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 та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей»

КвРКІП. 170159.02.01.01 ПЗ

Виконав: студент 2 курсу, група КІ2м-21-1 _____ Барчук Д.О.
Підпис Ініціали, прізвище

Керівник кандидат техн. наук, доцент _____ Нічепорук А.О.
Науковий ступінь, вчене звання Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КІІС, д.т.н., проф.
Т.О. Говорущенко
__ _____ 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 ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ)

Барчуку Денису Олександровичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей

Керівник проекту (роботи) Нічепорук А.О., 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)

Огляд відомих методів і засобів збору даних у мережах Інтернету речей та основи процесу отримання даних в мережах Інтернету речей

Модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні

Система і метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні

Моделювання мережі IoT в операційній системі Contiki, отримання даних та оцінка ефективності системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні

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

6. Консультанти розділів дипломного проекту (роботи)

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

7. Дата видачі завдання «_01_»_____09_____2022р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Прим
1	Вибір напряму дослідження та узгодження тематики КрМ з керівником	01.09.2022	викон
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.10.2022	викон
3	Робота над розділом 1 – Огляд відомих методів і засобів збору даних у мережах IoT; постановка задачі дослідження	01.11.2022	викон
4	Робота над розділом 2 – розробка моделі процесу збору даних із протоколу маршрутизації RPL.	01.12.2022	викон
5	Робота над тезою доповіді	01.02.2023	викон
6	Робота над розділом 3 – розробка методу та системи для вирішення поставленої задачі	15.02.2023	викон
7	Робота над розділом 4 – моделювання мережі IoT, імплементація системи збору дани для виявлення розподілених атак; оцінка ефективності	01.04.2023	викон
8	Оформлення пояснювальної записки згідно вимог	18.04.2023	викон
9	Попередній захист КрМ	28.04.2023	викон
10	Захист КрМ на засіданні ЕК	До 15.05.2023	

Студент _____ Підпис Д.О. Барчук Ініціали, прізвище

Керівник роботи _____ Підпис А.О. Нічепорук Ініціали, прізвище

РЕФЕРАТ

Тема кваліфікаційної роботи магістра: Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей

Автор роботи: Барчук Д.О.

Керівник роботи: Нічепорук А.О.

Пояснювальна записка: 71 с., 20 рис., 6 табл., 3 дод., 81 джерело.

ЗБІР ДАНИХ, СНІФЕР, МЕРЕЖУ ІНТЕРНЕТУ РЕЧЕЙ, РОЗПОДІЛЕНА АТАКА ВІДМОВА В ОБСЛУГОВУВАННІ

Об’єктом дослідження є процес збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей.

Предметом дослідження є метод та система збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні в мережах Інтернету речей.

Метою кваліфікаційної роботи магістра є отримання набору даних із мереж Інтернету речей на основі протоколу маршрутизації RPL, використання якого, дозволило б підвищити достовірність виявлення розподілених атак відмова в обслуговуванні у мережах IoT.

Для розв’язання поставлених задач використовувалися методи теорії графів, методи множин, оцінки ефективності та методи побудови теоретико-множинних моделей.

Наукова новизна отриманих результатів:

- набула подальшого розвитку модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні, яка на відміну від відомих здійснює послідовний опис процесу обробки та збору інформації із мережевого трафіку, що дозволило удосконалити метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні.

- набули подальшого розвитку система та метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні, які на відмінну від відомих залучають декілька мережевих сніферів для агрегації даних, що дозволило на основі використання отриманого набору даних здійснити виявлення розподілених атак відмова в обслуговувані.

Практична значимість отриманих результатів полягає у тому, що запропонована система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей може бути інтегрована в існуючі системи, що виконують задачі моніторингу стану мережі та виявлення зловмисної активності у IoT мережах.

ЗМІСТ

Скорочення та умовні познаки 4
Вступ 5
1 Огляд відомих методів і засобів збору даних у мережах Інтернету речей та основи процесу отримання даних в мережах Інтернету речей 8
1.1 Узагальнена архітектура 6LoWPAN-RPL мереж 8
1.2 Процес отримання даних в мережах Інтернету речей 12
1.3 Аналіз відомих підходів та стратегій до збору даних в мережах Інтернету речей 16
1.4 Аналіз відомих наукових методів збору даних та виявлення розподілених атак відмова в обслуговуванні 17
1. 5 Постановка задачі дослідження 20
2 Модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні 22
2.1 Модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні 22
2.2 RPL протокол маршрутизації для мереж Інтернету речей із низьким енергоспоживанням і втратами 28
2.3 Методи машинного навчання для виявлення атак відмова в обслуговувані 29
2.3.1 Метод опорних векторів 29
2.3.2 Штучні нейронні мережі 32
2.4 Висновки 36
3 Система і метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні 37
3.1 Узагальнена структура системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні 37
3.1.1 Модуль збору даних 38

3.1.2 Модуль класифікації 41
3.1.3 Модуль виявлення 42
3.2 Метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні 44
3.2.1 Фаза попереднього навчання 46
3.2.2 Фаза після навчання 48
3.3 Висновки 50
4 Моделювання мережі IoT в операційній системі Contiki, отримання даних та оцінка ефективності системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні 52
4.1 Операційна система Contiki та Cooja симулятор 52
4.2 Отримання даних для перевірки достовірності виявлення розподілених атак відмова в обслуговуванні 53
4.3 Імплементація модулю збору даних 59
4.3.1 Захоплення мережевого трафіку 60
4.3.1 Модуль агрегації 60
4.3.2 Блок черги 61
4.3.3 Отримання ознак 62
4.3.4 Маркування даних 63
4.4 Реалізація атаки скидання пакетів (blackhole attack) у мережі Інтернету речей 64
4.5 Створення та вибір моделі виявлення 66
4.6 Оцінка ефективності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих із протоколу RPL 71
4.7 Висновки 72
Висновки 74
Перелік посилань 76
Додаток А Встановлення та налаштування середовища Contiki 83
Додаток Б Копія наукової публікації 84

Додаток В Копія презентації до захисту кваліфікаційної роботи 89

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БД - база даних

ГА - генетичний алгоритм

ОС - операційна система

ПЗ - програмне забезпечення

МЗД - модуль збору даних

МВ - модуль виявлення

МК - модуль класифікації

СВВ - система виявлення вторгнень

DDoS - Distributed Denial of Service (розподілена відмова в обслуговуванні)

IDS - система виявлення вторгнень

IoT - Internet of Things

ВСТУП

На сьогоднішній день концепція Інтернету речей демонструє стійку тенденцію до розвитку у галузі інформаційних технологій, перш за все завдяки поширенню безпровідних сенсорних мереж, пришвидшенню переходу на IPv6 адресацію, використання хмарних обчислень та розвитку машинного навчання. Інтернет речей (IoT) об'єднує пристрої у комп'ютерну мережу й дозволяє їм збирати, аналізувати, обробляти та передавати дані іншим об'єктам (речам), що поєднані між собою через програмне забезпечення, програми або технічні пристрої. Піонерами у цій галузі є рішення від компаній Amzaon (Amazon Alexa hubs), Google (Google Home), Xiaomi (Xiaomi Smart Home), що надають кінцевим користувачам всі переваги від застосування Інтернету речей. Проте гетерогенність середовища та безпровідний спосіб обміну даними робить мережі Інтернету речей потенційними цілями для зловмисників.

Серед одних із основних загроз безпеці мережам IoT є атаки типу відмова в обслуговуванні (DoS). Даний тип атак призводить до втрати доступу до пристрою або ресурсів, які він пропонує. Зловмисники реалізують велике коло різних способів атаки, але найпоширеніші з них полягають у бомбардуванні системи величезною кількістю непотрібних даних, щоб заповнити доступну пропускну здатність мережі цілі або її обчислювальну потужність. Іншим варіантом впливу IoT мережу є пере направлення пакетів або їх відкидання. Даний види атак особливо гостро проявляється у IoT мережах з огляду на характер реалізації алгоритмів маршрутизації, що передбачають використання повнозв'язних топологій та передачу даних від джерела до приймача через ланцюжок проміжних вузлів. В загальному даний вид атак призводить до того, що легітимні користувачі втрачають доступ до ресурсів або пристроїв.

На сьогоднішній день традиційні підходи виявлення атак відмова в обслуговуванні не відповідають поточним вимогам безпеки. Існуючі методи та засоби

не дозволяють у повному обсязі протистояти постійно зростаючим загрозам. Разом із тим розробка будь-якого методу чи інтелектуальної системи, що використовує методи машинного навчання, вимагає збору даних, пов'язаних із сферою дослідження. У порівняні із традиційними мережами IT інфраструктур, для яких створено значні набори даних (наприклад KDD Cup), що дозволяють реалізувати навчання методів штучного інтелекту, для мереж Інтернету речей на сьогоднішній день не має аналогічного набору даних. Тому розробка нових методів збору даних в мережах IoT для їх подальшого аналізу та використання у системах виявлення розподілених атак типу відмова в обслуговуванні є актуальним завданням.

Метою роботи є отримання набору даних із мереж Інтернету речей на основі протоколу маршрутизації RPL, використання якого, дозволило б підвищити достовірність виявлення розподілених атак відмова в обслуговуванні у мережах IoT.

Об'єктом дослідження є процес збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей.

Предметом дослідження є метод та система збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні в мережах Інтернету речей.

Методи дослідження. У роботі було застосовано наступні теорії та засоби:

- математичні та аналітичні методи дослідження;
- теорія множин;
- теоретико-множинні моделі;
- методи оцінки ефективності;

Наукова новизна роботи:

- набула подальшого розвитку модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні, яка на відміну від відомих здійснює послідовний опис процесу обробки та збору інформації із мережевого трафіку, що дозволило

удосконалити метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні;

- набули подальшого розвитку система та метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні, які на відмінну від відомих залучають декілька мережевих сніферів для агрегації даних, що дозволило на основі використання отриманого набору даних здійснити виявлення розподілених атак відмова в обслуговувані.

Практична цінність роботи у тому, що запропонована система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей може бути інтегрована в існуючі системи, що виконують задачі моніторингу стану мережі та виявлення зловмисної активності у IoT мережах.

Рекомендації з використання результатів роботи. Отримані результати роботи можуть бути використанні при створенні антивірусних програмних комплексів для діагностування комп'ютерних систем на наявність шкідливого програмного забезпечення.

Апробація результатів. Наукові та практичні результати роботи доповідались та обговорювались на XIV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2022», Хмельницький національний університет, 18-19 листопада 2022 р.

Публікації. За темою роботи опубліковано одну тезу доповіді.

Структура та об'єм дипломної роботи. Дипломна робота складається з вступу, чотирьох розділів, висновку та додатків, її повний зміст 107 сторінок, основний зміст викладено на 75 сторінках, 3-х додатках на 21 сторінці, містить 20 рисунків, 6 таблиць, включає 61 найменуваня вітчизняної та зарубіжної літератури.

1 ОГЛЯД ВІДОМИХ МЕТОДІВ І ЗАСОБІВ ЗБОРУ ДАНИХ У МЕРЕЖАХ ІНТЕРНЕТУ РЕЧЕЙ ТА ОСНОВИ ПРОЦЕСУ ОТРИМАННЯ ДАНИХ В МЕРЕЖАХ ІНТЕРНЕТУ РЕЧЕЙ

1.1 Узагальнена архітектура 6LoWPAN-RPL мереж

6LoWPAN – стандарт взаємодії за протоколом IPv6 поверх малопотужних бездротових персональних мереж стандарту IEEE 802.15.4. Вихідний канал до Інтернету у таких мережах забезпечується точкою доступу (AP), що діє як маршрутизатор IPv6. У типовій конфігурації точки доступу підключається кілька різних пристроїв, таких як робочі станції, сервер, тощо. 6LoWPAN-мережа пов'язана з IPv6-мережею за допомогою використання граничного маршрутизатора. Граничний маршрутизатор виконує три дії: обмін даними між пристроями 6LoWPAN та Інтернетом (або іншою IPv6-мережею), локальний обмін даними між пристроями в 6LoWPAN-мережі та формування й обслуговування радіопідмережі (6LoWPAN-мережі). На рис. 1.1 приведено узагальнену архітектуру мережі IPv6, що включає комірчасту (mesh) 6LoWPAN-RPL мережу.

Взаємодіючи природним способом з IP у рідному форматі, 6LoWPAN-мережі зв'язуються з іншими мережами через стандартні IP-маршрутизатори. Згідно із рис. 1.1, мережі 6LoWPAN, як правило, працюватимуть як кінцеві структури на межі глобальної мережі. Це означає, що вхідні дані призначені для одного з пристроїв 6LoWPAN. Одна 6LoWPAN мережа може бути пов'язана з іншими IP-мережами через один або більше граничних маршрутизаторів, які відправляють IP-датаграми між різними середовищами передачі даних.

Забезпечення зв'язку із іншими IP-мережами може реалізуватись через будь-який довільний канал, такий як Ethernet, Wi-Fi або 3G/4G. Оскільки 6LoWPAN тільки конкретизує операції IPv6 поверх стандарту IEEE 802.15.4, граничні маршрутизатори можуть також підтримувати механізми переходу IPv6, щоб з'єднувати 6LoWPAN мережі із IPv4-мережами.

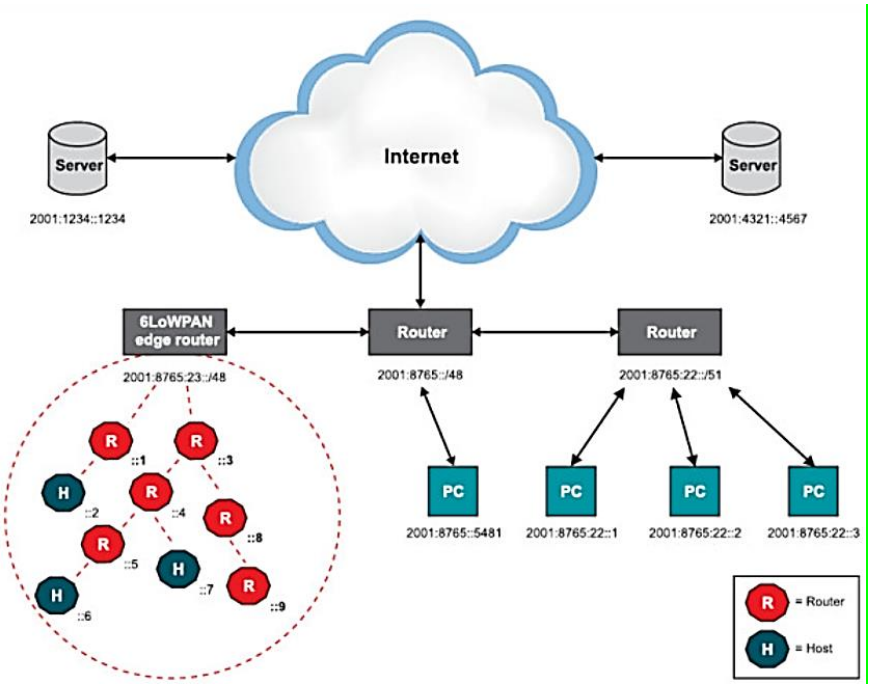


Рисунок 1.1 – Узагальнена архітектура 6LoWPAN-RPL мережі

Оскільки граничні маршрутизатори надсилають датаграми на мережному рівні, вони не підтримують стан прикладного рівня. Мережі із іншою архітектурою, такі як ZigBee, Z-Wave, Bluetooth або інші фірмові мережі, вимагають, щоб шлюзи виконували складні прикладні програми, які дозволяють перетворити специфічний мережевий трафік бездротової мережі перед надсиланням даних у стандартний IP-канал. Ці прикладні шлюзи повинні сприймати будь-які прикладні профілі, які можуть використовуватися в мережі, і будь-які зміни прикладних протоколів бездротових вузлах повинні також супроводжуватися змінами на шлюзі. Навпаки, засновані на протоколах IP маршрутизатори на межах мереж, наприклад, граничний маршрутизатор, не залежать від прикладних протоколів, що використовуються в 6LoWPAN мережах. Це знижує навантаження на граничний маршрутизатор, наприклад, на його обчислювальну потужність, дозволяючи тим самим використовувати пристрої з низькою вартістю, більш простим програмним забезпеченням і менш складними апаратними засобами. При цьому архітектура IP не

перешкоджає використанню для оптимізації роботи мережі проксі та кеш-пам'яті, які широко використовуються у сучасному Інтернеті.

У типову 6LoWPAN мережу зазвичай включають ще два пристрої: маршрутизатори (router) та хости. Маршрутизатори, можуть ретранслювати дані, призначені іншим вузлам у 6LoWPAN-мережі. Хости, що також називаються кінцевими пристроями, не в змозі спрямовувати дані на інші пристрої в мережі. Хост може також бути «сплячим» пристроєм, який «прокидається» періодично, щоб перевірити наявність даних у свого «батька» (роутера), що дозволяє споживати дуже малу потужність.

Усі системи зв’язку використовують набір правил чи стандартів форматування та управління обміном даних. Найбільш загальна модель систем передачі — модель взаємодії відкритих систем (OSI), яка, у спрощеному вигляді, розбиває зв'язок п'ять фундаментальних рівнів. На рис 1.2 показано цю спрощену модель OSI з двома типовими прикладами стеків, що використовуються в пристроях IoT. Перший приклад - пристрій, що запускає стек Wi-Fi, другий приклад - пристрій, підключений до IoT і заснований на 6LoWPAN.

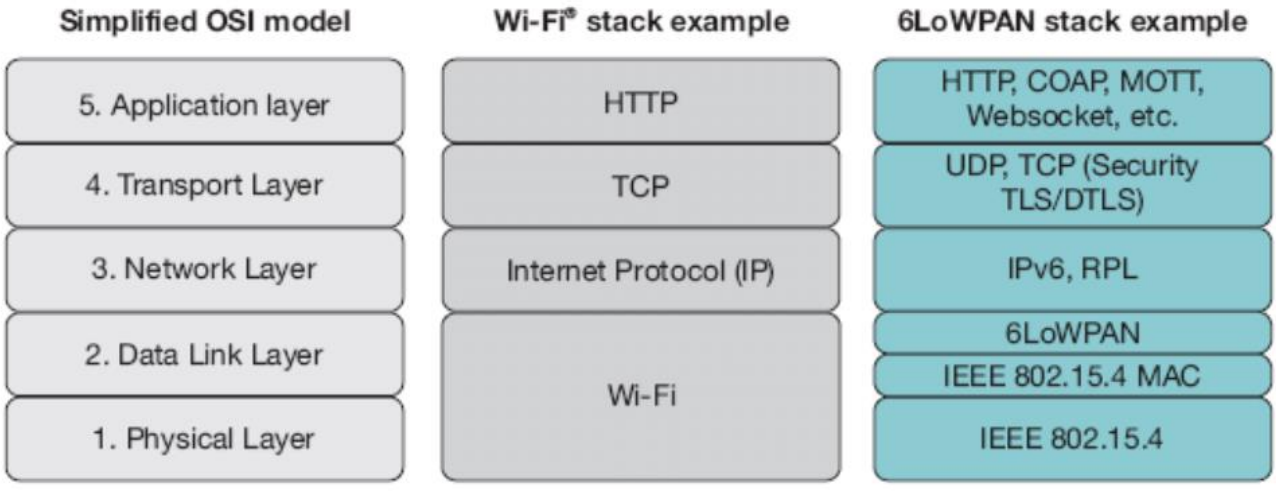


Рисунок 1.2 – Порівняння моделі OSI та стеків протоколів, що використовуються у мережах Інтернету речей

Фізичний рівень перетворює біти даних сигнали, які передаються і виходять за ефіром. У 6LoWPAN мережах, наприклад, використовується IEEE 802.15.4. Крім широко відомої версії стандарту 2006 року існують дві важливі поправки: IEEE 802.15.4e – поправка до MAC, вона пропонує такі розширення, як перемикання каналів із виділенням квантів часу (TSCH – time slotted channel hopping) та координоване виборче прослуховування (CSL – coordinated sampled listening). Обидва розширення націлені на додаткове зниження витрати енергії та роблять бездротовий інтерфейс більш стійким. IEEE 802.15.4g – поправка до PHY (фізичного рівня), мета якої – забезпечити додатковий діапазон радіочастот.

Канальний рівень забезпечує надійну передачу даних між двома безпосередньо пов’язаними вузлами за рахунок виявлення та виправлення помилок, які можуть виникнути фізично під час передачі та прийому. Канальний рівень включає рівень доступу до середовища передачі (MAC), що використовує метод множинного доступу з контролем несучої та виключенням зіткнень (CSMA-CA), відповідно до якого радіостанція прослуховує ефір, визначаючи, чи веде передачу якась інша станція, перед початком передачі даних з ефіру. Цей рівень також забезпечує синхронізацію даних. У 6LoWPAN, наприклад, рівень MAC визначається стандартом IEEE 802.15.4. Рівень адаптації 6LoWPAN, забезпечуючи перехід від IPv6 до IEEE 802.15.4, також знаходиться на канальному рівні.

Мережевий рівень підтримує адресацію та маршрутизацію даних через мережу, якщо потрібно зробити кілька ретрансляцій. IP (міжмережевий протокол) — мережевий протокол, який використовується для надання всім пристроям IP-адрес для транспортування пакетів від одного пристрою до іншого.

Транспортний рівень відкриває сесії зв’язку між прикладними програмами, що запускаються на кінцевих пристроях. Транспортний рівень дозволяє декількома програмами на кожному пристрої мати власний канал зв’язку. TCP – домінуючий транспортний протокол в Інтернеті. Однак TCP заснований на протоколі з'єднання (включаючи впорядкування пакетів) з великим обсягом службової інформації і тому

не завжди підходить для пристроїв, що потребують споживання енергії. Для систем такого типу найкращою опцією може бути UDP – протокол без встановлення з'єднання з меншим обсягом службової інформації. Прикладом безпечного транспортного рівня є протокол TLS (безпека транспортного рівня), який виконується поверх TCP та DTLS та заснований на UDP.

Прикладний рівень відповідає за форматування даних. Він також перевіряє, що дані транспортуються за оптимальною для застосування схемою. Найбільш відомими протоколом прикладного рівень є HTTP, що запускається поверх TCP. HTTP використовує XML – текстова мова з великим обсягом службової інформації. Тому не оптимально використовувати HTTP у багатьох системах 6LoWPAN. Однак HTTP все ще може бути дуже корисним для зв’язку між 6LoWPAN та глобальною мережею Інтернет. Тому промислові та громадські організації розробили альтернативні протоколи прикладного рівня, наприклад, обмежений прикладний протокол (COAP) – протокол передачі повідомлень поверх UDP з біт-оптимізованим механізмом REST, що є дуже схожим на протокол HTTP. Протокол COAP віднесений IETF RFC 7252 і визначає ретрансляцію повідомлень як із підтвердженням так і без, підтримку «сплячих» пристроїв, передачу блоків, підтримку підписки та виявлення сервісів. COAP також легко перетворити на HTTP через проксі.

Таким чином специфіка роботи 6LoWPAN-RPL мереж потребує розгляду специфічних процесів та засобів моніторингу та аналізу мережевого трафіку, що дозволить ґрунтуючись на цьому, здійснити процедуру виявлення кібератак на мережі Інтернету речей.

1.2 Процес отримання даних в мережах Інтернету речей

Сьогодні технології такі як Інтернет речей, мобільний інтернет, хмарні, обчислення стають центральними елементами, що допомагають людству у вирішенні різноманітних завдань. При цьому телекомунікаційні мережі забезпечують

безперервний рух великих обсягів даних між цими пристроями, які часто мають критично важливе значення. На протязі всього життєвого циклу функціонування телекомунікаційна інфраструктура, сервери, програмні продукти, віртуальна інфраструктура досить часто перебувають під загрозою непередбачуваних подій, зокрема таких як збої в роботі обладнання та вихід його із ладу, вплив кібератак, прояв програмних помилок, збої в роботі мережі та інші зовнішні загрози. Тому у будь-якій організації втрата або припинення функціонування мережевої інфраструктури через аварії або катастрофи, неминуче може призвести до репутаційних витрат і, відповідно, втрати прибутку. Якщо ж мова йде про мережі Інтернету речей, то припинення функціонування IoT інфраструктури призведе до паралізації сервісів та припинення комунікації між пристроями, що несе із собою значні незручності для кінцевих користувачів. Враховуючи ситуацію, яка склалася, із впевненістю можна сказати, що багато фахівців почали висувати більш високі вимоги до питання забезпечення умов безперервності роботи мережевих сервісів. Проте для створення оптимальної із точки зору витрат і максимально ефективної мережевої інфраструктури необхідно мати чітке уявлення, про те, які саме події відбуваються в мережевій системі та яким чином на них реагувати. Такі завдання вирішуються за допомогою систем збору даних у режимі реального часу, де мережеві пристрої, безперервно передають дані, пов’язані із станом мережі та центром обробки, з метою їх подальшого аналізу.

