, що безпечність пристрою перестає бути конкурентною перевагою – вона стає базовою вимогою, без виконання якої вихід на ринок стає неможливим. Поєднання обов'язкових законів (CRA, IoT Cybersecurity Improvement Act) та деталізованих технічних стандартів (NIST, ETSI) створює необхідний тиск, який стимулює всю IoT-екосистему до досягнення справжньої кіберстійкості.

1.5 Постановка задачі

Проведений аналіз дозволяє формалізувати задачу дослідження.

Існує суперечність між необхідністю забезпечення надійного кіберзахисту децентралізованих мереж Інтернету речей та непридатністю існуючих засобів захисту до експлуатації в умовах жорстких апаратних обмежень периферійних

обчислювальних пристроїв IoT. Традиційні підходи вимагають надлишкових ресурсів, а існуючі легковагові рішення не забезпечують достатньої точності класифікації загроз.

За мету кваліфікаційної роботи визначаємо підвищення рівня захищеності мереж IoT шляхом розробки та програмної реалізації агентного методу моніторингу мережевого трафіку, який забезпечує високу точність виявлення атак при мінімальному споживанні системних ресурсів.

З урахуванням визначеної мети уточнимо об’єкт дослідження – процеси моніторингу та аналізу мережевого трафіку в розподілених системах Інтернету речей.

Предметом дослідження є методи та засоби агентного моніторингу мережевих аномалій в умовах обмежених обчислювальних ресурсів.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

– обґрунтувати концепцію агентного методу моніторингу мережевого трафіку IoT;

– визначити принципи організації системи реалізації агентного методу моніторингу мережевого трафіку IOT;

– розробити архітектуру розподіленої мультиагентної системи, що включає функціональні ролі агентів, та визначити протоколи їх ефективної взаємодії.

– обґрунтувати вибір набору статистичних та ентропійних ознак мережевого трафіку, достатніх для ідентифікації аномалій без застосування глибокої інспекції пакетів (DPI).

– розробити та адаптувати алгоритм класифікації, оптимізований для виконання на процесорній архітектурі ARM;

– створити програмний прототип системи та провести експериментальну валідацію його ефективності та продуктивності на платформі Raspberry Pi у порівнянні з існуючими аналогами (Suricata, Tshark).

2 АГЕНТНИЙ МЕТОД МОНІТОРИНГУ МЕРЕЖЕВОГО ТРАФІКУ ІОТ ТА ЗАСОБИ ЙОГО РЕАЛІЗАЦІЇ

2.1 Архітектурні вимоги до розподіленої агентної системи моніторингу

Розробка архітектури є основним етапом проєктування системи, тому що в проєктуючи систему, потрібно визначити масштабованість, відмовостійкість та швидкодію. Це дозволить зрозуміти, можливі ризики ще на початку проєктування.

Головними проблемами, які вимагають уваги є великі об’єми сирого трафіку, який не фільтрується, затримка на реагування, аналіз та відповідь та можливі мережеві втрати на високих швидкостях передачі.

Щоб уникнути цих уникнути зазначених проблем та забезпечити стабільну роботу системи було реалізовано архітектуру мультиагентних систем (MAS - Multi-agent system). Цей вибір зумовлений необхідністю комплексного рішення проблеми з недопущенням обробки даних на одному пристрої, а децентралізувати систему на окремі пристрої, кожний з яких відповідальний за свою задачу.

Архітектура системи базується на логічному (взаємодія агентів між собою) та фізичному рівні (топологія мережі), щоб визначити процес прийняття рішень та процес транспортування даних.

Першочергово треба визначити логічну архітектуру з орієнтацією на мультагентний підхід.

Логічна архітектура базується на взаємодії автономних програмних модулів між собою. На відміну від монолітних програмних рішень, де весь процес обробляється на одному пристрої, в цьому випадку, є можливість робити процес паралельним на кожному підключеному пристрої. Така децентралізація дозволяє виявляти кіберзагрози, їх класифікувати та унеможливлювати їхнє забруднення трафіку не обмежуючись одним пристроєм та його обчислювальними ресурсами.

Першою та визначальною властивістю є автономність, яка надає агентам здатність виконувати функції моніторингу та приймати рішення щодо блокування загроз без постійного керування з боку людини або центрального сервера. Ця

властивість є критичною для одноплатних комп'ютерів, які можуть працювати в умовах нестабільного з'єднання.

Другим важливим аспектом є швидкодія на загрози, що визначає спроможність агентів реагувати на появу аномалій у реальному часі

Це дозволяє не очікувати результатів аналізу на віддаленому сервері, а напряму аналізувати аномалію, приймати окремі рішення, для зменшення часу перебування системи у вразливому стані.

Третьою властивістю виступає проактивність, яка надає системі можливість визначати аномалію, до моменту виникнення інциденту, заблокувавши її.

Четвертою властивістю системи є комунікативність. Забезпечення агентів взаємодією з іншими вузлами мережі для координації спільних дій, дозволяє формувати колективне рішення проблеми. Фіксація проблеми на одному вузлі, миттєво ініціює захист, повідомляючи всю мережу на можливу загрозу.

Інтеграція зазначених властивостей дозволяє досягти ключової переваги розробленої системи – архітектурного ізоморфізму. Сутність цього підходу полягає у повній відповідності логічної структури захисту фізичній топології мережі IoT. Це створює заміну класичної обробки даних на віддаленому сервері на обробку даних в місці виникнення аномалії. Також це дозволяє поєднувати систему з подібними реалізації, збільшуючи масштаб та наявні ресурси на обробку даних без ускладнень і можливих проблем сумісності.

Практична реалізація такої архітектури здійснюється через дворівневу організацію логіки. На локальному рівні, що розгортається на обчислювальних потужностях кожного окремого одноплатного комп’ютера, забезпечується повна автономність процесів моніторингу.

Це гарантує безперервність аналізу та миттєве реагування на інциденти навіть за умов втрати зв'язку із зовнішнім світом.

Паралельно з цим функціонує кооперативний (мережевий) рівень, де агенти різних вузлів об'єднуються у єдину топологію, формуючи розподілену мережу. Взаємодія на цьому рівні передбачає обмін не об'ємним "сирим" трафіком, а компактними метаданими про виявлені аномалії через протоколи, зокрема MQTT.

Такий механізм дозволяє валідувати загрози та координувати захист у масштабах всієї мережі, уникаючи при цьому перевантаження каналів зв'язку та центральних вузлів системи.

2.2 Функціональний розподіл агентів для реалізації методу та механізми їх взаємодії

Архітектурна ефективність запропонованої мультиагентної системи ґрунтується на парадигмі функціональної декомпозиції, яка передбачає розбиття складної задачі моніторингу на ряд ізольованих, але взаємопов'язаних процесів.

Враховуючи обмеження апаратних ресурсів одноплатних комп’ютерів, як обсяг оперативної пам'яті та обчислювальної потужності процесора, виконання всіх етапів обробки трафіку в межах одного процесу є неефективним. Така централізація на рівні вузла викликала, підвищення затримки та ризику втрати мережевих пакетів під час пікових навантажень.

Тому пропонується здійснити розподіл обов’язків між чотирма спеціалізованими класами агентів, кожен з яких оптимізовано під специфіку роботи агента. Пропонований розподіл функціональних ролей та рівнів підпорядкування компонентів відображено у розробленій ієрархії агентів (таблиця 2.1).

Першим агентом обробки даних, є Агент-Сенсор (Sensor Agent), основною задачею якого, є захоплення "сирого" трафіку та його первинна обробка. Замість ресурсоємного глибокого аналізу повного вмісту пакетів, агент фокусується на процедурі вилучення основних ознак. Він виокремлює лише ключові статистичні характеристики потоку, такі як IP-адреси, порти, TCP-прапори (SYN, FIN, RST), розмір пакету та часові мітки активності. Паралельно здійснюється процес нормалізації даних, під час якого агент виконує фільтрацію шумів, відсіюючи нерелевантний трафік, наприклад, широкомовні ARP-запити, що дозволяє суттєво зменшити обсяг оброблюваної інформації.

Таблиця 2.1. – Функціональні ролі та обов'язки агентів.

Тип агента	Призначення	Вхідні дані	Основні функції	Вихідні дані
Агент-Сенсор	Збір та первинна обробка трафіку	Мережеві пакети	Захоплення пакетів, екстракція метаданих та фільтрація шуму	Нормалізовані вектори ознак
Агент-Аналітик	Локальне виявлення аномалій	Вектори ознак від Сенсора	Сигнатурний аналіз, контроль метрик, порівняння зі списками	Локальні події безпеки
Агент-Координатор	Забезпечення колективного аналізу	Локальні події та запити від сусідніх вузлів	Верифікація загроз, кореляція подій від різних джерел, синхронізація списків блокування	Підтвердження глобального інциденту
Агент-Репортер	Взаємодія з адміністратором/SIEM	Підтверджені інциденти від Координатора	Агрегація повідомлень, форматування звіту, буферизація та відправка	Фінальні звіти про інциденти

Результатом роботи сенсора є створення компактного потоку векторних ознак, який передається далі з мінімальними витратами ресурсів операційної системи на копіювання буферів пам'яті.

Отриманий нормалізований потік даних надходить до Агента-Аналітика, який реалізує аналіз на локальному рівні.

Цей агент виконує обробку векторів ознак з метою виявлення певного шаблону вторгнень. Процес виявлення базується на гібридному підході, що поєднує сигнатурний аналіз для ідентифікації відомих атак шляхом порівняння з

локальною базою сигнатур. Додатково застосовується механізм контролю доступу, який звіряє параметри трафіку з динамічними "білими" та "чорними" списками.

Така спеціалізація дозволяє ізолювати важкі аналітичні алгоритми від процесу захоплення пакетів, гарантуючи, що складні обчислення не сповільнять роботу. Після класифікації активності сформований опис інциденту передається на наступний рівень ієрархії.

Забезпечення узгодженості дій та реалізацію кооперативного рівня покладено на Агента-Координатора.

Агент виступає ключовим елементом у механізмі прийняття рішень, відповідаючи за агрегацію результатів аналізу та комунікацію з іншими вузлами мережі.

Отримавши дані про підозрілу активність, координатор ініціює процедуру перехресної перевірки, надсилаючи запити сусіднім агентам для підтвердження наявності загрози в їхніх сегментах мережі. Це дозволяє підтвердити інцидент, знизити рівень хибних спрацювань та класифікувати подію, як підтверджену атаку.

Окрім того, координатор відповідає за синхронізацію даних системи, оновлюючи фільтрацію на основі даних від інших вузлів, що забезпечує превентивний захист ще не атакованих сегментів мережі.

Останнім в архітектурі мультиганетної системи є Агент-Репортер, який відповідає за інтерфейс взаємодії із зовнішніми системами моніторингу.

Основна задача цього агента полягає в агрегації однотипних подій в єдині пакети повідомлень, що дозволяє уникнути перевантаження каналу керування у разі масованої атаки.

Репортер виконує стандартизацію даних, конвертуючи внутрішні об'єкти системи у загальноприйняті формати, такі як JSON, роблячи їх зрозумілими для зовнішніх SIEM-систем.

В результаті роботи агента, система не лише фіксує загрози, а й забезпечує інтеграцію з корпоративною інфраструктурою безпеки, передаючи лише фінальні,

підтверджені звіти, а не масиви необроблених даних.

Взаємодія між програмними агентами системи, реалізується на базі протоколу MQTT. Даний протокол обрано як галузевий стандарт IoT та здатності ефективно функціонувати в мережах з обмеженою пропускною здатністю.

Використання асинхронної моделі обміну повідомленнями дозволяє мінімізувати накладні витрати при передачі керуючих команд та телеметричних даних. Така архітектура забезпечує необхідну масштабованість системи та стійкість до затримок у каналах зв'язку.

2.3 Фізичні топології розгортання системи і функції її компонентів

Ключовим критерієм розроблюваної мультиагентної системи є її архітектурна гнучкість та здатність до адаптації в умовах зміни апаратних пристроїв.

Система проєктувалася таким чином, щоб мінімізувати залежності від конкретної апаратної конфігурації, що дозволяє уникнути обмежень при масштабуванні інфраструктури в майбутньому.

Для перевірки працездатності алгоритмів та аналізу мережевої взаємодії агентів було розроблено три базові топології розгортання, моделювання яких здійснювалося у середовищі Cisco Packet Tracer.

Вибір даного інструментарію обумовлений необхідністю точної емуляції роботи стека протоколів TCP/IP та специфічних служб IoT без залучення вартісного фізичного обладнання на етапі раннього проєктування.

Архітектурну основу периферійного рівня складають обчислювальні вузли (у симуляції позначені як PC-PT), які емулюють роботу реальних одноплатних комп’ютерів.

Ці пристрої виступають безпосередніми носіями агентів, виконуючи локальні обчислення в умовах обмежених ресурсів. На кожному вузлі розгортається повний функціональний стек, що включає: Сенсора для захоплення

трафіку, Аналітика для первинної обробки даних, Координатора для комунікації та Репортера для звітування.

Важливою особливістю конфігурації є те, що попри використання протоколу DHCP для автоматизації, кожному вузлу призначено статичну прив'язку для забезпечення стабільності сервісів. Вузли функціонують, одночасно генеруючи фоновий трафік та здійснюють пасивний моніторинг для виявлення аномалій.

Централізація управління покладено на виділений серверний вузол (Server-PT), який виконує роль ядра системи. На ньому розгорнуто критично важливі служби, першою з яких є MQTT-брокер, що реалізує транспортний рівень та гарантує маршрутизацію повідомлень між агентами навіть за умов нестабільного зв'язку.

Другою функцією є збір подій, де сервер виступає централізованим сховищем, приймаючи структуровані JSON-звіти про інциденти для їх подальшого архівування та аналізу.

Сервер забезпечує роботу інфраструктурних сервісів (DNS та DHCP), без яких неможлива коректна адресація та розв'язання доменних імен у локальному сегменті мережі.

Комунікаційний каркас системи та фізичні межі захищеного периметра формуються спеціалізованим мережевим обладнанням. Маршрутизатор у даній схемі виконує роль шлюзу безпеки: окрім маршрутизації пакетів, він реалізує механізм трансляції мережевих адрес (NAT), що дозволяє приховати внутрішню топологію мережі від зовнішнього спостерігача, та забезпечує базову фільтрацію вхідного трафіку.

Безпосередня комутація кадрів у дротовому сегменті забезпечується комутатором, який організовує фізичне середовище передачі даних між компонентами системи.

Зовнішнє середовище, що знаходиться поза контролем адміністратора, представлене у моделі як Інтернет-хмара.

Цей сегмент емулює глобальну мережу і розглядається як основне джерело

потенційних загроз. Саме з боку Інтернет-хмари, сценарії зовнішніх кібератак, які виступають початком атаки для активації захисних механізмів агентної системи, дозволяючи перевірити її реакцію на несанкціоновані втручання ззовні.

Керування конфігурацією та оперативний моніторинг покладено на окремий пристрій адміністратора (Laptop-PT Admin).

Цей вузол забезпечує два незалежних канали взаємодії із системою: моніторинг інцидентів здійснюється через веб-інтерфейс із доступом до серверної статистики в реальному часі, тоді як безпосереднє керування мережевим обладнанням реалізовано через пряме підключення до консольних портів (інтерфейс RS-232), що гарантує доступ до налаштувань навіть у разі відмови основної мережі.

Схематичне зображення реалізації описаної безпровідної топології мережі наведено на рисунку 2.1.

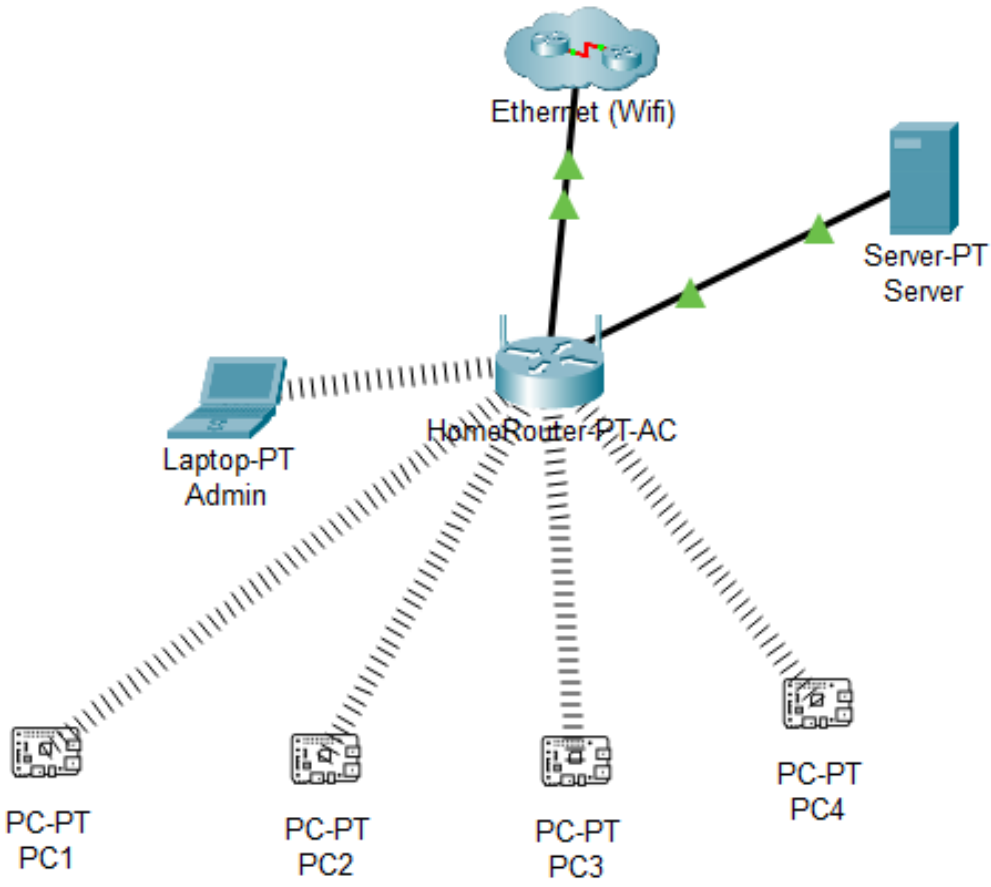


Рисунок 2.1 – Безпровідна (Wi-Fi) топологія розгортання системи

Топологія побудована за принципом «зірка» з центром комутації на маршрутизаторі HomeRouter-PT-AC.

У даній конфігурації агенти моніторингу (PC1–PC4) та робоча станція адміністратора підключені до спільного середовища передачі через радіоканал, тоді як центральний сервер з’єднаний з маршрутизатором фізичною лінією зв'язку. Таке гібридне підключення дозволяє мінімізувати затримки на стороні сервера, залишаючи агентів у мобільному сегменті.

Основною технічною особливістю реалізації системи в даній топології є метод отримання первинних даних. Оскільки захоплення трафіку в безпровідному середовищі вимагає специфічної конфігурації мережевих адаптерів одноплатних комп'ютерів.

Для забезпечення функціоналу агентів-сенсорів інтерфейси переводяться в режим моніторингу (monitor mode). На відміну від стандартного режиму (promiscuous mode), це дозволяє, використовуючи системні бібліотеки рівня libpcap, здійснюючи пасивне перехоплення.

Такий підхід забезпечує повну видимість ефіру, дозволяючи аналізувати не лише корисне навантаження, але й службові заголовки протоколів управління безпровідною мережею.

На рисунку 2.2 представлено альтернативне схематичне зображення провідної топології мережі.

Ця конфігурація системи базується на стандарті Ethernet і передбачає підключення агентів до центрального керованого комутатора, який виступає точкою агрегації трафіку перед маршрутизатором.

Така топологія характеризується підвищеною стійкістю до завад та стабільністю каналів зв'язку, що є критичним для промислових сегментів IoT.

Оскільки технологія комутації ізолює потоки даних, унеможливлюючи пряме перехоплення, для моніторингу застосовано метод дзеркалювання портів (SPAN).

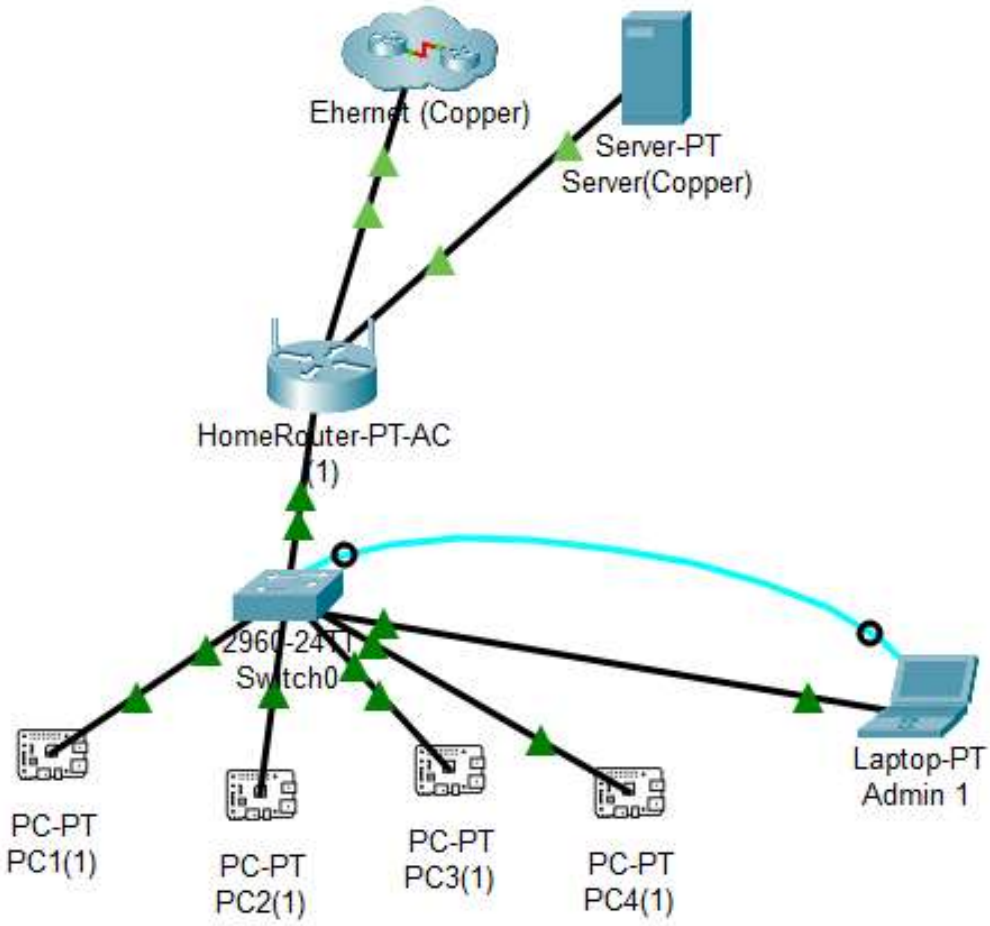


Рисунок 2.2 – Провідна (Ethernet) топологія розгортання системи

Механізм налаштовано на дуплікацію всього вхідного та вихідного трафіку з магістрального інтерфейсу на виділений порт агента-аналітика, що дозволяє здійснювати перевірку пакетів у реальному часі без внесення затримок у функціонування продуктивної мережі.

Рисунок 2.3 репрезентує комбіновану (гібридну) топологію, яка найбільш точно відображає реальні умови розгортання сучасних IoT-систем.

Ця топологія об'єднує обидва сегменти мережі (дротовий та безпровідний) у єдиний контур безпеки під керуванням центрального маршрутизатора, що забезпечує з’єднання між середовищами та вихід до глобальної мережі.

Моніторинг реалізується комплексно: SPAN-технологія контролює дротовий периметр, а агенти в режимі радіомоніторингу – безпровідний ефір.

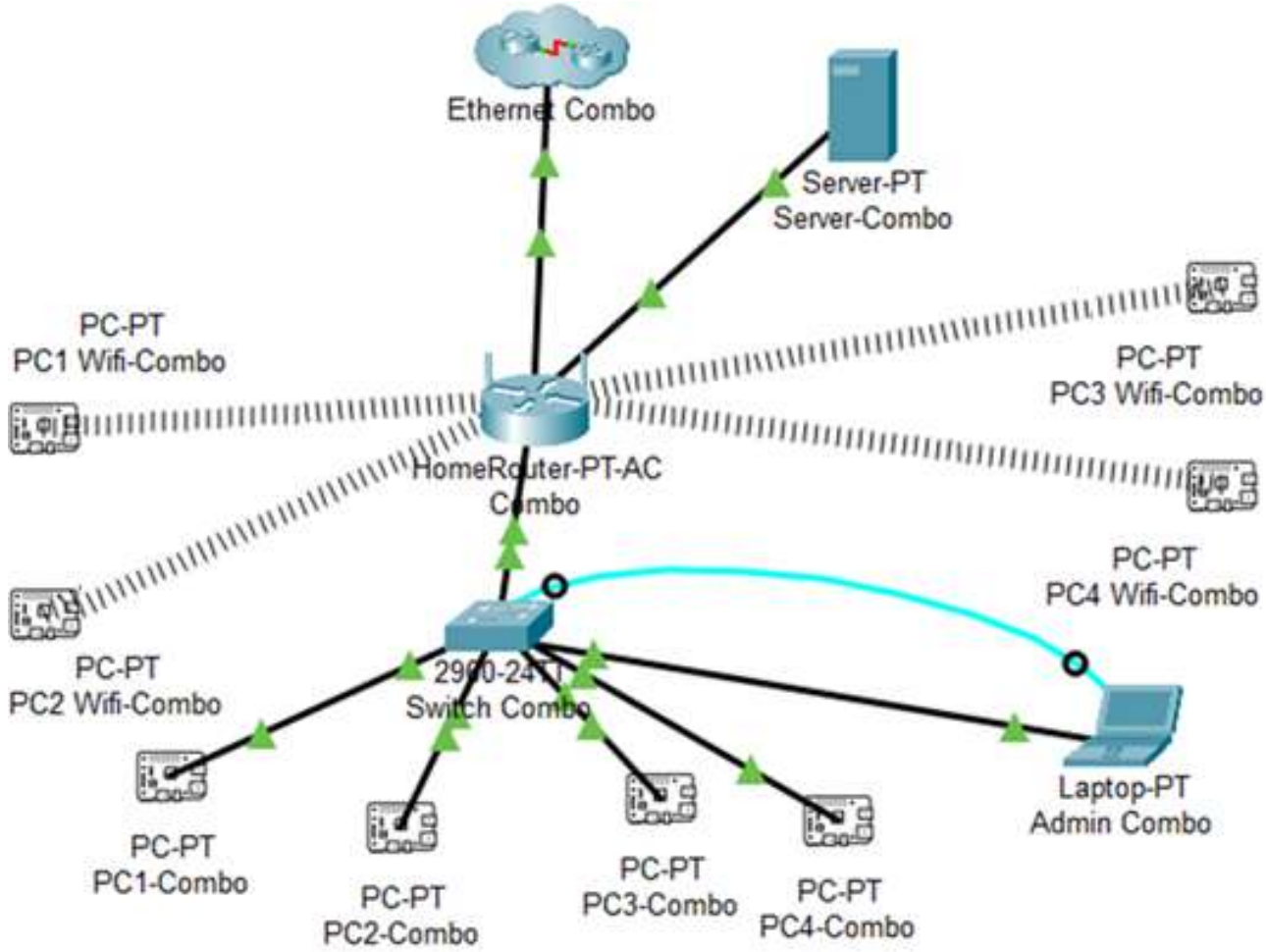


Рисунок 2.3 – Комбінована (гібридна) топологія системи

В таблиці 2.2 наведено порівняльну характеристику трьох розглянутих топологій, що дозволяє оцінити їх ефективність залежно від умов експлуатації та наявного апаратного забезпечення.

Таблиця 2.2 – Порівняння топологій системи

Критерій	Безпровідна (Wi-Fi)	Провідна (Ethernet)	Комбінована
1	2	3	4
Метод перехоплення	Пасивний моніторинг у режимі Monitor Mode засобами бібліотеки libpcap.	Технологія SPAN для дзеркалювання портів на рівні комутатора.	Комплексне використання SPAN для LAN та Monitor Mode для WLAN.

Кінець таблиці 2.2

1	2	3	4
Апаратні вимоги	Мінімальний набір обладнання маршрутизатора та сумісного адаптера.	Високі вимоги через обов'язкову наявність керованого комутатора (L2/L3).	Максимальні вимоги до повного стека маршрутизатора та комутатора.
Охоплення	Локальне покриття в межах радіусу дії та за відсутності перешкод.	Сегментоване покриття з повним контролем кабелю, але неможливо охопити Wi-Fi.	Повний контроль фізичних ліній та радіоефіру.
Якість даних	Варіативна якість із ризиком втрати пакетів через інтерференцію.	Висока якість із гарантованою доставкою копій пакетів.	Максимальна надійність Ethernet у поєднанні з гнучкістю Wi-Fi.
Ключова перевага	Розгортання системи та відсутність зайвих кабелів.	Стабільність аналізу великих потоків даних без затримок.	Синергетичний ефект кореляції загроз між різними середовищами.
Недоліки	Залежність від радіозавад та відносно низька пропускна здатність.	Відсутність контролю мобільних пристроїв та прив'язка до розеток.	Підвищена складність налаштування та висока вартість інфраструктури.

За результатами порівняльного аналізу саме гібридну конфігурацію обрано як еталонну модель для впровадження, оскільки вона дозволяє експериментально

підтвердити універсальність розробленого методу та гарантує цілісність контролю безпеки в умовах конвергенції різнорідних середовищ передачі даних.

2.4 Проєктування логічної структури програмного агента для одноплатного комп'ютера

Розроблена архітектура базується на конвеєрному принципі обробки даних, що дозволяє оптимізувати використання обмежених обчислювальних ресурсів одноплатного комп'ютера шляхом чіткого розподілу навантаження між спеціалізованими модулями.

Концептуальну схему алгоритму функціонування та взаємодії компонентів агента наведено на рисунку 2.4.

Життєвий цикл обробки інформації розпочинається на фізичному рівні з ініціалізації моніторингу мережевих інтерфейсів. Для мінімізації затримок Агент-Сенсор використовує низькорівневі системні виклики бібліотеки libpcap, що дозволяє працювати з «сирим» трафіком в обхід стандартного стека обробки операційної системи.

Це є необхідною умовою для високошвидкісного аналізу на пристроях з обмеженими ресурсами. Замість передачі повних тіл пакетів, що перевантажило б канал зв'язку, модуль здійснює збір даних, вилучення ключових метаданих та їх агрегацію у компактні вектори ознак, придатні для подальшого аналізу.

Сформований вектор передається до Агента-Аналітика, який виконує функцію локального фільтра прийняття рішень, реалізуючи принцип граничних обчислень. Якщо трафік класифікується як нормальний, цикл обробки миттєво завершується, вивільняючи ресурси процесора і не створюючи навантаження на мережу.

Лише у випадку виявлення статистичного відхилення генерується локальний сигнал тривоги, який ініціює перехід системи до етапу координації.

На етапі верифікації загрози в роботу вступає Агент-Координатор, завданням якого є мінімізація кількості помилкових спрацювань.

Рисунок 2.4 – Концептуальна логічна структура програмних агентів

Система використовує механізм розподіленого консенсусу через протокол MQTT: координатор публікує попередження про аномалію та одночасно перевіряє аналогічні топіки від сусідніх вузлів.

Остаточне підтвердження інциденту відбувається лише за умови кореляції локальної аномалії з даними інших агентів або ж у випадку виявлення критичної сигнатури атаки.

