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

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

на тему «Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей»

КвРКІП. 170260.23.01.20 ПЗ

Виконав: студент 2 курсу, група КІ2м-21-1 _____ Бойчук Я.А.
Підпис Ініціали, прізвище

Керівник кандидат техн. наук, доцент _____ Бобровнікова К.Ю.
Науковий ступінь, вчене звання Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КІІС, д.т.н., проф.
Т.О. Говорущенко
18 05 2023 р.

Хмельницький, 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

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

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

"01" 09 2022 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Бойчуку Ярославу Анатолійовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Керівник проекту (роботи) Бобровнікова К.Ю., к.т.н, доцент

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

Затверджена наказом ректора університету від 09.01.2023 р. № 1

2. Строк подання студентом проекту (роботи) на кафедру 01.05.2023 р.

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

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

Аналіз відомих технологій підвищення безпеки інфраструктури Інтернету речей

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Реалізація та експериментальні дослідження методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

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

6. Консультанти розділів кваліфікаційної роботи магістра

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М, професор кафедри КІС		
Антиплагіат	Нічепорук А.О, доцент кафедри КІС		

7. Дата видачі завдання «_06_»_____09_____2022р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи магістра	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики КвРМ з керівником	05.09.2022	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	05.10.2022	виконано
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	05.11.2022	виконано
4	Робота над розділом 2 – розробка моделей для вирішення поставленої задачі	05.12.2022	виконано
5	Робота над науковою статтею	05.01.2023	виконано
6	Робота над розділом 3 – розробка методів для вирішення поставленої задачі	15.02.2022	виконано
7	Робота над розділом 4 – проектування та розробка ПЗ для вирішення поставленої задачі, експериментальна частина	05.04.2023	виконано
8	Оформлення пояснювальної записки згідно вимог	15.04.2023	виконано
9	Попередній захист ДРМ	18.04.2023	виконано
10	Захист ДРМ на засіданні ЕК	До 10.05.2023	

Студент _____ Я.А.Бойчук
Підпис Ініціали, прізвище

Керівник роботи _____ К.Ю.Бобровнікова
Підпис Ініціали, прізвище

РЕФЕРАТ

Тема дипломної роботи: Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Автор роботи: Я.А.Бойчук

Керівник роботи: К.Ю.Бобровнікова

Пояснювальна записка: 94 с, 17 рис, 5 дод, 83 джерела.

ПЕРЕЛІК КЛЮЧОВИХ СЛІВ: Інтернет речей, апаратно-програмні засоби, безпека, криптографічні алгоритми

Об’єктом дослідження є процес синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Предметом дослідження є модель та метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей на основі врахування стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів Інтернету речей.

Метою роботи є синтез апаратно-програмних засобів для підвищення безпеки інфраструктури Інтернету речей.

Для розв’язання поставлених задач використовувалися методи:

1. Аналіз відомих технологій підвищення безпеки інфраструктури Інтернету речей.

2. Моделювання процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

3. Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

4. Реалізація та експериментальні дослідження методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Наукова новизна отриманих результатів:

− Удосконалено модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, яка, на відміну від відомих моделей, заснована на врахуванні стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів;

− Удосконалено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, який, на відміну від відомих, заснований на запропонованій моделі та ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей. Застосування розробленого методу надасть можливість синтезувати апаратно-програмні засоби підвищення безпеки інфраструктури Інтернету речей, в порівнянні з відомими методами.

На основі проведених досліджень розроблено та реалізовано метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Практична значимість отриманих результатів полягає в розробленні методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що надасть можливість синтезувати пристрої Інтернету речей з врахуванням стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів Інтернету речей.

ЗМІСТ

ВСТУП 5

1 АНАЛІЗ ВІДОМИХ ТЕХНОЛОГІЙ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ 8

1.1 Ключові проблеми Інтернету речей як технології 8

1.2 Стандарти безпеки в Інтернеті речей 15

1.3 Безпека на етапі проектування 19

1.4 Архітектура Інтернету речей 20

1.5 Постановка задачі 21

2 МОДЕЛЬ ПРОЦЕСУ СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ 23

2.1. Загрози безпеці в інфраструктурі Інтернету речей з врахуванням архітектури безпеки апаратно-програмних засобів Інтернету речей 23

2.1.1 Загрози безпеці на рівні сприйняття 23

2.1.2 Загрози безпеці на мережному рівні 25

2.1.3 Загрози безпеці на прикладному рівні 26

2.1.4 Загрози безпеки апаратно-програмних засобів Інтернету речей 26

2.2 Вимоги безпеки апаратно-програмних засобів Інтернету речей 27

2.3 Функції досягнення вимог безпеки апаратно-програмних засобів Інтернету речей 37

2.4 Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей 39

2.5 Висновок 41

3 МЕТОД СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ 43

3.1 Основи методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей 43

3.2 Підсистема формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей 45

3.3 Підсистема генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей 60

3.4 Підсистема формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей 62

3.5 Висновок 67

4 РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ МЕТОДУ СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ 68

4.1 Програмна реалізація методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей 68

4.2 Алгоритми програмної реалізації методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей 76

4.3 Експериментальні дослідження методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей 80

4.4 Висновок 92

ВИСНОВКИ 94

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ 96

ДОДАТОК А Фрагмент лістингу коду програмного забезпечення реалізації методу 105

ДОДАТОК Б Вхідні дані та результати експериментальних досліджень 125

ДОДАТОК В Копія публікації у виданні, що індексується в наукометричній базі Scopus..... 134

ДОДАТОК Г Довідка про прийняття публікації до друк 139

ДОДАТОК Д Презентація до дипломної роботи..... 140

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

НФВ – нефункційні вимоги

ОС – операційна система

ПЛІС – програмована логічна інтегральна схема

ФВ – функційні вимоги

ШІ – штучний інтелект

AMQP – Advanced Message Queuing Protocol

ASIC – Application-Specific Integrated Circuit

BLE – Bluetooth Low Energy

CoAP – Constrained Application Protocol

DdoS – Distributed Denial of Service

DDS – Data Distribution Service

DoS – Denial of Service

FPGA – Field-Programmable Gate Array

GE – gate equivalent

GPS – Global Positioning System

HMI – Human machine interface

IIoT – Industrial Internet of Things

IoT – Internet of Things

LTE – Long-Term Evolution

MCU – microcontroller unit

MQTT – Message Queuing Telemetry Transport

NAC – Network Access Control

RFID – Radio Frequency Identification

SBC – Session Border Controller

SQL – Structured Query Language

VPN – Virtual Private Network

XMPP – eXtensible Messaging and Presence Protocol

XSS – Cross-Site Scripting

ВСТУП

Зростаюча кількість підключених до Інтернету речей (IoT) програмно-апаратних засобів створює нові можливості для розвитку інфраструктури Інтернету речей та покращення якості життя людей. Однак, разом з цим підвищується ризик безпеки, оскільки багато з цих апаратно-програмних засобів мають вразливості, які можуть бути використані зловмисниками для кібератак. Тому дотримання вимог безпеки має стати першочерговим завданням при проектуванні пристроїв Інтернету речей. В поєднанні з використанням криптографічних методів захисту даних та мережевого зв’язку це дозволить підвищити безпеку інфраструктури Інтернету речей.

Тому метою роботи є синтез апаратно-програмних засобів для підвищення безпеки інфраструктури Інтернету речей.

Задачі дослідження можуть бути сформульовані наступним чином:

1. Провести огляд відомих стандартів та рішень синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

3. Удосконалити модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей на основі врахування стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів.

4. Розробити метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей.

5. Провести експериментальні дослідження для перевірки ефективності розробленого методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Об’єктом дослідження є процес синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Предметом дослідження є модель та метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей на основі врахування стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів Інтернету речей.

Наукова новизна одержаних результатів полягає в наступному:

1. Удосконалено модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, яка, на відміну від відомих моделей, заснована на врахуванні стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів;

2. Удосконалено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, який, на відміну від відомих, заснований на запропонованій моделі та ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей. Застосування розробленого методу дозволить синтезувати апаратно-програмні засоби підвищення безпеки інфраструктури Інтернету речей, в порівнянні з відомими методами.

Практична цінність дипломної роботи полягає в розробленні методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що надасть можливість синтезувати пристрої Інтернету речей з врахуванням стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів Інтернету речей.

За темою дипломної роботи опубліковано статтю на тему «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus [1], а також прийнято до публікації

статтю «Виявлення кібератак в інфраструктурі Інтернету речей на основі машинного навчання», «Вісник ХНУ» (Технічні науки).

1 АНАЛІЗ ВІДОМИХ ТЕХНОЛОГІЙ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ

1.1 Ключові проблеми Інтернету речей як технології

Інтернет речей (IoT) – це мережі, що складаються з різноманітних фізичних об’єктів або речей з адресами Інтернет-протоколу (IP) або іншими технологіями зв’язку, які дозволяють їм з’єднуватися один з одним, а також контролювати та/або бути контрольованими через Інтернет. В речі також вбудовані давачі, які дозволяють їм сприймати як фізичні, так і хімічні параметри з навколишнього середовища. Вхідні дані вбудованих давачів можуть надходити з різних джерел, наприклад: температура, тиск, світло, рух, концентрація іонів водню (pH), концентрація небезпечних газів тощо [2]. Ці давачі надають корисну інформацію, яка може бути передана іншим підключеним речам. Крім того, дані з двачів можна надсилати до систем управління для аналізу тенденцій, які можуть надати інформацію, необхідну для прийняття обґрунтованих рішень [3, 4]. Таким чином, річ IoT – це пристрій з вбудованими давачами та / або актуаторами, що підключається до мережі, обмінюється даними з іншими пристроями в мережі та передає дані через Інтернет [4].

По суті, Інтернет речей розширює підключення до Інтернету за межі традиційних пристроїв, таких як комп’ютери, планшети і смартфони до різноманітних повсякденних споживчих товарів, таких як холодильники та камери відеоспостереження.

Технології Інтернету речей підвищують продуктивність і ефективність праці у різноманітних галузях, а також забезпечують більш високий рівень комфорту для окремих користувачів. Сфери, де ці зміни вже стають помітними, включають моніторинг навколишнього середовища, автоматизацію промисловості та сільського господарства, охорону здоров’я, виробництво «розумного» одягу, транспорт, побутову техніку, роздрібну торгівлю, логістику та ланцюги постачання, моніторинг потоків продукції, управління запасами та безліч інших прикладів.

Кількість підключених до Інтернету пристроїв зростає прискореними темпами [5], таким чином, Інтернет речей стає центром підключення, що дозволяє обмінюватися даними про речі, людей і процеси. Google, Apple, Microsoft і Samsung, запустили власні платформи IoT, які повноцінно функціонують [2]. Крім того, Інтернет речей тісно пов'язаний із кіберфізичними системами, і, таким чином, є ключовим чинником індустрії 4.0, також відомої як четверта промислова революція [6].

Ще однією новою сферою застосування Інтернету речей є промисловий Інтернет речей (IIoT). Крім підвищення ефективності та продуктивності, переваги IIoT включають значне зниження витрат і збитків для компаній. Перспективним варіантом використання IIoT є прогнозне обслуговування [6], яке дозволить компаніям виявляти потенційні збої та уникати дорогих простоїв. Шляхом інтеграції можливостей штучного інтелекту (ШІ), таких як машинне навчання та глибоке навчання, в IIoT можна вирішити більш складні проблеми в промисловості [7], включаючи зменшення викидів вуглецю, розливи нафти в нафтогазовій промисловості тощо.

Відомо багато досліджень і прогнозів, оцінок ринку галузевими аналітиками, що грунтуються на наявних статистичних даних з різних джерел і вказують на перспективне майбутнє та вбачають значний потенціал для подальшого розвитку IoT [8]. Разом з тим, незважаючи на поточні тенденції та майбутні перспективи, Інтернет речей стикається з багатьма проблемами, які обмежують його широке впровадження.

Нові бездротові стандарти, такі як стільникова мережа п'ятого покоління (5G) і Wireless Fidelity (Wi-Fi) 6 (тобто IEEE 802.11ax), які покликані вирішити деякі проблеми, такі як покращення підключення та сприяння розгортанню пристроїв Інтернету речей, також мають свої проблеми [9, 10]. Розгортання мережі 5G, наприклад, пов'язане з додатковими проблемами, включаючи проблеми розподілу діапазонів частот, вартість розгортання та проблеми фізичного рівня [10].

Розглянемо ключові проблеми Інтернету речей як технології.

1. Неоднорідність.У концепції IoT передбачається безперебійний обмін інформацією та даними без втручання людини. Однак наразі Інтернет речей характеризується значною неоднорідністю з точки зору мереж і пристроїв, які беруть участь в обміні даними, що призводить до дуже різних можливостей з комунікаційної та обчислювальної точок зору. Різноманітні підключені пристрої та інтелектуальні додатки висувають суперечливі вимоги та викликають проблеми, які необхідно вирішити, щоб повноцінно реалізувати переваги Інтернету речей. Несумісність між фізичними рівнями Bluetooth Low Energy (BLE) і Wi-Fi є одним із прикладів, який показує, що такий високий рівень неоднорідності є серйозною проблемою в роботі екосистеми Інтернету речей [11, 12].

2. Сумісність. Щоб досягти безперебійного підключення в Інтернеті речей, пристрої повинні спілкуватися за допомогою загальних протоколів зв'язку, отже, сумісність є вирішальною вимогою. Незважаючи на це, наразі Інтернет речей є середовищем, де пристрої мають різні представлення даних, різні протоколи та різнорідні Application Programming Interfaces (API), що становить проблему для взаємодії між пристроями Інтернету речей та інтелектуальними програмами [13].

3. Масштабованість. Адаптація до змін у навколишньому середовищі, наприклад, здатність підтримувати розширення за потреби, є важливою характеристикою систем Інтернету речей. Незважаючи на те, що це бажана функція, масштабованість може створити деякі проблеми, наприклад проблеми з підключенням до широкомасштабного розгортання та розширення систем Інтернету речей. Наприклад, забезпечення висхідного підключення до дуже великої кількості підключених пристроїв може створити проблеми для стільникових мереж [14].

4. Обмежений енергетичний ресурс. Живлення периферійних пристроїв Інтернету речей, таких як давачі, приводи, приймачі Global Positioning System (GPS) і камери, становить значну проблему для багатьох розгортань Інтернету речей [15]. Це стає особливо важливим, якщо врахувати, що таких пристроїв може бути мільярди. Крім того, деякі периферійні пристрої можуть

використовуватися у дуже віддалених районах, де заміна джерел живлення може бути проблемною. Ще більше ускладнює проблему те, що деякі периферійні пристрої можуть бути вбудовані в бетонну інфраструктуру або заховані під водою, що робить заміну джерела живлення надзвичайно важкою або неможливою.

5. Безпека даних. Підключені пристрої та інтелектуальні програми для споживчих і корпоративних сфер створюють величезні обсяги даних, які постійно зростають. Оскільки Інтернет речей і дані нерозривно пов'язані між собою, а також враховуючи той факт, що багато підключених пристроїв і інтелектуальних додатків вразливі до кібератак, вони стають привабливими мішенями для зловмисників. Таким чином, ці пристрої та інтелектуальні додатки наражають організації та споживачів на нові вразливості безпеки, привнесені Інтернетом речей [16, 17].

6. Конфіденційність користувача. Багато систем Інтернету речей можуть надавати персоналізовані послуги, які можуть вимагати розуміння уподобань та інтересів користувачів, їх повсякденної діяльності та моделей поведінки. Прикладами таких «розумних» пристроїв є «розумні» переносні та імплантовані медичні пристрої, такі як фітнес-трекери та кардіостимулятори. Ці пристрої можуть відстежувати щоденну діяльність користувачів, включаючи фізичні вправи, сон і частоту серцевих скорочень, і, отже, отримувати величезні обсяги даних, які передаються через Інтернет для зберігання, обробки та аналізу на хмарних платформах Інтернету речей. На жаль, такі дані можуть прямо чи опосередковано розкривати різноманітну конфіденційну та приватну інформацію користувача, таку як ім'я, місцезнаходження, номер кредитної картки та номер соціального страхування, і таким чином піддавати користувачів різним типам атак на конфіденційність [18].

Отже, першочерговими задачами є дослідження та усунення таких ключових проблем Інтернету речей як безпека даних і конфіденційність користувачів.

Проблеми із безпекою та конфіденційністю в Інтернеті речей охоплюють різні рівні абстракції Інтернету речей [18], починаючи від нижнього рівня, також відомого як рівень сприйняття або фізичний рівень, до верхнього рівня, інакше відомого як рівень додатків. Незважаючи на те, що кібератаки існують вже дуже давно, основною проблемою Інтернету речей є відносна простота атак в IoT [19] і їх масштаб [5]. Наприклад, зловмисники можуть використовувати скомпрометовані пристрої Інтернету речей, підключені до домашньої чи корпоративної мережі, щоб здійснювати серйозні атаки на критично важливі системи чи програми. Наприклад, під час атак бот-мереж кіберзлочинці використовують мережу скомпрометованих пристроїв Інтернету речей, щоб знищити мережі, IT-інфраструктури або важливі веб-сайти шляхом здійснення атак розподіленої відмови в обслуговуванні (DDoS). Серед інших зловмисних дій, які можна здійснювати за допомогою бот-мереж, є надсилання спаму і фішингових електронних листів, використання даних онлайн-банкінгу та викрадення особистої інформації [20-22].

Питання безпеки та конфіденційності Інтернету речей є такими критичними через те, що велика кількість периферійних пристроїв, на які припадає більшість кінцевих вузлів Інтернету речей, обмежена з точки зору таких ресурсів, як енергоспоживання, пам’ять і обчислювальні можливості. Ці властиві обмеження вимагають нових підходів для підвищення безпеки інфраструктури Інтернету речей, оскільки багато традиційних методів безпеки не можуть бути безпосередньо застосовані для захисту систем Інтернету речей [23-27]. Крім перерахованих обмежень пристроїв щодо ресурсів, є й інші унікальні для Інтернету речей фактори, які спричиняють порушення безпеки та конфіденційності в Інтернеті речей.

1. Низька вартість приладів Інтернету речей. Реалізація алгоритмів безпеки в пристроях Інтернету речей є великою проблемою через вимоги до пам’яті та обчислювальних можливостей [28], що може значно збільшити вартість пристроїв. Тим не менш, більшість периферійних пристроїв Інтернету речей, таких як давачі, мають бути дешевими та одноразовими, тому виробники мають

пропонувати привабливі ціни, щоб конкурувати на ринку. У результаті виробники стикаються з проблемою досягнення компромісу між вартістю та достатніми ресурсами пристрою Інтернету речей.

2. Деякі пристрої залишаються доступними та без нагляду. У деяких додатках периферійні пристрої Інтернету речей розгортаються на великих територіях, де вони роками залишаються відкритими та без нагляду [29]. В таких випадках зловмисники можуть фізично або логічно втручатися в погано захищені пристрої.

3. Безпека за межами периметра підприємства. Хоча захист периметра традиційно використовується для захисту комп'ютерних систем, використання лише традиційного захисту периметра недостатньо для захисту мережі, що складається з пристроїв Інтернету речей. Це пояснюється тим, що підключення пристроїв Інтернету речей безпосередньо до корпоративної мережі може вивести критично важливий кіберактив за межі її периметра, оскільки вразливість в одному пристрої Інтернету речей може створити прогалину в безпеці в захисті периметра. Крім того, застарілі підходи, такі як Network Access Control (NAC), *Virtual Private Network* (VPN) і брандмауери, є вразливими, також можуть бути використані зловмисниками [30, 31].

4. Пристрої можуть не підтримуватися виробниками. Деякі пристрої Інтернету речей можуть пережити компанії, які їх виготовили [32], тому такі пристрої більше не отримуватимуть оновлення безпеки. Хоча ці пристрої все ще можуть бути підключені, вони залишаться без оновлень прошивки, і отже, вразливі до кібератак [33].

5. Складність оновлень. Деякі пристрої Інтернету речей не мають механізмів оновлення. Навіть якщо такі механізми існують і доступні оновлення, які можуть виправити виявлені недоліки безпеки, оновлення великої кількості пристроїв може бути справжньою проблемою [34]. Більше того, навіть якщо дослідники безпеки виявляють вразливості в пристроях Інтернету речей і виробники випускають виправлення, користувачі можуть не захотіти оновлювати свої «розумні» пристрої через недостатню обізнаність про безпеку. Крім того, для

користувачів інколи неможливо вручну виконати оновлення на деяких «розумних» пристроях через відсутність відповідних інтерфейсів користувача.

6. Відсутність достатнього досвіду безпеки у деяких виробників. По мірі того, як «розумні» давачі та технології зв'язку все більше впроваджуються в пристрої, а інтелектуальні програми швидко стають новою бізнес-платформою, у світі спостерігається експоненціальне зростання кількості компаній, що займаються розробкою програмного та апаратного забезпечення Інтернету речей. Хоча це можна розглядати як поштовх для інновацій в Інтернеті речей, деякі з цих компаній не надають кібербезпеці належної уваги [35]. Тому вони виробляють пристрої Інтернету речей і інтелектуальні програми з вразливостями. Однією з основних причин проблеми є те, що деякі з цих стартапів складаються з розробників, які мають незначний досвід у сфері безпеки або взагалі його не мають [36]. Інша проблема полягає в тому, що деякі компанії, які виробляли традиційні споживчі товари, такі як лампочки та тостери, раптом стали компаніями Інтернету речей. Деякі з цих компаній просто додають датчики та віджети підключення до Інтернету до своїх продуктів, не розуміючи наслідків для безпеки та небажаних наслідків цього. На додаток до відсутності досвіду безпеки іншою можливою причиною цього може бути те, що багато таких компаній невідомі, і, отже, не мають брендів чи репутації, яку потрібно захищати [37, 38]. Такий стан речей є серйозною проблемою при розробці безпечного апаратно-програмного забезпечення Інтернету речей і інтелектуальних додатків.

Необхідність впровадження вимог безпеки в проектування пристроїв Інтернету речей стала цілком очевидною через значні наслідки кібератак, пов'язаних з апаратно-програмним забезпеченням Інтернету речей. Наприклад, галузі охорони здоров'я та виробництва зазнали значних втрат через вразливості апаратно-програмного забезпечення Інтернету речей в останні роки [39], що наголошує на необхідності вирішення цієї проблеми.

Таким чином, дотримання вимог безпеки при проєктуванні апаратно-програмних засобів Інтернету речей повинно бути вирішальною концепцією.

1.2 Стандарти безпеки в Інтернеті речей

Відсутність достатнього досвіду в галузі безпеки у деяких виробників апаратно-програмних засобів Інтернету речей є одним з ключових факторів, які роблять проблеми безпеки та конфіденційності Інтернету речей унікальними.

Зі збільшенням масштабів кібератак на Інтернет речей виникає необхідність для всіх виробників апаратно-програмних засобів Інтернету речей і компаній, що займаються розробкою інтелектуальних додатків, забезпечити узгодження своїх політик безпеки і конфіденційності з правилами і керівними принципами безпеки і конфіденційності, заснованими на передових галузевих практиках та стандартах Інтернету речей [40-42]. Відповідно, різні галузі, регуляторні органи та політики розглядають можливість встановлення базового рівня безпеки та конфіденційності Інтернету речей, який забезпечить захист даних, громадську безпеку та безперервність надання послуг. Однак, з огляду на різноманітність сфер застосування Інтернету речей і різних галузей, що беруть у ньому участь, питання про те, що має включати в себе цей базовий рівень безпеки і як контролювати його виконання, все ще залишається предметом дискусій [43]. Ще однією проблемою є неоднозначність класифікації терміну «дані», який може включати в себе дані про людей, функціональні дані, колекції даних та інтелектуальну власність компаній. Крім того, в різних країнах діють різні нормативні акти та закони щодо того, що є конфіденційними даними, і як потрібно захищати дані під час їх передачі та зберігання.

На даний час стандарти безпеки та конфіденційності Інтернету речей здебільшого контролюються стандартами, розробленими закордонними державними установами та галузевими альянсами в екосистемі Інтернету речей [44]. Наприклад, ряд галузевих гігантів, таких як IBM (IBM Watson IoT), AT&T (American Telephone & Telegraph), Trusted Computing Group (TCG) і Cisco, розгортають свої різні передові практики безпеки Інтернету речей [45]. Хоча в останні роки спостерігається значний ступінь конвергенції в напрямку загальних базових специфікацій безпеки і конфіденційності Інтернету речей, серйозною

проблемою є обмежена співпраця між галузевими альянсами, академічними інститутами та державними установами, що в основному пов'язано зі значною конкуренцією на ринку в цій галузі [46, 47]. Як наслідок, існує багато різних керівних принципів і стандартів безпеки та конфіденційності Інтернету речей зі значною кількістю суперечностей і дублювань. Тому існує нагальна потреба в узагальненні існуючих керівних принципів і стандартів безпеки та конфіденційності Інтернету речей.

Разом з тим, в останні роки в усьому світі докладаються узгоджені зусилля для забезпечення безпеки та конфіденційності, а також прискорення розвитку екосистеми Інтернету речей [48]. Так, комплекс заходів Європейського Союзу, наприклад, включає в створення низки альянсів, нормативних документів, експертних центрів та пілотних проектів. Яскравим прикладом, який заслуговує на увагу, є General Data Protection Regulation (GDPR) [49] – регламент, спрямований на захист приватності та особистої інформації користувачів. Хоча цей регламент не спрямований безпосередньо на Інтернет речей, той факт, що Інтернет речей покладається на обмін, обробку і зберігання величезних обсягів даних, робить питання безпеки і конфіденційності Інтернету речей підпадаючими під дію GDPR. Зосереджуючись на нормативних актах, спрямованих на деякі сектори індустрії Інтернету речей, у листопаді 2018 року Агентство Європейського Союзу з мережевої та інформаційної безпеки (European Union Agency for Network and Information Security, ENISA) запропонувало «Передові практики безпеки Інтернету речей в контексті Індустрії 4.0 та «розумного» виробництва» [50]. Результати цього дослідження призначені для використання в країнах-членах ЄС для підвищення обізнаності про загрози та ризики Інтернету речей, а також для використання в якості довідкового посібника з безпеки та захисту в Індустрії 4.0 та «розумному виробництві». Серед важливих внесків цього дослідження: визначення відповідних термінів; надання всеосяжної таксономії кібер-активів Індустрії 4.0 та виробничого ланцюга ціноутворення; надання детальної таксономії загроз Індустрії 4.0 на основі відповідних сценаріїв атак та ризиків; картографування виявлених загроз для кіберактивів; детальний

перелік заходів безпеки, пов'язаних з Індустрією 4.0 та «розумним» виробництвом, а також їхнє зіставлення з вищезазначеними загрозами. Дослідження також окреслює заходи безпеки у трьох вимірах, а саме: політики, організаційні та технічні заходи.

Крім того, ENISA випустила два звіти: ENISA Good Practices for Security of Smart Cars [50], який визначає належні практики безпеки «розумних» автомобілів, включаючи підключені, напівавтономні та автономні транспортні засоби; та Good Practices for Security of IoT – Secure Software Development Lifecycle [51], з особливим акцентом на розробку «розумних» додатків. Перший звіт спрямований на сприяння розвитку кіберцікавості до «розумних» автомобілів і має на меті підвищити обізнаність про загрози та ризики, а також слугувати довідником з питань безпеки та захисту в цій сфері застосування. Серед основних результатів цього дослідження: надання детальної таксономії активів/загроз для «розумних» автомобілів; надання передового досвіду, який може підтвердити ландшафт кібербезпеки в сфері «розумних» автомобілів; і картографування існуючих законодавчих, стандартизаційних і політичних ініціатив. Другий звіт ґрунтується на створенні найкращих практик безпечного розвитку в екосистемі Інтернету речей. Мета і внесок цього звіту подібні до попередніх звітів, але з акцентом на розробку програмного забезпечення. З огляду на те, що безпека розробки програмного забезпечення Інтернету речей є фундаментальним будівельним блоком для безпеки і конфіденційності Інтернету речей, в звіті представлені рекомендації з безпеки, які можуть сприяти безпечній розробці продуктів і послуг Інтернету речей протягом усього терміну їх служби.

Також ISO випустила п'ять наборів стандартів, які охоплюють всі аспекти кібербезпеки. Ці стандарти мають широку сферу застосування і охоплюють безпеку та конфіденційність у кіберпросторі загалом, а отже, Інтернет речей можна вважати частиною цих стандартів.

1. Стандарт ISO/ International Electrotechnical Commission (IEC) 27001 належить до сімейства стандартів ISO/IEC 27000, покликаних допомогти організаціям забезпечити безпеку інформаційних активів [52]. ISO/IEC 27001

визначає найкращі практики для систем управління інформаційною безпекою для захисту та збереження інформації відповідно до моделі тріади CIA. Стандарт також надає набір засобів контролю, які організації можуть застосовувати на основі очікуваних ризиків.

2. Стандарт ISO/IEC 27032 визначає, як покращити стан кібербезпеки [53]. Він виокремлює унікальні аспекти діяльності та їхню залежність від інших сфер безпеки, таких як інформаційна безпека, мережева безпека та безпека в Інтернеті. Хоча засоби контролю, рекомендовані в цьому стандарті, не такі жорсткі, як у стандарті ISO/IEC 27001, цей стандарт визначає вектори атак, а також надає рекомендації щодо захисту інформації за межами організацій.

3. Стандарт ISO/IEC 27035 є стандартом для управління інцидентами [54], оскільки він визначає структурований підхід до виявлення, реагування, звітування та оцінки інцидентів інформаційної безпеки. Кіберстійкість є важливим аспектом управління інцидентами, тому цей стандарт передбачає плановий підхід до оновлення політик безпеки з метою посилення існуючих засобів контролю безпеки після аналізу події.

4. Стандарт ISO/IEC 27031 описує концепції готовності інформаційних та телекомунікаційних технологій до безперервності бізнесу та визначає критерії ефективності, проектування та впровадження для підвищення готовності інформаційних та телекомунікаційних технологій до безперервності бізнесу організації [55].

5. ISO/IEC 22301 – це стандарт для систем управління безперервністю бізнесу, який орієнтований на всі організації незалежно від типу, розміру, характеру та складності [56]. Стандарт визначає захист від атак і способи підтримання безперервності, визначаючи аспекти кіберстійкості до руйнівних інцидентів.

Хоча не існує стандартів ISO, які були б спеціально зосереджені на Інтернеті речей, Національний інститут стандартів і технологій США (National Institute of Standards and Technology, NIST) випустив стандарт NI-STIR 8228 [57] для управління ризиками кібербезпеки та конфіденційності Інтернету речей. Цей

документ покликаний допомогти федеральним агентствам та іншим організаціям краще управляти ризиками кібербезпеки та конфіденційності, пов'язаними з пристроями Інтернету речей. Хоча документ не є офіційним набором правил, він описує, як такі організації повинні управляти безпекою і конфіденційністю своїх пристроїв Інтернету речей протягом усього життєвого циклу пристроїв.

1.3 Безпека на етапі проектування

У контексті Інтернету речей безпека за задумом, або ж безпека на етапі проектування (secure by design) – це концепція, яка передбачає включення безпеки в процес проектування апаратно-програмних засобів Інтернету речей з самого початку. Це означає, що безпека є вбудованою у апаратно-програмні засоби, має бути однією з основних вимог, які розглядаються на кожному етапі проектування та розробки апаратно-програмних засобів Інтернету речей та не потребує додаткових зусиль після введення апаратно-програмних засобів в експлуатацію. Такий підхід зробить апаратно-програмні засоби Інтернету речей максимально стійкими до вразливостей і атак [58, 59].

Для забезпечення безпеки на етапі проектування апаратно-програмних засобів Інтернету речей може бути застосований комплекс заходів [60-62]:

- визначення потенційних загроз та ризиків на кожному етапі розробки апаратно-програмних засобів Інтернету речей;
- використання безпечних стандартів та протоколів для забезпечення безпеки даних та мережевого зв'язку;
- розробка безпечних методів аутентифікації та авторизації користувачів апаратно-програмних засобів Інтернету речей;
- використання криптографічних методів захисту даних та мережевого зв’язку;
- розробка системи виявлення та усунення вразливостей.

Запровадження безпеки на етапі проектування апаратно-програмних засобів Інтернету речей є більш ефективним, аніж намагання додатково захистити вже існуючу систему [63].

1.4 Архітектура Інтернету речей

Велика різноманітність різнорідних мереж і пристроїв зробила побудову загальної архітектури Інтернету речей дуже складним завданням [64]. Незважаючи на це, архітектуру Інтернету речей загалом можна розділити на три окремі рівні, а саме рівень сприйняття (також відомий як рівень розпізнавання або фізичний рівень), мережевий рівень і прикладний рівень [64, 65], як показано на рисунку 1.1. Рівень сприйняття відповідає за збір усіх типів даних із фізичного світу за допомогою фізичних кінцевих пристроїв, таких як мітки та зчитувачі радіочастотної ідентифікації (Radio frequency identification, RFID), камери, GPS-приймачі та різноманітні давачі [66]. Мережевий рівень, який знаходиться посередині, охоплює різні протоколи та технології комунікаційних мереж, які служать мережами доступу [67]. Цей рівень також відповідає за асортимент даних, початкову обробку та передачу даних [68]. Найвищим рівнем є прикладний рівень, який забезпечує підтримку бізнес-сервісів і різних видів персоналізованих послуг для окремих користувачів [64].

Прикладний рівень складається з додатків або проміжного програмного забезпечення. Протоколи прикладного рівня включають Constrained Application Protocol (CoAP), Message Queuing Telemetry Transport (MQTT), eXtensible Messaging and Presence Protocol (XMPP), Advanced Message Queuing Protocol (AMQP) і Data Distribution Service (DDS) [68, 69]. Використовуючи інтерфейс прикладного рівня, користувачі можуть отримати доступ до Інтернету речей через комп’ютери, мобільні пристрої, такі як смартфони та планшети, «розумні» холодильники, «розумні» телевізори тощо. Мережевий рівень є критично важливим, оскільки він служить сполучною ланкою між рівнем сприйняття і прикладним рівнем [70].

Зв’язок на великій відстані може бути досягнутий за допомогою Інтернету на основі IP, другого покоління (2G), третього покоління (3G), четвертого покоління (4G), Long-Term Evolution (LTE) або мережі 5G, а IEEE 802.15.4 (ZigBee), Z-Wave, Thread, Ultra-Wide Band (UWB), BLE і Near Field

Communication (NFC) використовуються для зв’язку на короткій відстані між пристроями Інтернету речей через обмежену пропускну здатність, з’єднання з втратами, переривчасті зв’язки, і обмеження потужності [70].

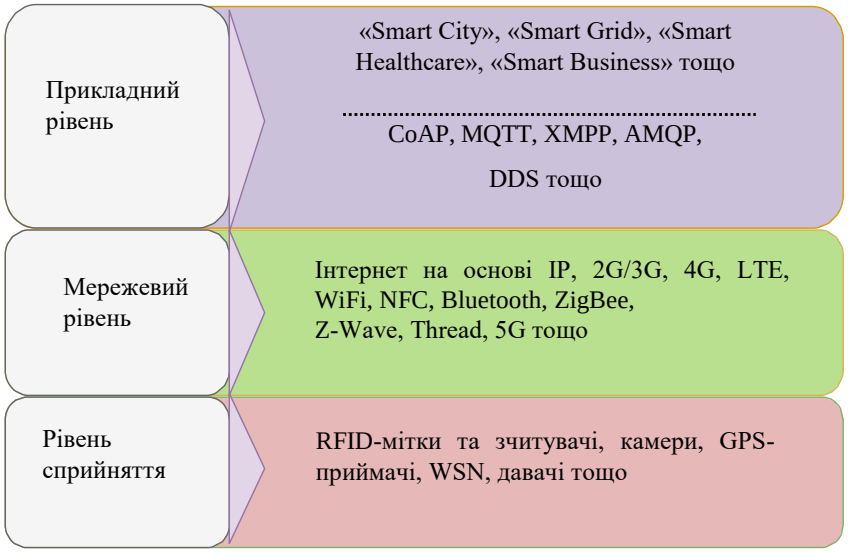


Рисунок 1.1 – Базова архітектура Інтернету речей

Необхідними або основними складовими безпеки, які вимагаються в системах і мережах Інтернету речей, є конфіденційність, цілісність даних, автентифікація, захист від відтворення та доступність [4, 61]. Але не існує універсального або єдиного механізму або рішення безпеки, яке могло б гарантувати безпеку на всіх трьох рівнях архітектури Інтернету речей. Це пов’язано з тим, що кожен рівень пред’являє унікальні вимоги до безпеки, які зазвичай відрізняються від традиційних архітектур комп’ютерних систем головним чином через обмеження енергоспоживання, обчислювальних можливостей, підключення та зберігання, властивих для багатьох пристроїв Інтернету речей.