В загальному, процес отримання даних в мережах Інтернету речей, які можна використовувати для моделей машинного навчання або аналізу проходить три послідовні фази. Початкова фаза – це генерація самих даних. Ця фаза реалізується на рівні пристроїв Інтернету речей, звідки дані далі передаються через мережу. Другий етап отримання даних IoT – це їх збір та впорядкування. Третій етап передбачає фактичне використання цих даних. На рис. 1.3 приведено узагальнений процес отримання даних в мережах Інтернету речей.

Рисунок 1.3 – Узагальнений процес отримання даних в мережах Інтернету речей

Потокова передача даних IoT. У концепції Інтернету речей кожна подія генерує дані. Для надсилання даних використовуються стандартні протоколи, такі як MQTT, WAMP, HTTP, CoAP або Sigfox. Кожен із них має свої сильні сторони та визначені випадки використання. Ці протоколи підтримують отримання оновлень або іншої інформації із пристроїв IoT з подальшим надсиланням цієї інформації в задане місце призначення для фактичної обробки.

На цьому етапі потрібно вирішити, як дані будуть агрегуватись та зберігатись для подальшого використання. На цьому кроці визначається чи потрібно надсилати дані в режимі реального часу або ж окремими партіями. Окрім того, на даному кроці визначається, в якому порядку слід отримувати точки даних для максимально точного аналізу.

Зберігання даних IoT. Використання даних у реальному часі забезпечує максимальну точність. Такий підхід гарантує доступ до всіх даних, згенерованих кожним пристроєм IoT. Однак зазвичай це означає досить велику кількість вхідних даних. Встановлення правильної позначки часу для сортування даних, що надходять із кількох пристроїв Інтернету речей, стає проблемою. На даному кроці слід вирішити завдання оптимізації між швидкістю та обсягом збору вхідних даних. Вибір підходу збору всіх доступних даних IoT у режимі реального часу та їхнього аналізу повинен мати чітке обґрунтування. В іншому випадку величезний обсяг даних може мати невиправдано сильний вплив на хмарні системи, тобто на мережеві та обчислювальні ресурси, необхідні для підтримки надходження даних IoT.

Слід також розглянути конкретні програми IoT. Вони мають унікальні вимоги щодо затримки, споживання енергії та точності. Деякі програми можуть допускати

затримки. Інші, такі як додатки безпеки, швидше розглядаються як критичні за часом і не можуть мати місця для затримок.

Багато варіантів використання можуть не вимагати високої точності та дозволять надсилати дані порціями. У такому варіанті дані надсилаються не у реальному часі, а лише через задані заздалегідь встановлені проміжки часу. Вибір конкретного випадку використання залежить від вимог. У деяких випадках для аналізу можуть знадобляться точні дані в реальному часі. В інших сценаріях цілком достатньо буде даних, що мають мітку часу в минулому.

1.3 Аналіз відомих підходів та стратегій до збору даних в мережах Інтернету речей

Підхід, обраний для збору та зберігання даних IoT, сильно залежить від цільових вимог конкретного випадку використання. Вони передбачають, але не обмежуються процедурами збору даних, що включають додаткові вимоги такі як точність даних, рівень споживання енергії, час відгуку та захист конфіденційності. Відомими підходами до зменшення обсягів зібраних даних IoT є агрегація даних, фільтрація, інтерпретація та стиснення на рівні сенсора або вузла-IoT, керуючись загальним принципом, що включає процес опрацювання якомога ближче до джерела даних.

Розглянемо детальніше відомі підходи, що використовуються для збору даних.

Стратегії, засновані на досягненні максимальної достовірності. Дана стратегія збору даних у мережах IoT враховує компроміс між частотою запитів на вимірювання та точністю даних. Тут частота адаптована з урахуванням цільової точності даних. Для моделей, що фокусуються на досягненні заданого рівня достовірності, отримання вищого показника достовірності не принесе додаткового зиску.

У рамках цієї стратегії необхідно зменшити частоту вимірювань даних. Це зменшує ресурси, необхідні для збору даних IoT, зберігаючи при цьому цільову точність, яка була розрахована як оптимальна для даного випадку використання.

Стратегії, мотивовані критичністю часу. При таких підходах до збору даних у мережах IoT максимальне значення затримки встановлюється заздалегідь. Час, що минув з часової позначки останнього вимірювання, має бути меншим за це максимальне значення затримки. У сценаріях збору даних IoT, мотивованих вимогами, пов'язаними із часом, кожне нове отримане вимірювання за вирахуванням мітки часу останнього вимірювання має бути меншим за максимальну затримку. Таким чином, час, що минув, відповідає «свіжості» даних вимірювань.

Стратегії збору даних IoT, мотивовані споживаною потужністю. При виборі такого підходу споживання енергії є ключовим фактором. Зусилля по збору даних спрямовані на досягнення заданої точності при оптимізації енергоспоживання. Енергетична стратегія збору даних IoT спрямована на максимальну ефективність. Вигода тут вимірюється як різниця між корисністю, досягнутою для певної цільової точності даних, мінус енергоспоживання для вимірювань, необхідних для досягнення цієї точності даних.

Подібно до сценарію, орієнтованого на точність, тут припускається, що залежно від застосування буде задано конкретну цільову точність, знаючи, що вища точність не принесе додаткових переваг.

Стратегії збору даних IoT, мотивовані проблемами конфіденційності. Даний підхід до збору даних використовується коли потрібно досягнути заданий рівень конфіденційності. Мета полягає в тому, щоб захистити конфіденційність кінцевих користувачів, змінюючи точність окремих вимірювань і водночас зберігаючи певний рівень «адекватної точності» для результатів у цілому.

1.4 Аналіз відомих наукових методів збору даних та виявлення розподілених атак відмова в обслуговуванні

Проблемі збору даних для виявлення розподілених атак відмова в обслуговуванні у науковій літературі приділяється значна увага. Найбільш відомими

підходами до збору даних та виявленні на їх основі розподілених атак відмова в обслуговуванні у мережах Інтернету речей є методи засновані на машинному навчанні, статистичних алгоритмах, алгоритмах часових рядів та на основі залучення програмно-конфігурованих мереж.

У роботі [1] представлено метод SEED (Secure and energy efficient data-collection – безпечний та енергоефективний збір даних). Представлений метод заснований на створенні вузлів агрегації та використанні алгоритмів виявлення шляху. В основі представленого методу оновлення вузла агрегатора виконується після кожного збою або передачі.

У дослідженні [2] запропоновано технологію збору даних SEEDGT в мережах IoT, що спрямована на досягнення заданих параметрів безпеки. В основі представленого рішення задіяно алгоритм відкритого ключа та методи Compressive Sensing для досягнення безпеки на основі справедливого енергетичного навантаження. Запропонована методика включає три фази, а саме формування кластера, роботу мережі та фазу реконфігурації. Під час фази формування кластерів для створення кластерів і вибору голови кластера застосовуються методи, що ґрунтуються на показнику довіри та із урахуванням показника споживаної енергії. Метою етапу роботи мережі є досягнення безпеки шляхом використання алгоритму відкритого ключа для шифрування мережевих даних під час процесу збору даних. Крім того, на цьому етапі стратегія CS використовується для зменшення початкового розміру даних, що призводить до зменшення споживання енергії. Нарешті, зміни, які можуть відбутися під час роботи мережі, розглядаються на етапі реконфігурації.

У роботі [3] автори запропонували модель для аналізу та моделювання мережевого трафіку в мережах IIoT. У цьому дослідженні автори аналізують вразливості IIoT екосистем не лише з точки зору окремих вузлів, а й як інтегрованої інфраструктури цифрових і фізичних систем, що взаємодіють із доменами. Автори пропонують структуру моделі загроз для аналізу атак на середовища додатків IIoT. Автори визначили потоки конфіденційних даних всередині пристроїв IIoT, щоб

визначити ризики конфіденційності на рівні програми та дослідили обмін пристроями на фізичному рівні. Автори також провели аналіз безпеки від фізичних до цифрових доменів.

У роботі [5] запропоновано підхід до виявлення розподілених атак відмова в обслуговуванні на основі побудови топологічної структури даних трафіку із залученням теорії графів. Дані про мережевий трафік переставляються у вигляді орієнтованого графу. У якості ребер у графі автори використовують інформацію про зв'язки між вузлами, частота, тривалість потоку та інша інформацію із мережевого трафіку. Для зменшення розмірності даних використано метод PCA. Процес виявлення реалізовано із використанням нечіткої C-means класифікації.

Автори роботи [6] запропонували підхід до виявлення DDoS атак у мережах Інтернету речей, що передбачає перетворення мережевого трафіку у форму зображення із подальшим залученням моделі залишкової нейронної мережі. Результати запропонованого методу покази високі результати виявлення для бінарної класифікації (99%) та 87% для багато класової класифікації.

Автори [7] підійшли до вирішення проблеми безпеки в мережах IoT шляхом оптимізації й зменшення розмірів вхідних даних та дослідили проблему вилучення підмножини найбільш релевантних функцій із мережевого трафіку. Запропоновано економічно ефективну з точки зору ефективності модель для очищення та підготовки необроблених даних перед зменшенням розмірності. Запропонували гібридний метод відбору ознак на основі міри взаємної інформації (mutual information), дисперсійного аналізу (ANOVA), методу хі-квадрат, алгоритму дерева рішень.

Автори дослідження [8] запропонували статистичний механізм виявлення, заснований на безперервно-ранжованих оцінках ймовірності (continuous ranked probability score – CRPS) та експоненціальному згладжуванні (exponentially smoothing) для ефективного виявлення атак відмови в обслуговуванні (DoS) і DDoS. Автори використовують CRPS для визначення кількісної оцінки відмінності між новим спостереженням і розподілом звичайного трафіку. Для перевірки ефективності

запропонованого рішення авторами було проведено ряд експериментів на трьох наборах даних. Представлене рішення продемонструвало досить високий показник ефективності, проте при низькому мережевому трафіку ефективність запропонованого рішення погіршується.

Інший статистичним підхід до виявлення розподілених атак відмова в обслуговуванні представлено у роботі [9]. Авторами досліджено прояви шкідливої активності у мережах розумного будинку. Для виявлення активних атак автори застосували технологію VPN разом із системою виявлення вторгнень Snort.

У роботі [4] запропонували метод виявлення розподілених атак відмова в обслуговуванні на основі аналізу часових рядів. Автори розглядають профіль схожості мета часового ряду (meta time-series), що представляє дані часового ряду. Для знаходження різниці між даними часових рядів використано Евклідову метрику.

Автори [10] представили спосіб виявлення DDoS атак у мережах Інтернету речей на основі залучення недорогих алгоритмів машинного навчання та даних, отриманих із мережевого трафіку на основі потоків і протоколів. У цій роботі були розглянуті деякі обмежені особливості поведінки мережі IoT, такі як розрахунок кінцевих точок і часу, необхідного для переходу від одного пакета до іншого (часові інтервали між пакетами), розмір пакетів, смуга пропускання (Bandwidth) та інші. В якості класифікаторів для виявлення атак, було використано алгоритми KNN, KDTree, SVM із лінійним ядром (LSVM), DT із використанням показників домішок Gini, RF із використанням показників домішок Gini, NN (KNN, KDTree algorithm, SVM with the linear kernel (LSVM), DT using Gini impurity scores, RF using Gini impurity scores, NN). В роботі стверджується, що запропоновані методи можуть ідентифікувати DDoS-атаки на локальних пристроях IoT, що працюють разом із домашніми маршрутизаторами та іншими проміжними блоками мережі.

Ще одним підходом протидії атак відмова в обслуговуванні на мережі IoT є використання технології блокчейн [11]. Автори запропонували модель блокчейну Ethereum для виявлення та запобігання DDoS-атак на системи IoT. Окрім того, за

ствердженням авторів запропоновану систему можна використовувати для усунення окремих точок збою, конфіденційності та безпеки в системах IoT. Авторами запропоновано реалізацію децентралізованої платформи на противагу відомих централізованих системних рішень для запобігання DDoS-атакам на пристрої IoT на прикладному рівні шляхом автентифікації та перевірки цих пристроїв. Також автори запропонували відстежувати та записувати IP-адреси шкідливих пристроїв усередині блокчейну, щоб запобігти їх підключенню та спілкуванню з мережами IoT.

1.5 Постановка задачі дослідження

Розробка будь-якого методу чи інтелектуальної системи, що використовує методи машинного навчання, вимагає збору даних, пов’язаних із сферою дослідження. У порівняні із традиційними мережами IT інфраструктур, для яких створено значні набори даних (наприклад KDD Cup), що дозволяють реалізувати навчання методів штучного інтелекту, для мереж Інтернету речей на сьогоднішній день не має аналогічного набору даних. Тому розробка нових методів збору даних в мережах IoT для їх подальшого аналізу та використання у системах виявлення розподілених атак типу відмова в обслуговуванні є актуальним завданням. Вирішення поставленого завдання досягається шляхом виконання наступних етапів:

1. Проаналізувати відомі методи та засоби збору даних в мережах Інтернету речей.
2. Проаналізувати методи машинного навчання, що використовуються для виявлення розподілених атак відмова в обслуговуванні.
3. Дослідити RPL протокол маршрутизації для мереж Інтернету речей із низьким енергоспоживанням і втратами.
4. Розробити модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні.

5. Запропопонувати метод та систему збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні.

6. Провести моделювання мережі Інтернету речей на основі операційної системи Contiki та симулятора Cooja з метою оцінки достовірності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих запропонованою системою.

1.6 Висновки

Представлені методи і засоби збору даних у мережах інтернету речей В ході роботи над першим розділом було проведено аналіз всіх представлених методів із методів і засобів збору даних в мережах інтернету речей та основи процесу отримання даних в мережах інтернету речей описує процес отримання даних в мережах, аналіз відомих підходів та їх стратегій. На основі аналізу виявлено недоліки в існуючих методах та проведено постановку задачі дослідження для створенні безпечнішого методу.

2 МОДЕЛЬ ПРОЦЕСУ ЗБОРУ ДАНИХ ІЗ ПРОТОКОЛУ МАРШРУТИЗАЦІЇ RPL У МЕРЕЖАХ ІНТЕРНЕТУ РЕЧЕЙ ДЛЯ ВИЯВЛЕННЯ РОЗПОДІЛЕНИХ АТАК ВІДМОВА В ОБСЛУГОВУВАННІ

2.1 Модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні

Оскільки об’єктом дослідження є процес збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні, то важливою задачею є розроблення його моделі, що дозволить формалізувати основні етапи.

Представлена узагальнена теоретико-множинна модель процесу описує послідовний процес обробки та збору інформації із мережевого трафіку з метою виявлення розподілених атак відмова в обслуговуванні та із подальшою ізоляцією скомпрометованих вузлів. Вхідними даними для представленої моделі процесу є вхідний трафік, тоді як вихідними даними новий маршрут, що представляє нове дерево DODAG у протоколі RPL, у якому будуть відсутні скомпрометовані вузли.

Запропонована модель складається із трьох підпроцесів: збір даних, виявлення та блокування підозрілої активності. Узагальнене схематичне представлення підпроцесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні на рис. 2.3.

Розглянемо послідовно кожний із підпроцесів.

Підпроцес збору даних визначає активності по отриманню із мережевого трафіку векторів ознак.

Представимо функцію обміну даними в мережі Інтернету речей у вигляді:

$$f_{de} : N \times S \to T , \qquad (2.1)$$

де $N = \{ n_i^l \cup n_j^m \}_{i=2,j=0}^{N_N}$ – множина вузлів у мережі, причому n_i^l – легітимний вузол, без шкідливої активності, n_j^m – скомпрометований вузол, N_N – загальна кількість вузлів у мережі;

$S = \{ s_i \}_{i=1}^{N_S}$ – множина сніферів, що функціонують у мережі Інтернету речей, де N_S – загальна кількість сніферів у мережі;

T – мережевий трафік.

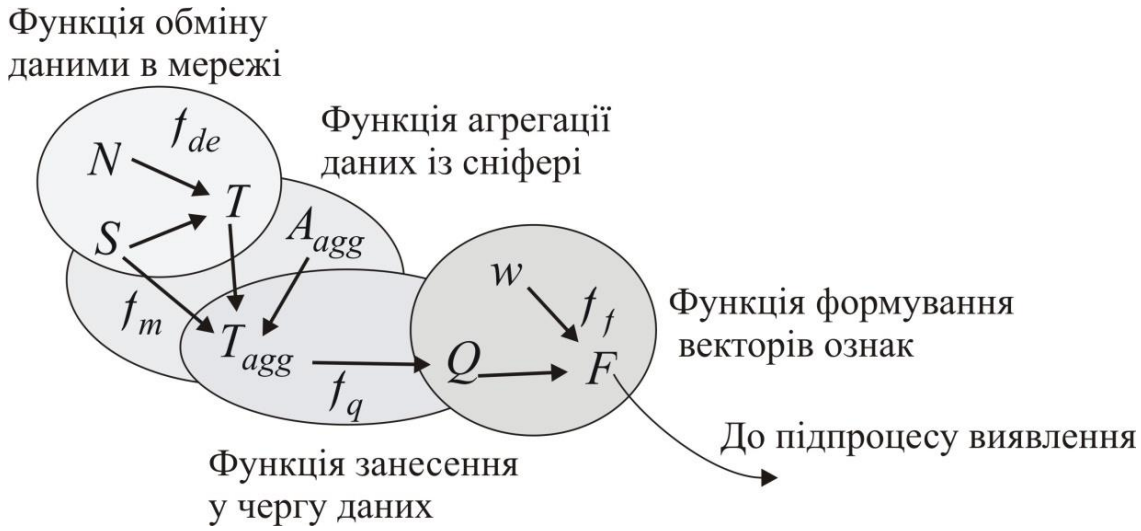


Рисунок 2.1 – Узагальнене схематичне представлення підпроцесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні

Тоді визначимо функцію агрегації даних із сніферів наступним чином:

$$f_m : S \times T \times A_{agg} \to T_{agg}, \tag{2.2}$$

де A_{agg} – алгоритм агрегації трафіку;

Результатом роботи даної функції є агрегований трафік T_{agg}, у якому відсутні дублювання пакетів за ідентифікатором вузла та за міткою часу.

Тоді наступними під процесами будуть занесення даних у чергу та формування векторів ознак. Представимо дані під процеси у вигляді відповідних функцій.

Функція занесення у чергу даних:

$$f_q : T_{agg} \to Q, \tag{2.3}$$

де Q – черга пакетів.

Функція формування векторів ознак:

$$f_f : Q \times w \to F, \tag{2.4}$$

де w – часове вікно для формування ознак; $F = \{ f^p_{RSSI}, f^p_{RdBM}, f^p_{TdBM}, f^n_{LQI}, f^n_{ETX}, f^n_{NDIO}, f^n_{NDIS}, f^n_{NDIS}, f^n_{LRPL}, f^a_{MeCP}, f^a_{MoCP}, f^a_{NID} \}_{i=1}^{F_N}$ – множина векторів ознак, де ознаками є: f^p_{RSSI} – показник рівня приймаючого сигналу RSSI, f^p_{RdBM} – значення отриманого сигналу dBm, f^p_{TdBM} – значення переданого сигналу dBm, f^n_{LQI} – значення якості зв'язку, f^n_{ETX} – значення очікуваної кількості передач ETX, f^n_{NDIO} – кількість повідомлень DIO, f^n_{NDIS} – кількість повідомлень DIS, f^n_{LRPL} – зміна рівня RPL (ранг) вузла, f^a_{MeCP} – середнє значення споживаної потужності, f^a_{MoCP} – модальне значення споживаної потужності, f^a_{NID} – ідентифікатор вузла.

Опишемо підпроцес виявлення у вигляді функцій моніторингу мережевого трафіку, навчання моделі виявлення, активації процедури виявлення та безпосереднього виявлення.

Представимо функцію навчання моделі виявлення наступним чином:

$$f_l : F \to M\ , \tag{2.5}$$

де $M = < m_1, m_2, ..., m_N >$ – модель виявлення, яка представлена набором гіперпараметрів $m_1, m_2, ..., m_N$.

Задамо функцію моніторингу мережевого трафіку у вигляді:

$$f_m : S \times T \times P_m \to T_{raw}, \tag{2.6}$$

де $P_m = \{ p_1, p_2 \}$ – параметри моніторингу, p_1 – часове вікно моніторингу, p_2 – максимальний розмір необробленого мережевого трафіку;

T_{raw} – необроблений трафік мережі Інтернету речей отриманий відповідно до параметрів P_m.

Наступною активністю є бінарна функція активації процедури виявлення, яку задамо наступним чином:

$$f_a : T_{raw} \times R \to \{ true, false \}, \tag{2.7}$$

де $R = \{ r_i \}_{i=1}^{N_R}$ – множина правил активації виявлення; A_{det} – алгоритм виявлення.

У випадку, якщо результат роботи бінарної функції f_a буде true виконується функція виявлення, що представлена наступним чином:

$$f_{dt} : A_{det} \times M \to X \,, \tag{2.8}$$

де $X = \{ x_i \}_{i=1}^{N_X}$ – множина скомпрометованих вузлів.

Останній процес визначає блокування зловмисної активності. Подамо даний під процес у вигляді двох функцій: трансляції попереджувальних повідомлень та пере направлення мережевого трафіку:

Функція трансляції попереджувальних повідомлень, що містять ID скомпрометованого вузла, усім вузлам у мережі, визначимо наступним чином:

$$f_{al} : X \times \{ N \setminus X \} \times mDIO \to \{ N \setminus X \}, \tag{2.9}$$

де, $\{ N \setminus X \}$ – множина вузлів мережі, окрім скомпрометованих вузлів; *mDIO* – повідомлення DIO протоколу маршрутизації RPL.

Функцію перенаправлення мережевого трафіку, який надходить до вузла-жертви та від нього, використовуючи найближчого сусіда із найвищим рангом у дереві DODAG визначимо наступним чином:

$$f_{al} : N \times A_f \to DodagTree, \tag{2.10}$$

де, A_f – алгоритм пошуку найближчого сусіда із найвищим рангом у дереві DODAG; DodagTree – новий маршрут (нове дерево DODAG), без скомпрометованих вузлів.

Таким чином запропонована модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні є представленням узагальних підпроцесів основною метою

яких є отримання даних із IoT мережі та виявлення і блокування розподілених атак відмова в обслуговуванні.

2.2 RPL протокол маршрутизації для мереж Інтернету речей із низьким енергоспоживанням і втратами

RPL – це протокол маршрутизації на основі IPv6, спеціально розроблений для вузлів-давачів у безпровідних сенсорних мережах (WSN). Цей протокол маршрутизації також розроблений для інтеграції протоколу маршрутизації за стандартом IEEE 802.15.4 ss IP-протоколом на основі IPv6, щоб впоратися із гетерогенністю (неоднорідністю) концепції IoT. Протокол маршрутизації RPL реалізує топологію multi-hop шляхом формування Directed Acyclic Graph (DAG) – мережі з топологією дерева у вигляді орієнтованого графа. DAG (як показано на рис. 2.1) може складатися з однієї або кількох мереж. DAG формується на основі одного або кількох критеріїв, визначених самим вузлом, зокрема такими як кількість передач (ETX), затримка, кількість переходів (hop) або енергія вузла. Орієнтований ациклічний граф (DODAG) формується з DAG, який має один корінь вузла, і всі решта вузлів поєднані у деревоподібну структуру.