Кінцевий результат процесу здійснює Агент-Репортер, який підтверджує інцидент у формат JSON для відправки на центральний сервер, після чого система повертається у стан очікування нових пакетів.

Розглянемо програмно-технологічний стек та потік даних агента.

Основою розробки обрано мову програмування Python, що обумовлено її розвиненою системою бібліотек для мережевого аналізу та можливістю інтеграції низькорівневих C-модулів для критичних ділянок коду.

Застосовано оптимізовану Python-обгортку над системною бібліотекою libpcap, що дозволяє здійснювати захоплення пакетів безпосередньо на рівні ядра операційної системи, мінімізуючи ризики втрати даних при пікових навантаженнях.

Обробка захоплених даних покладена на спеціалізований модуль виділення ознак, який використовує бібліотеку NumPy для виконання швидких векторних операцій та розрахунку статистичних метрик, таких як ентропія чи стандартне відхилення.

Для парсингу заголовків пакетів залучено бібліотеку dpkt, що забезпечує значно вищу швидкість розбору структур даних порівняно зі стандартними засобами. Сформовані вектори ознак передаються до аналітичного ядра, яке базується на алгоритмах дерев рішень, реалізованих за допомогою оптимізованих бібліотек статистичного моделювання.

Взаємодія агента із зовнішнім середовищем та сервером забезпечується комунікаційним модулем на базі клієнта paho-mqtt, що гарантує надійний асинхронний обмін повідомленнями за протоколом MQTT.

Функціонал розділено на незалежні логічні потоки: високопріоритетний

потік захоплення (Thread A) відповідає виключно за зчитування пакетів з мережевого інтерфейсу та їх розміщення у проміжному буфері (черзі), тоді як аналітичний потік (Thread B) асинхронно вибирає дані з черги, формує вектори ознак та виконує їх класифікацію.

Такий підхід гарантує безперервність моніторингу ефіру навіть у моменти тимчасового пікового навантаження на обчислювальне ядро аналітика.

2.5 Вибір ознак мережевого трафіку для виявлення аномалій в IoT

Фундаментальною основою розробленої системи виявлення вторгнень є концепція легкої архітектури (L-IDS), яка не проводить глибокого аналізу вмісту пакетів на користь обробки статистичних метаданих. Такий підхід є можливим в умовах Інтернету речей, де шифрування трафіку та обмежені обчислювальні потужності одноплатних комп'ютерів унеможливлюють аналіз корисного навантаження в реальному часі.

Для забезпечення високої точності детекції пропонується використовувати гібридний вектор ознак, що базується на комбінації характеристик мережевих потоків, аналізі заголовків транспортного рівня та математичній оцінці ентропії в межах ковзних часових вікон.

Першими базовими характеристиками потоку, що формуються на основі унікального п’ятикомпонентної бази (IP-адреси та порти джерела і призначення, протокол). Ключовим індикатором тут виступає тривалість потоку (flow_duration): аномально короткі сесії до великої кількості хостів свідчать про мережеве сканування, тоді як надмірно довгі з’єднання можуть вказувати на тунелювання даних або наявність каналів управління ботнетом.

Для виявлення атак на відмову в обслуговуванні (DoS) критично важливим є аналіз асиметрії трафіку через метрики кількості пакетів у прямому та зворотному напрямках (fwd/bwd_pkt_count), оскільки значний дисбаланс є характерною ознакою флуду.

Окрім обсягу, аналізуються часові характеристики, зокрема середнє значення та стандартне відхилення інтервалів між прибуттям пакетів (IAT).

Це дозволяє розрізняти низькошвидкісні атаки,що характеризуються аномально високими паузами, та інтенсивні флуд-атаки, де інтервали спрямовуються до нуля.

Наступний рівень аналізу базується на прапорцях протоколу TCP, що дозволяє виявляти порушення логіки рукостискань.

Стрімке зростання кількості пакетів із прапорцем SYN (syn_flag_count) за відсутності відповідних ACK-підтверджень однозначно класифікується як атака SYN Flood.

Додатково моніторяться специфічні комбінації прапорців, такі як FIN та RST, які використовуються зловмисниками для прихованого сканування (Stealth Scans) з метою обходу традиційних брандмауерів.

Для виявлення розподілених атак (DDoS) та масштабних сканувань, які важко помітити на рівні окремих пакетів, застосовуються інтегральні метрики на основі часових вікон.

Головним математичним інструментом тут виступає ентропія Шеннона, яка дозволяє кількісно оцінити ступінь хаотичності розподілу мережевих адрес. Розрахунок ентропії (H) для випадкової величини (X) здійснюється за формулою:

$$H(X) = -\sum_{i=1}^{n} P(x_i)\, log_2\, P(x_i) \quad (1)$$

де $H(X)$ – ентропія, яка визначає міру невизначеності в бітах; n – кількість унікальних елементів у вибірці (наприклад, унікальних IP-адрес за 3 секунди); $P(x_i)$ – ймовірність появи конкретного і-го елемента, що розраховується як відношення його частоти до загального обсягу вибірки.

Інтерпретація результатів базується на відхиленні від норми: різке зростання ентропії IP-адрес джерела (*src_ip_entropy*) свідчить про одночасну активність тисяч хостів, що є сигнатурою DDoS-атаки, тоді як висока ентропія

портів призначення (dst_port_entropy) вказує на спробу сканування вразливостей на одному вузлі.

Оскільки загальні транспортні ознаки можуть бути недостатніми для виявлення атак прикладного рівня, система додатково аналізує специфічні протоколи IoT. Для протоколу MQTT (TCP) контролюється частота з'єднань (connect_rate), сплеск якої вказує на спробу перебору паролів, та наявність підписок із використанням символів підстановки (wildcard_sub_count), що є ознакою шпигунства за даними.

Для протоколу CoAP (UDP) критичним є аналіз коефіцієнта підсилення (amplification_ratio), що визначається як співвідношення розміру відповіді до запиту, оскільки це дозволяє виявляти спроби використання IoT-пристроїв як підсилювачів у DDoS-атаках.

Узагальнена класифікація обраних метрик та їх цільове призначення наведені в таблиці 2.3.

Таблиця 2.3 – Обрані метрики

Категорія	Приклади метрик	Призначення та тип атак, що виявляються
Ознаки потоку	flow_duration, fwd_pkt_count	Виявлення аномалій обсягу даних, DoS/DDoS-атак та тунелювання.
Часові інтервали	fwd_iat_mean, fwd_iat_std	Ідентифікація часових аномалій, атак типу «Slow Loris» та флуду.
Прапорці TCP	syn_flag_count, fin_flag_count	Виявлення порушень протоколу, SYN Flood, сканування (XMAS/FIN).
Статистика вікна	src_ip_entropy, dst_port_entropy	Оцінка розподіленості атак (DDoS) та горизонтального сканування.
Специфіка MQTT	connect_rate, wildcard_sub_count	Захист від Brute-force, несанкціонованого доступу та витоку даних.
Специфіка CoAP	coap_req_amplification_ratio	Виявлення DDoS-атак із підсиленням.

2.6 Методики оцінки ефективності методу

Невід'ємним етапом проєктування системи захисту для ресурсно-обмежених середовищ (L-IDS) є формалізація суворої методології її валідації, що дозволяє перетворити абстрактні вимоги до ефективності у вимірювані кількісні показники.

2.6.1 Методика порівняльної оцінки продуктивності

Оскільки поняття «легковагості» є відносним, розроблена методика передбачає проведення прямого порівняльного тестування запропонованого агента з аналогами в ідентичних умовах експлуатації.

Комплексна оцінка базується на тріангуляції критичних для IoT метрик: обчислювального навантаження на центральний процесор, ефективності менеджменту оперативної пам'яті та граничної пропускної здатності мережевого інтерфейсу.

2.6.2 Методика вимірювання навантаження

Визначення реальної обчислювальної вартості функціонування агента реалізується через дворівневу процедуру вимірювання ресурсів, що дозволяє оцінити вплив програмного забезпечення на апаратну платформу.

На першому етапі фіксується базовий профіль споживання у стані спокою (Idle Mode), що демонструє статичні накладні витрати архітектури на підтримку життєвого циклу процесу без активної обробки даних.

Другий етап передбачає динамічний моніторинг пікових та середніх показників під час обробки еталонного потоку трафіку (Under Load), що дозволяє дослідити лінійність масштабування алгоритмів при зростанні навантаження.

Для забезпечення чистоти експерименту та точності результатів використовуються інструменти ізольованого системного процесу, що дозволяє нівелювати вплив фонових служб операційної системи на кінцеві метрики споживання процесора та оперативної пам’яті.

2.6.3 Методика вимірювання пропускної здатності

Найбільш критичним параметром надійності системи визначено її граничну

пропускну здатність, оскільки в контексті задач виявлення вторгнень будь-яка втрата пакетів еквівалентна виникненню неконтрольованих у периметрі безпеки.

Методика стрес-тестування полягає у генерації синтетичного трафіку з контрольованим нарощуванням інтенсивності до моменту досягнення точки насичення апаратних ресурсів, яка фіксується за зростанням системного лічильника відкинутих кадрів на мережевому інтерфейсі.

Ключовим критерієм успішності у даному дослідженні затверджено поріг «Нульової втрати» — це максимальна швидкість обробки пакетів, при якій гарантується абсолютна повнота аналізу без втрати жодної одиниці даних, що є необхідною умовою для кваліфікації системи як надійного інструменту захисту критичної інфраструктури.

2.6.4 Методика порівняльної оцінки продуктивності

Окрім оцінки ефективності, необхідною умовою валідації системи є визначення перевірки коректності роботи алгоритмів класифікації.

Для забезпечення об'єктивності дослідження, оцінка якості рішень Агента-Аналітика здійснюється на основі еталонних наборів даних (Labeled Datasets), де кожному вектору ознак попередньо присвоєно верифіковану мітку належності до класу нормальної поведінки або атаки.

Цей етап проводиться для підтвердження того, що розроблена система здатна не лише швидко обробляти трафік, але й приймати вірні рішення в умовах реальних загроз.

2.6.5 Матриця помилок

Фундаментальним інструментом для структурування результатів роботи класифікатора є матриця помилок (Confusion Matrix), яка формалізує простір подій у чотирьохвимірній площині.

Вона дозволяє чітко розмежувати коректні спрацювання системи, такі як істинно позитивні (True Positive) та істинно негативні (True Negative) рішення, від помилкових висновків.

В контексті IoT приділяється аналізу помилок: відхилення першого роду, або хибні позитивні рішення (False Positive), є критичними через витрачання

обмежених енергетичних ресурсів пристроїв на обробку фантомних загроз.

Натомість помилки другого роду, або хибні негативні рішення (False Negative), є ще небезпечнішими, оскільки означають факт пропуску реальної атаки, що створює «сліпу зону» в периметрі безпеки та компрометує надійність усієї системи.

2.6.6 Ключові метрики оцінки ефективності

Для кількісної формалізації надійності системи на основі компонентів матриці помилок розраховується комплекс стандартизованих коефіцієнтів.

Першим базовим індикатором є інтегральна коректність класифікації A_{int} , що відображає загальний відсоток безпомилкових рішень системи відносно всього обсягу вибірки:

$$A_{int} = \frac{TP+TN}{TP+TN+FP+FN}, \quad (2)$$

де TP – істинно позитивні рішення (True Positive); TN – істинно негативні рішення (True Negative); FP – хибні позитивні рішення (False Positive); FN – хибні негативні рішення (False Negative).

Однак, враховуючи дисбаланс класів у реальному трафіку, цей показник доповнюється метрикою прогностичної значущості позитивного результату P_{val}.

Метрика P_{val} визначає рівень довіри до системи, розраховуючи ймовірність того, що згенерована тривога є реальною загрозою, а не шумом.

$$P_{val} = \frac{TP}{TP+FP}. \quad (3)$$

Метрика P_{val} визначає рівень довіри до системи, розраховуючи ймовірність того, що згенерована тривога є реальною загрозою, а не шумом.

Паралельно обчислюється коефіцієнт повноти детектування R_{det}, який характеризує чутливість сенсорів і визначається як здатність алгоритму виявляти всі наявні загрози у потоці даних:

$$R_{det} = \frac{TP}{TP+FN}. \quad (4)$$

Оскільки між чутливістю та точністю існує зворотна кореляція, як основний критерій успішності навчання моделі обрано узагальнену оцінку F_{score}:

$$F_{score} = 2 * \frac{P_{val}*R_{det}}{P_{val}+R_{det}}. \quad (5)$$

Інтегральний показник F_{score} дозволяє знайти баланс між мінімізацією пропущених атак та зниженням рівня хибних тривог. Саме максимізація значення F_{score} визначена як цільова функція при налаштуванні параметрів Агента-Аналітика.

2.6.7 Методика аналізу трафіку за обмежених ресурсів

Останнім етапом проєктування архітектури є вибір способу класифікації, який зможе ефективно функціонувати в умовах жорсткого дефіциту оперативної пам'яті та процесорного часу одноплатного комп'ютера. Ключовим архітектурним рішенням для досягнення необхідної швидкодії стало повне логічне та фізичне розмежування етапів роботи системи. Ресурсоємний процес статистичного аналізу та генерації вирішальних правил (калібрування) винесено за межі вбудованих систем – він виконується в офлайн-режимі на потужному сервері адміністратора з використанням еталонних наборів даних.

У режимі реального часу аналітичний модуль виконує лише функцію безпосередньої класифікації, пропускаючи вхідний вектор ознак через заздалегідь сформовану систему умов. Цей процес є обчислювально легким та не створює критичного навантаження на процесор, що дозволяє агенту працювати без затримок навіть при інтенсивному мережевому трафіку.

Для програмної реалізації модуля прийняття рішень обрано метод «Випадковий ліс» (Random Forest), який у контексті даної роботи розглядається як статистичний ансамблевий алгоритм. Вибір саме цього підходу обумовлений його унікальною архітектурною придатністю для IoT-пристроїв.

На відміну від складних обчислювальних структур, процедура класифікації у цьому методі технічно зводиться до набору простих логічних умов (IF-THEN).

Застосування ансамблевого підходу, де кінцевий висновок формується шляхом усереднення результатів роботи множини незалежних дерев рішень, забезпечує системі високу стійкість до випадкового «шуму» в даних, характерного для безпровідних мереж. Це дозволяє досягти вищої точності та мінімізувати кількість хибних спрацювань у порівнянні з поодинокими методами.

Численні експериментальні дослідження підтверджують, що саме Random Forest демонструє оптимальний баланс між точністю виявлення аномалій та швидкістю обробки пакетів, а наявність оптимізованих бібліотек дозволяє легко інтегрувати цей алгоритм у програмний код агента.

2.7 Висновки

У розділі надано опис мультиагентного методу моніторингу мережевого трафіку IoT, спрямованого на створення легковагової та масштабованої системи виявлення аномалій для ресурсно-обмежених середовищ. На цьому етапі сформовано концептуальну архітектуру системи, визначено логічну взаємодію агентів, проаналізовано можливі топології розгортання та обґрунтовано вибір ключових технічних рішень для мінімізації обчислювальних витрат. Створена модель дозволяє сформувати цілісне уявлення про майбутню систему та закладає фундамент для її подальшої практичної реалізації.

В жоді досліджень обґрунтовано архітектуру розподіленої мультиагентної системи моніторингу IoT, що базується на децентралізованій взаємодії через протокол MQTT.

Серед проаналізованих топологій пріоритетною визначено гібридну конфігурацію, яка об’єднує контроль дротових та безпровідних сегментів у єдиний контур безпеки, забезпечуючи наскрізну кореляцію інцидентів.

Для оптимізації роботи на одноплатних комп’ютерах спроєктовано

модульну структуру агента з асинхронною обробкою даних, що дозволило нівелювати ризики втрати пакетів. В якості аналітичного ядра обрано статистичний ансамблевий алгоритм «Випадковий ліс».

Застосування концепції розділення процесів, де ресурсоємна генерація правил виконується офлайн на сервері, а агент здійснює лише швидку перевірку умов, гарантує мінімальне використання процесорного часу.

Розроблена структурна схема взаємодії агентів передбачає чіткий розподіл функцій між спеціалізованими модулями: збором і фільтрацією трафіку, локальною аналітикою, координацією між вузлами та формуванням повідомлень для зовнішніх систем. Такий підхід забезпечує гнучкість архітектури та можливість масштабування, але на даному етапі він розглядається як логічна конструкція, що потребує подальшої верифікації в реальних умовах.

Окремо визначено набір метрик та ознак мережевого трафіку, які планується використовувати для детектування аномалій у середовищах IoT. Набір ознак сформовано з урахуванням обмежень одноплатних комп’ютерів та характеру типових атак. Водночас ці ознаки ще не пройшли практичне тестування на реальних вибірках і потребуватимуть додаткового експериментального уточнення. Методологічний опис способів оцінки ефективності системи також носить попередній характер і визначає рамки для майбутніх досліджень, включаючи плановане порівняння продуктивності, вимірювання граничної пропускної здатності, аналіз матриці помилок та застосування показників точності.

Високу точність виявлення забезпечує сформований вектор діагностичних ознак, що базується на статистичних метриках потоку та ентропії, замінюючи ресурсоємний.

Також формалізовано методологію верифікації, яка включає оцінку апаратної ефективності та математичні метрики достовірності з цільовою максимізацією узагальненої гармонійної оцінки.

Описана архітектура, вибрані метрики та методики оцінювання є фундаментом, на якому базуватиметься подальша практична апробація системи.

3 РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНА ВАЛІДАЦІЯ МЕТОДУ І СИСТЕМИ АНАЛІЗУ ТРАФІКУ

3.1 Архітектура та програмна реалізація модуля аналізу мережевого трафіку

Програмна реалізація системи виконана на основі принципів модульної архітектури та мікросервісної декомпозиції, що є критично важливим для забезпечення гнучкості розгортання та можливості горизонтального масштабування рішення.

Головним завданням на етапі реалізації стало створення оптимізованого програмного ядра агента, здатного виконувати повний цикл захоплення, попередньої обробки, агрегації ознак та статистичного аналізу мережевих пакетів, без створення критичних затримок у передачі даних.

Цільовою апаратною платформою для розгортання агента визначено одноплатні комп’ютери (ОПК) під управлінням операційної системи Linux.

Архітектурне рішення передбачає ізоляцію окремих функціональних модулів агента, що дозволяє реалізувати їх незалежне оновлення, для забезпечення роботи агента незалежно від версії базової операційної системи хоста.

3.1.1 Обґрунтування вибору інструментальних засобів розробки

Вибір технологічного стека базується на жорстких вимогах до швидкодії системи в реальному часі, реалізації програмного коду та наявності спеціалізованих інструментів для мережевого аналізу.

Основною мовою програмування, було вибрано Python версії 3.9+. Використання цієї мови, дозволяє ефективно поєднувати високорівневу логіку агентів із низькорівневими системними викликами через C-розширення.

Реалізація захоплення трафіку базується на бібліотеці libpcap, взаємодія з якою здійснюється через інтерфейс pcapy-ng.

Такий підхід дозволяє виконувати захоплення пакетів на рівні ядра операційної системи, що мінімізує накладні витрати ресурсів процесора та запобігає втраті пакетів при високому навантаження.

Бібліотека Scapy, була відхилена через значні архітектурні накладні витрати при обробці інтенсивних потоків даних, що спричиняло критичні затримки в контурі реального часу.

Для десеріалізації бінарних даних використано бібліотеку dpkt, яка забезпечує швидкий розбір заголовків Ethernet, IP та TCP/UDP без створення об’ємних данних для кожного поля, що зменшує споживання пам'яті в 4–5 разів.

Підсистема обробки даних реалізована з використанням бібліотеки NumPy, що забезпечує векторизацію математичних операцій та дозволяє обробляти масиви статистичних ознак значно ефективніше за стандартні структури даних.

Аналітичне ядро створено на базі бібліотеки scikit-learn, яка забезпечує оптимізовані методи інференсу для алгоритму Random Forest.

Для організації асинхронного обміну даними між вузлами моніторингу та сервером агрегації використано бібліотеку paho-mqtt, що реалізує протокол MQTT версії 5.0.

З метою нівелювання ризику блокування критичного процесу захоплення трафіку під час виконання аналітичних операцій, архітектуру програмного модуля побудовано на основі моделі асинхронної декомпозиції завдань.

Така організація забезпечує логічну ізоляцію процедур взаємодії з мережевим обладнанням, що характеризуються очікуванням надходження даних, від ресурсоємних алгоритмів математичної обробки.

Це забезпечує конкурентне виконання різнорідних потоків, нівелюючи внутрішні архітектурні обмеження інтерпретатора мови Python щодо паралельних обчислень. Структура програмного забезпечення реалізована через три незалежні вектори виконання, синхронізовані за допомогою буферів обміну.

Перший потік, що виконує функції підсистеми сенсорів, функціонує з найвищим пріоритетом операційної системи. Функціональне навантаження мінімізовано до операцій безперервного зчитування пакетів з буфера мережевого інтерфейсу та їх розміщення у черзі попередньої обробки.

Такий підхід гарантує своєчасне вивільнення апаратних ресурсів адаптера навіть при екстремальних навантаженнях, запобігаючи відкиданню кадрів

драйвером.

Другий потік, відповідальний за аналітичне ядро, здійснює вибірку даних з черги, проводить синтаксичний розбір заголовків протоколів, розраховує статистичні метрики та виконує класифікацію векторів ознак.

Третій потік забезпечує асинхронну взаємодію із зовнішнім середовищем, обробляючи чергу інцидентів та ініціюючи передачу телеметрії через протокол MQTT, що дозволяє ізолювати контур безпеки від потенційних затримок у каналах зв'язку.

3.1.2 Реалізація багатопотокової архітектури обробки даних

З метою нівелювання ризику блокування критичного процесу захоплення трафіку під час виконання аналітичних операцій, архітектуру програмного модуля побудовано на основі моделі асинхронної декомпозиції завдань.

Така організація забезпечує логічну ізоляцію процедур взаємодії з мережевим обладнанням, що характеризуються очікуванням надходження даних, від ресурсоємних алгоритмів математичної обробки.

Це забезпечує конкурентне виконання різнорідних потоків, нівелюючи внутрішні архітектурні обмеження інтерпретатора мови Python щодо паралельних обчислень. Структура програмного забезпечення реалізована через три незалежні вектори виконання, синхронізовані за допомогою буферів обміну.

Перший потік, що виконує функції підсистеми сенсорів, функціонує з найвищим пріоритетом операційної системи. Функціональне навантаження мінімізовано до операцій безперервного зчитування пакетів з буфера мережевого інтерфейсу та їх розміщення у черзі попередньої обробки.

Такий підхід гарантує своєчасне вивільнення апаратних ресурсів адаптера навіть при екстремальних навантаженнях, запобігаючи відкиданню кадрів драйвером.

Другий потік, відповідальний за аналітичне ядро, здійснює вибірку даних з черги, проводить синтаксичний розбір заголовків протоколів, розраховує статистичні метрики та виконує класифікацію векторів ознак.

Третій потік забезпечує асинхронну взаємодію із зовнішнім середовищем,

обробляючи чергу інцидентів та ініціюючи передачу телеметрії через протокол MQTT, що дозволяє ізолювати контур безпеки від потенційних затримок у каналах зв'язку (Додаток.Б. Лістинг Б.3).

3.1.3 Програмна реалізація алгоритмів виділення ознак

Критичним елементом архітектури є модуль виділення ознак, ефективність якого визначає загальну латентність системи. Основну увагу при реалізації зосереджено на програмній оптимізації розрахунку статистичних метрик розподілу даних, які створюють найбільше навантаження на процесор.

Замість класичних ітераційних алгоритмів, було створено підхід, на основі хеш-структур для підрахунку частот появи унікальних елементів, що дозволило досягти лінійної асимптотичної складності $O(N)$, де N – кількість пакетів у вибірці.

Математичні операції над отриманими частотними масивами виконуються виключно засобами бібліотеки NumPy, що забезпечує векторизацію обчислень та перенесення навантаження з повільного інтерпретатора Python на оптимізований машинний С-код.

Для агрегації даних система використовує механізм адаптивного ковзного вікна, реалізованого на базі кільцевого буфера фіксованого розміру.

Цей параметр винесено у конфігураційний файл, що дозволяє гнучко балансувати між чутливістю виявлення та споживанням пам'яті.

Використання підходу, що базується на кількості подій, а не на часових інтервалах, забезпечує автоматичну синхронізацію швидкості аналізу з інтенсивністю вхідного трафіку: при масованих атаках буфер заповнюється миттєво, гарантуючи швидку реакцію, тоді як у режимі спокою система накопичує достатній обсяг статистики без створення надлишкових обчислень.

3.1.4 Структура даних та протоколи обміну

З метою забезпечення повної інтероперабельності розробленого рішення із зовнішніми екосистемами кібербезпеки, розроблено уніфіковану структуру даних для інкапсуляції інформації про інциденти.

Формат обміну повідомленнями базується на стандарті JSON, який має

певні накладні витрати на текстову серіалізацію, є стандартом для інтеграції систем.

Схема повідомлення загроз спроєктована, щоб забезпечити вичерпний контекст події. Технічна реалізація включає схему для підтримки зворотної сумісності при оновленнях агентів, а також часову мітку генерації події з точністю до мікросекунд, що є критичним для кореляції інцидентів на сервері.

Ідентифікація джерела здійснюється через унікальний UUID сенсора, що дозволяє однозначно прив'язати подію до конкретного вузла топології.

Наповнення повідомлення включає: тип виявленої аномалії, визначений класифікатором, та коефіцієнт статистичної достовірності.

Коефіцієнт статистичної достовірності розраховується як нормалізована частка, що підтвердили даний клас загрози, і набуває значень у діапазоні від 0.0 до 1.0.

На основі цього показника та класу загрози формується інтегральний рівень критичності інциденту. Для забезпечення можливості верифікації рішень адміністратором, структура містить вкладений об'єкт із даними, куди записуються ключові технічні деталі, такі як: підозрілі IP-адреси, цільові порти та розраховані значення ентропії, що слугували математичною підставою для виявлення.

Реалізація методів серіалізації та асинхронної відправки даних структур наведена у відповідному додатку роботи (Додаток.Б. Лістинг Б.4).

3.1.5 Механізми забезпечення відмовостійкості

Першим механізмом захисту стабільності системи є керування пам'яттю в умовах перенавантаження.

Оскільки обсяг оперативної пам'яті одноплатних комп'ютерів є суворо лімітованим, неконтрольоване зростання черг обробки при пікових навантаженнях може призвести до вичерпання ресурсів та аварійної зупинки процесу операційною системою.

Для запобігання цьому сценарію в програмному коді створено механізм ротації буфера з фіксованим лімітом ємності. При спробі додавання даних у заповнену чергу packet_queue система перехоплює виключення переповнення та застосовує політику відкидання найстаріших пакетів.

Такий підхід гарантує, що споживання пам'яті залишатиметься в межах детермінованих границь навіть ціною втрати частини історичних даних, що є прийнятним компромісом для збереження життєздатності вузла.

На рівні логіки виконання забезпечено сувору ізоляцію помилок всередині багатопотокової архітектури.

Робочий цикл кожного потоку інкапсульовано в захищені блоки обробки виключень на верхньому рівні ієрархії викликів.

Це архітектурне рішення є критичним для забезпечення безперервності сервісу: у випадку виникнення непередбачуваної помилки, якщо при спробі розбору пошкодженого мережевого пакету або збої в математичному модулі, система не припиняє роботу аварійно, а лише фіксує інцидент у журналі подій та автоматично переходить до наступної ітерації циклу обробки.

Враховуючи нестабільність безпровідних каналів зв'язку, характерну для периферійних обчислень, комунікаційний модуль обладнано підсистемою відновлення з'єднання.

Реалізація базується на використанні фонових процесів бібліотеки paho-mqtt із застосуванням алгоритму експоненціальної затримки.

Цей механізм дозволяє агенту автоматично відновлювати сесію з брокером після фізичних розривів зв'язку, поступово збільшуючи інтервали між спробами повторного підключення.

Такий алгоритм не лише забезпечує автономність агента, але й запобігає створенню потоку запитів на центральний сервер у момент відновлення роботи мережевої інфраструктури.

3.2 Порівняльне тестування продуктивності та аналіз ресурсоємності

Для верифікації нефункціональних вимог щодо ефективності роботи системи на периферійних пристроях було проведено серію навантажувальних тестів. Метою експерименту є встановлення залежності споживання системних

ресурсів (CPU, RAM) від інтенсивності мережевого трафіку та порівняння отриманих показників з еталонними значеннями існуючих рішень.

3.2.1 Архітектура віртуалізованого випробувального полігону

Оскільки спроєктована система базується на принципах розподіленої архітектури, де множина агентів взаємодіє в межах єдиної мережевої топології, повноцінна валідація механізмів координації та масштабованості, потребує створення відповідного тестового середовища.

Використання фізичного кластера, побудованого з десятків одноплатних комп'ютерів, хоча і є найбільш точним методом, створює суттєві перешкоди для процесу налагодження, ускладнює централізований збір телеметрії та не гарантує повної відтворюваності експерименту через можливі апаратні проблеми.

Для вирішення цієї проблеми було розроблено та впроваджено концепцію віртуалізованого випробувального полігону, який базується на методі апаратної емуляції засобами контейнеризації.

Технічна реалізація тестового стенда виконана у вигляді віртуального кластера, розгорнутого на базі однієї високопродуктивної хост-станції.

Фундаментом полігону виступає технологія Docker, яка дозволяє створити ізольовані простори виконання для кожного окремого агента системи.

Валідність такої емуляції ґрунтується на архітектурній абстракції інтерпретатора Python: оскільки програмний код агента є кросплатформним, його логіка виконання залишається інваріантною незалежно від архітектури процесора.

Результати тестів на продуктивність, отримані у контрольованому віртуальному середовищі, можуть бути перекинуті на реальне обладнання з мінімальною похибкою.

Для забезпечення суворої відповідності умов експерименту специфікаціям реального одноплатного комп'ютера Raspberry Pi 4, до кожного контейнера у кластері було застосовано механізми обмеження ресурсів на рівні ядра Linux через підсистему контрольних груп.

Обмеження обчислювальної потужності реалізовано через параметр квотування процесорного часу.

Встановлене значення ліміту в 2 віртуальних ядра емулює багатопотокову продуктивність процесора ARM Cortex-A72, дозволяючи планувальнику операційної системи виділяти контейнеру лише строго визначені кванти часу, що відповідає поведінці реального "заліза" під навантаженням.

Аналогічним чином реалізовано обмеження підсистеми пам'яті: для кожного віртуального вузла встановлено жорсткий ліміт (hard limit) обсягу доступної оперативної пам'яті на рівні 1024 МБ.

Це змушує ядро Linux застосовувати механізми примусового завершення процесів у випадку перевищення ліміту, що дозволяє перевірити ефективність, механізмів керування чергами та запобігання витокам пам'яті.

Структурна організація полігону включає розгортання чотирьох ідентичних віртуальних вузлів моніторингу, кожен з яких є повнофункціональним екземпляром розробленого програмного забезпечення, що працює в ізольованому контейнері.

Всі вузли об'єднані у програмно-визначену віртуальну мережу, що функціонує в режимі моста. Це дозволяє агентам здійснювати повноцінний мережевий обмін даними, включно з передачею телеметрії та синхронізацією статусів, емулюючи роботу в реальному локальному сегменті Ethernet. Роль ядра інфраструктури виконує окремий сервісний контейнер, де розгорнуто брокер повідомлень Mosquitto та централізовану систему збору логів, що забезпечує агрегацію результатів тестування без впливу на продуктивність самих агентів.