1.5 Постановка задачі

Дослідження джерел показало, що задача синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей є надзвичайно

актуальною, оскільки запровадження безпеки на етапі проектування апаратно-програмних засобів Інтернету речей є більш ефективним, аніж намагання додатково захистити вже існуючу систему. Отже, метою роботи є синтез апаратно-програмних засобів для підвищення безпеки інфраструктури Інтернету речей.

Для досягнення мети роботи, необхідно вирішити наступні завдання:

1. Провести огляд відомих стандартів та рішень синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

3. Удосконалити модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей на основі врахування стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів.

4. Розробити метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей.

5. Провести експериментальні дослідження для перевірки ефективності розробленого методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

2 МОДЕЛЬ ПРОЦЕСУ СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ

З метою підвищення безпеки інфраструктури Інтернету речей необхідно розробити модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Модель повинна враховувати як відомі стандарти безпеки в галузі Інтернету речей та вимоги безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування, так і вимоги до синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення безпеки апаратно-програмних засобів Інтернету речей, а також особливості впровадження полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей та апаратні плтаформи, на основі яких будуються апаратно-програмні засоби Інтернету речей.

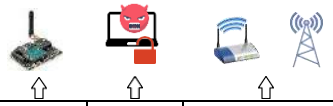
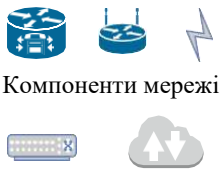
2.1. Загрози безпеці в інфраструктурі Інтернету речей з врахуванням архітектури безпеки апаратно-програмних засобів Інтернету речей

Представимо архітектуру безпеки апаратно-програмних засобів Інтернету речей з врахуванням наявності трьох рівнів Інтернету речей: Ü – рівня сприйняття, Ü – мережного рівня та Ü прикладного рівня (рис. 2.1). Подана схема відображає різні компоненти Інтернету речей на кожному з рівнів, відображення протоколів Інтернету речей на модель протоколу керування передачею (TCP)/IP, типові загрози, пов’язані з кожним рівнем, і деякі можливі механізми безпеки.

2.1.1 Загрози безпеці на рівні сприйняття

Пристрої на рівні сприйняття Ü обмежені такими ресурсами, як енергія живлення, пам'ять, здатність до обробки та часто демонструють низьку швидкість

передачі даних, оскільки вони зазвичай працюють у ненадійному бездротовому середовищі. Наприклад, сенсорні вузли зазвичай невеликі за розміром і обмежені з точки зору обчислювальної та накопичувальної ємності та покладаються на обмежені джерела енергії, такі як малоємнісні батареї. Отже, функції безпеки на деяких із цих сенсорних вузлів дуже обмежені.

<table>
<tr><td>Рівень</td><td colspan="3">Сприйняття</td><td>Мережний</td><td>Прикладний</td></tr>
<tr><td>Опис</td><td colspan="3">«Розумні» термінали рівня сприйняття</td><td>Інтернет, бездротові мережі</td><td>Інтернет або кінцевий користувач послуги</td></tr>
<tr><td rowspan="3">Компо-ненти</td><td colspan="3">  </td><td rowspan="3"> 
Компоненти мережі
</td><td>Програми та послуги
 
  </td></tr>
<tr><td>давач шлюз</td><td>RFID зчиту-вач</td><td>GSM/GPRS інфраструк-тура</td><td>Зберігання даних і аналітика
  </td></tr>
<tr><td>Сенсо-рні вузли</td><td>RFID теги</td><td>GPS приймачі</td><td></td></tr>
<tr><td>Прото-коли IoT, зіставлені з моделлю TCP/IP</td><td colspan="3">Мережний та фізичний доступ

Бездротовий зв’язок (Wi-Fi, GSM/GPRS, GPS, CDMA, 3G, 4G, LTE, 5G, BLE, NFC тощо), дротовий (Ethernet)</td><td>Транспртний

TCP, UDP

Інтернет
IPv6, 6LoWPAN, RPL та ін.</td><td>застосування
HTTP, HTTPS, MQTT, AMQP, XMPP, DSS, CoAP, тощо</td></tr>
<tr><td>Загрози</td><td colspan="3">Втручання, несанкціонований доступ до пристроїв, прослуховування, радіочастотні перешкоди, клонування тегів, спуфінг, DoS тощо</td><td>Вичерпання мережних ресурсів, спуфінг, глушіння сигналів, sinkhole, підробка, атаки MitM, перехоплення сесії, атаки DoS і DDoS тощо</td><td>DoS і DDoS атаки, впровадження зловмисного коду, перехоплення, фішинг, соціальна інженерія тощо</td></tr>
<tr><td>Можливі механізми безпеки</td><td colspan="3">Підходи до анонімності, використання хеш-алгоритмів для перевірки цілісності даних, виявлення вторгнень, легкі механізми шифрування, оцінка ризиків тощо</td><td>Механізм безпеки контролю маршрутизації, виявлення вторгнень, легкі механізми шифрування, перевірка цілісності даних за допомогою спільного закритого ключа, тощо</td><td>«Розумні» брандмауери, механізми шифрування, виявлення вторгнень, методи управління ідентифікацією для контролю доступу користувачів до ресурсу, оцінка ризиків тощо</td></tr>
</table>

Рисунок 2.1 – Трирівнева архітектура безпеки Інтернету речей

Крім того, забезпечити фізичну безпеку для таких пристроїв буде складно. Наприклад, у сфері моніторингу навколишнього середовища забезпечення

фізичної безпеки сенсорних вузлів є серйозною проблемою. У той час як деякі сенсорні вузли можуть мати базові функції безпеки, такі як імена користувачів і паролі, інші можуть не мати їх взагалі. Практично всі комунікації, такі як взаємозв'язок між пристроями в мережі моніторингу, здійснюються через радіочастоти (РЧ), що залишає її відкритою для різних видів загроз.

Отже, важко налаштувати надійний механізм захисту безпеки в мережі давачів, оскільки більшість механізмів безпеки, таких як алгоритми шифрування з відкритим ключем, як правило, потребують значних ресурсів. Пристрої на цьому рівні схильні до різних типів загроз безпеці, включаючи втручання або несанкціонований доступ до пристроїв, прослуховування, радіочастотні перешкоди, клонування тегів, підробку та атаки на відмову в обслуговуванні (DoS).

Пристрої на рівні ü схильні до різних типів загроз безпеці, включаючи втручання або несанкціонований доступ до пристроїв, прослуховування, радіочастотні перешкоди, клонування тегів, підробку та атаки на відмову в обслуговуванні (DoS).

2.1.2 Загрози безпеці на мережному рівні

Мережний рівень ü складається з низки різних комунікаційних технологій, зокрема Wi-Fi, 3G, 4G, LTE та 5G. Базовою мережею є Інтернет, який має відносно кращі можливості захисту, а протоколи маршрутизації, що використовуються на цьому рівні, подібні до протоколів стандартного Інтернету. Однак надзвичайно обмежені пристрої IoT, як-от сенсорні вузли та пристрої RFID, можуть бути схильні до різних типів атак, у тому числі підробних атак, флуду та атак типу Man-in-the-Middle (MitM). Тому безпека на цьому рівні дуже важлива. Загрози безпеці на мережному рівні включають загрози проти протоколів маршрутизації, такі як підроблена інформація про маршрутизацію; інші загрози на цьому рівні включають виснаження мережевих ресурсів, глушіння сигналів, Sinkhole, атаки вибіркової переадресації, Sybil, контрафактні атаки, атаки MitM, перехоплення сесії, DoS-атаки та флуд, наприклад, DDoS-атаки.

2.1.3 Загрози безпеці на прикладному рівні

Сегмент підтримки додатків прикладного рівня $Ü_1$ підтримує різні можливості бізнес-сервісів хмарних обчислень, такі як обробка та зберігання даних. Тому в цьому сегменті необхідний високий рівень безпеки, щоб захистити конфіденційні дані. Використовуючи сегмент додатків рівня $Ü_1$, користувачі можуть підключатися до Інтернету речей і отримувати доступ до різноманітних персоналізованих послуг відповідно до своїх прав доступу або підписки. Послуги на прикладному рівні поширюються на різні сфери, такі як моніторинг навколишнього середовища, охорона здоров'я, сільське господарство, логістика, підключені автомобілі тощо. Оскільки цей рівень передбачає інтеграцію численних бізнес-додатків у різних доменах, ключовими проблемами мають бути безпечний доступ користувачів до даних, зловживання та неправильне використання привілеїв користувача, конфіденційність користувача та проблеми автентифікації, зокрема неправильний вибір пароля.

Крім того, останніми роками людино-машинний інтерфейс (Human machine interface, HMI) широко використовується в індустрії Інтернету речей для моніторингу, дистанційного керування процесами та в ситуаціях, коли необхідне втручання людини за допомогою «розумних» пристроїв. Однак, як і більшість IT-клієнтів із HMI, додатки в межах цього рівня можуть бути вразливими до вразливостей безпеки, включаючи ін'єкцію Structured Query Language (SQL) і Cross-Site Scripting (XSS) [70] тощо. Інші загрози безпеці, пов'язані з прикладним рівнем, включають DDoS, DoS, впровадження шкідливого коду, перехоплення, фішинг і атаки соціальної інженерії.

2.1.4 Загрози безпеки апаратно-програмних засобів Інтернету речей

Таким чином, опишемо множину загроз безпеки апаратно-програмних засобів Інтернету речей кортежем:

$$T = \{T_{P}, T_{N}, T_{A}\} \tag{2.1}$$

де T_{P} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття; $T_{P} = \{t_{Pi}\}_{i=1}^{n}$, де n – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття; t_{P1} – несанкціонований доступ до апаратно-програмних засобів Інтернету речей, t_{P2} – прослуховування, t_{P3} – радіочастотні перешкоди, t_{P4} – клонування тегів, t_{P5} – спуфінг, t_{P6} – DoS, тощо;

T_{N} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на мережному рівні; $T_{N} = \{t_{Ni}\}_{i=1}^{m}$, де m – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на мережному рівні; t_{N1} – вичерпання мережних ресурсів, t_{N2} – спуфінг, t_{N3} – глушіння сигналів, t_{N4} – sinkhole, t_{N5} – підробка, t_{N6} – атаки MitM, t_{N7} – перехоплення сесії, t_{N8} – DoS і DDoS, тощо;

T_{A} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні; $T_{A} = \{t_{Ai}\}_{i=1}^{k}$, де k – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні; t_{A1} – DoS і DDoS, t_{A2} – впровадження зловмисного коду, t_{A3} – перехоплення, t_{A4} – фішинг, t_{A5} – соціальна інженерія, тощо.

2.2 Вимоги безпеки апаратно-програмних засобів Інтернету речей

На початковому етапі процесу розроблення апаратно-програмних засобів системні вимоги зазвичай поділяються на дві групи: функціональні вимоги (ФВ) та нефункціональні вимоги (НФВ). ФВ описують функціональність, яка відповідає вимогам користувача, тоді як НФВ накладають обмеження на функціональність. НФВ включають зручність використання, продуктивність, безпеку та конфіденційність. Поточна практика багатьох розробників у сфері Інтернету речей полягає в тому, щоб розглядати ФВ як першочергові вимоги в

процесі проєктування, в той час як деякі з НФВ, такі як безпека і конфіденційність, розглядаються під час впровадження або як додаткові вимоги в ситуаціях виникнення інцидентів кібербезпеки.

Важливим фактором забезпечення безпеки апаратно-програмних засобів Інтернету речей є визначення та виконання вимог безпеки як набору умов, що необхідно задовольнити для досягнення цілей безпеки апаратно-програмних засобів Інтернету речей. Хоча існують міжнародні стандарти для ідентифікації та визначення вимог безпеки для комп'ютерних систем, аналогів таких стандартів для Інтернету речей недостатньо, вони незрілі і не отримали широкого розповсюдження серед виробників апаратно-програмних засобів Інтернету речей.

Задоволення певних вимог безпеки в деяких апаратно-програмних засобах Інтернету речей є складною задачею через особливості цих пристроїв, пов'язані з обмеженими ресурсами. Разом з тим, навіть для таких апаратно-програмних засобів Інтернету речей можна визначити важливі вимоги безпеки і намагатися задовольнити принаймні ті з них, які є критично важливими для підвищення безпеки інфраструктури Інтернету речей.

Цілі інформаційної безпеки можна описати в термінах трьох фундаментальних властивостей або атрибутів безпеки: конфіденційність, цілісність і доступність, які одержали назву тріади CIA (data confidentiality, data integrity, data availability). З часом ці атрибути безпеки були розширені, включивши в себе інші атрибути безпеки, властивості або вимоги, такі як автентичність, авторизація, неспростування, підзвітність, надійність, конфіденційність користувачів, фізична безпека тощо. Разом з тим, апаратні засоби Інтернету речей піддаються різноманітним внутрішнім і зовнішнім загрозам, які можуть відрізнятися в різних галузях. Аналогічно, вимоги до безпеки можуть відрізнятися для різних програмних застосунків Інтернету речей, оскільки деякі застосунки є більш важливими та вразливими, ніж інші. Це пов'язано з тим, що в деяких сферах застосування можуть існувати більш суворі вимоги до безпеки і конфіденційності, продиктовані регулюючими органами. Крім того, оскільки різні апаратно-програмні засоби Інтернету речей, як правило,

складаються з різних компонентів і кінцевих вузлів, а також той факт, що різні організації можуть використовувати власні апаратно-програмні засоби Інтернету речей для різних цілей, а отже, матимуть різні політики безпеки, вимоги до безпеки можуть істотно відрізнятися в різних сферах застосування.

Опишемо вимоги безпеки апаратно-програмних засобів Інтернету речей з врахуванням відомих стандартів в галузі Інтернету речей кортежем:

ʹY سي ǫ̇Yؤہ ȟYؤ. ȟYؤ١ Ǿȟ (2.2)

де ʹYüρ – множина вимог безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття;

ʹYüς – множина вимог безпеки апаратно-програмних засобів Інтернету речей на мережному рівні;

ʹYüσ – множина вимог безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні.

Вимоги безпеки рівня сприйняття відрізняються залежно від програм та типу пристроїв, які використовуються для збору даних. Опишемо множину вимог безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття наступним чином:

ʹYؤہ سي ỉؤہ ἷuơ, (2.3)

де ΰ – загальна клькість вимог безпеки апаратно-програмних засобів Інтернету речей рівні сприйняття;

ỉؤہȟ – конфіденційність даних – необхідна для запобігання несанкціонованому доступу до даних, створених пристроями Інтернету речей, такими як «розумний» давач;

ỉؤہȟ – автентичність – необхідна для накладення обмежень на логічний доступ до пристроїв Інтернету речей і конфіденційної інформації;

– цілісність даних – необхідна для забезпечення правильності, точності та валідності даних, одержаних від кінцевих вузлів Інтернету речей;

– доступність – необхідна для того, щоб ресурси Інтернету речей, такі як дані в реальному часі, одержані з кінцевих вузлів або інших служб, були легкодоступними, таким чином авторизовані користувачі могли отримати до них доступ в будь-який час;

– резильєнтність – дає змогу кінцевим вузлам Інтернету речей самоналаштовуватися для обробки збоїв і уникати єдиної точки збою;

– фізична безпека – має важливе значення для захисту пристроїв Інтернету речей від несанкціонованого доступу. Наприклад, деякі сенсорні вузли можуть бути розгорнуті в середовищі, яке є відкритим для зловмисників. Такі давачі можуть залишатись без нагляду на дуже тривалий час, тому їх можна легко скомпрометувати;

– захист від несанкціонованого доступу та виявлення – необхідні для забезпечення фізичного захисту відкритих пристроїв Інтернету речей для запобігання злому криптографічних компонентів зловмисниками та виявлення будь-якої активної спроби порушити цілісність пристрою Інтернету речей або пов’язаних із ним даних;

– полювання на кіберзагрозу – важлива вимогою безпеки, яка може забезпечити проактивний пошук виявлення кіберзагроз, які можуть бути приховані в пристрої чи мережі Інтернету речей, які можуть тихо збирати облікові дані для входу чи інші конфіденційні дані [];

– надійність пристроїв на граничному рівні – важлива для гарантування узгодженої передбачуваної поведінки периферійних пристроїв Інтернету речей.

Множина вимог безпеки апаратно-програмних засобів Інтернету речей на мережевому рівні може бути представлена наступним чином:

(2.4)

де Ѽ– загальна кількість вимог безпеки апаратно-програмних засобів Інтернету речей на мережному рівні;

 – автентичність – важлива вимога, яка запобігатиме здійсненню доступу несанкціонованими особами до інформації в мережі Інтернету речей;

 – авторизація – необхідна для того, щоб зловмисники не мали неналежних привілеїв, які можуть дозволити їм наражати на небезпеку інших користувачів у мережі Інтернету речей;

 – шифрування на рівні мережі, що реалізоване над канальним рівнем даних, є вимогою, яка використовує криптографічні служби для шифрування даних під час передачі та може бути досягнута за допомогою безпеки Інтернет-протоколу (IPsec);

 – цілісність даних – гарантує, що кожна частина даних, що передається в мережі Інтернету речей, не піддається маніпуляціям на шляху передачі;

 – доступність – вимога, яка визначає потребу в тому, щоб мережа Інтернету речей та її послуги були доступними, коли це необхідно для авторизованих користувачів, пристроїв або програм, незважаючи на інші механізми, що використовуються для підтримки конфіденційності та цілісності;

 – схеми виявлення, запобігання або пом'якшення атак – необхідні для виявлення, запобігання абопом'якшення наслідків мережевих вторгнень; можуть включати захисту від DDoS і надійні оновлення системи;

 – резильєнтність мережі – гарантує, що мережа Інтернету речей продовжує працювати в умовах цілеспрямованих атак, стихійних лих або збоїв

 – виявлення аномалій – важлива вимога безпеки, яка забезпечує підхід до ідентифікації або виявлення неочікуваного порушення безпеки до того, як воно станеться; включає, наприклад, виявлення загроз безпеки в мережах Інтернету речей на основі сигнатур пакетів, які відрізняються від нормальних;

 – надійність – необхідна для того, щоб компоненти мережі Інтернету речей працювали узгоджено протягом усього життєвого циклу системи Інтернету речей.

Опишемо множину вимог безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні кортежем:

ʽYؤ١ سي ̀اؤ١ ﬀuo   (2.5)

де ʺO– загальна кількість вимог безпеки апаратно-програмних засобів Інтернету речей на мережному рівні;

ì – автентичність – гарантує, що транзакція інформації походить із джерела, з якого вона походить;

ì – авторизація – гарантує, що користувачі або пристрої мають права та привілеї на отримання доступу до ресурсу;

ì – доступність – гарантує, що пристрої Інтернету речей або інтелектуальні програми доступні та придатні для використання авторизованими користувачами або організаціями за запитом;

ì – безпечний API – необхідний для забезпечення належної автентифікації та авторизації кожного переміщення даних між «розумними» пристроями, внутрішніми системами та інтелектуальними програмами, які використовують API на основі REST, а також додавання позначки часу для запобігання атакам повторного відтворення;

ì – захист конфіденційності користувачів – означає контроль над розкриттям конфіденційної інформації користувача; це важлива вимога безпеки, яка забезпечить довіру до Інтернету речей, оскільки користувачі можуть бути впевнені, що їх конфіденційна інформація добре захищена;

ì – неспростовність – гарантує, що сторони-учасники не можуть заперечити передачу повідомлень або облікових даних між двома об’єктами Інтернету речей;

ì – відповідальність – гарантує, що кожну дію можна відстежити до конкретного одного користувача чи організації;

ì – надійність – гарантує узгоджену передбачувану поведінку системи Інтернету речей;

ìüσϖ – інформаційна криміналістика є ще однією важливою властивістю, яка гарантує, що пристрої Інтернету речей та інтелектуальні програми підтримуються існуючими методами та інструментами цифрової експертизи без шкоди для конфіденційності користувачів;

ìüσϸ π – безпечне хмарне середовище є надзвичайно важливою вимогою, беручи до уваги обсяг даних, які зберігаються та обробляються в хмарі;

ìüσϸ ρ – навчання та підвищення обізнаності користувачів щодо важливості освіти з питань безпеки є надзвичайно важливою вимогою, зокрема вибирати, зберігати та керувати своїми паролями.

Також визначимо множину вимог безпеки до апаратно-програмних засобів інфраструктури Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, ˊϒ, ˊϒ ᴺ ˊϒ, наступним чином:

ˊϒ سي ǭϒ ĩȟϒ ịȟϒ ỉȟϒ ɪȟϒ ĩ ǥϒ ȿȟϒ ⁴ȟϒ ب ȟϒ Ǫ́ (2.6)

де ˊϒ ĩ – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» будинку;

ˊϒ ị – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» мереж;

ˊϒϒϒ – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» міста;

ˊϒ ɪ – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» транспорту;

ˊϒϒΟ ǒ – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумної» охорони здоров’я;

ˊϒϒ ü – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» виробництва;

ˊϒϒ ǒ – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» ланцюгів постачання;

ʹϓγώ– множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» носимих пристроїв;

ʹϒ꞉ ̥– множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» сільського господарства.

Визначимо множину правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей:

ʹϒخ سي ̀خ ꞉ꜰuo, (2.7)

де ̃ϒ– загальна кількість правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей;

̀خ – безпечне завантаження (двійкові файли з цифровим підписом; безпечні та надійні завантажувачі; шифрування завантажувального файлу або захищені мікропроцесори; зберігання важливих даних у захищених від несанкціонованого доступу апаратних сховищах, такі як підсистема безпеки мікроконтролера TPM або модуль захищеного доступу SAM; перевірка завантажувального коду безпосередньо перед запуском; наявність на кожному етапі завантажувальної послідовності лише легітимних пристроїв; забезпечення того, щоб жоден етап у послідовності завантаження не було пропущено; функції безпечного відновлення, що перевіряють дійсність образу прошивки у випадку збою або компрометації пристрою);

̀خ. – безпечна операційна система (ОС пристрою складається лише з необхідних компонентів, таких як бібліотеки, пакети та модулі, які потрібні для функціонування пристрою; ОС можуть безпечно завантажуватися; надавання оновлень компонентів ОС протягом усього життєвого циклу; дозволи налаштовані належним чином, щоб користувачі або програми не могли записувати до кореневої файлової системи; всі непотрібні протоколи, служби та порти вимкнені);

$ì_{ióσ}$ – безпечне керування обліковими даними (кожен пристрій Інтернету речей має бути унікально ідентифікований за допомогою захищеного від несанкціонованого доступу заводського обладнання; використання належних методів управління паролями, таких як шифрування ключів, що передаються мережею, запобігання використанню порожніх паролів або паролів за замовчуванням, а також дозвіл на використання небуквено-цифрових символів разом з цифрами та літерами; надійний і безпечний механізм скидання пароля; збережені паролі хешовані; облікові дані безпеки, такі як ключі шифрування, надійно зберігаються в TPM, SAM або в надійному сховищі ключів; використання двофакторної автентифікації для доступу до конфіденційної інформації);

$ì_{iτ}$ – алгоритми шифрування (використання лише найновіших версій стандартних полегшених шифрів; не використовувати алгоритми та протоколи, які не пройшли перевірку криптографічною спільнотою; використання відповідного рівня шифрування пропорційно до чутливості даних, що обробляються; ключі шифрування надійно зберігаються в TPM, SAM або в надійному сховищі ключів; уникнення використання незахищених протоколів, таких як протокол передачі файлів (FTP) і Telnet, які покладаються на відкриті текстові імена користувачів і паролі для автентифікації; уникнення використання однакових ключів або глобальних ключів при впровадженні шифрування на пристроях Інтернету речей, кожен пристрій повинен мати власний унікальний ключ; сертифікати належним чином підтверджені для імен хостів, для яких вони призначені; уникнення використання підставних сертифікатів; зберігання лише тих конфіденційних даних, які потрібні; надійність пароля повинна бути достатньою для надійності ключів, які він захищає; використання криптографічно стійких випадкових чисел для криптографічних додатків, таких як генерація ключів тощо; передбачення можливості безпечної віддаленої заміни ключів шифрування);

$Ò_{ë_{3}}$ – безпека додатків Інтернету речей (розумні програми працюють з найнижчим рівнем привілеїв, а не від імені користувача root; надання додаткам доступу лише до необхідних ресурсів, потрібних для їхньої нормальної роботи; ізоляція інтелектуальних програм одна від одної за допомогою режиму безпечних

обчислень, який обмежує системні виклики, що здійснюються процесом, інші методи ізоляції можуть включати контейнеризацію та віртуальні машини; впровадження безпеки на кожному етапі життєвого циклу розробки смарт-додатків і документація етапів проєктування системи безпеки; використання безпечного кодування – техніки кодування, яка захищає від випадкового введення вразливостей у системі безпеки; уникнення переповнення буфера, використання лише надійних легких шифрів та безпечних протоколів, перевірка вхідних даних перед обробкою; журнали помилок та інші повідомлення не розкривають конфіденційну інформацію; заборона логінів та паролів за замовчуванням, обрання паролів, які зловмисникам буде складно вгадати; забезпечення того, щоб коментарі до коду, компілятори та інші зайві файли, які можуть дозволити зловмисникам здійснити зворотний інжиніринг коду, не були включені до складу коду; ОС та бібліотеки мають найновіші та найстабільніші версії);

 – фізична безпека в Інтернеті речей (вимкнення портів тестового доступу (TAP), таких як Joint Test Action Group (JTAG), які можуть створювати ризики для апаратної безпеки та слугувати бекдором, через який зловмисники можуть подавати фальшиві вхідні/вихідні сигнали; вимкнення всіх інтерфейсів, встановлених для тестування або адміністрування, або їх фізична недоступність; безпечні протоколи та надійні механізми контролю доступу для захисту адміністративних портів, залишених для віддаленого управління, які вважаються необхідними для нормальної роботи; захист схеми пристрою від несанкціонованого втручання за допомогою смоляної інкапсуляції, епоксидної мікросхеми на друкованій платі тощо; смарт-пристрої, які будуть розгорнуті у відкритому середовищі, надійно захищені за допомогою міцного корпусу; використання інтелектуальної власності процесора, яка включає можливості захисту від побічних каналів; вимкнення усіх функцій за замовчуванням, які не потрібні, наприклад, протокол UPnP – протокол, який дозволяє мережевим пристроям виявляти інші пристрої в мережі, що може дозволити зловмисникам відстежувати пристрої Інтернету речей).

Визначимо правила вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей ő, які повинні [72-76]: (1) забезпечувати прийнятний рівень безпеки, (2) споживати менше процесорного часу [77], (3) мати низькі вимоги до пам'яті [78], (4) споживати низьку потужність [79]; (5) використовувати малу площу ланцюга на апаратному рівні [80, 81]:

ő سي ῷ ئ ﬁUo, (2.8)

де 1– загальна кількість правил вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей;

ῷ – правила впровадження блокових шифрів;

ῷ – правила впровадження потокових шифрів;

ῷ – правила впровадження хеш-функцій;

ῷ – правила впровадження коду автентифікації повідомлень (MAC);

ῷ – правила впровадження алгоритмів автентифікованого шифрування (AE).

2.3 Функції досягнення вимог безпеки апаратно-програмних засобів Інтернету речей

Для досягнення вимог безпеки апаратно-програмних засобів Інтернету речей може бути застосована множина вдповідних функцій:

ɦ سي џ ﬁUo, (2.9)

де " – загальна кількість функцій досягнення вимог безпеки апаратно-програмних засобів Інтернету речей;

 џ طي застосування полегшених схем шифрування – це механізми захисту даних, необхідні для запобігання доступу зловмисників до даних з систем Інтернету

речей, а отже, для забезпечення конфіденційності даних, наприклад PRESENT, Piccolo, SIMON та SPECK [74];

- забезпечення контролю доступу в контексті Інтернету речей, що відноситься до механізму безпеки, який відстежує, реєструє, обмежує або дозволяє дії або операції законного користувача або пристрою в системі Інтернету речей, а також визначає обмеження для програм, що виконуються від імені законного користувача;
- функції виявлення та запобігання вторгненням на основі штучного інтелекту, такі як машинне навчання, можуть бути використані для виявлення та аналізу вхідного та вихідного мережевого трафіку Інтернету речей на предмет аномальних дій;
- застосування полегшених механізмів автентифікації для Інтернету речей та інших міжмашинних комунікацій (M2M) – це легкі схеми автентифікації, що характеризуються низьким рівнем зв'язку, а також низькими накладними витратами на обчислення і зберігання, і можуть бути використані для досягнення взаємної автентифікації та узгодження сеансового ключа;
- функції захисту від несанкціонованого доступу та виявлення несанкціонованого доступу до даних – це схеми виявлення атак несанкціонованого доступу до пристроїв та/або даних або їх модифікації, що мають на меті спричинити збої та перебої в роботі систем Інтернету речей;
- застосування ефективних та швидких алгоритмів запобігання колізіям необхідних для RFID [71] та гібридних систем, де є інтеграція сенсорних вузлів та RFID-пристроїв [70]. Алгоритми боротьби з колізіями можуть зменшити перешкоди і запобігти колізіям на зчитувачі, коли кілька міток передають дані одночасно;
- функції безпечного завантаження – це функція безпеки, яка дозволяє пристрою перевіряти програмне забезпечення за допомогою цифрового підпису або контрольних сум, включаючи операційну систему, під час завантаження, при першому ввімкненні пристрою або при його перезавантаженні. Це забезпечить

запуск лише авторизованого програмного забезпечення на пристроях Інтернету речей;

- функція безпечного оновлення – це функція безпеки, яка гарантує, що системи Інтернету речей автентифікують виправлення безпеки від операторів за допомогою цифрових підписів, щоб виправлення не могли бути перехоплені, витягнуті та змінені, тим самим запобігаючи перетворенню інтерфейсів оновлення на прогалини в безпеці;
- механізми управління ключами необхідні на канальному рівні, щоб дозволити двом віддаленим вузлам узгодити облікові дані безпеки, такі як секретні ключі;
- захист фізичної локації полягає у встановлення фізичних перешкод, таких як замки та огорожі, на шляху людей зі зловмисними намірами, а також зміцнення пристроїв на випадок аварій та екологічних катастроф.

2.4 Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Визначимо модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей кортежем:

Ü سي ǒYřY; řYخ ř̋ řŋ ŘOřö ř Ǫ (2.10)

де ˝Y– множина загроз безпеки апаратно-програмних засобів Інтернету речей;

˙Y; – множина вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей;

˙Yخ– правила синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей;

ő – правила вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей;

ɧ– множина функцій, які можуть бути застосовані для досягнення вимог безпеки апаратно-програмних засобів Інтернету речей;

Ŏ– множина апаратних платформ для побудови апаратно-програмних засобів інфраструктури Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей;

˥ – проєктований апаратно-програмний пристрій Інтернету речей.

Також визначимо функцію досягнення вимог безпеки апаратно-програмних засобів Інтернету речей (рис. 2.2):

ɧ˥ ȟYȟY꜠ ȟYخȟő ꞓ ʿY꜠ ˩ ȟYخ ˩ȟő˩ ȟO˩, (2.11)

де ʿYϕ – множина вимог безпеки до апаратно-програмного засобу Інтернету речей ˥ з врахуванням галузі застосування та відомих стандартів безпеки в галузі Інтернету речей;

ʿYί˥ – правила синтезу апаратно-програмного засобу ˥ на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей;

ő˩ – правила вибору та впровадження полегшених криптографічних програмних або апаратних алгоритмів для˥ ;

Ŏ˩ – апаратна платформа для побудови˥ з врахуванням галузі застосування.

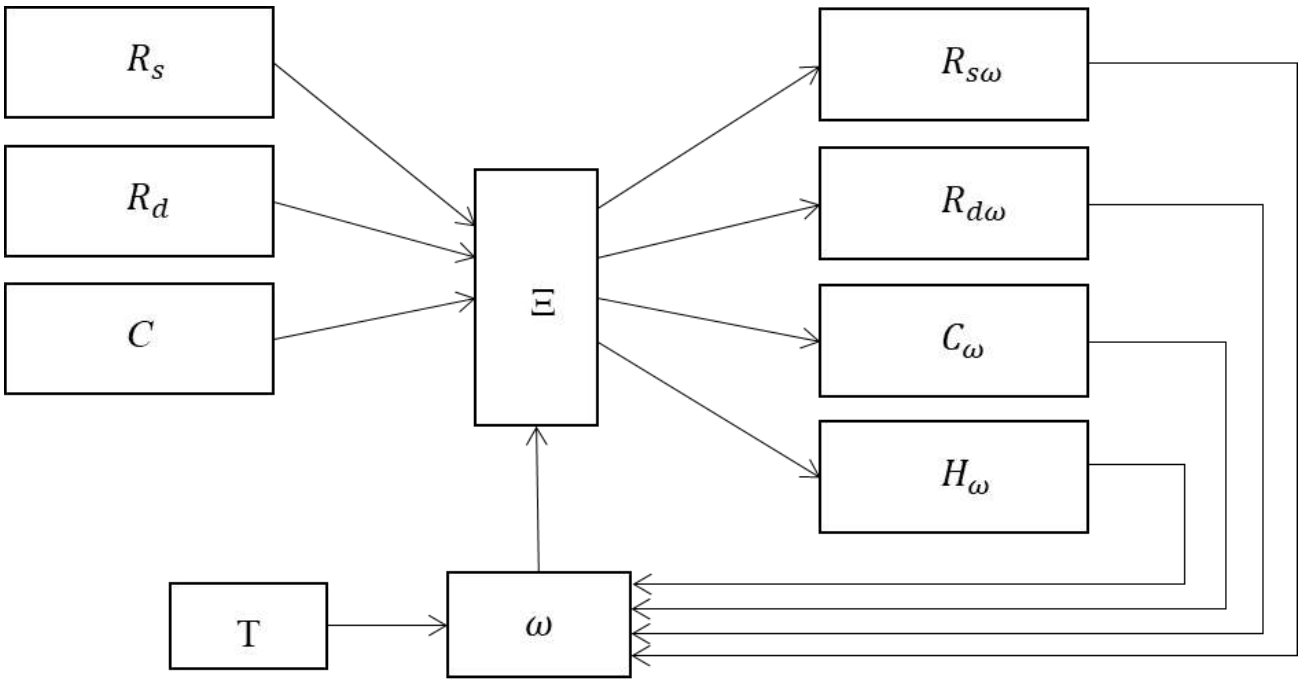


Рисунок 2.2 – Схематичне представлення функції досягнення вимог безпеки апаратно-програмних засобів Інтернету речей

2.5 Висновок

У другому розділі було представлено модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. На відміну від відомих моделей, запропонована модель враховує множину вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, а також грунтується на правилах синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей та використовує множину правил впровадження полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей. Запропонована модель також враховує апаратну плтаформу апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування.