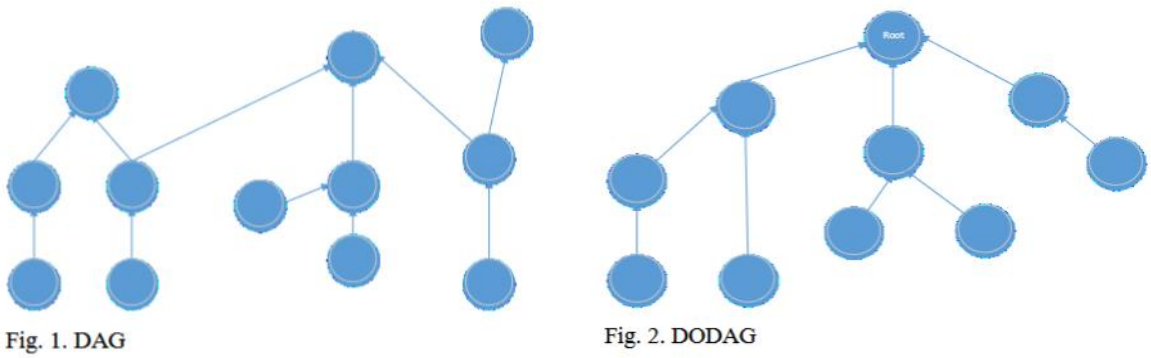


Рисунок 2.2 – DAG та DODAG дерева

В основі функціонування протоколу маршрутизації RPL існує 5 керуючих повідомлень [60]:

Інформаційний об’єкт DODAG (DIO): це повідомлення розповсюджується вниз по дереву. Заданий вузол у DODAG може розсилати це повідомлення, яке дозволяє іншим вузлам дізнатись про нього. Це повідомлення використовується з метою отримати інформацію про те, чи є вузли, які хочуть приєднатись до дерева.

Інформаційний запит DODAG (DIS): якщо відсутнє повідомлення DIO, і якщо вузол хоче приєднатися до дерева DODAG, він надсилає дане контрольне повідомлення. Таким чином DIS дозволяє згенерувати запит на пошук будь-яких DODAG.

Об’єкт оновлення DODAG (DAO): це запит, надісланий дочірнім вузлом батьківському або кореневому. У цьому повідомленні (від нащадка до батьківського вузла) батьківському вузлі пропонується дозволити нащадку приєднатися до DODAG.

DAO-ACK: це відповідь від батьківського до дочірнього вузла, що дозоляє або не дозоляє нащадку приєднатись до дерева.

Consistency check: використовується для забезпечення безпеки з’єднання.

Таким чином організація побудови DODAG дерева може бути представлена

Схематичне представлення організації зв’язку між двома вузлами (батьківським та нащадком) представлено на рис.2.3.

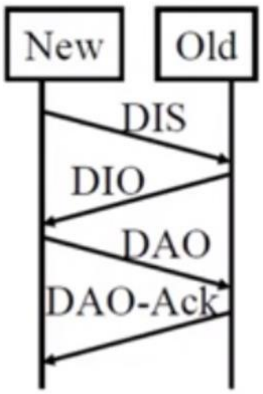


Рисунок 2.2 – Схематичне представлення організації зв’язку між двома вузлами

Кожен вузол у безпровідній сенсорній мережі має значення рангу. Ранг вузла в RPL представлено як скалярне число, яке відображає розташування цього вузла у дереві DODAG. Ранг вузла визначає відносну єдину позицію від вузла щодо вузла базової станції DODAG, і він обчислюється на основі відстаней, що залежить від його сусідів. Також значення рангу можна обчислити як функцію показників споживання енергії, а також враховувати інші властивості або обмеження.

2.3 Методи машинного навчання для виявлення атак відмова в обслуговувані

2.3.1 Метод опорних векторів

З метою розмежування нормальної поведінки від аномальної на основі набору ознак, отриманих із мережевого трафіку Інтернету речей дослідимо алгоритм опорних векторів та штучну нейронну мережу [20-23].

Метод опорних векторів (SVM) – це алгоритм машинного навчання, який широко використовується у системах виявлення вторгнень. Це алгоритм машинного навчання, який використовується як для регресії та класифікації, отримавши на вхід навчальний вектор як вхідні дані та спробувавши розділити дані на класи на основі ознак n-вимірності. Межа, яка розділяє різні класи, називається гіперплощиною, яка розділяє точки даних на різні класи на основі їхніх ознак. Однак слід відзначити, що в одно або багатовимірному наборі даних може існувати кілька гіперплощин. Тому завдання SVM полягає у визначенні найкращого та найбільш оптимального поля гіперплощини для класифікації. На рис. 2.4 зображено гіперплощину, що розділяє два класи.

Для класифікації точок даних, алгоритм SVM має максимізувати відстань між гіперплощиною та точкою даних за допомогою функції втрат, яку можна визначити наступним чином:

$$\omega.x + b = 0\,, \tag{2.11}$$

де b – зміщення; x – вектор точки вхідних даних; а w – вага змінної. Для кожної точки даних існує два варіанти: значення 1 означає, що значення є частиною класу (+), або −1, коли точка даних не є частиною класу (-):

$$\begin{cases} \omega.x + b = 1 \\ \omega.x + b = -1 \end{cases}, \tag{2.12}$$

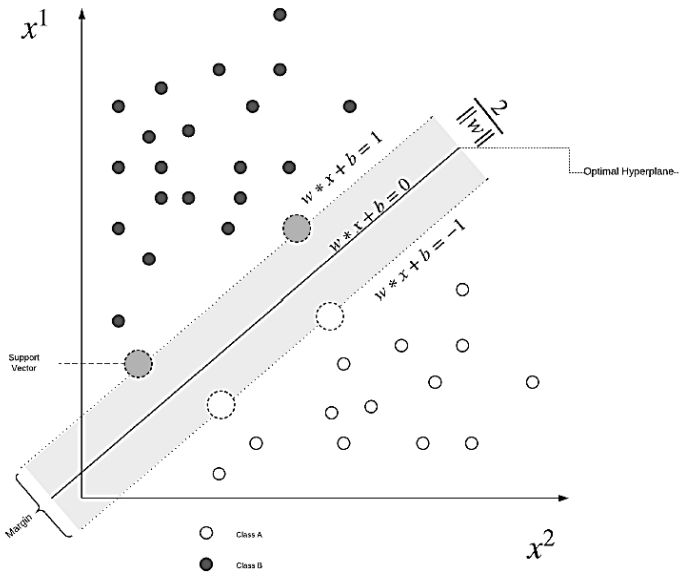


Рисунок 2.4 – Розділова площина методу опорних векторів

Нехай задано набір даних $x_i \in R^d, i = 1,...,t$ де x представляє точку даних (що є вектором, тобто набором ознак), а індекс i, визначає вигляд цього екземпляра, припускаючи, що x_i є частиною одного з двох класів yi $\in \{1,-1\}$. Для цих двох класів у нашому випадку, коли ми намагаємося виявити аномалії в мережевому трафіку Інтернету речей, можна зазначити, що клас 1 представляє нормальний трафік, а -1

представляє аномальний трафік. Тому виходчи із цього вирази 2.11 та 2.12 можна переписати:

$$\begin{cases} \omega^T \cdot x_i + b \geq 1, \forall x_i \in normal \\ \omega^T \cdot x_i + b \leq -1, \forall x_i \in anomaly \end{cases}, \tag{2.13}$$

Якщо множину можна розділити лінійно, тоді можна записати:

$$y_i(\omega^T \cdot x_i + b) \geq 1, \forall i = 1,...,L, \tag{2.14}$$

Відповідно до рис. 2.3, то пунктирна лінія, яка розділяє гіперплощину, називається запасом, який є відстанню між опорним вектором з обох класів, і її можна обчислити, отримавши норму w наступним чином $\frac{2}{||w||}$. Це можна записати як мінімізацію значення $||w||^2$, подібно до максимізації відстані гіперплощини між класами. Таким чином, основна проблема, яку SVM намагається вирішити, це знайти оптимальну гіперплощину. Цього можна досягти шляхом максимізації запасу між гіперплощиною та двома класами, як зазначено раніше, що можна представити наступним рівнянням:

$$min(\omega) = \frac{1}{2}||\omega||^2, \tag{2.15}$$

Наведене вище рівняння є задачею квадратичної оптимізації, яка є основною математичною проблемою, яку намагається вирішити алгоритм SVM. Цю задачу оптимізації найкраще використовувати із наборами, які мають чіткий розподіл між класами, які можна розділити лінійно. Однак для складних наборів, які не можна

виокремити лінійно, до яких відноситься задача аналізу даних, зібраних у мережах Інтернету речей, буде важко їх класифікувати. Тому для розв'язання цієї задачі використаємо вираз Лагранжа та двоїсту задача Вульфа. Тоді представлення Лагранжа для SVM виглядатиме наступним чином:

$$\mu(\omega,b,a)=\frac{1}{2}\omega\cdot\omega-\sum_{i=1}^{m}\alpha_i[\,y_i(\omega\cdot x+b)-1], \tag{2.16}$$

де α – множник Лагранжа, застосування якого полягає в тому, щоб максимізувати його для кожного екземпляра точки даних x_i та мінімізувати значення $\frac{1}{2}||w||^2$.

Класифікуюча функція F набуває вигляду $F(x)=sign(\langle w_1\varphi(x)\rangle+b)$. Вираз $k(x,x')=\langle\varphi(x),\varphi(x')\rangle$ називається ядром класифікатора. З математичної точки зору ядром може бути будь-яка позитивно визначена симетрична функція двох змінних. Позитивна визначеність необхідна у тому, щоб відповідна функція Лагранжа у задачі оптимізації була обмежена знизу, тобто завдання оптимізації було б коректно визначено. В даній роботі використаємо радіальну базисну функцію $K(x_i,x_j)=exp(-\gamma||x_i-x_j||^2)$, де γ, використовується для визначення того, як ядро має відповідати даним. Невелике значення γ спричинить недостатню підгонку моделі, в результаті чого модель буде лінійна. Велике значення γ зробить модель перенавченою з хорошою точністю, але з поганим узагальненням.

2.3.2 Штучні нейронні мережі

Нейронні мережі використовуються для додатків безпеки протягом тривалого часу і можуть застосовуватися в різних секторах безпеки в мережах Інтернету речей. У даній роботі для виявлення аномалій дослідимо один із видів ШНМ, який

називається багатошаровим персептроном. Алгоритм нейронних мереж вважається одним із найкращих алгоритмів для прогнозування та класифікації. Причина цього очевидна, оскільки вона відображає нейронну систему людини, яка допомагає людям думати та приймати щоденні життєві рішення. Структура нейронної мережі є багатошаровою структурою з вхідним і прихованим шарами в багатошаровому персептроні (MLP). Простий MLP називається одношаровою нейронною мережею персептрона. Це нейронна мережа з одним вхідним і одним вихідним шарами, які можна використовувати для простої класифікації. Однак він не підтримує поглиблену класифікацію, коли для комплексної обробки вводяться кілька прихованих шарів.

Кількість атрибутів визначає вхідний рівень, який представляє кількість об'єктів або класів набору даних. Вихідний рівень має кількість входів/2 виходів. Також кожному нейрону присвоєно певне числове значення, яке називається вагами, а вхід кортежу даних називається вхідними значеннями (функціями). Для опрацювання інформації слід обчислити вихідне значення наступного нейрона, яке можна обчислити шляхом множення ваги та вхідного значення:

$$вихід = вхід \cdot ваги + зміщення\ , \tag{2.17}$$

Це значення розглядається як значення виходу наступного нейрона або значення вихідного рівня. Однак, залежно від вхідних нейронів, обчислюються всі вхідні значення, і найбільше значення вважається виходом цієї нейронної мережі. Загальна формула для одного персептрона виглядатиме наступним чином:

$$вихід = b \cdot x_i \cdot w_i\ , \tag{2.18}$$

Існує ще один тип нейронної мережі, яка дає більш точніші вихідні результати – багатошаровий персептрон. Він більш точно відображає структуру нейронної

мережі людини. Він представляє концепцію прихованого шару, яка додатково забезпечує класифікацію атрибутів, забезпечуючи ти самим не лінійність результату. Кількість прихованих шарів повністю залежить від гнучкості даних або обсягу даних. Чим більше буде прихованих шарів, тим точнішими будуть результати, проте відповідно зростатиме і складність роботи алгоритму.

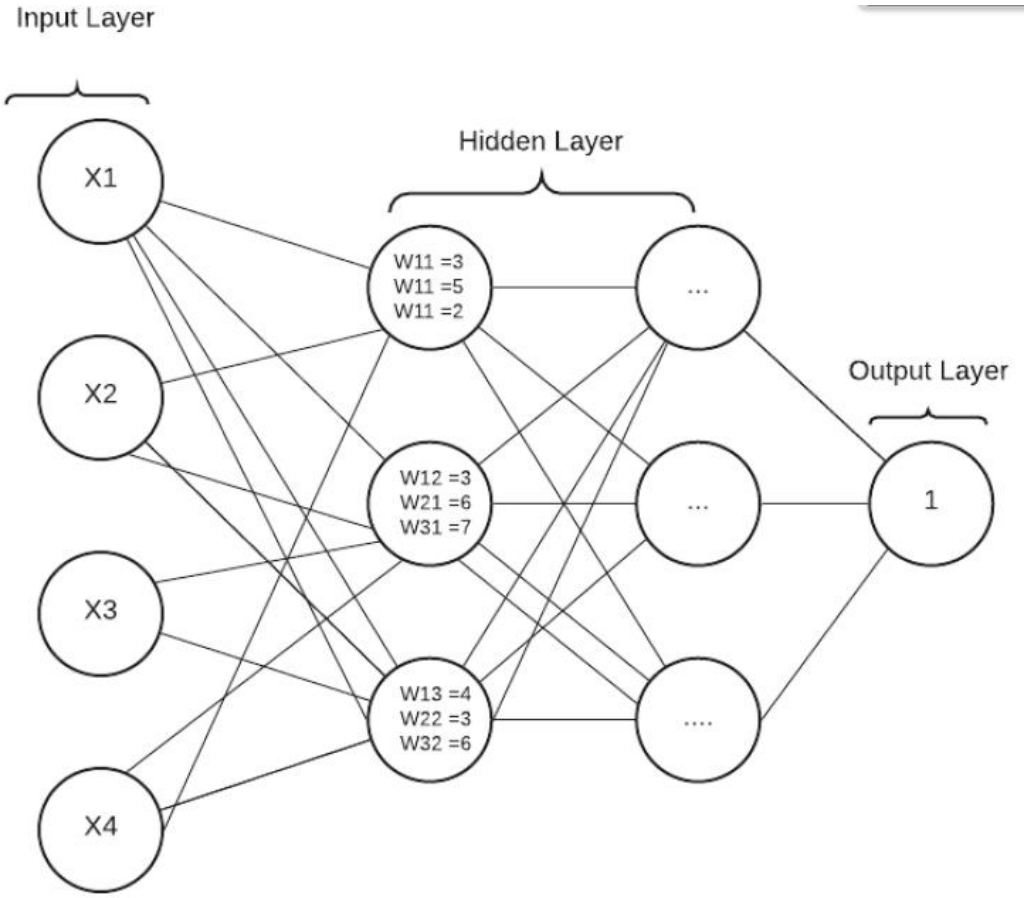


Рисунок 2.5 – Схематичне представлення багатошарового персептрона

Як видно з рисунка 2.5, є чотири вхідні нейрони, які подаються в прихований шар із трьома нейронами. Кожен прихований шар представлений значенням *Wn*, де n – кількість прихованих шарів. Також можна відмітити, що кожне з'єднання між нейронами має вагу (w31 = 7). Якщо звернути увагу на вхід x2, який з'єднується з нейронами w31, які мають найвище значення 7, то нейрони в цьому випадку

вважатимуть, що нейрон X2 є найважливішою ознакою із поміж інших ознак, оскільки він має найвищу вагу.

Окрім вагів нейронні мережі характеризуються також вектор зсуву (bias), що використовується разом із вагою для модифікації результату, і обидва вони допомагають моделі нейронної мережі точно підібрати дані для отримання найкращого результату.

Кожен нейрон робить внесок у нейронну мережу, вкладаючи невеликий результат у загальне підсумкове рішення. Цей процес називається функцією активації, а процес агрегування всіх цих малих внесків представлений z. В нашому досліджені в якості функції активації використаємо сигмо їд, що представлений наступним рівнянням:

$$f(x) = 1/(1 + e^{-x} \cdot z), \tag{2.19}$$

2.4 Висновки

Представлена узагальнена теоретико-множинна модель процесу описує послідовний процес обробки та збору інформації із мережевого трафіку з метою виявлення розподілених атак відмова в обслуговуванні та із подальшою ізоляцією скомпрометованих вузлів. Вхідними даними для представленої моделі процесу є вхідний трафік, тоді як вихідними даними новий маршрут, що представляє нове дерево DODAG у протоколі RPL, у якому будуть відсутні скомпрометовані вузли.

3 СИСТЕМА І МЕТОД ЗБОРУ ДАНИХ ПРОТОКОЛУ МАРШРУТИЗАЦІЇ RPL ДЛЯ ВИЯВЛЕННЯ РОЗПОДІЛЕНИХ АТАК ВІДМОВА В ОБСЛУГОВУВАННІ

3.1 Узагальнена структура системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні

Завдання збору даних у мережах Інтернету речей є одним напрямків процесу зворотної розробки (reverse engineering) та може бути імплементоване з метою виконання двох основних функцій: аналізу зібраних даних з метою підвищення ефективності взаємодії між пристроями в мережі або з метою здійснення діагностики мережі на предмет пошуку несправностей [23-28]. В свою чергу одним із основних напрямків діагностики мереж є аналіз даних мережевого трафіку на предмет виявлення зловмисної активності або впливу кібератак. Це дозоляє реалізувати одну із головних вимог що ставиться до інфраструктури Інтернету речей – забезпечення її безпеки функціонування з точки зору здатності протидії впливу зловмисного програмного забезпечення та кібератак. В даній роботі представлено систему збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні у мережах Інтернету речей. Основу функціонування запропонованої системи складають три основних модулі (рис. 3.1):

- Модуль збору даних (МЗД);
- Модуль класифікації (МК);
- Модуль виявлення (МВ).

Модуль збору даних (МЗД) можна розглядати як міжфазний модуль, оскільки він залучається у двох фазах методу: попереднього навчання та після навчання. Модуль виявлення та модуль агента вузла є частиною фази після навчання та відповідають за виявлення атак і формування реакції протидії. Крім того, на цьому етапі відбувається моніторинг трафіку, класифікація даних та ізоляція зловмисних вузлів. Кожен модуль системи докладніше буде розглянуто у наступних пунктах цього розділу. Узагальнену

схему систему збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні у мережах Інтернету речей наведено на (рис. 3.1).

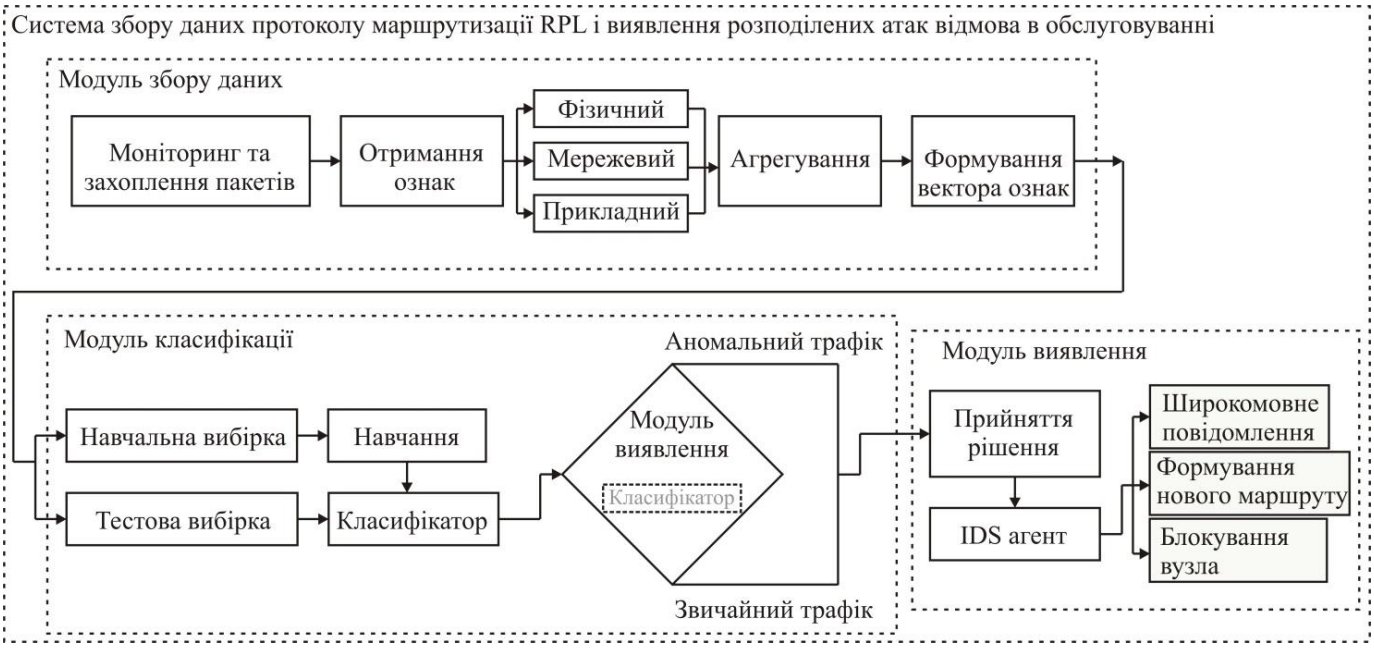


Рисунок 3.1 – Архітектура системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні

3.1.1 Модуль збору даних

Основною метою цього модуля є збір даних у реальній мережі Інтернету речей (або у модельованій мережі), що функціонує на основі протоколів 6LoWPAN і RPL. Слід відзначити, що запропонована архітектура системи збору даних і виявлення розподілених атак відмова в обслуговуванні не обмежується даними протоколами і може бути узагальнена та розширена і для інших протоколів обміну даними в мережах Інтернету речей.

Перш ніж здійснити виявлення будь-якої кібератаки, слід отримати ознаки (features) із мережі, що дозволили б ідентифікувати появу аномалій.

Дані, що передаються через будь-яку мережу, відповідають певним протоколам (що є набором правил для організації взаємодії по мережі).

У даній системі пропонується використати ознаки із трьох логічних рівнів: фізичного, мережевого та прикладного рівнів.

Розглянемо детальніше ознаки, які використовуватимуться у даній роботі.

Опрацювання ознак фізичного рівня, зокрема таких як прийняті та передані сигнали на рівні MAC, пов'язано із атаками глушіння фізичного рівня (jamming attacks), що перешкоджають пересилці сигналам у мережі.

Збір ознак мережевого рівня є вирішальним для інтегрованої системи виявлення вторгнень, оскільки отримані ознаки тісно пов'язані із багатьма відомими атаками (наприклад атаки вибіркового пересилання пакетів та black hole атака).

Із пакетів цього рівня виділяються такі ознаки, як кількість повідомлень DIS і DIO (вхідні та вихідні контрольні пакети), середнє значення очікуваної кількості передач, рівень вузла RPL (rank), кількість змін рівня із плином часу та кількість вузлів-сусідів у дереві DODAG.

На прикладному рівні даний модуль збирає специфічну для програми інформацію, таку як рівень потужності вузла та температура.

Ознаки прикладного рівня, можна отримати шляхом програмування вузлів для розрахунку споживаної потужності електроенергії та інших пов'язаних функцій.

У таблиці 3.1 наведено ознаки, що отримуються модулем збору даних із фізичного, мережевого та прикладного рівнів у мережі Інтернету речей.

На рисунку 3.2 показано процес отримання окремих ознак із файлу pcap, отриманого із мережі Інтернету речей. Процес вилучення ознак передбачає послідовне отримання ознак із кожного рівня та збереження їх до бази даних з метою їх подальшого опрацювання.