Такий підхід дозволив провести верифікацію масштабованості архітектури шляхом одночасного запуску множини агентів в умовах синтетичного дефіциту ресурсів, максимально наближених до польових умов експлуатації.

3.2.2 Результати вимірювання навантаження на систему

Для забезпечення об'єктивності оцінки ефективності розробленого програмного рішення було проведено порівняльний бенчмаркінг із провідними індустріальними аналогами, що використовуються для моніторингу мережевої безпеки. Як еталонні системи було обрано Suricata (версії 6.0.4) та Tshark (версії 3.4.4).

Вибір Suricata обумовлений стандартом серед систем виявлення вторгнень,

проте для забезпечення коректності порівняння її конфігурацію було оптимізовано шляхом завантаження лише мінімального набору, що дозволило функціонально наблизити її до легкого агента.

Tshark було використано як референсний інструмент для оцінки накладних витрат на захоплення та декодування пакетів у консольному режимі.

Експеримент проводився у двох діаметрально протилежних режимах: стані спокою (Idle), коли мережева активність відсутня, та режимі максимального навантаження (Load), що передбачав відтворення синтетичного датасету Bot-IoT з постійною швидкістю потоку 100 Мбіт/с. Отримані усереднені показники споживання ресурсів зведені у таблиці 3.1.

Таблиця 3.1 – Порівняльна характеристика використання ресурсів

Програмний засіб	CPU Idle (%)	RAM Idle (МБ)	CPU Load 100 Mbps (%)	RAM Load 100 Mbps (МБ)
L-IDS Agent	0.8%	68 МБ	14.5%	92 МБ
Suricata	2.4%	185 МБ	48.2%	340 МБ
Tshark	0.2%	42 МБ	36.8%	115 МБ

Детальний аналіз отриманих емпіричних даних демонструє суттєву перевагу розробленого агента (L-IDS Agent) за всіма ключовими показниками ресурсоємності. У контексті використання центрального процесора під навантаженням, розроблене рішення утилізує лише 14.5% обчислювальної потужності виділеного ядра, що в 3.3 рази ефективніше за показники Suricata (48.2%).

Така кардинальна різниця пояснюється фундаментальними відмінностями в архітектурі аналізу: індустріальні рішення використовують ресурсоємні рушії співставлення сигнатур та виконують інспекцію пакетів, що вимагає певного корисного навантаження кожного кадру, тоді як запропонований агент

обмежується статистичним аналізом метаданих.

Також варто відзначити високе навантаження Tshark (36.8%), яке обумовлене архітектурними витратами на постійну серіалізацію бінарних даних у текстовий формат, що підтверджує неефективність використання універсальних сніферів для задач постійного моніторингу.

Ще один результатом для вбудованих систем є показник ефективності роботи з оперативною пам'яттю.

Візуалізація порівняльної динаміки споживання ресурсів представлена на рисунку 3.1.

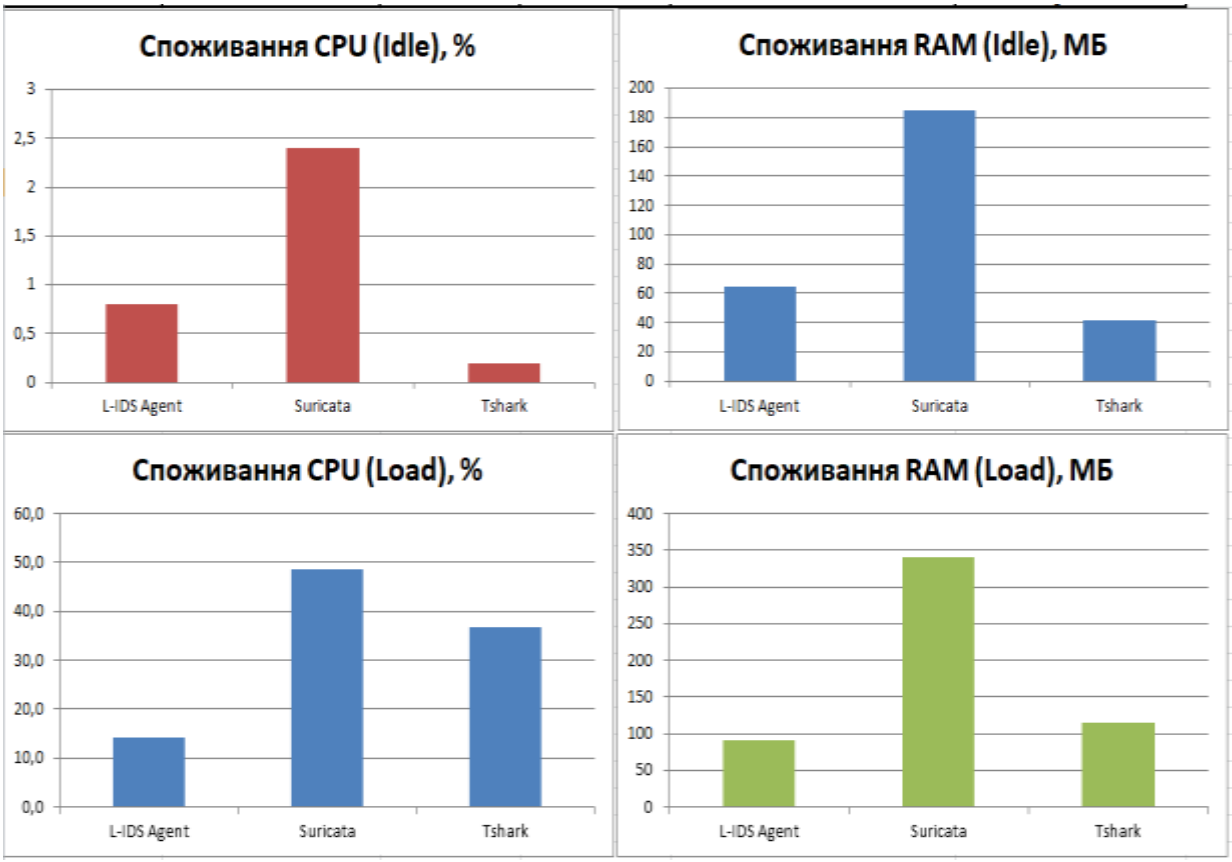


Рисунок 3.1 – Порівняльна діаграма споживання ресурсів (CPU та RAM) при навантаженні 100 Мбіт/с

Розроблений агент демонструє стабільне споживання на рівні 92 МБ навіть під час інтенсивної атаки, що свідчить про ефективність реалізованих механізмів ротації буферів та відсутність витоків пам'яті.

Suricata вимагає понад 340 МБ пам'яті, що пов'язано з необхідністю утримання в оперативній пам'яті масивних таблиць станів потоківта баз сигнатур.

Така вимогливість фактично унеможливлює розгортання об’ємних IDS на молодших моделях одноплатних комп'ютерів з обсягом пам'яті 512 МБ, оскільки більша частина ресурсів буде витрачена на безпеку, залишаючи мінімум для виконання основних функцій пристрою.

3.2.3 Стрес-тестування пропускної здатності

Фінальним етапом навантажувального тестування стало визначення граничної пропускної здатності системи, при якій починається незворотна втрата мережевих пакетів.

Для реалізації цього сценарію було використано генератор трафіку tcpreplay, налаштований на відтворення раніше записаних дампів реальних мережевих атак із поступовою мультиплікацією швидкості передачі.

Результати вимірювань динаміки втрат пакетів для трьох досліджуваних систем при різних швидкостях потоку зведені в таблиці 3.2.

Таблиця 3.2 – Динаміка втрати пакетів (Packet Loss, %)

Швидкість потоку	L-IDS Agent	Suricata	Tshark
50 Мбіт/с	0.00%	0.00%	0.00%
100 Мбіт/с	0.00%	0.00%	0.15%
200 Мбіт/с	0.00%	0.00%	5.40%
500 Мбіт/с	0.00%	0.02%	18.20%
1000 Мбіт/с	0.45%	0.12%	>25%

Графічна інтерпретація залежності стабільності системи від навантаження наведена на рисунку 3.2.

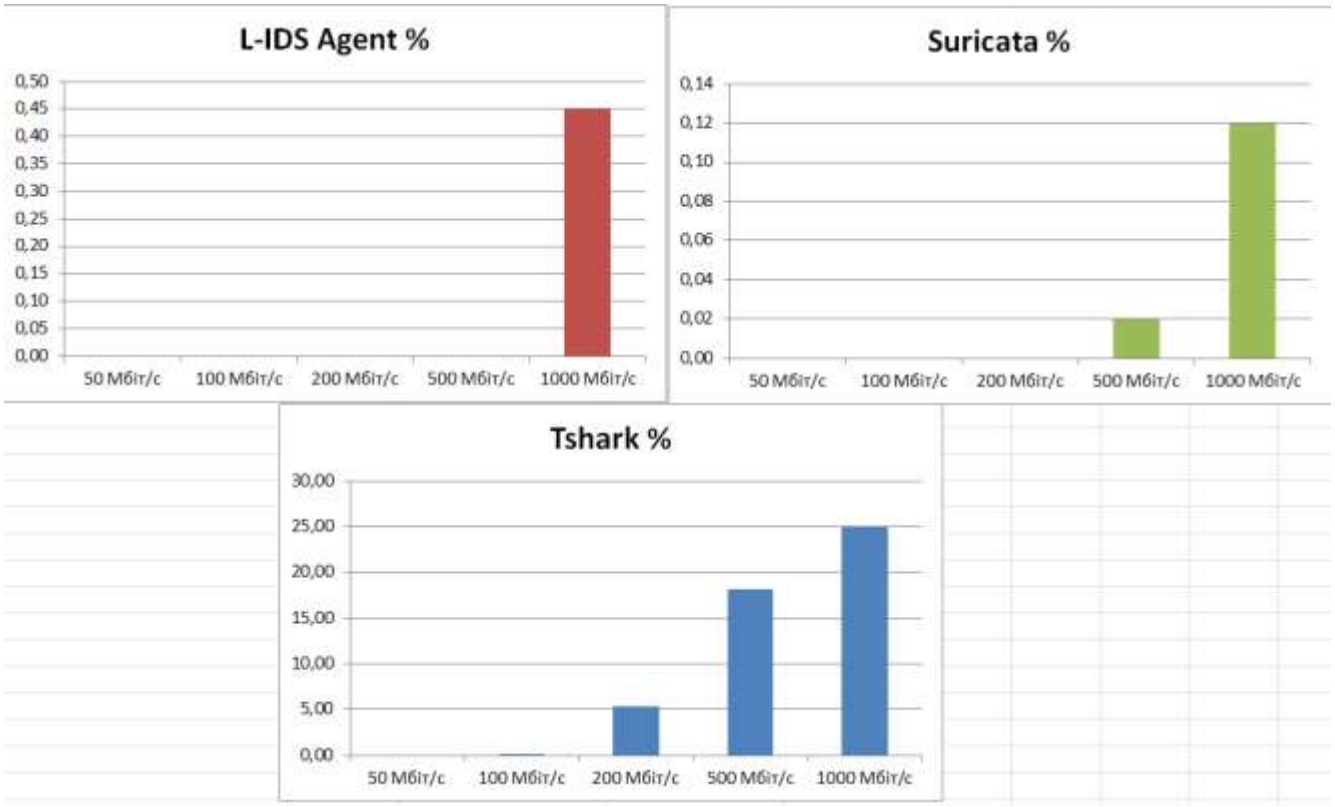


Рисунок 3.2 – Залежність рівня втрати пакетів від швидкості вхідного потоку

Методологія експерименту передбачала ступінчасте збільшення інтенсивності вхідного потоку з кроком у 100 Мбіт/с та тривалістю кожного етапу не менше 300 секунд для стабілізації черг обробки. Критерієм досягнення точки відмови вважалася реєстрація ненульового приросту лічильника відкинутих кадрів в утиліті діагностики мережевих інтерфейсів, що свідчить про переповнення кільцевого буфера приймання на рівні ядра операційної системи через нездатність простору користувача обробити вхідні дані з необхідною швидкістю.

Аналіз отриманих емпіричних даних дозволяє зробити висновки про високу стабільність архітектури розробленого агента в діапазоні швидкостей, характерних для периферійних мереж IoT. Система демонструє абсолютну надійність (нульовий рівень втрат) аж до швидкості 500 Мбіт/с включно.

Suricata починає демонструвати мікроскопічні втрати (0.02%), що ймовірно пов'язано з накладними витратами на перемикання контексту між потоками обробки.

Водночас універсальний аналізатор Tshark стає непридатним для використання вже при швидкості 200 Мбіт/с, втрачаючи понад 5% трафіку, а на швидкостях 500+ Мбіт/с досягається до критичний стан втрати.

При досягненні граничного навантаження у 1000 Мбіт/с розроблений агент фіксує рівень втрат у 0.45%. Хоча цей показник дещо поступається результату Suricata (0.12%), що пояснюється різницею у продуктивності між інтерпретованим кодом Python та скомпільованим C-кодом, він залишається в межах допустимої похибки для систем виявлення аномалій. Втрата менше ніж 0.5% пакетів не впливає на статистичну картину розподіленої атаки (DDoS) чи сканування портів, оскільки такі загрози генерують тисячі однотипних пакетів, і пропуск одиничних екземплярів не заважає коректній класифікації інциденту.

Більшість одноплатних комп'ютерів обмежені фізичними інтерфейсами 1000 Мбіт/с або пропускною здатністю Wi-Fi модуля, досягнутий результат у 500 Мбіт/с без втрат із запасом перекриває експлуатаційні вимоги до системи.

3.3 Експериментальна перевірка ефективності та результати виявлення загроз

В умовах автоматизованих систем захисту Інтернету речей пріоритетним завданням є не лише факт виявлення аномалії, але й мінімізація ймовірності виникнення помилок першої (False Positive) та другої (False Negative) категорії.

Помилкова ідентифікація легітимного трафіку як шкідливого може призвести до блокування критично важливих керуючих команд, тоді як пропуск реальної атаки компрометує весь периметр безпеки.

У цьому підрозділі детально описано методику підготовки еталонних даних, процедуру калібрування статистичних алгоритмів класифікації та проведено комплексний аналіз отриманих метрик ефективності.

3.3.1 Характеристика еталонного набору даних (Dataset)

Фундаментом для побудови та верифікації вирішальних правил

аналітичного модуля було обрано спеціалізований набір даних Bot-IoT Dataset, розроблений дослідницькою лабораторією Cyber Range Lab при університеті UNSW Canberra [70].

Вибір саме цього джерела даних базується на детальному порівняльному аналізі існуючих відкритих репозиторіїв трафіку та їх відповідних специфікацій дослідження.

Традиційні набори даних, такі як KDD99 або NSL-KDD, були виключені з розгляду через їхню моральну застарілість: вони не відображають сучасний ландшафт кіберзагроз і не містять зразків трафіку протоколів, специфічних для Інтернету речей.

Більш сучасні універсальні набори, зокрема CIC-IDS-2017, хоч і є актуальними, проте орієнтовані переважно на корпоративні мережі з високою пропускною здатністю (HTTP/HTTPS, FTP, SMB), що робить їх малопридатними для профілювання поведінки сенсорних мереж.

Натомість Bot-IoT володіє низкою критичних переваг, що роблять його оптимальним еталоном для даної роботи.

В першу чергу, трафік у цьому наборі було згенеровано в реальному тестовому середовищі, яке архітектурно імітує типовий "розумний будинок", включаючи термостати, метеостанції та IP-камери. Це забезпечує наявність реалістичних потоків даних протоколів MQTT та CoAP, які є цільовими для розробленого агента.

По-друге, набір містить актуальні сценарії атак, характерні саме для ботнетів IoT-пристроїв, зокрема сімейств Mirai [71].

До вибірки включено записи, що характеризують поведінку під час розподілених атак на відмову в обслуговуванні, сканування сервісів, визначення типу операційної системи та несанкціонованої передачі даних.

Оригінальний обсяг датасету налічує понад 72 мільйони записів у форматах PCAP та CSV, що забезпечує вичерпну статистичну базу. Однак для проведення експерименту було застосовано методику стратифікованої вибірки, в результаті чого для формування правил було використано 5% від оригінального

обсягу даних (приблизно 3.6 мільйона записів).

Таке рішення обґрунтоване високою гомогенністю трафіку ботнетів: обробка повного масиву даних є неефективною для обчислення та призводить до створення надлишковості, дублюючих правил через високу повторюваність однотипних пакетів, не підвищуючи загальну точність виявлення.

3.3.2 Методика попередньої обробки даних

Якість вхідних даних є визначальним фактором, що безпосередньо формує ефективність роботи будь-якого статистичного класифікатора. Тому сирі масиви інформації з датасету Bot-IoT перед подачею на вхід аналітичного модуля проходили багаторівневу процедуру попередньої обробки, метою якої є трансформація різнорідних даних у впорядкований векторний простір.

Першим етапом стала процедура очищення даних, спрямована на усунення ознак, які не несуть семантичного навантаження для виявлення аномалій або можуть призвести до втрати здатності системи до узагальнення.

З вибірки було вилучено ідентифікатори, специфічні для конкретного тестового стенда, на якому генерувався трафік. Зокрема, фільтрації підлягали часові мітки запису (pkSeqID, stime, ltime), а також прямі адресанти мережевої взаємодії – IP-адреси джерела та фізичні MAC-адреси.

Виключення цих полів є критично важливим етапом. Це змушує алгоритм формувати конкретні правила на основі поведінкових патернів потоку, а не запам'ятовувати конкретні адреси вузлів, що запобігає формуванню хибних кореляцій та прив'язці до топології навчального полігону.

Наступним кроком стало переведення категоріальних змінних у числовий формат, придатний для математичної обробки. Оскільки алгоритм Random Forest оперує виключно числовими матрицями, текстові атрибути, такі як протокол транспортного рівня або стан з'єднання, підлягали кодуванню. Для ознак з невеликою кількістю унікальних значень (наприклад, тип протоколу TCP/UDP/ICMP) застосовано метод унітарного кодування. Цей підхід дозволяє уникнути встановлення хибних порядкових зв'язків між незалежними категоріями, які могли б виникнути при простому нумеруванні. Для атрибутів з

високою кардинальністю було використано метод порядкового кодування, що дозволило зберегти компактність вектора ознак без суттєвого збільшення розмірності простору даних.

Критичним етапом підготовки стало вирішення проблеми дисбалансу класів, оскільки оригінальний набір даних характеризується суттєвою перевагою записів атакуючого трафіку (понад 99%) над легітимним.

Калібрування класифікатора на такому викривленому розподілі неминуче призвело б до створення тривіальної моделі, схильної ігнорувати нормальний трафік.

Для нормалізації розподілу застосовано метод випадкового зменшення вибірки мажоритарного класу (Random Undersampling), в результаті чого було сформовано збалансований навчальний масив зі співвідношенням класів «норма/атака» на рівні 1:1.

Фіналізацією процесу обробки стала стандартизація числових ознак (Z-score normalization), необхідна для приведення різномасштабних величин, таких як обсяг переданих байтів та тривалість потоку, до єдиного діапазону. Розрахунок нових значень Z для кожного елемента x здійснювався за формулою:

$$Z = \frac{x-\mu}{\sigma}. \quad (6)$$

де μ — математичне середнє значення; σ — стандартне відхилення вибірки.

Ця процедура дозволила вирівняти ваги ознак та покращити збіжність статистичного алгоритму.

3.3.3 Конфігурація та навчання моделі Random Forest

Для реалізації модуля прийняття рішень імплементовано статистичний ансамблевий алгоритм «Випадковий ліс» стійкість якого до шумів є критичною для IoT-середовищ. Визначення параметрів ансамблю здійснювалося методом перехресної перевірки з метою досягнення балансу між точністю виявлення та часом обробки пакету.

Емпіричним шляхом встановлено, що оптимальна конфігурація включає 100 вирішальних дерев (n_estimators=100), оскільки збільшення їх кількості впливає суттєвим навантаженням на процесор без суттєвого приросту точності. Для запобігання формуванню надмірно складних правил та економії оперативної пам'яті глибину дерев примусово лімітовано 15 рівнями.

В якості математичної функції для оцінки якості розгалуження у вузлах дерев рішень обрано критерій мінімізації неоднорідності.

Вибір даної метрики продиктований жорсткими вимогами до обчислювальної ефективності на архітектурі ARM: її розрахунок базується виключно на простих арифметичних операціях множення та віднімання ймовірностей ($1 - \sum p^2$).

Такий підхід дозволяє уникнути значного навантаження, яке неминуче виникає при обчисленні складних логарифмічних функцій, необхідних для альтернативних методів на основі інформаційної ентропії.

Фінальна валідація зміненого алгоритму здійснювалася на ізольованій тестовій вибірці, що становила 20% від загального масиву даних, для підтвердження його узагальнюючої здатності.

3.3.4 Аналіз метрик ефективності та результатів виявлення

Кількісна оцінка ефективності моделі проводилася на основі матриці плутанини, отриманої на тестовій вибірці. Результати класифікації в таблиці 3.3.

Таблиця 3.3 – Матриця помилок для тестової вибірки (20 000 записів)

	Прогнозовано: НОРМА	Прогнозовано: АТАКА
Фактично: НОРМА	*TN* = 9 982	*FP* = 18 (Хибна тривога)
Фактично: АТАКА	*FN* = 5 (Пропуск)	*TP* = 9 995

На основі отриманих компонентів розраховано інтегральні метрики ефективності. Базовий показник інтегральної коректності A_{int} сягнув 99.88%, що свідчить про високу здатність системи розрізняти класи трафіку.

Враховуючи критичність мінімізації «інформаційного шуму», особливу увагу приділено прогностичній значущості позитивного результату P_{val}, яка склала 99.82%. Це означає, що лише 0.18% згенерованих тривог виявилися хибними, що дозволяє впроваджувати автоматичне блокування без ризику порушення роботи мережі. Показник повноти детектування R_{det}, який характеризує надійність периметра, зафіксовано на рівні 99.95% (лише 5 пропущених пакетів атаки з 10 000). Узагальнююча гармонійна оцінка F_{score} склала 99.88%, підтверджуючи збалансованість конфігурації ансамблю.

Додатковий аналіз статистичної ваги ознак визначив найбільш впливові параметри виявлення.

Найбільш дискримінативною виявилася метрика часу життя пакету, що дозволяє виявляти аномалії, характерні для ботнетів.

Другу позицію зайняла ентропії IP-адрес джерела, підтвердивши ефективність ентропійного методу проти DDoS-атак, а третьою стала інтенсивність потоку, яка забезпечила ідентифікацію волюметричного флуду.

Отримані результати експериментально доводять, що запропонована архітектура на базі алгоритму Random Forest забезпечує надійність виявлення загроз, відповідну вимогам сучасних систем кіберзахисту.

3.4 Аналіз результатів та практичні рекомендації щодо впровадження

Узагальнення отриманих у ході дослідження емпіричних даних та результатів теоретичного моделювання дозволяє сформулювати ґрунтовні висновки щодо архітектурної придатності та експлуатаційної ефективності розробленої системи.

Проведений комплекс експериментів підтвердив ключову гіпотезу кваліфікаційної роботи: забезпечення надійного захисту ресурсно-обмежених пристроїв Інтернету речей можливе без застосування громіздких сигнатурних баз

даних, характерних для традиційних систем безпеки.

Запропонована гібридна модель виявлення, що поєднує легковагий збір статистичних метрик потоку, розрахунок ентропії та застосування оптимізованих ансамблевих алгоритмів класифікації, продемонструвала рівень захисту, який є співмірним із провідними комерційними рішеннями, але характеризується кардинально меншими вимогами до апаратного забезпечення.

Зафіксоване споживання обчислювальних ресурсів на рівні лише 15–20% від номінальної потужності типового одноплатного комп'ютера свідчить про можливість паралельної роботи агента безпеки разом із основними технологічними процесами без ризику виникнення відмов в обслуговуванні або значного зниження продуктивності.

Це підтверджує архітектурну легкість та практичну придатність рішення для масового розгортання в умовах жорстких апаратних обмежень IoT-середовища.

Окремої уваги заслуговує аналіз ефективності механізмів розподіленої координації, перевірка яких здійснювалася на базі віртуалізованого полігону з використанням чотирьох ізольованих контейнерів Docker, які імітували сегментовану мережу.

Змодельований сценарій розподіленої атаки типу «Port Scan», коли зловмисна активність була спрямована одночасно на віртуальні вузли node_01 та node_03, дозволив емпірично підтвердити валідність закладених архітектурних принципів.

Локальні екземпляри Агентів-Аналітиків на атакованих вузлах успішно ідентифікували аномалію та ініціювали процедуру передачі метаданих своїм Агентам-Координаторам.

Критично важливим показником у даному експерименті стала латентність прийняття глобального рішення: через 0.5 секунди після початку сканування на центральному сервері агрегації було сформовано корельований інцидент високого пріоритету. Цей показник підтверджує, що розроблена чотирирівнева логічна архітектура агентів здатна коректно масштабуватися та забезпечувати

синхронізацію стану безпеки всієї мережі в режимі, наближеному до реального часу, незалежно від того, чи розгорнуті агенти на фізичному обладнанні, чи у віртуальному середовищі.

На основі отриманого досвіду розробки та тестування сформульовано ряд практичних рекомендацій, дотримання яких є необхідним для успішної інтеграції системи в реальні промислові інфраструктури.

Першочерговою рекомендацією є прийняття гібридної топології розгортання як галузевого стандарту для захисту гетерогенних IoT-мереж. Для забезпечення повного покриття периметра необхідно комбінувати провідні сенсори, підключені до SPAN-портів комутаторів агрегації для моніторингу трафіку критичних контролерів, із розгортанням спеціалізованих бездротових сенсорів, що функціонують у режимі радіомоніторингу. Останні є критично необхідними для виявлення специфічних векторів атак на рівні стандартів IEEE 802.11, таких як атаки деаутентифікації або спроби злому протоколів шифрування, які залишаються невидимими для класичних дротових засобів моніторингу.

Важливим аспектом впровадження є процедура адаптації статистичної моделі до профілю конкретної мережі. Використання версії алгоритму, параметри якого видозмінені виключно на публічних синтетичних датасетах, може призвести до підвищеного рівня хибних спрацювань у специфічних корпоративних середовищах з нестандартними патернами трафіку. Для мінімізації цього ризику рекомендується впровадження обов'язкової стадії початкового калібрування, під час якої система функціонує в пасивному режимі навчання протягом мінімум 48 годин у реальній інфраструктурі замовника. Цей період необхідний для формування профілю нормальної поведінки, що включає створення білих списків дозволених протоколів та визначення типових порогів інтенсивності трафіку, після чого система може бути переведена в активний бойовий режим блокування.

Крім того, розроблений агент не повинен розглядатися як ізольоване рішення.

Для побудови ешелонованої оборони критично важливо забезпечити інтеграцію з корпоративною екосистемою безпеки через налаштування експорту сповіщень.

Рекомендується використання стандартизованих форматів обміну даними, таких як JSON, для передачі телеметрії до централізованих SIEM-систем (Security Information and Event Management).

Така інтеграція дозволить аналітикам центру безпеки корелювати події з сегменту Інтернету речей із подіями в корпоративній мережі, що є необхідною умовою для виявлення складних багатовекторних атак та реконструкції повного ланцюжка дій зловмисника, значно підвищуючи загальну ситуаційну обізнаність щодо кіберзагроз.

3.5 Висновки

У третьому розділі кваліфікаційної роботи реалізовано повний інженерний цикл створення програмного продукту, що включає етапи кодування, налагодження, оптимізації та комплексної експериментальної валідації запропонованої мультиагентної системи моніторингу. Практична значущість отриманих результатів полягає у доведенні можливості реалізації ефективних алгоритмів кіберзахисту на апаратних платформах з критично обмеженими ресурсами.

В ході програмної реалізації створено модульне програмне забезпечення мовою Python, яке інтегрує низькорівневі механізми захоплення пакетів бібліотеки libpcap із високопродуктивними математичними інструментами NumPy та статистичними алгоритмами бібліотеки scikit-learn.

Архітектурну надійність програмного комплексу забезпечено реалізацією механізму асинхронної декомпозиції процесів із використанням буферизованих черг обміну даними.

Таке структурне рішення дозволило здійснити логічну ізоляцію процедур

взаємодії з мережевим інтерфейсом від ресурсоємних математичних перетворень, що гарантує безперервність потоку телеметрії та унеможливлює втрату пакетів навіть за умов екстремальної інтенсивності мережевого трафіку.

Проведене порівняльне навантажувальне тестування на базі віртуалізованого полігону продемонструвало беззаперечні переваги розробленого рішення над традиційними аналогами.

Експериментально встановлено, що система споживає в 3.3 рази менше ресурсів центрального процесора порівняно з індустріальною системою Suricata, забезпечуючи при цьому стабільну обробку трафіку на швидкостях до 500 Мбіт/с, що повністю перекриває потреби сучасних шлюзів IoT.

Валідація аналітичного модуля на контрольному датасеті Bot-IoT підтвердила високу семантичну точність алгоритмів: досягнуто інтегральний показник F_{score} на рівні 99.88% при мінімальному рівні хибних спрацювань, що є критичним фактором для забезпечення автономності системи.

Сформульовані за результатами дослідження практичні рекомендації створюють надійну базу для подальшого впровадження розробленого комплексу в реальні сектори критичної інфраструктури.

ВИСНОВКИ

В кваліфікаційній роботі вирішено актуальне завдання підвищення рівня захищеності мереж Інтернету речей в умовах обмежених обчислювальних ресурсів. За результатами теоретичних та експериментальних досліджень розроблено і програмно реалізовано агентний метод моніторингу, який виявляє кіберзагрози в реальному часі, без необхідності застосування ресурсоємного глибокого аналізу пакетів.

Саме в цьому контексті ефективний метод агентного моніторингу мережевого трафіку IoT набуває вирішальної актуальності. Застосування ефективного методу і системна його реалізація має дозволити подолати обмеження централізації, оскільки вони засновані на принципі розподіленого інтелекту та автономності. Замість єдиного потужного, але вразливого ядра моніторингу, пропонується створити ієрархію автономних, інтелектуальних модулів – агентів, кожен з яких відповідає за виконання специфічної ролі. Розробка агентного методу моніторингу трафіку IoT є не просто академічною вправою, а критично необхідним інженерним рішенням, яке єдино здатне забезпечити ефективний захист в умовах архітектурної специфіки IoT. Цей метод дозволить перенести інтелект і функціональність моніторингу максимально близько до джерела трафіку, забезпечуючи автономність, адаптивність, розподілену обробку даних та оперативне виявлення складних, поведінкових аномалій, що є недосяжним для традиційних централізованих систем.