Врахування цих складових надасть можливість дотримуватись вимог безпеки до апарано-програмних засобів Інтернету речей з врахуванням галузі їх засосування.

Запропонована модель може бути використана як основа для створення методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

3 МЕТОД СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ

3.1 Основи методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Концептуальну структуру Інтернету речей можна визначити як структуру, що складається з керівних правил, протоколів та стандартів, призначених для спрощення розробки та реалізації апаратно-програмних засобів Інтернету речей і програмних застосунків Інтернету речей. З метою підвищення безпеки інфраструктури Інтернету речей розроблено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, який грунтується на запропонованій моделі процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей та заснований на використанні експертних знань. Розроблений метод може бути корисним для надання необхідних технічних вказівок експертам, не пов’язаним з безпекою, які беруть участь у процесах розробки апаратно-програмних засобів Інтернету речей.

Укрупнена схема процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей подана на рис. 3.1.

Розроблений метод складається з п’яти етапів:

1. Формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей.

2. Формування правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки апаратно-програмних засобів Інтернету речей.

3. Формування правил вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей.

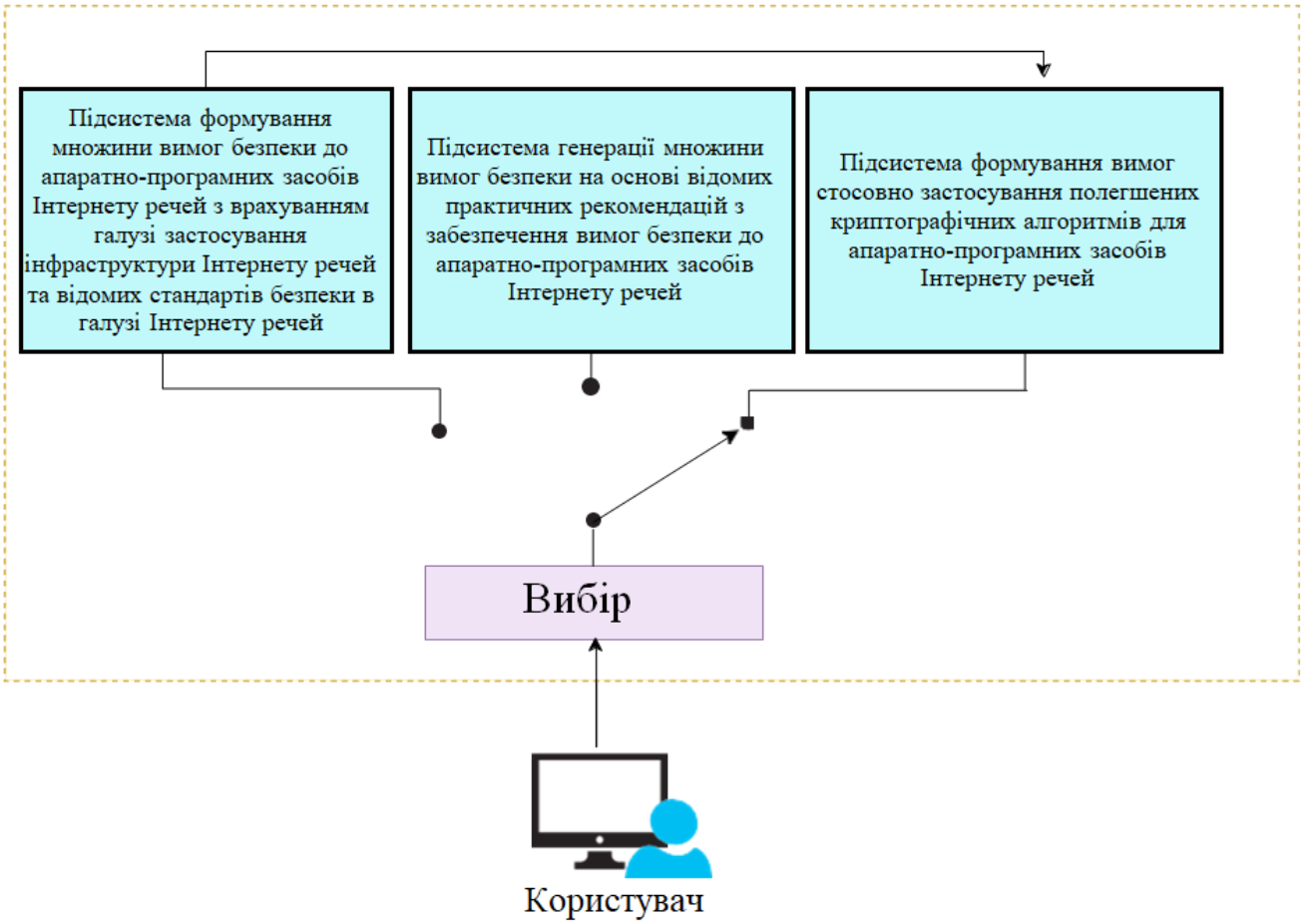


Рисунок 3.1 – Укрупнена схема процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

4. Визначення вимог до апаратно-програмного засобу Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та апаратної платформи.

5. Синтез апаратно-програмного засобу Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів.

3.2 Підсистема формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

Підсистема формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей слугує для створення множини вимог безпеки для апаратно-програмних засобів Інтернету речей на основі вхідних даних користувача та складається з п'яти компонентів: інтерфейсу користувача та генератора вхідних даних; генератора вимог безпеки; сховища даних та оновлень; інтерфейсу адміністратора та інтерфейсу виведення. Архітектура підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей представлена на рис. 3.2.

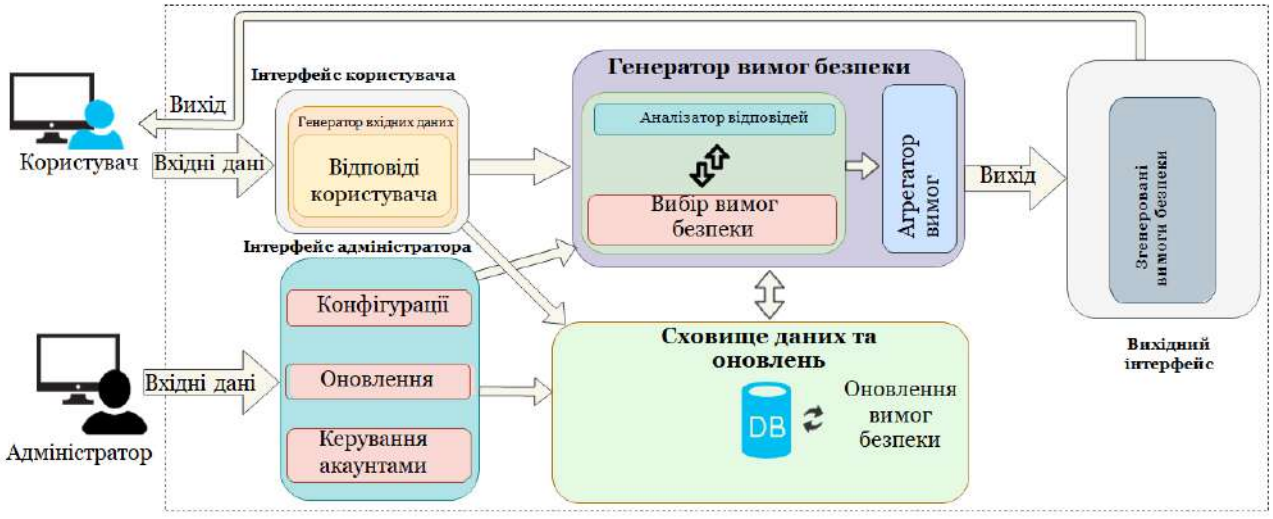


Рисунок 3.2 – Архітектура підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

З метою формування множини вимог безпеки необхідно одержати відповіді на множину питань, які характеризують апаратно-програмний засіб Інтернету речей, наприклад:

1. Чи буде у системи користувач?
2. Чи буде він мати логін користувача?
3. Чи зберігатиметься в ньому інформація про користувача?
4. Чи зберігатиме вона якусь іншу інформацію?
5. Який тип інформації буде в ньому зберігатися (звичайна, конфіденційна чи критична)?
6. Чи буде користувач надсилати дані в хмару?
7. Чи є ризик, що доступ до інформації може отримати зловмисник?

Загальна кількість запитань, які можуть бути запропоновані користувачеві, залежить від його відповіді на деякі з попередніх запитань. Наприклад, якщо користувач відповів «ні» на перше з наведених вище запитань, то друге запитання йому не буде запропоновано.

Генератор вхідних даних витягує необхідні дані з відповідей користувача і передає їх на вхід генератора вимог безпеки.

Генератор вимог безпеки складається з трьох компонентів: аналізатора відповідей користувача, компонента вибору вимог безпеки та агрегатора вимог безпеки. Аналізатор відповідей користувача оцінює вхідні дані від генератора вхідних даних і витягує конкретну інформацію з кожної відповіді, наданої користувачем, наприклад, «так», «ні», «конфіденційна інформація», «критична інформація» тощо. Ця інформація використовується компонентом визначення вимог безпеки для вибору відповідної вимоги безпеки з пулу вимог безпеки в базі даних. Агрегатор вимог безпеки збирає окремі вимоги безпеки для певного користувача, поєднує їх у набір вимог безпеки, а потім надсилає їх у вихідний інтерфейс.

Розглянемо множину вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету

речей та відомих стандартів безпеки в галузі Інтернету речей, що використовується в розробленому методі.

Визначимо множину вимог безпеки ʽYۀ ̃ які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» будинку, кортежем:

ʽYۀ ̃سي Ộì٥ ȟì. ȟì١ ȟì٢ ȟì٣ ȟì٤Ợ (3.1)

де ì٥ – інформація захищена від розголошення стороннім особам або пристроям (конфіденційність даних, конфіденційність користувачів);

ì. – дані не модифікуються та не видаляються неавторизованими особами з будь-яких зловмисних причин, і будь-яка така дія може бути відстежена до відповідальної особи (цілісність даних, облікові можливості);

ì١ – тільки авторизовані пристрої та користувачі мають право доступу до будь-якої інформації в мережі, і що користувачі не можуть обійти перевірку дозволів та отримати доступ до інформації, на яку вони не мають права доступу (автентичність, авторизація);

ì٢ – послуги апаратно-програмних засобів Інтернету речей повинні бути доступні уповноваженим особам у будь-який час, коли це необхідно (доступність, надійність);

ì٢ – стійкість з'єднання між основною мережею та периферійними пристроями повинна зберігатися навіть в умовах збоїв або інших проблем (резильєнтність мережі);

ì٣ – забезпечення автентифікації та авторизації всіх RESTful API, а також додавання мітки часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

ì٤ – кіберзлочинці не можуть мати фізичного доступу до апаратно-програмних засобів «розумного» будинку (фізична безпека).

Визначимо множину вимог безпеки ʽYۀ ̦ які повинні бути гарантованими для підвищення безпеки «розумних» мереж:

ʽYᵢ إ سي Ộ̀ᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ ȟᵢ Ớ (3.2)

де Q̀ – забезпечення захисту інформації від розголошення несанкціонованим пристроям або особам (конфіденційність даних, конфіденційність користувачів);

Q̀ – забезпечення точності, достовірності та узгодженості інформації протягом усього життєвого циклу повідомлення (цілісність даних);

Q̀ – доступ до інформації обмежується лише авторизованими користувачами та апаратно-програмними засобами Інтернету речей, і ці апаратно-програмні засоби повинні мати можливість перевіряти джерело повідомлення (автентичність);

Q̀ ي– користувачам, апаратним засобам та програмним додаткам дозволяється мати доступ лише до ресурсів, необхідних для їхньої нормальної роботи (авторизація);

Q̀ ه – кінцеві вузли Інтернету речей на рівні сприйняття здатні самостійно адаптуватися до збоїв і уникати єдиної точки відмови (резильєнтність апаратно-програмного засобу);

Q̀ .– забезпечення доступу авторизованих апаратно-програмних засобів або осіб до інформації або послуг, коли це необхідно (доступність, надійність);

Q̀ ١– забезпечення того, щоб жодна сторона не могла заперечити свою участь у передачі повідомлення, і щоб будь-яку транзакцію можна було відстежити до окремих суб'єктів, які брали в ній участь (недопущення відмови, підзвітність);

Q̀ ٢– забезпечення стійкості апаратно-програмних засобів «розумних» мереж до несанкціонованого доступу та виявлення будь-яких спроб несанкціонованого втручання (стійкість до несанкціонованого доступу та виявлення);

Q̀ ٣ – вторгнення в мережу виявляються, запобігаються або пом'якшуються (виявлення, запобігання або пом'якшення атак);

Q̀ ٤– забезпечення автентифікації та авторизації RESTful API для вимірювань «розумних» давачів, а також додавання мітки часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

Q_3 – моніторинг зв'язку апаратно-програмних засобів «розумної» мережі для виявлення аномальної комунікаційної поведінки, яка може бути спричинена шкідливим програмним забезпеченням (виявлення аномалій);

Q_4 – «розумна» мережа проактивно шукає загрози, які можуть причаїтися в мережі, не будучи виявленими (полювання на загрози);

Q_5 – забезпечення стійкості з'єднання між основною мережею та периферійними апаратно-програмними засобами навіть в умовах збоїв або інших проблем (резильєнтність мережі);

Q_6 – забезпечення фізичного захисту всіх чутливих апаратно-програмних засобів «розумної» мережі (фізична безпека).

Визначимо множину вимог безпеки Y_{SS} які повинні бути гарантованими для підвищення безпеки «розумного» міста:

$$Y_{SS} = \{Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, O_1, O_2, O_3, O_4, O_5\} \quad (3.3)$$

де Q_7 – забезпечення того, щоб жодному авторизованому користувачеві або апаратно-програмному засобу не було відмовлено в доступі до будь-якої послуги або інформації «розумного» міста (доступність, надійність);

O_1 – забезпечення того, щоб доступ до будь-яких послуг «розумного» міста був обмежений лише авторизованими користувачами та апаратно-програмними засобами, а джерело кожного повідомлення було перевірено, і щоб повідомлення не були змінені (автентичність);

O_2 – забезпечення доступу користувачів і апаратно-програмних засобів лише до тих ресурсів, до яких вони мають право доступу (авторизація);

O_3 – забезпечення того, щоб жодна інформація про «розумне» місто не була розкрита несанкціонованим користувачам і кіберзловмисникам (конфіденційність даних, конфіденційність користувачів);

Q_1 – забезпечення того, щоб жодна сторона не могла заперечити участь у будь-якій транзакції і щоб будь-яку дію можна було відстежити до залучених суб'єктів (невідмовність, підзвітність);

Q_2 – забезпечення того, щоб апаратно-програмні засоби «розумного» міста на рівні сприйняття могли самостійно адаптуватися до збоїв і уникати єдиної точки відмови (резильєнтність апаратно-програмних засобів);

Q_3 – забезпечення моніторингу зв'язку апаратно-програмних засобів «розумного» міста для виявлення аномальної комунікаційної поведінки, яка потенційно може бути спричинена зловмисним програмним забезпеченням (виявлення аномалій);

Q_4 – забезпечення того, щоб жодна частина збережених даних або даних, що передаються, не була змінена або видалена (цілісність даних);

Q_5 – забезпечення того, щоб апаратно-програмні засоби «розумного» міста могли протистояти значним спробам несанкціонованого втручання і щоб будь-які такі спроби були виявлені (стійкість до несанкціонованого доступу та виявлення);

Q_6 – забезпечення проактивного пошуку в мережі «розумного» міста загроз, які можуть причаїтися в мережі, не будучи виявленими (полювання на загрози);

Q_7 – забезпечення стійкості з'єднання між основною мережею «розумного» міста та периферійними пристроями навіть в умовах збоїв або інших проблем (резильєнтність мережі);

Q_8 – забезпечення автентифікації та авторизації всіх RESTful API, а також додавання мітки часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

Q_9 – забезпечення виявлення, запобігання або пом'якшення наслідків вторгнень в мережі «розумного» міста (виявлення, запобігання або пом'якшення наслідків атак);

Q_{10} – забезпечення фізичного захисту всіх чутливих пристроїв «розумного» міста (фізична безпека).

Визначимо множину вимог безпеки Y_1, які повинні бути гарантованими для підвищення безпеки «розумного» транспорту:

$$[illegible] \qquad (3.4)$$

де Q_1 – дані під час передачі повинні бути захищені таким чином, щоб жодні перехоплені дані не розкривали жодної частини повідомлення (конфіденційність даних, конфіденційність користувача);

[illegible] – походження кожного повідомлення має бути перевірено, а давачі на «розумних» автомобілях, «розумних» поїздах та автомобільній/залізничній інфраструктурі повинні відповідати лише на запити від уповноважених суб'єктів, щоб несанкціоновані особи не мали доступу до даних або не могли вносити їх (автентичність);

[illegible] – користувачі або апаратно-програмні засоби повинні мати доступ лише до тієї інформації або ресурсу, до якого вони мають дозвіл (авторизація);

[illegible] – жодні передані або отримані дані не можуть бути зловмисно змінені або видалені (цілісність даних);

[illegible] – «розумна» транспортна мережа проактивно шукає загрози, які можуть бути приховані в мережі (полювання на загрози);

[illegible] – апаратно-програмні засоби на фізичному рівні здатні самоналаштовуватися, щоб впоратися з несправностями і уникнути єдиної точки відмови (резильєнтність апаратно-програмних засобів);

[illegible] – API автентифіковані та авторизовані, а також додається позначка часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

[illegible] – щоб послуги «розумних» транспортних засобів та інфраструктури завжди були доступними для авторизованих суб'єктів (доступність, надійність);

[illegible] – щоб сенсорні пристрої та інфраструктура були стійкими до несанкціонованого втручання і щоб будь-які спроби несанкціонованого втручання виявлялися (стійкість до несанкціонованого доступу та виявлення);

[illegible] – стійкість з'єднання між ядром «розумної» транспортної мережі та периферійними апаратно-програмними засобами зберігається навіть в умовах збоїв або інших проблем (резильєнтність мережі);

[illegible] – щоб усі сенсорні пристрої, дорожня інфраструктура або транспортні засоби були фізично захищені від несанкціонованого доступу (фізична безпека);

[illegible] – жодна сторона не може заперечувати участь у будь-якій транзакції і будь-яка транзакція може бути відстежена до залучених суб'єктів (невідмовність, підзвітність);

[illegible] – забезпечення виявлення, запобігання або пом'якшення наслідків мережевих вторгнень (виявлення, запобігання або пом'якшення наслідків атак);

[illegible] – зв'язок «розумних» транспортних пристроїв контролюється з метою виявлення аномальної поведінки в комунікаціях, яка потенційно може бути спричинена зловмисним програмним забезпеченням (виявлення аномалій).

Визначимо множину вимог безпеки [illegible], які повинні бути гарантованими для підвищення безпеки «розумної» охорони здоров’я:

[illegible] (3.5)

де [illegible] – конфіденційність пацієнтів повинна бути захищена таким чином, щоб медичні записи або будь-які конфіденційні дані не могли бути прочитані сторонніми особами або пристроями (конфіденційність даних, конфіденційність користувачів);

[illegible] – жодні сторонні особи не повинні мати доступ до будь-яких «розумних» медичних апаратно-програмних засобів або мереж, а «розумні» медичні апаратно-програмні засоби повинні мати можливість перевіряти джерело повідомлення (автентичність);

[illegible] – уповноважений суб'єкт може мати доступ лише до інформації, необхідної для виконання своїх звичайних операцій (авторизація);

[illegible] – медичні записи або дані не модифікуються та не видаляються неавторизованими особами з будь-яких зловмисних причин (цілісність даних);

[illegible] – вторгнення в «розумну» мережу охорони здоров'я виявляються, запобігаються або пом'якшуються (виявлення, запобігання або пом'якшення наслідків атак);

ỉ٣ .– жодній уповноваженій особі, наприклад, пацієнту чи лікарю, не відмовлено в доступі до будь-якого медичного виробу або медичної послуги (доступність, надійність);

ỉ٣ ١ – «розумні» медичні апаратно-програмні засоби на периферійному рівні повинні мати можливість самоналаштовуватися, щоб впоратися зі збоями і уникнути єдиної точки відмови (резильєнтність апаратно-програмних засобів);

ỉ٣ ٢– відмовостійкість з'єднання між ядром «розумної» мережі охорони здоров'я та апаратно-програмними засобами на периферії підтримується навіть в умовах збоїв або інших проблем (резильєнтність мережі);

ỉ٣ ٣ – всі RESTful API автентифіковані та авторизовані, а також додається позначка часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

ỉ٣ ٤– «розумна» мережа охорони здоров'я проактивно шукає загрози, які можуть бути приховані в мережі (полювання на загрози);

ỉ٣ ٥ – «розумні» медичні апаратно-програмні засоби фізично недоступні для несанкціонованих користувачів (фізична безпека);

ỉ٣ ؏– моніторинг комунікацій «розумних» медичних апаратно-програмних засобів з метою виявлення аномальної поведінки в комунікаціях, яка потенційно може бути спричинена зловмисним програмним забезпеченням (виявлення аномалій);

ỉ٣ ٧ – «розумні» медичні апаратно-програмні засоби повинні мати захищені корпуси для захисту від несанкціонованого втручання та виявлення будь-яких спроб несанкціонованого доступу до таких пристроїв (стійкість до несанкціонованого доступу та виявлення);

ỉ٤ ي – будь-яке порушення конфіденційності пацієнта можна відстежити до порушників, і жодна сторона не може заперечувати свою участь у будь-якій транзакції (підзвітність, невідмовність).

Визначимо множину вимог безпеки ˈΥ٤ ؤ які повинні бути гарантованими для підвищення безпеки «розумного» виробництва:

ʹYؤ سي Ộiۥ ȟiۥ .ȟiۥ ١ȟiۥ ٢ȟiۥ ٣ȟiۥ ٤ȟiۥ ٥ȟiۥ ۶ȟiۥ ٧ȟiۥ ِȟiۏ ٍȟiۏ .ȟiۏ ١ȟiۏ Ộ (3.6)

де ỉۥ ه– лише уповноважені суб'єкти повинні мати доступ до «розумних» апаратно-програмних засобів або мереж, а «розумні» апаратно-програмні засоби повинні мати можливість перевіряти джерело повідомлення (автентичність);

ỉۥ .– авторизовані співробітники, пристрої та додатки можуть мати доступ лише до інформації, необхідної для виконання їхніх звичайних операцій (авторизація);

ỉۥ ١– «розумні» виробничі апаратно-програмні засоби на периферійному рівні, такі як давачі, повинні мати можливість самоналаштовуватися, щоб впоратися з несправностями і уникнути єдиної точки відмови (резильєнтність апаратно-програмних засобів);

ỉۥ ٢– всі API автентифіковані та авторизовані, а також додається позначка часу запиту для запобігання базовим атакам повторного відтворення (безпечний API);

ỉۥ ٣– інформація доступна лише для авторизованого персоналу та апаратно-програмних засобів (конфіденційність даних);

ỉۥ ٤– жодні дані під час транспортування або на пристрої зберігання не були зловмисно змінені або видалені (цілісність даних);

ỉۥ ٥– жодному авторизованому персоналу або апаратно-програмним засобам не відмовлено в доступі до будь-якого «розумного» обладнання або сервісу (доступність, надійність);

ỉۥ ۶ – моніторинг комунікацій «розумних» виробничих апаратно-програмних засобів з метою ідентифікації або виявлення аномальної комунікаційної поведінки, яка може бути спричинена зловмисним програмним забезпеченням (виявлення аномалій);

ỉۥ ٧– стійкість зв'язку між основною мережею «розумного» виробництва та давачами на фізичному рівні підтримується навіть в умовах збоїв або інших проблем (резильєнтність мережі);

ỉۏ ِ– жоден співробітник або апаратно-програмний засіб не може заперечувати свою участь у будь-якій транзакції, і будь-яка транзакція може бути відстежена до залученого співробітника або організації (невідмовність, підзвітність);

ὶ₀ ₀– вторгнення в «розумну» виробничу мережу виявляються, запобігаються або мінімізуються (виявлення, запобігання або пом'якшення наслідків атак);

ὶ₀ .– «розумна» виробнича мережа проактивно шукає загрози, які можуть бути приховані в мережі (полювання на загрози);

ὶ₀ ١ – «розумні» апаратно-програмні засоби повинні бути стійкими до несанкціонованого втручання, а будь-яка спроба несанкціонованого втручання в «розумні» апаратно-програмні засоби повинна виявлятися (стійкість до несанкціонованого доступу та виявлення);

ὶ₀ ٢ – несанкціонований персонал або особи не мають фізичного доступу до «розумних» апаратно-програмних засобів (фізична безпека).

Визначимо множину вимог безпеки ʽYᵢ ; які повинні бути гарантованими для підвищення безпеки «розумних» ланцюгів постачання:

ʽYᵢ ᵕ سي ôὶ₀ ȟὶ₀ ȟὶ₀ ȟὶ₀ ȟὶ₀ ȟὶ₆ ȟὶ₆ ȟὶ₆ .ȟὶ₆ ȟὶ₆ ȟὶ₆ ȟὶ₆ ȟὶ₆ ȟὶ₆ Ợ (3.7)

де ὶ₀ ٣– забезпечення того, щоб уповноваженим суб'єктам не було відмовлено в доступі до будь-якої послуги або інформації (доступність, достовірність);

ὶ₀ ٤– забезпечення того, щоб жодна інформація не була доступна стороннім особам (конфіденційність даних, конфіденційність користувачів);

ὶ₀ ٥– переконатися, що жодні дані під час транспортування або на пристрої зберігання не були зловмисно змінені або видалені будь-яким суб'єктом (цілісність даних);

ὶ₀ ٦– забезпечення того, щоб неавторизовані суб'єкти не мали доступу до будь-якої послуги, а також забезпечення того, що повідомлення надходять від заявленого відправника, і що вони не були змінені в процесі проходження (автентичність);

ὶ₀ ٧– забезпечення того, щоб суб'єкти мали доступ лише до того, до чого їм дозволено (авторизація);

ì۶ ي – забезпечення стійкості з'єднання між основною мережею «розумного» ланцюга поставок і давачами на фізичному рівні навіть в умовах несправностей або інших збоїв (резильєнтність мережі);

ì۶ ٥ – «розумні» апаратно-програмні засоби ланцюга постачання на периферійному рівні, такі як давачі, повинні мати можливість самоналаштовуватися, щоб впоратися з несправностями і уникнути єдиної точки відмови (резильєнтність апаратно-програмних засобів);

ì۶ . – забезпечення моніторингу зв'язку «розумних» апаратно-програмних засобів ланцюга постачання в режимі реального часу з метою виявлення аномальної комунікаційної поведінки, яка може бути спричинена зловмисним програмним забезпеченням (виявлення аномалій);

ì۶ ١ – вторгнення в «розумну» мережу ланцюгів постачання виявляються, запобігаються або пом'якшуються (виявлення, запобігання або пом'якшення наслідків атак);

ì۶ ٢ – забезпечення захисту «розумних» апаратно-програмних засобів ланцюга постачання від несанкціонованого втручання та виявлення будь-яких спроб несанкціонованого втручання в роботу «розумних» апаратно-програмних засобів (захист від несанкціонованого доступу та виявлення);

ì۶ ٣ – забезпечення проактивного пошуку загроз, які можуть бути приховані в «розумній» мережі ланцюгів постачання (полювання на загрози);

ì۶ ۴ – забезпечення автентифікації та авторизації всіх RESTful API, а також додавання мітки часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

ì۶ ۵ – забезпечення того, щоб жоден суб'єкт не міг заперечити участь у будь-якій транзакції, і щоб будь-яку транзакцію можна було простежити до залучених суб'єктів (невідмовність, підзвітність);

ì۶ ۶ – забезпечення фізичного захисту як зовнішнього, так і внутрішнього периметру об'єктів «розумного» ланцюга постачання (фізична безпека).

Визначимо множину вимог безпеки $Y_{безп}$ які повинні бути гарантованими для підвищення безпеки «розумних» носимих пристроїв:

$$Y_{безп} = \{з_1, з_2, з_3, з_4, з_5, з_6, з_7\} \quad (3.8)$$

де $з_1$ – інформація захищена від розголошення несанкціонованим пристроям або особам, а конфіденційність користувачів захищена таким чином, що жодна конфіденційна особиста інформація не може бути переглянута несанкціонованою особою (конфіденційність даних, конфіденційність користувачів);

$з_2$ – несанкціонований доступ до будь-яких ресурсів на «розумних» носимих пристроях або в мережах, а також перевірка повідомлень на предмет того, що вони надходять від заявленого відправника, і що вони не були змінені (автентичність);

$з_3$ – авторизованому користувачеві не відмовляють у доступі до будь-якого пристрою або послуги (доступність, надійність);

$з_4$ – суб'єкти можуть мати доступ лише до тих даних, на які вони мають дозвіл (авторизацію)

$з_5$ – точність, достовірність і узгодженість даних під час передачі підтримується протягом усього життєвого циклу повідомлення (цілісність даних);

$з_6$ – «розумні» носимі пристрої фізично недоступні для несанкціонованих користувачів (фізична безпека).

Визначимо множину вимог безпеки $Y_{безп}$ які повинні бути гарантованими для підвищення безпеки «розумного» сільського господарства:

$$Y_{безп} = \{з_1, з_2, з_3, з_4, з_5, з_6, з_7, з_8, з_9\} \quad (3.9)$$

де $з_3$ – жодні сторонні особи не повинні мати доступ до будь-яких «розумних» сільськогосподарських апаратно-програмних засобів або автоматизованих машин, а давачі сільськогосподарських апаратно-програмних засобів повинні мати можливість перевіряти походження та достовірність кожного повідомлення (автентичність);

– суб'єкти можуть мати доступ лише до інформації та ресурсів, необхідних для їхньої нормальної діяльності (авторизація);

– не відбувається розголошення інформації стороннім особам (конфіденційність даних);

– жодна частина даних під час передачі не змінюється і не видаляється (цілісність даних);

– уповноважений персонал, сільськогосподарські апаратно-програмні засоби або «розумне» обладнання можуть отримати доступ до інформації з джерела, коли це необхідно (доступність, надійність);

– сільськогосподарські апаратно-програмні засоби захищені від несанкціонованого втручання, і будь-яка спроба несанкціонованого доступу до сільськогосподарських апаратно-програмних засобів буде виявлена (стійкість до несанкціонованого доступу та виявлення);

– будь-яку транзакцію можна відстежити до залучених суб'єктів (підзвітність);

– забезпечення автентифікації та авторизації всіх RESTful API, а також додавання мітки часу запиту для запобігання базовим атакам повторного відтворення (безпечні API);

– жодні сільськогосподарські апаратно-програмні засоби не є фізично доступними до несанкціонованих користувачів (фізична безпека).

Узагальнені вимоги безпеки для інженерних галузей застосування Інтернету речей наведені в таблиці 3.1.

Таблиця 3.1 – Вимоги безпеки для різних галузей застосування Інтернету речей

	«Розумний» будинок	«Розумні» мережі	«Розумне» місто	«Розумний» транспорт	«Розумна» охорона здоров'я	«Розумне» виробництво	«Розумні» ланцюги постачання	«Розумні» носимі пристрої	«Розумне» сільське господарство
Конфіденційність даних	+	+	+	+	+	+	+	+	+
Цілісність даних	+	+	+	+	+	+	+	+	+
Доступність	+	+	+	+	+	+	+	+	+
Автентичність	+	+	+	+	+	+	+	+	+
Авторизація	+	+	+	+	+	+	+	+	+
Резильєнтність пристрою		+	+	+	+	+	+		
Резильєнтність мережі	+	+	+	+	+	+	+		
Виявлення аномалій		+	+	+	+	+	+		
Безпечний API	+	+	+	+	+	+	+		+
Конфіденційність користувачів	+	+	+	+	+		+	+	
Полювання на загрозу		+	+	+	+	+	+		

Продовження таблиці 3.1 – Вимоги безпеки для різних галузей застосування Інтернету речей

	«Розумний» будинок	«Розумні» мережі	«Розумне» місто	«Розумний» транспорт	«Розумна» охорона здоров'я	«Розумне» виробництво	«Розумні» ланцюги постачання	«Розумні» носимі пристрої	«Розумне» сільське господарство
Підзвітність	+	+	+	+	+	+	+		+
Невідмовність		+	+	+	+	+	+		
Виявлення, запобігання, пом'якшення атак		+	+	+	+	+	+		
Стійкість до несанкціонованого доступу та виявлення		+	+	+	+	+	+		+
Надійність	+	+	+	+	+	+	+	+	+
Фізична безпека	+	+	+	+	+	+	+	+	+

3.3 Підсистема генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

Підсистема генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей генерує набір рекомендацій на основі найкращих практик для безпечного розроблення апаратно-програмних засобів Інтернету речей, грунтуючись на вхідних даних, наданих користувачем.

На рисунку 3.3 наведено архітектуру підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей. Архітектура підсистеми також складається з п'яти компонентів, а саме: інтерфейса користувача, генератора звітів, сховища даних та оновлень, інтерфейса адміністратора та інтерфейса виведення.

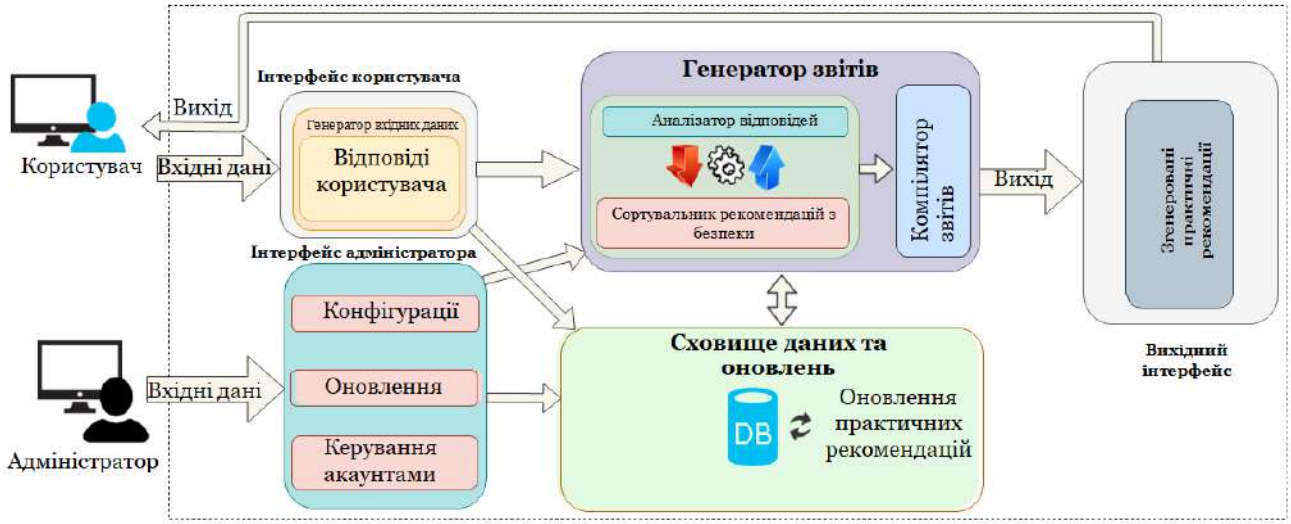


Рисунок 3.3 – Архітектура підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

Множина вимог безпеки генерується на основі опрацювання даних, витягнутих з відповідей кристувача на запитання, що характеризують апаратно-програмний засіб Інтернету речей, наприклад:

1. Чи буде в системі передбачена реєстрація користувачів?
2. Хто буде реєструвати користувачів?
3. Чи дозволить система користувачам вводити будь-які дані?
4. Чи буде система зберігати інформацію про користувача?
5. Який тип автентифікації буде реалізовано?
6. Чи буде він зберігати дані в базі даних?
7. Чи дозволить він завантажувати файли?