Таблиця 3.1 Ознаки, що отримані із фізичного, мережевого та прикладного рівнів

Ознака	Опис
Ознаки фізичного рівня	
Отриманий сигнал DBM	Середнє значення прийнятого сигналу на рівні MAC
Переданий сигнал DBM	Середнє значення переданого сигналу на рівні MAC
Показник рівня отриманого сигналу RSSI	Середнє значення показника рівня шуму RSSI
Інтервал широкомовного індикатора мережі	Середнє значення широкомовного індикатора мережі
Ознаки мережевого рівня	
Індикатор якості зв’язку (LQI) (на основі потужності отриманого сигналу (RSS))	Значення якості зв’язку
ETX	Середнє значення очікуваної кількості передач
Кількість повідомлень DIS	Кількість повідомлень DIS
Кількість повідомлень DIO	Кількість повідомлень DIO
Рівень RPL	Кількість змін рівня із плином часу
Кількість вузлів-сусідів	Кількість вузлів-сусідів
Ознаки прикладного рівня	
Температура	Середнє значення температури
Вологість	Середнє значення вологості
Рівень потужності	Середнє значення рівня потужності
Ідентифікатора вузла (Node ID)	Ідентифікатора вузла (Node ID)

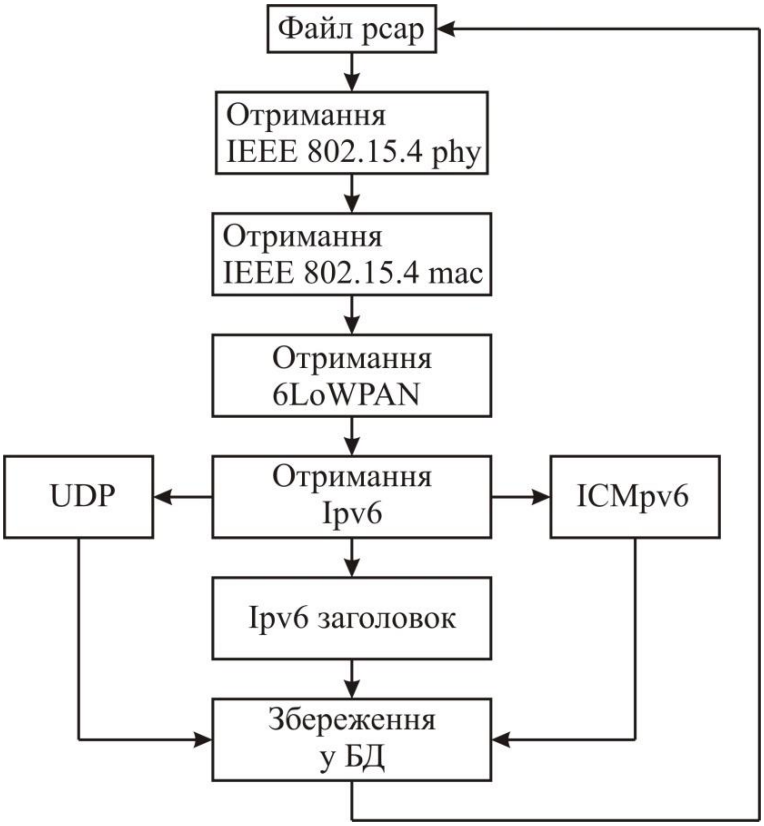


Рисунок 3.2 – Процес отримання ознак із мережевого трафіку Інтернету речей

Крім того, перед вилученням ознак слід визначити часове вікно для агрегування даних у записи. Це часове вікно буде використано пізніше для отримання відхилення або середнього значення для заданої ознаки. Залежно від мережевої програми Інтернету речей часове вікно може змінюватися відповідно. Оскільки кожна програма має різний рівень даних, створених під час зв'язку, визначення часового вікна залежить від типу використовуваної програми IoT, а також типу реалізованих протоколів.

Таким чином в результаті опрацювання мережевого трафіку модулем збору даних буде отримано набір даних на основі протоколів RPL та LoWPAN, який буде використано для навчання та тестування алгоритму машинного навчання та створення моделі виявлення (фаза попереднього навчання). Також слід відзначити, що ідентичні кроки по відбору ознак проводяться і для фази після навчання, коли буде використана

створена модель машинного навчання для аналізу невідомої активності в режимі реального часу.

Для отримання необробленого мережевого трафіку Інтернету речей використаємо програмний засіб sensniff для покриття всієї мережі.

Програмний засіб sensniff використовується для захоплення та аналізу мережевого трафіку у режимі реальному часі для мереж IEEE 802.15.4 та складається із двох компонентів:

– периферійний пристрій: це вбудований пристрій із трансивером, який захоплює всі мережеві кадри та передає їх на хост;

– хост: це скрипт на Python, який працює на робочій станції. Він зчитує мережеві пакети, захоплені периферійним пристроєм, перетворює їх у PCAP формат і передає дані на Wireshark. Окрім захоплення мережевих пакетів, хост може надсилати команди периферійному пристрою для досягнення додаткової функції, наприклад, змінити радіоканал.

3.1.2 Модуль класифікації

Набір даних, згенерований модулем збору даних, використовуватиметься для навчання та тестування алгоритмів машинного навчання. На цьому рівні виконується аналіз різних методів машинного навчання, а також здійснюється вибір того алгоритму, який має найкращі результати з точки зору ефективності та достовірності виявлення атак. Якість результатів, отриманих на цьому етапі, сильно залежить від того, як дані були зібрані на попередньому етапі. На рисунку 3.3 показано схематичне зображення процесу навчання та перевірки алгоритму методу опорних векторів для виявлення кібератак.

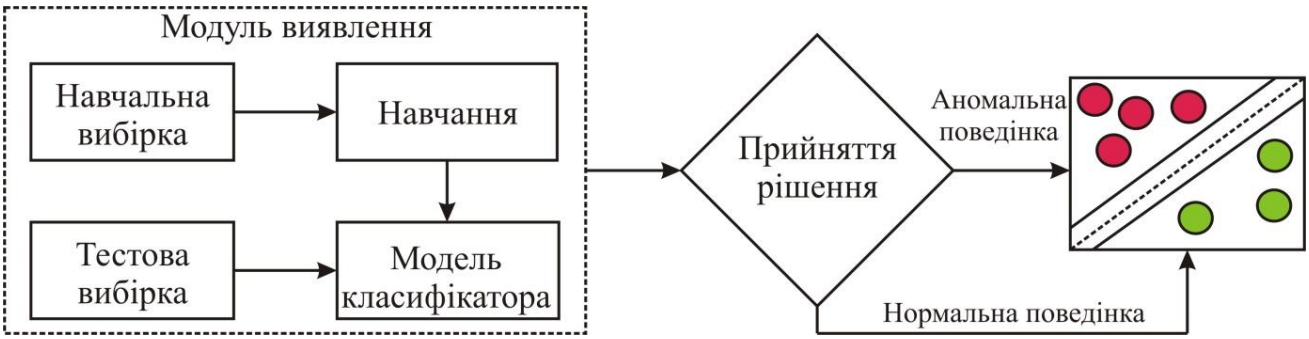


Рисунок 3.3 – Процес проведення навчання моделі

3.1.3 Модуль виявлення

Цей модуль працює як точка з'єднання між локальною мережею та системою виявлення розподілених атак відмова в обслуговувані на інфраструктуру Інтернету речей. Він побудований на вершині базової станції мережі (sink node), оскільки всі вузли підключені до базової станції або безпосередньо, або на відстані кількох переходів (hop). Основна функція цього модуля полягає в трансляції повідомлення про аномальну поведінку на решту вузлів IoT мережі, що містять ідентифікатор зловмисника та шлях до зловмисника. Це дозволить іншому незачепленому вузлу додати вузол зловмисника до чорного списку та уникнути будь-якого зв'язку зі зловмисним вузлом. Крім того, агент IDS змінює маршрут вузла-жертви та створює новий альтернативний шлях до вузла-приймача. Потім агент IDS ініціює реконфігурацію топології мережі, щоб ізолювати зловмисний вузол шляхом встановлення нового маршруту до приймача від вузла-жертви. Усі вузли заносять у чорний список шкідливий вузол, а весь мережевий трафік від нього ігнорується та відкидається. Схематичне зображення роботи модуль виявлення представлено на рис. 3.5. На даній схемі позначення S відповідає базовій станції, а m – скомпрометованому вузлу. Для досягнення заданого результату модуль агенту IDS реалізує дві основні функції:

– AlertBroadcast(AttackerNodeID): основною функцією цього методу є трансляція попереджувальних повідомлень, що містять AttackerNodeID, усім вузлам

у мережі. Цього можна досягти, використовуючи повідомлення DIO, що доступне поверх протоколу маршрутизації RPL;

– RouteModify(): дана функція несе відповідальність за перенаправлення мережевого трафіку, який надходить до вузла-жертви та від нього, використовуючи найближчого сусіда із найвищим рангом у дереві DODAG.

–

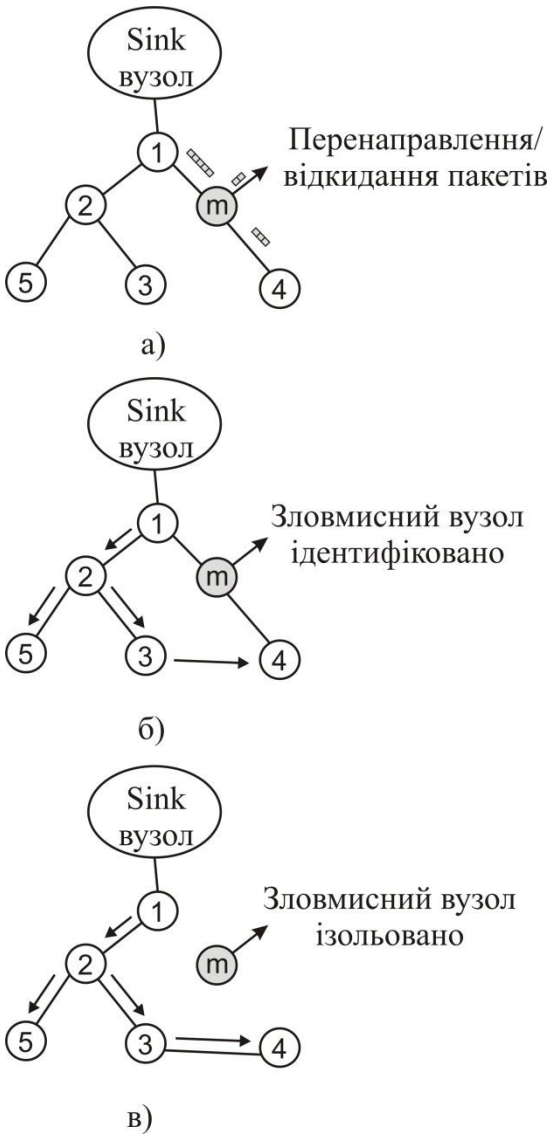


Рисунок 3.4 – Схематичне зображення роботи модуль виявлення: а) прояв зловмисної активності вузлом m; б) ідентифікація зловмисного вузла; в) ізоляція зловмисного вузла

3.2 Метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні

В основі запропонованої системи закладено метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні, що включає дві основні фази: фазу попереднього навчання та фазу після навчання. Розглянемо детальніше запропонований метод.

На рисунку 3.5 показано дві фази запропонованого методу та пов’язані із ними завдання, що виконуються в рамках кожної фази.

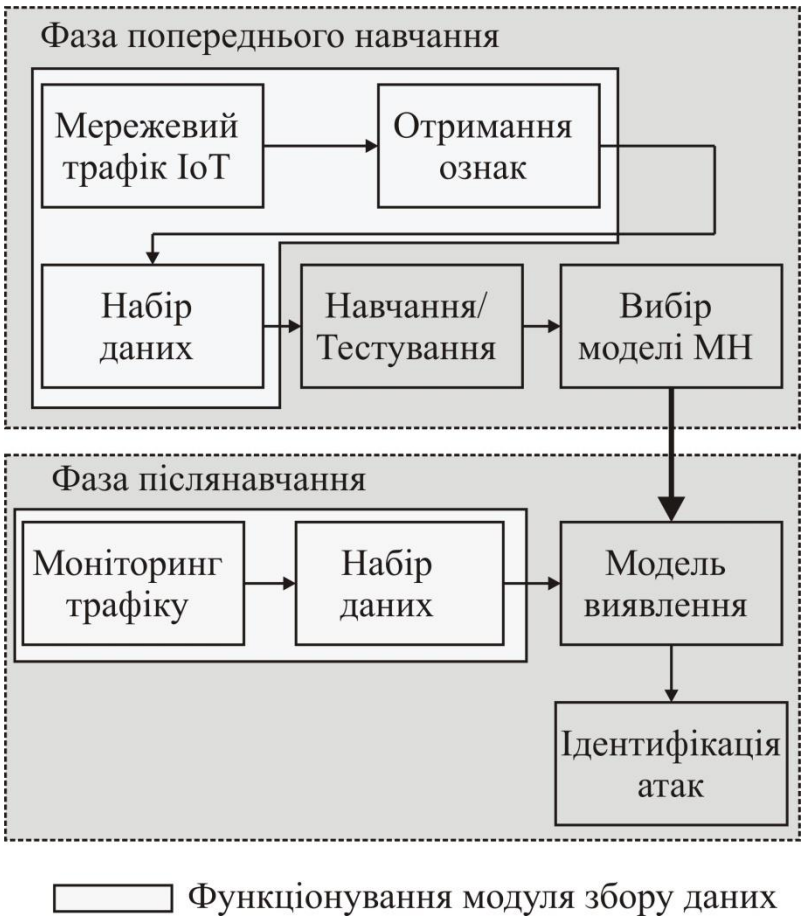


Рисунок 3.5 – Функціонування методу (фаза попереднього навчання та фаза після навчання) та місце у ній модуля збору даних

Слід відзначити, що модуль збору даних використовується на обох етапах, починаючи зі збору даних для навчання системи, і закінчуючи моніторингом трафіку в режимі виявлення атак.

На етапі попереднього навчання модуль збору даних використовується для збору офлайн-даних, що використовуються для навчання та тестування алгоритмів машинного навчання.

На етапі після навчання модуль збору даних використовується для глибокої перевірки пакетів (deep packet inspection), а також для моніторингу мережевого трафіку на основі сформованої на попередній фазі моделі машинного навчання.

3.2.1 Фаза попереднього навчання

На етапі попереднього навчання модель машинного навчання навчається та тестується на основі зібраних даних МЗД. У даній роботі буде досліджено два алгоритми машинного навчання та проведено набір тестів для визначення найефективнішої моделі. Слід відзначити, що опрацювання моделей машинного навчання здійснюється на основі отриманих даних МЗД.

На рис 3.6 показано робочий процес вибору найкращої моделі для кожного вибраного методу машинного навчання. Процес вибору найкращого методу машинного навчання можна описати наступними кроками:

1. Вибір алгоритму: перед навчанням моделі необхідно вибрати тип машинного навчання. Загальну базу алгоритмів складають метод опорних векторів та штучна нейронна мережа. Слід відзначити, що даний набір може бути розширений, шляхом додавання інших алгоритмів машинного навчання.

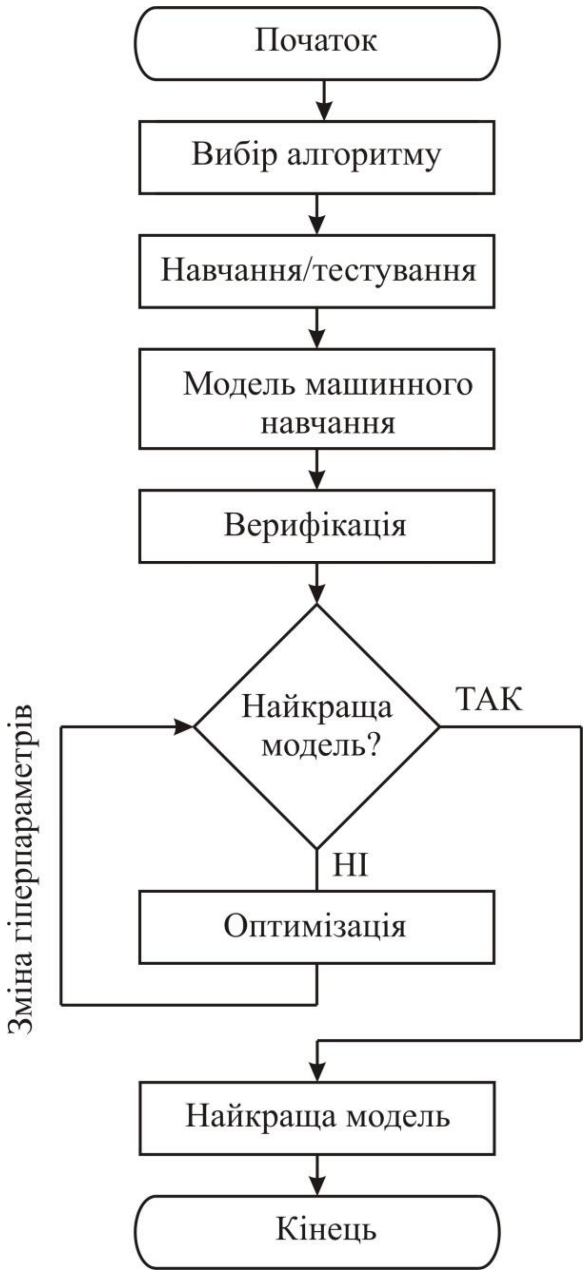


Рисунок 3.5 – Робочий процес вибору найкращої моделі для кожного вибраного методу машинного навчання

2. Навчання/тестування: це фаза навчання для моделі машинного навчання, на якій дані передаються в обраний алгоритм для створення моделі машинного навчання.

3. Перевірка: на цьому етапі модель перевіряється за допомогою набору атрибутів і оцінок.

4. Оптимізація: на цьому кроці задана модель повторюється кілька ітерацій із іншим набором гіперпараметрів. Зазначені кроки повторюються допоки не буде отримано найоптимальніший модель для заданого алгоритму машинного навчання.

Наприкінці цих кроків генеруються дві оптимізовані моделі машинного навчання. На основі результатів на етапі верифікації буде обрано найкращу модель, яка і буде розгорна в інтелектуальній системі виявлення розподілених атак відмова в обслуговувані на інфраструктуру Інтернету речей.

3.2.2 Фаза після навчання

Фаза після навчання відповідає за обробку даних і виконання активностей у режимі реальному часі. Роботу методу збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні у фазі після навчання подамо у вигляді наступної послідовності кроків:

1. Агрегація трафіку. Основна мета цього кроку – зібрати дані з кількох сніферів, що встановленні у мережі Інтернету речей. Підтримка кількох сніферів у мережі має важливе значення для забезпечення масштабованості мережі та покриття виявлення атак, особливо, якщо мова йде про розподілені атаки, що націлені на декілька вузлів. Тому процес агрегування повинен гарантувати, що дані, які надходять у процес отримання ознак, не дублюються.

Агрегування даних виконаємо відповідно до наступної формули:

$$f(\{timestamp, ID\}_{s_j}, L) = \begin{cases} 1, if\ (timestamp == L[i].timestamp \wedge ID == L[i].ID \\ 0, в\ іншому\ випадку \end{cases} \quad (3.1)$$

де timestamp – часова мітка отримання пакету, ID – ідентифікатор вузла, s_j j сніфер ($S = \{s_j\}_{j=1}^{N}$, N – кількість сніферів), L – черга пакетів.

Процес забезпечення того, що вузол аналізатора агрегує лише недубльовані дані, полягає в тому, щоб порівняти підписи отриманих пакетів у момент часу *T* і переконатися, що вони не збігаються. Далі якщо відбулось спів падіння, здійснюється перевірка по значенню ідентифікатора вузла. Таким чином, визначимо сигнатуру даних як змінну *Ts*, яка складається із мітки часу та ідентифікатора вузла:

$$T_S = \{ timestamp, ID \} \tag{3.2}$$

Якщо підпис пакета дорівнює будь-яким пакетам, отриманим від будь-якого іншого сніфера, один із пакетів буде проігноровано, і лише одну версію пакета буде додано до черги. В іншому випадку додаткова процедура не потрібна, і пакети пересилаються до наступного набору. Цей процес гарантує відсутність дублювання даних у режимі реальному часі. Цей блок складається з двох обробників:

– Обробник аналізатора (Sniffer[]): цей обробник приймає масиви аналізаторів як вхідні дані та обробляє кожен пакет, що надходить від кожного аналізатора. Вихід від цього обробника передається в обробник агрегації. Результатом є багатовимірний масив сніфферів і пакетів, пов’язаних з кожним сніфером;

– Обробник агрегації (Sniffers[node][packet]): обробник агрегації проходить по всіх аналізаторах та їх пакетах, для пошуку схожості пакетів. Після цього процесу в чергу надсилається лише один масив пакетів вузлів.

2. З метою забезпечення виділення ознак на основі часового вікна *wt* та формування порції даних (вектора ознак) використовується формування черги. Часові вікна визначаються на основі типу використовуваного застосунка Інтернету речей. Надання такої гнучкості у виборі часового вікна є важливим для забезпечення масштабованості всієї системи. Опишемо процес додавання пакетів до черги наступним алгоритмом:

Алгоритм 3.1 - Додавання у чергу

Вхід: Q – черга пакетів, packets
Вихід: Q
packets = вхідні пакети
while (packets ≠ 0) do
  tmp = packet[0].timestamp – packet[1].timestamp
  Q.add(tmp)
end

Слід відзначити, що часові вікна wt, які використовуються в процесі попереднього навчання, мають бути однаковими, щоб забезпечити точний та неупереджений збір даних.

3. Вилучення ознак. У даному кроці застосовується такий самий процес вилучення ознак, що й у фазі попереднього навчання (рис. 3.3), але замість офлайнового режиму процес відбувається онлайн у сценаріях реальної мережі Інтернету речей.

4. Класифікація атак: це крок, на якому сформована модель машинного навчання класифікує аномалії у мережевому трафіку Інтернету речей. Вибір моделі машинного навчання здійснюється відповідно до фази попереднього навчання. Таким чино найкраща модель з точки зору достовірності виявлення машинного навчання буде вбудована в структуру системи виявлення розподілених атак відмова в обслуговувані.

5. Генерації результатів. На цьому кроці здійснюється генерація результату виявлення та створення й надсилання спеціального UDP пакету агенту виявлення. Варто зазначити, що згенерований пакет має мінімальний розмір, щоб не забивати каналу зв’язку службовим трафіком. Дайни пакет надсилається за допомогою корисного навантаження в пакеті UDP. Пакет містить такі параметри як ідентифікатор вузла, часова мітка, батьки вузла, ранг і результат виявлення. Результат виявлення може бути 0 або 1. Якщо результат дорівнює 0, то це вказує на те, що жодної атаки не виявлено, і подальші пакети не будуть надіслані агенту

виявлення. Потім цей результат зберігається в локальному репозитарії. В іншому випадку, якщо результат виявлення дорівнює 1, пакет виявлення надсилається агенту виявлення, а копія результату також зберігається в локальному сховищі з метою подальшого дослідження та аналізу.

Основною причиною створення локального репозитарію (бази даних) є створення бази знань для майбутнього аналізу та вдосконалення моделі. Наприклад, використовуючи дані із локального депозитарію за допомогою сторонніх програм верхнього рівня візуалізувати в режимі реального часу, як працює мережа.

Для реалізації цієї концепції потрібно два обробники, функції яких будуть розподілені таким чином, що:

– GetData() Handler: відповідає за вилучення даних із сховища даних і публікації їх в потрібній програмі за допомогою REST API;

– Обробник PostData(): відповідає за введення даних із джерел до бази даних. Цей обробник зазвичай запускається блоком вилучення ознак і блоком генерації результатів для збереження інформації в базі даних.

3.3 Висновки

Виявлення розподілених атак відмова в обслуговані у мережах Інтернету речей є складною проблемою, особливо в промислових додатках IoT. Через наслідки, пов’язані із такою атакою, яка впливає на доступність мережі та її послуг, вони спричиняють значні фінансові втрати, а також порушення конфіденційності та доступності інформації. Тому для постачальників і кінцевих користувачів IoT мереж вкрай важливо швидко та ефективно виявляти та пом’якшувати такі види атак.

Запропоновано структуру системи та кроки методу збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні, яка складається з трьох основних модулів: модуль збору даних, модуль класифікації та модуль виявлення. Модуль збору даних можна розглядати як міжфазний модуль,

оскільки він залучається у двох фазах функціонування методу: попереднього навчання та після навчання. Модуль виявлення та модуль агента вузла є частиною фази після навчання та відповідають за виявлення атак і формування реакції протидії. Крім того, на цьому етапі відбувається моніторинг трафіку, класифікація даних та ізоляція зловмисних вузлів. Головною особливістю модуля збору даних було те, що збір даних забезпечувався декількома сніферами, що встановлені у мережі, і з подальшою агрегацією зібраних даних.

4 МОДЕЛЮВАННЯ МЕРЕЖІ ІОТ В ОС CONTIKI, ОТРИМАННЯ ДАНИХ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМИ ЗБОРУ ДАНИХ ПРОТОКОЛУ МАРШРУТИЗАЦІЇ RPL ДЛЯ ВИЯВЛЕННЯ РОЗПОДІЛЕНИХ АТАК ВІДМОВА В ОБСЛУГОВУВАННІ

4.1 Операційна система Contiki та Cooja симулятор

Розробка будь-якого методу чи системи, що використовує методи машинного навчання, вимагає збору даних, пов’язаних із сферою дослідження. У порівняні із традиційними мережами для IT інфраструктур, для яких створено значні набори даних (наприклад KDD Cup[25]), що дозволяють реалізувати навчання методів машинного навчання, для IoT мереж на сьогоднішній день не має такого попередньо визначеного набору даних. Тому для перевірки достовірності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих запропонованою системою пропонується розгорнути інфраструктуру на основі операційної системи Contiki та Cooja симулятора.