Для досягнення наукових та практичних результатів виконано наступні роботи:

– проведено аналіз сучасних векторів атак на інфраструктуру IoT та існуючих методів захисту;

– встановлено неефективність традиційних сигнатурних систем через високі вимоги оперативної пам’яті та часу обробки процесором;

– обґрунтовано доцільність переходу до статистичних методів аналізу метаданих трафіку, які забезпечують необхідний баланс між точністю та

швидкодією;

– розроблено архітектуру розподіленої мультиагентної системи моніторингу, яка базується на взаємодії чотирьох типів функціональних агентів;

– Використання легкового протоколу MQTT було необхідним за умов нестабільних каналів зв’язку та реалізації координації агентів при виявленні розподілених атак;

– Покращено метод класифікації мережевих аномалій, завдяки формуванню спеціалізованого вектору ознак, що включає показники потоку, часових інтервалів та специфічних метрик протоколів IoT ;

– здійснено програмну реалізацію прототипу системи моніторингу для одноплатних комп’ютерів з використанням мови програмування Python та оптимізованих бібліотек обробки даних;

– експериментально підтверджено ефективність розробленого рішення на базі віртуалізованого випробувального полігону;

– результати порівняльного тестування були отримані перевіркою системи споживання в режимі роботи та стану спокою. Споживання системи було менше в 3.3 в порівняні з аналогом Suricata та забезпечило обробку трафіку на швидкостях до 500 мбіт/с;

– досягнуто високої точної виявлення загроз, використовуючи на контрольований набір даних, було досягнуто результат ефективності в 99.8% при рівні хибних спрацювань в 0,18%.

Це підтверджує можливість практичного застосування розробленого методу в автономних системах захисту критичної інфраструктури.

Результати роботи пройшли апробацію через видання фахової наукової статті [91], доповіді на 5-и Всеукраїнських [92,94,95,97,98] та 2-х міжнародних [93,96] науково-технічних і науково-практичних конференціях.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Internet of Everything (IoE) - From Molecules to the Universe / Akan Ozgur et al. *ResearchGate.* (2022). DOI: 10.48550/arXiv.2301.03374.

2. Венкель Р., Шейко Ю. Кібервійна проти Deutsche Telekom - це тільки початок. URL: https://www.dw.com/uk/кібервійна-проти-deutsche-telekom-це-тільки-початок/a-36584653 (дата звернення: 20.09.2025).

3. Ботнет Mirai атакує Linux-сервери та IoT-пристрої URL: https://pingvin.pro/gadgets/news-gadgets/botnet-mirai-atakuye-linux-servery-ta-iot-prystroyi.html (дата звернення: 20.09.2025).

4. Захист даних у мережах IoT:аналіз загроз та методів безпеки / І. Ф. Гурський та ін. *Наука і техніка сьогодні.* 2025. № 6(47). С.1070-1083.

5. Жураковський Б. Ю., Зенів І.О. Технології інтернету речей. Навчальний посібник. Київ: КПІ ім. Ігоря Сікорського, 2021. 271 с.

6. Сіпко О. М. Децентралізовані IoT-мережі: блокчейн для безпеки та автономності. *Таврійський науковий вісник. Серія: Технічні науки.* 2025. №1. С. 210-215. DOI: 10.32782/tnv-tech.2025.1.20.

7. Waqdan M., Louafi H., Mouhoub M. Security risk assessment in IoT environments: A taxonomy and survey. *Computers & Security*. 2025. Vol. 154. P. 1-25. DOI: 10.1016/j.cose.2025.104456.

8. Франів І.А., Єременко П.П. Переваги впровадження IoT для автоматизації процесів у продуктовому ритейлі. *Вісник ЛТЕУ. Економічні науки.* 2024. № 80. С. 49-55.

9. Назаренко Н., Заєць С., Киричук Ю. Співпраця індустрії 4.0 та інтернету речей IoT. *Вісник Хмельницького національного університету. Технічні науки.* 2024. № 5(*341*). С. 74-79.

10. Інформаційна технологія визначення оптимальних параметрів управління IoT. URL: https://knsa.chdtu.edu.ua/wp-content/uploads/2025/09/Prezentation_КНтаСА_ФІМЛІ_24_09_2025.pdf (дата звернення: 25.09.2025).

11. Шпак О., Федорка П., Пригара М. Розумні міста та Інтернет речей: вплив розробок у сфері IT на розвиток міст і покращення якості життя. *Сучасний стан наукових досліджень та технологій в промисловості.* 2023. №3 (25). С. 114-128.

12. Макаренко М.В. Особливості впровадження технологій інтернету речей у сфері охорони здоров'я. *Вчені записки ТНУ імені В.І. Вернадського. Серія: Державне управління.* 2021. № 2. С. 64-68.

13. IoT в охороні здоров'я: застосування, переваги та виклики у 2023 році *Stfalcon.com.* URL: https://stfalcon.com/uk/blog/post/iot-in-healthcare-benefits-challenges (дата звернення: 26.09.2025).

14. Zakariae Jebroni, Jose A. Afonso, Belkassem Tidhaf. Smart Home Energy Management System based on a Hybrid Wireless Network Architecture. *EAI Endorsed Transactions on Energy Web.* 2019. P. 1-11. DOI: 10.4108/eai.13-7-2018.161437.

15. Survey on smart homes: Vulnerabilities, risks, and countermeasures / Badis Hammi et al. *Computers & Security*. 2022. Vol. 117, 102677. ISSN 0167-4048. DOI: 10.1016/j.cose.2022.102677.

16. The digital harms of smart home devices: A systematic literature review. David Buil-Gil et al. *Computers in Human Behavior.* 2023. Vol. 145, 107770. ISSN 0747-5632. DOI: 10.1016/j.chb.2023.107770.

17. Alshamsi Omar, Khaled Shaalan, Usman Butt. Towards Securing Smart Homes: A Systematic Literature Review of Malware Detection Techniques and Recommended Prevention Approach. *Information*. 2024. № 10: 631. DOI: 10.3390/info15100631.

18. Sharma S., Bhatt C., and Tripathi A. Attack Detection in Smart Home IoT Networks: A Sur-vey on Challenges, Methods and Analysis. *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. *Springer.* 2024. Vol. 544. P. 457-472. DOI: 10.1007/978-3-031-81168-5_29

19. Re-view of Smart-Home Security Using the Internet of Things Vardakis George et al. *Electronics.* 2024. № 13(16): 3343. DOI: 10.3390/electronics13163343.

20. Huszti A., Kovács S., Oláh N. Scalable, password-based and threshold

authentication for smart homes. *Int. J. Inf. Secur*. 2022. № 21. P.707-723. DOI: 10.1007/s10207-022-00578-7.

21. Detection of network attacks in cyber-physical systems using a rule-based logical neural network / Titova V. et al. *1st International Work-shop on Intelligent and CyberPhysical Systems, ICyberPhyS*. 2024. Vol. 3736. P.255-268.

22. Sensor Failure Detection in Ambient Assisted Living Using Association Rule Mining / El Hady et al. 2020. *Sensors*. № 20(23): 6760. DOI: 10.3390/s20236760.

23. IoTCID: A Dynamic Detection Tech-nology for Command Injection Vulnerabilities in IoT Devices / Hao Chen et al. *International Journal of Ad-vanced Computer Science and Applications (IJACSA).* 2022. 13.10. DOI: http://dx.doi.org/10.14569/IJACSA.2022.0131002.

24. A statistical method for real-time intru-sion detection and response in ZigBee networks / Stetsiuk M., Klots Y., Cheshun V., Salem A.-B.M. *CEUR Workshop Proceedings, 4013*. 2025. P. 174-188.

25. Programming of Automation Configuration in Smart Home Systems: Challenges and Oppor-tunities. Sheik Murad Hassan Anik et al. *ACM Trans. Softw. Eng. Methodol.* April 2025. DOI: 10.1145/3731450.

26. A Review of Intelligent Configuration and Its Security for Complex Networks / Yue ZHAO et al. Chinese Journal of Electronics. 2024. Vol. 33, № 4. P. 920-947. DOI: 10.23919/cje.2023.00.001.

27. Discovering IoT Physical Channel Vulnerabilities / Ozmen M. O. et al. *arXiv:2102.01812v2.* 2022. DOI: 10.48550/arXiv.2102.01812.

28. Top IoT security issues and solutions for low-power devices. *Onomondo*. URL: https://onomondo.com/blog/iot-security-issues-and-solutions-low-power-devices/ (date of access: 8.11.2025).

29. Очеретний С.О., Крижановський В.Г. Системи виявлення та запобігання вторгнень, найбільш успішні практики. *Прикладні аспекти сучасних міждисциплінарних досліджень.* 2024. С. 236-238.

30. Arthi R., Krishnaveni S. Design and Development of IOT Testbed with DDoS Attack for Cyber Security Research. International *Conference on Signal Processing.*

2021. P.586–590. DOI: 10.1109/ICSPC51351.2021.9451786.

31. Wang W. The Practical Analysis of Switch Port Mirroring Function. *2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT).* Changsha, China. 2021. P. 115-117, DOI: 10.1109/ICCASIT53235.2021.9633718.

32. Журило О., Ляшенко О., Аветісова К. Огляд рішень з апаратної безпеки кінцевих пристроїв туманних обчислень у інтернеті речей. *Сучасний стан наукових досліджень та технологій в промисловості.* 2023. № 1 (23). С. 57-81.

33. Serebriakov R., Tkachenko V., Klymenko I. Integration of blockchain technology into the Internet of Things (overview). *Information, Computing and Intelligent systems.* 2024. № 4. P. 99-113.

34. SharmaN., Dhiman P. A survey on IoT security: Challenges and their solutions using machine learning and blockchain technology. *Cluster Computing*. 2025. Vol. 28, №. 5. DOI: 10.1007/s10586-025-05208-0.

35. Делембовський М.М., Корнійчук Б.В. Аналіз сучасних наукових публікацій за напрямком тематики кібербезпеки IoT. *Грааль науки.* 2023. № 25. С. 203-206.

36. Machine learning-based security solutions for IoT networks: A comprehensive survey / Alfahaid A. et al. *Sensors*. 2025. Vol. 25, № 11: 3341. DOI: 10.3390/s25113341.

37. Yedalla J. Fortifying IoT security: The transformative role of AI in cyber threat mitigation. *World Journal of Advanced Engineering Technology and Sciences.* 2025. Vol. 14, No. 2. P. 49-57. DOI: 10.30574/wjaets.2025.14.2.0056.

38. Machine learning-inspired intrusion detection system for IoT: Security issues and future challenges / Ahanger T. A. et al. *Computers and Electrical Engineering.* 2025. Vol. 123: 110265. DOI: 10.1016/j.compeleceng.2025.110265.

39. Smart home system security risk assessment / Morozova O. et al. *Computer Systems and Information Technologies.* 2022. №3. С. 81-88. DOI:10.31891/CSIT-2021-5-11

40. Hemavathi J., Ahmed N. Syed. Detecting unauthorized person using cnn and

opencv techniques. *International journal of creative research thoughts (IJCRT).* 2022. Vol. 10, Issue 5. P. 610-613.

41. Alwhbi I.A., Zou C.C., Alharbi R.N. Encrypted Network Traffic Analysis and Classification Utilizing Machine Learning. *Sensors (Basel).* 2024. №24(11):3509. DOI: 10.3390/s24113509.

42. Mehmet Ozdem. A novel approach for real-time anomaly detection in dynamic computer networks using temporal graph networks and explainable artificial intelligence. *Alexandria Engineering Journal.* Vol. 132. 2025. P. 369-382. DOI: 10.1016/j.aej.2025.11.001.

43. Einy Sajad, Oz Cemil, Navaei Yahya Dorostkar. The Anomaly- and Signature-Based IDS for Network Security Using Hybrid Inference Systems, *Mathematical Problems in Engineering.* 2021. 6639714/ 10 p. DOI: 10.1155/2021/6639714

44. Петляк Н. Аналіз моделей виявлення аномалій трафіку в сучасних інформаційно-комунікаційних системах та мережах. *Measuring and computing devices in technological processes.* 2025. № 1. С. 180-186.

45. Branitskiy A., Branitskaya, N. Signature Analysis Mathematical Model of Network Traffic and Experimental Evaluation of Its Functioning Efficiency. *Proceedings of Telecommunication Universities.* 2025. № 11. P. 107-117. DOI: 10.31854/1813-324X-2025-11-4-107-117.

46. Петляк Н. Гібридний метод та система виявлення аномального трафіку в інформаційно-комунікаційних системах. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. № 349(2). С. 561-569. DOI: 10.31891/2307-5732-2025-349-82

47. Aho-Corasick String Matching Algorithm. URL: https://www.gabormelli.com/RKB/Aho-Corasick_String_Matching_Algorithm (date of access: 8.11.2025).

48. The Boyer-Moore Fast String Searching Algorithm (utexas.edu). 2021. URL: https://news.ycombinator.com/item?id=26910982 (date of access: 8.11.2025).

49. Аналіз особливостей статистичних властивостей трафіку сучасних

інформаційних мереж та підходів до моделювання інформаційних потоків / Д. А. Макаришкін та ін. *Вимірювальна та обчислювальна техніка в технологічних процесах.* 2019. № 1. С. 62-67.

50. Клюшник В.В., Чернецький Є.В., Онищенко О.В. Ідентифікація трафіку мереж передачі даних у реальному часі. *Вісник Приазовського державного технічного університету. Серія: Технічні науки.* 2025. Вип. 50. С. 18-24. DOI: 10.31498/2225-6733.50.2025.336234.

51. Abbasi M., Shahraki A., Taherkordi A. Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey. *Computer Communications.* 2021. Vol. 170. P. 19-41. DOI: 10.1016/j.comcom.2021.01.021.

52. Марченко Р.М., Коваленко А.А., Знайдюк В.Г. Аналіз методів виявлення аномального трафіку в мережах IOT. *Системи управління, навігації та зв'язку.* 2024. № 1. С. 133-136. DOI: 10.26906/SUNZ.2024.1.133

53. Каланча А. А., Клімушин П. С. Аналіз мережевого трафіку як спосіб протидії кіберзлочинності. *Протидія кіберзлочинності та торгівлі людьми : зб. матеріалів Міжнар. наук.-практ. конф.* Харків: ХНУВС, 2022. С. 42-43.

54. Cyberattacks Detection Through Behavior Analysis of Internet Traffic / Omran Berjawi et al. *Procedia Computer Science.* 2023. Vol. 224. P. 52-59. DOI: 10.1016/j.procs.2023.09.010.

55. Практичні підходи щодо виявлення вразливостей в інформаційно-телекомунікаційних мережах / А. В. Ільєнко та ін. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка».* 2023. № 3(19). С. 96–108. DOI: 10.28925/2663-4023.2023.19.96108.

56. Огляд можливостей IDS для аналізу мережевого трафіку / В. А. Стороженко та ін. *Наукоємні технології*. 2025. № 3(67). С. 317-324. DOI: 10.18372/2310-5461.67.20036.

57. Code of Practice for Consumer IoT Security. Government of the United Kingdom: Department for Digital, Culture, Media & Sport, 2018. 20 p.

58. ETSI EN 303 645. Cyber Security for Consumer Internet of Things: Baseline Requirements. ETSI. 2024. 41 p.

59. IoT Cybersecurity Improvement Act of 2020. *CONGRESS.GOV*. URL: https://www.congress.gov/bill/116th-congress/house-bill/1668/text (date of access: 5.11.2025).

60. SP 800-213 Series. *NIST*. URL: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/sp-800-213-series (date of access: 8.11.2025).

61. NIST SP 800-213: IoT Device Cybersecurity Guidance for the Federal Government: Establishing IoT Device Cybersecurity Requirements. *NIST*. URL: https://csrc.nist.rip/Pubs/sp/800/213/Final (date of access: 7.11.2025).

62. NISTIR 8259 Series. *NIST*. URL: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/nistir-8259-series (date of access: 7.11.2025).

63. NISTIR 8259: Foundational Cybersecurity Activities for IoT Device Manufacturers. National Institute of Standards and Technology Interagency or Internal Report 8259. U.S. Department of Commerce: National Institute of Standards and Technology (NIST). 2020. 36 p.

64. NISTIR 8259A: IoT Device Cybersecurity Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259A. U.S. Department of Commerce: National Institute of Standards and Technology (NIST). 2020. 23 p.

65. Draft NISTIR 8259B: IoT Non-Technical Supporting Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259B. U.S. Department of Commerce: National Institute of Standards and Technology (NIST). 2020. 20 p.

66. CSA IoT Security Controls Framework v2. *Cloud Security Alliance*. URL: https://cloudsecurityalliance.org/artifacts/csa-iot-security-controls-framework-v2 (date of access: 8.11.2025).

67. Understanding IoT Security: Threats, Standards & Best Practices. URL: https://sternumiot.com/iot-blog/understanding-iot-security-challenges-standards-and-best-practices/ (date of access: 8.11.2025).

68. ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection – Information security management systems – Requirements. *ISO.* URL: https://www.iso.org/standard/27001 (date of access: 4.11.2025).

69. ISO/IEC 27701:2019 Security techniques – Extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management – Requirements and guidelines. *ISO.* URL: https://www.iso.org/standard/71670.html (date of access: 3.11.2025).

70. The Bot-IoT Dataset. Intelligent Security Group, UNSW Canberra, Australia. URL: https://research.unsw.edu.au/projects/bot-iot-dataset (date of access: 12.11.2025).

71. Alosaimi S., Almutairi S. M. An Intrusion Detection System Using BoT-IoT. *Applied Sciences*, 2023. № *13*(9): 5427. DOI: 10.3390/app13095427.

72. Регулювання питань безпеки інтернету речей в фрейворках Європейського Союзу, Великобританії s США / Басистий В. та ін. *Measuring and computing devices in technological processes.* 2025. № 4. (прийнято до видання).

73. Басистий В.А., Чешун О.В., Чешун В.М. Комлекс моніторингу і аналізу мережевого трафіку IOT на одноплатних мікрокомп’ютерах. *Тези доповідей XXVII Всеукраїнської науково-практичної конференції «Могилянські читання – 2024: досвід та тенденції розвитку суспільства в Україні: глобальний, національний та регіональний аспекти», Технічні науки.* 2024. С.103-108.

74. Басистий В.А., Чешун О.В., Чешун В.М. Застосування одноплатних мікрокомп'ютерів для підвищення стійкості інтернету речей до DDoS атак. *Військова освіта і наука: сьогодення та майбутнє : зб. тез доповідей XX Міжнародної науково-практичної конференції.* Київ : Військовий інститут Київського національного університету імені Тараса Шевченка, 2024. С.36.

75. Басистий В.А., Чешун В.М., Чешун О.В. Мережева інфраструктура інформаційної безпеки IOT на одноплатних мікрокомп’ютерах. *Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп’ютерних наук АПКН-2024»*. Хмельницький. 2024. С.35-39.

76. База даних системи моніторингу і аналізу мережевого трафіку інтернету речей. Басистий В.А., Чешун О.В., Чешун В.М. *Сучасні інформаційні системи та*

технології: матеріали VII Всеукр. наук.- практ. інтернет-конф. за тематикою «Сучасні комп'ютерні системи та мережі в управлінні» (29 листопада 2024 р., м. Херсон, м. Хмельницький). / за ред. А. А. Григорової. Херсон: Книжкове видавництво ФОП Вишемирський В. С., 2024. С.243-245.

77. Басистий В.А., Чешун О.В., Чешун В.М. Організація комутаційних підключень мікрокомп'ютерів в комлексі моніторингу і аналізу мережевого трафіку IoT. *Тези XV Міжнародної науково-технічної конференції «Інформаційно-комп'ютерні технології», м. Житомир, 28-29 березня 2025 р.* Житомир: Житомирська політехніка, 2025. С.147-148.

78. Мережева інфраструктура інформаційної безпеки IOT на одноплатних мікрокомп'ютерах. Басистий В.А. та ін. *Збірник наукових праць за матеріалами XVII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2025»*. Хмельницький. 2025. С.23-27.

79. Basystyi V. A., Cheshun D. V., Cheshun V. M. Multi-agent organization of IoT network traffic monitoring system. *Сучасні комп'ютерні системи та технології: матеріали VIII Всеукр. наук.-практ. інтернет-конф. студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні» (24 листопада 2025 р., м. Херсон, м. Хмельницький)* / за ред. А. А. Григорової. Херсон: Книжкове видавництво ФОП Вишемирський В. С., 2025. С. 199-201.

ДОДАТОК А

Копії наукових публікацій

УДК 004.056

БАСИСТИЙ ВІТАЛІЙ
Хмельницький національний університет
https://orcid.org/0009-0009-1978-614X
e-mail: basistavitalij@gmail.com
СТЕЦЮК МИКОЛА
Хмельницький національний університет
https://orcid.org/0000-0003-3875-0416
e-mail: mykola.stetsiuk@khmnu.edu.ua
ЧЕШУН ВІКТОР
Хмельницький національний університет
https://orcid.org/0000-0002-3935-2068
e-mail: cheshunvn@khmnu.edu.ua
ЧЕШУН ДМИТРО
Хмельницький фаховий економіко-технологічний коледж УЕП
https://orcid.org/0009-0007-9937-9450
e-mail: dmitry_95@ukr.net

РЕГУЛЮВАННЯ ПИТАНЬ БЕЗПЕКИ ІНТЕРНЕТУ РЕЧЕЙ В ФРЕЙВОРКАХ ЄВРОПЕЙСЬКОГО СОЮЗУ, ВЕЛИКОБРИТАНІЇ І США

У сучасних умовах стрімкого розвитку цифрових технологій та глобального поширення Інтернету речей (IoT) питання безпеки таких систем набуває ключового значення для захисту критичної інформаційної інфраструктури, конфіденційності користувачів і безперебійності бізнес-процесів. У цій статті проведено комплексний огляд міжнародних стандартів, рекомендацій і нормативних актів, спрямованих на підвищення рівня кіберзахисту IoT-пристроїв та середовищ, зокрема проаналізовано положення NISTIR 8259, IoT Cybersecurity Improvement Act of 2020, ETSI EN 303 645, UK Code of Practice, CSA IoT Security Controls Framework, а також рамку відповідності IoT Security Foundation (IoTSF). Особливу увагу приділено аналізу призначення, структури та основних вимог до безпеки, які охоплюють ідентифікацію пристроїв, захист конфіденційності, управління оновленнями, контроль доступу, моніторинг подій, реагування на інциденти та зниження ризиків, пов'язаних із вразливостями в IoT-інфраструктурі.

Ключові слова: захист інформації, інтернет речей, нормативно-правове регулювання.

BASYSTYI VITALII
STETSIUK MYKOLA
CHESHUN VIKTOR
Khmelnytsky National University
CHESHUN DMYTRO
Khmelnytskyi Vocational Economic and Technological College of the UEE

REGULATION OF INTERNET OF THINGS SECURITY ISSUES IN THE FRAMEWORKS OF THE EUROPEAN UNION, THE UNITED KINGDOM, AND THE USA

In the context of rapid technological progress and the global expansion of the Internet of Things (IoT), ensuring the cybersecurity of IoT systems has become a crucial challenge for protecting critical infrastructure, safeguarding user privacy, and maintaining operational continuity. This article provides a comprehensive analysis of international standards, frameworks, and legislative acts aimed at enhancing IoT security. Specifically, it examines the core provisions, structures, and objectives of documents such as NISTIR 8259, the IoT Cybersecurity Improvement Act of 2020, ETSI EN 303 645, the UK's Secure by Design Code of Practice, the CSA IoT Security Controls Framework, and the IoT Security Foundation Compliance Framework. The article highlights the key requirements for IoT cybersecurity, including device identification, privacy protection, secure update management, access control, event monitoring, incident response, and the mitigation of vulnerabilities within diverse IoT environments. The analysis emphasizes the importance of aligning technical security measures with enterprise risk management. In addition, the article discusses practical tools and techniques relevant to modern IoT defense strategies: unified asset discovery tools that support real-time detection of managed and unmanaged devices, intrusion detection systems (IDS) adapted for industrial and embedded IoT contexts, and the role of virtual patching as a mitigation technique for legacy or unpatchable devices using network-level controls.

The findings conclude that a holistic combination of regulatory compliance, technical innovation, and risk-based

governance is essential for building a resilient IoT security architecture. At the same time, the article outlines persistent challenges such as standard fragmentation, varying policy maturity across jurisdictions, and the limited capabilities of low-resource IoT devices.

Keywords: information protection, Internet of Things, regulatory framework.

Вступ

У сучасну епоху стрімкої цифровізації Інтернет речей (IoT) перетворився з перспективної концепції на один із ключових технологічних чинників розвитку промисловості, міської інфраструктури та побутового середовища. IoT-технології, що об'єднують фізичні об'єкти в єдину мережу через сенсори, мережеві протоколи та аналітичні платформи, створюють принципово нові можливості для збору, обробки й використання даних у реальному часі. Швидке зростання кількості підключених пристроїв, розширення можливостей обробки даних у реальному часі та інтеграція IoT з технологіями штучного інтелекту створюють передумови для глибоких трансформацій у багатьох галузях економіки й суспільного життя.

В економічних реаліях підприємства все частіше впроваджують IoT для оптимізації витрат, покращення контролю якості продукції, віддаленого моніторингу стану обладнання та прогнозного технічного обслуговування [1]. Це дозволяє зменшувати простої, запобігати аваріям і будувати більш гнучкі бізнес-моделі, що відповідають динаміці ринку. IoT виступає ключовим елементом розвитку індустрії 4.0, де виробничі процеси інтегруються з інформаційними технологіями для створення автоматизованих та адаптивних систем [2]. Не менш важливим напрямом залишається впровадження IoT у побутовій сфері. Смарт-пристрої для будинків, такі як інтелектуальні термостати, освітлення, охоронні системи чи побутова техніка, підвищують комфорт і енергоефективність житла. У містах IoT-технології застосовуються для організації інтелектуального управління транспортом, енергомережами, екологічним моніторингом і безпекою, що стає дедалі актуальнішим у контексті сталого розвитку [3]. Значну роль IoT відіграє у сфері охорони здоров'я, де дистанційний моніторинг пацієнтів, носимі медичні пристрої та автоматизовані системи збору даних допомагають оперативно реагувати на зміни стану здоров'я, що особливо важливо в умовах глобальних викликів, таких як пандемії чи демографічне старіння населення [4,5].

Водночас із зростанням значення IoT зростає й кількість викликів. Зокрема, це складнощі управління мережею великої кількості пристроїв, проблеми кібербезпеки та конфіденційності даних. Чисельність підключених пристроїв IoT постійно зростає і прогнозується, що найближчими роками їх будуть десятки мільярдів [6]. Ці пристрої генерують величезні обсяги даних. Саме тому актуальність IoT тісно пов'язана з розвитком нових стандартів безпеки і нормативних підходів, які повинні забезпечувати захищене та відповідальне використання цієї технології.

Аналіз досліджень та публікацій

Для підвищення безпеки та надійності IoT застосовуються різноманітні рішення.

Виявляти та контролювати керовані та некеровані пристрої системи IoT в режимі реального часу допомагають інструменти уніфікованого виявлення активів [7]. Їхня основна функція полягає у тому, щоб автоматично ідентифікувати всі підключені пристрої в мережі, незалежно від типу, виробника чи операційної системи, та надати IT- і безпековим командам повну та актуальну інформацію про стан цих активів у режимі реального часу. Такі інструменти зазвичай поєднують кілька технологій виявлення – активне сканування, пасивний моніторинг мережевого трафіку, аналіз протоколів та інтеграцію з іншими системами управління активами. Це дозволяє створювати детальні профілі пристроїв: визначати їхній тип, модель, версію прошивки, відкриті порти, підключені сервіси та рівень ризику тощо.

Системи виявлення вторгнень (Intrusion Detection Systems – IDS) для IoT-середовищ забезпечують моніторинг, адаптований до промислових систем [7,8]. Такі IDS орієнтовані на специфіку промислових протоколів (Modbus, DNP3, OPC UA, BACnet та інші протоколи управління промисловим обладнанням і автоматизованими системами керування ICS/SCADA). На відміну від традиційних IT IDS, які зазвичай фокусуються на загальних IP-протоколах і мережевих аномаліях, IoT-IDS враховують особливості форматів команд, дозволених операцій і типових сценаріїв взаємодії пристроїв у виробничих мережах. Ці IDS мають підтримувати пасивний моніторинг, щоб уникнути порушення технологічних процесів. У промислових середовищах дуже важливо не створювати додаткового навантаження або затримок у роботі критичного обладнання. Тому більшість промислових IDS використовують дзеркальне копіювання трафіку (port mirroring) або TAP-пристрої для непомітного спостереження за мережею.

Віртуальне патчування у сфері IoT – ще один із практичних методів захисту середовищ, де велика кількість пристроїв часто не підтримує регулярних оновлень програмного забезпечення або взагалі не має технічної можливості для їх впровадження. Це характерна проблема для промислового IoT, медичних приладів, розумних побутових систем та вбудованих сенсорів, де будь-яке втручання в роботу пристрою може порушити його функції або суперечити вимогам сертифікації [7]. Рішення для віддаленого управління та моніторингу (Remote Management and Monitoring – RMM) полегшують управління патчами та моніторинг пристроїв [5].

Платформи безпеки IoT надають комплексний захист для пристроїв та систем IoT, а шлюзи безпеки IoT забезпечують безпечне підключення та обробку даних [9]. Апаратні модулі безпеки (Hardware Security Modules – HSM) забезпечують надійну криптографічну обробку та захист ключів [9]. Технологія блокчейн пропонує децентралізований та захищений від підробок спосіб управління даними IoT та підвищення безпеки [10]. Багатооператорські eSIM підвищують надійність та гнучкість підключення пристроїв IoT, дозволяючи автоматичне перемикання між операторами [11]. Штучний інтелект та машинне навчання використовуються для виявлення шаблонів атак та захисту пристроїв IoT. Безпечні елементи та довірені середовища виконання (Trusted Execution Environments – TEEs) забезпечують захищені області на процесорах для виконання чутливих операцій [9].

Поряд із великим різноманіттям технологічних рішень для забезпечення безпеки IoT характерною рисою сучасного етапу їх розвитку є відсутність єдиних підходів в їх реалізації. Для вирішення існуючих проблем та недоліків необхідні комплексні рішення, які враховують аспекти безпеки, конфіденційності та операційної ефективності систем IoT і базуються на міжнародних нормативно-регулюючих документах.

Формулювання цілей статті

Метою роботи є: аналіз міжнародних і національних фреймворків, що безпосередньо або опосередковано регулюють питання безпеки пристроїв, систем і технологій Інтернету речей.

Виклад основного матеріалу

Хоча безпека IoT є однією із найактуальніших задач сьогодення, рішенню цієї проблеми на рівні законів і стандартів увагу почали придідяти не так давно. На сьогодні існує сукупність фреймворків та нормативних документів, які допомагають забезпечити безпеку пристроїв та систем IoT.

Однією з перших спроб у світі стимулювати ринок масового IoT до самоорганізації у сфері кібербезпеки і сформулювати на рівні національного уряду загальнодержавні мінімальні вимоги до безпеки споживчих IoT-пристроїв в формі практичних рекомендацій для виробників, імпортерів, розробників та дистриб'юторів стали Рекомендації щодо безпеки споживчих IoT-пристроїв (Code of Practice for Consumer IoT Security) [12] – політичний документ, опублікований Міністерством цифрових технологій, культури, медіа та спорту Великої Британії (UK DCMS) у 2018 році.