8. Чи буде створено лог-файл?

Генератор звітів складається з трьох компонентів: аналізатора відповідей користувача, сортувальника рекомендацій з безпеки та компілятора звітів. Аналізатор відповідей користувача відповідає за оброблення вхідних даних від генератора вхідних даних, який разом із сортувальником рекомендацій з безпеки вибирає найбільш відповідні рекомендації з колекції попередньо визначених рекомендацій з безпеки, що зберігаються в сховищі даних. Компілятор звітів упорядковує індивідуальні рекомендації з безпеки для певного користувача, компілює їх у звіт, а потім передає звіт до інтерфейсу виведення.

3.4 Підсистема формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

Підсистема формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей призначена для полегшення вибору безпечних полегшених криптографічних алгоритмів.

На рисунку 3.4 представлено архітектуру підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей. Вона складається з шести компонентів, а саме: інтерфейс користувача, фільтрація та обробка запитів, менеджер безпеки, сховище даних та оновлень, інтерфейс адміністратора та інтерфейс виведення.

Від користувача очікуються такі дані: (1) специфікація апаратного обладнання; (2) розмір корисного навантаження; (3) вимоги до енергоспоживання (лише для апаратних реалізацій); (4) вимоги до безпеки; (5) галузь застосування:

1. Специфікація апаратного обладнання. Для програмної реалізації користувачам пропонується вибрати тип апаратної платформи (Advanced Virtual RISC (AVR), Mixed-Signal Processor (MSP), Advanced RISC Machines (ARM), Programmable Interface Controllers (PIC), SBC або інші). Користувачам також пропонується вказати об'єм флеш-пам'яті, оперативної пам'яті та частоту

процесора. Для існуючих апаратно-програмних засобів Інтернету речей очікується, що користувач надасть розміри флеш-пам'яті та оперативної пам'яті для кожної вимоги безпеки. Для апаратної реалізації є лише два варіанти апаратної платформи, а саме: ASIC та FPGA. Крім того, очікується, що користувачі нададуть значення GE (gate equivalent) та пропускної здатності для кожної вимоги безпеки. Це завдання спрощується для користувача, оскільки інтерфейс відображає діапазон значень, які слугують орієнтиром для користувача.

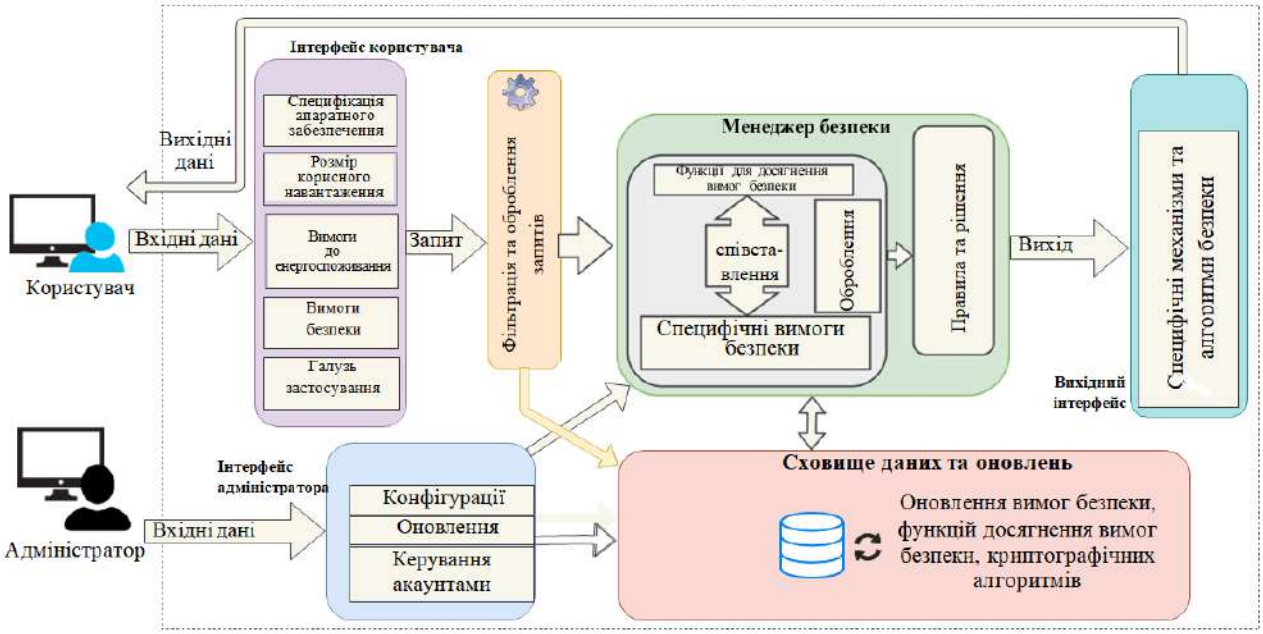


Рисунок 3.4 – Архітектура підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

2. Розмір корисного навантаження повідомлення. Користувачам пропонується вибрати розмір корисного навантаження: малий (1-128 байт), середній (129-256 байт), великий (> 256 байт), безперервний (наприклад, аудіо та відео) або невідомий.

3. Вимоги до енергоспоживання. Це стосується лише варіанту апаратної реалізації, де користувачам пропонується вибрати між варіантами з низьким та наднизьким енергоспоживанням.

4. Вимоги безпеки. Користувачі повинні вибрати один з таких варіантів вимог безпеки: (1) конфіденційність даних; (2) цілісність; (3) автентичність; (4) конфіденційність користувача; (5) невідмовність; (6) конфіденційність плюс автентичність. Користувача також запитують, чи використовував він підсистему формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, і якщо відповідь «так», то запитують, чи хоче користувач імпортувати створені ним вимоги безпеки, чи він хоче вибрати вимоги безпеки вручну. Якщо користувач хоче імпортувати свої вимоги безпеки, його попросять ввести ідентифікатор запиту, який він використовував для запиту в підсистемі формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей. Після введення ідентифікатора запиту та натискання клавіші Enter, вищезгадані вимоги безпеки, знайдені у створеному списку, будуть автоматично імпортовані. Користувачі, які не користувалися підсистемою формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, повинні вибрати вимоги безпеки вручну.

5. Галузь застосування. Користувачу надається список галузей застосування, з яких він може вибирати необхідний варіант: «розумний» будинок, «розумна» мережа, «розумне» місто, «розумний» транспорт, «розумна» охорона здоров’я, «розумне» виробництво, «розумні» ланцюги постачання, «розумні» носимі пристрої, «розумне» сільське господарство. Крім того, користувачі можуть скористатися окремою опцією, щоб вказати іншу галузь застосування, якщо її немає в списку.

Усі вхідні дані користувача надходять до модуля фільтрації та обробки запитів для перевірки та попередньої обробки.

Менеджер безпеки є центральним компонентом підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей, оскільки він приймає рішення.

Менеджер безпеки аналізує запит користувача і приймає рішення на основі набору правил (бази правил) і метрик безпеки, а саме: механізмів безпеки, вимог безпеки і знань експертів. Механізм виведення є компонентом бази знань, застосовує правила і відповідні знання в базі знань до запиту користувача та рекомендує на їх основі безпечні полегшені криптографічні алгоритми з пулу алгоритмів в базі даних. Ключові фактори, які відіграють важливу роль у процесі прийняття рішень, включають апаратні можливості, чутливість галузі застосування, тип корисного навантаження повідомлення та вимоги до енергоспоживання (у випадку апаратних запитів).

Вищезазначені фактори відіграють важливу роль у визначенні бази правил, яка є набором правил, що керують рішенням менеджера безпеки. База правил слугує представленням знань, яке базується на знаннях експертів. Кожне правило складається з операторів ЯКЩО-ТО. Приклади правил, які застосовуються до запитів на програмну реалізацію, описані нижче:

1. ЯКЩО ємність оперативної пам'яті дуже велика І флеш-пам'ять дуже велика, ТО апаратне забезпечення є дуже потужним;

2. ЯКЩО обсяг оперативної пам'яті достатній І флеш-пам'ять достатня, ТО апаратне забезпечення є потужним;

3. ЯКЩО тип корисного навантаження повідомлення малий АБО середній АБО великий І апаратне забезпечення дуже потужне АБО потужне, ТО вибрати відповідний блоковий шифр;

4. ЯКЩО тип корисного навантаження повідомлення є безперервним АБО невідомим І апаратне забезпечення дуже потужне АБО потужне, ТО вибрати відповідний потоковий шифр;

5. ЯКЩО галузь застосування є чутливою І апаратне забезпечення є дуже потужним, ТО вибрати найбільш безпечний алгоритм із великим розміром блоку та/або великим розміром ключа;

6. ЯКЩО галузь застосування є чутливою І апаратне забезпечення потужне, ТО вибрати безпечний алгоритм із помірним розміром блоку та/або помірним розміром ключа;

7. ЯКЩО апаратне забезпечення дуже обмежене, ТО вибрати дуже легкий алгоритм, який відповідає або майже відповідає цій умові;

8. ЯКЩО апаратне забезпечення обмежене, ТО вибрати легкий алгоритм, який відповідає цій умові.

База правил працює за принципом «зверху вниз», де перше правило застосовується першим.

Наприклад, перед тим, як вибрати полегшені криптографічні алгоритми для запиту на програмну реалізацію, менеджер безпеки перевіряє доступну оперативну пам'ять, а також місце на флеш-пам'яті відповідного апаратного забезпечення, застосовуючи правило 1 або правило 2. Відповідно, він рекомендує шифр з відповідним розміром блоку та/або розміром ключа, виходячи з чутливості галузі застосування (наприклад, «розумна» охорона здоров'я вважається чутливою галуззю), застосовуючи правило 3 або правило 4, звичайно, якщо апаратне забезпечення може його підтримувати. Тип корисного навантаження повідомлення використовується для визначення типу алгоритму шифрування, який буде рекомендовано (за допомогою правила 5 або правила 6), наприклад, менеджер безпеки рекомендує потоковий шифр для запиту з безперервним або невідомим розміром корисного навантаження повідомлення.

Нижче наведено правила, які застосовуються до запитів стосовно апаратної реалізації, такі:

1. ЯКЩО тип корисного навантаження повідомлення малий АБО середній АБО великий, ТО вибрати відповідний блоковий шифр;

2. ЯКЩО тип корисного навантаження повідомлення неперервний АБО невідомий, ТО вибрати відповідний потоковий шифр;

3. ЯКЩО площа ланцюга невелика, а пропускна здатність висока, ТО вибрати алгоритм, який відповідає або майже відповідає цим умовам;

4. ЯКЩО площа ланцюга не надто мала, а пропускна здатність помірна, ТО вибрати алгоритм, який відповідає цим умовам;

5. ЯКЩО галузь застосування є чутливою, ТО вибрати безпечний алгоритм з помірним розміром блоку та/або помірним розміром ключа;

6. ЯКЩО вимоги до енергоспоживання є низьке енергоспоживання, ТО вибрати енергоефективний алгоритм;

7. ЯКЩО вимоги до енергоспоживання є наднизьке енергоспоживання, ТО вибрати дуже енергоефективний алгоритм.

Як і у випадку з правилами реалізації програмного забезпечення, база правил для апаратної реалізації працює на основі стратегії «зверху вниз», коли перше правило виконується першим.

3.5 Висновок

У третьому розділі було представлено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Запропонований метод надає можливість синтезувати апаратно-програмні засоби Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів.

Врахування цих вимог надасть можливість дотримуватись вимог безпеки до апарано-програмних засобів Інтернету речей з врахуванням галузі їх засосування. Розроблений метод може бути корисним для надання необхідних технічних вказівок експертам, не пов’язаним з безпекою, які беруть участь у процесах розробки апаратно-програмних засобів Інтернету речей.

4 РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ МЕТОДУ СИНТЕЗУ АПАРАТНО-ПРОГРАМНИХ ЗАСОБІВ ПІДВИЩЕННЯ БЕЗПЕКИ ІНФРАСТРУКТУРИ ІНТЕРНЕТУ РЕЧЕЙ

4.1 Програмна реалізація методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Розроблений метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей був реалізований у вигляді системи з інтерфейсом командного рядка.

Реалізаія включає чотири діалогових консольних меню: меню входу, меню реєстрації/входу, головного меню та меню адміністратора, кожне з яких має різні опції, з яких користувач або адміністратор може обирати варіанти. Як користувачі, так і адміністратор повинні зареєструватися і пройти автентифікацію перед входом в систему. Дані їхніх облікових записів надійно зберігаються у двох різних таблицях бази даних MySQL. Зокрема, збережені паролі захищені хешем SHA256. Приклади головного меню та меню адміністратора наведені на рис. 4.1 та 4.2 відповідно.

Після реєстрації та входу в систему користувач потрапляє в головне меню (рис. 4.1), яке складається з 14 пунктів меню: пункти 1-4 – для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, пункти 5-8 – для підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей, пункти 9-12 – для підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей.

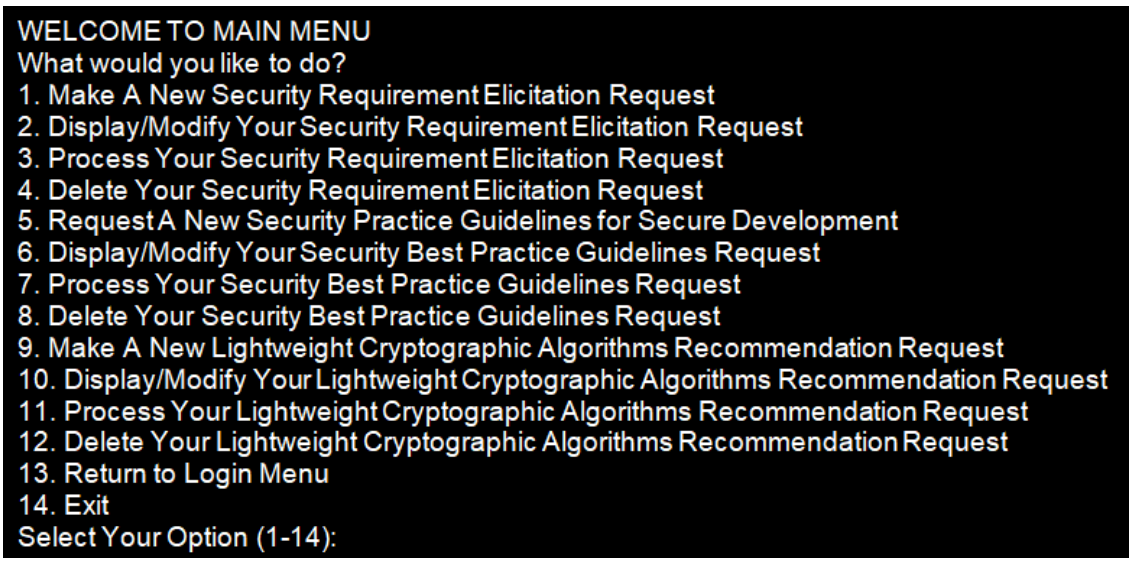


Рисунок 4.1 – Консольний інтерфейс головного меню

Вибравши пункт меню №1, користувач потрапляє у вікно консолі компонента визначення вимог безпеки до апаратно-програмного засобу Інтернету речей, щоб зробити запит. Перед тим, як зробити запит, користувач повинен вибрати унікальний ідентифікатор запиту (ID), що складається з літери R та чотирьох цифр. Вибір з чотирьох цифр має на меті зменшити ймовірність того, що два користувачі виберуть однакові ідентифікатори запитів. Це важливо, оскільки при вставці нового запиту в базу даних MySQL виникне помилка, якщо ідентифікатор запиту, вказаний у запиті на вставку, вже використовувався іншим користувачем. Використання чотирьох цифр (від 0 до 9) надає користувачеві 10 000 можливих комбінацій, з яких він може вибрати чотири цифри свого ідентифікатора запиту. Після введення унікального ідентифікатора запиту користувачеві пропонується вибрати галузь Інтернету речей, а також етап розроблення апаратно-програмного засобу (чи знаходиться апаратно-програмний засіб в стадії розроблення, чи це вже існуючий апаратно-програмний засіб).

Сховище даних та оновлень – це репозиторій, який складається з бази даних MySQL та в якому зберігаються облікові записи користувачів, а також запити користувачів до підсистем.

Сховище даних та оновлень також зберігає пул вимог безпеки для підсистеми формування множини вимог безпеки до апаратно-програмних засобів

Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей; множину рекомендацій з відомих практик безпеки для підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей та множину даних стосовно застосування полегшених криптографічних алгоритмів для формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей. Ці дані вносяться адміністратором і за потребою можуть бути ним оновлені.

Інтерфейс адміністратора складається з восьми пунктів меню, як показано на рис. 4.2.

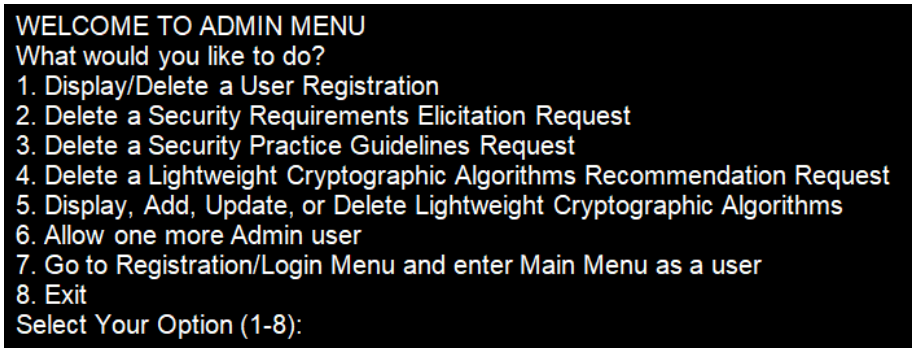


Рисунок 4.2 – Консольний інтерфейс адміністратора

Вибравши пункт меню №5 в головному меню (рис. 4.1), користувач потрапляє у вікно консолі підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей, де він може зробити запит, обравши унікальний ідентифікатор запиту, що починається з літери B, за якою слідують чотири цифри. Після введення ідентифікатора запиту користувачеві пропонується вибрати етап розроблення апаратно-програмного засобу та архітектуру та апаратну платформу, яка обрана для створення апаратно-програмного засобу Інтернету речей. Після цього користувач повинен дати відповіді на запитання, які характеризують апаратно-програмний засіб Інтернету речей.

Як і в попередньому випадку, кількість запитань, представлених користувачеві, повністю залежить від відповіді користувача на попередні запитання. Наприклад, якщо користувач відповів негативно на перше запитання, друге запитання буде пропущено. Генератор вхідних даних витягує важливу інформацію з відповідей користувача, що становить вхідні дані користувача. Вхідні дані передаються як до генератора звітів, так і до сховища даних та оновлень.

Користувач потрапляє до вікна консолі підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей, вибравши пункт №9 у головному меню (рис. 4.1).

Користувач може зробити запит, вибравши унікальний ідентифікатор запиту, що починається з S або H (залежно від того, чи запит стосується програмної або апаратної реалізації), за яким слідують чотири цифри. Користувачеві буде запропоновано вибрати етап розроблення апаратно-програмного засобу Інтернету речей (чи знаходиться він на етапі розроблення, чи це вже існуючий апаратно-програмний засіб Інтернету речей).

Від користувача очікуються такі дані: (1) специфікація апаратного обладнання; (2) розмір корисного навантаження; (3) вимоги до енергоспоживання (лише для апаратних реалізацій); (4) вимоги до безпеки; (5) галузь застосування.

Вхідні дані користувача надходять до модуля фільтрації та обробки запитів для перевірки та попередньої обробки. На рисунках 4.3 та 4.4 представлено блок-схеми алгоритмів оброблення запитів стосовно програмної та апаратної реалізації криптографічних алгоритмів відповідно.

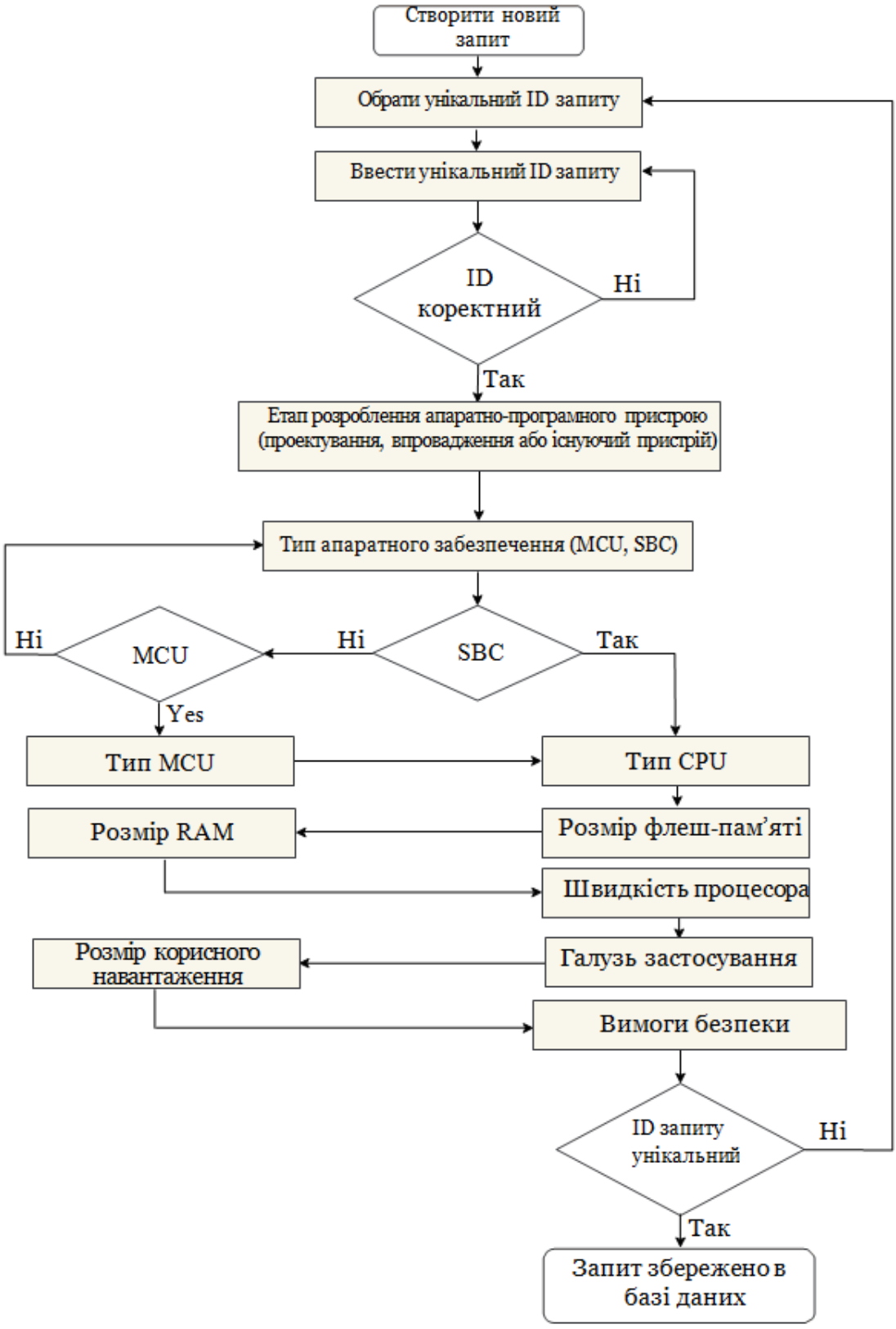


Рисунок 4.3 – Блок-схема алгоритму оброблення запитів стосовно програмної реалізації криптографічних алгоритмів

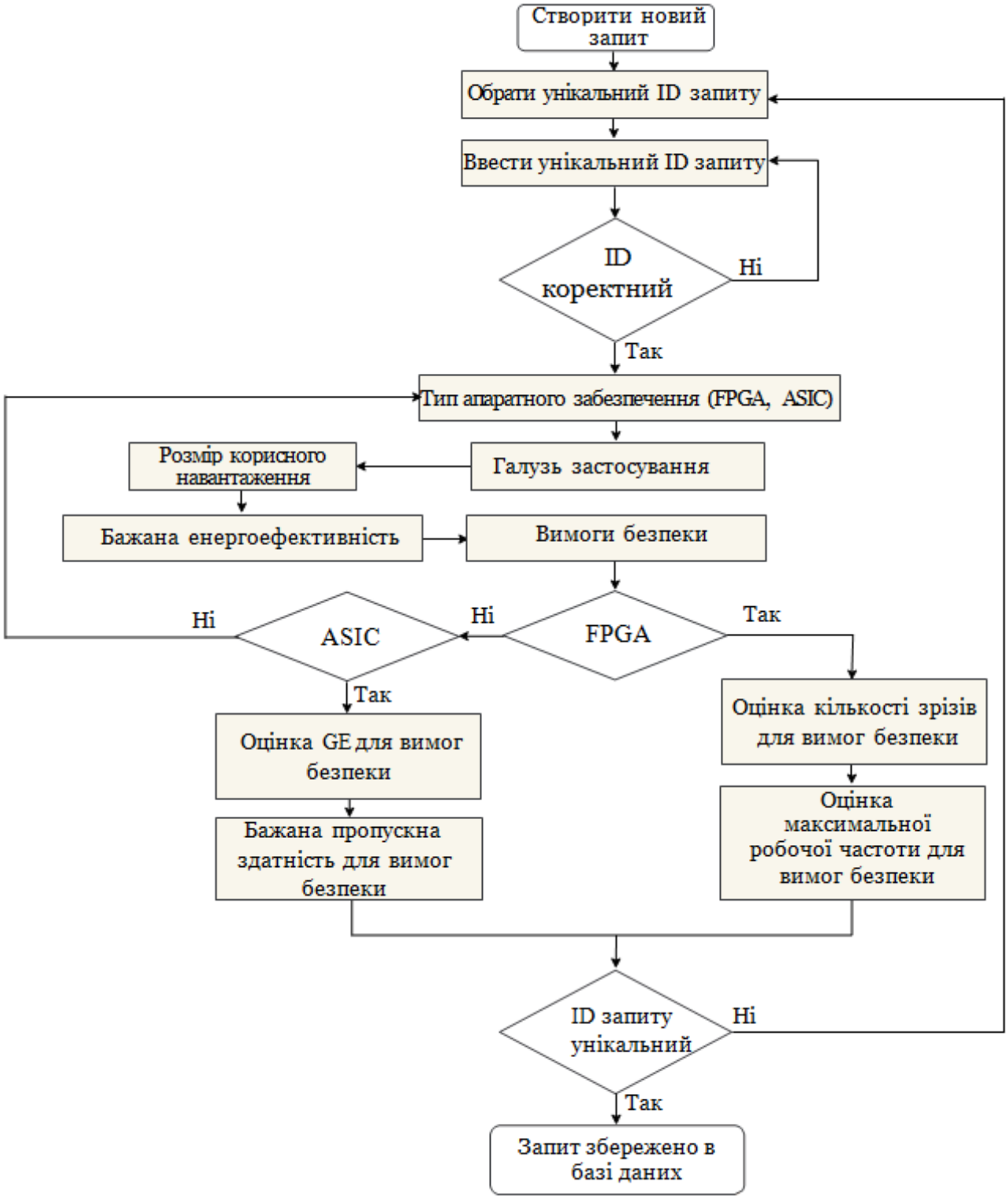


Рисунок 4.4 – Блок-схема алгоритму оброблення запитів стосовно апаратної реалізації криптографічних алгоритмів

Компонент фільтрації та обробки запитів керує введенням даних користувачем і виконує задачу перевірки на наявність помилок і вилучення небажаних даних із запиту користувача. Цей компонент функціонує як попередня

процедура, яка забезпечує ретельну перевірку вхідних даних, щоб уникнути хибних результатів. Він аналізує запити користувачів і забезпечує відповідність записів формату даних, визначеному в підсистемі формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей. Дані, які не відповідають чинному стандарту, відхиляються, і користувачам пропонується повторно ввести свої запити. Коли вводяться правильні дані, попередньо оброблені запити користувачів надходять до менеджера безпеки для подальшої обробки. Ці запити також зберігаються в сховищі даних для подальшої обробки.

Менеджер безпеки здійснює визначення типу запиту користувача (тобто, чи це є запит на програмну або апаратну реалізацію). Для запитів на програмну реалізацію користувач, яка приймає рішення, перевіряє можливості апаратного забезпечення (наприклад, розміри флеш-пам'яті та оперативної пам'яті мікроконтролера). Однак, якщо апаратне забезпечення є SBC, користувач перевіряє ці параметри лише для того, щоб переконатися, що вони є типовими для SBC, і якщо ні, то менеджер безпеки генерує повідомлення про помилку, щоб попередити користувача. Це пов'язано з тим, що більшість SBC можуть використовувати стандартні алгоритми безпеки. Крім того, як для програмної, так і для апаратної реалізації, чутливість галузі застосування є ключовим фактором при виборі відповідних криптографічних алгоритмів. Наприклад, лише найефективніші алгоритми обираються для критично важливих сфер застосування, таких як охорона здоров'я, інтелектуальний моніторинг людей похилого віку, банківська справа, роздрібна торгівля, інтелектуальні мережі та інші чутливі сфери застосування.

Менеджер безпеки аналізує запит користувача і приймає рішення на основі набору правил (бази правил) і метрик безпеки, а саме: механізмів безпеки, вимог безпеки і знань експертів.

Коли користувачі імпортують згенеровані вимоги безпеки з підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та

відомих стандартів безпеки в галузі Інтернету речей до підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей, весь перелік вимог безпеки імпортується до менеджера безпеки. За можливістю менеджер безпеки надає деяку інформацію про те, як користувачі можуть задовольнити вимоги безпеки, для яких не може бути рекомендовано жодного полегшеного криптографічного алгоритму. Крім того, якщо сформовані вимоги безпеки не включають конфіденційність плюс автентичність, особа, яка приймає рішення, перевіряє сформований перелік на предмет конфіденційності (та/або приватності) та цілісності. Якщо вони знайдені, то користувачам рекомендується повернутися до головного меню, щоб вибрати опцію 10 і змінити свої запити (рис. 4.1), включивши конфіденційність плюс автентичність, механізмом безпеки яких є автентифіковане шифрування, що може забезпечити як конфіденційність та/або приватність, так і автентичність.

Кожна з підсистем має модуль інтерфейсу виведення. Наприклад, модуль інтерфейсу виведення підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей виводить користувачеві згенеровані вимоги безпеки. Модуль інтерфейсу виведення підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей створює звіт, що складається з набору рекомендацій щодо відомих практик безпеки. Модуль інтерфейсу виведення підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей виводить на екран таблицю, що складається з вимог безпеки користувача, відповідних їм рекомендованих механізмів безпеки і полегшених криптографічних алгоритмів, які їх забезпечують.

Консольний додаток реалізовано на C++, він може працювати на ОС Windows і Linux. Реалізція надає можливість додавати нові функції, такі як інтеграція нових вимог безпеки. З метою реалізації сховища даних та оновлень,

підтримки запитів та реєстрації користувачів використано БД MySQL. Кожна з реалізованих підсистем (підсистема формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей; підсистема генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей; підсистема формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей) може функціонувати як окремо, так і в комплексі з іншими підсистемами.

Хоча кожна з підсистем не залежить від інших, користувач, який раніше використовував підсистему формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, може імпортувати згенеровані вимоги безпеки в підсистему формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей.

4.2 Алгоритми програмної реалізації методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

На рисунку 4.5 наведено алгоритм, який описує приклад функціонування підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей. В наведеному алгоритмі передбачається, що апаратно-програмний засіб знаходиться на етапі проєктування.

Вхідними даними алгоритму є галузь застосування, етап розроблення системи та відповіді на опитувальник, а вихідними – набір вимог безпеки до апаратно-програмного засобу Інтернету речей.

Алгоритм 1. Приклад алгоритму функціонування підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

1: Enter a request ID;
2: Select application domain;
3: Select system development phase.
4: Will the system have a user? (yes_1 or no_1);
5: **if** yes_1 **then**
6: Will the system have a user login? (yes_2 or no_2);
7: **if** yes_1 **then**
8: Will it store any user information? (yes_3 or no_3);
9: **if** yes_3 **then**
10: Will it store any other information? (yes_4 or no_4);
11: **else**
12: Will it store any information? (yes_5 or no_5);
13: **if** yes_3 or yes_5 **then**
14: What type of information? (*normal*, *sensitive*, or *critical*);
15: **if** yes_3 or yes_4 or yes_5 **then**
16: Will the information be sent to an entity? (yes_6 or no_6);
17: Will it be connected to the Internet? (yes_7 or no_7);
18: Will it send data to a cloud? (yes_8 or no_8);
19: Will it store data in a database? (yes_9 or no_9);
20: Will it receive regular updates? (yes_{10} or no_{10});
21: Will it use third-party software? (yes_{11} or no_{11});
22: Is there possibility of eavesdropping? (yes_{12} or no_{12});
23: Could messages sent between system components be captured and re-played? (yes_{13} or no_{13});
24: Can someone try to impersonate a user? (yes_{14} or no_{14});
25: Can someone with bad intentions gain physical access to the system? (yes_{15} or no_{15});

Рисунок 4.5 – Алгоритм функціонування підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

На рисунку 4.6 наведено алгоритм, який описує процедуру створення запиту та приклад функціонування підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей. В алгоритмі передбачається, що апаратно-програмний засіб є існуючою системою. Вхідними даними є етап розроблення апаратно-програмного засобу Інтернету речей, архітектура

апаратно-програмного засобу Інтернету речей та відповіді користувача; а вихідними – набір практичних рекомендацій з забезпечення вимог безпеки.

Алгоритм 2 . Приклад алгоритму функціонування підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

1: Enter a request ID;
2: Select system development phase;
3: Choose IoT system architecture.
4: Does the system have a user? (yes_1 or no_1);
5: **if** yes_1 **then**
6: Does it have a provision for user registration? (yes_2 or no_2);
7: **if** yes_2 **then**
8: Who register users? (*admin*, *users themselves*);
9: **if** yes_2 **then**
10: Is there user login? (yes_3 or no_3);
11: **if** yes_1 **then**
12: Does it allow users to enter any input? (yes_4 or no_4);
13: **if** yes_1 or yes_4 **then**
14: Does it store user information? (yes_5 or no_5);
15: **if** yes_5 **then**
16: Does it store any other information? (yes_6 or no_6);
17: **else**
18: Does it store any information? (yes_7 or no_7);
19: **if** yes_5 or yes_7 **then**
20: What type of information? (*normal*, *sensitive*, or *critical*);
21: What is the current authentication type? (*no authentication*, *username and password*, *etc.*);
22: Does it store data in a database? (yes_8 or no_8);
23: What is the type of data storage? (*SQL*, *NoSQL*, *Local storage*, *etc.*);
24: What type of database is used? (*SQL server*, *MySQL*, *SQLite*, *etc.*);
25: What programming language is use? (*C/C + +*, *Java*, *Ruby*, *Python*, *PHP* , *Javascript*, *etc.*);
26: Does it allow file uploads? (yes_9 or no_9);
27: Does it generate a log file? (yes_{10} or no_{10});

Рисунок 4.6 – Алгоритм функціонування підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

На рисунку 4.7 наведено алгоритм, який узагальнює процедуру обробки запиту підсистемою формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей.

Вхідними даними в алгоритмі є специфікація апаратного обладнання, розмір корисного навантаження, потреба в енергоспоживанні, вимоги до безпеки та галузь застосування; а вихідними даними є механізми безпеки та рекомендовані криптографічні алгоритми.