Contiki OS [26] – це операційна система із відкритим кодом, спеціально розроблена для симуляції сенсорних пристроїв. ContikiOS має функції ядра керування подіями, превентивну багатопотоковість, а також повністю підтримує інтеграцію стека протоколів TCP/IP.

Симулятор Cooja був вперше представлений у технічному звіті та може симулювати рідну платформу операційних систем ContikiOS і TinyOS. Основна перевага симулятора Cooja полягає в тому, що він може імітувати сенсорний вузол на основі його реальних характеристик використовуючи Java Native Interface (JNI) для виконання програмного коду ContikiOS і TinyOS. JNI забезпечує взаємозв’язок між програмним кодом на мові C (зазвичай саме ця мова програмування використовується для прошивки сенсорних вузлів) і віртуальною машиною Java. Таким чином, симулятор Cooja може імітувати будь-який сенсорний вузол платформи, що

максимально схожий на реальний сенсорний вузол, що функціонує у мережі Інтернету речей.

4.2 Отримання даних для перевірки достовірності виявлення розподілених атак відмова в обслуговуванні

Для отримання набору даних для проведення експериментів було розгорнуто інфраструктуру на основі операційної системи Ubuntu та симулятора Cooja. Схематичне представлення досліджуваної 6LoWPAN-RPL мережі наведено на рис. 4.1.

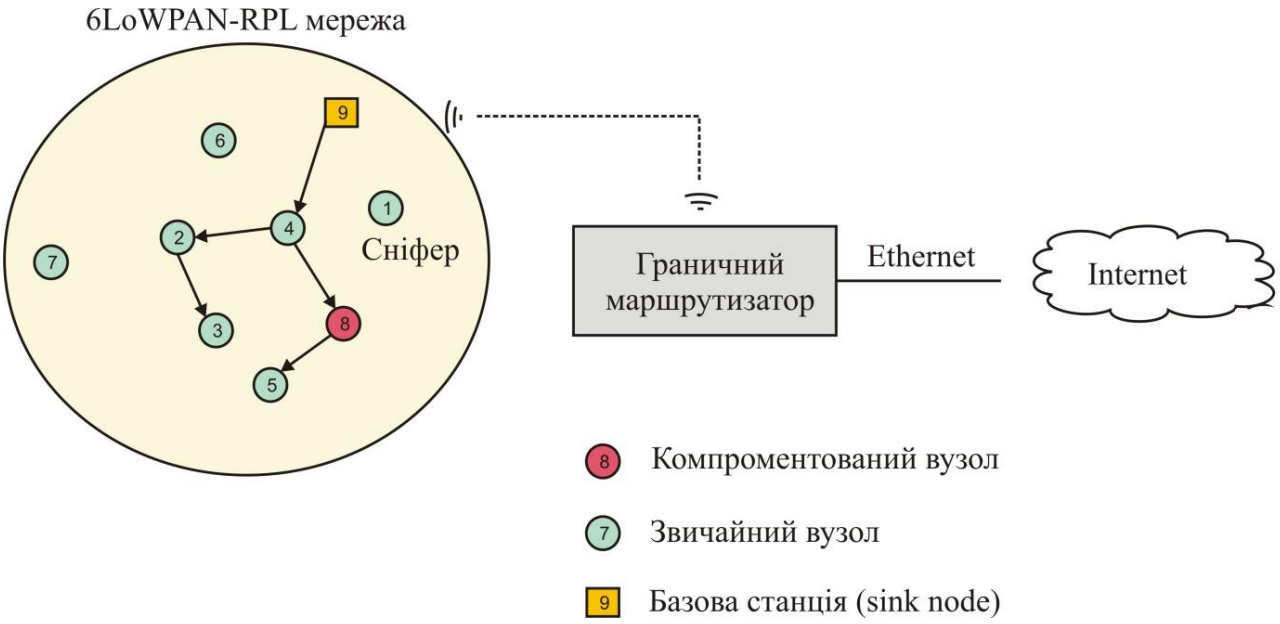


Рисунок 4.1 – Досліджувана 6LoWPAN-RPL мережа

Процес отримання даних для проведення дослідження проводився в Cooja симуляторі. В результаті моделювання у середовищі Cooja вузлам-давачам надаються обчислювальні можливості у вигляді потужностей центрального процесора та пам’яті.

Усі параметри для досліджуваної 6LoWPAN-RPL мережі приведені у таблиці 4.1.

Таблиця 4.1 Параметри середовища для процесу моделювання в Cooja

Параметр	Значення
Модель бездротового каналу	UDGM
Кількість вузлів	21
Протокол маршрутизації	RPL
Транспортний протокол	UDP
MAC протокол	CSMA + ContikiMAC
Розмір мережі	50 x 100 метрів
Тип вузлів	Zolteria Z1
Час моделювання	3 години

В якості вузлів для симуляції використано давач SHT11, що збирає дані про температуру. Параметри вузлів, що використовувалися при моделюванні наведено у таблиці 4.2.

Таблиця 4.2 Параметри вузлів для процесу моделювання в Cooja

Параметр	Значення
Тип вузлів	Zolertia Z1
CPU	16 bit RISC
Оперативна пам’ять	8 Кб
Флеш пам’ять	92 Кб
Мікросхема передавача	CC2420
Живлення	3.3/5 В
Вузол	STH21
Безпровідне з’єднання	IEEE 802.15.4, 2.4 ГГц

При моделюванні мережі на основі протоколу RPL усі давачі (звичайні вузли і вузли, що представляють базові станції) використовують один і той самий тип мота – Z1 (рис. 4.2). Z1 оснащений мікроконтролером другого покоління MSP430F2617 із низьким енергоспоживанням, який оснащений потужним 16-розрядним процесором RISC із тактовою частотою 16 МГц, вбудованим заводським калібруванням годинника, 8 КБ оперативної пам’яті та 92 КБ флеш-пам’яті. Також даний пристрій включає трансивер CC2420, сумісний зі стандартом IEEE 802.15.4, який працює на частоті 2,4 ГГц із ефективною швидкістю передачі даних до 250 Кбіт/с. Обладнання Z1 забезпечує максимальну ефективність і надійність при низькому значенні витраченої енергії.

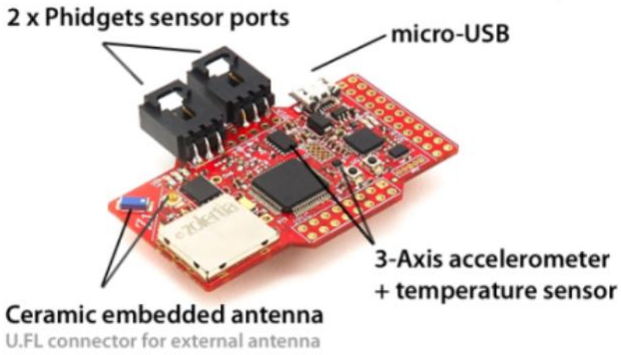


Рисунок 4.2 – Безпровідний сенсорний пристрій Zolertia Z1 для вимірювання температури у мережах 6LowPAN-RPL

На всіх вузлах мережі розгорнуто програмне забезпечення для вимірювання температури та вологості. Налаштування файлу конфігурації для давача Z1 наведено у наступному лістингу:

```
#include "contiki.h"
#include "dev/sht11/sht11.h"
#include <stdio.h>
PROCESS(test_sht11_process, "SHT11 test");
```

```
AUTOSTART_PROCESSES(&test_sht11_process);
PROCESS_THREAD(test_sht11_process, ev, data)
{
 static struct etimer et;
 static unsigned rh;
 PROCESS_BEGIN();
 sht11_init();
 for(etimer_set(&et, CLOCK_SECOND);; etimer_reset(&et)) {
  PROCESS_YIELD();
  printf("Temperature:   %u degrees Celsius\n",
       (unsigned)(-39.60 + 0.01 * sht11_temp()));
  rh = sht11_humidity();
  printf("Rel. humidity: %u%%\n",
       (unsigned)(-4 + 0.0405 * rh - 2.8e-6 * (rh * rh)));
 }
 PROCESS_END();
}
```

Вузли мережі здійснюють надсилання даних на базову станцію із інтервалом 20 секунд. Таким чином у модельованій мережі присутні два типи вузлів: вузол базова станція (sink node) та звичайний вузол (node).

Вузол базова станція або кореневий вузлом є вузлом куди пересилаються всі мережеві пакети. Sink node має найвищий ранг у мережі, оскільки він підтримує ієрархію зв’язків між вузами у мережі RPL. Окрім того даний вузол відповідає за збереження маршрутів. На рівні додатків даний вузол працює як UDP сервер, обробляючи всі вхідні UDP пакети від вузлів клієнта. Окрім того, базова станція працює як міст між мережею RPL та граничним маршрутизатором. Функціональність

базової станції розроблена із використанням основних функцій, доступних у Contiki OS.

Звичайний вузол: це сенсорний вузол, який визначає температуру та передає значення базовій станції. Разом із даними про температуру цей вузол також надсилає іншу цінну інформацію про свій статус, наприклад рівень RSSI, LQI (індикатор якості зв'язку) та значення ETX (очікувана кількість передач), інкапсульовані як корисне навантаження в пакетах UDP.

Для організації бездротового зв'язку обидва види вузлів реалізують протокол 6LoWPAN поверх стандарту IEEE802.15.4. Таким чином досліджувана мережа складалась із однорідних елементів (homogeneous network), у якій всі вузли в мережі використовують однакові протоколи та апаратне забезпечення. Усі вузли у модельованій мережі працюють під керуванням модифікованої операційної системи Contiki 3.0, включаючи вузли-сніфери. З точки зору маршрутизації було використано стандартний мережевий стек в Contiki OS на основі протоколу RPL. Граничний маршрутизатор реалізовано на основі операційної системи Ubuntu 22.04, яка обробляє всі з'єднання, що надходять від сніферів і базових станцій. Налаштування Ubuntu передбачало налаштування пересилки пакетів (packet forwarding) та iptables. Зв'язок граничного маршрутизатора із глобальною мережею реалізовано за допомогою Ehternet.

На рисунку 4.3 приведено результати процесу моделювання розгорнутої IoT мережі в Cooja.

Для збору мережевого трафіку в запропонованій мережі використовуються сніфери. Сніфери постійно відстежують і збирають дані та надсилають їх до блоку агрегації даних. Для реалізації сніферів було використано засіб із відкритим вихідним кодом для моніторингу та анлізу пакетів у безпровідних сенсорних мережах wireshark [13]. Також для захоплення мережевого трафіку канального рівня було використано бібліотеку Libpcap [14].

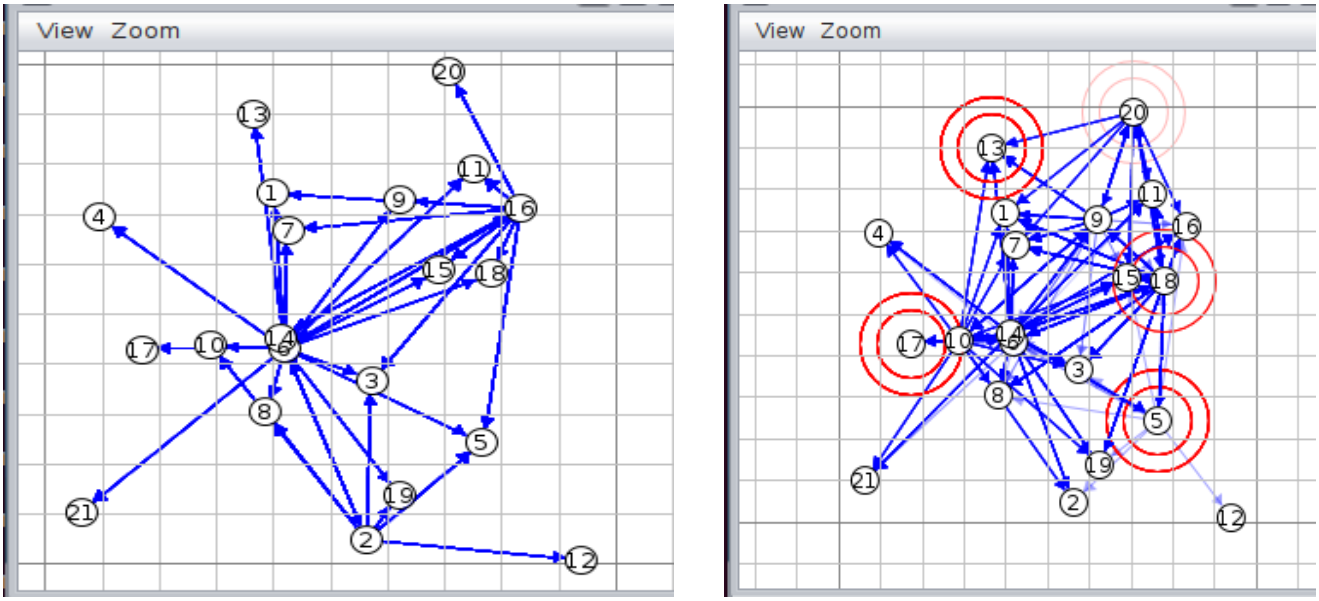


Рисунок 4.3 – 6LoWPAN-RPL мережа для моделювання в Cooja

4.3 Імплементація модулю збору даних

Відповідно до розгорнутої інфраструктури модуль збору даних системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні складався із чотирьох компонентів: модуль захоплення, агрегації даних, блоку черги та блоку отримання ознак.

Процес функціонування модуля збору даних у розгорнутій 6LoWPAN-RPL мережі зображено на рис. 4.4.

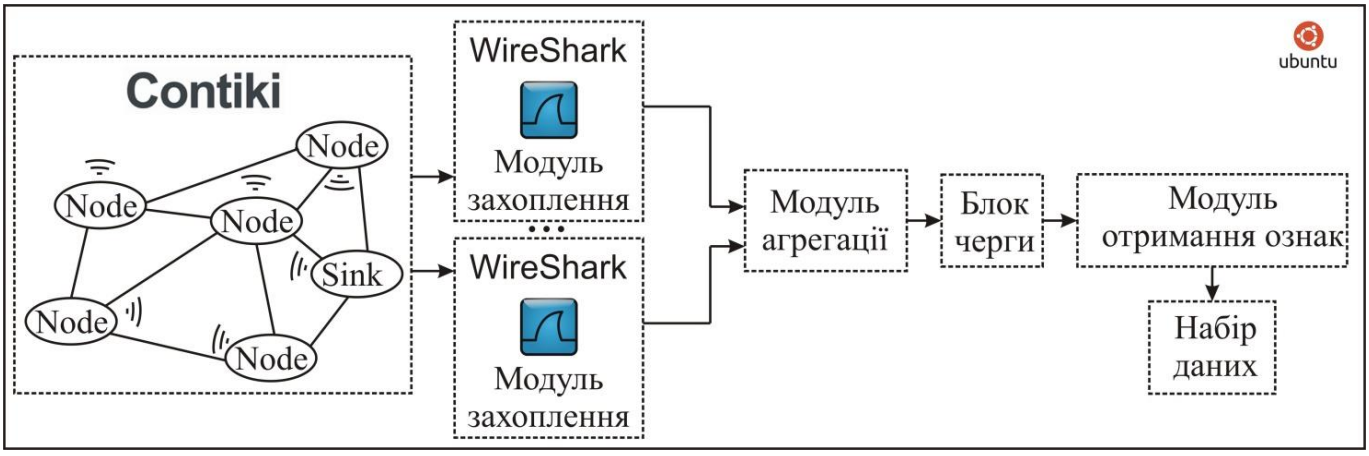


Рисунок 4.4 – Процес функціонування модуля збору даних у розгорнутій 6LoWPAN-RPL мережі

4.3.1 Захоплення мережевого трафіку

Основою роботи системи збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей є захоплення мережевих необроблених мережевих пакетів (raw packet). З цією метою в даній роботі використано програмний засіб wireshark та бібліотеку lippcap. Алгоритм захоплення мережевих необроблених мережевих пакетів представимо наступним чином:

Result: packet

packet = incoming packet;

while *packets. size != 0* do

tmp = packet[0]. timestamp - packet. timestamp;

if *packet.size =¡ 40* then

send packet payload to queue;

else

continue;

end

end

Даний алгоритм фіксує необроблені дані рівня MAC для подальшого надсилання до блоку агрегації даних. Також слід відзначити, що дані, зібрані на цьому етапі, є необробленими даними протоколу IEEE802.15.4, що містять усю пов’язану інформацію про сигнали.

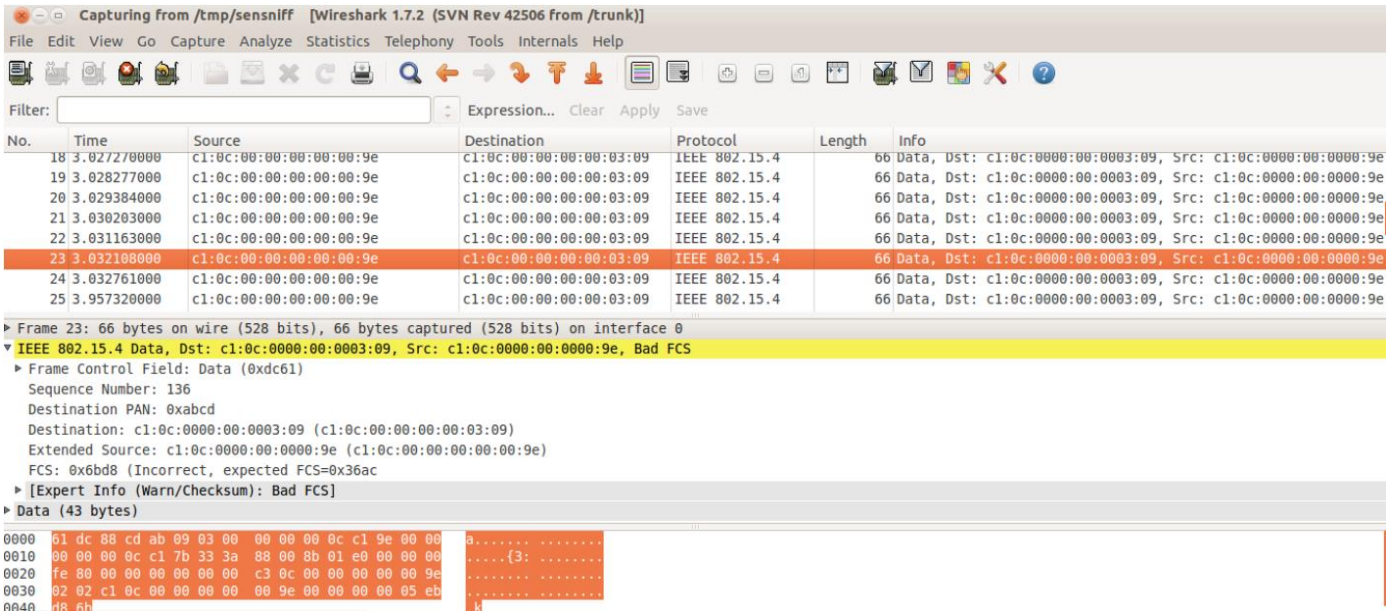


Рисунок 4.1 – вікно Wireshark у процесі захоплення мережевих пакетів у змодельованій мережі

4.3.1 Модуль агрегації

Основною функцією цього модуля є агрегування даних із багатьох джерел і забезпечення цілісності даних. Головна мета даного модуля полягає в тому, щоб у зібраних даних не було дублювання. Алгоритм роботи модуля агрегації подамо наступним чином:

1. Перевірка присутності активних сніферів у мережі та збережіть кожного із цих сніфферів у масиві. Слід звернути увагу, що для ефективного виявлення розподілених атак важливо агрегувати весь мережевий трафік від усіх вузлів мережі. Незважаючи на те, що через природу бездротового сигналу знадобиться кілька вузлів-сніферів, які будуть збирати мережевий трафік, існує ймовірність накладання сигналів, що може спричинити надмірність даних і їх дублювання. Також більша кількість сніферів (більше одного) дозволить покрити велику мережу Інтернету речей. Тому потреба в блоці агрегації даних є досить важливим.

2. Перевірити кожний вузол та визначити чи є дублювання ідентифікатора вузла. Виконати повторну перевірки за допомогою мітки часу для кожного пакета (виконується перевірка спочатку по ідентифікатору вузла, а потім по часовій мітці). Цей крок є ключовим, оскільки він має кілька рівнів перевірки цілісності. Якщо пакет дублюється, це можна визначити шляхом порівняння різних атрибутів у кожному пакеті. Кожен пакет має унікальний ідентифікатор, який є комбінацією трьох значень: IP джерела, IP призначення та мітки часу. Ці значення використовуються для перевірки надмірності пакетів і усунення дублювання при додаванні у чергу.

4.3.2 Блок черги

Агреговані пакети надсилається до сховища даних, що функціонує за принципом черги. Система отримує всі дані, надіслані з агрегатора даних, і виконує їх перевірку в межах часового вікна *T*. Далі усі пакети, які знаходяться в межах часового вікна, надсилаються до блоку вилучення ознак. Крім того, на основі кількості спостережень створюється вікно із змінним часом (різні значення *T*), яке вказує, скільки пакетів надіслав вузол. Наступний алгоритм демонструє організацію черги.

Sniffer: S = {S1,...,Sn.} of size*n*

output: A list of queued packets

node - list of all nodes in each Sn of size i;

while *i* < *n* **do**

Q = 0 ;

for k - 0 **to** n **do**

for j - 0 **to** i **do**

if Q > 0 then

if node *[k].pkt* = node *[i] pkt* then node is duplicated add

one of them add node k.pkt to Q;

else add node [k].pkt to Q;

else Q) < 0;

featureExtraction(Q);

continue

foreach *element e of the line i* do FindCompress (p);

4.3.3 Отримання ознак

Процес отримання ознак є останнім етапом у процесів роботи модуля збору даних. Його функціонування передбачає циклічне виконання послідовних етапів:

– на першому етапі здійснюється отримання ознак, пов’язаних із рівнем MAC, наприклад значення RSSI та потужності передачі. Це допоможе зрозуміти параметри нижнього рівня взаємодії;

– на другому етапі виділяються ознаки рівня мережі, такі як ознаки, пов’язані із протоколом маршрутизації RPL [10]. Оскільки атака відмова в обслуговуванні відбувається на мережевому рівні, ознаки зібрані на цьому рівні є найбільш релевантні для набору даних, що використовується для виявлення DDoS атак;

– на третьому етапі отримуються функції прикладного рівня, такі як корисне навантаження даних у пакеті UDP. На цьому рівні витягується інформація, пов’язана з датчиком, наприклад інформація про рівень заряду акумулятора та температуру. Структура UDP пакету наведено на рис. 4.5. Для визначення рівня заряду акумулятора використано плагін Powertrace [https://sci-hub.se/10.1109/icicos.2017.8276353].

В результаті роботи модуля отримання ознак буде отримано файл, що міститиме отримані вектори ознак. На рис. 4.6 приведено приклад векторів ознак, отриманих модулем збору даних (рядки таблиці).