У початковій редакції Code of Practice визначено 13 принципів безпеки, що покривають повний життєвий цикл IoT-пристрою від розробки і виробництва до утилізації, серед яких:

– паролі мають бути унікальними для кожного примірника пристроїв IoT або змінюватися під час першого налаштування;

– виробник має впровадити політику управління уразливостями і призначити контактну точку для повідомлення про знайдені вразливості;

– пристрої повинні мати зрозумілі механізми оновлення прошивки і програмного забезпечення (ПЗ) протягом погодженого строку підтримки, а виробник має чітко інформувати користувача про те, як довго пристрій буде підтримуватися оновленнями безпеки;

– пристрої мають обробляти особисті дані відповідно до принципів GDPR (мінімізувати обсяг зібраних даних, обґрунтовувати мету збирання, захищати дані шифруванням);

– функціонал і доступ мають ґрунтуватися на принципі мінімальних привілеїв, забезпечуючи мінімальні права доступу;

– реалізація захисту цілісності через впровадження механізмів виявлення несанкціонованих змін конфігурації, контролю цілісності ПЗ та прошивки;

– дані, що передаються між пристроями, мають бути захищені сучасними протоколами шифрування (захист каналу зв'язку);

– пристрої повинні мати функціонал для безпечного стирання (знеособлення) персональних даних при перепродажі чи утилізації;

– рекомендується забезпечити механізми журналювання безпечної діяльності пристрою, щоб вчасно виявляти інциденти;

– пристрої мають бути стійкими до збоїв, зокрема, здатними відновлюватися після аварійного оновлення;

– виробники мають чітко інформувати користувачів про налаштування безпеки, політики конфіденційності та терміни підтримки.

Завдяки цьому документу Велика Британія стала одним із лідерів формування практики «безпечного дизайну» для IoT у Європі та вплинула на глобальні рекомендації.

Базисом для розвитку системи стандартизації питань безпеки IoT в США став федеральний закон США «IoT Cybersecurity Improvement Act of 2020» [13], ухвалений Конгресом і підписаний у грудні 2020 року. Він став першим системним нормативним документом на федеральному рівні, що безпосередньо регулює вимоги до кібербезпеки IoT

у державному секторі. Його головна мета – підвищити безпеку федеральних інформаційних систем за рахунок встановлення мінімальних стандартів кіберзахисту для IoT-пристроїв, які закуповуються та експлуатуються урядовими структурами. Цей закон покликаний усунути ризики, пов'язані з тим, що IoT-пристрої (від окремих датчиків та камер і до медичних пристроїв та розумних будівельних систем тощо) часто не мають належних механізмів автентифікації, оновлення або контролю доступу, що робить їх уразливими до атак. Закон зобов'язує Національний інститут стандартів і технологій США (NIST) розробляти та публікувати рекомендації й технічні стандарти для безпечного проєктування, ідентифікації, конфігурації, оновлення й управління IoT-пристроями. Ці керівництва мають ґрунтуватися на передових практиках кібербезпеки. Федеральні агентства зобов'язані закуповувати лише ті IoT-пристрої та послуги, які відповідають стандартам і рекомендаціям NIST. Це означає, що будь-який постачальник, який хоче співпрацювати з урядом США, має гарантувати, що його пристрої відповідають базовим вимогам щодо конфіденційності, цілісності, доступності та можливості управління уразливостями.

На вимогу розглянутого закону Національним інститутом стандартів і технологій США розроблено серію спеціальних публікацій SP 800-213 [14], які надають керівництво для федеральних установ щодо використання пристроїв IoT у їхніх системах. Зокрема, SP 800-213 визначає вимоги до кібербезпеки пристроїв IoT та пояснює їх роль у федеральних системах, а також розглядає ризики, які вони можуть становити. Серія SP 800-213 пропонує комплексний підхід до управління ризиками, пов'язаними з використанням пристроїв IoT, враховуючи їхню роль у федеральних системах та їхні специфічні загрози. Документи серії розглядають пристрої IoT як невід'ємну частину федеральних систем та підкреслюють їх унікальні ризики для кібербезпеки, які необхідно враховувати.

Фреймворк кібербезпеки IoT Device Cybersecurity Guidance for the Federal Government [15], опублікований NIST у рамках серії SP 800-213 та суміжних документів, є одним із ключових нормативних орієнтирів для забезпечення безпеки IoT у державному секторі США. Його поява зумовлена швидким зростанням використання IoT-пристроїв у федеральних установах, що створює суттєві ризики для національної кібербезпеки через специфічні вразливості цих технологій. Головна мета керівництва – надати федеральним органам виконавчої влади чіткі рекомендації щодо того, як безпечно впроваджувати, експлуатувати й управляти IoT-пристроями протягом усього їхнього життєвого циклу. Фреймворк містить рекомендації щодо розгляду безпеки системи IoT з точки зору окремих пристроїв. Це дозволяє визначити вимоги до кібербезпеки пристроїв – можливості та дії, яких організація очікує від пристрою IoT, його виробника та третіх сторін. Документ допомагає замовникам і керівникам IT-проєктів враховувати ризики кібербезпеки вже на етапі закупівель і визначає вимоги до виробників і постачальників IoT-рішень, які мають відповідати мінімальним критеріям безпеки.

Для реалізації комплексного підходу у забезпеченні безпеки IoT Національним інститутом стандартів і технологій США також розроблено серію публікацій NISTIR 8259 [16], яка спрямована на формування уніфікованого підходу до забезпечення кібербезпеки IoT-пристроїв у контексті їхнього життєвого циклу: проєктування, виробництво, впровадження, експлуатація та виведення з експлуатації. Головна мета NISTIR 8259 – допомогти виробникам розуміти мінімальні обов'язкові характеристики безпеки IoT-пристроїв, які очікують замовники у федеральному секторі, і забезпечити прозорість у спілкуванні між постачальниками і користувачами.

Серія складається з трьох ключових документів:

– NISTIR 8259: Foundational Cybersecurity Activities for IoT Device Manufacturers [17] – основоположний документ, який описує базові заходи з кібербезпеки, що мають бути виконані виробниками IoT-пристроїв;

– NISTIR 8259A: IoT Device Cybersecurity Capability Core Baseline [18] – містить перелік мінімальних технічних можливостей безпеки, які мають бути вбудовані у будь-який IoT-пристрій;

– NISTIR 8259B (Draft): IoT Non-Technical Supporting Capability Core Baseline [19] – описує нефункціональні вимоги (політики, документи, інформаційна підтримка та супровід), що мають забезпечувати безпечну експлуатацію IoT.

Таким чином, NISTIR 8259 задає загальну рамку, NISTIR 8259A деталізує технічний мінімум, а NISTIR 8259B фокусується на нефункціональних та організаційних аспектах експлуатації IoT.

Рамковий документ CSA IoT Security Controls Framework [20] створено Cloud Security Alliance (CSA) для організацій, які проєктують, розробляють або впроваджують IoT-системи. Фреймворк є актуальним для корпоративних систем IoT, що включають різні типи підключених пристроїв, хмарних сервісів та мережевих технологій. Його головна мета – надати структуровану систему заходів контролю безпеки, що дозволяє виробникам, операторам і інтеграторам IoT зменшувати ризики вразливостей та забезпечувати безпечне функціонування IoT-рішень протягом життєвого циклу.

CSA структурувала фреймворк за принципом розподілу контролів безпеки на ключові домени відповідно до компонентів типової IoT-архітектури:

– пристрої (Device/Thing Security);
– мережевий рівень (Network Security);
– шлюзи (Gateway Security);

– IoT-платформи / хмара (Cloud/Platform Security);
– додатки (Application Security);
– операційні процеси та управління (Operations and Lifecycle Security).

Ключові принципи та групи контролів, на яких зосереджується CSA IoT Security Controls Framework:

– ідентифікація та автентифікація: усі IoT-пристрої мають бути однозначно ідентифіковані, має бути впроваджена багатофакторна автентифікація для пристроїв і користувачів у всіх можливих випадках, уникнення «загальних» облікових даних за замовчуванням;

– управління конфігурацією та цілісністю: захист конфігурацій від несанкціонованої зміни, механізми перевірки цілісності ПЗ/прошивки, контроль і моніторинг змін конфігурації;

– захищені оновлення: можливість безпечного оновлення прошивки та ПЗ, підпис цифрових оновлень для перевірки автентичності, політика своєчасного усунення відомих уразливостей;

– захист даних: шифрування даних «у спокої» та «у русі», управління ключами шифрування, мінімізація збирання персональних даних;

– безпека комунікацій: захищені протоколи (TLS, VPN), сегментація мережевих зон для IoT, засоби запобігання атакам «людина посередині»;

– моніторинг та реагування: постійний моніторинг подій безпеки, інтеграція з SIEM/IDS/IPS, журналювання активності пристроїв;

– управління доступом: принцип мінімальних привілеїв, рольовий або атрибутивний контроль доступу (RBAC/ABAC), регулярна ревізія прав доступу;

– безпечний життєвий цикл: урахування безпеки ще на стадії проєктування (security by design), політика безпечної утилізації або перепродажу пристроїв, управління інцидентами й оновлення планів реагування.

Цей фреймворк не є обов'язковим стандартом або законом, а радше практичним методичним посібником, що допомагає врахувати різні аспекти безпеки IoT від пристроїв до хмарних платформ і мережевої інфраструктури. CSA Framework часто використовується як відправна точка для аудиту IoT-середовищ, розробки політик безпеки та планування відповідності вимогам регуляторів. Він особливо актуальний для організацій, які поєднують IoT із хмарними сервісами, промисловими мережами або великими екосистемами IoT.

Рамка відповідності (Compliance Framework) Фонду безпеки IoT (IoT Security Foundation – IoTSF) [21] розроблена як комплексна система відповідності, яка цілісно вирішує питання безпеки IoT. Фреймворк є практичним, добровільним методичним документом, який допомагає виробникам, постачальникам і операторам IoT-систем розробляти, впроваджувати та перевіряти безпечні IoT-продукти і сервіси. Головна мета Рамки відповідності – уніфікувати мінімальні вимоги до безпеки IoT, забезпечити їх перевірку й стимулювати ринок виробників і постачальників гарантувати споживачам і бізнесу належний рівень кіберстійкості.

Рамка відповідності IoTSF організована в чотири основні розділи, які визначають конкретні міркування безпеки, які повинні бути адресовані для забезпечення надійності систем IoT:

– розділ «Процес» фокусується на загальному управлінні та процесах, які повинні бути введені для ефективного управління безпекою IoT;

– розділ «Програмне забезпечення» охоплює конкретні міркування програмного забезпечення, які мають вирішальне значення для підтримки безпеки та цілісності систем IoT;

– розділ «Фізичний» охоплює фізичні аспекти пристроїв IoT, включаючи аспекти апаратного забезпечення;

– розділ «Зв'язок» окреслює міркування безпеки для зв'язку між пристроями та системами IoT.

Рамка відповідності IoTSF – це гнучкий інструмент для організацій, які прагнуть реалізувати принципи «безпечного IoT за замовчуванням» на практиці. Вона виступає містком між високорівневими політичними вимогами, технічними стандартами і реальними практиками розробки IoT-продуктів.

Стандарт кібербезпеки CC для споживчих пристроїв IoT ETSI EN 303 645 [22] охоплює широкий спектр споживчих пристроїв IoT, включаючи підключені дитячі іграшки, пристрої безпеки, розумні камери, телевізори, колонки, системи домашньої автоматизації та побутову техніку. ETSI EN 303 645 передбачає відсутність паролів за замовчуванням, впровадження політики розкриття уразливостей, забезпечення актуальності програмного забезпечення, захист даних споживачів. Стандарт спрямований на запобігання великомасштабним атакам на підключені споживчі пристрої, встановлюючи базовий рівень безпеки, і допомагає виробникам IoT-пристроїв вбудувати безпеку в продукти на етапі проєктування, що підвищує довіру споживачів та знижує ризики, пов'язані з вразливостями. Стандарт ETSI EN 303 645 не є обов'язковим згідно із законом, але його дотримання вважається важливим для виробників, що працюють на європейському ринку. Потенційно, ETSI EN 303 645 може бути основою для майбутніх схем сертифікації IoT, що підтверджують відповідність пристроїв вимогам безпеки.

Комплексну методологічну основу для побудови надійної системи управління інформаційною безпекою та конфіденційністю в IoT-середовищах забезпечують міжнародні стандарти ISO/IEC 27001 і ISO/IEC 27701.

Стандарт ISO/IEC 27001 [23] визначає вимоги до побудови, впровадження, підтримки та постійного

вдосконалення системи управління інформаційною безпекою. Він встановлює рамкові умови для управління ризиками безпеки інформації незалежно від сфери застосуванняі може бути адаптований до IoT-середовищ. У цьому контексті ключову роль відіграє проведення комплексної оцінки ризиків, ідентифікація активів IoT як об'єктів захисту та впровадження відповідних політик контролю доступу, шифрування, а також моніторингу та реагування на інциденти. Особливої уваги набувають питання захисту мережевої інфраструктури, автентифікації пристроїв і забезпечення цілісності даних, що передаються в IoT-середовищі.

Доповненням до ISO/IEC 27001 є стандарт ISO/IEC 27701 [24], який фокусується на управлінні конфіденційністю інформації та персональних даних. Цей стандарт розширює вимоги системи управління інформаційною безпекою до системи управління конфіденційністю інформації та пропонує конкретні механізми для дотримання законодавства у сфері захисту даних, таких як Загальний регламент ЄС про захист даних (GDPR). У випадку IoT це має критичне значення, оскільки такі системи часто обробляють великі обсяги даних про користувачів, їхню поведінку та фізичне середовище. Стандарт ISO/IEC 27701 передбачає запровадження процедур інформованої згоди, обмеження цілей обробки даних, мінімізації обсягів збору інформації та управління доступом до персональних даних.

Застосування цих стандартів у сфері IoT сприяє формуванню структурованих та прозорих процесів управління безпекою й конфіденційністю. Це особливо актуально для критичних галузей, таких як охорона здоров'я, транспорт, енергетика, де компрометація IoT-пристроїв може мати серйозні наслідки не лише для бізнесу, а й для безпеки людей. Імплементація ISO/IEC 27001 та ISO/IEC 27701 дозволяє організаціям формалізувати підходи до оцінки ризиків, впроваджувати технічні та організаційні заходи захисту, а також документувати відповідність нормативним вимогам.

Висновки

Швидке поширення Інтернету речей відкриває значні можливості, але також створює серйозні проблеми, особливо у сферах безпеки, конфіденційності та операційної надійності. Існує широкий спектр проблем, починаючи від економічних бар'єрів та технічних складнощів, закінчуючи соціальними та регуляторними викликами. Для ефективного вирішення цих проблем необхідний комплексний підхід, що включає впровадження передових технологічних рішень, дотримання найкращих практик безпеки, використання існуючих фреймворків та стандартів, а також постійне навчання користувачів. Надійні механізми автентифікації, наскрізне шифрування даних, безпечні оновлення програмного забезпечення та сегментація мережі є критично важливими для забезпечення безпеки екосистеми IoT. Захист конфіденційності користувачів вимагає мінімізації даних, анонімізації та прозорого управління згодою відповідно до чинних правил. Забезпечення операційної ефективності та надійності передбачає стандартизацію, масштабованість, ефективне управління даними та надійне підключення.

Майбутнє безпеки та надійності IoT залежить від безперервних інновацій, тісної співпраці між галузевими гравцями та розробки глобальних стандартів. Лише спільними зусиллями в правовому полі можливо створити безпечну та надійну екосистему IoT, яка повністю реалізує свій трансформаційний потенціал.

Література

1. Франів І.А. Переваги впровадження IoT для автоматизації процесів у продуктовому ритейлі / І.А. Франів, П.П. Єременко // Вісник ЛТЕУ. Економічні науки. – 2024. – № 80. – С. 49–55.

2. Назаренко Н. Співпраця індустрії 4.0 та інтернету речей IoT / Н. Назаренко, С. Заєць, Ю. Киричук // Вісник Хмельницького національного університету. Технічні науки. – 2024. – № 5(*341*). – С. 74–79.

3. Шпак О. Розумні міста та Інтернет речей: вплив розробок у сфері IT на розвиток міст і покращення якості життя / О. Шпак, П. Федорка, М. Пригара // Сучасний стан наукових досліджень та технологій в промисловості. – 2023. – №3 (25). – С. 114–128.

4. Макаренко М.В. Особливості впровадження технологій інтернету речей у сфері охорони здоров'я / М.В. Макаренко // Вчені записки ТНУ імені В.І. Вернадського. Серія: Державне управління. – 2021. – № 2. – С. 64-68.

5. IoT в охороні здоров'я: Застосування, переваги та виклики у 2023 році [Електронний ресурс] // Stfalcon.com. – Режим доступу: https://stfalcon.com/uk/blog/post/iot-in-healthcare-benefits-challenges (Дата звернення: 13.06.2025). – Назва з екрана.

6. What are the emerging threats and challenges in securing Internet of Things (IoT) devices and networks? [Electronic resource] // ResearchGate. – Access mode: https://surl.li/rzzluj (Accessed: 13.06.2025). – Screen Title.

7. Top IoT security issues and solutions for low-power devices [Electronic resource] // Onomondo. – Access mode: https://onomondo.com/blog/iot-security-issues-and-solutions-low-power-devices/ (Accessed: 13.06.2025). – Screen Title.

8. Очеретний С.О. Системи виявлення та запобігання вторгнень, найбільш успішні практики / С.О. Очеретний, В.Г. Крижановський // Прикладні аспекти сучасних міждисциплінарних досліджень. – 2024. – С. 236-238.

9. Журило О. Огляд рішень з апаратної безпеки кінцевих пристроїв туманних обчислень у інтернеті речей

/ О. Журило, О. Ляшенко, К. Аветісова // Сучасний стан наукових досліджень та технологій в промисловості. – 2023. – № 1 (23). – С. 57-81.

10. Serebriakov R. Integration of blockchain technology into the Internet of Things (overview) / R. Serebriakov, V. Tkachenko, I. Klymenko // Information, Computing and Intelligent systems. – 2024. – № 4. – P. 99-113.

11. Делембовський М.М. Аналіз сучасних наукових публікацій за напрямком тематики кібербезпеки IoT технологій / М.М. Делембовський, Б.В. Корнійчук // Грааль науки. – 2023. – № 25. – С. 203-206.

12. Code of Practice for Consumer IoT Security. – Government of the United Kingdom: Department for Digital, Culture, Media & Sport, 2018. – 20 p.

13. IoT Cybersecurity Improvement Act of 2020 [Electronic resource] // CONGRESS.GOV. – Access mode: https://www.congress.gov/bill/116th-congress/house-bill/1668/text (Accessed: 13.06.2025). – Screen Title.

14. SP 800-213 Series [Electronic resource] // NIST. – Access mode: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/sp-800-213-series (Accessed: 13.06.2025). – Screen Title.

15. NIST SP 800-213: IoT Device Cybersecurity Guidance for the Federal Government: Establishing IoT Device Cybersecurity Requirements [Electronic resource] // NIST. – Access mode: https://csrc.nist.rip/Pubs/sp/800/213/Final (Accessed: 13.06.2025). – Screen Title.

16. NISTIR 8259 Series [Electronic resource] // NIST. – Access mode: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/nistir-8259-series (Accessed: 13.06.2025). – Screen Title.

17. NISTIR 8259: Foundational Cybersecurity Activities for IoT Device Manufacturers. National Institute of Standards and Technology Interagency or Internal Report 8259. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 36 p.

18. NISTIR 8259A: IoT Device Cybersecurity Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259A. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 23 p.

19. Draft NISTIR 8259B: IoT Non-Technical Supporting Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259B. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 20 p.

20. CSA IoT Security Controls Framework v2 [Electronic resource] // Cloud Security Alliance. – Access mode: https://cloudsecurityalliance.org/artifacts/csa-iot-security-controls-framework-v2 (Accessed: 13.06.2025). – Screen Title.

21. Understanding IoT Security: Threats, Standards & Best Practices [Electronic resource] – Access mode: https://sternumiot.com/iot-blog/understanding-iot-security-challenges-standards-and-best-practices/ (Accessed: 13.06.2025). – Screen Title.

22. ETSI EN 303 645. Cyber Security for Consumer Internet of Things: Baseline Requirements. – ETSI. – 2024. – 41 p.

23. ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection – Information security management systems – Requirements [Electronic resource] // ISO. – Access mode: https://www.iso.org/standard/27001 (Accessed: 13.06.2025). – Screen Title.

24. ISO/IEC 27701:2019 Security techniques – Extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management – Requirements and guidelines [Electronic resource] // ISO. – Access mode: https://www.iso.org/standard/71670.html (Accessed: 13.06.2025). – Screen Title.

References

1. Franiv I.A. Perevahy vprovadzhennia IoT dlia avtomatyzatsii protsesiv u produktovomu ryteili / I.A. Franiv, P.P. Yeremenko // Visnyk LTEU. Ekonomichni nauky. – 2024. – № 80. – S. 49–55.

2. Nazarenko N. Spivpratsia industrii 4.0 ta internetu rechei IoT / N. Nazarenko, S. Zaiets, Yu. Kyrychuk // Herald of Khmelnytskyi National University. Technical Sciences. – 2024. – № 5(341). – S. 74–79.

3. Shpak O. Rozumni mista ta Internet rechei: vplyv rozrobok u sferi IT na rozvytok mist i pokrashchennia yakosti zhyttia / O. Shpak, P. Fedorka, M. Pryhara // Suchasnyi stan naukovykh doslidzhen ta tekhnolohii v promyslovosti. – 2023. – №3 (25). – S. 114–128.

4. Makarenko M.V. Osoblyvosti vprovadzhennia tekhnolohii internetu rechei u sferi okhorony zdorovia / M.V. Makarenko // Vcheni zapysky TNU imeni V.I. Vernadskoho. Seriia: Derzhavne upravlinnia. – 2021. – № 2. – S. 64-68.

5. IoT v okhoroni zdorovia: Zastosuvannia, perevahy ta vyklyky u 2023 rotsi [Elektronnyi resurs] // Stfalcon.com. – Rezhym dostupu: https://stfalcon.com/uk/blog/post/iot-in-healthcare-benefits-challenges (Data zvernennia: 13.06.2025). – Nazva z ekrana.

6. What are the emerging threats and challenges in securing Internet of Things (IoT) devices and networks? [Electronic resource] // ResearchGate. – Access mode: https://surl.li/rzzluj (Accessed: 13.06.2025). – Screen Title.

7. Top IoT security issues and solutions for low-power devices [Electronic resource] // Onomondo. – Access mode: https://onomondo.com/blog/iot-security-issues-and-solutions-low-power-devices/ (Accessed: 13.06.2025). – Screen Title.

8. Ocheretnyi S.O. Systemy vyiavlennia ta zapobihannia vtorhnen, naibilsh uspishni praktyky / S.O. Ocheretnyi, V.H. Kryzhanovskyi // Prykladni aspekty suchasnykh mizhdystsyplinarnykh doslidzhen. – 2024. – C. 236-238.

9. Zhurylo O. Ohliad rishen z aparatnoi bezpeky kintsevykh prystroiv tumannykh obchyslen u interneti rechei / O. Zhurylo, O. Liashenko, K. Avetisova // Suchasnyi stan naukovykh doslidzhen ta tekhnolohii v promyslovosti. – 2023. – № 1 (23). – S. 57-81.

10. Serebriakov R. Integration of blockchain technology into the Internet of Things (overview) / R. Serebriakov, V. Tkachenko, I. Klymenko // Information, Computing and Intelligent systems. – 2024. – № 4. – P. 99-113.

11. Delembovskyi M.M. Analiz suchasnykh naukovykh publikatsii za napriamkom tematyky kiberbezpeky IoT tekhnolohii / M.M. Delembovskyi, B.V. Korniichuk // Hraal nauky. – 2023. – № 25. – S. 203-206.

12. Code of Practice for Consumer IoT Security. – Government of the United Kingdom: Department for Digital, Culture, Media & Sport, 2018. – 20 p.

13. IoT Cybersecurity Improvement Act of 2020 [Electronic resource] // CONGRESS.GOV. – Access mode: https://www.congress.gov/bill/116th-congress/house-bill/1668/text (Accessed: 13.06.2025). – Screen Title.

14. SP 800-213 Series [Electronic resource] // NIST. – Access mode: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/sp-800-213-series (Accessed: 13.06.2025). – Screen Title.

15. NIST SP 800-213: IoT Device Cybersecurity Guidance for the Federal Government: Establishing IoT Device Cybersecurity Requirements [Electronic resource] // NIST. – Access mode: https://csrc.nist.rip/Pubs/sp/800/213/Final (Accessed: 13.06.2025). – Screen Title.

16. NISTIR 8259 Series [Electronic resource] // NIST. – Access mode: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program/nistir-8259-series (Accessed: 13.06.2025). – Screen Title.

17. NISTIR 8259: Foundational Cybersecurity Activities for IoT Device Manufacturers. National Institute of Standards and Technology Interagency or Internal Report 8259. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 36 p.

18. NISTIR 8259A: IoT Device Cybersecurity Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259A. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 23 p.

19. Draft NISTIR 8259B: IoT Non-Technical Supporting Capability Core Baseline. National Institute of Standards and Technology Interagency or Internal Report 8259B. – U.S. Department of Commerce: National Institute of Standards and Technology (NIST). – 2020. – 20 p.

20. CSA IoT Security Controls Framework v2 [Electronic resource] // Cloud Security Alliance. – Access mode: https://cloudsecurityalliance.org/artifacts/csa-iot-security-controls-framework-v2 (Accessed: 13.06.2025). – Screen Title.

21. Understanding IoT Security: Threats, Standards & Best Practices [Electronic resource] – Access mode: https://sternumiot.com/iot-blog/understanding-iot-security-challenges-standards-and-best-practices/ (Accessed: 13.06.2025). – Screen Title.

22. ETSI EN 303 645. Cyber Security for Consumer Internet of Things: Baseline Requirements. – ETSI. – 2024. – 41 p.

23. ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection – Information security management systems – Requirements [Electronic resource] // ISO. – Access mode: https://www.iso.org/standard/27001 (Accessed: 13.06.2025). – Screen Title.

24. ISO/IEC 27701:2019 Security techniques – Extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management – Requirements and guidelines [Electronic resource] // ISO. – Access mode: https://www.iso.org/standard/71670.html (Accessed: 13.06.2025). – Screen Title.

Підсекція:

- **КОМП'ЮТЕРНА ІНЖЕНЕРІЯ**

Міністерство освіти і науки України
Чорноморський національний університет імені Петра Могили
ДНУ «Інститут модернізації змісту освіти»
Південний науковий центр НАН та МОН
Інститут української археографії та джерелознавства
імені М. С. Грушевського НАН України
Первинна профспілкова організація ЧНУ ім. Петра Могили

«МОГИЛЯНСЬКІ ЧИТАННЯ – 2024: досвід та тенденції розвитку суспільства в Україні: глобальний, національний та регіональний аспекти»

XXVII Всеукраїнська науково-практична конференція

ТЕЗИ ДОПОВІДЕЙ

ТЕХНІЧНІ НАУКИ

Миколаїв, 6–10 листопада 2024 року

Миколаїв – 2024

шети, ноутбуки, стаціонарні і переносні персональні комп'ютери (ПК), камери відеоспостереження, «розумні пристрої» тощо. Кожен такий пристрій в активному режимі постійно передає якусь інформацію для користувачів, щоб забезпечити максимальну зручність, продуктивність і ефективність від їх використання, але при цьому є постійна загроза втрати інформації і не тільки через технічні проблеми, але і через людський фактор.

Щодня різні зловмисні групи здійснюють спроби несанкціонованого доступу до інформаційних систем і ресурсів з метою пошкодження інформації, несанкціонованого отримання особистих даних, шантажу, заспамлювання, припинення роботи серверних станцій та кінцевих пристроїв тощо. Кількість зловмисних дій і атак в кіберпросторі постійно зростає. В статті [1] наводиться статистика DDoS-атак, які в 2023 році зросли на 68% в порівнянні з 2022 роком, а в першому півріччі 2024 р. зафіксоване чергове зростання кількості DDoS-атак на 46% і досягнення потужності атаки 1,7 Тбіт/с [2]. Кіберзлочинці комерційного спрямування можуть мати за мету як нанесення фінансових збитків через заволодіння активами компанії-жертви, так і через примушування компанії витрачати кошти на захист від сторонніх і власних уражених пристроїв, які генерують шкідливий трафік та зменшують ефективність роботи інформаційної системи жертви або призводять до її блокування перевантаженням.

Додатковим ризиком є програмне забезпечення (ПЗ), яке може бути пошкодженим або модифікованим кіберзлочинцями для власних цілей. Прикладом такої небезпеки є атака на телекомунікаційну компанію «Київстар».

З відкритих статистичних даних [3] можна визначити пріоритетні сектори, види і динаміку кібератак (рис. 1–2).

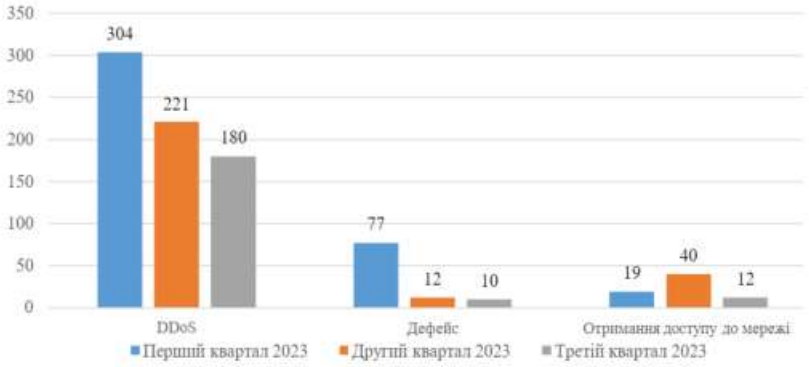


Рис.1. Динаміка активності російських хакерських атак

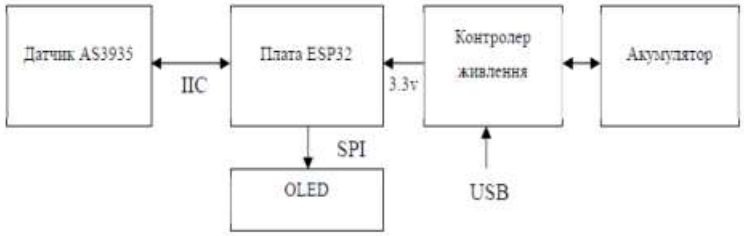


Рис.4. Загальна схема пристрою

Перспективним напрямом розвитку системи є створення мережі зі стаціонарних датчиків, яка дозволить за допомогою тріангуляційних алгоритмів відслідковувати переміщення грозового фронту. Подібні проєкти існують (наприклад, Blitzortung [2]), але в глобальному масштабі та з використанням більш коштовного обладнання.

Список використаних джерел

1. Gravity: Lighting Sensor SKU: SEN0290 - DRFobot. URL: https://wiki.dfrobot.com/Gravity: Lightning Sensor SKU: SEN0290 (Last accessed: 07.10.2024).

2. Lighting & Thunderstorm – World Map. URL: https://www.blitzortung.org (Last accessed: 07.10.2024).

УДК 004.056.5

Басистий В. А.,
студент групи КБЗІм-24-1,
Чешун О. В.,
студент групи ІПЗ-23-1,
Чешун В. М.,
канд. техн. наук, доцент кафедри кібербезпеки,
Хмельницький національний університет, м. Хмельницький, Україна

КОМПЛЕКС МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ ІОТ НА ОДНОПЛАТНИХ МІКРОКОМП'ЮТЕРАХ

В нинішній час різновиди пристроїв сучасної людини, які підключені до всесвітньої мережі інтернет, за кількістю вираховуються десятками. До цих пристроїв відносять смарт-годинники, смартфони, план-

Рис.2. Сектори, які зазнали цілеспрямованих атак у І півріччі 2023 р.

З наведених на рис. 1–2 діаграм видно, що цілеспрямованим атакам піддаються різні сектори економіки та суспільного життя, а найбільш поширеними є атаки на відмову в обслуговуванні (DDoS-атак) через переповнення мережевого трафіку або вичерпування обчислювальної потужності серверів [4].