Алгоритм 3. Приклад алгоритму функціонування підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

```
1: begin
2: Enter a request ID;
3: Check request type (for software or hardware implementation);
4: while RAM and flash memory sizes are okay do
5:     Check sensitivity of application domain (sensitive, not sensitive);
6:     Check message payload type (small, average, large, continuous,
       unknown);
7:     if request is for hardware implementation then
8:         Check power requirement (low power, ultra low power);
9: Select the right security mechanism;
10: Recommend the most appropriate cryptographic algorithm;
11: end
```

Рисунок 4.7 – Алгоритм обробки запиту підсистемою формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

Більшість метричних параметрів, використаних у реалізації, отримано з Керівництва з криптографічних технологій CRYPTREC [82], а також з фреймворку FELICS [83]. У цій реалізації кілобайт (кБ) використовується як одиниця вимірювання пам'яті для запитів стосовно програмної реалізації через обмежений обсяг пам'яті на пристроях MCU з обмеженими ресурсами. Хоча SBC мають достатній обсяг пам'яті, для них також використовується кБ з метою уніфікації; а мегагерц (МГц) використовується як одиниця вимірювання тактової частоти процесора як для MCU, так і для SBC з тих же причин. Для реалізації кожного з доступних варіантів апаратних вимог використовуються дві метрики: площа л а н ц ю г а та пропускна здатність для ASIC, а також кількість слотів та максимальна робоча частота для ПЛІС. GE та кількість шарів використовуються

як одиниці вимірювання площі для платформ ASIC та FPGA відповідно. Кілобіти в секунду (Кбіт/с) і МГц приймаються як одиниці для пропускної здатності і максимальної робочої частоти відповідно.

Реалізація також включає механізми виявлення помилок, які попереджають користувачів про очевидно помилкове або непослідовне введення даних. Наприклад, в реалізації передбачено попередження користувачів про те, що їхні апаратні характеристики не є типовими для SBC, якщо вони помилково обирають SBC замість MCU. Також генерується попередження, коли можливості MCU занадто обмежені для алгоритмів у базі даних, а також, коли вводиться недійсний ідентифікатор запиту. Крім того, якщо користувач вводить неунікальний ідентифікатор запиту, виводиться повідомлення користувачу, що введений ідентифікатор запиту вже існує в базі даних. У такій ситуації користувачеві пропонується натиснути клавішу Enter, щоб повернутися в головне меню і повторити процес з унікальним ідентифікатором запиту.

4.3 Експериментальні дослідження методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

З метою перевірки ефективності розробленого методу було проведено експериментальні дослідження. З цією метою було розроблено ряд тестових завдань, які були виконані за допомогою програмної реалізації розробленого методу.

У таблицях 4.1 та 4.2 наведені тестові питання, відповіді на які було використано в якості вхідних даних для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей.

Таблиця 4.1 – Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувач	Чи буде (або чи є) логін у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати (або чи надсилає) інформацію іншим суб'єктам
R1111	Розроблення	Розумне місто	Ні			Так	Не чутлива	Так
R2345	Розроблення	Іграшка	Так	Так	Так	Так	Не чутлива	Так
R4444	Існуюча	Розумний будинок	Так	Так	Так	Ні	Чутлива	Так
R1234	Розроблення	Носимий пристрій	Так	Так	Так	Так	Не чутлива	Так
R6548	Розроблення	Домашній улюбленець	Так	Так	Так	Так	Не чутлива	Ні
R0601	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Не чутлива	Так

Продовження таблиці 4.1 – Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувач	Чи буде (або чи є) логін у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати (або чи надсилає) інформацію іншим суб'єктам
R7788	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Чутлива	Ні
R5432	Розроблення	Розумна мережа	Так	Так	Так	Так	Чутлива	Так
R2278	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Чутлива	Так
R9128	Розроблення	Навколишнє середовище	Ні			Так	Критична	Так
R6666	Розроблення	Штучний інтелект	Ні			Так	Не чутлива	Ні
R1115	Розроблення	Сільське господарство	Так	Так	Так	Так	Не чутлива	Так
R1995	Розроблення	Іграшка	Так	Так	Ні	Так	Не чутлива	Так
R4321	Розроблення	Екологія	Так	Ні	Так	Ні	Чутлива	Так
R1235	Розроблення	Роздрібна торгівля	Так	Так	Ні	Так	Не чутлива	Так
R8374	Розроблення	Охорона здоров'я	Так	Так	Так	Ні	Чутлива	Так

Кінець таблиці 4.1 – Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувач	Чи буде (або чи є) логін у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати (або чи надсилає) інформацію іншим суб'єктам
R4040	Розроблення	Розумна мережа	Так	Так	Ні	Так	Не чутлива	Ні
R8789	Розроблення	Розумне місто	Так	Так	Так	Так	Критична	Так
R1008	Розроблення	Люди похилого віку	Так	Так	Так	Так	Критична	Так
R5555	Розроблення	Автомобіль	Так	Так	Так	Так	Не чутлива	Ні
R1287	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Критична	Так
R5461	Розроблення	Розумна мережа	Так	Так	Ні	Ні		
R0011	Існуюча	Розумний будинок	Ні			Так	Критична	Так
R8126	Розроблення	Розумне місто	Так	Так	Так	Так	Чутлива	Так

Таблиця 4.2 – Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Чи підключена (чи буде підключена) система до Інтернету	Чи буде система надсилати (або надсилає) дані на будь-яку хмарну платформу	Чи буде система зберігати (або зберігає) дані в базі даних	Чи буде система отримувати (або отримує) регулярні оновлення	Чи буде система використовувати (або використовує) стороннє програмне забезпечення	Чи існує ймовірність атак на підслуховування	Чи існує ймовірність атак повторного відтворення	Чи існує ймовірність того, що зловмисники видають себе за користувача	Чи існує ймовірність того, що зловмисник може мати фізичний доступ до системи
R1111	Так	Так	Так	Так	Ні	Так	Так		Так
R2345	Так	Так	Так	Так	Так	Так	Так	Так	Так
R4444	Так	Так	Так	Так	Так	Так	Так	Ні	Так
R1234	Так	Так	Так	Так	Ні	Ні	Ні	Так	Так
R6548	Так	Так	Так	Так	Так	Так	Так	Так	Так
R0601	Так	Ні	Так	Так	Так	Так	Так	Ні	Ні
R7788	Так	Так	Так	Ні	Ні	Ні	Ні	Так	Так
R5432	Так	Ні	Ні	Так	Ні	Так	Ні	Так	Ні
R2278	Так	Так	Так	Ні	Ні	Ні	Так	Так	Так
R9128	Так	Так	Так	Так	Так	Ні	Ні		Так
R6666	Так	Ні	Так	Ні	Так	Ні	Ні		Так
R1115	Так	Так	Ні	Так	Ні	Так	Так	Так	Так
R1995	Так	Ні	Ні	Ні	Ні	Так	Ні	Ні	Так
R4321	Так	Так	Так	Так	Ні	Ні	Так	Так	Ні
R1235	Так	Так	Так	Так	Так	Так	Так	Ні	Ні
R8374	Так	Так	Так	Ні	Ні	Ні	Ні	Так	Ні

Продовження таблиці 4.2 – Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Чи підключена (чи буде підключена) система до Інтернету	Чи буде система надсилати (або надсилає) дані на будь-яку хмарну платформу	Чи буде система зберігати (або зберігає) дані в базі даних	Чи буде система отримувати (або отримує) регулярні оновлення	Чи буде система використовувати (або використовує) стороннє програмне забезпечення	Чи існує ймовірність атак на підслуховування	Чи існує ймовірність атак повторного відтворення	Чи існує ймовірність того, що зловмисники видають себе за користувача	Чи існує ймовірність того, що зловмисник може мати фізичний доступ до системи
R4040	Так	Ні	Ні	Ні	Ні	Так	Так	Так	Ні
R8789	Так	Так	Ні	Ні	Так	Так	Ні	Ні	Ні
R1008	Так	Ні	Ні	Так	Так	Ні	Ні	Так	Ні
R5555	Так	Так	Ні	Ні	Ні	Ні	Ні	Так	Так
R1287	Так	Ні	Так	Так	Так	Так	Так	Так	Так
R5461	Так	Ні	Так	Так	Так	Так	Так	Так	Так
R0011	Так	Ні	Так	Так	Ні	Ні	Ні		Так
R8126	Так	Ні	Ні	Ні	Ні	Так	Так	Так	Так

Для прикладу розглянемо результати сформованих вимог безпеки для запитів R1995, R5432 та R1287, для яких в результаті експерименту було сформовано перелік з 6, 11 та 15 вимог безпеки відповідно.

Проаналізувавши одержані результати, можна стверджувати, що підсистема формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей згенерувала достатньо

повний перелік вимог безпеки на основі тестових даних. Це твердження можна перевірити, провівши ретельне вивчення та порівняльний аналіз запитів із зазначеними вище ідентифікаторами запитів (табл.4.1, 4.2) та одержаних результатів, представлених на рисунках 4.8, 4.9 та 4.10 відповідно.

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Physical Security	Refers to the security measures designed to deny unauthorized physical access to IoT devices or systems, and to protect them from damage or tampering.

Рисунок 4.8 – Перелік вимог безпеки для запиту R1995

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Privacy	Refers to users control over the disclosure of their personal information, meaning that only the users should decide whether they want to share their data or not.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Authorization	Refers to the property that determines whether the user or device has rights/privileges to access a resource, or issue commands.
Forgery Resistance	This is the propriety that ensures that data shared between entities and updates cannot be forged by a third party trying to damage or harm the system or its users.
Non-Repudiation	Refers to the security property that ensures that the transfer of messages or credentials between 2 IoT entities is undeniable.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Accountability	This is the property that ensures that every action can be traced back to a single user or device.
Reliability	Is the property that guarantees consistent intended behavior of an IoT system.

Рисунок 4.9 – Перелік вимог безпеки для запиту R5432

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Privacy	Refers to users control over the disclosure of their personal information, meaning that only the users should decide whether they want to share their data or not.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Physical Security	Refers to the security measures designed to deny unauthorized physical access to IoT devices or systems, and to protect them from damage or tampering.
Authorization	Refers to the property that determines whether the user or device has rights/privileges to access a resource, or issue commands.
Forgery Resistance	This is the propriety that ensures that data shared between entities and updates cannot be forged by a third party trying to damage or harm the system or its users.
Non-Repudiation	Refers to the security property that ensures that the transfer of messages or credentials between 2 IoT entities is undeniable.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Accountability	This is the property that ensures that every action can be traced back to a single user or device.
Reliability	Is the property that guarantees consistent intended behavior of an IoT system.
Counterfeit Resistance	Is the property that ensures effective validation of software such that any fake or maliciously modified software is rejected.
Data Freshness	Ensures that data is the most recent, and that old messages cannot be replayed.
Tamper Detection	Ensures all devices are physically secured, such that any tampering attempt is detected.

Рисунок 4.10 – Перелік вимог безпеки для запиту R1287

Варто зазначити, що в деяких випадках позитивна відповідь не на одне, а на декілька запитань може спричинити необхідність включення певної вимоги безпеки до згенерованого результату вимог безпеки для даного користувача. Наприклад, позитивна відповідь на будь-яке з питань: «Чи буде система надсилати дані?» та «Чи існує можливість підслуховування?» може спричинити необхідність включення вимоги «Конфіденційність» до переліку вимог безпеки.

Аналогічно, наступні питання та відповіді користувачів можуть спровокувати необхідність включення «Стійкості до підробки» до списку вимог безпеки: «Який тип інформації буде зберігатися в системі?» – «Конфіденційна», «Чи буде вона надсилати дані до хмари?» – «Так», «Чи надсилатиме дані до бази даних?» – «Так», і «Чи буде вона отримувати регулярні оновлення?» – «Так». Отже, щоб уникнути багаторазового включення вимоги безпеки, у сформованому списку вимог безпеки відображається лише перший екземпляр.

У запиті з ідентифікатором R1995 було вказано, що розроблюваний пристрій Інтернету речей матиме користувачів і логін користувача, що зумовлює необхідність для користувачів перевірити або довести, що вони дійсно є тими суб'єктами, за яких себе видають, а отже, виникає потреба в автентифікації (перша вимога, рис. 4.8), яка перевіряє автентичність суб'єктів. Також було зазначено, що пристрій буде ділитися своїми даними з іншими об'єктами, що може дозволити зловмисникам підслуховувати комунікації між пристроєм та іншими об'єктами; зловмисники також можуть спробувати модифікувати дані під час передачі між пристроєм та іншими об'єктами. Це зумовлює необхідність забезпечення конфіденційності (друга вимога, рис. 4.8), яка обмежує несанкціонований доступ до інформації, та цілісність (третя вимога, рис. 4.8), яка стосується забезпечення достовірності, походження та коректності даних відповідно.

Також пристрій буде зберігати певну інформацію, і як пристрій Інтернету речей, який буде підключений до Інтернету (в даному випадку це розумна іграшка), може стати недоступним для авторизованих користувачів через DoS-атаки або через MitM-атаки шляхом перехоплення і знищення повідомлень, що може призвести до припинення зв'язку між пристроєм та іншими об'єктами і, таким чином, викликати проблеми з доступністю. Таким чином, доступність (четверта вимога, рис. 4.8), яка забезпечує доступність і зручність використання системи на вимогу, є важливою вимогою безпеки, яка повинна бути виконана. Крім того, зловмисники можуть використовувати MitM-атаки для викрадення або пошкодження пристрою, що підкреслює необхідність ізоляції (п'ята вимога безпеки на рис. 4.8) – вимога безпеки, яка обмежує поширення наслідків атак на

інші пристрої. Крім того, під час опитування було зазначено, що особа зі зловмисними намірами може отримати фізичний доступ до пристрою. Отже, існує потреба в заходах фізичної безпеки, які можуть запобігти несанкціонованому фізичному доступу до пристрою (шоста вимога безпеки на рис. 4.8). Оскільки розумна іграшка не вважається чутливою галуззю застосування, тому такі вимоги безпеки, як неспростовність, підзвітність, виявлення несанкціонованого втручання та надійність не вважаються необхідними, тому вони не були включені в результат.

В той же час, запит R5432 (рис. 4.9) показує, що розумна мережа буде збирати і зберігати інформацію про користувача, що викликає занепокоєння щодо конфіденційності користувачів. Таким чином, конфіденційність є важливою вимогою безпеки, яка необхідна для захисту прав користувачів на конфіденційність їхньої особистої інформації, що є другою вимогою безпеки (рис. 4.9).

Оскільки у розумній мережі будуть користувачі і логін користувача, і вона розглядається як чутлива галузь Інтернету речей, тому крім необхідності аутентифікації користувача існує також необхідність призначення користувачам набору дозволів, прав або привілеїв. Призначення рівня привілеїв визначає, що саме користувачі можуть робити в системі на основі політики безпеки організації, яку зазвичай визначає системний адміністратор, і саме це забезпечує авторизація (шоста вимога безпеки на рис. 4.9). Виходячи з чутливості галузі та у поєднанні з тим, що додаток буде надсилати дані іншим суб'єктам, буде підключений до Інтернету та отримуватиме регулярні оновлення, стійкість до підробки (сьома вимога безпеки на рис. 4.9) є життєво важливою вимогою безпеки, яка може гарантувати, що додаток буде стійким до підробки повідомлень та оновлень, які можуть призвести до збою в роботі системи або її пошкодження. Крім того, чутливість галузі зумовлює необхідність включення невідмовності та підзвітності до переліку вимог безпеки, які гарантують, що суб'єкти не можуть заперечувати свою участь у будь-якій дії і що кожна дія може бути відстежена до одного суб'єкта відповідно (восьма та десята вимоги безпеки на рис. 4.9). Параметр

чутливості галузі також призводить до включення надійності, яка гарантує послідовну передбачувану поведінку системи, що є останньою вимогою безпеки (рис. 4.9).

Аналізуючи запит з ідентифікатором R1287 (табл.4.1, 4.2), можна побачити, що відповіді «Так» було надано на всі питання з варіантами відповіді «Так» або «Ні», крім одного. Це, а також той факт, що «розумна» охорона здоров'я вважається критично важливою сферою застосування, це призвело до генераціх 15 вимог до безпеки. Пристрій буде використовувати стороннє програмне забезпечення, а отже, необхідно захиститися від використання зловмисно модифікованого програмного забезпечення, яке може завдати шкоди системі, і саме цього намагається досягти вимога безпеки щодо стійкості до підробок (тобто тринадцята вимога безпеки на рис. 4.10). Крім того, зловмисники можуть захопити і відтворити повідомлення. Отже, оскільки система матиме логін користувача, і ці дані будуть надсилатися іншим суб'єктам та до бази даних, існує потреба у забезпеченні безпеки свіжості даних. Це гарантує, що дані будуть найсвіжішими і, таким чином, захистить систему від атак на відтворення; свіжість даних є чотирнадцятою вимогою безпеки (рис. 4.10). Як видно з таблиць 4.1 та 4.2, особа зі зловмисними намірами може отримати фізичний доступ до пристрою. Однак, зважаючи на чутливість сфери застосування, окрім необхідності вжиття заходів фізичного захисту, існує також потреба у виявленні будь-яких активних спроб порушити цілісність пристрою або даних, пов'язаних з ним, і саме на це спрямована вимога безпеки щодо виявлення несанкціонованого доступу, яка є останньою вимогою безпеки (рис. 4.10).

Результати експериментів для підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей та підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей наведені в додатку Б. З метою проведення експериментів для підсистеми формування вимог стосовно застосування

полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей було використано наступні сценарії тестування:

- сценарій тестування програмної реалізації з використанням MCU: у цьому сценарії користувач, який розробляє апаратно-програмний пристрій Інтернету речей з використанням MCU (або інтелектуального додатку, який буде працювати на MCU), запитує криптографічні алгоритми для програмної реалізації, які задовольнять вимоги безпеки апаратно-програмного пристрою Інтернету речей;
- сценарій тестування програмної реалізації з SBC: цей сценарій також складається із запиту на реалізацію програмного забезпечення, в якому користувач, що проєктує апаратно-програмний пристрій Інтернету речей з використанням апаратного забезпечення SBC (або інтелектуального додатку, який буде працювати на SBC), запитує криптографічні алгоритми, які будуть відповідати вимогам безпеки апаратно-програмного пристрою Інтернету речей;
- сценарій тестування апаратної реалізації з використанням ASIC: у цьому сценарії користувач запитує інформацію про відповідні криптографічні алгоритми, які забезпечать механізми безпеки для апаратно-програмного пристрою Інтернету речей з використанням ASIC;
- сценарій тестування апаратної реалізації з ПЛІС: у цьому сценарії запиту на апаратну реалізацію користувач використовує ПЛІС для проектування апаратно-програмного пристрою Інтернету речей і робить запит на визначення криптографічних алгоритмів для апаратної реалізації, які можуть бути використані для досягнення вимог безпеки апаратно-програмного пристрою Інтернету речей;
- тестовий сценарій, який демонструє імпорт вимог безпеки з підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей до підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей: у цьому тестовому сценарії

користувач спочатку використовує підсистему формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей для створення вимог безпеки апаратно-програмного пристрою Інтернету речей, який він хоче спроектувати, а потім імпортує створені вимоги безпеки в підсистему формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей як частину своїх вхідних даних.

Аналіз результатів експериментальних досліджень показує, що використання запропонованого методу дозволяє одержати достатньо повний перелік вимог безпеки, що надає можливість синтезувати апаратно-програмні засоби Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів, що дозволить підвищити безпеку інфраструктури Інтернету речей.

4.4 Висновок

В четвертому розділі було представлено програмну реалізацію та алгоритми методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей. На відміну від відомих підходів, комплексне врахування відомих стандартів безпеки в інфраструктурі Інтернету речей, правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки та вимог з вибору та впровадження полегшених програмних, а також апаратних криптографічних алгоритмів надає можливість дотримуватись вимог безпеки до

апарано-програмних засобів Інтернету речей з врахуванням галузі їх засосування та апаратної платформи.

Також, було проведено експериментальні дослідження, які показали результативність розробленого методу. Таким чином, застосування розробленого методу надає можливість синтезувати апаратно-програмні засоби Інтернету речей з врахуванням важливих вимог безпеки, що надасть можливість підвищити безпеку інфраструктури Інтернету речей.

ВИСНОВКИ

В дипломній роботі за результатами виконаних теоретичних та практичних досліджень було розроблено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

У першому розділі була досліджена предметна область, досліджені відомі стандарти та рішення синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

У другому розділі удосконалено модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. На відміну від відомих моделей, запропонована модель враховує множину вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, а також грунтується на правилах синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей та використовує множину правил впровадження полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей. Запропонована модель також враховує апаратну плтаформу апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування. Запропонована модель є основою методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Третій розділ представляє метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Запропонований метод надає можливість синтезувати апаратно-програмні засоби Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів.

У четвертому розділі представлено програмну реалізацію та алгоритми методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Також, було проведено експериментальні дослідження, які показали результативність розробленого методу. Таким чином, застосування розробленого методу надає можливість синтезувати апаратно-програмні засоби Інтернету речей з врахуванням важливих вимог безпеки, що надасть можливість підвищити безпеку інфраструктури Інтернету речей.

За темою дипломної роботи опубліковано статтю «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus, а також прийнято до публікації статтю «Виявлення кібератак в інфраструктурі Інтернету речей на основі машинного навчання». у науковому фаховому виданні «Вісник Хмельницького національного університету» (Технічні науки).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining Denysiuk, D., Bobrovnikova, K., Lysenko, S., Havryliuk, R., Boichuk, Y. *Proceedings of the 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, IDAACS 2021. P. 779-784.

2. Kumar N., Vidyarthi D. P. A Green Routing Algorithm for IoT-Enabled Software Defined Wireless Sensor Network. IEEE Sensors Journal. 2018. Vol. 18, No. 22. P. 9449–9460.

3. Patel J., Trivedi P., Patel D. A Performance Analysis of “Light Fidelity” and “Internet of Things” It’s Application. *Proceedings of the IEEE International Conference on Transforming Engineering Education (ICTEE)*. 2017, P. 1-4.

4. Ngu A. H., Gutierrez M., Metsis V., Nepal S., Sheng Q. Z. IoT Middleware: A Survey on Issues and Enabling Technologies. *IEEE Internet of Things Journal*. 2017. Vol. 4, No. 1. P. 1-20.

5. Lokuhitige S., Brown S. Forecasting Maturity of IoT Technologies in Top 5 Countries Using Bibliometrics and Patent Analysis. *Proceedings of the IEEE International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*. 2017. P. 338-341.

6. Ramadan M. Industry 4.0: Development of Smart Sunroof Ambient Light Manufacturing System for Automotive Industry. *Proceedings of the Advances in Science and Engineering Technology International Conferences (ASET)*. 2019. P. 1–5.

7. Truong H. Integrated Analytics for IIoT Predictive Maintenance Using IoT Big Data Cloud Systems. *Proceedings of the IEEE International Conference on Industrial Internet (ICII)*. 2018. P. 109-118.

8. Arumugam S. S., Badrinath R., Herranz A. H., Höller J., Azevedo C. R. B., Xiao B., Tudor V. Accelerating Industrial IoT Application Deployment through Reusable AI Components. *Proceedings of the Global Internet of Things Summit (GIoTS)*. 2019. P. 1-4.

9. Annamalai P., Bapat J., Das D. Emerging Access Technologies and Open Challenges in 5G IoT: From Physical Layer Perspective. *Proceedings of the IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. 2018. P. 1-6.

10. Wang N., Wang P., Alipour-Fanid A., Jiao L., Zeng K. Physical Layer Security of 5G Wireless Networks for IoT: Challenges and Opportunities. *IEEE Internet of Things Journal*. 2019. P. 1-1.

11. Cyr B., Mahmod J., Guin U. Low-Cost and Secure Firmware Obfuscation Method for Protecting Electronic Systems From Cloning. *IEEE Internet of Things Journal*. 2019. Vol. 6, No. 2. P. 3700-3711.

12. Oh S., Kim Y. Development of IoT Security Component for Interoperability. *Proceedings of the 13th International Computer Engineering Conference (ICENCO)*, 2017. P. 41-44.

13. Wang W., He S., Sun L., Jiang T., Zhang Q. Cross-Technology Communications for Heterogeneous IoT Devices Through Artificial Doppler Shifts. *IEEE Transactions on Wireless Communications*. 2019. Vol. 18, No. 2. P. 796-806.

14. Konduru V. R., Bharamagoudra M. R. Challenges and Solutions of Interoperability on IoT: How Far have we Come in Resolving the IoT Interoperability Issues. *Proceedings of the IEEE International Conference On Smart Technologies For Smart Nation (SmartTechCon)*. 2017. P. 572-576.

15. Gharbieh M., ElSawy H., Bader A., Alouini M. Spatiotemporal Stochastic Modeling of IoT Enabled Cellular Networks: Scalability and Stability Analysis. *IEEE Transactions on Communications*. 2017. Vol. 65, No. 8. P. 3585-3600.

16. Spehar J., Fuks A., Vauclair M., Meijer M., van Beek J., Shao B. Power Challenges Caused by IOT Edge Nodes: Securing and Sensing Our World. *Proceedings of the 2019 31st International Symposium on Power Semiconductor Devices and ICs (ISPSD)*. 2019. P. 17-22.

17. Frustaci M., Pace P., Aloi G., Fortino G. Evaluating Critical Security Issues of the IoT World: Present and Future Challenges. *IEEE Internet of Things Journal*. 2018. Vol. 5, No. 4. P. 2483-2495.

18. Moongilan D. 5G Internet of Things (IOT) Near and Far-Fields and Regulatory Compliance Intricacies. *Proceedings of the 2019 IEEE 5th World Forum on Internet of Things (WF-IoT).* 2019. P. 894-898.

19. Saghezchi F. B., Mantas G., Ribeiro J., Al-Rawi M., Mumtaz S., Rodriguez J. Towards a Secure Network Architecture for Smart Grids in 5G Era. *Proceedings of the 13th IEEE International Wireless Communications and Mobile Computing Conference (IWCMC).* 2017. P. 121-126.

20. Pal S., Hitchens M., Rabehaja T., Mukhopadhyay S. Security requirements for the Internet of Things: A systematic approach. Sensors. Vol. 20, No. 20. P. 5897.

21. Trend Micro. Inside the Smart Home: IoT Device Threats and Attack Scenarios. URL: https://www.trendmicro.com/vinfo/it/security/news/internet-of-things/inside-the-smart-home-iot-device-threats-and-attack-scenarios (Last accessed: 02.03.2023).

22. Check point software technologies LTD. Cyber security report 2022. URL: https://www.checkpoint.com/pages/cyber-security-report-2022/ (Last accessed: 02.03.2023).

23. OWASP Internet of Things. URL: https://owasp.org/www-project-internet-of-things/#tab=IoT_Attack_Surface_Areas (Last accessed: 12.03.2023).

24. Nozomi Networks Labs. What IT Needs to Know about OT/IoT Security Threats in 2023. URL: https://www.nozominetworks.com/blog/what-it-needs-to-know-about-ot-io-security-threats-in-2023/ (Last accessed: 10.03.2023).

25. McAfee Labs Threats Report. URL: https://www.mcafee.com/enterprise/en-us/lp/threats-reports/oct-2022.html (Last accessed: 8.03.2023).

26. Global System for Mobile Communications. URL: https://www.gsma.com/ (Last accessed: 02.03.2023).

27. Ravi, N., & Shalinie, S. M. Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture. *IEEE Internet of Things Journal*. 2020. 7(4). P. 3559-3570.

28. Hassan Wan Haslina. Current research on Internet of Things (IoT) security: A survey. *Computer networks*. 2019. Vol. 148. P. 283-294.

29. Najib A.A., Munadi R., Karna N.B.A. Security system with RFID control using E-KTP and internet of things. *Bulletin of Electrical Engineering and Informatics*. 2021. P.1436-1445.

30. Alharbi A., Alosaimi W. Botnet attack detection using local global best bat algorithm for industrial internet of things. *Electronics*, 10(11). 2021, P.1341.

31. Khan M. A., Khan Khattk M. A.Voting classifier-based intrusion detection for IoT networks. In *Advances on Smart and Soft Computing*, Springer, Singapore. 2022. P. 313-328.

32. Abrishamchi M.A., Cheok A.D., Abdullah A.H., Bielawski K.S. In-Home Surveillance Systems and Privacy Considerations for Malaysians: A Survey. *Int. J. Innov. Comput.* 2018. №8 P. 47-51.

33. Mihoub A., Fredj O. B. Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques. *Computers & Electrical Engineering*. 2022. 98. P. 107716.

34. Ding F., Li Z., Ai C., Su R., Zhang D., Zhu H. July. Design of an IoT-Based Efficient Security Scheme in Home Wireless System. *International Conference on Artificial Intelligence and Security*. 2018. P. 287-296.

35. Ahmad S., Saha A., Chek L.W., Mekhilef S., Azam T., Ahmed M., Orabi M., Ghoneim S., Alharthi M., Alamri B. Smart home automation and security system design based on iot applications. *Asean engineering journal*. 2019. P. 57-71.

36. Cardin O. Classification of cyber-physical production systems applications: proposition of an analysis framework. *Computers in Industry*. 2019. V.104. P. 11–21.

37. Pangaribowo E.H., Keban Y.T., Darwin M. Elderly care: A study on community care services in Sleman, DIY. *Indonesia. Journal of Aging Research*. 2020. P. 259-317.

38. Hong S. Technology trends and policies for IoT security. 2020. P. 1-6.

39. Ibarra-Esquer J. E., González-Navarro F. F., Flores-Rios B. L., Burtseva L., Astorga-Vargas M. A., Tracking the evolution of the internet of things concept across different application domains. *Sensors*. 2017. № 17(6). P. 1379.

40. Park E., Del Pobil A.P., Kwon S.J. The role of Internet of Things (IoT) in smart cities: *Technology roadmap-oriented approaches*. 2018. P.1388.

41. Tiruvayipati S., Yellasiri R., Viability of an Uncomplicated IoT SaaS Development for Deployment of DIY Applications Over HTTP with Zero Investment. In Advances in Decision Sciences, Image Processing, Security and Computer Vision. 2020. P. 206-213.

42. Ande R., Adebisi B., Hammoudeh M., Saleem J. Internet of Things: Evolution and technologies from a security perspective. *Sustainable Cities and Society*. 2020. P. 101-128.

43. Abraham S., Vurkaç M., Miguel A., Nguyen N.K., Ong O.J.S. Teaching Embedded Systems in the Context of Internet of Things. 2019.

44. Fortino, G., Guerrieri, A., Pace, P., Savaglio, C., Spezzano, G. IoT Platforms and Security: An Analysis of the Leading Industrial/Commercial Solutions. *Sensors*. 2022. P. 296.

45. Stojkoska B., Trivodaliev K. A review of Internet of Things for smart home: Challenges and solutions. *J. Clean. Prod.* 2018 №140 P. 1454-1464.

46. Wang P., Yao C., Zheng Z., Sun G., Song L. Joint Task Assignment, Transmission, and Computing Resource Allocation in Multilayer Mobile Edge Computing Systems. *IEEE IoT J*. 2018. №6. P. 2872–2884.

47. Atzori L., Iera A., Morabito G. The Internet of Things: A Survey, Computer Networks. 2018. V. 54. № 15. P. 2787-2805.

48. Gubbi J, Buyya R, Marusic S, Palaniswami M, Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*. 2023. V. 29, № 7. P. 1645-1660.

49. GDPR**.** URL: https://gdpr.eu/what-is-gdpr/ (Last accessed: 02.03.2023).

50. ENISA**.** URL: https://www.enisa.europa.eu/ (Last accessed: 04.03.2023).

51. ENISA**.** Good Practices for Security of IoT – Secure Software Development Lifecycle**.** URL: https://www.enisa.europa.eu/publications/good-practices-for-security-of-iot-1 (Last accessed: 02.03.2023).

52. ISO/IEC 27001 Information security management systems**.** URL: https://www.iso.org/standard/27001 (Last accessed: 21.03.2023).

53. ISO/IEC 27032:2012. Information technology – Security techniques – Guidelines for cybersecurityhttps**.** URL: //www.iso.org/ru/standard/44375.html (Last accessed: 21.03.2023).

54. ISO/IEC 27035-1:2023. Information technology – Information security incident management – Part 1: Principles and process**.** URL: https://www.iso.org/standard/78973.html (Last accessed: 21.03.2023).

55. ISO/IEC 27031:2011. Information technology – Security techniques – Guidelines for information and communication technology readiness for business continuity**.** URL: https://www.iso.org/standard/44374.html (Last accessed: 23.03.2023).

56. ISO 22301:2012. Societal security – Business continuity management systems – Requirements**.** URL: https://www.iso.org/ru/standard/50038.html (Last accessed: 23.03.2023).

57. NISTIR 8228. Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks**.** URL: https://csrc.nist.gov/publications/detail/nistir/8228/final (Last accessed: 21.03.2023).

58. Jayakumar H., Lee K., Lee W. S., Raha A., Kim Y., Raghunathan V., Powering the Internet of Things. *International Symposium on Low Power Electronics and Design (ISLPED).* 2023. P. 375- 380.

59. Guo K., Li N., Kang J., Zhang J. Towards efficient federated learning- based scheme in medical cyber-physical systems for distributed data. *Software: Practice and Experience.* 2021. №51. P. 2274-2289.

60. Boubekeur M. Industrial applications for cyber-physical systems. *First International Conference on Embedded & Distributed Systems*. 2017. P. 17-18.

61. Grispos G., Glisson W.B., Choo K. R. Medical Cyber-Physical Systems Development: A Forensics-Driven Approach. In Proceedings of IEEE. *ACM Conference on Connected Health: Applications, Systems and Engineering Technologies.* 2017. P. 108-114.

62. Jiafu Wan, Hehua Yan, Hui Suo, Fang Li. Advances in CyberPhysical Systems. Research School of Computer Science and Engineering. *South China University of Technology Guangzhou*. *China* DOI: 2011.11.001. 10.3837/tiis.

63. Silva L.C., Almeida H.O., Perkusich A., Perkusich M. A Model-Based Approach to Support Validation of Medical Cyber-Physical Systems. *Sensors.* 2019. P.27625-27670.

64. Sankavaram C, Kodali A, Pattipati K, An integrated health management process for automotive cyber-physical systems. *International Conference on Computing*. *Networking and Communications (ICNC).* 2018. P. 82-86.

65. Kyoung-Dae Kim, Behrad Bagheri P.R., Shanhu Yang, Hung-An Kao, Jay Lee. An Overview and Some Challenges in Cyber-Physical Systems Some Challenges in Cyber-Physical Systems. *Cyber-physical Systems Architecture for SelfAware Machines in Industry 4.0 Environment, IFAC-Papers On Line.* 2018. № 48. P. 1622-1627.

66. Insup Lee. Medical Cyber Physical Systems. *47th Design Automation Conference* №10. 2018. P. 743-748.

67. Minerva R., Biru A., Rotondi D. Towards a definition of the Internet of Things (IoT) // *IEEE Internet Initiative*. 2015. №1. P. 1-86.

68. Greer C., Burns M., Wollman D., Griffor E. Cyber-physical systems and Internet of Things / *NST Special Publication 1900*. 2019. №202. P.52.

69. Ibarra-Esquer J.E, González-Navarro F.F, Flores-Rios B.L, Burtseva L, AstorgaVargas M.A. Tracking the evolution of the internet of things concept across different application domains. *Sensors*. 2017. № 17(6). P. 1379.

70. Alzubi A.A., Al-Maitah M., Alarifi A. Cyber-attack detection in healthcare using cyber-physical system and machine learning techniques. *Soft Comput 25*. 2021. P.12319-12332.

71. Khaitan S. K., McCalley J. D., Design techniques and applications of cyberphysical systems / *A survey IEEE Systems Journal*. 2018. № 9(2). P. 350-365.