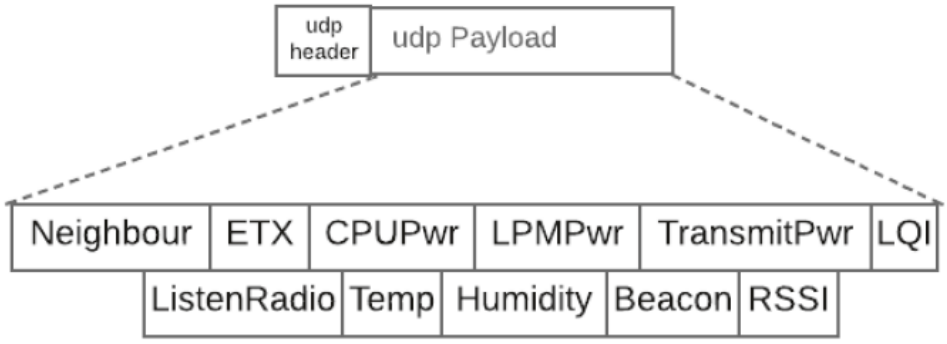


Рисунок 4.5 – Структура PDU в UDP пакеті

Таблиця 4.3 Приклад отриманих векторів ознак

IP адреса вузла	Розмір вікна	Отриманий сигнал DBM	Переданий сигнал DBM	RSSI	Інтервал широкомовного індикатора мережі	LQI	Кількість DIS	Кількість DIO	Споживана потужність	Залишкова потужність	Температура
ffff:0000:0001	1	1.0412E+13	0.03134235	20	0	32716	0	5	0.06418177	0	42
ffff:0000:0002	1	1.2436E+13	0.03871539	10	0	20155	0	3	0.03352426	0	44
ffff:0000:0003	1	1.1512E+13	0.02327442	20	0	26355	0	3	0.10853607	0	52
ffff:0000:0001	1	1.0312E+13	0.03134235	20	0	32716	0	5	0.06418175	0	43
ffff:0000:0002	1	1.0135E+13	0.03747539	20	0	20155	0	3	0.03352426	0	43
ffff:0000:0003	1	1.2345E+13	0.02997442	20	0	26355	0	3	0.10853607	0	52
ffff:0000:0001	1	1.1112E+13	0.03134235	20	0	31723	0	5	0.09181767	0	43
ffff:0000:0002	1	1.1035E+13	0.03747539	20	0	21155	0	3	0.03352426	0	43
ffff:0000:0003	1	1.13484E+1	0.02997442	20	0	25004	0	3	0.15854607	0	43

4.3.4 Маркування даних

Оскільки ми маємо справу з сенсорними мережами та трафіком 6LoWPAN, маркування даних на основі потоку є неможливим через величезну кількість даних, що передаються з кожного вузла. Крім того, мережеві дані в мережі IoT не передаються як звичайний мережевий потік, як наприклад у TCP, відстежуючи процес трьохстороннього рукостискання можна було б використовувати для вилучення мережевого потоку для кожного вузла. Окрім того, оскільки існує канал з'єднання між джерелом і одержувачем у традиційному мережевому потоці, його можна легко відстежити. Тому, для вирішення цієї задачі, позначатимемо набори даних на основі

вектора моделювання та часу. Таким чином частина пакетів, що належать сегменту у якому присутній шкідливий вузол, на протязі інтервалу часу t_m будуть позначені як malicious та визначатимуть розподілену атаку відмова в обслуговувані.

4.4 Реалізація атаки скидання пакетів (blackhole attack) у мережі Інтернету речей

Атака скидання пакетів є атакою типу «відмови в обслуговуванні», в якій маршрутизатор, за принципом роботи повинен ретранслювати пакети, проте натомість відкидає їх. Зазвичай це відбувається через компрометацію маршрутизатора з низки причин. Однією зі згаданих є атака типу "відмова в обслуговуванні" на маршрутизатор з використанням відомого інструменту DDoS. Оскільки пакети на шкідливих маршрутизаторах зазвичай просто відкидаються, таку атаку дуже складно виявити та запобігти.

Шкідливий маршрутизатор також може виконувати цю атаку вибірково, наприклад, відкидаючи пакети для певного мережного призначення, у певний час дня, відкидати кожен n-ий пакет або через кожні t секунд, або випадково вибрану частину пакетів. Якщо шкідливий маршрутизатор намагається відкинути всі вхідні пакети, атака може бути виявлена досить швидко за допомогою звичайних мережевих інструментів, таких як traceroute. Крім того, коли інші маршрутизатори помічають, що шкідливий маршрутизатор відкидає весь трафік, вони зазвичай починають видаляти цей маршрутизатор зі своїх таблиць пересилання, і зрештою трафік не спрямовуватиметься на атаку. Однак, якщо шкідливий маршрутизатор починає відкидати пакети в певний період часу або через кожні n-пакети, його часто буває важче виявити, оскільки певний трафік все ще протікає через мережу.

Атака скидання пакетів може часто використовуватися для атаки на бездротові мережі. Оскільки бездротові мережі мають архітектуру, що значно відрізняється від архітектури типової провідної мережі, хост може повідомляти іншим вущлам, що у нього найкоротший шлях до місця призначення, через що весь трафік буде

направлений на цей скомпрометований хост, і він зможе відкидати пакети за власним бажанням. Також у мобільній мережі ad-hoc хости особливо вразливі для спільних атак: коли кілька хостів будуть скомпроментовані вони зможуть порушувати коректну роботу інших хостів у мережі.

Для реалізації атаки скидання пакетів було використано RPL Attacks Framework [12].

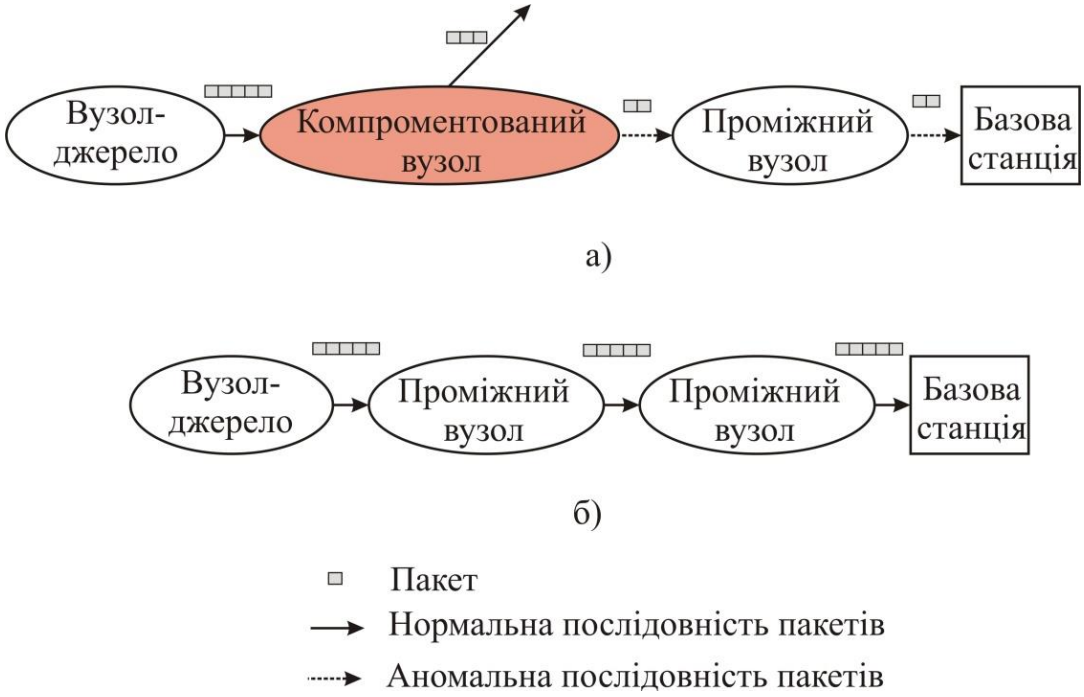


Рисунок 4.6 – Схематичне представлення процесу передачі даних в мережі 6LowPAN: а) під час атаки скидання пакетів; б) при усталеному функціонуванні

4.5 Створення та вибір моделі виявлення

В результаті моделювання бездротової сенсорної мережі було отримано 24 023 векторів ознак, що отримані із IEEE 802.15.4, 6LoWPAN, IPv6 та ICMPv6 пакетів. Із отриманих векторів ознак до класу malicious traffic віднесено 14596 зразків, а до класу legitimate traffic 9426 векторів ознак (таблиця 4.3).

Зібрані дані були проаналізовані на предмет наявності аномалій, зокрема досліджено зміну рангу, кількість DAO та DIO повідомлень у мережевому трафіку під час атаки скидання пакетів та при нормальному функціонуванню мережі.

Крім того, коли інші маршрутизатори помічають, що шкідливий маршрутизатор відкидає весь трафік, вони зазвичай починають видаляти цей маршрутизатор зі своїх таблиць пересилання, і зрештою трафік не спрямовуватиметься на атаку.

Графік порівняння DIO повідомлень наведено на рис. 4.6. Загалом, якщо порівнювати нормальне функціонування та Black hole атаку, то при атаці можна помітити дещо вищу кількість повідомлень.

Це пояснюється тим, що значення рангу інкапсулюється в повідомленні DIO і, отже, надсилається більша кількість пакетів.

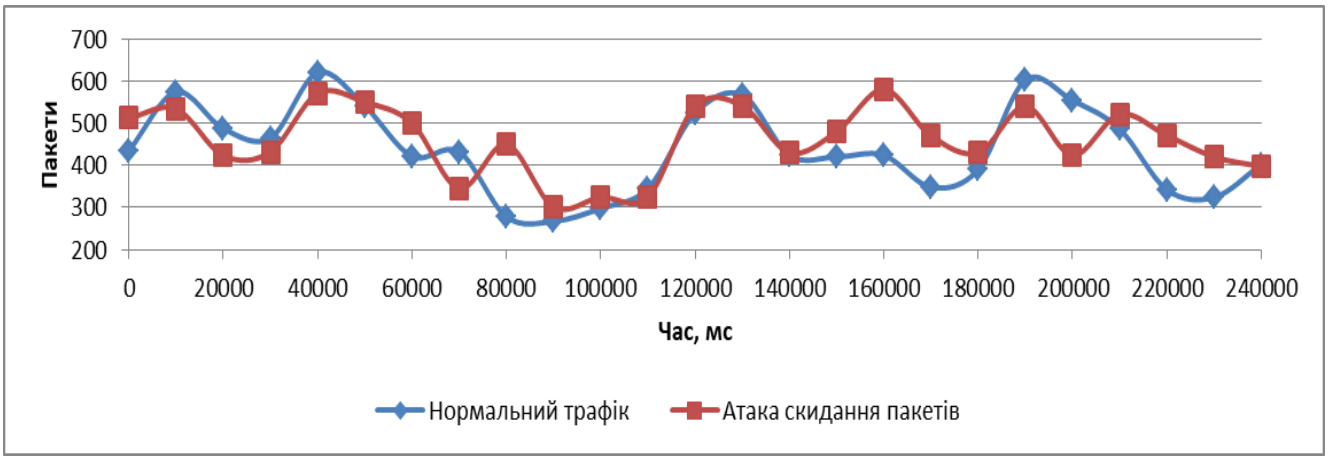


Рисунок 4.6 – Порівняння DIO повідомлень під час атаки скидання пакетів та при нормальному функціонуванні мережі

Для створення моделі виявлення для модуля виявлення у системі збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні весь набір даних був поділений на 2 частини: навчальна та тестова вибірка.

Таблиця 4.3 – Отримана кількість векторів ознак за класами мережевого трафіку

Клас	Кількість
Кількість векторів ознак, що промарковані як malicious traffic	14596
Кількість векторів ознак, що промарковані як legitimate traffic	9426

Навчальний набір даних це набір векторів ознак, які використовуються для процесу навчання та підгонки параметрів класифікатора. Даний набір складає 80% всіх векторів ознак із обох класів (тобто 7 540 зразків легітимного трафіку та 11 676 зразків, що промарковані як malicious traffic). Таким чином навчальний набір даних використовується для створення моделей, які є кандидатами для розпізнавання шкідливої активності у мережевому трафіку.

Для підбору оптимальних гіперпараметрів для кожної моделі було використано метод K-перехресної перевірки. Даний метод використовується для пошуку оптимальних гіперпараметрів моделі та нівелювання процесів недонавчання та перенавчання моделі.

Для виконання K-перехресної перевірки вся множина навчальної вибірки була поділена на дві частини: навчальну та вибірку для валідації (рис. 3.2). В якості K було вибрано значення 8. Це означає, що із 8 частин, навчання моделі проводиться на 7 частинах, а перевірка здійснюється на тій, що залишилась. Даний процес ітеративно продовжувався допоки кожна із 8 частин була використана як тестовий набір. На кожній ітерації проводилось оцінка моделі класифікатора із використанням міри F1. За результатами всі K навчань та перевірок класифікатора було визначено усереднене значення міри F1 (рис. 4.7):

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{4.1}$$

В результаті процесу K-перехресної перевірки було отримано оптимальний варіант моделі для кожного класифікатора (SVM та ШНМ). Для налаштування моделі та вибору гіперпараметрів було використано підхід на основі RandomizedSearchCV.

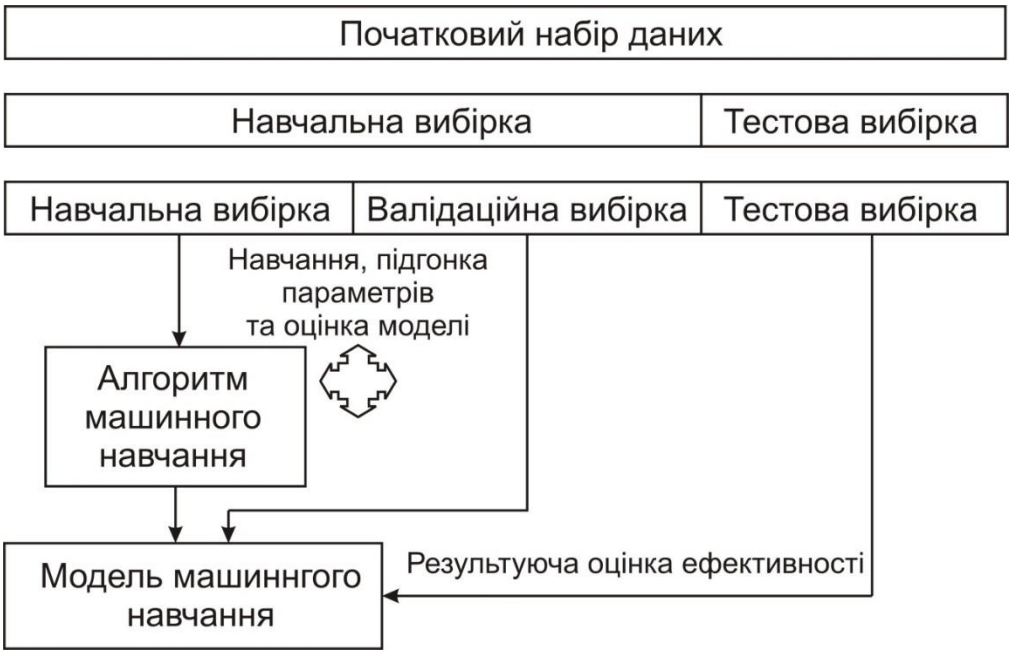


Рисунок 4.7 – Процес розбиття вибірки даних

Для моделі штучної нейронної мережі було використано багатошаровий персептрон із зворотним розповсюдженням помилки. В якості гіперпараметрів для запропонованої ШНМ було досліджено кількість прихованих шарів, значення альфа та функцію активації. Кількість прихованих шарів використовується для визначення кількості шарів між входом мережі та виходом мережі та кількості нейронів у кожному прихованому шарі. Чим вище число, тим більше часу потрібно на обробку, але точніші результати. Значення альфа використовується для регулярізації, та визначає штрафне значення, яке використовується для визначення розміру ваг, що використовуються для запобігання перенавчанню. У деяких випадках високе значення альфа

використовується для вирішення проблеми перенавчання, де є велика дисперсія. З іншого боку, його також можна використовувати для вирішення проблеми недостатнього пристосування шляхом зниження значення показника альфа.

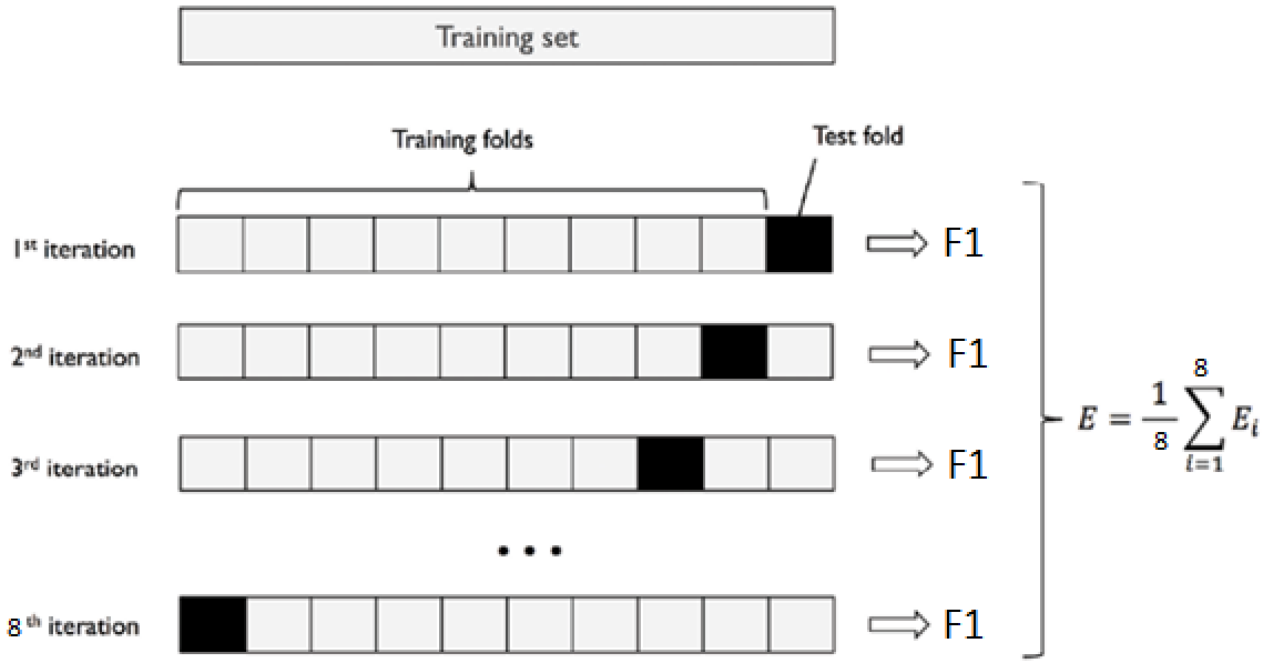


Рисунок 4.8 – Процес проведення K-перехресної перевірки

За результатами проведених експериментів по визначенню оптимальних параметрів оптимальне значення кількості прихованих шарів становить (6,4), функція активація ReLU, а значення альфа 0,001.

В якості гіперпараметрів для моделі на основі SVM обрано значення C та Гамма. Параметр C повідомляє у SVM визначає, наскільки потрібно уникнути неправильної класифікації кожного прикладу при навчанні. Для великих значень C оптимізація вибере гіперплощину з меншим запасом, якщо ця гіперплощина краще справляється з правильною класифікацією всіх навчальних точок. І навпаки, дуже мале значення C змусить оптимізатор шукати роздільну гіперплощину з більшим запасом, навіть якщо ця гіперплощина неправильно класифікує більше точок.

Таблиця 4.4 – Приклад одного тесту із K

№	F1 міра	Функція активації	Прихований шар	Значення альфа
1	0,654	Логістична	(1,10)	0,0001
2	0,711	ReLU	(1,10)	0,001
3	0,622	Логістична	(5,10)	0,005
4	0,723	ReLU	(5,10)	0,001
5	0,541	Логістична	(50,10)	0,001
6	0,702	ReLU	(10,10)	0,05
7	0,548	Логістична	(10,10)	0,001
8	0,698	ReLU	(50,10)	0,05

Ядром SVM обрано радіальну базисну функцію, у якій параметр гама визначає вплив точки на кривизну рішення.

За результатами проведених експериментів оптимальними гіперпараметрами для SVM було визначено значення C на рівні 1 та параметром гамма, що складає 0,001. Як і для ШНМ визначення гіперпараметрів для SVM проводилось на основі K-перехресної перевірки.

4.6 Оцінка ефективності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих із протоколу RPL

Для визначення оцінки ефективності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих системою збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні було використано дві моделі класифікаторів, що описані у попередньому розділі.

В якості метрик для оцінки було використано значення Accuracy, а також показників TP, TN, FP, FN.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4.2)$$

В якості нульової гіпотези H_0 було визначено твердження, яке можна сформулювати наступним чином: «зразок мережевого трафіку має ознаки аномальності та може бути атакою скидання пакетів». Тоді показники TP, FP, TN та FN визначають:

TP – визначає кількість векторів ознак, що промарковані як malicious та вірно розпізнані системою як аномальна активність, яка відповідає атаці відмова в обслуговуванні;

TN – визначає кількість векторів ознак, що промарковані як legitimate та вірно розпізнані системою як нормальний трафік;

FP – визначає кількість векторів ознак, що промарковані як legitimate проте помилково розпізнані системою як аномальний трафік;

FN – визначає кількість векторів ознак, що промарковані як malicious проте помилково розпізнані системою як нормальний трафік;

Таблиця 4.5 – Результати оцінки достовірності виявлення розподілених атак відмова на основі даних отриманих запропонованою системою

Модель класифікатора	Спостереження				Метрика
	TP	*FP*	*TN*	*FN*	*Accuracy*
ШНМ	2546	264	1622	374	0,867
SVM	2686	298	1588	234	0,896

За результатами проведених експериментів можна зробити висновок, що обидві моделі класифікаторів, що представляють ядро модуля виявлення у запропонованій системі, продемонстрували достовірність виявлення більшу за 85%. Кращим отриманні результати у моделі на основі SVM (достовірність виявлення 89,6%) із рівнем хибних позитивних спрацювань (помилок першого роду) 6% та рівнем хибно негативних спрацювань 4,87%. Слід відзначити, що модель на основі штучної нейронної мережі показала результати помилок першого роду на рівні 5,5, що є меншим відповідне значення у моделі SVM. Проте з точки зору критичності для кінцевих користувачів важливішим є помилки другого роду, які є в даному експерименті кращими саме у моделі на основі SVM. Таким чином за результатами експериментів можна зробити висновок, що запропонована система збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні дозволяє отримати набори даних, що можуть бути використанні для ефективного виявлення кібератак у мережах Інтернету речей.

4.7 Висновки

Розробка будь-якого методу чи системи, що використовує методи машинного навчання, вимагає збору даних, пов’язаних із сферою дослідження. У порівняні із традиційними мережами для IT інфраструктур, для яких створено різні набори даних, що дозволяють реалізувати навчання методів машинного навчання, для мереж Інтернету речей на сьогоднішній день не має такого попередньо визначеного набору даних. Тому з метою перевірки достовірності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих запропонованою системою було розгорнуто інфраструктуру на основі операційної системи Contiki та Cooja симулятора. Мережа IoT складалась із 21 вузла, кожен із яких представляє представляв давач Zolertia Z1, що збиравдані про температуру. Вузли мережі здійснювали надсилання даних на базову станцію із інтервалом 20 секунд. Для збору

даних у змодельованій мережі було імплементовано модуль збору даних, що складався із двох сніферів, модуля агрегації, блоку черги та блоку формування векторів ознак. В якості сніферів використано Wireshark та бібліотеку libpcap. Перевірка достовірності виявлення кібератак передбачала отримання даних із протоколу RPL при двох сценаріях – нормальному функціонуванні мережі та при впливі атаки black hole. Всього в результаті експериментів було отримано 24 023 векторів ознак. Для імплементації модуля класифікації використано дві відомі моделі класифікаторів SVM та багатошаровий персептрон. Для обох моделей проведено підбір гіперпараметрів. За результатами проведених експериментів обидві моделі класифікаторів, продемонстрували достовірність виявлення більшу за 85%. Кращим отриманні результати у моделі на основі SVM (достовірність виявлення 89,6%) із рівнем хибних позитивних спрацювань (помилок першого роду) 6% та рівнем хибно негативних спрацювань 4,87%. Слід відзначити, що модель на основі штучної нейронної мережі показала результати помилок першого роду на рівні 5,5, що є меншим відповідне значення у моделі SVM. Таким чином за результатами експериментів можна зробити висновок, що запропонована система збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні дозволяє отримати набори даних, що можуть бути використанні для ефективного виявлення кібератак у мережах Інтернету речей.

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень запропоновано метод та систему збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні в мережах Інтернету речей, що функціонують на основі давачів Zolertia Z1.

У першому розділі досліджено архітектуру 6LoWPAN-RPL мереж, проведено аналіз відомих підходів та стратегій, а також наукових методів до збору даних в мережах Інтернету речей.

У другому розділі запропоновано узагальнену теоретико-множинну модель процесу, що описує послідовний процес обробки та збору інформації із мережевого трафіку з метою виявлення розподілених атак відмова в обслуговуванні та із подальшою ізоляцією скомпрометованих вузлів. Вхідними даними для представленої моделі процесу є вхідний трафік, тоді як вихідними даними новий маршрут, що представляє нове дерево DODAG у протоколі RPL, у якому будуть відсутні скомпрометовані вузли.

У третьому розділі представлено структуру системи та кроки методу збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні, яка складається з трьох основних модулів: модуль збору даних, модуль класифікації та модуль виявлення. Головною особливістю модуля збору даних було те, що збір даних забезпечувався декількома сніферами, що встановлені у мережі, і з подальшою агрегацією зібраних даних. Для реалізації модуля класифікації проведено дослідження методу опорних векторів та багатошарового персептрона. Модуль виявлення використовувався для трансляції повідомлення про аномальну поведінку на решту вузлів IoT мережі, що містять ідентифікатор скопроментованого вузла та шлях до нього.