Одним із найпопулярніших об'єктів DDoS-атак на сьогодні є мережі інтернету речей (англ. *Internet of Thing, IoT*), значна кількість ресурсів яких є менш потужними і захищеними порівняно із серверами та робочими станціями традиційних комп'ютерних мереж, а тому вони є більш вразливими [5]. Щоб зменшити ризик втрати даних IoT є актуальним використання ПЗ і ліцензійного обладнання від компаній, які несуть відповідальність клієнтську інформацію та її цілісність, конфіденційність і доступність. Такими компаніями виступають: CISCO, Microsoft, Mandiant, IBM, Oracle, Amazon Web Services, тощо [6]. Іншим пріоритетним напрямком захисту мереж IoT є виявлення DDoS-атак і протидія їм через моніторинг і аналіз мережевого трафіку [4; 5].

Особливістю систем IoT є непристосованість і недостатність ресурсів у більшості їх пристроїв до завдань моніторингу мережевого трафіку, а покладання відповідних задач на сервери не завжди є ефективним або можливим.

Для підвищення стійкості систем IoT до DDoS та інших мережевих атак розглядається можливість утворення системи захисту із застосуванням одноплатних мікрокомп'ютерів Raspberry Pi або їх аналогів. Ціллю виконуваних робіт постало створення програмно-апаратного комплексу, який проводить розподілений моніторинг всієї мережі IoT, аналізує трафік, повідомляє і зменшує збитки від можливих атак на ресурси IoT.

На рис. 3 представлена базова демонстраційна модель комплексу, зроблена засобами візуального моделювання CISCO Packet Tracer.

Програмні складові комплексу або налаштування розгортаються на декількох пристроях:

– пристрої моніторингу;
– сервер, який збирає дані і надає доступ для їх подальшого аналізу;
– користувацькі персональні комп'ютери або ноутбука;
– комутатор або маршрутизатор безпровідного з'єднання.

Для розробки програмної складової комплексу було обрано мову програмування Python з бібліотеками: Psutil, Scapy, Pandas. Ця комбінація надає можливість отримувати детальну інформацію, активні процеси та підключені пристрої [7].

Програма моніторингу розгортається на мікрокомп'ютері Raspberry Pi або подібному мікрокомп'ютері з урахуванням можливостей пристроїв IoT і їх призначення.

Сервер виступає сховищем інформації, через який користувач може збирати окремі пакети за стандартним протоколом від певного пристрою або всіх одночасно.

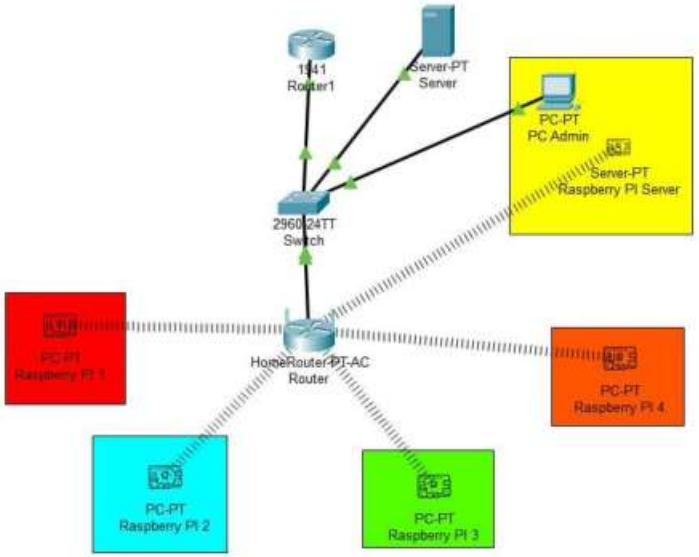


Рис.3. IoT-комплекс на базі локальної мережі

Аналіз проводиться на комп'ютері або ноутбуці користувача, що дозволяє не використовувати потужності основного пристрою.

Апробацію комплексу проведено для кабельних з'єднань і підключень через маршрутизатор з бездротовою точкою доступу. Для функціонування мережі IoT визначено критерії моніторингу, які дозволяють оптимізувати роботу адміністратора, виявляти проблеми і забезпечувати стабільну роботу системи моніторингу.

Такий комплекс може використовуватись в малому бізнесі, коли немає можливості, оплатити весь пакет захисту з обладнанням, або виступати тимчасовою альтернативою.

Список використаних джерел

1. Bala Bindu, Sunny Behal. AI techniques for IoT-based DDoS attack detection: Taxonomies, comprehensive review and research challenges. *Computer science review*. 2024. Vol. 52. P. 100631. DOI: 10.1016/j.cosrev.2024.100631.
2. Зростання кількості DDoS-атак на 46% у першому півріччі 2024 року. *10 guards*. URL: https://10guards.com/ua/blog/2024/09/23/surge-in-ddos-attacks-gcore-report-reveals-46-increase-in-first-half-of-2024/ (дата звернення: 01.10.2024).
3. Сідікі О. С., Пась Я. І., Тарчинець М. В. Проблеми інформаційної безпеки в контексті міжнародної співпраці туристичної індустрії в умовах посиленої євроінтеграції. *Наукові записки Львівського університету бізнесу та права.* 2024. № 40. С. 166–173. URL: https://nzlubp.org.ua/index.php/journal/article/view/1034 (дата звернення: 03.10.2024).
4. Савченко В. А., Кожухівський А. Д., Ільїн О. Ю. Діагностування початку повільної HTTP DDOS атаки на основі двопараметричного кореляційного аналізу трафіку. *Телекомунікаційні та інформаційні технології.* 2021. № 4 (73). DOI: 10.31673/2412-4338.2021.042840.
5. Метод виявлення DDoS атак на IoT мережі / Нічепорук А. О. та ін. *Вісник Хмельницького національного університету, Технічні науки*. 2020. № 1 (281). С.184–191. DOI: 10.31891/2307-5732-2020-281-1-184-191
6. Засоби ТЗІ, які мають експертний висновок про відповідність вимогам технічного захисту інформації. *Державна служба спеціального зв'язку та захисту інформації України.* URL: https://cip.gov.ua/ua/news/zasobi-tzi-yaki-mayut-ekspertnii-visnovok-pro-vidpovidnist-do-vimog-tekhnichnogo-zakhistu-informaciyi (дата звернення: 05.10.2024).

7. Distributed function chaining with anycast routing / Wion Adrien et al. *Proceedings of the 2019 ACM Symposium on SDN Research.* 2019. P. 91–97. DOI: 10.1145/3314148.3314355.

УДК 004.9:007.52

Дарнапук Є. С.,
старший викладач кафедри комп'ютерної інженерії, аспірант,
Гуляєв І. С.,
бакалаврант,
ЧНУ імені Петра Могили, м. Миколаїв, Україна

КОМЛЕКС ДИСТАНЦІЙНОГО СПОСТЕРЕЖЕННЯ НА БАЗІ КОЛІСНОЇ РОБОПЛАТФОРМИ ТА ESP32-CAM

В умовах сучасних загроз і викликів, з якими зіштовхується наша країна, зростає потреба у розвитку автономних та надійних систем для дистанційного спостереження і моніторингу небезпечних територій. Зокрема, це актуально для забезпечення безпеки на об'єктах інфраструктури та контрольованих зонах. Використання колісної робоплатформи з інтегрованим мікроконтролером ESP32-CAM [1–3] дозволяє створити ефективний та мобільний комплекс для оперативного отримання візуальної інформації з віддалених або небезпечних місць, мінімізуючи ризик для життя людей.

Концептуально, комплекс складається з наступних компонентів: центральний мікроконтролер ESP32-CAM, що відповідає за керування платформою та отриманням зображень навколишнього середовища з подальшим поширенням їх на клієнтські пристрої; колісна робоплатформа, L298N драйвер моторів та Li-Po батарей 18650 з контролером заряду.

Мікроконтролер ESP32-CAM (рис. 1) є ключовим елементом комплексу дистанційного спостереження, який використовується для отримання візуальних даних навколишнього середовища. Завдяки своїм можливостям, ESP32-CAM дозволяє інтегрувати відеоспостереження та аналіз зображень у мобільні роботизовані платформи, створюючи ефективну та економічно вигідну систему. Його двоядерний процесор Xtensa LX6 з тактовою частотою до 240 МГц забезпечує достатню обчислювальну потужність для обробки відеопотоків у режимі реального часу, що є критично важливим для оперативного реагування.

Військова освіта і наука: сьогодення та майбутнє : зб. тез доповідей XX Міжнародної науково-практичної конференції, м. Київ, 29 листопада 2024 р. Київ : Військовий інститут Київського національного університету імені Тараса Шевченка, 2024. 532 с.

Рекомендовано до друку Вченою радою Військового інституту Київського національного університету імені Тараса Шевченка
(протокол від 21.11.2024 № 3).

У збірнику тез доповідей друкуються матеріали виступів наукових і науково-педагогічних працівників, курсантів (студентів) Військового інституту Київського національного університету імені Тараса Шевченка та інших вищих військових та закладів вищої освіти України.

У публікаціях розглядаються: технічні проблеми озброєння і військової техніки та технології подвійного призначення; актуальні проблеми лінгвістичного забезпечення Збройних Сил України; актуальні питання військової психології та соціальної роботи; інформаційна та психологічна боротьба у воєнній сфері; інформаційно-медійне забезпечення МОУ та ЗСУ в умовах правового режиму воєнного стану; фінанси; актуальні проблеми військового права в умовах воєнного стану; актуальні проблеми геопросторової підтримки військ в умовах ведення російсько-української війни; наукові проблеми воєнної політології та морально-психологічного впливу; аналіз бойового застосування частин (підрозділів) Сухопутних військ Збройних Сил України у сучасному загальновійськовому бою (тактичних діях)

ВІЙСЬКОВИЙ ІНСТИТУТ
КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ЗБІРНИК ТЕЗ ДОПОВІДЕЙ

XX Міжнародної науково-практичної конференції

«Військова освіта і наука: сьогодення та майбутнє»

29 листопада 2024 року

Київ – 2024

Басистий В.А. (ХмНУ)
Чешун О.В. (ХмНУ)
к.т.н., доц. Чешун В.М. (ХмНУ)

ЗАСТОСУВАННЯ ОДНОПЛАТНИХ МІКРОКОМП'ЮТЕРІВ ДЛЯ ПІДВИЩЕННЯ СТІЙКОСТІ ІНТЕРНЕТУ РЕЧЕЙ ДО DDOS АТАК

Одним із найпопулярніших об'єктів DDoS-атак на сьогодні є мережі інтернету речей (IoT), значна кількість ресурсів яких є менш потужними і захищеними порівняно з серверами та робочими станціями традиційних комп'ютерних мереж, а тому вони є більш вразливими. Особливістю систем IoT є непристосованість і недостатність ресурсів у більшості їх пристроїв до завдань моніторингу мережевого трафіку, а покладання відповідних задач на сервери не завжди є ефективним або можливим.

Для підвищення стійкості систем IoT до DDoS та інших мережевих атак розглядається можливість утворення системи захисту із застосуванням одноплатних мікрокомп'ютерів Raspberry Pi або їх аналогів. Ціллю виконуваних робіт постало створення програмно-апаратного комплексу, який проводить розподілений моніторинг всієї мережі IoT, аналізує трафік, повідомляє і зменшує збитки від можливих атак на ресурси IoT.

Програмні складові комплексу або налаштування розгортаються на декількох пристроях, до числа яких входять: пристрої моніторингу; сервер, який збирає дані і надає доступ для їх подальшого аналізу; користувацькі персональні комп'ютери або ноутбука; комутатор або маршрутизатор безпровідного з'єднання.

Для розробки програмної складової комплексу було обрано мову програмування Python з бібліотеками: Psutil, Scapy, Pandas. Ця комбінація надає можливість отримувати детальну інформацію, активні процеси та підключені пристрої.

Програма моніторингу розгортається на мікрокомп'ютері Raspberry Pi або подібному мікрокомп'ютері з урахуванням можливостей пристроїв IoT і їх призначення.

Сервер виступає сховищем інформації, через який користувач може збирати окремі пакети за стандартним протоколом від певного пристрою або всіх одночасно.

Аналіз проводиться на комп'ютері або ноутбуку користувача, що дозволяє не використовувати потужності основного пристрою.

Апробацію комплексу проведено для кабельних з'єднань і підключень через маршрутизатор з бездротовою точкою доступу. Для функціонування мережі IoT визначено критерії моніторингу, які дозволяють оптимізувати роботу адміністратора, виявляти проблеми і забезпечувати стабільну роботу системи моніторингу.

Такий комплекс може використовуватись в малому бізнесі, коли немає можливості, оплатити весь пакет захисту з обладнанням, або виступати тимчасовою альтернативою.

Зміст

УДК 004:37:001:62

Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024». Хмельницький. 2024. 582с.

У збірнику наукових праць подані перспективні практичні розробки аспірантів, студентів та здобувачів в області сучасних інформаційних технологій. Розглянуто актуальні проблеми комп'ютерних наук, комп'ютерної інженерії, прикладної математики й інженерії програмного забезпечення, приведено ряд робіт по впровадженню інформаційних технологій у виробництво та управління. Висвітлено перспективні розробки сучасних систем пошуку, обробки й захисту інформації, медійних та комунікаційних системи.

УДК 004:37:001:62

Матеріали конференції відтворені з авторських оригіналів, друкуються в авторській редакції та наведені в алфавітному порядку прізвищ авторів. При макетуванні можливі незначні зміни компоновки контенту авторських оригіналів. Відповідальність за якість та зміст публікацій несе автор.

Участь у конференції та складові всіх її етапів (розгляд праць, перевірка на плагіат, макетування, публікація збірника наукових праць та видача сертифікатів) є безкоштовними для всіх учасників. Оргкомітет конференції висловлює подяку учасникам конференції та сподівається на подальшу співпрацю.

З питань проведення конференції та подальшого обміну інформацією звертатись на e-mail конференції: *apkt.khnu@gmail.com*

Міністерство освіти і науки України
Хмельницький національний університет

ЗБІРНИК НАУКОВИХ ПРАЦЬ

за матеріалами XVI Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2024»

15-16 листопада 2024

Хмельницький 2024

УДК 004.056.5

Басистий В.А., Чешун В.М., Чешун О.В.

Хмельницький національний університет

МЕРЕЖЕВА ІНФРАСТРУКТУРА ІНФОРМАЦІЙНОЇ БЕЗПЕКИ IOT НА ОДНОПЛАТНИХ МІКРОКОМП'ЮТЕРАХ

Розглянуто тенденції розвитку і супутні їм вразливості мереж інтернету речей, продемонстровано потенційну можливість утворення мережевої інфраструктури інформаційної безпеки IoT на одноплатних мікрокомп'ютерах, що дає високі показники захисту за відносно малі кошти. Наведено схемні рішення сегментів інфраструктури з бездротовим і кабельним з'єднанням.

The development trends and accompanying vulnerabilities of IoT networks are considered, the potential possibility of forming an IoT information security network infrastructure on single-board microcomputers, which provides high protection indicators for relatively small costs, is demonstrated. Schematic solutions of infrastructure segments with wireless and cable connection are presented.

Мережі Інтернету речей (IoT) швидко стали невід'ємною частиною сучасного бізнесу та побуту, об'єднуючи мільярди пристроїв і систем у єдину мережу. Вони охоплюють усе – від побутових пристроїв до промислового обладнання та систем критичної інфраструктури, як-от системи електропостачання чи охорони здоров'я. Розвиток IoT відкриває нові можливості, але водночас створює й нові ризики для безпеки. Попри свої переваги, IoT несе значні загрози через численні вразливості, що роблять ці мережі популярною мішенню для кіберзлочинців[1, 2].

Щодня різноманітні кіберзлочинні групи намагаються отримати несанкціонований доступ до інформаційних систем та ресурсів IoT. Із розвитком цифрових технологій та збільшенням кількості підключених до мережі пристроїв зловмисники стають більш витонченими, а кількість кіберзагроз та атак в інформаційному просторі IoT неухильно зростає.

Так, згідно з дослідженнями, опублікованими у статті [3], кількість DDoS-атак, що спрямовані на перевантаження та відмову в обслуговуванні систем, зросла на 68% у 2023 році порівняно з попереднім роком. Протягом першого півріччя 2024 року зростання кількості таких атак досягло вже 46%, а пікова потужність атаки досягла 1,7 Тбіт/с [4]. Ці показники демонструють масштаби та серйозність загроз, які можуть паралізувати роботу організацій та завдати значних фінансових збитків.

З огляду на численні вразливості IoT, компаніям і користувачам необхідно впроваджувати надійні заходи безпеки, щоб захистити себе від сучасних

ЗМІСТ

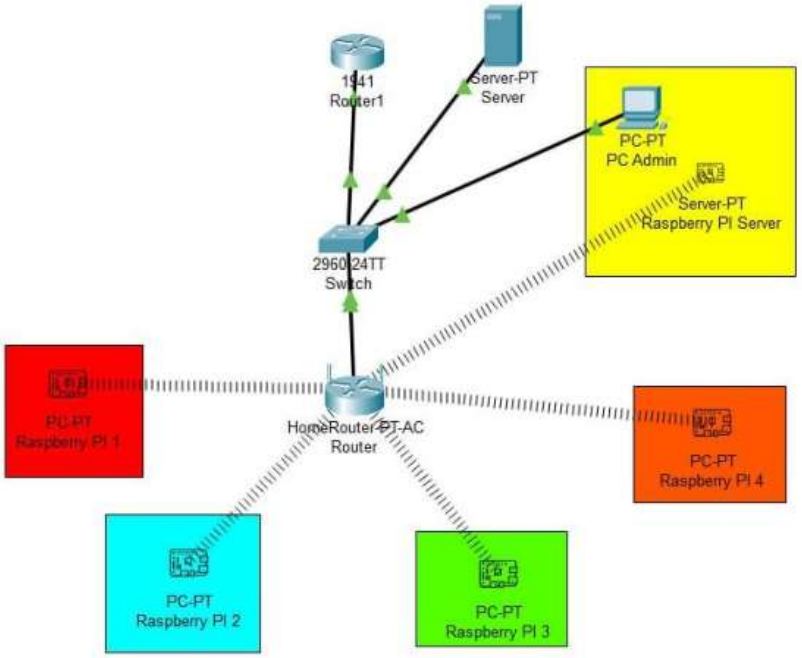


Рисунок 1 – IoT-комплекс з Raspberry Pi на базі локальної мережі

Різноманіття кольорової гами зображень Raspberry Pi (виділення прямокутниками) означає, що визначення пристрою за типом даних, способом моніторингу, місцем розташування або іншими відповідними критеріями буде робити адміністратор.

В якості місця зберіганням даних, використовується сервер Raspberry Pi, який надає можливість адміністратору фільтрувати трафік за критеріями і переглядати активність мережі.

Маршрутизатор з'єднує між собою сервер і пристрої для створення підмережі в якій будуть виконуватись основні задачі (мережеві налаштування за протоколом DHCP). Комутатор 2960 з'єднує безпровідне підключенням з провідним, що дозволяє ідентифікувати підмережу і пристрої, які зможуть її ідентифікувати. Маршрутизатор 1941 надає доступ до Інтернету та засоби захисту. Головний сервер з назвою «Server» надає службу DHCP для автоматичного конфігурування мережевих налаштувань.

Комп'ютер PC-Admin використовується для віддаленого доступу до засобів моніторингу для адміністратора, і з якого також вносяться корегування або

кіберзагроз.

Системи IoT мають особливу ваду – більшість пристроїв у них не розрахована на виконання складних задач, таких як моніторинг мережевого трафіку. Основна маса IoT-пристроїв мають обмежені ресурси, невелику обчислювальну потужність, низьку енергетичну ємність і обмежений обсяг пам'яті. Це унеможливлює встановлення на них засобів моніторингу та аналізу, які б могли постійно відстежувати потоки даних та своєчасно виявляти загрози.

Через такі обмеження IoT-систем часто виникає потреба перенести завдання моніторингу на централізовані сервери або хмарні платформи, які мають більше ресурсів для зберігання та обробки даних [5]. Однак таке рішення не завжди є оптимальним і практичним. По-перше, постійне передавання даних з кожного IoT-пристрою до сервера може значно перевантажувати мережу, особливо якщо йдеться про велику кількість пристроїв або великий обсяг даних. По-друге, залежність від центрального сервера створює додаткові ризики: у разі відмови або компрометації сервера вся система моніторингу може бути паралізована, що підвищує вразливість мережі до атак.

Крім того, централізовані сервери для обробки та зберігання даних можуть бути недостатньо ефективними для IoT-систем, які потребують швидкої реакції на загрози та події в реальному часі. Затримки, пов'язані з передачею даних до сервера і зворотно, можуть негативно вплинути на безпеку мережі та збільшити ймовірність успішної атаки[6].

Таким чином, зважаючи на особливості архітектури IoT, стає зрозумілим, що для забезпечення безпеки таких систем потрібні нові підходи, які враховують обмеження ресурсів та необхідність локального аналізу і захисту даних на рівні самих пристроїв.

Для розв'язку зазначеної проблеми пропонується застосувати одноплатні комп'ютери. Одноплатні комп'ютери (SBC) набули широкого використання в сучасних мережах Інтернету речей (IoT) завдяки своїй доступності, компактності та достатній обчислювальній потужності. Вони є ідеальним рішенням для побудови гнучких і масштабованих систем безпеки, які можуть інтегруватися в інфраструктуру IoT, забезпечуючи додатковий рівень захисту та моніторингу. Одноплатні комп'ютери, такі як Raspberry Pi, Banana Pi, Arduino, BeagleBone тощо, дозволяють виконувати завдання, які раніше вимагали повноцінних серверів, тим самим значно знижуючи витрати на інфраструктуру.

На рисунку 1 наведена схема мережевої інфраструктури інформаційної безпеки IoTз використанням SBC Raspberry Pi.

На схемі наведено декілька різновидів пристроїв, на яких здійснюється налаштування або розгортаються програмні складові системи безпеки: пристрої моніторингу, сервер (збирає дані і надає доступ для їх подальшого аналізу, користувацькі комп'ютери, комутатор або маршрутизатор бездротового з'єднання.

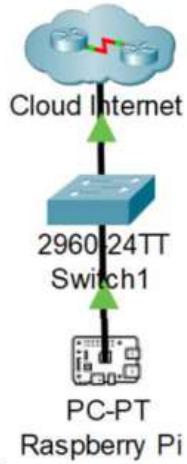


Рисунок 3 – Сегмент з дротовим підключенням

Використавши такий варіант підключення можна забезпечити надійний і стабільний зв'язок, який не обмежується перешкодами. Але це робить підключення пристроїв обмеженим, на кількість вільних портів комутатора.

Перелік посилань

1. Власенко М., Хлапонін Ю. Інтернет речей (IoT) у світовій практиці: огляд та аналіз. Pidvodni Tehnologii, 2024. №13. С.21-27.
2. Проблеми та загрози безпеці IoT пристроїв / Opirskyy, I. та ін. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка». 2021. №3(11). С. 31-42. https://doi.org/10.28925/2663-4023.2021.11.3142.
3. Bala Bindu, Sunny Behal. AI techniques for IoT-based DDoS attack detection: Taxonomies, comprehensive review and research challenges. Computer science review. 2024. Vol. 52. P. 100631.
4. Зростання кількості DDoS-атак на 46% у першому півріччі 2024 року. 10 guards. URL: https://10guards.com/ua/blog/2024/09/23/surge-in-ddos-attacks-gcore-report-reveals-46-increase-in-first-half-of-2024/ (дата звернення: 30.10.2024).
5. IoT Monitor Traffic: Unveiling a Smarter Approach to Monitoring Traffic. Temok.com. URL: https://www.temok.com/blog/iot-monitor-traffic/. (дата звернення: 30.10.2024)
6. Network Security Monitoring. Foresite cybersecurity. URL: https://foresite.com/blog/network-security-monitoring/ (дата звернення: 30.10.2024).

доповнення до налаштувань мережі.

Цей тип підключення, створюється для локальної мережі з метою збору даних з основних IoT-пристроїв, передачі їх на Server-Traffic для моніторингу.

Моделювання комплексу проведено для підключень через маршрутизатор з бездротовою точкою доступу і кабельних з'єднань.

Бездротове підключення (рисунок 2) вимагає використання маршрутизатора, який також має функцію бездротової точки доступу для підключення до Інтернету.

Такий варіант підключення, надає можливість підключати велику кількість пристроїв, з високою швидкістю. Завдяки таким можливостям, таке підключення є найкращим і обмежується тільки специфічним обладнанням і швидкістю передачі даних пристроїв, яку вони можуть надати.

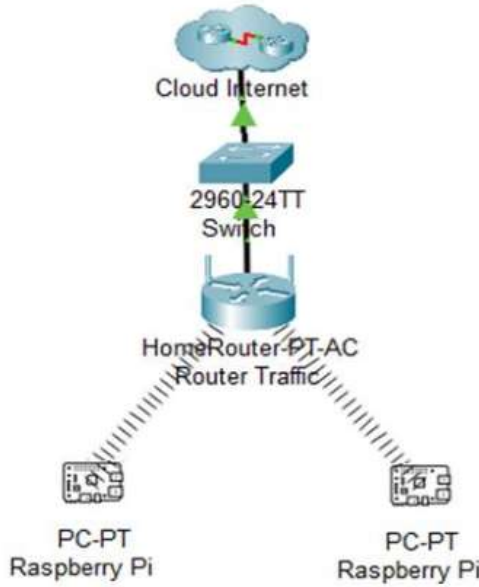


Рисунок 2 – Підключення з бездротовою точкою доступу

Cloud Internet надає IP-адресу для пристроїв в мережі. Комутатор отримує дані з пристроїв і Raspberry Pi збирає дані і надає їх через кабель або використовується, як точка, яка виконує операцію моніторингу трафіку в мережі.

У випадку, коли немає можливості реалізувати безпровідне підключення, є можливість підключити IoT-пристрій через кабель (рис. 3).

УДК 330.111.66:005.8
С 91

С 91 **Сучасні інформаційні системи та технології**: матеріали VII Всеукр. наук.-практ. інтернет-конф. за тематикою «Сучасні комп'ютерні системи та мережі в управлінні» (29 листопада 2024 р., м. Херсон, м. Хмельницький) / за ред. А. А. Григорової. – Херсон: Книжкове видавництво ФОП Вишемирський В. С., 2024. – 263 с.

ISBN 978-617-8187-36-1 (електронне видання)

Доповіді наукової конференції містять результати наступних досліджень: сучасні тенденції розвитку інформаційних технологій; впровадження інновацій та сучасних технологій; моделювання та оптимізація систем управління; інформаційні технології в науці, освіті, економіці, логістиці, туристичній сфері, транспорті; новітні технології в енергетичних системах та в галузі енергозбереження.

Роботи друкуються в авторській редакції, в збірці максимально зменшено втручання в обсяг та структуру відібраних до друку матеріалів. Редакційна колегія не несе відповідальність за достовірність статистичної та іншої інформації, що надано в рукописах, та залишає за собою право не розподіляти поглядів деяких авторів на ті чи інші питання.

Збірник становить інтерес для студентів, аспірантів, викладачів та наукових працівників.

УДК 330.111.66:005.8

ISBN 978–617–8187–36–1 (електронне видання)

Міністерство освіти і науки України
Херсонський національний технічний університет
Вінницький національний технічний університет
Криворізький національний університет
Кременчуцький національний університет ім. М. Остроградського
Хмельницький національний університет
Львівський національний університет природокористування

Матеріали

VII Всеукраїнської

науково-практичної інтернет-конференції

молодих вчених та студентів

«Сучасні інформаційні системи та технології»

за тематикою:

«Сучасні комп'ютерні системи та мережі в управлінні»

29 листопада 2024 року

Хмельницький

УДК 004.056.5

Басистий В.А.
студент 1 курсу спеціальності «Кібербезпека та захист інформації» ОПП «Кібербезпека та захист інформації»

Чешун О.В.
студент 2 курсу спеціальності «Інженерія програмного забезпечення» ОПП «Інженерія програмного забезпечення»

Чешун В.М.
канд. техн. наук, доцент кафедри кібербезпеки,

БАЗА ДАНИХ СИСТЕМИ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ ІНТЕРНЕТУ РЕЧЕЙ

Хмельницький національний університет, Україна

Постановка проблеми

Сучасне суспільство активно впроваджує пристрої Інтернету речей (IoT) у різні сфери життя: від домашніх господарств до великих промислових підприємств. Смарт-годинники, розумні колонки, системи моніторингу здоров'я, автоматизоване освітлення, відеоспостереження та «розумні» термостати є лише частиною цього розмаїтого світу. Завдяки IoT пристрої взаємодіють через інтернет, що значно спрощує виконання повсякденних завдань та підвищує якість життя [1].

Водночас, широке використання IoT несе із собою і серйозні ризики в сфері кібербезпеки [2,3]. Багато IoT пристроїв мають слабкі паролі або зовсім позбавлені шифрування даних, що робить їх легкою мішенню для хакерів. Через IoT пристрої хакери можуть отримати доступ до персональних даних користувачів, включно з їхнім місцезнаходженням, звичками і навіть розмовами. У випадках з індустріальними IoT пристроями, злом може спричинити зупинку критичних систем, таких як енергетичні станції або медичні пристрої.

З урахуванням широкого розповсюдження та великої важливості IoT в житті сучасного суспільства, що одночасно супроводжується суттєвими ризиками і загрозами в аспекті кібербезпеки, технології підвищення кіберстійкості мереж IoT до різних видів атак сьогодні набувають особливої важливості [3,4].

Аналіз останніх досліджень та публікацій

Питанням забезпечення безпеки мереж IoT сьогодні приділяється багато уваги. В роботах [3,4] описується технологія виявлення DDoS атак на інфраструктуру IoT, автори [5] досліджують проблеми та загрози безпеці IoT пристроїв. В зазначених та багатьох інших публікаціях робиться наголос на вразливості мереж і пристроїв IoT до кібератак та потребі розробки і впровадження ефективних рішень для захисту IoT.

В [6] пропонуються рішення з захисту IoT пристроїв у наявній інфраструктурі комп'ютерної мережі закладу освіти, в [7] запропонований варіант побудови захищеної мережі IoT з використанням блокчейн-технології, але ці та більшість інших рішень з безпеки IoT є вузькоаспектними і передбачають обмежене застосування.

Постановка задачі

Дана робота є продовженням досліджень авторів, присвячених побудові комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах.

В роботі [8] авторами продемонстровано потенційну можливість утворення мережевої інфраструктури інформаційної безпеки IoT на одноплатних мікрокомп'ютерах, що дає високі показники захисту за відносно малі кошти, а також запропоновано схемні рішення сегментів інфраструктури з бездротовим і кабельним з'єднанням.

– таблиця «IoT-пристрої» ідентифікує наявні пристрої IoT і містить такі параметри: Id пристрою, назва і опис пристрою. Ця таблиця надає основну інформацію про трафік даних, які можуть бути використані для аналізу і вдосконалення продуктивності мережі.