72. Beaulieu R., Treatman-Clark S., Shors D. The SIMON and SPECK Lightweight Block Ciphers. *Proceedings of the 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*. 2015. P. 1-6.

73. Mohd B. J., Hayajneh T. A Survey on Lightweight Block Ciphers for Low-Resource Devices: Comparative Study and Open Issues. *Journal of Network and Computer Applications*. 2015. Vol. 58. P. 73-93.

74. Sallam S. A Survey on Lightweight Cryptographic Algorithms. *Proceedings of the TENCON 2018 - 2018 IEEE Region 10 Conference*. 2018. P. 1784-1789.

75. Santis F. D., Schauer A., Sigl G. ChaCha20-Poly1305 Authenticated Encryption for High-speed Embedded IoT Applications. *Proceedings of the Design, Automation Test in Europe Conference Exhibition (DATE)*. 2017. P. 692-697.

76. Sharafi M., Fotouhi-Ghazvini F., Shirali M. A Low Power Cryptography Solution Based on Chaos Theory in Wireless Sensor Nodes. *IEEE Access*. 2019. Vol. 7. P. 8737-875.

77. Ammar M., Russello G. Internet of Things: A Survey on the Security of IoT Fameworks. *Journal of Information Security and Applications*. Vol. 38. 2018. P. 8 - 27.

78. Arthur Gatouillat, Youakim Badr, Bertrand Massot, Ervin Sejdić. Internet of Medical Things: A Review of Recent Contributions Dealing with Cyber-Physical Systems in Medicine. *IEEE internet of things journal, IEEE*. 2018. №5. P.3810.

79. Sen S., Koo J., Bagchi S. TRIFECTA: Security, Energy Efficiency, and Communication Capacity Comparison for Wireless IoT Devices. *IEEE Internet Computing*. 2018. Vol. 22, No. 1. P. 74-81.

80. Mbanaso U. M., Chukwudebe G. A. Requirement Analysis of IoT Security in Distributed Systems. *Proceedings of the 2017 IEEE 3rd International Confer- ence on Electro-Technology for National Development (NIGERCON)*. 2017. P. 777-781.

81. Singh R. Accurate Area Estimation Model for FPGA Based Implementation. *IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)*. 2016. Vol. 6, No. 4. P. 26-32.

82. CRYPTREC. URL: https://www.cryptrec.go.jp/en/ (Last accessed: 02.03.2023).

83. Dinu D., Corre Y. Triathlon of Lightweight Block Ciphers for the Internet of Things. *Journal of Cryptographic Engineering*. 2018.

ДОДАТОК А

Фрагмент лістингу коду програмного забезпечення реалізації методу

```
#include <iostream>
#include <string>
#include <windows.h>
#include <mysql.h>
#include <vector>
#include <sstream>
#include <string.h>
#include <tuple>
#include <conio.h>
#include <cmath>
#include <cstdlib>
#include <iomanip>
#include <stdlib.h>
#include<bits/stdc++.h>
#include <chrono>
#include <thread>
#include <fstream>
#include <ctime>
#include "sha256.h"
using std::string;
using std::cout;
using std::endl;

using namespace std;

MYSQL* conn;
MYSQL_ROW row;
MYSQL_RES *res;
int qstate;
int strength = 0;
bool authorized_to_be_Admin = false;
string status = "";

static const char alphanum[] =
"0123456789"
```

```
"!@$&%^*"
"ABCDEFGHIJKLMNOPQRSTUVWXYZ"
"abcdefghijklmnopqrstuvwxyz";

int stringLength = sizeof(alphanum) -1;


char genRandom()
{
return alphanum[rand() % stringLength];
}

std::string generateSalt()
{
srand(time(0));
std::string salti;

for(unsigned int i = 0; i < 20; ++i)
{
salti += genRandom();
}

return salti;
}

class Processing_and_Output
{
public:
Processing_and_Output();
int searchString(string s1, string s2);
bool isCapable(int cpu, double fms, double rs, double cs, int cpu_, double flash, double ram, double clock);
string concatStrings(string r1, string r2, string r3, string r4, string r5, string r6);
auto mapping1(string userR, string r_1, string r_2, string r_3, string r_4, string r_5, string r_6, string r1, string r2, string r3, string r4, string r5, string r6);
auto mapping1b(string userR, string r_1, string r_2, string r_3, string r_4, string r_5, string r_6, string r1, string r2, string r3, string r4, string r5, string r6);
int checkSensitivity_of_User_ApplcArea(string as);
int Check_if_streamCipherNeeded(string ps);
auto select_MarchingAlgo(int flagx, int ds, int n_SC, int cpu_int, double fms_doub, double rs_doub);
auto select_MarchingAlgo2_ThreeSteps(int flagx, int ds, int n_SC, double cctax, double tpx, string type);
auto select_MarchingAlgo2_FourSteps(int flagx, int ds, int n_SC, double cctax, double tpx, string type);
auto select_MarchingAlgo2_SixSteps(int flagx, int ds, int n_SC, double cctax, double tpx, string type);
auto select_MarchingAlgo2_SevenSteps(int flagx, int ds, int n_SC, double cctax, double tpx, string type);
```

```
auto mapping2(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, int cpu_int, double fms_doub, double rs_doub, string as, string ps);

auto mapping2Hardware(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, double ccta1_doub, double tp1_doub, double ccta2_doub, double tp2_doub, double ccta3_doub, double tp3_doub, double ccta4_doub, double tp4_doub, double ccta5_doub, double tp5_doub, double ccta6_doub, double tp6_doub, string as, string ps, string type);

string fetch_Algo(string IDx);

auto display_Mech_Algo_mapping(int algo1, int algo2, int algo3, int algo4, int algo5, int algo6, int ps1, int ps4, int ps6);

auto display_Mech_Algo_mapping2(int algo1, int algo2, int algo3, int algo4, int algo5, int algo6, int ps1, int ps4, int ps6, string energy);

void format_Print_output(string s1, string s2, string s3, string s4, string s5, string s6, int f1, int f2, int f3, int f4, int f5, int f6, string algo_name1, string algo_name2, string algo_name3, string algo_name4, string algo_name5, string algo_name6);

void format_TextOutput(string s1, string s2, string s3, string s4, string s5, string s6, int f1, int f2, int f3, int f4, int f5, int f6, string algo_name1, string algo_name2, string algo_name3, string algo_name4, string algo_name5, string algo_name6, double total_reqWeight, double fms, double rs, double fms_, double rs_, string request_id);

double reqWeighting(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6);

void displayReq_Mech_mapping(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, string algo_name1, string algo_name2, string algo_name3, string algo_name4, string algo_name5, string algo_name6, double total_reqWeight, int warn1, string request_id);

void write_Req_Mech_mapping(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, string algo_name1, string algo_name2, string algo_name3, string algo_name4, string algo_name5, string algo_name6, double total_reqWeight, double fms, double rs, double fms_, double rs_, string request_id);

auto select_MarchingAlgo3(int flagx, int ds, int n_SC, double FMx, double RAMx);

auto display_Mech_Algo_mapping3(string type, int algo1, int algo2, int algo3, int algo4, int algo5, int algo6, int ps1, int ps4, int ps6);

auto mapping2_3(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, double FM1_doub, double RAM1_doub, double FM2_doub, double RAM2_doub, double FM3_doub, double RAM3_doub, double FM4_doub, double RAM4_doub, double FM5_doub, double RAM5_doub, double FM6_doub, double RAM6_doub, string ad, string ps);

void displayReq_Mech_mapping3(int flag1, int flag2, int flag3, int flag4, int flag5, int flag6, string algo_name1, string algo_name2, string algo_name3, string algo_name4, string algo_name5, string algo_name6, string request_id);

void write_security_requirements_in_textFile(string Reqst_ID);

void write_bestPracticeGuide_in_textFile(string Reqst_ID);

};


Processing_and_Output::Processing_and_Output()

{


}


void Processing_and_Output::write_security_requirements_in_textFile(string Reqst_ID)

{

system("cls");

string findbyID_query = "select * from users_requests_re where Reqst_ID = " + ("'"+Reqst_ID+"'");

const char* qn = findbyID_query.c_str();

qstate = mysql_query(conn, qn);


string state, Domain, anyUsr, Login, stoUsrInfo, stoAnyInfo, InfoType, infoSent2E, connected, dataSent2Cloud, dataStoredInDb, update, use3rdPrtySfw, evesdrop, capt_Resent, impersontUsr, physiclAcces;
```

```
if(!qstate)
{
res = mysql_store_result(conn);
while((row = mysql_fetch_row(res)))
{
state = row[1];
Domain = row[2];
anyUsr = row[3];
Login = row[4];
stoUsrInfo = row[5];
stoAnyInfo = row[6];
InfoType = row[7];
infoSent2E = row[8];
connected = row[9];
dataSent2Cloud = row[10];
dataStoredInDb = row[11];
update = row[12];
use3rdPrtySfw = row[13];
evesdrop = row[14];
capt_Resent = row[15];
impersontUsr = row[16];
physiclAcces = row[17];
}
}
else
{
cout << "Query Execution Problems!" << mysql_errno(conn) << endl;
}

int domainSensitivity = checkSensitivity_of_User_ApplcArea(Domain);

string flag1 = "1", flag2 = "1", flag3 = "1", flag4 = "1", flag5 = "1", flag6 = "1", flag7 = "1", flag8 = "1", flag9 = "", flag10 = "1", flag11 = "1", flag12 = "1", flag13 = "1", flag14 = "1", flag15 = "1", flag16 = "1", flag17 = "1", flag18 = "1", flag19 = "1", flag20 = "1";

string flag21 = "1", flag22 = "1", flag23 = "1", flag24 = "1", flag25 = "1", flag26 = "1", flag27 = "1", flag28 = "1", flag29 = "1", flag30 = "1", flag6_, flag_6;

string flag31 = "1", flag32 = "1", flag33 = "1", flag34 = "1", flag35 = "1", flag36 = "1", flag37 = "1", flag38 = "1", flag39 = "1", flag40 = "1", flag41 = "1", flag42 = "1", flag43 = "1", flag44 = "1";

string Reqmt_1, Reqmt_2, Reqmt_3, Reqmt_4, Reqmt_5, Reqmt_6, Reqmt_7, Reqmt_8, Reqmt_9, Reqmt_10, Reqmt_11, Reqmt_12, Reqmt_13, Reqmt_14, Reqmt_15;
```

```
fstream file;

file.open("Security_Requirements.txt", ios::out | ios::trunc);

if(file.is_open())

{

file << "\n"
"**********************************************************************************************************
*********" << endl;

file << "\t THE SECURITY REQUIREMENTS FOR THE IoT SYSTMEM OF THE USER WITH REQUEST ID No.: "
<< Reqst_ID << endl << endl;


file << left << setw(21) << setfill('-') << left << '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;


file << setfill(' ') << '|' << left << setw(20) << "SECURITY REQUIREMENT"

<< setfill(' ') << '|' << left << setw(83) << "DESCRIPTION" << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;


if(anyUsr == "Yes" && Login == "Yes")//

{

flag1 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_3 = "AUTH";

}

if(stoUsrInfo == "Yes" && anyUsr == "Yes")

{

flag2 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Privacy"

<< setfill(' ') << '|' << left << setw(83) << "Refers to users control over the disclosure of their personal information,
menani- " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ng that only the users should decide whether they want to share their data or
not." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;
```

```
Reqmt_4 = "PRIV";

}

if(stoUsrInfo == "Yes" && stoAnyInfo == "Yes")

{

flag3 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Confidentiality"

<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that information is not disclosed or made availa-" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ble to any unauthorized entity." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_1 = "CONF";

}

if(stoAnyInfo == "Yes" && flag3 != "2")

{

flag4 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Confidentiality"

<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that information is not disclosed or made availa-" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ble to any unauthorized entity." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_1 = "CONF";

}

if(InfoType == "Normal" && flag2 != "2" && !stoUsrInfo.empty())

{

flag2 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Privacy"

<< setfill(' ') << '|' << left << setw(83) << "Refers to users control over the disclosure of their personal information, menani- " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ng that only the users should decide whether they want to share their data or not." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_4 = "PRIV";
```

```
}
if(InfoType == "Normal" && flag3 != "2" && flag4 != "2")
{
flag6 = "2";
file << setfill(' ') << '|' << left << setw(20) << "Confidentiality"
<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that information is not disclosed or made availa-" << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << "ble to any unauthorized entity." << '|' << endl;

file << left << setw(21) << setfill('-') << left<< '+'
<< setw(84) << setfill('-') << '+' << '+' << endl;
Reqmt_1 = "CONF";
}
if(InfoType == "Normal" || InfoType == "Sensitive" || InfoType == "Critical")
{
flag6_ = "2";
file << setfill(' ') << '|' << left << setw(20) << "Integrity"
<< setfill(' ') << '|' << left << setw(83) << "Is the property of safeguarding the correctness, consistency, and trustworthiness " << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << "of data over its entire life cycle in an IoT system." << '|' << endl;

file << left << setw(21) << setfill('-') << left<< '+'
<< setw(84) << setfill('-') << '+' << '+' << endl;
Reqmt_2 = "INTG";
}
if(InfoType == "Normal" || InfoType == "Sensitive" || InfoType == "Critical")
{
flag_6 = "2";
file << setfill(' ') << '|' << left << setw(20) << "Availability"
<< setfill(' ') << '|' << left << setw(83) << "Refers to the property which ensures that an IoT device or system is accessible and" << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << " usable upon demand by authorized entities." << '|' << endl;

file << left << setw(21) << setfill('-') << left<< '+'
<< setw(84) << setfill('-') << '+' << '+' << endl;
Reqmt_6 = "AVAI";
}

if(InfoType == "Normal" || InfoType == "Sensitive" || InfoType == "Critical")
```

```
{
flag7 = "2";
file << setfill(' ') << '|' << left << setw(20) << "Physical Security"
<< setfill(' ') << '|' << left << setw(83) << "Refers to the security measures designed to deny unauthorized physical access to " << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << "IoT devices or systems, and to protect them from damage or tampering." << '|' << endl;

file << left << setw(21) << setfill('-') << left<< '+'
<< setw(84) << setfill('-') << '+' << '+' << endl;
Reqmt_7 = "PHYS";
}
if(InfoType == "Sensitive" && stoUsrInfo == "Yes")
{

}
if(InfoType == "Sensitive" && stoUsrInfo != "Yes")
{

}
if(InfoType == "Sensitive" && flag3 != "2" && flag4 != "2" && flag6 != "2")
{
flag8 = "2";
file << setfill(' ') << '|' << left << setw(20) << "Confidentiality"
<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that information is not disclosed or made availa-" << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << "ble to any unauthorized entity." << '|' << endl;

file << left << setw(21) << setfill('-') << left<< '+'
<< setw(84) << setfill('-') << '+' << '+' << endl;
Reqmt_1 = "CONF";
}
if(InfoType == "Sensitive")
{
flag9 = "2";
file << setfill(' ') << '|' << left << setw(20) << "Authorization"
<< setfill(' ') << '|' << left << setw(83) << "Refers to the property that determines whether the user or device has rights/privi-" << '|' << endl;
file << setfill(' ') << '|' << left << setw(20) << " "
<< setfill(' ') << '|' << left << setw(83) << "leges to access a resource, or issue commands." << '|' << endl;
```

```
file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_8 = "AUTR";

}

if(InfoType == "Sensitive")

{

flag10 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Forgery Resistance"

<< setfill(' ') << '|' << left << setw(83) << "This is the propriety that ensures that the data shared between entities cannot be
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "forged by a third party trying to damage or harm the system or its users." << '|'
<< endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_9 = "FORE";

}

if(InfoType == "Sensitive" && flag1 != "2")

{

flag11 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


cout << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_3 = "AUTH";

}

if(InfoType == "Critical" && flag2 != "2" && flag5 != "2")

{

flag2 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Privacy"

<< setfill(' ') << '|' << left << setw(83) << "Refers to users control over the disclosure of their personal information,
menani- " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ng that only the users should decide whether they want to share their data or
not." << '|' << endl;
```

```
file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_4 = "PRIV";

}

if(InfoType == "Critical" && flag3 != "2" && flag4 != "2" && flag6 != "2" && flag8 != "2")

{

flag13 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Confidentiality"

<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that information is not disclosed or made availa-" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "ble to any unauthorized entity." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_1 = "CONF";

}

if(InfoType == "Critical" && flag7 != "2")

{

flag14 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Physical Security"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security measures designed to deny unauthorized physical access to " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "IoT devices or systems, and to protect them from damage or tampering." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_7 = "PHYS";

}

if(InfoType == "Critical" && flag9 != "2")

{

flag15 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authorization"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the property that determines whether the user or device has rights/privi-" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "leges to access a resource, or issue commands." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'
```

```
<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_8 = "AUTR";

}

if(InfoType == "Critical" && flag10 != "2")

{

flag16 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Forgery Resistance"

<< setfill(' ') << '|' << left << setw(83) << "This is the propriety that ensures that the data shared between entities cannot be
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "forged by a third party trying to damage or harm the system or its users." << '|'
<< endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_9 = "FORE";

}

if(InfoType == "Critical" && domainSensitivity == 1)

{

flag17 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Non-Repudiation"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security property that ensures that the transfer of messages or cred-
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "entials between 2 IoT entities is undeniable." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_5 = "NONR";

}

if(InfoType == "Critical" && flag1 != "2" && flag11 != "2")

{

flag18 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;
```

```
Reqmt_3 = "AUTH";

}

if(infoSent2E == "Yes" && flag17 != "2" && domainSensitivity == 1)

{

flag19 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Non-Repudiation"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security property that ensures that the transfer of messages or cred-
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "entials between 2 IoT entities is undeniable." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_5 = "NONR";

}

if(infoSent2E == "Yes" && flag1 != "2" && flag11 != "2" && flag18 != "2")

{

flag20 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_3 = "AUTH";

}

if(infoSent2E == "Yes")

{

flag21 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Confinement"

<< setfill(' ') << '|' << left << setw(83) << "Ensures that even if a party is corrupted, the spreading of the effects of the" <<
'|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "attack is as confined as possible." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_10 = "CFMT";

}
```

```
if(connected == "Yes" && flag17 != "2" && flag19 != "2" && domainSensitivity == 1)

{

flag22 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Non-Repudiation"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security property that ensures that the transfer of messages or cred-
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "entials between 2 IoT entities is undeniable." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_5 = "NONR";

}

if(connected == "Yes" && domainSensitivity == 1)

{

flag23 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Accountability"

<< setfill(' ') << '|' << left << setw(83) << "This is the property that ensures that every action can be traced back to a single
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "user or device." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_11 = "ACCT";

}

if(connected == "Yes")

{

flag24 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Reliability"

<< setfill(' ') << '|' << left << setw(83) << "Is the property that guarantees consistent intended behavior of an IoT system."
<< '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_12 = "RELI";

}

if(dataSent2Cloud == "Yes" && flag6_ != "2")

{

flag25 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Integrity"
```

```
<< setfill(' ') << '|' << left << setw(83) << "Is the property of safeguarding the correctness, consistency, and trustworthiness
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "of data over its entire life cycle in an IoT system." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_2 = "INTG";

}

if(dataSent2Cloud == "Yes" && flag_6 != "2")

{

flag26 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Availability"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the property which ensures that an IoT device or system is accessible
and" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << " usable upon demand by authorized entities." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_6 = "AVAI";

}

if(dataSent2Cloud == "Yes")

{

flag27 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Data Freshness"

<< setfill(' ') << '|' << left << setw(83) << "Ensures that data is the most recent, and that old messages cannot be replayed."
<< '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_13 = "DAFR";

}

if(dataSent2Cloud == "Yes" && flag10 != "2" && flag16 != "2")

{

flag28 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Forgery Resistance"

<< setfill(' ') << '|' << left << setw(83) << "This is the propriety that ensures that the data shared between entities cannot be
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "forged by a third party trying to damage or harm the system or its users." << '|'
<< endl;
```

```
file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_9 = "FORE";

}

if(dataSent2Cloud == "Yes" && flag17 != "2" && flag19 != "2" && flag22 != "2" && domainSensitivity == 1)

{

flag29 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Non-Repudiation"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security property that ensures that the transfer of messages or cred-
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "entials between 2 IoT entities is undeniable." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_5 = "NONR";

}

if(dataStoredInDb == "Yes" && flag7 != "2" && flag14 != "2" && flag22 != "2" )

{

flag30 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Physical Security"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security measures designed to deny unauthorized physical access
to " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "IoT devices or systems, and to protect them from damage or tampering." << '|'
<< endl;


file<< left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_7 = "PHYS";

}

if(dataStoredInDb == "Yes" && flag6_ != "2" && flag25 != "2")

{

flag31 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Integrity"

<< setfill(' ') << '|' << left << setw(83) << "Is the property of safeguarding the correctness, consistency, and trustworthiness
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "of data over its entire life cycle in an IoT system." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'
```

```
<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_2 = "INTG";

}

if(dataStoredInDb == "Yes" && flag_6 != "2" && flag26 != "2")

{

flag32 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Availability"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the property which ensures that an IoT device or system is accessible
and" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << " usable upon demand by authorized entities." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_6 = "AVAI";

}

if(dataStoredInDb == "Yes" && flag10 != "2" && flag16 != "2" && flag28 != "2")

{

flag33 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Forgery Resistance"

<< setfill(' ') << '|' << left << setw(83) << "This is the propriety that ensures that the data shared between entities cannot be
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "forged by a third party trying to damage or harm the system or its users." << '|'
<< endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_9 = "FORE";

}

if(dataStoredInDb == "Yes" && flag1 != "2" && flag11 != "2" && flag18 != "2" && flag20 != "2")

{

flag34 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;
```

```
Reqmt_3 = "AUTH";

}

if(dataStoredInDb == "Yes" && flag17 != "2" && flag19 != "2" && flag22 != "2" && flag29 != "2" &&
domainSensitivity == 1)

{

flag35 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Non-Repudiation"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security property that ensures that the transfer of messages or cred-
" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "entials between 2 IoT entities is undeniable." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_5 = "NONR";

}

if(update == "Yes" && flag_6 != "2" && flag26 != "2" && flag32 != "2")

{

flag36 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Availability"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the property which ensures that an IoT device or system is accessible
and" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << " usable upon demand by authorized entities." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_6 = "AVAI";

}


if(use3rdPrtySfw == "Yes" && flag21 != "2")

{

flag37 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Confinement"

<< setfill(' ') << '|' << left << setw(83) << "Ensures that even if a party is corrupted, the spreading of the effects of the" <<
'|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "attack is as confined as possible." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_10 = "CFMT";
```

```
}

if(evesdrop == "Yes" && flag9 != "2" && flag15 != "2" && (infoSent2E == "Yes" || dataSent2Cloud == "Yes" ||
dataStoredInDb == "Yes"))

{

flag39 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authorization"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the property that determines whether the user or device has
rights/privi-" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "leges to access a resource, or issue commands." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_8 = "AUTR";

}

if(capt_Resent == "Yes" && (infoSent2E == "Yes" || dataSent2Cloud == "Yes" || dataStoredInDb == "Yes") && flag1 !=
"2" && flag11 != "2" && flag18 != "2" && flag20 != "2" && flag34 != "2")

{

flag40 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to"
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_3 = "AUTH";

}

if(capt_Resent == "Yes" && flag27 != "2" && (infoSent2E == "Yes" || dataSent2Cloud == "Yes" || dataStoredInDb ==
"Yes"))

{

flag41 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Data Freshness"

<< setfill(' ') << '|' << left << setw(83) << "Ensures that data is the most recent, and that old messages cannot be replayed."
<< '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_13 = "DAFR";

}

// impersontUsr
```

```
if(impersontUsr == "Yes" && Login == "Yes" && flag1 != "2" && flag11 != "2" && flag18 != "2" && flag20 != "2" && flag34 != "2" && flag40 != "2")

{

flag42 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Authentication"

<< setfill(' ') << '|' << left << setw(83) << "This is the assurance that information transaction is from the source it claims to" << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "be from." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_3 = "AUTH";

}

if(physiclAcces == "Yes" && flag7 != "2" && flag14 != "2" && flag22 != "2" && flag30 != "2")

{

flag43 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Physical Security"

<< setfill(' ') << '|' << left << setw(83) << "Refers to the security measures designed to deny unauthorized physical access to " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "IoT devices or systems, and to protect them from damage or tampering." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_7 = "PHYS";

}

if(use3rdPrtySfw == "Yes")

{

file << setfill(' ') << '|' << left << setw(20) << "Counterfeit "

<< setfill(' ') << '|' << left << setw(83) << "Is the property that ensures effective validation of software such that any fake " << '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << "Resistance "

<< setfill(' ') << '|' << left << setw(83) << "or maliciously modified software is rejected." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_14 = "CNFR";

}

if(physiclAcces == "Yes")

{
```

```
flag44 = "2";

file << setfill(' ') << '|' << left << setw(20) << "Tamper Detection"

<< setfill(' ') << '|' << left << setw(83) << "Ensures all devices are physically secured, such that any tampering attempt is "
<< '|' << endl;

file << setfill(' ') << '|' << left << setw(20) << " "

<< setfill(' ') << '|' << left << setw(83) << "detected." << '|' << endl;


file << left << setw(21) << setfill('-') << left<< '+'

<< setw(84) << setfill('-') << '+' << '+' << endl;

Reqmt_15 = "TAMD";

}

flag1.erase(); flag2.erase(); flag3.erase(); flag4.erase(); flag5.erase();

state.erase(); Domain.erase(); anyUsr.erase(); Login.erase(); stoUsrInfo.erase(); stoAnyInfo.erase(); InfoType.erase();
infoSent2E.erase(); connected.erase(); dataSent2Cloud.erase(); dataStoredInDb.erase(); update.erase();
use3rdPrtySfw.erase(); evesdrop.erase(); capt_Resent.erase(); impersontUsr.erase(); physiclAcces.erase();


string request_query = "insert into generated_requirements (Reqst_ID, Reqmt_1, Reqmt_2, Reqmt_3, Reqmt_4, Reqmt_5,
Reqmt_6, Reqmt_7, Reqmt_8, Reqmt_9, Reqmt_10, Reqmt_11, Reqmt_12, Reqmt_13, Reqmt_14, Reqmt_15)
values('"+Reqst_ID+"', '"+Reqmt_1+"', '"+Reqmt_2+"', '"+Reqmt_3+"', '"+Reqmt_4+"', '"+Reqmt_5+"', '"+Reqmt_6+"',
'"+Reqmt_7+"', '"+Reqmt_8+"', '"+Reqmt_9+"', '"+Reqmt_10+"', '"+Reqmt_11+"', '"+Reqmt_12+"', '"+Reqmt_13+"',
'"+Reqmt_14+"', '"+Reqmt_15+"')";

const char* qr = request_query.c_str();

qstate = mysql_query(conn, qr);

if(!qstate)

{

Reqmt_1.erase(); Reqmt_2.erase(); Reqmt_3.erase(); Reqmt_4.erase(); Reqmt_5.erase(); Reqmt_6.erase();
Reqmt_7.erase(); Reqmt_8.erase(); Reqmt_9.erase(); Reqmt_10.erase(); Reqmt_11.erase(); Reqmt_12.erase();
Reqmt_13.erase(); Reqmt_14.erase(); Reqmt_15.erase();

}

file.close();

}

else

{

cout <<"\n\tFile failed to open!" << endl;

}

}
```

ДОДАТОК Б

Вхідні дані та результати експериментальних досліджень

Таблиця Б.1 – Вхідні дані для підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

ID	Етап життєвого циклу	Архітектура або призначення системи (SAoP) - Керування пристроями	SAoP - Збір даних	SAoP - Управління підключенням	Сервіси SAoP -API	SAoP - Управління даними	SAoP - Обробка даних	SAoP - Аналітика великих даних / Розширена аналітика	SAoP - центри обробки даних та хмарні сервіси	SAoP - Веб-сервіси/ Веб-додатки	SAoP - Вбудовані системи	SAoP - Інше
B1111	Розроблення	+	+									
B2345	Розроблення	+	+									
B4444	Існуюча		+									
B1234	Розроблення									+		
B6548	Розроблення						+	+				
B0601	Розроблення						+					
B7788	Існуюча				+					+		
B5432	Розроблення									+		
B2278	Розроблення	+				+	+					
B9128	Розроблення									+		
B6666	Розроблення	+	+					+				
B1115	Розроблення	+	+			+	+	+	+			
B1995	Розроблення	+										
B4321	Розроблення	+										
B1235	Розроблення						+					
B8374	Розроблення					+						
B4040	Розроблення	+										
B8789	Існуюча									+		
B1008	Розроблення	+										
B5555	Розроблення	+										
B1287	Розроблення									+		
B5461	Розроблення						+					
B0011	Існуюча	+										
B8126	Розроблення									+		

Таблиця Б.2 – Вхідні дані для підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

ID	Чи є у системи користувач	Чи буде реєстрація користувачів	Тип реєстрації користувача	Чи буде логін у користувача	Чи будуть користувачі надавати якісь дані	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Тип автентифікації	Чи буде система використовувати базу даних	Тип зберігання даних
B1111	Ні						Так	Не чутлива		Так	SQL
B2345	Так	Так	Користувачі	Так	Так	Так	Так	Не чутлива	Ім’я користувача - пароль	Так	SQL
B4444	Так	Ні			Так	Так	Ні	Чутлива	Ім’я користувача - пароль	Так	NoSQL
B1234	Так	Так	Адміністратор	Так	Так	Так	Так	Критична	Двофакторна авторизація	Так	SQL
B6548	Так	Так	Адміністратор	Так	Так	Так	Так	Чутлива	Ім’я користувача - пароль	Так	SQL
B0601	Так	Так	Користувачі	Так	Так	Так	Так	Не чутлива	Ім’я користувача - пароль	Так	SQL
B7788	Так	Так	Адміністратор	Так	Так	Так	Так	Критична	Двофакторна авторизація	Так	SQL
B5432	Ні						Так	Чутлива		Так	SQL
B2278	Так	Так	Адміністратор	Так	Так	Так	Так	Чутлива	Ім’я користувача - пароль	Так	Розподілене зберігання
B9128	Так	Так	Користувачі	Так	Ні	Так	Ні	Не чутлива	Соц. мережі – електронна пошта	Так	SQL
B6666	Ні						Так	Не чутлива		Так	SQL
B1115	Так	Так	Користувачі	Так	Ні	Так	Так	Не чутлива	Ім’я користувача - пароль	Так	SQL
B1995	Ні						Так	Не чутлива		Так	Локальне зберігання
B4321	Так	Ні							Ім’я користувача - пароль	Так	Локальне зберігання
B1235	Так	Так	Користувачі	Так	Так	Так	Так	Не чутлива	Ім’я користувача - пароль	Так	Розподілене зберігання
B8374	Так	Так	Користувачі	Так	Ні	Так	Ні	Чутлива	Мультифакторна авторизація	Так	SQL
B4040	Так	Ні			Так	Ні	Так	Не чутлива	Немає	Ні	
B8789	Так	Так	Користувачі	Так	Так	Ні	Так	Не чутлива	Ім’я користувача - пароль	Ні	
B1008	Так	Так	Користувачі	Так	Так	Так	Ні	Не чутлива	Немає	Так	SQL
B5555	Ні						Так	Не чутлива		Ні	
B1287	Так	Так	Користувачі	Так	Так	Ні	Ні		Немає	Ні	
B5461	Так	Так	Користувачі	Так	Так	Ні	Так	Чутлива	Ім’я користувача - пароль	Так	SQL
B0011	Так	Так	Користувачі	Так	Так	Ні	Так	Не чутлива	Ім’я користувача - пароль	Ні	
B8126	Так	Так	Користувачі	Так	Так	Ні	Так	Критична	Ім’я користувача - пароль	Так	SQL

Таблиця Б.3 – Вхідні дані для підсистеми генерації множини вимог безпеки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

ID	Тип бази даних	Мова(и) програмування, яка(і) буде(уть) використовуватися - C#	Мова(и) програмування, яка(і) буде(уть) використовуватися -C/C++	Мова(и) програмування, яка(і) буде(уть) використовуватися -Java	Мова(и) програмування, яка(і) буде(уть) використовуватися -JavaScript	Мова(и) програмування, яка(і) буде(уть) використовуватися -PHP	Мова(и) програмування, яка(і) буде(уть) використовуватися -Python	Мова(и) програмування, яка(і) буде(уть) використовуватися - Ruby	Мова(и) програмування, яка(і) буде(уть) використовуватися - Інше	Чи дозволить система завантажувати файли	Чи будуть вестися системні журнали
B1111	MySQL						Python			Ні	Так
B2345	SQLite						Python			Так	Так
B4444	SQL-Server				JavaSc		Python			Так	Так
B1234	MySQL		C/C++			PHP				Ні	Так
B6548	MySQL					PHP	Python			Так	Так
B0601	PostgreSQL						Python			Так	Так
B7788	SQLite	C#				PHP				Ні	Так
B5432	SQL-Server				JavaSc					Так	Так
B2278	SQL-Server		C/C++		JavaSc					Так	Ні
B9128	PostgreSQL		C/C++		JavaSc		Python			Так	Так
B6666	SQL-Server						Python			Ні	Так
B1115	MySQL						Python			Так	Так
B1995	MySQL		C/C++				Python			Ні	Так
B4321	SQL-Server		C/C++				Python			Ні	Так
B1235	SQL-Server	C#		Java						Так	Так
B8374	SQL-Server			Java						Ні	Так
B4040			C/C++							Ні	Ні
B8789				Java						Ні	Так
B1008	SQL-Server			Java						Так	Так
B5555	SQL-Server		C/C++							Ні	Так
B1287							Python			Ні	Так
B5461	MySQL						Python			Так	Ні
B0011			C/C++							Так	Ні
B8126	SQLite			Java	JavaSc					Так	Так

S/No	SECURITY BEST PRACTICES
1	Strong device authentication is necessary to ensure that connected devices can be trusted to be what they claim to be. Hence, where applicable adopt strong password authentication; where possible, implement two factor authentication. Ensure secure boot, use tamper-resistant hardware-based storage like TPM, ensure that each stage of boot code is trusted before running it, and ensure that no boot sequence is skipped. Ensure that devices are shipped with latest and stable versions of OSes, and with proper OS security configuration. Also ensured that OSes can boot securely; use good password management techniques. Do not use the same keys when implementing encryption on many IoT devices; bake security into every stage of smart apps development lifecycle; and disable every port and interfaces that were installed on IoT devices for testing purposes.
2	Ensure that any newline characters in system log files are appropriately handled to prevent log forging; and ensure that any logged HTML characters are appropriately encoded to prevent XSS when viewing logs.
3	Do not use algorithms and protocols that are not vetted by the cryptographic community; ensure that certificates are properly validated against the hostnames whom they are meant for. To reduce the risk of compromising many servers, avoid using wildcard certificates unless there is a business need for it. Store only the sensitive data that you need. If a password is being used to protect keys then the password strength should be sufficient for the strength of the keys it is protecting. Use cryptographically strong random numbers for cryptographic parameters like keys.
4	Ensure that IoT and IIoT data both in-flight and at-rest is encrypted, and be very careful when selecting and implementing cryptographic algorithms because a cryptographic algorithm is only as strong as how it is implemented. Avoid using insecure protocols such as File Transfer Protocol (FTP) and Telnet because of lack of encryption, and their reliance on clear-text usernames and passwords for authentication.
5	Ensure that smart devices that will be deployed in open environments are securely protected using strong casing. If possible, protect IoT device circuitry from tampering using resin encapsulation and epoxy resin, etc.