У четвертому розділі проведено перевірку достовірності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих запропонованою системою.

Для проведення експериментів було змодельовано інфраструктуру на основі операційної системи Contiki та Cooja симулятора. Мережа IoT складалась із 21 вузла, кожен із яких представляє представляв давач Zolertia Z1, що збиравдані про температуру. Вузли мережі здійснювали надсилання даних на базову станцію із інтервалом 20 секунд. Для збору даних у змодельованій мережі було імплементовано модуль збору даних, що складався із двох сніферів, модуля агрегації, блоку черги та блоку формування векторів ознак. В якості сніферів використано Wireshark та бібліотеку libpcap. Перевірка достовірності виявлення кібератак передбачала отримання даних із протоколу RPL при двох сценаріях – нормальному функціонуванні мережі та при впливі атаки black hole. Всього в результаті експериментів було отримано 24 023 векторів ознак. Для імплементації модуля класифікації використано дві відомі моделі машинного навчання – метод опорних векторів та багатошаровий персептрон. Для обох моделей проведено підбір гіперпараметрів. За результатами проведених експериментів обидві моделі класифікаторів, продемонстрували достовірність виявлення більшу за 85%.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Arora S. Et al. Seed: secure and energy efficient data-collection method for IoT network. *Multimedia Tools and Applications.* 2022, 82(2), p.1-15

2. Salim A., Osamy W., Aziz A., M. K. Ahmed. SEEDGT: Secure and energy efficient data gathering technique for IoT applications based WSNs. *Journal of Network and Computer Applications*. 2022. Vol. 202. 103353

3. Bhardwaj A., Kaushik K., Bharany S., Rehman A.U., Hu Y.-C., Eldin E.T., Ghamry N.A. IIoT: Traffic Data Flow Analysis and Modeling Experiment for Smart IoT Devices. *Sustainability.* 2022, *14*, 14645. https://doi.org/10.3390/su142114645

4. Alzahrani M.A., Alzahrani A.M., Siddiqui M.S., Detecting DDoS Attacks in IoT-Based Networks Using Matrix Profile, *Appl. Sci.* 2022 12(16). doi: 10.3390/app12168294

5. Jing H., Wang J., Chen C.L. Detection of DDoS Attack within Industrial IoT Devices Based on Clustering and Graph Structure Features, *Security and Communication Networks.* 2022. 1401683 doi: 10.1155/2022/1401683

6. Hussain F., Abbas S. G., Husnain M., Fayyaz U. U., Shahzad F. and Shah G. A. IoT DoS and DDoS Attack Detection using ResNet, Proceedings of *2020 IEEE 23rd International Multitopic Conference (INMIC)*, Bahawalpur, Pakistan, 2020, pp. 1-6. doi: 10.1109/INMIC50486.2020.9318216.

7. L. Hong, K. Wehbi and T. H. Alsalah, Hybrid Feature Selection for Efficient Detection of DDoS Attacks in IoT, Proceedings of the 2022 6th International Conference on Deep Learning Technologies (ICDLT '22), Association for Computing Machinery, New York, NY, USA, 2022, pp. 120–127. doi: 10.1145/3556677.3556687

8. Bouyeddou B., Kadri B., Harrou F., Sun Y. DDOS-attacks detection using an efficient measurement-based statistical mechanism, *Engineering Science and Technology, an International Journal*. 2020. Vol. 23(4) pp. 870-878. doi: 10.1016/j.jestch.2020.05.002

9. Wibowo B., Alaydrus M. Smart Home Security Analysis Using Arduino Based Virtual Private Network, Proceedings of *2019 Fourth International Conference on*

Informatics and Computing (ICIC), Semarang, Indonesia, 2019, pp. 1-4. doi: 10.1109/ICIC47613.2019.8985669.

10. Doshi R., Apthorpe N., Feamster N. , Machine learning ddos detection for consumer internet of things devices, Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 24 May 2018; pp. 29–35.

11. Ibrahim R. F., Al-Haija Q. A., Ahmad A. DDoS Attack Prevention for Internet of Thing Devices Using Ethereum Blockchain Technology, *Sensors.* 2022. Vol. 22(18) 6806. doi: 10.3390/s22186806

12. RPL Attacks Framework, URL: https://github.com/dhondta/rpl-attacks

13. Wireshark, URL: https://www.wireshark.org/

14. Tcpdump & Libpcap, URL:https://www.tcpdump.org/

15. Pomorova O., Savenko O., Lysenko S., Nicheporuk A. Metamorphic Viruses Detection Technique based on the Modified Emulators, *CEUR Workshop Proceedings*. 2016. Vol. 1614, pp. 375–383.

16. Savenko O., Lysenko S., Nicheporuk A. et al. Metamorphic Viruses' Detection Technique Based on the Equivalent Functional Block Search, *CEUR Workshop Proceedings*. 2017. Vol. 1844 pp. 555-569.

17. Savenko O., Nicheporuk A., Hurman I., Lysenko S. Dynamic signature-based malware detection technique based on API call tracing, *CEUR Workshop Proceedings.* Vol. 2393, pp. 633–643

18. Savenko O., Lysenko S., Nicheporuk A., Savenko B. Approach for the Unknown Metamorphic Virus Detection, Proceedings of the *8-th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, Bucharest Romania, September 21–23, 2017, pp. 71–76.

19. Pomorova O., Savenko O., Lysenko S., Kryshchuk A., Nicheporuk A. A Technique for detection of bots which are using polymorphic code, *Communications in Computer and Information Science. 2014.* Vol. 431, pp. 265-276.

20. Nanthiya D., Keerthika P., Gopal S. B., et al. SVM Based DDoS Attack Detection in IoT Using Iot-23 Botnet Dataset, *Proceedings of 2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, Kuala Lumpur, Malaysia, 2021, pp. 1-7, doi: 10.1109/i-PACT52855.2021.9696569.

21. Ye J., Cheng X., Zhu J., Song L.. A DDoS Attack Detection Method Based on SVM in Software Defined Network, *Security and Communication Networks*. 2018. pp. 1-8. doi:10.1155/2018/9804061.

22. Farukee M. B., Zaman Shabit M. S. et al, DDoS Attack Detection in IoT Networks Using Deep Learning Models Combined with Random Forest as Feature Selector. *Advances in Cyber Security*, Penang, Malaysia, 2021, pp.118-134. doi:10.1007/978-981-33-6835-4_8.

23. Al-hadhram Y., Hussain F. K., A Machine Learning Architecture Towards Detecting Denial of Service Attack in IoT. *Conference on Complex, Intelligent, and Software Intensive Systems*, Springer, 2019, pp.417-429.

24. Fotiadou K., Velivasaki T-H., Voulkidis A., et al, Network Traffic Anomaly Detection via Deep Learning. *Information*, 2021. Vol. 5 (215). doi: 10.3390/info12050215

25. KDD Cup Archives, URL: https://kdd.org/kdd-cup

26. Osterlind F., Dunkels A., Eriksson J., Finne N. and Voigt T., Cross-Level Sensor Network Simulation with COOJA, Proceedings. 2006 *31st IEEE Conference on Local Computer Networks*, Tampa, FL, USA, 2006, pp. 641-648. doi: 10.1109/LCN.2006.322172.

27. Іванченко Н.О. Густера О.М. Основні проблеми безпеки IoT в умовах цифровізації економіки України. Економіка та держава. 2019. № 11. С. 50-54.

28. Miettinen M, Marchal S., Hafeez I., Asokan N. et al. IoT Sentinel: Automated Device-Type Identification for Security Enforcement in IoT. *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*: Proceedings (Atlanta, GA, USA, June 5-8 2017), Atlanta, 2017. P. 2177-2184.

29. Muhs D., Haas S., Strufe T., Fischer M. On the Robustness of Random Walk Algorithms for the Detection of Unstructured P2P Botnets. *The 11th International Conference on IT Security Incident Management & IT Forensics*: Proceedings (Hamburg, Germany, 7–9 May 2018). Hamburg, 2018. P. 3–14.

30. Sagirlar G., Carminati B., Ferrari E. AutoBotCatcher: Blockchain-Based P2P Botnet Detection for the Internet of Things. *2018 IEEE 4th International Conference on Collaboration and Internet Computing*: Proceedings (Philadelphia, PA, USA., 18–20 October 2018). Philadelphia, 2018. P. 1–8.

31. Wressnegger C., Freeman K., Yamaguchi F., Rieck K. Automatically Inferring Malware Signatures for Anti-Virus Assisted Attacks. *2017 ACM on Asia Conference on Computer and Communications Security*: Proceedungs (Abu Dhabi, United Arab Emirates, April 2-6, 2017). Abu Dhabi, 2017. P. 587-598.

32. Ndibanje B., Kim K.H. Kang Y.J., Kim H.H., Kim T.Y. Lee, H.J. Cross-Method-Based Analysis and Classification of Malicious Behavior by API Calls Extraction. *Applied Sciences*. 2019. Vol. 9. No. 2. P. 1-15.

33. Lim H. Detecting Malicious Behaviors of Software through Analysis of API Sequence k-grams. *Computer Science and Information Technology*. 2016. Vol. 4 (3). P. 85-91.

34. Brad G.V. Uses and misuses of Bayes' rule and Bayesian classifiers in cybersecurity. The 43-rd International Conference Applications of mathematics in engineering and economics: Proceedings (Sozopol, Bulgaria, 8-13 June 2017). Sozopol, 2017. P. 1-8.

35. Al-Garadi M.A., Mohamed A., Al-Ali A., Du X., Guizani M. A survey of machine and deep learning methods for internet of things (iot) security. *arXiv preprint* arXiv:1807.11023. 2018.

36. Antonakakis M, April T., Bailey M., et al. Understanding the mirai botnet. 26th fUSENIXg Security Symposium (fUSENIXg Security 17): Proceedings (Vancouver, BC, Canada, August 16–18, 2017). Vancouver, 2017. P. 1093-1110.

37. Ceron J. M., Steding-Jessen K., Hoepers C., Granville L.Z., Margi C. B. Improving iot botnet investigation using an adaptive network layer. *Sensors*. 2019. No. 19(3):727.

38. Raiyn J. A survey of Cyber Attack Detection Strategies. International Journal of Security and Its Applications. 2014. No. 8(1). P. 247-256.

39. Hugo Bezerra V, Turrisi da Costa V. et al. One-class classication to detect botnets in iot devices. Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais: Proceedings (Natal, Brazil). Natal, 2018. P 43-56.

40. Alhanahnah M., Lin Q., Yan Q., Zhang N., Chen Z. Efficient signature generation for classifying cross-architecture iot malware. 2018 IEEE Conference on Communications and Network Security (CNS): Proceedings (Beijing, China, 30 May – 01 June 2018). Beijing, 2018. P. 1-9.

41. Abomhara M., Koien G.M. Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks. *Journal of Cyber Security*. 2015. No. 4(1). P. 65-88.

42. Mansour A.M. Texture Classification using Naïve Bayes Classifier. *International Journal of Computer Science and Network Security*. 2018. Vol.18, No.1. P. 112-120.

43. Ki Y., Kim E., Kim H.K. A novel approach to detect malware based on API call sequence analysis. *International Journal of Distributed Sensor Networks - Special issue on Advanced Big Data Management and Analytics for Ubiquitous Sensors*. 2015. Vol. 2015.

44. Ding W., Ren W., Xia Z., Wang L. Botnet tracing based on distributed denial of service activity analysis. *The 8th International Conference on Biomedical Engineering and Informatics*: Proceedings (Shenyang, China, October 14–16 2015). Shenyang, 2015. P. 685–689.

45. Muhs D., Haas S., Strufe T., Fischer M. On the Robustness of Random Walk Algorithms for the Detection of Unstructured P2P Botnets. *The 11th International*

Conference on IT Security Incident Management & IT Forensics: Proceedings (Hamburg, Germany, 7–9 May 2018). Hamburg, 2018. P. 3–14.

46. Miettinen M, Marchal S., Hafeez I., Asokan N. et al. IoT Sentinel: Automated Device-Type Identification for Security Enforcement in IoT. *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*: Proceedings (Atlanta, GA, USA, June 5-8 2017), Atlanta, 2017. P. 2177-2184.

47. Jurafsky D., Martin J. M. Speech and Language Processing, 2nd Edition: Prentice Hall, 2008. 1032 p.

48. Meidan Y., Bohadana M., Mathov Y., et al. N-BaIoT-Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. *IEEE Pervasive Computing*. 2018. Vol. 17, Issue: 3. P. 12–22.

49. Doshi R., Apthorp N., Feamster N. Machine Learning DDoS Detection for Consumer Internet of Things Devices. *2018 IEEE Security and Privacy Workshops (SPW)*: Proceedings (San Francisco, CA, USA, 24-24 May 2018). San Francisco, 2018.

50. Raiyn J. A survey of Cyber Attack Detection Strategies. International Journal of Security and Its Applications. 2014. No. 8(1). P. 247-256.

51. Elzen I. v D., Heugten J. V. Techniques for detecting compromised IoT Devices. University of Asterdam. Amsterdam. 2017.

52. Apthorpe N., Reisman D., Feamster N. A Smart Home is No Castle: Privacy Vulnerabilities of Encrypted IoT Traffic. *arXiv:1705.06805*. 2017.

53. Ronen E., Shamir A., Weingarten A.-O., O'Flynn C. IoT Goes Nuclear: Creating a ZigBee Chain Reaction. *2017 IEEE Symposium on Security and Privacy:* Proceedings (San Jose, CA, USA, May 22-26, 2017). San Jose, 2017. P. 195-212.

54. Zhang C. Green R. Communication security in internet of thing:preventive measure and avoid ddos attack over iot network. *The 18th Symposium on Communications & Networking. Societyfor Computer Simulation International*: Proceedings (Alexandria, Virginia, USA, 2015). Alexandria, 2015. P. 8–1.

55. Xiang Y., Li K., Zhou W. 2011. Low-rate DDoS attacks detection andtraceback by using new information metrics. *IEEE Transactions on Information Forensics and Security.* 2011. No. 2. P. 426–437.

56. Jerkins J. A. Motivating a market or regulatory solution to iot inse-curity with the mirai botnet code. *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*: Preoceedings (Las Vegas, NV, USA, 2017). Las Vegas, 2017. P. 1–5.

57. Jun C., Chi C. Design of complex event-processing idsin internet of things. *2014 Sixth International Conference on Measuring Technology and Mechatronics Automation*: Proceedings (Zhangjiajie, China, 2014). Zhangjiajie, 2014. P. 226–229.

58. Du P., Abe S. IP packet size entropy-based scheme for detection of DoS/DDoS attacks. *IEICE transactions on information and system*. 2008. Vol. E91.D, Issue 5. P.1274–1281.

59. Nobakht M., Sivaraman V., Boreli R. A host-based intrusiondetection and mitigation framework for smart home iot using openflow. *11-th International Conference on Availability, Reliability and Security (ARES)*: Proceedings (Salzburg, Austria, August 31 - September 2, 2016). Salzburg, 2016. P. 147–156.

60. Barnawi A. Performance Analysis of RPL Protocol for Data Gathering Applications in Wireless Sensor Networks, *Proceedings of the 10th International Conference on Ambient Systems, Networks and Technologies (ANT)* April 29 - May 2, 2019, Leuven, Belgium, pp. 185–193

61. Hakeem S. A. et al. New Real Evaluation Study of Rpl Routing Protocol Based on Cortex M3 Nodes of Iot-Lab Test Bed. *Computer Science.* 2015. Pp. 456-472

62. Farukee M. B., Zaman Shabit M. S. et al, DDoS Attack Detection in IoT Networks Using Deep Learning Models Combined with Random Forest as Feature Selector. *Computer Science and Information Technology*, Penang, Malaysia, 2021, pp.118-134. doi:10.1007/978-981-33-6835-4_8.

63. Ceron J. M., Steding-Jessen K., Hoepers C., Granville L.Z., Margi C. B. Improving iot botnet investigation using an adaptive network layer. *Security and Privacy Workshops*. 2019. No. 19(3):727.

64. June C., Chink C. Design of complex event-processing idsin internet of things. *2015 Sixth International Conference on Measuring Technology and Mechatronics Automation*: Proceedings (Zhangjiajie, China, 2015). Zhangjiajie, 2015. P. 226–229.

65. Vigario, R.: Extraction of ocular artifacts from EEG using independent component analysis. *Electroencephalogr.* Clin. Neurophysiol. 103, 395-404 2017. P. 226–229.

66. Turrisi Bezerra V, Hego da Costa V. et al. One-class classication to detect botnets in iot devices. Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais: Proceedings (Natal, Brazil). Natal, 2018. P 43-56.

67. Mittinen M, Marchal S., Hafeez I., Asoka N. et al. IoT Sentinel: Automated Device-Type Identification for Security Enforcement in IoT. *2018 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*: Proceedings (Atlanta, GA, USA, June 5-8 2018), Atlanta, 2018. P. 2177-2184.

68. Abomhara M., Koien G.M. Cyber security and the internet of things: vulnerabilities, threats, intruders and attacks. *Journal of Cyber Safety*. 2014. No. 4(1). P. 65-88.

69. Haas S., Muhs D., Strufe T., Fischer M. On the Robustness of Random Walk Algorithms for the Detection of Unstructured P2P Botnets. *The 11th International Conference on IT Security Incident Management & IT Forensics*: Proceedings (Berlin, Germany, 7–9 May 2018). Berlin, 2018. P. 3–14.

70. Alzen I. v D., Heugten J. V. Techniques for detecting compromised IoT Devices. University of Asterdam. Amsterdam. 2019.

71. Kimi Y., Kimu E., Kan H.K. A novel approach to detect malware based on API call sequence analysis. *International Journal of Distributed Sensor Networks - Special issue on Advanced Big Data Management*. 2017. Vol. 2017.

72. Liam A. Detecting Malicious Behaviors of Software through Analysis of API Sequence k-grams. *Computer Technology*. 2014. Vol. 5 (4). P. 75-81.

73. Hinko M., Midori I. Design of complex event-processing idsin internet of things. *2014 Conference on Measuring Technology and Mechatronics Automation*: Proceedings (Zhangjiajie, China, 2017). Zhangjiajie, 2017. P. 116–119.

74. Jay M. H. Motivating a market or regulatory solution to iot inse-curity with the mirai botnet code. *2016 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*: Preoceedings (California, NV, USA, 2016). California, 2016. P. 1–5.

75. Al-Faadir M. A., Al-Haija Q. A., Ahmad A. DDoS Attack Prevention for Internet of Thing Devices Using Ethereum Blockchain Technology, *Computer Technology*. 2022. Vol. 22(18) 6806. doi: 10.3390/s22186806.

76. Yamaguchi C., Freeman K., Fritefmad F., Rieck K. Automatically Inferring Malware Signatures for Anti-Virus Assisted Attacks. *2017 ACM on Asia Conference on Computer and Communications Security*: Proceedungs (Abu Dhabi, United Arab Emirates, April 2-6, 2019). Abu Dhabi, 2019. P. 587-598.

77. Macwell R, Mitchel S., Hafeez I., Asoka N. et al. IoT Sentinel: Automated Device-Type Identification for Security Enforcement in IoT. *2018 IEEE 34th International Conference on Distributed Computing Technology (ICDCT)*: Proceedings (Missuri, GA, USA, April 20-23 2014), Missuri, 2014. P. 2177-2184.

78. Noah M., Sivaraman V., Vargas R. A host-based intrusiondetection and mitigation framework for smart home iot using openflow. *9-th International Conference on Availability, Reliability and Security (ARES)*: Proceedings (Salzburg, Austria, August 25 - September 5, 2014). Salzburg, 2014. P. 126–130.

79. Ronen E., Marko A., Shinso J. IoT Goes Nuclear: Creating a ZigBee Chain Reaction. *2015 IEEE Symposium on Security and Privacy:* Proceedings (Florida, USA, May 22-26, 2015). Florida, 2015. P. 155-162.

80. Boreli F., Haas S. G., Husnaik M., Fayyaz U. U., Shahzad F. and Shah G. A. IoT DoS and DDoS Attack Detection using ResNet, Proceedings of *2021 IEEE 23rd International Multitopic Conference (INMIC)*, Bahawalpur, Pakistan, 2021, pp. 9-13. doi: 10.1109/INMIC50486.2021.9318216.

81. Alzahrani M.A., Sqali M.M., Siddiqui M.S., Detecting DDoS Attacks in IoT-Based Networks Using Matrix Profile, *Appl. Sci.* 2018 8(12). doi: 10.3390/app12168294.

ДОДАТОК А
ВСТАНОВЛЕННЯ ТА НАЛАШТУВАННЯ СЕРЕДОВИЩА CONTIKI

sudo apt get update

sudo apt-get install git

sudo apt-get install build-essential binutils-msp430 gcc-msp430 msp430-libc msp430mcu mspdebug gcc-arm-none-eabi gdb-arm-none-eabi openjdk-8-jdk openjdk-8-jre ant libncurses5-dev lib32ncurses6

git clone –b release-3-0 https://github.com/contiki-os/contiki.git

https://www.youtube.com/watch?v=T-w3cKkpbIE

cd contiki

git submodule update --init --recursive

cd tools

cd coojs

ant run

ДОДАТОК Б
КОПІЇ НАУКОВИХ ПУБЛІКАЦІЙ

ЗМІСТ

Авсієвич В.Р., Кузьмін А.А.
Дослідження вразливостей системи розумної парковки та способи їх усунення 11

Алексейко В.О., Бармак О.В.
Інформаційна система прогнозування поширення респіраторних захворювань в невеликих популяціях 15

Барчук Д.О., Нічепорук А.А., Казанцев А.Д., Нічепорук А.О.
Оцінка ризиків інформаційної безпеки системи розумного будинку на основі методології Octave Allegro 20

Башта А.Р., Кравчук С.С.
Концепція застосування доповненої реальності для інтерфейсу користувача програмної системи пошуку громадських місць з можливостями інклюзивного доступу 24

Башук І.О., Микитенко Д.А., Частоколенко І.П.
Система програмно-апаратного комплексу для моніторингу ключових кліматично-пожежних параметрів приміщення у режимі реального часу 30

Бельфер Р.Е.
Архітектура багаторівневої однорангової мережі 32

Білик О.В.
Інформаційна система «Вчена рада факультету» 35

Богатирчук Д.В.
Сучасний стан та перспективи України на світовому рівні ІТ технологій 39

Борусевич А.В., Куперштейн Л.М.
Інтелектуальна інформаційна технологія визначення типу операційної системи віддаленого вузла 43

Буднік І.Ю., Підченко С.К.
Метод стабілізації параметрів кварцових радіотехнічних пристроїв 46

Вакулко Я.І., Шевченко В.Л.
Програмне забезпечення виділення об'єктів піксельного зображення і зображення і пошуку шаблонів в задачах доповненої реальності 49

УДК 004.092

Барчук Д.О., Нічепорук А.А., Казанцев А.Д., Нічепорук А.О.

Хмельницький національний університет

ОЦІНКА РИЗИКІВ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ СИСТЕМИ РОЗУМНОГО БУДИНКУ НА ОСНОВІ МЕТОДОЛОГІЇ OCTAVE ALLEGRO

Розглянуто кроки методології оцінки ризиків OCTAVE Alegro та запропоновано оцінку ризику для інформаційного об'єкту «інформація, зібрана пристроями» для середовища розумного будинку.

The steps of the OCTAVE Alegro risk assessment methodology are reviewed and a risk assessment is proposed for the information object "information collected by devices" for the smart home environment.

Зростаюча популярність Інтернету речей (IoT) надає широкі можливості для покращення, планування та автоматизації повсякденного життя. Проте разом із очевидними перевагами та зручностями, що несе із собою використання IoT, концепція "інтернет речей" залишає для зловмисників ряд потенційних "вузьких" місць у безпеці таких систем. Персональні дані користувачів, зібрані розумними пристроями, завжди мають цінність для хакерів і викрадачів конфеденційної інформації. Крім того, кібератака на IoT-рішення потенційно здатна завдати шкоди фізичним сервісам та фізичній інфраструктурі. При проектуванні та експлуатації систем Інтернету речей важливим завданням є оцінка цих потенційних "вузьких" місць та розроблення повних та вичерпних стратегій по пом'якшенню та усуненню негативних впливів кібератак. Тому актуальним завданням є визначення можливих кіберзагроз та оцінка їх впливів на критичні інформаційні об'єкти в системі розумного будинку.