– таблиця «Активні процеси» містить інформацію про Id процесу, назву процесу, час створення і його опис. таблиця містить інформацію про завантаження даних, вивантаження даних, швидкість завантаження і вивантаження, час виміру, назву даних, тип даних і приклад запису даних. Ця таблиця надає основну інформацію про трафік даних, які можуть бути використані для аналізу і вдосконалення продуктивності мережі;

– таблиці «Мережеві інтерфейси» і «Запис за інтерфейсами» містять інформацію мережеві інтерфейси і про параметри трафіку мережевих інтерфейсів, за яким можна визначити завантаженість інтерфейсу: Mac-адресу, IP-адресу, Інтернет шлюз, DNS сервер і опис. Ці параметри можуть оновлюватись і добавлятись в залежності від даних, які приймаються пристроями. Основні параметри таблиці «Мережеві інтерфейси», такі як: Id процесу, назва процесу, час створення є початковою необхідністю, за якою визначають процес. За допомогою даних цих таблиць можна визначити кожен процес, який створюється і здійснити класифікацію процесу. Таблиця «Запис за інтерфейсами» надає основну інформацію про інтерфейс, за якою можна класифікувати і виявляти інциденти, які можуть виникнути на цьому рівні.

– таблиця «Активні програми» є проміжною між процесами та пристроями IoT і містить дані для ідентифікації програмного продукту, що має вразливості або несе загрози.

Висновки

Запропоновані утиліта моніторингу мережевого трафіку і база даних орієнтовані на використанні в роботі комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах. Такий комплекс може використовуватись в малому бізнесі, коли немає можливості оплатити потужний пакет захисту з обладнанням, або виступати його тимчасовою альтернативою.

Перелік джерел посилання

1. Власенко М., Хлапонін Ю. Інтернет речей (IoT) у світовій практиці: огляд та аналіз. *Pidvodni Tehnologii*, 2024. №13. С.21-27.

2. Проблеми та загрози безпеці IoT пристроїв / Opirskyy, I. та ін. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*. 2021. №3(11). С. 31-42. https://doi.org/10.28925/2663-4023.2021.11.3142.

3. Bala Bindu, Sunny Behal. AI techniques for IoT-based DDoS attack detection: Taxonomies, comprehensive review and research challenges. *Computer science review*. 2024. Vol. 52. P. 100631. DOI: https://doi.org/10.1016/j.cosrev.2024.100631 (дата звернення: 1.10.2024).

4. Метод виявлення DDOS атак на IOT мережі / Нічепорук А.О. та ін. *Вісник Хмельницького національного університету, Технічні науки*. 2020. №1 (281). С.184-191.

5. Проблеми та загрози безпеці IoT пристроїв / Опірський І. та ін. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*. 2021. №3(11), С.31–42. DOI: https://doi.org/10.28925/2663-4023.2021.11.3142 (дата звернення: 12.11.2024).

6. Інтегрування та захист IoT пристроїв у наявній інфраструктурі комп'ютерної мережі закладу освіти / Лахно В. та ін. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*. 2021. №3(11), С.85–99. DOI: https://doi.org/10.28925/2663-4023.2021.11.8599. (дата звернення: 12.11.2024).

7. Л. В. Чепель, Ю. В. Бойко. Підхід до безпеки та організації мереж IoT з використанням блокчейн технології. *Вісник Вінницького політехнічного інституту*. 2024. №4. С.129-138.

8. Басистий В.А., Чешун В.М., Чешун О.В. Мережева інфраструктура інформаційної безпеки IoT на одноплатних мікрокомп'ютерах. *Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024»*. Хмельницький. 2024. С.358-362.

Для отримання повноцінного комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах постала задача розробки програмної складової, яка здатна вести облік і обробку даних кіберінцидентів та приймати і реалізовувати рішення з протидії кіберзагрозам.

Виклад основного матеріалу

Для забезпечення ефективної роботи комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах було проаналізовано і визначено критерії моніторингу, які дозволяють оптимізувати роботу адміністратора, виявляти проблеми і забезпечувати стабільну роботу системи моніторингу.

Для здійснення моніторингу і аналізу трафіку створена утиліта мовою програмування Python з використанням бібліотек Psutil, Scapy, Pandas на базі ядра Linux, що надає можливість запускати користувачу потрібну кількість пристроїв для моніторингу не обмежуючись пристроями, які можуть не підтримуватись програмно або апаратно операційною системою Windows або MacOS.

Структура бази даних комплексу моніторингу і аналізу мережевого трафіку IoT представлена на рисунку 1.

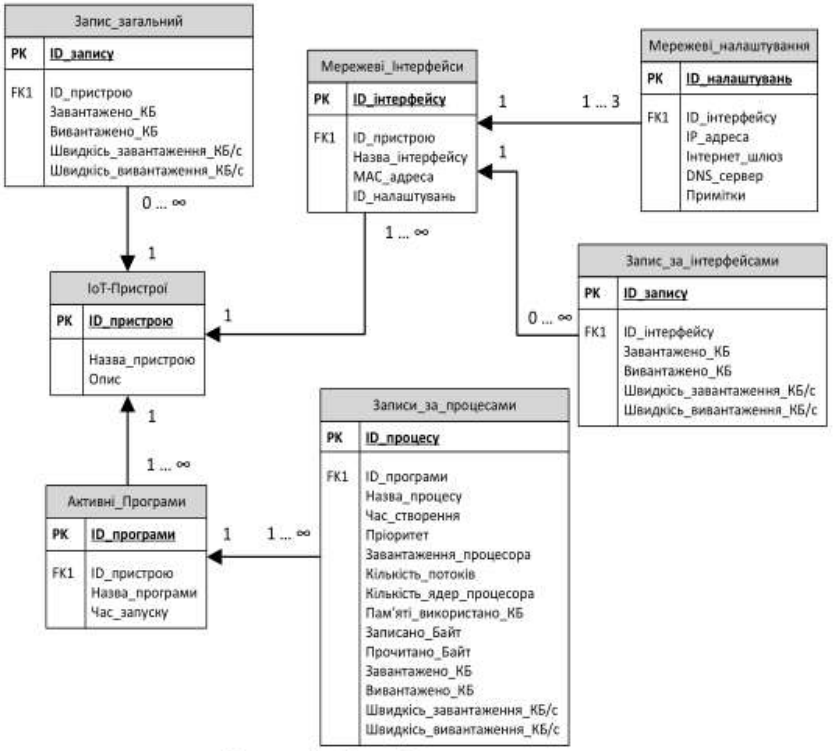


Рисунок 1 – Схема бази даних комплексу

База даних містить основні таблиці наступного призначення:

УДК 004
Т11

Рекомендовано до друку Вченою радою Державного університету «Житомирська політехніка» (протокол № 6 від 24.03.2025 р.)

Т11 Тези XV Міжнародної науково-технічної конференції «Інформаційно-комп'ютерні технології», м. Житомир, 28-29 березня 2025 р. – Житомир: Житомирська політехніка, 2025. – 352 с.

ISBN 978-966-683-698-7

Представлено доповіді учасників XV Міжнародної науково-технічної конференції. Наведено аналіз та результати досліджень сучасних проблем інформаційних технологій, математичного моделювання та розробки програмного забезпечення, інформаційних систем, комп'ютерної інженерії та кібербезпеки, цифрової обробки сигналів та зображень, комп'ютерно-інтегрованих технологій, робототехніки та приладобудування, інформаційних технологій в телекомунікаціях та біомедицині, інформаційно-комунікаційних технологій в освіті.

УДК 004

ISBN 978-966-683-698-7

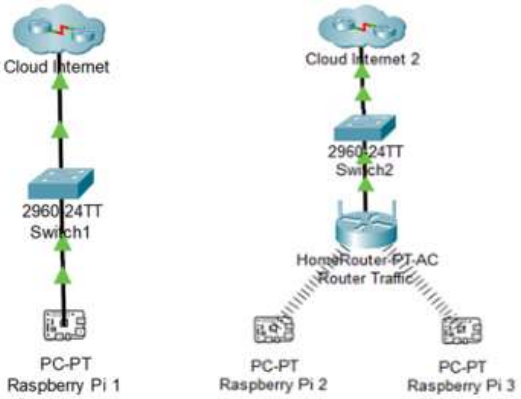
Наукове видання

Тези XV Міжнародної науково-технічної конференції «Інформаційно-комп'ютерні технології», Житомир, 28-29 березня 2025 р.

Відповідальний за випуск В.В. Болотіна

Свідоцтво про внесення до Державного реєстру суб'єктів видавничої справи ДК № 7177 ВІД 04.11.2021 р.

Адреса редакції: Державний університет «Житомирська політехніка», вул. Чуднівська, 103, м.Житомир, 10005



а) б)

Рис. 1 – Організація комутаційних підключень в системі моніторингу: а) кабельне; б) з бездротовою точкою доступу

У випадку, коли немає можливості реалізувати безпровідне підключення, є можливість підключити IoT-пристрій через кабель (рис. 1.а). Використавши такий варіант підключення можна забезпечити надійний і стабільний зв'язок, який не обмежується перешкодами, але це робить підключення пристроїв обмеженим через кількість портів комутатора.

Бездротове підключення (рис. 1.б) вимагає використання маршрутизатора з функцією бездротової точки доступу для підключення до Інтернету. Цей варіант підключення надає можливість підключати велику кількість пристроїв, з високою швидкістю, тому воно є найкращим і обмежується тільки специфічним обладнанням і швидкістю передачі даних пристроїв.

Список використаних джерел:

1. Метод виявлення DDOS атак на IOT мережі / Нічепорук А.О. та ін. Вісник Хмельницького національного університету, Технічні науки. 2020. №1 (281). С.184-191.
2. Засоби моніторингу мережі в IoT інфраструктурі з гібридною архітектурою / Каплунов А. В., Гайдай А. Р., Гер В. М., Нікольський С. С. Телекомунікаційні та інформаційні технології. 2023. № 2(79). С.22-32.
3. Басистий В.А., Чешун О.В., Чешун В.М. Комлекс моніторингу і аналізу мережевого трафіку IOT на одноплатних мікрокомп'ютерах. Тези доповідей XXVII Всеукраїнської НПК «Могилянські читання – 2024», Технічні науки. С.103-108.

УДК 004.056.5

Басистий В. А., магістрант
Чешун О. В., здобувач
Чешун В. М., к.т.н., доцент
Хмельницький національний університет

ОРГАНІЗАЦІЯ КОМУТАЦІЙНИХ ПІДКЛЮЧЕНЬ МІКРОКОМП'ЮТЕРІВ В КОМЛЕКСІ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ IOT

Системи IoT відіграють все більшу роль в різних сферах життєдіяльності людини, але мають значний недолік – вразливість до кібератак [1]. Більшість пристроїв і систем IoT не здатні виконувати завдання захисту, зокрема, здійснювати моніторинг мережевого трафіку. Це пояснюється їхніми обмеженими ресурсами: низькою обчислювальною потужністю, невеликим обсягом пам'яті та обмеженою енергетичною ємністю. Через ці фактори встановлення засобів для постійного моніторингу та аналізу даних безпосередньо на IoT-пристроях стає неможливим.

Внаслідок таких обмежень завдання з моніторингу часто передаються на централізовані сервери або хмарні платформи, які мають достатньо ресурсів для обробки великих обсягів даних [2]. Однак, безперервне передавання даних від великої кількості пристроїв до сервера може спричинити перевантаження мережі, а обробка даних на централізованих серверах може бути недостатньо швидкою для IoT-систем, які потребують миттєвої реакції на загрози.

В [3] авторами запропоноване рішення для створення комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах.

Базовим елементом комплексу є мікрокомп'ютер Raspberry Pi (можлива реалізація на інших одноплатних мікрокомп'ютерах), на якому розгортається програма моніторингу. Для розробки програмної складової комплексу було обрано мову програмування Python з бібліотеками Psutil, Scapy, Pandas, що дає можливість отримувати детальну інформацію про активні процеси та підключені пристрої.

Сховищем інформації виступає сервер, який дозволяє користувачу збирати окремі пакети за стандартним протоколом від певного пристрою або від всіх одночасно. Аналіз проводиться на комп'ютері користувача без використання ресурсу основного пристрою.

Комплекс передбачає можливість підключень на основі кабельних з'єднань і через маршрутизатор з бездротовою точкою доступу (рис. 1).

ЗМІСТ

СЕКЦІЯ 1. ІНФОРМАЦІЙНА БЕЗПЕКА ДЕРЖАВИ, СУСПІЛЬСТВА ТА ОСОБИСТОСТІ

УДК 004.4

Матеріали VIII Міжнародної науково-практичної конференції "Інформаційна безпека та комп'ютерні технології": тези доповідей, 24-25 квітня 2025 р. – Кропивницький: ЦНТУ, 2025. – 97 с.

Наведені тези пленарних та секційних доповідей за теоретичними та практичними результатами наукових досліджень і розробок. Представлені результати теоретичних досліджень в галузях проектування інформаційних систем, технологій захисту інформації, використання сучасних інформаційних технологій в управлінні системами за різними галузями народного господарства.

Матеріали публікуються в авторській редакції.

За достовірність викладених фактів, цитат та інших відомостей відповідальність несуть автори.

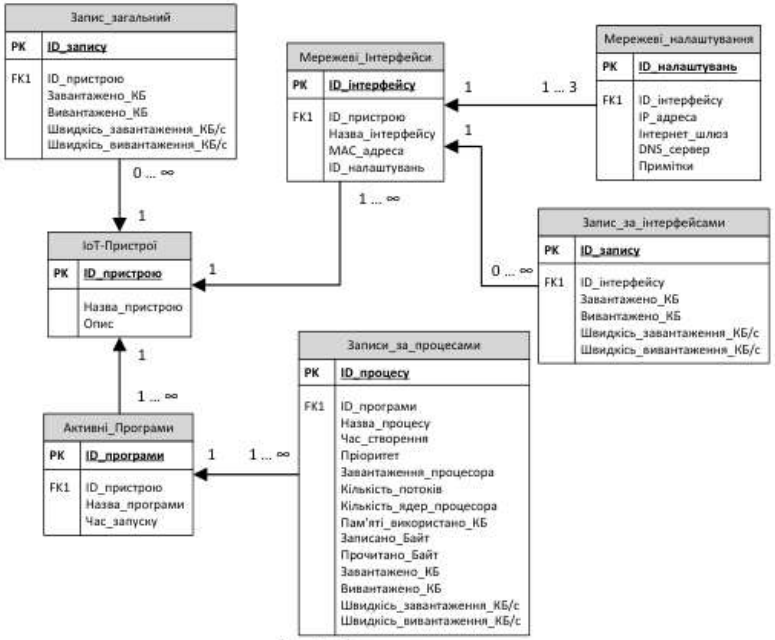


Рис. 1. База даних комплексу

Таблиці «Мережеві інтерфейси» і «Запис за інтерфейсами» містять інформацію мережеві інтерфейси і про параметри трафіку мережевих інтерфейсів, за яким можна визначити завантаженість інтерфейсу: Mac-адресу, IP-адресу, Інтернет шлюз, DNS сервер і опис. Ці параметри можуть оновлюватись і добавлятись в залежності від даних, які приймаються пристроями. Основні параметри таблиці «Мережеві інтерфейси», такі як: Id процесу, назва процесу, час створення є початковою необхідністю, за якою визначають процес. За допомогою даних цих таблиць можна визначити кожен процес, який створюється і здійснити класифікацію процесу. Таблиця «Запис за інтерфейсами» надає основну інформацію про інтерфейс, за якою можна класифікувати і виявляти інциденти, які можуть виникнути на цьому рівні.

Запропоновані утиліта моніторингу мережевого трафіку і база даних орієнтовані на використанні в роботі комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах. Такий комплекс може використовуватись в малому бізнесі, коли немає можливості оплатити потужний пакет захисту з обладнанням, або виступати його тимчасовою альтернативою.

Список літератури

1. Засоби моніторингу мережі в IoT інфраструктурі з гібридною архітектурою / Каплунов А. В., та ін. Телекомунікаційні та інформаційні технології. 2023. № 2(79). С.22-32.

2. Власенко М., Хлапонін Ю. Інтернет речей (IoT) у світовій практиці: огляд та аналіз. Pidvodni Tehnologii, 2024. №13. С.21-27.

3. Проблеми та загрози безпеці IoT пристроїв / Opirskyy, I. та ін. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка». 2021. №3(11). С. 31-42.

4. IoT Monitor Traffic: Unveiling a Smarter Approach to Monitoring Traffic. Temok.com. URL: https://www.temok.com/blog/iot-monitor-traffic/. (date of access:30.03.2025).

5. Басистий В.А., Чешун О.В., Чешун В.М. Комплекс моніторингу і аналізу мережевого трафіку IOT на одноплатних мікрокомп'ютерах. Тези доповідей XXVII Всеукраїнської НПК «Могилянські читання – 2024», Технічні науки. С.103-108.

УДК 004.056.5

В.А. Басистий[1], В.М. Чешун[1], Д.В. Чешун[2], А.Р. Школьняк[1]
basistavitalij@gmail.com, cheshunvn@khmnu.edu.ua, dmytro.cheshun@gmail.com, androidartem2014@gmail.com
Хмельницький національний університет, Хмельницький
[2]Хмельницький фаховий економіко-технологічний коледж, Хмельницький

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ МІКРОКОМП'ЮТЕРНОЇ СИСТЕМИ ЗАХИСТУ IOT

Системи IoT, поряд з широким розповсюдженням, стрімким розвитком і великими перспективами на майбутнє, мають значний недолік – більшість пристроїв не здатні виконувати складні завдання інформаційної безпеки, такі як, наприклад, моніторинг мережевого трафіку[1]. Це пояснюється їхніми обмеженими ресурсами: низькою обчислювальною потужністю, невеликим обсягом пам'яті та обмеженою енергетичною ємністю. Через ці фактори встановлення засобів для постійного моніторингу та аналізу даних безпосередньо на IoT-пристроях стає неможливим. Численні вразливості роблять мережі IoT популярною мішенню для кіберзлочинців[2,3].

Внаслідок таких обмежень завдання з моніторингу часто передаються на централізовані сервери або хмарні платформи, які мають достатньо ресурсів для обробки великих обсягів даних. Однак і цей підхід має свої недоліки. По-перше, безперервне передавання даних від великої кількості пристроїв до сервера може спричинити перевантаження мережі, особливо якщо обсяг переданої інформації значний. По-друге, централізована система створює ризик повної зупинки у разі виходу з ладу сервера або його компрометації, що робить всю мережу вразливою до атак.

Крім того, обробка даних на централізованих серверах може бути недостатньо швидкою для IoT-систем, які потребують миттєвої реакції на загрози. Затримки, пов'язані з передаванням інформації до сервера та отриманням зворотного зв'язку, підвищують ймовірність успішної атаки та негативно впливають на загальну безпеку[4].

Таким чином, обмеження архітектури IoT вимагають нових підходів до забезпечення безпеки. Нобхідно розробляти рішення, які дозволять проводити локальний аналіз та захист даних на рівні самих пристроїв, враховуючи їхні обмежені ресурси.

В [5] авторами запропоноване рішення для створення комплексу моніторингу і аналізу мережевого трафіку IoT на одноплатних мікрокомп'ютерах.

Базовим елементом комплексу є мікрокомп'ютер Raspberry Pi (можлива реалізація на інших одноплатних мікрокомп'ютерах), на якому розгортається програма моніторингу. Для розробки програмної складової комплексу було обрано мову програмування Python з бібліотеками Psutil, Scapy, Pandas, що дає можливість отримувати детальну інформацію про активні процеси та підключені пристрої. Сховищем інформації виступає сервер, який дозволяє користувачу збирати окремі пакети за стандартним протоколом від певного пристрою або від всіх одночасно. Аналіз проводиться на комп'ютері користувача без використання ресурсу основного пристрою.

Після визначення концепції організації і функціонування мікрокомп'ютерної системи захисту IoT постала задача розробки програмного забезпечення зазначеної системи.

Для моніторингу і аналізу трафіку, була створена утиліта мовою програмування Python з використанням бібліотек Psutil, Scapy, Pandas [4] на базі ядра Linux, що надає можливість запускати певну кількість пристроїв для моніторингу, яка буде потрібна користувачеві, не обмежуючись пристроями, які можуть не підтримуватись програмно або апаратно операційною системою Windows або MacOS.

Структура бази даних комплексу моніторингу і аналізу мережевого трафіку IoT представлена на рис. 1.

Розглянемо призначення основних таблиць бази даних, що наведені на рис.1

Таблиця «IoT-пристрої» ідентифікує наявні пристрої IoT і містить такі параметри: Id пристрою, назва і опис пристрою. Ця таблиця надає основну інформацію про трафік даних, які можуть бути використані для аналізу і вдосконалення продуктивності мережі.

Таблиця «Активні процеси» містить інформацію про Id процесу, назву процесу, час створення і його опис. таблиця містить інформацію про завантаження даних, вивантаження даних, швидкість завантаження і вивантаження, час виміру, назву даних, тип даних і приклад запису даних. Ця таблиця надає основну інформацію про трафік даних, які можуть бути використані для аналізу і вдосконалення продуктивності мережі.

Таблиця «Активні програми» є проміжною між процесами та пристроями IoT і містить дані для ідентифікації програмного продукту, що має вразливості або несе загрози.

УДК 004:37:001:62

Збірник наукових праць за матеріалами XVII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2025». Хмельницький. 2025. 500с.

У збірнику наукових праць подані перспективні практичні розробки аспірантів, студентів та здобувачів в області сучасних інформаційних технологій. Розглянуто актуальні проблеми комп'ютерних наук, комп'ютерної інженерії, прикладної математики й інженерії програмного забезпечення, приведено ряд робіт по впровадженню інформаційних технологій у виробництво та управління. Висвітлено перспективні розробки сучасних систем пошуку, обробки й захисту інформації, медійних та комунікаційних системи.

УДК 004:37:001:62

Матеріали конференції відтворені з авторських оригіналів, друкуються в авторській редакції та наведені в алфавітному порядку прізвищ авторів. При макетуванні можливі незначні зміни компоновки контенту авторських оригіналів. Відповідальність за якість та зміст публікацій несе автор.

Участь у конференції та складові всіх її етапів (розгляд праць, перевірка на плагіат, макетування, публікація збірника наукових праць та видача сертифікатів) є безкоштовними для всіх учасників. Оргкомітет конференції висловлює подяку учасникам конференції та сподівається на подальшу співпрацю.

З питань проведення конференції та подальшого обміну інформацією звертатись на e-mail конференції: *apkt.khnu@gmail.com*

Міністерство освіти і науки України
Хмельницький національний університет

ЗБІРНИК НАУКОВИХ ПРАЦЬ

за матеріалами XVII Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2025»

14-15 листопада 2025

Хмельницький 2025

УДК 004.056.5

Басистий В.А., Городецька А.О., Чешун В.М., Чешун О.В.

Хмельницький національний університет

ФІЗИЧНІ ТОПОЛОГІЇ РОЗГОРТАННЯ АГЕНТНОЇ СИСТЕМИ МОНІТОРИНГУ МЕРЕЖЕВОГО ТРАФІКУ IOT

Розглядається мультиагентний підхід до моніторингу мережевого трафіку IoT, що базується на принципах розподілених систем. Запропоновано три фізичні топології розгортання агентної системи – безпровідну, провідну та комбіновану, які забезпечують гнучкість, масштабованість і надійність аналізу. Моделювання в Cisco Packet Tracer підтвердило ефективність архітектури та її придатність для виявлення кіберзагроз у реальному часі.

A multi-agent approach to monitoring IoT network traffic based on distributed systems principles is considered. Three physical topologies for deploying the agent system are proposed - wireless, wired, and combined, which provide flexibility, scalability, and reliability of analysis. Simulation in Cisco Packet Tracer confirmed the effectiveness of the architecture and its suitability for real-time detection of cyber threats.

Проєктування архітектури будь-якої технічної системи є визначальним етапом, оскільки саме архітектурна невідповідність досить часто стає ключем проблеми її застосування. Замість спроб адаптувати централізовану модель, пропонується підхід, що базується на парадигмі розподілених систем – мультиагентних систем (MAS - Multi-agent system) [1].

Проєктована мультиагентна система являє собою сукупність автономних програмних сутностей (агентів), які взаємодіють між собою для вирішення складної задачі, що не може бути вирішена одним агентом. Можна ідентифікувати наступні ключові властивості програмних агентів, що роблять цей підхід ідеальним для цілей забезпечення кібербезпеки в IoT:

– автономність – здатність діяти без прямого втручання людини;

– реактивність – здатність своєчасно реагувати на зміни в оточенні;

– проактивність – здатність діяти за власною ініціативою;

– соціальність – здатність взаємодіяти з іншими агентами для досягнення спільної мети.

Головна перевага MAS полягає в архітектурному ізоморфізмі. Замість збору трафіку в єдину точку, MAS "приносить" аналітичні обчислення безпосередньо до джерел трафіку. Пропонована агентна система може бути розгорнута у трьох основних фізичних топологіях, що ілюструють її гнучкість. Моделювання цих

ЗМІСТ

топологій було виконано за допомогою програмного симулятора Cisco Packet Tracer, який дозволяє віртуально створювати та тестувати мережеві інфраструктури.

На всіх трьох схемах, компоненти системи мають наступне призначення:

– вузли (PC-PT) – одноплатні комп'ютери (ОПК), на яких розгорнуто програмний комплекс агентів моніторингу;

– центральний вузол (Server-PT) – внутрішній сервер. Він ідеально підходить для розгортання MQTT-брокера (для координації агентів) та сервера SIEM (для прийому фінальних звітів);

– мережеве обладнання (HomeRouter, Switch) забезпечує зв'язність мережі. Маршрутизатор також виступає як шлюз до інтернету;

– інтернет (Ethernet (Cloud)) – зовнішня, неконтрольована мережа, що є основним джерелом зовнішніх загроз;

– пристрій адміністратора (Laptop-PT Admin) – робоча станція адміністратора. У провідних сценаріях він підключений через консольний кабель для прямого налаштування комутатора.

На рисунку 1 представлено схематичне зображення безпровідної топології мережі.

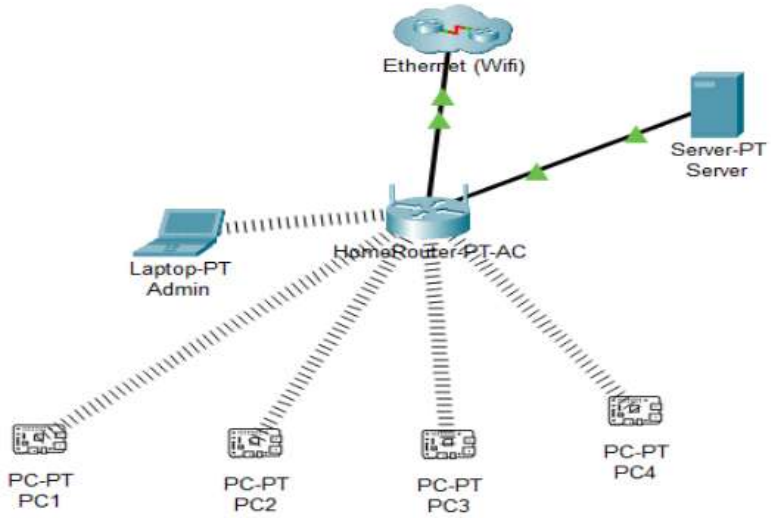


Рисунок 1 – Безпровідна (Wi-Fi) топологія розгортання системи

У даній топології всі пристрої (PC-PT PC1-PC4, Laptop-PT Admin та Server-PT Server) підключені до мережі через безпровідний маршрутизатор HomeRouter-PT-AC.

Захоплення відбувається в наступний спосіб. Вузли ОПК переводяться в режим моніторингу. Це дозволяє кожному агенту пасивно моніторити та захоплювати весь Wi-Fi трафік у своєму радіусі дії, використовуючи системні бібліотеки, такі як libpcap [2]. Таким чином досягається гнучкість, мобільність, легкість масштабування та виявлення специфічних Wi-Fi атак.

На рисунку 2 представлено схематичне зображення провідної топології мережі.

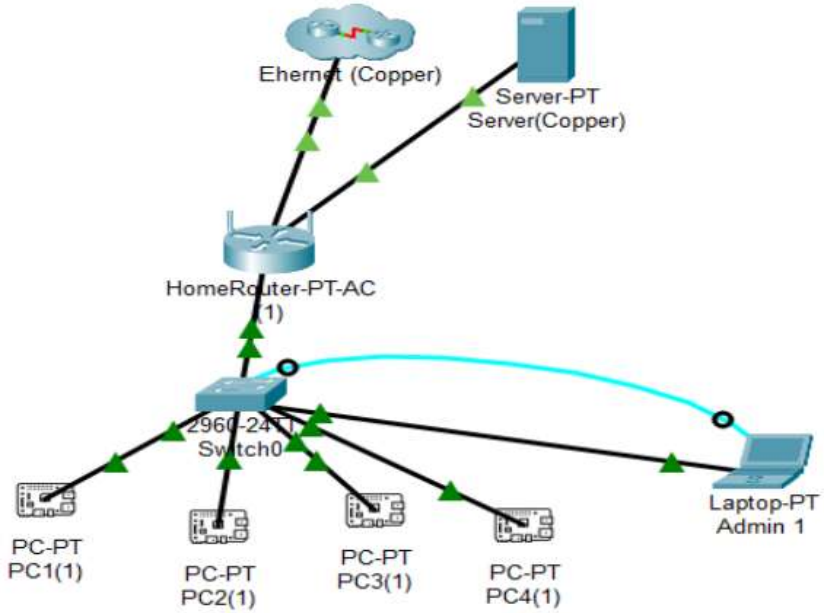


Рисунок 2 – Провідна (Ethernet) топологія розгортання системи

У провідній конфігурації пристрої (PC-PT PC1(1)-PC4(1)) підключені до центрального керованого комутатора (Switch0). Комутатор підключений до маршрутизатора HomeRouter-PT-AC(1), який також обслуговує Server-PT Server(Copper) та має вихід в Інтернет.

Спосіб захоплення полягає в наступному. Використовується технологія SPAN (Switched Port Analyzer) на керованому комутаторі Switch0. Технологія SPAN є функцією мережевих комутаторів, що дозволяє дзеркалізувати (копіювати) мережевий трафік з одного або кількох портів на інший порт для моніторингу безпеки або аналізу продуктивності. Комутатор Switch0 налаштовується на

Ключова перевага гібридної топології – хоча фізично трафік збирається різними агентами з різних середовищ, логічно вони становлять єдину систему. Так досягається вся потужність агентного підходу.

Використовуючи агентів-координаторів та спільний MQTT-брокер (на Server-Combo), система може виявляти складні атаки, що перетинають сегменти. Наприклад, якщо зловмисник *з інтернету* починає одночасно сканувати Wi-Fi та провідні пристрої, агент-координатор скорелює алерти від *обох* агентів в єдиний, більш серйозний "інцидент" безпеки [4].

Висновки. Запропонована мультиагентна архітектура доводить переваги розподіленого підходу для моніторингу IoT-трафіку, дозволяючи «приносити аналітику» до джерел замість централізованого збору. Агентна модель забезпечує автономність, реактивність, проактивність і соціальність – властивості, критичні для виявлення та локалізації атак у реальному часі. Три фізичні топології (безпровідна, провідна, комбінована) демонструють гнучкість розгортання: Wi-Fi – мобільність і масштабованість; Ethernet – надійність захоплення через SPAN; гібридний варіант – комплексний моніторинг мережі. Моделювання в Cisco Packet Tracer підтвердило практичну застосовність схем і дозволило відпрацювати сценарії захоплення та кореляції подій.