Рисунок Б.1 – Перелік практичних рекомендацій для запиту B5555

Таблиця Б.4 – Результати тестування підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

№ зп.	Ідентифікаційний номер запиту.	Вимоги до безпеки користувачів	Механізми безпеки	Алгоритми безпеки
1.	H1598	- Конфіденційність даних - Аутентифікація - Конфіденційність та автентичність	Шифрування MAC-адреси Шифрування з автентифікацією	*No matching algo found! *No matching algo found! ACORN
2.	S8888	- Конфіденційність даних/Приватність користувачів - Цілісність повідомлень - Аутентифікація - Неприйняття відмови	Шифрування Хеш-функція MAC Цифровий підпис	Clefia128/192 PHOTON-256/32/32 SipHash-128 *No matching algo found!
3.	S3833	-Цілісність повідомлень - Неприйняття відмови - Конфіденційність та автентичність	Хеш-функція Цифровий підпис Шифрування з автентифікацією	PHOTON-128/16/16 *No matching algo found! CLOC-AES
4.	S6868	- Конфіденційність даних/Приватність користувачів - Цілісність повідомлень - Аутентифікація - Неприйняття відмови	Шифрування Хеш-функція MAC Цифровий підпис	Clefia128/192 PHOTON-256/32/32 SipHash-128 *No matching algo found!
5.	S6789	- Конфіденційність даних/Приватність користувача - Неприйняття відмови	Шифрування Цифровий підпис	ChaCha20-256 *No matching algo found!

Продовження таблиці Б.4 – Результати тестування підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

№ зп	Ідентифікаційний номер запиту.	- Вимоги до безпеки користувачів	Механізми безпеки	Алгоритми безпеки
6.	H3456	- Конфіденційність даних/приватність користувачів - Конфіденційність та автентичність	Шифрування Шифрування з автентифікацією	Grain_v1-128 ACORN
7.	S6000	- Цілісність повідомлень - Аутентифікація	Хеш-функція ГДК	PHOTON-128/16/16 *No matching algo found!
8.	S4219	- Конфіденційність даних/Приватність користувача - Цілісність повідомлень - Аутентифікація - Конфіденційність та автентичність	Шифрування Хеш-функція MAC Шифрування з автентифікацією	SPECK64/96 PHOTON-80/20/16 *No matching algo found! CLOC-TWINE
9.	H8791	- Цілісність повідомлень - Конфіденційність та автентичність	Хеш-функція Шифрування з автентифікацією	Keccak-f[100] Deoxys
10.	S4452	- Цілісність повідомлень - Неприйняття відмови - Конфіденційність та автентичність	Хеш-функція Цифровий підпис Шифрування з автентифікацією	PHOTON-128/16/16 *No matching algo found! CLOC-AES
11.	H2648	- Конфіденційність даних - Цілісність повідомлень - Аутентифікація - Конфіденційність та автентичність	Шифрування Хеш-функція MAC Шифрування з автентифікацією	*No matching algo found! *No matching algo found! *No matching algo found! Ascon
12.	S1682	- Конфіденційність даних - Цілісність повідомлень - Аутентифікація - Конфіденційність та автентичність	Шифрування Хеш-функція MAC Шифрування з автентифікацією	SPECK64/128 PHOTON-128/16/16 *No matching algo found! CLOC-AES
13.	S6001	- Цілісність повідомлень	Хеш-функція	PHOTON-80/20/16
14.	H9850	- Конфіденційність даних - Цілісність повідомлень - Аутентифікація - Конфіденційність та автентичність	Шифрування Хеш-функція MAC Шифрування з автентифікацією	SIMON64/96 SPONGENT-128/128/8 *No matching algo found! SILC-AES
15.	H5942	- Цілісність повідомлень - Конфіденційність та автентичність	Хеш-функція Шифрування з автентифікацією	PHOTON-80/20/16 Deoxys
16.	S0001	- Конфіденційність даних/Приватність користувача - Цілісність повідомлень	Шифрування Хеш-функція	Clefia128/256 PHOTON-224/32/32
17.	S7788	- Конфіденційність даних/Приватність користувачів - Цілісність повідомлень	Шифрування Хеш-функція	SPECK64/128 PHOTON-128/16/16

*No matching algo found! означає: Не знайдено алгоритму, що відповідає вимогам безпеки

Таблиця Б.5 – Запит даних, що зберігаються в базі даних, підсистемою формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

ID	Тип апаратного забезпечення	CPU	Розмір флеш пам'яті	Розмір RAM	Тактова частота	Галузь застосування	Розмір корисного навантаження	Вимога 1	Вимога 2	Вимога 3	Вимога 4	Вимога5	Вимога 6
S0001	SBC	32	8000000	4096000	2000	Розумний будинок	Невеликий	Конф.	Ціл.		Конф.		
S1682	ARM	32	1024	96.00	72	Розумне місто	Невеликий	Конф.	Ціл.	Авт.			Авт.+ конф.
S3833	ELEC	32	1024	128	120	Охорона здоров'я	Середній		Ціл.			Невідм.	Авт. + конф.
S4219	RL78	16	64	4	16	Роздрібна торгівля	Невеликий	Конф.	Ціл.	Авт.	Конф.		Авт. + конф.
S4452	TENS	32	16384	520	240	Розумна мережа	Невеликий		Ціл.			Невідм.	Авт. + конф.
S6000	AVR	8	136	6	16	Автомобіль	Невеликий		Ціл.	Авт.			
S6001	AVR	8	128	4	16	Домашній улюбленець	Невеликий		Ціл.				
S6789	SBC	64	32000000	2048000	900	Розумний будинок	Безперервний	Конф.			Конф.	Невідм.	
S6868	SBC	64	8388608	1048576	900	Розумне сільське господарство	Невеликий	Конф.	Ціл.	Авт.	Конф.	Невідм.	
S7788	PIC	32	256	1024	16	Розумний будинок	Невеликий	Конф.	Ціл.		Конф.		
S8888	SBC	64	8388608	1048576	1200	Фінанси	Середній	Конф.	Ціл.	Авт.	Конф.	Невідм.	

Таблиця Б.6 – Запит даних, що зберігаються в базі даних, підсистемою формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей для апаратної реалізації криптографічних алгоритмів

ID	Тип апаратного забезпечення	Галузь застосування	Розмір корисного навантаження	Енергоспоживання	Вимога 1	Площа ланцюга для першої вимоги	Пропускна здатність для першої вимоги безпеки	Вимога 2	Площа ланцюга для другої вимоги	Пропускна здатність для другої вимоги безпеки
H1598	FPGA	Розумний будинок	Невеликий	Наднизьке	Конф.	1022	100		0	0
H2648	FPGA	Охорона здоров'я	Невеликий	Наднизьке	Конф.	578	230	Ціл.	734	158
H3456	ASIC	Автомобіль	Безперервний	Низьке	Конф.	1450	200		0	0
H5942	ASIC	Розумне сільське господарство	Середній	Низьке		0	0	Ціл.	757	1
H8791	ASIC	Розумний будинок	Невеликий	Низьке		0	0	Ціл.	1254	2.3
H9850	ASIC	Розумний будинок	Невеликий	Низьке	Конф.	825	5.1	Ціл.	1398	16

Таблиця Б.7 – Запит даних, що зберігаються в базі даних, підсистемою формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей для апаратної реалізації криптографічних алгоритмів

ID	Вимога 3	Площа ланцюга для третьої вимоги	Пропускна здатність для третьої вимоги безпеки	Вимога 4	Площа ланцюга для четвертої вимоги	Пропускна здатність для четвертої вимоги безпеки	Вимога 5	Площа ланцюга для п'ятої вимоги	Пропускна здатність для п'ятої вимоги безпеки	Вимога 6	Площа ланцюга для шостої вимоги	Пропускна здатність для шостої вимоги безпеки
H1598	Авт.	270	3.5		0	0		0	0	Авт. + конф.	137	240
H2648	Авт.	356	280		0	0		0	0	Авт. + конф.	415	310
H3456		0	0	Конф.	1500	200		0	0	Авт. + конф.	3150	4.3
H5942		0	0		0	0		0	0	Авт. + конф.	2879	4.9
H8791		0	0		0	0		0	0	Авт. + конф.	2863	4.9
H9850	Авт.	748	4.3		0	0		0	0	Авт. + конф.	3200	5.8

Рисунок Б.2 – Результат для програмної реалізації криптографічних алгоритмів для запиту S4219

```
YOUR SECURITY REQUIREMENTS AND RECOMMENDED SECURITY MECHANISMS AND SECURITY ALGORITHMS ARE:

+-------------------------------+-------------------------+-----------------------+
|SECURITY REQUIREMENT(S)        |SECURITY MECHANISM(S)    |SECURITY ALGORITHM(S)  |
+-------------------------------+-------------------------+-----------------------+
|Data Confidentiality/User Privacy|Encryption             |Clefia128/192          |
+-------------------------------+-------------------------+-----------------------+
|Message Integrity              |Hash Function            |PHOTON-256/32/32       |
+-------------------------------+-------------------------+-----------------------+
|Authentication                 |Message Authentication Code|SipHash-128          |
+-------------------------------+-------------------------+-----------------------+
|Non-repudiation                |Digital Signature        |*No matching Algo found!|
+-------------------------------+-------------------------+-----------------------+

*No algorithm matching the security requirement is found!

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2

Press Enter to return to MAIN MENU
```

Рисунок Б.3 – Результат для програмної реалізації криптографічних алгоритмів для запиту S8888

```
YOUR SECURITY REQUIREMENTS AND RECOMMENDED SECURITY MECHANISMS AND SECURITY ALGORITHMS ARE:

+-------------------------------+-------------------------+-----------------------+
|SECURITY REQUIREMENT(S)        |SECURITY MECHANISM(S)    |SECURITY ALGORITHM(S)  |
+-------------------------------+-------------------------+-----------------------+
|Data Confidentiality           |Encryption               |*No matching Algo found!|
+-------------------------------+-------------------------+-----------------------+
|Authentication                 |Message Authentication Code|*No matching Algo found!|
+-------------------------------+-------------------------+-----------------------+
|Confidentiality & Authenticity |Authenticated Encryption |ACORN                  |
+-------------------------------+-------------------------+-----------------------+

*No algorithm matching the security requirement is found!

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2
```

Рисунок Б.4 – Результат для апаратної реалізації криптографічних алгоритмів для запиту H1598

Рисунок Б.5 – Результат для апаратної реалізації криптографічних алгоритмів для запиту H3456

ДОДАТОК В

Копія публікації у виданні, що індексується в наукометричній базі Scopus

The 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
22-25 September, 2021, Cracow, Poland

The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining

Dmytro Denysiuk [1], Kira Bobrovnikova [1], Sergii Lysenko [1], Oleg Savenko [1], Piotr Gaj [2], Roman Havryliuk[1], Yaroslav Boiko[1]

[1] Khmelnytskyi National University, Institutska str., 11, Khmelnytskyi, 29016, Ukraine,
web.developer.den@gmail.com, bobrovnikova.kira@gmail.com, sirogyk@ukr.net, savenko_oleg_st@ukr.net,
http://ki.khnu.km.ua/

[2] Silesian University of Technology, Akademicka str. 2A, Gliwice, 44-100, Poland,
piotr.gaj@polsl.pl

***Abstract* – In recent years, the area of IoT malware detection has been at the forefront of the security community. Due to lack the latest cybersecurity requirements in IoT devices design, lack of security updates and transparency of security posture, as well as due to the IoT devices specific characteristics, such as heterogeneity of the processor architecture, unsafe deployment of IoT devices, as well as exponential growth in the number of IoT devices - all this leads to an increase in the number of malware targeting IoT devices.**

The paper presents the new approach for IOT malware detection based on opcodes (operation codes) sequences pattern mining. The proposed approach uses mining of maximal sequential patterns of operation codes from assembly representation of IoT binary executable. Most distinctive of maximal sequential patterns are selected and for each of them estimations of their relevance are performed. Based on these relevance the feature vectors are built which described assembly representation of IoT binary executable. As a classifier Fuzzy Semi-Supervised C-Means classifier was applied.

The experimental results shows that the proposed approach allows detecting IOT malware with height effectiveness, with accuracy 99,51% and F_1-score at level 99,50%.

Keywords — Internet of Things; malware; detection of malicious software; sequential pattern mining; opcodes analysis

I. INTRODUCTION

Today, the Internet of Things has become an important part of modern society and has enormous potential, providing new services needed in everyday life. Every year, new types of devices appear on the IoT device market. The main weakness of these devices is the low level of protection against malicious software and cyberattacks.

Smart devices infected with malicious software can be used to steal a user's personal data or become a source of cyberattacks on other devices on the same network or outside it. A compromised smart device can also affect other critical components in the same network, such as database servers and Intranet, by being able to collect data and monitor other components in the IoT network [1, 2].

The projected rapid growth in the number of IoT devices [3] and the lack of basic monitoring and protection mechanisms for them will continue to contribute to the development and spread of malicious software in the IoT infrastructure.

II. RELATED WORKS

Today a number of works on the detection of IoT malware based on the analysis of operating codes are known.

In the study [4] for IoT malware detection and categorization fast fuzzy and fuzzy pattern tree approaches were applied. With this aim IoT programs operation codes were transmute into a vector space.

In the paper [5] a technique based on opcode N-grams analysis to classifying IoT malware was developed. With aim to detection IoT malware TF-IDF, Term frequency-Inverse document frequency for each of N-grams for IoT program is calculated. The feature vectors from obtained TF values for IoT programs are constructed. The machine-learning methods were used to carry out IoT malware classification.

In [6] an approach for malware detection in infrastructure of Internet Of Battlefield Things. This deep learning based approach applies class-wise selection of operation codes sequences as a signs for future classification. These operation codes are transformed into a vector space. Further for each IoT application a graph of selected features was build. To classify IoT benign applications and IoT malware a deep Eigen space learning technique was used.

In the work [7] combining machine learning techniques with the sequential pattern mining algorithm to find most frequent operation codes sequences of IoT malware were applied.

In the paper [8] a multi-view learning approach that applies multiple views such as operation codes, bytecodes, API calls, header information, permission and attacker's intent for IoT malware detection. The proposed approach presupposes automatically definition different weights to

the set of assembly representation of malicious IoT binary executables;

$|\{e_i \in E \mid MP \in e_i\}|$ – the number of the assembly representation of IoT binary executables e, in which appears certain maximal sequential pattern MP.

Next, for each of features in the feature vectors the order is determined according to the ascending values of the $IDF(MP,E)$:

$$\Theta = \{IDF(MP_i,E)\}_{i=1}^{N_{MP}},\ IDF(MP_i,E) < IDF(MP_{i+1},E), \quad (5)$$

where N_{MP} – the total number of different MP.

At the next step the relevance value for each MP estimated as weighted term frequency WTF [23]. WTF is calculated as the result of weighting the term frequency, TF, with the relevance of each operation code o when calculating TF, and is defined as the product of sequence frequency and the weight of every operation code o in MP:

$$WTF(MP,e) = TF(MP,e) \times \prod_{o \in MP} \frac{W(o)}{100}, \quad (2)$$

where $W(o)$ – the weight, by means of mutual information gain, for the operation code o;

$TF(MP,e)$ – the MP frequency measure within the assembly representation of IoT binary executable e.

Term frequency, $TF(MP,e)$, assessed the importance of a MP within an assembly representation of IoT binary executable e and can be calculated as following:

$$TF(MP,e) = \frac{f_{MP,e}}{\sum_{MP' \in e} f_{MP',e}}, \quad (3)$$

where $f_{MP,e}$ – the number of times of the maximal sequential pattern MP appears in the assembly representation of IoT binary executable e;

$\sum_{MP' \in e} f_{MP',e}$ – the total number of different MP in the assembly representation of IoT binary executable e.

Calculation the weight for the operation code o, $W(o)$, is based on the concept of Mutual Information. The Mutual Information $I(F;\Upsilon)$ is measure of the statistical dependence of certain two variables. In our case these variables are the opcode o and whether or not the IoT binary executable was malicious program:

$$I(F;\Upsilon) = \sum_{\gamma \in \Upsilon} \sum_{f \in F} p(f,\gamma) \log\left(\frac{p(f,\gamma)}{p(f) \times p(\gamma)}\right), \quad (4)$$

where F – the frequency of operation code;

Υ – the class of the IoT binary executables (benign or malware);

$p(f,\gamma)$ – the joint probability distribution function of F and Υ;

$p(f)$ – the marginal probability distribution function of F;

$p(\gamma)$ – the marginal probability distribution function of Υ.

Thus, the relevance values of MP are used as features of the feature vector which describes assembly representation of IoT binary executable based on maximal sequential patterns of operation codes. In case where the MP was not presented in the assembly representation of IoT binary executable, then the corresponding feature takes the value 0. Let us denote feature vector as:

$$\overline{V_{d,e}} = (r_i)_{i=1}^{N_{MP_e}}, \quad (5)$$

where $d \in D$ – the IoT device in the IoT network, $D = \{d_i\}_{i=1}^{N_d}$ – the set of IoT devices in the IoT network;

N_d – the amount number of IoT devices in the network;

r_i – the relevance value of MP;

N_{MP_e} – the number of MP in the assembly representation of IoT binary executable e.

B. Building Labelled and Unlabeled Data Matrix of Feature Vectors

From the constructed feature vectors $\overline{V_{d,e}}$ two matrix, a labelled data matrix M_l (with feature vectors labelled as "benign" or "malicious" respectively) and unlabeled data matrix M_{unl} (with unlabeled feature vectors) were built. These feature vectors $\overline{V_{d,e}}$, whose lengths are greater than the median length L of the rest all feature vectors in the data matrix M_{unl} and M_l, are truncated. Feature vectors, whose lengths are less than L, are padded with zeros. Further the labelled data matrix M_l is used as training data, and unlabeled data matrix M_{unl} is used as testing data.

The scheme of the process of constructing the feature vectors based on maximal sequential patterns of operation codes and building labelled and unlabeled data matrix presented in Fig. 1.

C. Data Classification

As a classifier in proposed technique the Semi-Supervised Fuzzy C-Means classifier was used. The benefit of applying the fuzzy clustering is the ability to reduce the requirements for the unambiguous correspondence of the clustering object to a specific

cluster. Fuzzy clustering apply functions of membership clustering objects to the fuzzy clusters which take values in the interval $[0,1]$. It allows increasing the information completeness of the results of clustering in cases where clustering object is located at the boundaries of the different clusters.

The distinctive particularity and main advantage of semi-supervised learning is the ability of identification of the initial centers of clusters. This feature improves the quality of clustering results. At the same time, to determine the initial centers of clusters, a training sample is required, the volume of which does not exceed 10% of the data collected for analysis.

Instead of the Euclidean metric, which is applied in the basic algorithm, it was decided to choose the Mahalanobis distance. The use of the Mahalanobis distance allows to take into account the presence of outliers, or observations, that stand out from the general sample in the classified data. This is possible due to the formation of clusters in the form of hyperellipsoids, whose axes can be oriented in different directions. Thus, a fuzzy partition matrix P is the result of clustering, and each element p_{ij} of the matrix P determines the belonging degree of the i -th clustering objects to the j -th cluster: $P=\left[p_{ij}\right], p_{ij}\in[0,1], \quad i=\overline{1,N_{\overline{V_{d,e}}}}, j=\overline{1,N_{\Upsilon}}$, $\sum_{j=1,N_{\Upsilon}} p_{ij}=1$, where $N_{\overline{V_{d,e}}}$ – the amount number of the feature vectors, N_{Υ} – the amount number of the clusters. So, the applying of fuzzy clustering implies that each feature vector $\overline{V_{d,e}}$ with a certain affiliation degree will be referred to each of the N_{Υ} clusters, which denotes malicious or benign IoT binary executable.

Let us denote the set of clusters as $\Upsilon=\Upsilon_b\cup\Upsilon_m$, where Υ_b is a cluster that correspond to benign class (benign IoT binary executables) and Υ_m is a cluster that correspond to malicious class (malicious IoT binary executables).

Let's take λ as the value of threshold that determines whether clustering object belonging to the certain cluster, at which the clustering object is considered as benign or malicious. If $p_{ij}\geq\lambda$, then the clustering object belongs to a j cluster, $j\in\{b,m\}$.

The scheme of the proposed approach for IoT malware detection based on opcodes sequences pattern mining presented in Fig. 2.

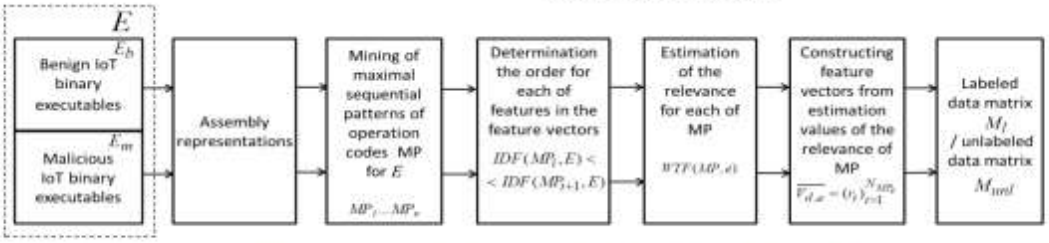


Figure 1. Constructing the feature vectors based on maximal sequential patterns of operation codes and building labelled and unlabeled data matrix

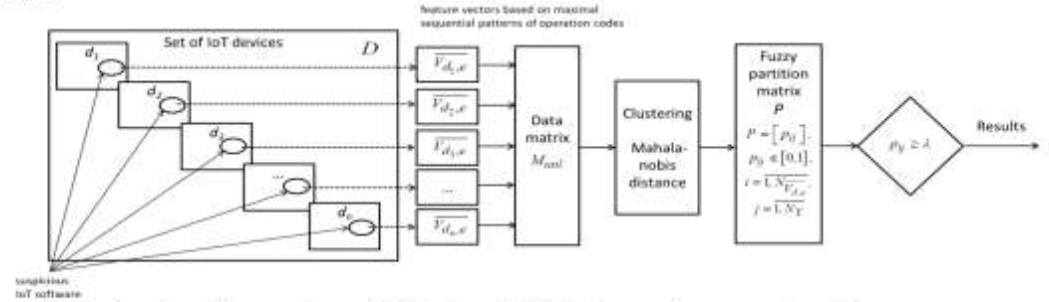


Figure 2. The scheme of the proposed approach for IoT malware detection based on opcodes sequences pattern mining

IV. Experimental Results

In order to evaluate the effectiveness of the proposed approach, a number of experiments were conducted. To conduct the experiment, it was necessary to choose a target IoT platform. For this purpose ARM platform was chosen as it is one of the most widespread IoT platforms.

In this research we have presented the new approach for IOT malware detection based on operation codes sequences pattern mining. Concerning the experimental results, we have used two sets of samples: [24] resource present the set of programs for IoT devices (benign programs for our experiments), while [25] presents the resource with a huge amount of the malware including malware for IoT devices. At this stage of research, the main purpose of the work was not to form the representative of software, but to verify the possibility of developed technique to detect IoT malware.

Thus 314 samples of benign ARM-based IoT software samples from [24] for such IoT devices as camcorders, smart TVs and routers and 250 ARM-based IoT malware samples from [25] (such as Mirai, Stuxnet, Dark Nexus, Gafgyt, Hajime and other) have been used. In addition from used malware 51 polymorphic malware samples were generated by applying the open-source obfuscation tool [26].

According to the proposed approach, the samples of IoT malicious and benign binary executable were processed and assembly representations from these samples were extracted. With the purpose of disassembling software samples the Interactive Disassembler (IDA Pro) [27] was used To mining of maximal sequential patterns of operation codes for the all samples of obtained assembly representation hash-based partition sequential pattern mining algorithm [28] was applied.

About 20% of obtained data were used for training, and about 80% data were used as testing data to get an assessment of the effectiveness of the proposed approach. As classifiers following machine learning methods were applied [29, 30]: Random Forest [31], K-Means [32], C-Means [33], Semi-Supervised Fuzzy C-Means [34] and Support Vector Machine [35].

With aim to evaluate the effectiveness of the proposed technique, two statistical metrics were applied: F_1 score and accuracy.

Accuracy provides a statistical estimation of proportion of correctly predictions among the total number of cases studied:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5)$$

where TP – correctly classified samples of IoT malware, true positive;

TN – correctly classified samples of IoT benign programs, true negative;

FN – samples of IoT malware, falsely classified as benign program, false negative;

FP – samples of benign IoT programs, falsely classified as malware, false positive.

F_1 score (also named as balanced F-score or F-measure) is another measure which was applied for estimation of experiments accuracy. These score is determined as the harmonic mean of recall and precision:

$$F_1 = 2 \times \frac{PREC \times REC}{PREC + REC}, \quad (6)$$

where $PREC$ – precision or positive predictive value, which is defined as part of relevant samples among the classified samples, $PREC = \frac{TP}{TP + FP}$;

REC – recall or sensitivity, which is defined as part of relevant samples that were classified, $REC = \frac{TP}{TP + FN}$.

The results of experiments, which demonstrated values of accuracy and F1 score for proposed approach for IoT malware detection based on operation codes sequences pattern mining presented in Table 1.

As shown in the Table 1, the highest efficiency was reached by applying Semi-Supervised Fuzzy C-Means as classifier.

TABLE I. EXPERIMENTAL RESULTS: ACCURACY AND F1 SCORE VALUES OF APPROACH FOR IOT MALWARE DETECTION BASED ON OPCODES SEQUENCES PATTERN MINING

Classifier	*TP*	*TN*	*FN*	*FP*	*ACC*	*F1*
Random Forest	280	300	21	14	94,31	94,12
K-Means	283	299	18	15	94,63	94,49
C-Means	292	307	9	7	97,40	97,33
Support Vector Machine	292	309	9	5	97,72	97,66
Semi-Supervised Fuzzy C-Means	299	313	2	1	99,51	99,50

V. CONCLUSIONS

Thus, the new approach for IOT malware detection based on operation codes sequences pattern mining was proposed. The proposed approach uses mining of maximal sequential patterns of operation codes from assembly representation of IoT binary executable. Most distinctive of maximal sequential patterns are selected based on inverse document frequency values and for each of these patterns estimations of their relevance are performed. To assess the relevance, the weighted term frequency was used. The feature vectors are built based on obtained relevance for selected maximal sequential patterns. These vectors allow describing assembly

representation of IoT binary executable take into account most distinctive of maximal sequential patterns. As a classifier Semi-Supervised Fuzzy C-Means classifier was used.

Thus, when implementing the proposed approach, it is important to take into account that different IoT platforms have their own specifics. Therefore, for the experiments conduction, attention was paid to the ARM platform as one of the most widespread IoT platforms.

The results of experiments shows that the proposed approach allows detecting IOT malware with height effectiveness, with accuracy 99,51% and F_1-score at level 99,50%.

REFERENCES

[1] Trend Micro. Inside the Smart Home: IoT Device Threats and Attack Scenarios. URL: https://www.trendmicro.com/vinfo/it/security/news/internet-of-things/inside-the-smart-home-iot-device-threats-and-attack-scenarios.

[2] McAfee Labs Threats Report. URL: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-nov-2020.pdf

[3] Global System for Mobile Communications. URL: https://www.gsma.com/

[4] E. M. Dovom, A. Azmoodeh, A. Dehghantanha, D. E. Newton, R. M. Parizi, H. Karimipour, "Fuzzy pattern tree for edge malware detection and categorization in IoT". *Journal of Systems Architecture*, 2019, pp. 97, 1-7.

[5] H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, A. K. Sangaiah, "Classification of ransomware families with machine learning based on N-gram of opcodes". *Future Generation Computer Systems*, 2019, Vol. 90, pp. 211-221.

[6] A. Azmoodeh, A. Dehghantanha, K. K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning". *IEEE transactions on sustainable computing*, 2018, Vol. 4 (1), pp. 88-95.

[7] H. Darabian, A. Dehghantanha, S. Hashemi, S. Homayoun, K. K. R. Choo, "An opcode-based technique for polymorphic Internet of Things malware detection. Concurrency and Computation": *Practice and Experience*, 2020, Vol. 32 (6), e5173.

[8] H. Darabian, A. Dehghantanha, S. Hashemi, M. Taheri, A. Azmoodeh, S. Homayoun, R. M. Parizi, "A multiview learning method for malware threat hunting: windows, IoT and android as case studies". *World Wide Web*, 2020, Vol. 23 (2), pp. 1241-1260.

[9] F. Manavi, A. Hamzeh, "A new approach for malware detection based on evolutionary algorithm". *In Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2019, pp. 1619-1624.

[10] S. Jeon, J. Moon, "Malware-detection method with a convolutional recurrent neural network using opcode sequences". *Information Sciences*, 2020, pp. 535, 1-15.

[11] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, Z. Qin, "MTHAEL: Cross-Architecture IoT Malware Detection Based on Neural Network Advanced Ensemble Learning". *IEEE Transactions on Computers*, 2020, 69(11), pp. 1654-1667.

[12] G. Radhakrishnan, K. Srinivasan, S. Maheswaran, K. Mohanasundaram, D. Palanikkumar, A. Vidyarthi, "A deep-RNN and meta-heuristic feature selection approach for IoT malware detection". *Materials Today: Proceedings*, 2021.

[13] R. Lu, "Malware detection with lstm using opcode language". arXiv preprint arXiv:1906.04593, 2019.

[14] S. Bezobrazov, A. Sachenko, M. Komar, & V. Rubanau, "The methods of artificial intelligence for malicious applications detection in Android OS". *International Journal of Computing*. 15(3), 2016, pp. 184-190. https://doi.org/10.47839/ijc.15.3.851

[15] I. Obeidat, & M. AlZubi, "Developing a faster pattern matching algorithms for intrusion detection system". *International Journal of Computing*, 18(3), 2019, pp. 278-284. https://doi.org/10.47839/ijc.18.3.1520

[16] G. Markowsky, O. Savenko, A. Sachenko, "Distributed Malware Detection System Based on Decentralized Architecture in Local Area Networks", *Advances in Intelligent Systems and Computing*, 2019, 871, pp. 582-598.

[17] M. Drozd, A. Drozd, "Safety-Related Instrumentation and Control Systems and a Problem of the Hidden Faults" *The 10th International Conference on Digital Technologies 2014*, Zhilina, Slovak Republic, 2014, pp. 137–140. DOI: 10.1109/DT.2014.6868692

[18] A. Cabri, G. Suchacka, S. Rovetta and F. Masulli, "Online Web Bot Detection Using a Sequential Classification Approach," 2018.

[19] S. Ustebay, Z. Turgut and M.A. Aydin, "Intrusion Detection System with Recursive Feature Elimination by Using Random Forest and Deep Learning Classifier", *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, Ankara, Turkey, 2018, pp. 71-76.

[20] T. Sochor, M. Zuzcak, "Attractiveness Study of Honeypots and Honeynets in Internet Threat Detection". *Communications in Computer and Information Science*, Springer, Cham, 2015, pp. 69-81.

[21] S.Lysenko, K.Bobrovnikova, R.Shchuka, O.Savenko,"A Cyberattacks Detection Technique Based on Evolutionary Algorithms". *In 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies*, 2020, pp. 127-132.

[22] S. Lysenko, K. Bobrovnikova, O. Savenko, R. Shchuka, "Technique for Cyberattacks Detection Based on DNS Traffic Analysis", *CEUR-WS*, Vol 2732. ISSN: 1613–0073, 2020. pp. 171-182.

[23] I. Santos, F. Brezo, X. Ugarte-Pedrero, P. G. Bringas, "Opcode sequences as representation of executables for data-mining-based unknown malware detection". *Information Sciences*, 2013, Vol. 231, pp. 64-82.

[24] Packages Search for Linux and Unix. URL: https://pkgs.org/

[25] VirusTotal. URL: http://www.virustotal.com

[26] Obfuscatoin-for-ARM-disassembled-binary. URL: https://github.com/darabian/Obfuscatoin-for-ARM-disassembled-binary

[27] Hex Rays. IDA Pro. URL: https://www.hex-rays.com/products/ida/

[28] R. Millham, I. E. Agbehadji, H. Yang, "Pattern Mining Algorithms". *Bio-inspired Algorithms for Data Streaming and Visualization, Big Data Management, and Fog Computing*. Springer, Singapore, 2021, pp. 67-80.

[29] S. Nakhodchi, A. Upadhyay, A. Dehghantanha, "A comparison between different machine learning models for IoT malware detection". *Security of Cyber-Physical Systems*. Springer, Cham, 2020, pp. 195-202.

[30] W. Peters, A. Dehghantanha, R. M. Parizi, & G. Srivastava, "A comparison of state-of-the-art machine learning models for OpCode-based IoT malware detection". *In Handbook of Big Data Privacy*, 2020, pp. 109-120. Springer, Cham.

[31] Khammas, B. M., "Ransomware Detection Using Random Forest Technique". *ICT Express*, 6(4), 2020, pp. 325-331.

[32] Q. Wang, L. Li, B. Jiang, Z. Lu, J. Liu, & S. Jian, "Malicious domain detection based on k-means and smote". *In International Conference on Computational Science*, 2020, pp. 468-481. Springer, Cham.

[33] I. Hafeez, M. Antikainen, A. Y. Ding, & S. Tarkoma, "IoT-KEEPER: Detecting malicious IoT network activity using online traffic analysis at the edge". *IEEE Transactions on Network and Service Management*, 2020, 17(1), pp. 45-59.

[34] S. Lysenko, O. Savenko, K. Bobrovnikova. "DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering". *CEUR-WS*, ISSN: 1613–0073, 2018. Vol. 2104, pp. 688-695.

[35] S. Lysenko, K. Bobrovnikova, O. Savenko, A. Kryshchuk "BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks". In: Gaj P., Sawicki M., Kwiecień A. (eds) *Computer Networks. CN 2019.*

ДОДАТОК Г

Довідка про прийняття публікації до друку

Довідка: ВХНУ ТН 16/05/23

Видання: Вісник Хмельницького національного університету. Технічні науки

Категорія фаховості видання: фахове видання України, у якому можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук та ступеня доктора філософії, категорії «Б» філософії, категорії «Б» (наказ МОН №1643 від 28.12.2019, наказ МОН №409 від 17.03.2020).

Напрям – технічні науки за спеціальностями – 101, 121, 122, 123, 124, 125, 141, 151, 161, 172, 181, 182 (28.12.2019), спеціальності – 131, 132, 133 (17.03.2020)

Назва статті: ВИЯВЛЕННЯ КІБЕРАТАК В ІНФРАСТРУКТУРІ ІНТЕРНЕТУ РЕЧЕЙ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Автори: Бобровнікова К. , Гурман І., Попов Ю., Бойчук Я., Качур В. (Хмельницький національний університет)

Номер, у який прийнято статтю: №3, до друку рекомендовано буде до 30 червня 2023 року.

16.05.2023

Начальника відділу
інтелектуальної власності та трансферу технологій Ю.В.Кравчик

ДОДАТОК Д

Презентація до дипломної роботи

Задачі дослідження

1. Провести огляд відомих стандартів та рішень синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

3. Удосконалити модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей на основі врахування стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів.

4. Розробити метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей.