При проектуванні та експлуатації систем розумного будинку важливим завданням є визначення кіберзагроз, оцінка їх впливу на потенційно "вузькі" місця системи та розроблення повних та вичерпних стратегій по пом'якшенню та усуненню негативних впливів кібератак. Причому, чим швидше буде проведено оцінку та прийнято відповідні заходи, тим більша імовірність забезпечення цілісності, доступності та конфіденційності інформації. Розглянемо процес оцінки ризиків інформаційної безпеки системи розумного будинку. Для оцінки ризиків використаємо методологію OCTAVE Allegro.

OCTAVE Allegro є методологію, що дозволяє упорядкувати та оптимізувати процес оцінки ризиків інформаційної безпеки, що дозволяє організації отримати

достатні результати за невеликі витрати часу, людських та інших обмежених ресурсів. Основний фокус методології OCTAVE Allegro полягає у розгляді людей, технології та засобів у контексті їх ставлення до інформації та бізнес-процесів та послуг, які вони підтримують.

Методологія OCTAVE Allegro визначає вісім послідовних етапів, організованих у 4 фази (рисунок 1): визначення критеріїв, профілювання об'єктів, визначення загроз, визначення та пом'якшення ризиків.. За допомогою таблиць OCTAVE Allegro, є можливість фіксувати результати кожного кроку оцінки ризику та використовувати їх як вхідні дані для наступних кроків. Окремі кроки застосовуються до кожного окремого інформаційного об'єкту. Для проведення оцінки ризиків безпеки використаємо шаблон OCTAVE Allegro, що представлено у [1].

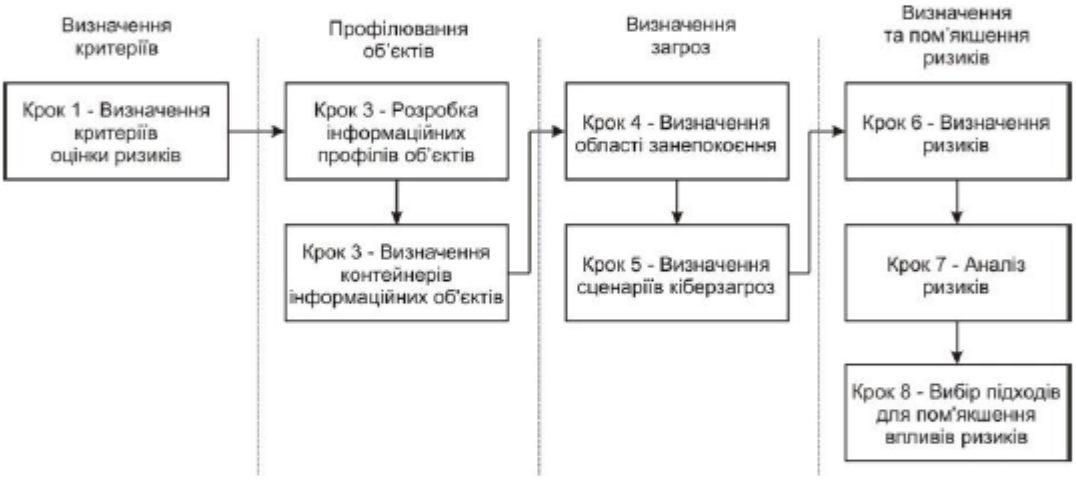


Рисунок 1 – Узагальнена схема процесу автоматизації керування розумним будинком на основі машинного навчання

Після виявлення ризиків (відповідно до загрози та вразливості) та оцінки ризиків можна визначити план пом'якшення, щоб уникнути або обмежити виявлені ризики та негативні наслідки, що випливають з них [2]. Виконаємо оцінку ризику для інформаційного об'єкту «інформація, зібрана пристроями (датчиками)» (рисунок 2).

Зазначені кроки методології OCTAVE Allegro проводяться для кожного критичного інформаційного об'єкту. Проведений процес оцінки ризиків дозволяє проаналізувати інформаційні об'єкти в системі розумного будинку, які є критичними з точки зору безпеки, провести аналіз ризиків та їх впливів на об'єкти, та запропонувати можливі контрзаходи з метою захисту інформаційних об'єктів та створення системи розумного дому більш безпечним.

Оцінка ризику для інформаційного об'єкту

Загроза

Інформаційний об'єкт	Інформація, зібрана пристроями (датчиками)
Сфера занепокоєння	1) Зміна показників датчика газу може призвести до хибного розпізнання наявності газу в приміщенні, що може позначитись на здоров'ї та житті мешканців 2) Отримання даних із датчика руху можна використати для визначення присутності мешканців будинку. 3) Зчитування стану замків дверей та системи сигналізації можна використати, щоб визначити, коли розумний будинок зайнятий. 4) DoS атаки на рівень сприйняття (компрометація каналу зв'язку) систем розумного будинку продукує неможливість сприйняття фізичних параметрів датчиками, що тим самим унеможливлює виявлення таких ризиків, як пожежа, повінь, несподівані рухи тощо.

(1) **Дійова особа** *Хто здійснюватиме вплив на інформаційний об'єкт створюючи загрозу безпеці?*	Зловмисник (хакер, недобросовісний постачальник програмних та апаратних засобів)		
(2) **Засоби** *Яким чином дійова особа здійснить це? Що вони повинні зробити для цього?*	Засоби злому Вразливості в апаратному забезпеченні		
(3) **Мотив** *Який виграш отримає дійова особа здійснивши порушення безпеки?*	Фінансова вигода, заволодіння персональних амбіцій		
(4) **Результат** *Яким чином це відобразиться на інформаційному об'єкті?*	○ **Розкриття** • **Зміна**	○ **Знищення** • **Переривання**	
(5) **Вимоги безпеки** *Яким чином будуть порушені вимоги безпеки інформаційного об'єкту?*	Лише авторизовані члени розумного будинку повинні мати доступ до цієї інформації та змінювати її.		
(6) **Імовірність** *Яка імовірність відтворення подібного впливу?*	• **Висока**	○ **Середня**	○ **Низька**

(7) **Наслідки** *Які будуть наслідки для організації або власника інформаційного об'єкта при порушенні вимог безпеки?*	(8) **Важкість** *Наскільки серйозними є ці наслідки для організації чи власника об'єкту в залежності від зони впливу?*

У випадку порушення вимог безпеки для цього інформаційного об'єкту системи розумного будинку не в змозі буде відстежувати та контролювати критично важливі показники давачів, що може призвести до негативних наслідків, пов'язаних як із фізичною природою (пожежа, підтоплення), так із людським фактором (проникнення, крадіжка речей). В обох випадках прояви негативних наслідків можуть призвести до великих фінансових втрат.

Зони впливу	**Значення**	**Оцінка**
Репутація та довіра клієнтів (4)	Середня (2)	4*2 = 8
Фінансова (3)	Висока (3)	9
Продуктивність (2)	Низька (1)	2
життя, здоров'я, безпека (5)	Висока (3)	15
штрафи та юридичні санкції (1)	Низька (1)	1
Відносне значення оцінки ризику		35

(9) Пом'якшення ризиків
Виходячи з загальної оцінки цього ризику, які дії слід вжити?

○ **Прийняти**	○ **Відхилити**	• **Пом'якшити**	○ **Передати**

Щодо ризиків, які було вирішено пом'якшити, слід виконати наступні дії.

Якого контейнера будуть стосуватись дії?	*Який адміністративний, технічний та фізичний контроль слід застосувати до цього контейнера? Який залишковий ризик все ще буде прийнято організацією?*
Технічний	Обмежити доступність маркерного трафіку типа для авторизованих користувачів; використання протоколів передачі даних із шифруванням (наприклад SSL/TLS)
Фізичний	Зберігайте всі фізичні дані в надійному місці. Регулярне оновлення апаратного забезпечення; створення резервних копій всієї важливої інформації.
Люди	Інформування мешканців стосовно безпечного управління розумним будинком

Рисунок 2 – Оцінка ризику для інформаційного об'єкту «інформація, зібрана пристроями (датчиками)»

Таким чином застосування OCTEAVE Alegro дозволяє визначити та систематизувати критичні інформаційні об'єкти в системі розумного будинку, критерії оцінки ризиків та сценарії кіберзагроз. В результаті дослідження було проведено оцінку ризиків інформаційної безпеки системи розумного будинку із залученням методології OCTAVE Allegro для інформаційного об'єкту, що представляє інформацію, зібрану датчиками розумного будинку. Подальшим дослідженням є формування комплексної оцінки ризиків інформаційної безпеки системи розумного будинку та реалізації програмної системи, що дозволить автоматизувати процес формування оцінки ризиків не тільки для системи розумного будинку, а й для інших систем, що імплементують принцип Інтернету речей.

Перелік посилань

1. Caralli, R. A., Stevens, J. F., Young, L. R., & Wilson, W. R. Introducing octave allegro: Improving the information security risk assessment process, No. CMU/SEI-2007-TR-012, 2007.
2. Morozova O., et al. Smart Home System Security Risk Assessment International Scientific Journal «Computer Systems and Information Technologies», 2021, № 3. Pp. 81-88.

ДОДАТОК В

ПРЕЗЕНТАЦІЯ ДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ

МЕТОД ТА СИСТЕМА ЗБОРУ ДАНИХ ПРОТОКОЛУ МАРШРУТИЗАЦІЇ RPL З ДАТЧИКІВ ZOLERTIA Z1 У МЕРЕЖАХ ІНТЕРНЕТУ РЕЧЕЙ

Барчук Д.О.

Науковий керівник: к.т.н.,Нічепорук А.О.

Хмельницький, 2023

- **Об'єктом дослідження** є процес збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей.
- **Предметом дослідження** є метод та система збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні в мережах Інтернету речей.
- **Метою** кваліфікаційної роботи магістра є отримання набору даних із мереж Інтернету речей на основі протоколу маршрутизації RPL, використання якого, дозволило б підвищити достовірність виявлення розподілених атак відмова в обслуговуванні у мережах IoT.
- **Методи**: теорії графів, методи множин, оцінки ефективності та методи побудови теоретико-множинних моделей.

Наукова новизна

Наукова новизна отриманих результатів:

- набула подальшого розвитку модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні, яка на відміну від відомих здійснює послідовний опис процесу обробки та збору інформації із мережевого трафіку, що дозволило удосконалити метод збору даних із мереж Інтернету речей для виявлення розподілених атак відмова в обслуговуванні.
- набули подальшого розвитку система та метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні, які на відмінну від відомих залучають декілька мережевих сніферів для агрегації даних, що дозволило на основі використання отриманого набору даних здійснити виявлення розподілених атак відмова в обслуговувані.

Практична значимість

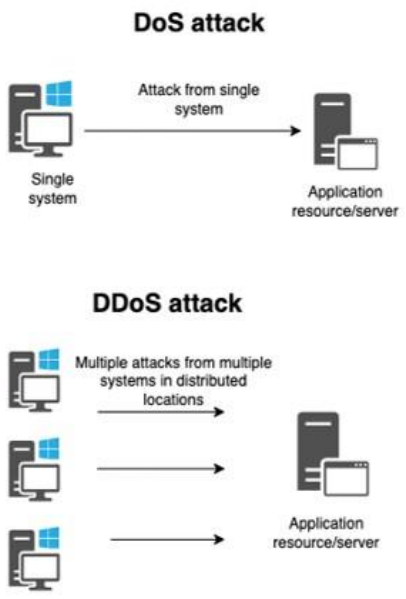
- Практична значимість отриманих результатів полягає у тому, що запропонована система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей може бути інтегрована в існуючі системи, що виконують задачі моніторингу стану мережі та виявлення зловмисної активності у IoT мережах.

DoS і DDoS атаки

Найпоширеніші принципи DoS і DDoS:

Бомбардування системи величезною кількістю непотрібних даних, щоб заповнити доступну пропускну здатність цільової мережі або її обчислювальну потужність (атаки затоплення).

Перенаправлення пакетів або їх відхилення (атаки blackhole або drops атаки). Цей тип атак особливо гостро відчувається в мережах IoT через характер реалізації алгоритмів маршрутизації, які передбачають використання повністю зв'язаних топологій і передачу даних від джерела до одержувача через ланцюжок проміжних вузлів.



Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Модуль збору даних

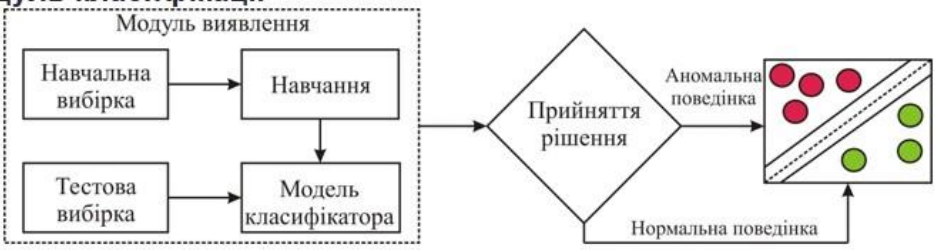
Features	Description
f^p_{RSSI}	average value received signal strength indicator
f^p_{RdBM}	average value of the received signal dBm
f^p_{TdBM}	average value of the transmitted signal dBm
f^n_{LQI}	value of link quality indicator
f^n_{ETX}	average value of the expected transmission count
f^n_{NDIO}	number of DIO messages
f^n_{NDIS}	number of DIS messages
f^n_{LRPL}	number of node's RPL rank changing
f^a_{MeCP}	modal value of power consumption
f^a_{MoCP}	average value of power consumption
f^a_{NID}	node ID

Набір функцій, зібраних із пакетів IoT

Рівні стеку протоколів, з яких збираються функції

Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Модуль класифікації



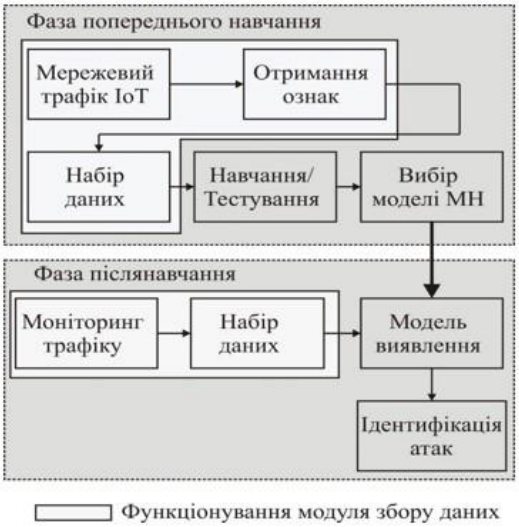
Процес вилучення функцій із мережевого трафіку IoT

Модуль виявлення

Основна мета цього модуля – передати повідомлення всім вузлам мережі IoT про аномальну поведінку, що включає ідентифікацію зловмисника та маршрут, яким він пройшов. Це дозволить незачепленим вузлам занести зловмисний вузол у чорний список і утримуватися від зв’язку з ним

Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Фаза попереднього навчання



На етапі попереднього навчання модель машинного навчання навчається та тестується на основі даних, зібраних DGM:

- Вибір алгоритму
- Навчання/тестування:
- Перевірка:
- Оптимізація:

Архітектура системи збору даних з протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

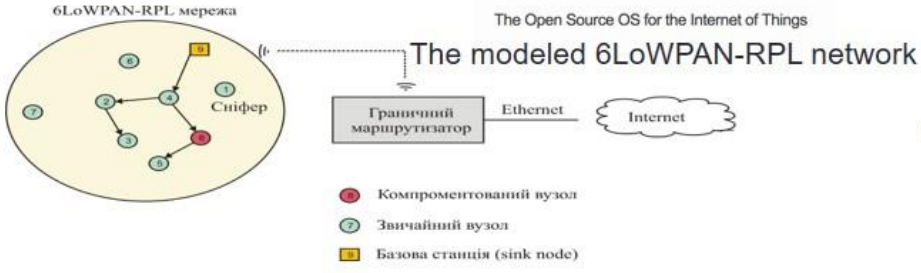
Фаза після навчання

1. Агрегація трафіку. Цей крок передбачає збір даних від кількох сніфферів, що працюють у мережі Інтернету речей. Щоб перевірити унікальність пакетів, пакети порівнюються за міткою часу. Далі, якщо є збіг, перевіряється ідентифікатор вузла. Таким чином, підпис даних визначається як пара значень <мітка часу, ідентифікатор вузла.
2. Особливості вилучення. На цьому етапі виконується така ж послідовність дій, як і на етапі попереднього навчання (в автономному режимі), за винятком того, що цей процес виконується в режимі реального часу для мереж Інтернету речей.
3. Класифікація атак. На основі оптимальної моделі машинного навчання, отриманої на етапі попереднього навчання, класифікуються аномалії мережевого трафіку.
4. Генерація результатів. На цьому кроці генерується результат виявлення, а також створюється та надсилається UDP-пакет агенту виявлення. Пакет містить такі параметри, як ідентифікатор вузла, мітка часу, батьківський вузол, ранг і результат виявлення.

Моделювання, отримання даних, оцінка

Збір даних і перевірка точності виявлення розподілених атак на відмову в обслуговуванні

Modeling IoT network



Contiki

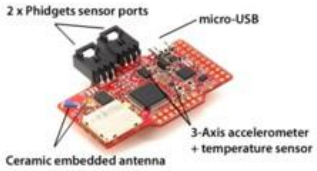
The Open Source OS for the Internet of Things

The modeled 6LoWPAN-RPL network

Параметри для процесу моделювання в Cooja

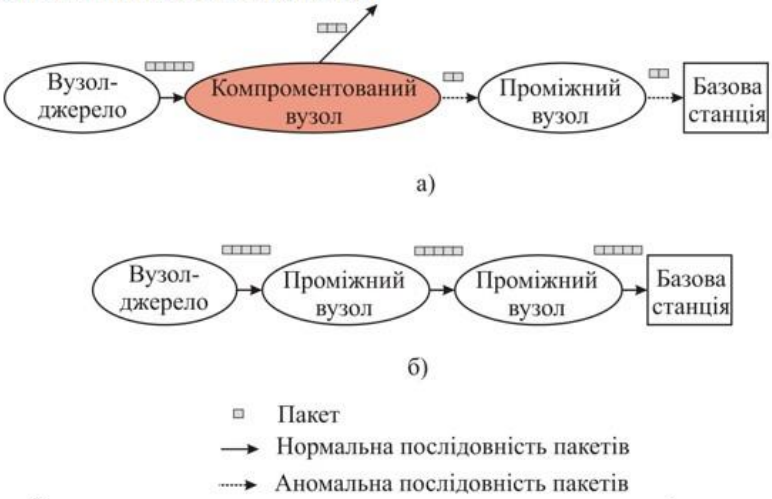
Параметр	Значення
Модель бездротового каналу	UDGM
Кількість вузлів	21
Протокол маршрутизації	RPL
Транспортний протокол	UDP
MAC протокол	CSMA + ContikiMAC
Розмір мережі	50 x 100 метрів
Тип вузлів	Zolteria Z1
Час моделювання	3 години

Бездротовий датчик Zolertia Z1 для вимірювання температури в мережах 6LowPAN-RPL



Збір даних і перевірка точності виявлення розподілених атак на відмову в обслуговуванні

Моделювання DDoS-атаки IoT



Атака **blackhole** — це DDoS-атака, під час якої маршрутизатор має повторно передавати пакети, але замість цього скидає їх.

Схематичне представлення процесу передачі даних в мережі 6LowPAN:
а) під час атаки скидання пакетів; б) при усталеному функціонуванні

Машинне навчання для системи збору даних із протоколу маршрутизації RPL для виявлення розподілених атак на відмову в обслуговуванні в мережах IoT

Отримана кількість векторів ознак за класами мережевого трафіку

Клас	Кількість
Кількість векторів ознак, що промарковані як malicious traffic	14596
Кількість векторів ознак, що промарковані як legitimate traffic	9426

За результатами проведених експериментів по визначенню оптимальних параметрів оптимальне значення кількості прихованих шарів становить (6,4), функція активація ReLU, а значення альфа 0,001.

За результатами проведених експериментів оптимальними гіперпараметрами для SVM було визначено значення C на рівні 1 та параметром гамма, що складає 0,001. Як і для ШНМ визначення гіперпараметрів для SVM проводилось на основі K-перехресної перевірки.

Оцінка ефективності виявлення розподілених атак відмова в обслуговуванні на основі даних отриманих із протоколу RPL

Результати оцінки достовірності виявлення розподілених атак відмова на основі даних отриманих запропонованою системою

Модель класифікатора	Спостереження				Метрика
	TP	FP	TN	FN	Accuracy
ШНМ	2546	264	1622	374	0,867
SVM	2686	298	1588	234	0,896

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN},$$

TP – визначає кількість векторів ознак, що промарковані як malicious та вірно розпізнані системою як аномальна активність, яка відповідає атаці відмова в обслуговуванні;
TN – визначає кількість векторів ознак, що промарковані як legitimate та вірно розпізнані системою як нормальний трафік;
FP – визначає кількість векторів ознак, що промарковані як legitimate проте помилково розпізнані системою як аномальний трафік;
FN – визначає кількість векторів ознак, що промарковані як malicious проте помилково розпізнані системою як нормальний трафік;

Дякую за увагу!

UNICHECK

Ім'я користувача:
Кафедра КІ

ID перевірки:
1014790010

Дата перевірки:
25.04.2023 11:30:29 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
25.04.2023 11:31:01 EEST

ID користувача:
100005591

Назва документа: **Барчук_Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у м**

Кількість сторінок: **83** Кількість слів: **15413** Кількість символів: **117670** Розмір файлу: **6.02 MB** ID файлу: **1014495205**

2.03%
Схожість

Найбільша схожість: **1.01%** з джерелом з Бібліотеки (ID файлу: **1014487461**)

1.24% Джерела з Інтернету	121	Сторінка 85
1.44% Джерела з Бібліотеки	87	Сторінка 85

0.1% Цитат

Цитати	7	Сторінка 86
Посилання	1	Сторінка 86

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	13

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 14.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 11%**

	Документ		Сумарний збіг по Базі Даних	
ID: 112555 Назва: МКР Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей Додано в БД: 2023-04-25 Автора: Барчук Д.О. Керівники: Нічепорук А.О. Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	97640	761	20059 (21%)	162 (21%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми
112067	Назва: ЗВІТ з науково-дослідної практики "Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей" Додано в БД: 2023-03-20 Автора: Д.О. Барчук Керівники: Бобровнікову К.Ю. Консультанти: Опоненти:	13931 (14.0%)	101 (13.0%)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Барчук Денис Олександрович

Тема: Метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень – ; кількість сторінок записки 75

1. Короткий зміст роботи та прийнятих рішень У роботі запропоновано метод та систему збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей.

2. Висновок про відповідність роботи дипломному завданню Дипломна робота відповідає виданому завданню

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проведено огляд відомих методів і засобів збору даних у мережах Інтернету речей та основи процесу отримання даних в мережах Інтернету речей. У другому розділі запропоновано модель процесу збору даних із протоколу маршрутизації RPL у мережах Інтернету речей для виявлення розподілених атак відмова в обслуговуванні. У третьому розділі приведено систему і метод збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні. У четвертому розділі здійснено моделювання мережі IoT в операційній системі Contiki, реалізовано отримання даних та оцінка ефективності системи збору даних протоколу маршрутизації RPL для виявлення розподілених атак відмова в обслуговуванні.

4. Позитивні сторони роботи: Запропоновані метод та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей дозволяють підвищити достовірність виявлення розподілених атак відмова в обслуговуванні.

5. Негативні сторони роботи: Було б доцільно приділити більше уваги питанню ізоляції вузла, що ідентифікований був як зловмисний. У роботі

6. Оцінка графічного оформлення та пояснювальної записки роботи: –

7. Відгук про роботу в цілому: В загальному робота виконана на високому рівні.

8. Інші зауваження: –

9. Оцінка дипломної роботи:

Розглянувши позитивні та негативні сторони представленої дипломної роботи вважаю, що робота заслуговує оцінки «добре» 4,5 (B)

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) д.т.н., проф., зав. кафедри автоматизації, комп'ютерно-інтегрованих технологій та робототехніки Мартинюк В.В.

"4" 05 2022р.

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 та система збору даних протоколу маршрутизації RPL з датчиків Zolertia Z1 у мережах Інтернету речей

Автор: Барчук Денис Олександрович

Спеціальність: 123 – Комп'ютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Нічепорук Андрій Олександрович, 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 аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 2,03% і адресується до 208 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ А. О. Нічепорук

Гарант ОП _____ О. С. Савенко

Завідувач кафедри КІС _____ Т. О. Говорущенко

Завідувачу кафедри КІС
д-р.техн.наук, проф. Говорущенко Т. О.

Барчук Денис Олександрович
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.

28.04.2023
дата


підпис