Перелік посилань

1. The confluence of evolutionary computation and multi-agent systems: A survey / T.Y. Chen et al. IEEE/CAA J. Autom. Sinica. 2025. P. 1-20. DOI: 10.1109/JAS.2025.125246 (date of access: 23.10.2025).
2. Cheng C., Ren S., Wang J. Implementation of Wireless Network Quality Evaluation Method for Agent Twins Based on Active Passive Dual Periodic Collaborative Measurement. 5th International Conference on Neural Networks, Information and Communication Engineering (NNICE), Guangzhou, China, 2025, P. 937-944. DOI: 10.1109/NNICE64954.2025.11064354 (date of access: 23.10.2025).
3. Saad R. A., Alkekli F. B., Gshera H. B. Network Monitoring Using Port-Mirroring Technology (SPAN and RSPAN). SUCP. 2025. Vol. 4, № 1. P. 84-89.
4. Resilient topology optimization on cyber-physical system of distribution networks under FDIA-Worm hybrid attacks / Li Huifeng et al. Journal of Electric Power Science and Technology. 2024. Vol. 39: Iss. 4. Article 3. P. 20-32. DOI: 10.19781/j.issn.1673-9140.2024.04.003 (date of access: 25.10.2025).

копіювання всього трафіку (наприклад, з порту, що веде до маршрутизатора) на один з портів, до якого підключений наш ОПК-агент (наприклад, PC-PT PC1(1)) [3].

Таким чином досягається повна видимість та висока надійність захоплення всього провідного трафіку без втрат, що є критичним для точного аналізу.

На рисунку 3 представлено схематичне зображення комбінованої топології мережі. Мережа складається з двох сегментів: провідного (PC-PT PC1-Combo...) та безпровідного (PC-PT PC1 Wifi-Combo...). Маршрутизатор HomeRouter-PT-PC Combo об'єднує ці сегменти, обслуговує сервер Server-PT Server-Combo та має вихід в Інтернет. У даній конфігурації, топологія є найбільш ефективною і поєднує обидва підходи.

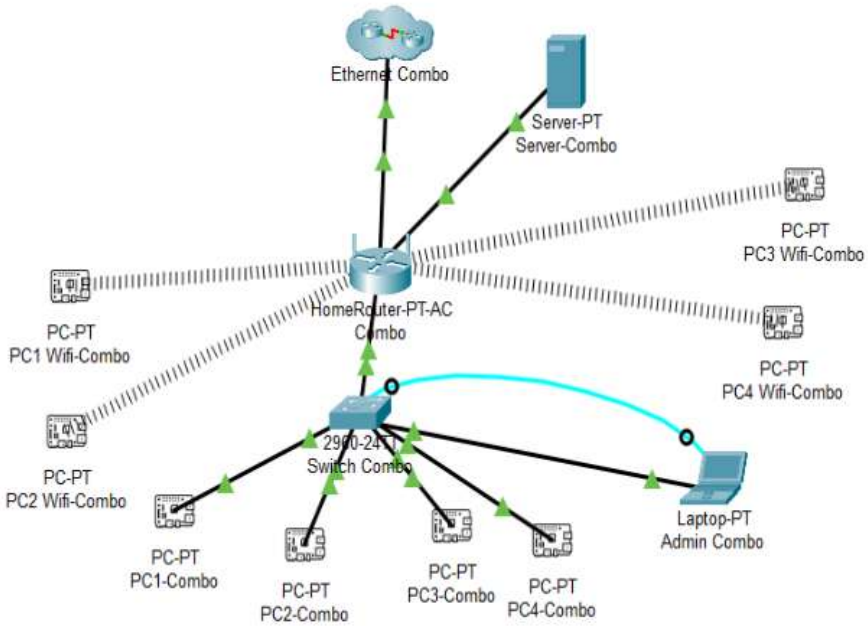


Рисунок 3 – Комбінована (гібридна) топологія системи

Для забезпечення повного моніторингу трафіку необхідна комбінація агентів:

моніторинг провідного сегмента – один з ОПК (наприклад, PC1-Combo) підключається до SPAN-порту комутатора Switch Combo;

моніторинг безпровідного сегмента – один з ОПК (наприклад, PC1 Wifi-Combo) налаштовується на режим моніторингу (Monitor Mode).

УДК 330.111.66:005.8
С 91

С 91 **Сучасні комп'ютерні системи та технології**: матеріали VIII Всеукр. наук.-практ. інтернет-конф. студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні» (24 листопада 2025 р., м. Херсон, м. Хмельницький) / за ред. А. А. Григорової. – Херсон: Книжкове видавництво ФОП Вишемирський В. С., 2025. – 229 с.

ISBN 978-617-8187-62-0 (електронне видання)
https://doi.org/10.5281/zenodo.17711825

Доповіді наукової конференції містять результати наступних досліджень: сучасні тенденції розвитку інформаційних технологій; впровадження інновацій та сучасних технологій; моделювання та оптимізація систем управління; інформаційні технології в науці, освіті, економіці, логістиці, туристичній сфері, транспорті; новітні технології в енергетичних системах та в галузі енергозбереження.

Роботи друкуються в авторській редакції, в збірці максимально зменшено втручання в обсяг та структуру відібраних до друку матеріалів. Редакційна колегія не несе відповідальність за достовірність статистичної та іншої інформації, що надано в рукописах, та залишає за собою право не розподіляти поглядів деяких авторів на ті чи інші питання.

Збірник становить інтерес для студентів, аспірантів, викладачів та наукових працівників.

УДК 330.111.66:005.8

ISBN 978–617–8187–62–0 (електронне видання)

УДК 004:37:001:62

Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024». Хмельницький. 2024. 582с.

У збірнику наукових праць подані перспективні практичні розробки аспірантів, студентів та здобувачів в області сучасних інформаційних технологій. Розглянуто актуальні проблеми комп'ютерних наук, комп'ютерної інженерії, прикладної математики й інженерії програмного забезпечення, приведено ряд робіт по впровадженню інформаційних технологій у виробництво та управління. Висвітлено перспективні розробки сучасних систем пошуку, обробки й захисту інформації, медійних та комунікаційних системи.

УДК 004:37:001:62

Матеріали конференції відтворені з авторських оригіналів, друкуються в авторській редакції та наведені в алфавітному порядку прізвищ авторів. При макетуванні можливі незначні зміни компоновки контенту авторських оригіналів. Відповідальність за якість та зміст публікацій несе автор.

Участь у конференції та складові всіх її етапів (розгляд праць, перевірка на плагіат, макетування, публікація збірника наукових праць та видача сертифікатів) є безкоштовними для всіх учасників. Оргкомітет конференції висловлює подяку учасникам конференції та сподівається на подальшу співпрацю.

З питань проведення конференції та подальшого обміну інформацією звертатись на e-mail конференції: *apkt.khnu@gmail.com*

Міністерство освіти і науки України
Херсонський національний технічний університет
Вінницький національний технічний університет
Криворізький національний університет
Кременчуцький національний університет ім. М. Остроградського
Хмельницький національний університет
Львівський національний університет
ветеринарної медицини та біотехнологій імені С.З. Ґжицького

Матеріали

VIII Всеукраїнської

науково-практичної інтернет-конференції

молодих вчених та студентів

«Сучасні інформаційні системи та технології»

за тематикою:
«Сучасні комп'ютерні системи та мережі в управлінні»

24 листопада 2025 року
Хмельницький

UDC 004.056.5

Basystyi V. A.,
2nd year master of the specialty "Cybersecurity and Information Protection" OPP "Cybersecurity and Information Protection"[1]

Cheshun D. V.,
lecturer[2]

Cheshun V. M.,
PhD, associate professor, Department of Cybersecurity[1]

MULTI-AGENT ORGANIZATION OF IOT NETWORK TRAFFIC MONITORING SYSTEM

Khmelnytskyi National University, Ukraine[1]
Khmelnytskyi Professional College of Economics and Technology UEP, Ukraine[2]

Statement of the problem

The current stage of information technology development is marked by the explosive expansion of the Internet of Things (IoT) ecosystem, which is fundamentally transforming not only the consumer sector but also core areas of the economy and critical national infrastructure. The global IoT market demonstrates exponential growth, with forecasts projecting a total volume reaching $2 trillion by 2030 [1]. The compounded annual growth rate (CAGR) is estimated to be in the range of 15.04% to 23.46%, indicating a profound and irreversible penetration of this technology into virtually all facets of societal life [2,3]. The physical scale of this phenomenon is unprecedented: the number of connected devices, which stood at an estimated 16.7 billion in 2023, is projected to surge to between 29.4 and 40 billion devices by 2030 [2,4]. This expansion means that, on average, approximately 127 new devices connect to the global network every second, thereby creating a continuously and rapidly expanding surface for potential cyberattacks [5].

While the technology was initially associated predominantly with consumer gadgets and "smart home" systems, the most significant investments and growth rates are now concentrated in mission-critical sectors. These include healthcare, where the IoT device market is projected to reach $534.3 billion by 2025, and the industrial sector (Industrial Internet of Things, IIoT), which currently accounts for approximately 60% of all new installations [2], alongside smart cities and the energy grid. Consequently, the challenge of securing the IoT has escalated from a narrow technical task to a strategic global imperative, the resolution of which is directly linked to the stability, resilience, and operational safety of fundamental public processes and critical infrastructure components.

Analysis of recent research and publications

The core problem in IoT security lies in a deep architectural incompatibility between the protection paradigms engineered for traditional corporate and cloud systems, which benefit of extensive computational resources, and the reality of the heterogeneous, geographically dispersed and resource-constrained IoT environment. IoT devices inherently possess numerous vulnerabilities. Their strictly limited computational resources (CPU speed, memory footprint) and strict low-power consumption requirements effectively preclude the utilization of conventional "heavyweight" security measures, such as full-scale antivirus software or complex, signature-based intrusion detection systems (IDS) [6,7]. This constraint is severely exacerbated at the network protocol level. Lightweight application-layer protocols, specifically Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP), are optimized for data transmission efficiency and bandwidth conservation over security. This design priority introduces multiple attack vectors, including Distributed Denial of Service (DDoS) attacks, Man-in-the-Middle (MITM) interceptions, the widespread exploitation of weak or default credentials, and message spoofing [8,9]. The scale of the threat is critical: evidence shows that unprotected IoT devices

are, on average, successfully attacked within the first five minutes of connecting to the network, and a significant portion of companies (48%) acknowledge that their current systems are incapable of effectively detecting intrusions within their IoT networks [5]. Furthermore, the lack of a standardized, energy-efficient security framework compels manufacturers to adopt minimal security settings, thus enabling the formation of massive botnets (like Mirai) which exploit compromised IoT devices as launching pads for attacks against high-value targets.

Problem statement

Attempts to adapt existing, centralized Network Intrusion Detection Systems (NIDS), such as Snort or Suricata, for deployment on resource-constrained single-board computers (SBCs) or IoT gateways have proven critically ineffective. Research has consistently demonstrated that under increasing network traffic load, these centralized systems encounter unacceptable levels of packet loss, rendering them functionally "blind" to a substantial portion of in-progress attacks [10]. This performance degradation is not merely a reflection of insufficient processor clock speed but rather a symptom of a deeper architectural mismatch.

The processing performance of NIDS is critically dependent on factors beyond CPU speed, including the memory architecture, cache efficiency, and the processor's microarchitecture. The core operation of Deep Packet Inspection (DPI), necessary for traditional signature matching, generates highly irregular memory access patterns. These patterns are inherently inefficient for the general-purpose, low-power processors that equip SBCs, leading to frequent cache misses and significant system slowdown. Consequently, the fundamental scientific challenge is identified as the dichotomy between the centralized, monolithic monitoring paradigm and the decentralized, geographically distributed nature of the protection target. An effective security solution must therefore possess an architecture that is isomorphic (structurally equivalent) to the underlying decentralized structure of the IoT ecosystem.

Presentation of the main material

The Multi-Agent System (MAS) approach, rooted in the principles of distributed artificial intelligence, offers a robust and powerful solution methodology for managing complex, dynamic, and geographically dispersed environments, such as the IoT. This methodology is founded upon the concept of an Intelligent Agent - an autonomous software entity capable of perceiving its operational environment, making independent, rational decisions, and executing actions to achieve specific security or operational goals. The MAS itself is a structured collection of these agents that interact and cooperate to collectively solve a problem that is computationally too complex or resource-intensive for any single agent [11].

The MAS paradigm is ideally suited for IoT security because its architecture is isomorphic to the network it protects. Its decentralized structure enables the distribution of computational security tasks across multiple nodes, ensuring that the processing occurs precisely at the data source (on the edge). This design philosophy simultaneously minimizes the performance load on individual resource-constrained devices and drastically reduces the latency associated with centralized data aggregation, which is crucial for real-time threat response. Furthermore, the MAS inherently incorporates resilience and fault tolerance - the compromise or temporary failure of a single agent or node does not result in a system-wide collapse of the security monitoring function, offering a critical advantage over traditional centralized Security Operations Centers (SOCs). The modularity of the agent architecture facilitates system-wide scalability and adaptability, allowing security functions to be easily expanded or updated without requiring wholesale network disruption. This approach enables the implementation of granular, micro-segmentation security policies, aligning perfectly with the modern Zero Trust security model at the device level, where constant, localized verification is mandatory for every interaction.

The proposed MAS architecture for securing the IoT environment is specifically engineered to harness distributed intelligence while strictly adhering to the resource limitations of the nodes. The system is architecturally designed as a set of cooperative agents, explicitly dividing the complex tasks of intrusion detection and incident reporting into four highly specialized roles. This division of labor is essential for minimizing the computational load on any single device and maximizing the efficiency of the security mechanism across the heterogeneous network. The functional roles and responsibilities of these four agent types are detailed in Table 1.

Table 1

Agent Type	Purpose	Input Data	Main Functions	Output Data	Key Constraints
Sensor	Traffic Collection and Primary Processing	Network Packets (raw)	Interception, Decoding, Feature Extraction	Feature Vectors	Minimal load on CPU/RAM
Analyst	Local Anomaly Detection	Feature Vectors from Sensor	Application of ML model, Traffic Classification	Local Alerts	Fast Inference Time
Coordinator	Ensuring Collective Analysis	Local Alerts from other Nodes	Event Correlation, Data Exchange, Distributed Attack Detection	Global Incidents	Communication Reliability
Reporter	Interaction with Administrator/ SIEM	Confirmed Incidents from Coordinator	Aggregation, Formatting, Secure Report Transmission	Incident Reports	Transmission Channel Security

Conclusions

The Multi-Agent System (MAS) for the security of the modern IoT is validated as the most promising solution due to its isomorphic architecture, which mirrors the decentralized nature of the IoT. The clear functional division among the Sensor, Analyst, Coordinator, and Reporter Agents enables efficient, distributed processing that adheres to strict resource limits. The localization of primary analysis minimizes computational load, while the cooperative interaction provided by the Coordinator Agent ensures the crucial ability to detect complex, distributed attacks. The MAS model thus provides the necessary foundation for achieving scalable, resilient, and effective monitoring of network traffic in resource-constrained IoT environments, ultimately contributing to the stability and safety of critical infrastructure.

References

1. IoT revenue forecast to 2030: charting growth by region, vertical and product. GsmaIntelligence. URL: https://tinyurl.com/478wjp8w (date of access: 18.11.2025).

2. Internet of Things (IoT) Market Size (2024-2030). Virtue Market Research. URL: https://virtuemarketresearch.com/report/internet-of-things-iot-market (date of access: 18.11.2025).

3. Internet Of Things (IOT) Market Size & Share Analysis - Growth Trends & Forecasts (2025-2030). Mordor Intelligence. URL: https://h7.cl/1eTYm (date of access: 18.11.2025).

4. State of IoT 2025: Number of connected IoT devices growing 14% to 21.1 billion globally. IOT ANALYTICS. URL: https://tinyurl.com/5b2jjn97 (date of access: 12.11.2025).

5. 72+ Eye-opening IoT Statistics, Facts, & Trends For 2024. Estuary. URL: https://estuary.dev/blog/iot-statistics/ (date of access: 18.11.2025).

6. Qahtani Noora. Lightweight IDS for Resource-Constrained IoT Devices Using Artificial Neural Networks. ResearchGate Publication. September, 2025. 10 p.

7. Towards Ensemble Feature Selection for Lightweight Intrusion Detection in Resource-Constrained IoT Devices. M. Fatima et al. Future Internet. 2024. №10:368. P.91-104.

8. Security of MQTT Protocol: A Brief Overview. The 4th Tunisian-Algerian Conference on applied Computing, 2024. URL: https://tinyurl.com/5t2j6bwx (date of access: 15.11.2025).

9. Attacks on the Constrained Application Protocol (CoAP). Network Working Group. URL: https://tinyurl.com/3un87kky (date of access: 18.11.2025).

10. Vadher Parag. Snort IDPS using Raspberry Pi 4. International Journal of Engineering Research & Technology (IJERT). 2020. Vol. 9, Issue 07. P.151-154.

11. ШІ-агенти та мульти-агентні системи: нові виклики для кібербезпеки. Портал QUO Vadis. URL: https://quovadis.in.ua/n/newsua/3212-ai-protect.html (дата звернення: 16.11.2025).

ДОДАТОК Б

Лістинги програмного коду

Б.1 Лістинг програмного коду Агента-Сенсора

```
import pcapy
import threading
import queue
import sys
import time


class SensorAgent(threading.Thread):
    """
    Клас, що реалізує функціональність Агента-Сенсора.
    Відповідає за низькорівневе захоплення мережевого трафіку.
    """
    def __init__(self, interface_name, packet_queue):
        threading.Thread.__init__(self)
        self.interface = interface_name
        self.packet_queue = packet_queue
        self.running = True
        self.daemon = True  # Потік завершується разом з основним процесом

    def run(self):
        print(f"[Sensor] Initializing capture on interface: {self.interface}")
        try:
            # Відкриття інтерфейсу в promiscuous mode
            # snaplen=65536 (захоплення повного пакету)
            # promisc=1 (увімкнено режим прослуховування)
```

```
            # timeout=100ms (таймаут читання для запобігання блокуванню)
            cap = pcapy.open_live(self.interface, 65536, 1, 100)

            print(f"[Sensor] Capture started. Datalink type: {cap.datalink()}")

            # Основний цикл захоплення
            while self.running:
                try:
                    # Отримання наступного пакету (header, data)
                    (header, packet) = cap.next()

                    if header:
                        # Формування метаданих пакету
                        timestamp = header.getts()
                        # Додавання в чергу для асинхронної обробки Аналітиком
                        self.packet_queue.put((timestamp, packet))

                except pcapy.PcapError as e:
                    print(f"[Sensor] Pcap Error: {e}")
                    continue

        except Exception as e:
            print(f"[Sensor] Critical Initialization Error: {e}")
            sys.exit(1)

    def stop(self):
        self.running = False
```

Б.2 Лістинг програмного коду Агента-Аналітика

```
import threading
import queue
import numpy as np
import dpkt
import math
from collections import Counter

class AnalystAgent(threading.Thread):
    """
    Реалізація Агента-Аналітика.
    Виконує парсинг, розрахунок ентропії та детектування аномалій (Z-Score).
    """
    def __init__(self, packet_queue, alert_queue, window_size=100):
        threading.Thread.__init__(self)
        self.packet_queue = packet_queue
        self.alert_queue = alert_queue
        self.window_size = window_size
        self.packet_buffer = []

        # Статистичні пороги (визначені на етапі калібрування)
        self.BASELINE_ENTROPY_MEAN = 1.25
        self.BASELINE_ENTROPY_STD = 0.45
        self.Z_SCORE_THRESHOLD = 3.0

    def calculate_entropy(self, data_list):
        """ Розрахунок ентропії Шеннона: H(X) = -sum(p(x) * log2(p(x))) """
        if not data_list: return 0.0

        counts = Counter(data_list)
        total = len(data_list)
        entropy = 0.0

        for count in counts.values():
            p = count / total
            if p > 0:
                entropy -= p * math.log2(p)
        return entropy

    def detect_anomaly(self, current_entropy):
        """ Статистичний тест Z-Score """
        z_score = (current_entropy - self.BASELINE_ENTROPY_MEAN) / self.BASELINE_ENTROPY_STD

        if abs(z_score) > self.Z_SCORE_THRESHOLD:
            # Нормалізація впевненості (confidence) від 0 до 1
            confidence = min(abs(z_score) / 5.0, 1.0)
            return True, z_score, confidence
        return False, z_score, 0.0
```

```
    def process_window(self, buffer):
        """ Обробка вікна пакетів """
        src_ips = []
        for _, raw_pkt in buffer:
            try:
                eth = dpkt.ethernet.Ethernet(raw_pkt)
                if isinstance(eth.data, dpkt.ip.IP):
                    src_ips.append(eth.data.src)
            except: continue

        # Розрахунок метрики
        entropy = self.calculate_entropy(src_ips)
        is_anomaly, z_val, conf = self.detect_anomaly(entropy)

        if is_anomaly:
            # Формування локального попередження для Координатора
            alert = {
                "type": "ENTROPY_ANOMALY",
                "metric": "src_ip_entropy",
                "value": round(entropy, 4),
                "z_score": round(z_val, 2),
                "confidence": conf
            }
            self.alert_queue.put(alert)

    def run(self):
        print("[Analyst] Statistical Engine started.")
        while True:
            pkt = self.packet_queue.get()
            self.packet_buffer.append(pkt)

            if len(self.packet_buffer) >= self.window_size:
                self.process_window(self.packet_buffer)
                self.packet_buffer = [] # Очищення вікна

            self.packet_queue.task_done()
```

Б.3 Лістинг програмного коду Агента-Координатора

```
import threading
import queue
import json
import paho.mqtt.client as mqtt

class CoordinatorAgent(threading.Thread):
    """
    Реалізація Агента-Координатора.
    Відповідає за горизонтальну комунікацію (Node-to-Node) та підтвердження інцидентів.
    """
    def __init__(self, alert_queue, incident_queue, broker_ip, node_id):
        threading.Thread.__init__(self)
        self.alert_queue = alert_queue      # Вхід: локальні алерти від Аналітика
        self.incident_queue = incident_queue # Вихід: підтверджені інциденти для Репортера
        self.node_id = node_id

        # Налаштування MQTT для координації
        self.client = mqtt.Client(client_id=f"{node_id}_coord")
        self.client.on_message = self.on_peer_message
        self.broker_ip = broker_ip
        self.peer_topic = "iot/coordination/alerts"

    def on_peer_message(self, client, userdata, msg):
        """ Обробка повідомлень від сусідніх агентів (кореляція) """
        try:
            payload = json.loads(msg.payload.decode())
            if payload['node_id'] != self.node_id:
                print(f"[Coordinator] Received peer alert from {payload['node_id']}")
                # ТУТ РЕАЛІЗУЄТЬСЯ ЛОГІКА КОРЕЛЯЦІЇ
                # Наприклад: якщо отримано схожий алерт від сусіда -> підтвердити інцидент
        except Exception as e:
            print(f"[Coordinator] Error parsing peer msg: {e}")

    def run(self):
        print("[Coordinator] Connecting to mesh network...")
        try:
            self.client.connect(self.broker_ip, 1883, 60)
            self.client.subscribe(self.peer_topic)
            self.client.loop_start()
        except Exception as e:
            print(f"[Coordinator] Connection failed: {e}")

        while True:
            # Отримання локального алерту
            local_alert = self.alert_queue.get()

            # 1. Публікація алерту для сусідів (Broadcast)
```

```
            peer_msg = {
                "node_id": self.node_id,
                "alert": local_alert
            }
            self.client.publish(self.peer_topic, json.dumps(peer_msg))

            # 2. Прийняття рішення про ескалацію (наразі: пряма ескалація)
            if local_alert['confidence'] > 0.8:
                # Передача підтвердженого інциденту Репортеру
                self.incident_queue.put(local_alert)

            self.alert_queue.task_done()
```

Б.4 Лістинг програмного коду Агента-Репортера

```
import threading
import queue
import json
import time
import paho.mqtt.client as mqtt

class ReporterAgent(threading.Thread):
    """
    Реалізація Агента-Репортера.
    Відповідає за вертикальну комунікацію (Node-to-Server) та звітність.
    """
    def __init__(self, incident_queue, broker_ip, node_id):
        threading.Thread.__init__(self)
        self.incident_queue = incident_queue # Вхід: підтверджені інциденти від Координатора
        self.node_id = node_id

        # Налаштування MQTT для звітності (SIEM)
        self.client = mqtt.Client(client_id=f"{node_id}_reporter")
        self.broker_ip = broker_ip
        self.siem_topic = "iot/security/incidents"

    def format_cef_report(self, incident):
        """
        Форматування звіту у стандарт Common Event Format (CEF) або JSON.
        """
        report = {
            "version": "1.0",
            "timestamp": time.time(),
            "source": {
                "id": self.node_id,
```

```
            "type": "L-IDS-Agent"
        },
        "event": {
            "category": incident['type'],
            "severity": "HIGH" if incident['confidence'] > 0.9 else "MEDIUM",
            "details": incident
        }
    }
    return json.dumps(report)

def run(self):
    print("[Reporter] Connecting to SIEM channel...")
    try:
        self.client.connect(self.broker_ip, 1883, 60)
        self.client.loop_start()
    except Exception as e:
        print(f"[Reporter] Connection failed: {e}")

    while True:
        # Очікування інциденту від Координатора
        incident = self.incident_queue.get()

        # Форматування
        report_payload = self.format_cef_report(incident)

        # Відправка на сервер (QoS=1 гарантує доставку)
        self.client.publish(self.siem_topic, report_payload, qos=1)
        print(f"[Reporter] >>> INCIDENT REPORTED: {incident['type']}")

        self.incident_queue.task_done()
```

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.
здобувача вищої освіти
Басистого Віталія Анатолійовича
студента ФІТ, 2 курсу, групи КБЗІм-24-1

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів академічної відповідальності, ознайомлений. Про використання спеціалізованих програмних засобів (СПЗ) StrikePlagiarism та Anti-Plagiarism для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений. Надаю університету право на передачу моєї роботи для обробки та збереження в базах даних СПЗ і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються СПЗ.

Також надаю свою згоду на обробку й збереження університетом моєї роботи в Інституційному репозитарії Хмельницького національного університету.

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

01.12.2025
дата

підпис

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 10%**

	Document		Sum coincidence on the DB	
ID: 252025 Title: Агентний метод моніторингу мережевого трафіку IOT Added in a DB: 2025-12-08 Authors: Басистий Віталій Анатолійович Heads: Чешун В.М. Consultants: Opponents:	Symbols	Lexemes	Symbols	Lexemes
	115089	1653	1400 (1%)	18 (1%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

Протокол аналізу звіту подібності науковим керівником

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

Автор: Басистий Віталій Анатолійович

Співавтор:

Назва: Агентний метод моніторингу мережевого трафіку IOT

Науковий керівник: Чешун Віктор Миколайович

Підрозділ: Кафедра кібербезпеки

Коефіцієнт подібності 1:1.2%

Коефіцієнт подібності 2:0.2%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-12-09 18:25:02.0

Після аналізу Звіту подібності констатую наступне:

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

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

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

Обгрунтування:

Дата 10.12.2025р. експерт Сергій МОСТОВИЙ

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи: Агентний метод моніторингу мережевого трафіку IOT
Автор: Басистий Віталій Анатолійович
Освітня програма: Кібербезпека та захист інформації
Рівень вищої освіти: другий (магістерський)
Спеціальність: 125 – Кібербезпека та захист інформації
Науковий керівник: ПІБ, регалії
На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту	відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою StrikePlagiarism складає 99%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism складає 98.9%

Згідно з Положенням про систему забезпечення академічної доброчесності у ХНУ (https://khmnu.edu.ua/wp-content/uploads/normatyvni-dokumenty/polozhennya/pro-systemu-zabezpechennya-akademichnoyi-dobrochesnosti.pdf, Додаток В) кваліфікаційна робота, виконана за освітньо-професійною програмою, кількісні показники рівня унікальності тексту у відсотках до загального обсягу матеріалу в якій складає 75-100 %, визнається роботою з високим рівнем унікальності тексту: «Текст вважається унікальним і не потребує додаткових дій щодо запобігання неправомірним запозиченням».

Дата: 9.12.2025

Завідувач кафедри кібербезпеки _____ Юрій КЛЬОЦ

Гарант освітньої програми _____ Віра ТІТОВА

Керівник кваліфікаційної роботи _____ Віктор ЧЕШУН

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «магістр»

Студент Басистий Віталій Анатолійович

Тема Агентний метод моніторингу мережевого трафіку IOT

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформації

Обсяг кваліфікаційної роботи освітнього ступеня «магістр»:

кількість листів креслень - ; кількість сторінок записки 89

1. Короткий зміст роботи та прийнятих рішень Кваліфікаційна робота присвячена вирішенню актуальної науково-прикладної задачі забезпечення кіберзахисту мереж Інтернету речей. Було розроблено та програмно реалізовано агентний метод моніторингу трафіку, який базується на децентралізованій архітектурі з використанням легковагого протоколу MQTT. Для виявлення аномалій було використано комбінацію аналізу ентропійних показників та ансамблевого алгоритму класифікації, щоб відмовитись від глибокої інспекції пакетів та забезпечити функціонування системи на одноплатних комп'ютерах

2. Висновок про відповідність кваліфікаційної роботи завданню Кваліфікаційна робота відповідає поставленому завданню як в теоретичній, так і в практичній частині.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі чітко сформульовано проблематику захисту пристроїв Інтернету речей, визначено предмет загроз, мету та задачі дослідження, обґрунтовано наукову новизну та практичну цінність роботи. У першому розділі проведено аналіз архітектурних вразливостей IoT та існуючих методів захисту. Наведено аргументи, щодо неефективності традиційних сигнатурних систем для пристроїв з обмеженими ресурсами та визначено вимоги для створення агента моніторингу. В другому розділі було розроблено архітектуру мультиагентної системи та обґрунтовано вибір гібридної топології розгортання, яка забезпечує контроль дротових, так і безпровідних сегментів мережі. В третьому розділі роботи описано програмну реалізацію системи моніторингу трафіку мовою програмування Python з використанням технологій контейнеризації Docker. На базі віртуалізованого кіберполігону було проведено порівняльне тестування, яке підтвердило перевагу розробленого рішення над аналогами та високу точність виявлення загроз.

4. Позитивні сторони роботи Кваліфікаційна робота має комплексну наукову і практичну цінність. Наукова цінність полягає у визначенні основних загроз для IoT та комплексного вирішення проблеми з використанням нинішніх технологій та їх порівняння з сучасними аналогами, які вимагають більше обчислювальних ресурсів та виконання однакових задач. Практична цінність полягає в розгортанні топології, з можливістю моніторингу трафіку на одноплатні комп'ютери з обмеженими обчислювальними ресурсами

5. Негативні сторони роботи В роботі при аналізі існуючих методів і рішень-прототипів недостатньо уваги приділено аналізу сучасних тенденцій реалізації і застосування мультиагентних систем.

6. Оцінка графічного оформлення та пояснювальної записки роботи Оформлення всіх матеріалів кваліфікаційної роботи є якісним, здійснене з дотриманням актуальних стандартів та інституційних положень ХНУ. Пояснювальна записка відповідає нормам щодо її оформлення як за структурою, так і за представленням і форматуванням матеріалу.

7. Відгук про роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки. Весь матеріал кваліфікаційної роботи структурований, чіткий та наскрізно пов'язаний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики кваліфікаційної роботи. Презентаційний та ілюстративний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу для досягнення поставленої мети.

8. Інші зауваження

9. Оцінка кваліфікаційної роботи Враховуючи всі позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує оцінки 96 балів.

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)
Підченко Сергій Костянтинович
завідувач кафедри ТМІТ, доктор технічних наук, професор

« 8 » 12 2025. (підпис)