5. Провести експериментальні дослідження для перевірки ефективності розробленого методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Наукова новизна

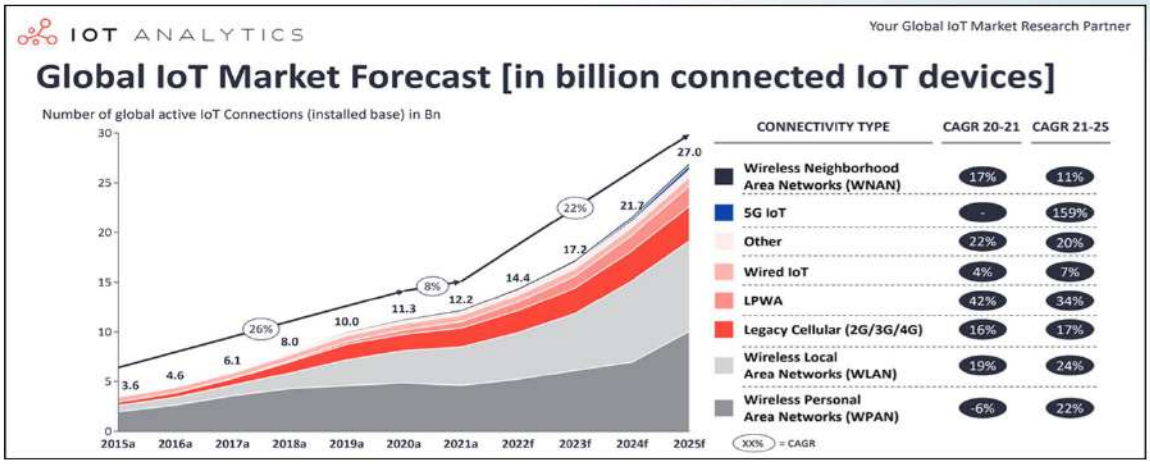
1. Удосконалено модель синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, яка, на відміну від відомих моделей, заснована на врахуванні стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів;

2. Удосконалено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, який, на відміну від відомих, заснований на запропонованій моделі та ґрунтується на комплексному врахуванні вимог безпеки до апаратно-програмних засобів і криптографічних алгоритмів Інтернету речей. Застосування розробленого методу дозволить синтезувати апаратно-програмні засоби підвищення безпеки інфраструктури Інтернету речей, в порівнянні з відомими методами.

Практична цінність

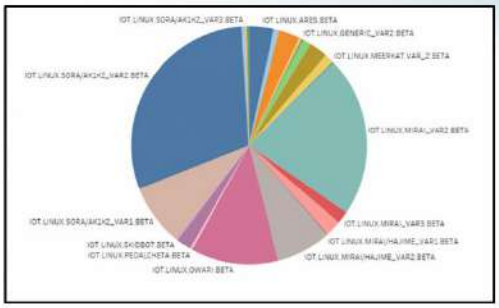
Практична цінність дипломної роботи полягає в розробленні методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, що надасть можливість синтезувати пристрої Інтернету речей з врахуванням стандартів та вимог безпеки до апаратно-програмних засобів Інтернету речей, а також вимог до апаратно-програмних криптографічних алгоритмів Інтернету речей.

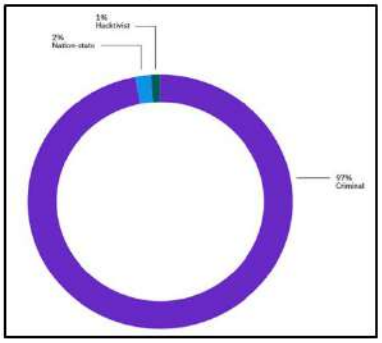
Актуальність роботи

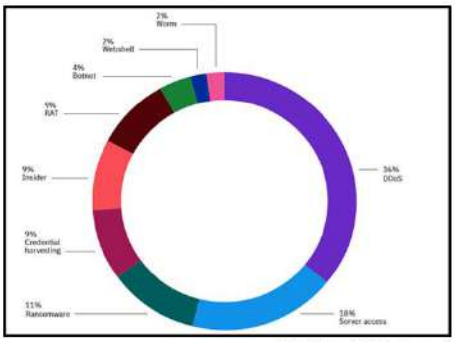


https://iot-analytics.com/

Актуальність роботи







https://www.bitdefender.com/

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Множина загроз безпеки апаратно-програмних засобів Інтернету речей:

$$T = \langle T_{L1}, T_{L2}, T_{L3} \rangle,$$

де T_{L1} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття; $T_{L1} = \{t_{L1}\}_{i=1}^{N}$, де N – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття; $t_{L1,1}$ – несанкціонований доступ до апаратно-програмних засобів Інтернету речей, $t_{L1,2}$ – прослуховування, $t_{L1,3}$ – радіочастотні перешкоди, $t_{L1,4}$ – клонування тегів, $t_{L1,5}$ – спуфінг, $t_{L1,6}$ – DoS, тощо;

T_{L2} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на мережному рівні; $T_{L2} = \{t_{L2}\}_{i=1}^{K}$, де K – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на мережному рівні; $t_{L2,1}$ – вичерпання мережних ресурсів, $t_{L2,2}$ – спуфінг, $t_{L2,3}$ – глушіння сигналів, $t_{L2,4}$ – sinkhole, $t_{L2,5}$ – підробка, $t_{L2,6}$ – атаки MitM, $t_{L2,7}$ – перехоплення сесії, $t_{L2,8}$ – DoS і DDoS, тощо;

T_{L3} – множина загроз безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні; $T_{L3} = \{t_{L3}\}_{i=1}^{M}$, де M – загальна кількість загроз безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні; $t_{L3,1}$ – DoS і DDoS, $t_{L3,2}$ – впровадження зловмисного коду, $t_{L3,3}$ – перехоплення, $t_{L3,4}$ – фішинг, $t_{L3,5}$ – соціальна інженерія, тощо.

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Вимоги безпеки апаратно-програмних засобів Інтернету речей з врахуванням відомих стандартів в галузі Інтернету речей:

$$R = \langle R_{L1}, R_{L2}, R_{L3} \rangle,$$

де R_{L1} – множина вимог безпеки апаратно-програмних засобів Інтернету речей на рівні сприйняття;

R_{L2} – множина вимог безпеки апаратно-програмних засобів Інтернету речей на мережному рівні;

R_{L3} – множина вимог безпеки апаратно-програмних засобів Інтернету речей на прикладному рівні.

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Множина вимог безпеки до апаратно-програмних засобів інфраструктури Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, R_d, $R_d \in R$:

$$R_d = \langle R_{SH}, R_{SN}, R_{SS}, R_{ST}, R_{SHP}, R_{SM}, R_{SC}, R_{SW}, R_{SF} \rangle,$$

де R_{SH} – множину вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» будинку;

R_{SN} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» мереж;

R_{SS} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» міста;

R_{ST} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» транспорту;

R_{SHP} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумної» охорони здоров'я;

R_{SM} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» виробництва;

R_{SC} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» ланцюгів постачання;

R_{SW} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумних» носимих пристроїв;

R_{SF} – множина вимог безпеки, які повинні бути гарантованими для підвищення безпеки апаратно-програмних засобів «розумного» сільського господарства.

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

$$M = \langle T, R_d, R_s, C, \Xi, H, A, \omega \rangle,$$

де T – множина загроз безпеки апаратно-програмних засобів Інтернету речей;

R_d – множина вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей;

R_s – правила синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей;

C – правила вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей;

Ξ – множина функцій, які можуть бути застосовані для досягнення вимог безпеки апаратно-програмних засобів Інтернету речей;

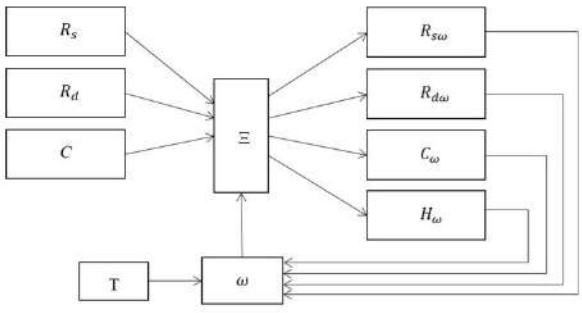
H – множина апаратних платформ для побудови апаратно-програмних засобів інфраструктури Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей;

ω – проєктований апаратно-програмний пристрій Інтернету речей.

Модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Функція досягнення вимог безпеки апаратно-програмних засобів Інтернету речей:

$$\Xi(\omega, T, R_d, R_s, C) \longrightarrow R_{d\omega}, R_{s\omega}, C_\omega, H_\omega,$$



де $R_{d\omega}$ – множина вимог безпеки до апаратно-програмного засобу Інтернету речей ω з врахуванням галузі застосування та відомих стандартів безпеки в галузі Інтернету речей;

$R_{s\omega}$ – правила синтезу апаратно-програмного засобу ω на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей;

C_ω – правила вибору та впровадження полегшених криптографічних алгоритмів для ω;

H_ω – апаратна платформа для побудови ω з врахуванням галузі застосування.

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

1. Формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей.

2. Формування правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки апаратно-програмних засобів Інтернету речей.

3. Формування правил вибору та впровадження множини полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей.

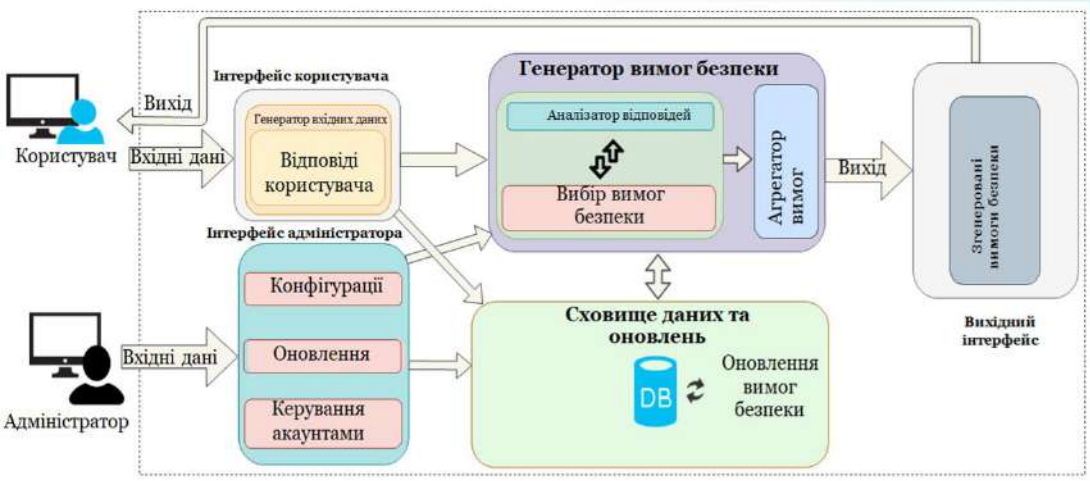
4. Визначення вимог до апаратно-програмного засобу Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та апаратної платформи.

5. Синтез апаратно-програмного засобу Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів.

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Укрупнена схема процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

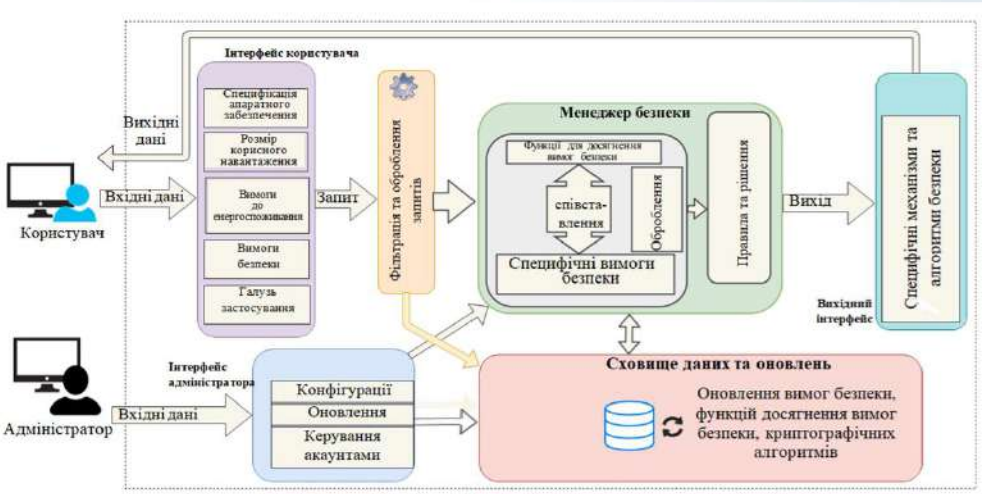


Архітектура підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Архітектура підсистеми генерації множини вимог безпки на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

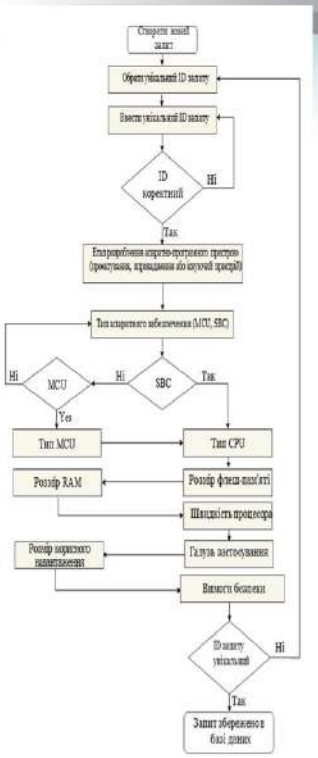


Архітектура підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей

Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Приклади правил, які застосовуються до запитів стосовно програмної реалізації криптографічних алгоритмів:

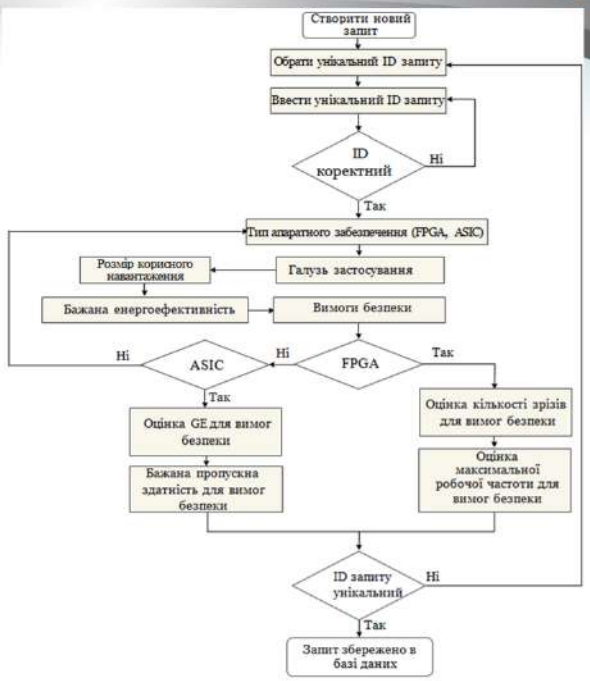
1. ЯКЩО ємність оперативної пам'яті дуже велика І флеш-пам'ять дуже велика, ТО апаратне забезпечення є дуже потужним;
2. ЯКЩО обсяг оперативної пам'яті достатній І флеш-пам'ять достатня, ТО апаратне забезпечення є потужним;
3. ЯКЩО тип корисного навантаження повідомлення малий АБО середній АБО великий І апаратне забезпечення дуже потужне АБО потужне, ТО вибрати відповідний блоковий шифр;
4. ЯКЩО тип корисного навантаження повідомлення є безперервним АБО невідомим І апаратне забезпечення дуже потужне АБО потужне, ТО вибрати відповідний потоковий шифр;
5. ЯКЩО галузь застосування є чутливою І апаратне забезпечення є дуже потужним, ТО вибрати найбільш безпечний алгоритм із великим розміром блоку та/або великим розміром ключа;
6. ЯКЩО галузь застосування є чутливою І апаратне забезпечення потужне, ТО вибрати безпечний алгоритм із помірним розміром блоку та/або помірним розміром ключа;
7. ЯКЩО апаратне забезпечення дуже обмежене, ТО вибрати дуже легкий алгоритм, який відповідає або майже відповідає цій умові;
8. ЯКЩО апаратне забезпечення обмежене, ТО вибрати легкий алгоритм, який відповідає цій умові.



Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Приклади правил, які застосовуються до запитів стосовно апаратної реалізації криптографічних алгоритмів:

1. ЯКЩО тип корисного навантаження повідомлення малий АБО середній АБО великий, ТО вибрати відповідний блоковий шифр;
2. ЯКЩО тип корисного навантаження повідомлення неперервний АБО невідомий, ТО вибрати відповідний потоковий шифр;
3. ЯКЩО площа ланцюга невелика, а пропускна здатність висока, ТО вибрати алгоритм, який відповідає або майже відповідає цим умовам;
4. ЯКЩО площа ланцюга не надто мала, а пропускна здатність помірна, ТО вибрати алгоритм, який відповідає цим умовам;
5. ЯКЩО галузь застосування є чутливою, ТО вибрати безпечний алгоритм з помірним розміром блоку та/або помірним розміром ключа;
6. ЯКЩО вимоги до енергоспоживання є низьке енергоспоживання, ТО вибрати енергоефективний алгоритм;
7. ЯКЩО вимоги до енергоспоживання є наднизьке енергоспоживання, ТО вибрати дуже енергоефективний алгоритм.



Експериментальні дослідження

Таблиця 1 - Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувачі	Чи буде (або чи є) потім у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати або надсилає інформацію іншим суб'єктам
R1111	Розроблення	Розумне місто	Ні			Так	Не чутлива	Так
R2345	Розроблення	Іграшка	Так	Так	Так	Так	Не чутлива	Так
R4444	Існуюча	Розумний будинок	Так	Так	Так	Ні	Чутлива	Так
R1234	Розроблення	Носимий пристрій	Так	Так	Так	Так	Не чутлива	Так
R6548	Розроблення	Домашній улюбленець	Так	Так	Так	Так	Не чутлива	Ні
R0601	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Не чутлива	Так

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувачі	Чи буде (або чи є) потім у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати або надсилає інформацію іншим суб'єктам
R7788	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Чутлива	Ні
R5432	Розроблення	Розумна мережа	Так	Так	Так	Так	Чутлива	Так
R2278	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Чутлива	Так
R9128	Розроблення	Навколишнє середовище	Ні			Так	Критична	Так
R6666	Розроблення	Штучний інтелект	Ні			Так	Не чутлива	Ні
R1115	Розроблення	Сільське господарство	Так	Так	Так	Так	Не чутлива	Так
R1995	Розроблення	Іграшка	Так	Так	Ні	Так	Не чутлива	Так
R4321	Розроблення	Екологія	Так	Ні	Так	Ні	Чутлива	Так
R1235	Розроблення	Роздрібна торгівля	Так	Так	Ні	Так	Не чутлива	Так
R8374	Розроблення	Охорона здоров'я	Так	Так	Так	Ні	Чутлива	Так

ID	Етап життєвого циклу	Галузь застосування	Чи є у системи користувачі	Чи буде (або чи є) потім у користувача	Чи буде система зберігати інформацію про користувача	Чи зберігає система якусь іншу інформацію	Чутливість інформації, що зберігається	Чи буде система надсилати або надсилає інформацію іншим суб'єктам
R4040	Розроблення	Розумна мережа	Так	Так	Ні	Так	Не чутлива	Ні
R8789	Розроблення	Розумне місто	Так	Так	Так	Так	Критична	Так
R1008	Розроблення	Люди похилого віку	Так	Так	Так	Так	Критична	Так
R5555	Розроблення	Автомобіль	Так	Так	Так	Так	Не чутлива	Ні
R1287	Розроблення	Охорона здоров'я	Так	Так	Так	Так	Критична	Так
R5461	Розроблення	Розумна мережа	Так	Так	Ні	Ні		
R0011	Існуюча	Розумний будинок	Ні			Так	Критична	Так
R8126	Розроблення	Розумне місто	Так	Так	Так	Так	Чутлива	Так

Експериментальні дослідження

ID	Чи підключена (чи буде підключена) система до Інтернету	Чи буде система надсилати (або надсилає) дані на будь-яку хмарну платформу	Чи буде система зберігати (або зберігає) дані в базі даних	Чи буде система отримувати (або отримує) регулярні оновлення	Чи буде система використовувати (або використовує) стороннє програмне забезпечення	Чи існує ймовірність атак на підслуховування	Чи існує ймовірність атак повторного відтворення	Чи існує ймовірність того, що зловмисники видають себе за користувача	Чи існує ймовірність того, що зловмисник може мати фізичний доступ до системи
R1111	Так	Так	Так	Так	Ні	Так	Так		Так
R2345	Так	Так	Так	Так	Так	Так	Так	Так	Так
R4444	Так	Так	Так	Так	Так	Так	Так	Ні	Так
R1234	Так	Так	Так	Так	Ні	Ні	Ні	Так	Так
R6548	Так	Так	Так	Так	Так	Так	Так	Так	Так
R0601	Так	Ні	Так	Так	Так	Так	Так	Ні	Ні
R7788	Так	Так	Так	Ні	Ні	Ні	Ні	Так	Так
R5432	Так	Ні	Ні	Так	Ні	Так	Ні	Так	Ні
R2278	Так	Так	Так	Ні	Ні	Ні	Так	Так	Так
R9128	Так	Так	Так	Так	Так	Ні	Ні		Так
R6666	Так	Ні	Так	Ні	Так	Ні	Ні		Так
R1115	Так	Так	Ні	Так	Ні	Так	Так	Так	Так
R1995	Так	Ні	Ні	Ні	Ні	Так	Ні	Ні	Так
R4321	Так	Так	Так	Так	Ні	Ні	Так	Так	Ні
R1235	Так	Так	Так	Так	Так	Так	Так	Ні	Ні
R8374	Так	Так	Так	Ні	Ні	Ні	Ні	Так	Ні

Таблиця 2 - Вхідні дані для підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей

ID	Чи підключена (чи буде підключена) система до Інтернету	Чи буде система надсилати (або надсилає) дані на будь-яку хмарну платформу	Чи буде система зберігати (або зберігає) дані в базі даних	Чи буде система отримувати (або отримує) регулярні оновлення	Чи буде система використовувати (або використовує) стороннє програмне забезпечення	Чи існує ймовірність атак на підслуховування	Чи існує ймовірність атак повторного відтворення	Чи існує ймовірність того, що зловмисники видають себе за користувача	Чи існує ймовірність того, що зловмисник може мати фізичний доступ до системи
R4040	Так	Ні	Ні	Ні	Ні	Так	Так	Так	Ні
R8789	Так	Так	Ні	Ні	Так	Так	Ні	Ні	Ні
R1008	Так	Ні	Ні	Так	Так	Ні	Ні	Так	Ні
R5555	Так	Так	Ні	Ні	Ні	Ні	Ні	Так	Так
R1287	Так	Ні	Так	Так	Так	Так	Так	Так	Так
R5461	Так	Ні	Так	Так	Так	Так	Так	Так	Так
R0011	Так	Ні	Так	Так	Ні	Ні	Ні		Так
R8126	Так	Ні	Ні	Ні	Ні	Так	Так	Так	Так

Експериментальні дослідження

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Physical Security	Refers to the security measures designed to deny unauthorized physical access to IoT devices or systems, and to protect them from damage or tampering.

Перелік вимог безпеки для запиту R1995

Експериментальні дослідження

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Privacy	Refers to users control over the disclosure of their personal information, meaning that only the users should decide whether they want to share their data or not.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Authorization	Refers to the property that determines whether the user or device has rights/privileges to access a resource, or issue commands.
Forgery Resistance	This is the propriety that ensures that data shared between entities and updates cannot be forged by a third party trying to damage or harm the system or its users.
Non-Repudiation	Refers to the security property that ensures that the transfer of messages or credentials between 2 IoT entities is undeniable.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Accountability	This is the property that ensures that every action can be traced back to a single user or device.
Reliability	Is the property that guarantees consistent intended behavior of an IoT system.

Перелік вимог безпеки для запиту R5432

Експериментальні дослідження

SECURITY REQUIREMENT	DESCRIPTION
Authentication	This is the assurance that a message is from the source it claims to be from.
Privacy	Refers to users control over the disclosure of their personal information, meaning that only the users should decide whether they want to share their data or not.
Confidentiality	This is the property that ensures that information is not disclosed or made available to any unauthorized entity.
Integrity	Is the property of safeguarding the correctness, consistency, and trustworthiness of data over its entire life cycle in an IoT system.
Availability	Refers to the property which ensures that an IoT device or system is accessible and usable upon demand by authorized entities.
Physical Security	Refers to the security measures designed to deny unauthorized physical access to IoT devices or systems, and to protect them from damage or tampering.
Authorization	Refers to the property that determines whether the user or device has rights/privileges to access a resource, or issue commands.
Forgery Resistance	This is the propriety that ensures that data shared between entities and updates cannot be forged by a third party trying to damage or harm the system or its users.
Non-Repudiation	Refers to the security property that ensures that the transfer of messages or credentials between 2 IoT entities is undeniable.
Confinement	Ensures that even if an entity is hijacked or corrupted, the spreading of the effects of the attack is as confined as possible.
Accountability	This is the property that ensures that every action can be traced back to a single user or device.
Reliability	Is the property that guarantees consistent intended behavior of an IoT system.
Counterfeit Resistance	Is the property that ensures effective validation of software such that any fake or maliciously modified software is rejected.
Data Freshness	Ensures that data is the most recent, and that old messages cannot be replayed.
Tamper Detection	Ensures all devices are physically secured, such that any tampering attempt is detected.

Перелік вимог безпеки для запиту R1287

Експериментальні дослідження

З метою проведення експериментів для підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей було використано наступні сценарії тестування:

- сценарій тестування програмної реалізації з використанням MCU;
- сценарій тестування програмної реалізації з SBC;
- сценарій тестування апаратної реалізації з використанням ASIC;
- сценарій тестування апаратної реалізації з ПЛІС;
- тестовий сценарій, який демонструє імпорт вимог безпеки з підсистеми формування множини вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей до підсистеми формування вимог стосовно застосування полегшених криптографічних алгоритмів для апаратно-програмних засобів Інтернету речей.

25

Експериментальні дослідження

```
+----------------------------------+---------------------------+----------------------+
|Confidentiality & Authenticity    |Authenticated Encryption   |CLOC-TWINE            |
+----------------------------------+---------------------------+----------------------+

*No algorithm matching the security requirement is found!

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2

WARNING! LIMITED RESOURCES
Implementing all algorithms may have negative impact on performance.

Press Enter to return to MAIN MENU
```

Результат для програмної реалізації криптографічних алгоритмів для запиту S4219

Експериментальні дослідження

```
YOUR SECURITY REQUIREMENTS AND RECOMMENDED SECURITY MECHANISMS AND SECURITY ALGORITHMS ARE:

+-------------------------------+-------------------------------+-------------------------------+
|SECURITY REQUIREMENT(S)        |SECURITY MECHANISM(S)          |SECURITY ALGORITHM(S)          |
+-------------------------------+-------------------------------+-------------------------------+
|Data Confidentiality/User Privacy|Encryption                   |Clefia128/192                  |
+-------------------------------+-------------------------------+-------------------------------+
|Message Integrity              |Hash Function                  |PHOTON-256/32/32               |
+-------------------------------+-------------------------------+-------------------------------+
|Authentication                 |Message Authentication Code    |SipHash-128                    |
+-------------------------------+-------------------------------+-------------------------------+
|Non-repudiation                |Digital Signature              |*No matching Algo found!       |
+-------------------------------+-------------------------------+-------------------------------+

*No algorithm matching the security requirement is found!

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2

Press Enter to return to MAIN MENU
```

Результат для програмної реалізації криптографічних алгоритмів для запиту S8888

Експериментальні дослідження

```
YOUR SECURITY REQUIREMENTS AND RECOMMENDED SECURITY MECHANISMS AND SECURITY ALGORITHMS ARE:

+-------------------------------+-------------------------------+-------------------------------+
|SECURITY REQUIREMENT(S)        |SECURITY MECHANISM(S)          |SECURITY ALGORITHM(S)          |
+-------------------------------+-------------------------------+-------------------------------+
|Data Confidentiality           |Encryption                     |*No matching Algo found!       |
+-------------------------------+-------------------------------+-------------------------------+
|Authentication                 |Message Authentication Code    |*No matching Algo found!       |
+-------------------------------+-------------------------------+-------------------------------+
|Confidentiality & Authenticity |Authenticated Encryption       |ACORN                          |
+-------------------------------+-------------------------------+-------------------------------+

*No algorithm matching the security requirement is found!

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2
```

Результат для апаратної реалізації криптографічних алгоритмів для запиту H1598

Експериментальні дослідження

```
YOUR SECURITY REQUIREMENTS AND RECOMMENDED SECURITY MECHANISMS AND SECURITY ALGORITHMS ARE:

+-------------------------------+-------------------------------+-------------------------------+
|SECURITY REQUIREMENT(S)        |SECURITY MECHANISM(S)          |SECURITY ALGORITHM(S)          |
+-------------------------------+-------------------------------+-------------------------------+
|Data Confidentiality/User Privacy|Encryption                   |Grain_v1-128                   |
+-------------------------------+-------------------------------+-------------------------------+
|Confidentiality & Authenticity |Authenticated Encryption       |ACORN                          |
+-------------------------------+-------------------------------+-------------------------------+

Would you like to see a detailed report on the recommended algorithms?:

1. Yes
2. No, thank you

Select Your Option (1-2): 2

Press Enter to return to MAIN MENU
```

Результат для апаратної реалізації криптографічних алгоритмів для запиту H3456

Публікації

1. Опубліковано статтю у матеріалах конференції, що індексуються у наукометричних базах Scopus та Web of Science:

D. Denysiuk, K. Bobrovnikova, S. Lysenko, O. Savenko, P. Gaj, R. Havryliuk, Y. Boichuk. The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining // Proceedings of the 2021 IEEE 11th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), IDAACS'2021, Cracow, Poland, September 22-25, 2021, 6 p.

2. Прийнято до публікації у науковому фаховому виданні «Вісник Хмельницького національного університету» (Технічні науки):

Бобровнікова К., Гурман І., Попов Ю., Бойчук Я., Качур В. «Виявлення кібератак в інфраструктурі Інтернету речей на основі машинного навчання».

Висновки

- В дипломній роботі за результатами виконаних теоретичних та практичних досліджень було розроблено метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.
- У першому розділі була досліджена предметна область, досліджені відомі стандарти та рішення синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.
- У другому розділі удосконалено модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. На відміну від відомих моделей, запропонована модель враховує множину вимог безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей та відомих стандартів безпеки в галузі Інтернету речей, а також грунтується на правилах синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки до апаратно-програмних засобів Інтернету речей та використовує множину правил впровадження полегшених криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей. Запропонована модель також враховує апаратну пптаформу апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування. Запропонована модель є основою методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.

Висновки

- Третій розділ представляє метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Запропонований метод надає можливість синтезувати апаратно-програмні засоби Інтернету речей на основі визначених вимог з врахуванням галузі застосування та експертних знань стосовно: (1) відомих стандартів безпеки; (2) правил синтезу апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки; (3) вимог з вибору та впровадження полегшених програмних та апаратних криптографічних алгоритмів інфраструктури Інтернету речей.
- У четвертому розділі представлено програмну реалізацію та алгоритми методу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей.
- Також, було проведено експериментальні дослідження, які показали результативність розробленого методу. Таким чином, застосування розробленого методу надає можливість синтезувати апаратно-програмні засоби Інтернету речей з врахуванням важливих вимог безпеки, що надасть можливість підвищити безпеку інфраструктури Інтернету речей.
- За темою дипломної роботи опубліковано статтю «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus, а також прийнято до публікації статтю «Виявлення кібератак в інфраструктурі Інтернету речей на основі машинного навчання». у науковому фаховому виданні «Вісник Хмельницького національного університету» (Технічні науки).

Дякую за увагу!

UNICHECK

Ім'я користувача:
Кафедра КІ

ID перевірки:
1015137315

Дата перевірки:
18.05.2023 09:00:19 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
18.05.2023 09:02:25 EEST

ID користувача:
100005591

Назва документа: Бойчук_Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури...

Кількість сторінок: 107 Кількість слів: 22166 Кількість символів: 168618 Розмір файлу: 1.26 MB ID файлу: 1014818411

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

3.22%
Схожість

Найбільша схожість: 0.88% з Інтернет-джерелом (https://neon-152.ru/gallery/%D0%92%D0%94-201%D0%90%D0%94%20(.

2.67% Джерела з Інтернету	318	Сторінка 109
1.35% Джерела з Бібліотеки	115	Сторінка 111

0.22% Цитат

Цитати	15	Сторінка 112
Посилання	1	Сторінка 112

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	14
Підозріле форматування	20 сторінок

Thu May 18 08:06:49 EEST 2023, Медзатий Дмитро Миколайович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 15.0%

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

ID: 113556 Назва: МКР Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей Додано в БД: 2023-05-18 Автора: Бойчук Я.А. Керівники: Бобровнікова К.Ю. Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	147553	869	24375 (17%)	191 (22%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми
112064	Назва: Я.А.Бойчука Додано в БД: 2023-03-20 Автора: Бобровнікова К.Ю. Керівники: ЗВІТ з науково-дослідної практики "Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей" Консультанти: Опоненти:	21659 (15.0%)	151 (17.0%)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Бойчук Ярослав Анатолійович

Тема: Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень ___; кількість сторінок записки 94

1. Короткий зміст роботи та прийнятих рішень В дипломній роботі удосконалено модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей, яка враховує вимоги безпеки до апаратно-програмних засобів Інтернету речей з врахуванням галузі застосування інфраструктури Інтернету речей, відомі стандарти безпеки в галузі Інтернету речей, грунтується на правилах побудови апаратно-програмних засобів на основі відомих практичних рекомендацій з забезпечення вимог безпеки та надає можливість впровадження криптографічних алгоритмів для програмного та апаратного забезпечення інфраструктури Інтернету речей.

2. Висновок про відповідність роботи дипломному завданню Дипломна робота відповідає виданому завданню як в теоретичній, так і в практичній частині.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: Розділ 1 – досліджена предметна область, відомі стандарти та рішення синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Розділ 2 – удосконалено модель процесу синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей. Розділ 3 – розроблено метод синтезу апаратно-програмних засобів Інтернету речей. Розділ 4 – представлено програмну реалізацію розробленого методу та продемонстровано результати її роботи. Всі розділи відповідають завданню.

4. Позитивні сторони роботи: Запропонований метод надає можливість синтезувати апаратно-програмні засоби Інтернету речей з врахуванням галузі застосування та експертних знань стосовно стандартів безпеки та вимог з вибору та впровадження програмних та апаратних криптографічних алгоритмів.

8. Негативні сторони роботи:

В роботі варто було б приділити більшу увагу апаратним платформам Інтернету речей

6. Оцінка графічного оформлення та пояснювальної записки роботи Пояснювальна записка відповідає нормам оформлення та виконана на достатньо високому рівні.

7. Відгук про роботу в цілому: Дипломна робота заслуговує оцінки. «добре». Дипломна робота присвячена вирішенню актуальної задачі розроблення методу синтезу апаратно-програмних засобів для підвищення безпеки інфраструктури Інтернету речей. Усі розділи роботи йдуть у вірній послідовності, що дозволяє розуміти викладений матеріал.

8. Інші зауваження:

9. Оцінка дипломної роботи: Розглянувши позитивні сторони представленої дипломної роботи, можна зробити висновок, що вона заслуговує на оцінку «добре».

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) Бармак О.В., зав. каф. КН

"18" травня 2023 р. _____(підпис)

Завідувачу кафедри КІІС
д-р.техн.наук, проф. Говорущенко Т. О.

Бойчука Ярослава Анатолійовича
ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2м-21-1

ЗАЯВА

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

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

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

18 травня 2023 року

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

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

Назва: Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей

Автор: Бойчук Ярослав Анатолійович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Бобровнікова К.Ю., к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

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

Підтвердження:

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

1) запозичення розміщені в розділах є збіг зі звітом з науково-дослідної практики автора Ярослава Бойчука "Метод синтезу апаратно-програмних засобів підвищення безпеки інфраструктури Інтернету речей", який було додано в репозитраій ХНУ 20 березня 2023 року;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) окремі виявлені збіги є загальновживаними фразами або виразами, про що свідчить посилання системи на збіг з джерелами на один фрагмент речення;
4) всі зафіксовані системою ознаки модифікації тексту відносяться до індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості Unicheck, складає 3.22% і адресується до 433 першоджерел; та системою Anti-Plagiarism складає 15%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ К.Ю.Бобровнікова

Гарант ОП _____ О. С. Савенко

Завідувач кафедри КІІС _____ Т. О. Говорущенко