Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації

Бакалаврська дипломна робота на тему:

«Засіб моніторингу бездротової мережі на базі мікрокомп'ютера»

Виконав: студент 2 курсу групи 1БС-21мс
спеціальності 125 Кібербезпека
_____ Согур А.А.

Керівник: к. т. н., доцент каф. ЗІ
_____ Куперштейн Л. М.
«19» червня 2023 р.

Рецензент: К.т.н., доц., доц. каф. ПЗ
_____ Коваленко О. О.
«19» червня 2023 р.

Допущено до захисту

Завідувач кафедри ЗІ

д. т. н., проф.

_____ Лужецький В. А.

«19» червня 2023 р.

Вінниця ВНТУ – 2023 року

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації
Рівень вищої освіти I (бакалаврський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 125 Кібербезпека
Освітньо-професійна програма – Безпека інформаційних і комунікаційних систем

ЗАТВЕРДЖУЮ
Завідувач кафедри ЗІ,
д. т. н., проф.
В. А. Лужецький
«20» березня 2023 року

ЗАВДАННЯ НА БАКАЛАВРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Согуру Антонію Андрійовичу

1. Тема роботи: «Засіб моніторингу бездротової мережі на базі мікрокомп'ютера» керівник роботи: Куперштейн Леонід Михайлович, к. т. н., доцент кафедри ЗІ, затверджені наказом ректора ВНТУ від 20 березня 2023 року № 67.
2. Строк подання студентом роботи 16 червня 2023 р.
3. Вихідні дані до роботи:
   - мова програмуваня – Python;
   - мікрокомп'ютер – Raspberry PI 3 A+;
   - моніторинг бездротової мережі – Wi-Fi мережа;
   - інтерфейс взаємодії – телеграм-бот;
   - режим моніторингу – активний.
4. Зміст текстової частини: Вступ Аналіз предметної області. Технічне проектування системи. Робоче проектування системи. Висновки. Перелік використаних джерел. Додатки.
5. Перелік ілюстративного матеріалу: Структурна схема системи (плакат, А4). Алгоритм роботи програмного засобу (плакат, А4). Інтерфейс програмного засобу (плакат, А4).

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Куперштейн Л. М., к.т.н., доц. каф.ЗІ	20.03	19.06
2	Куперштейн Л. М., к.т.н., доц. каф.ЗІ	20.03	19.06
3	Куперштейн Л. М., к.т.н., доц. каф.ЗІ	20.03	19.06

7. Дата видачі завдання 20 березня 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської дипломної роботи	Строк виконання етапів роботи	Прим.
1	Аналіз завдання. Вступ	20.03.23 – 26.03.23	
3	Аналіз літературних джерел за напрямком бакалаврської кваліфікаційної роботи	27.03.23 – 09.04.23	
4	Розробка рішень	10.04.23 – 23.04.23	
5	Практична реалізація, моделювання, експериментування, результати	24.04.23 – 21.05.23	
6	Розробка розділу тестування і обґрунтування доцільності розробки	22.05.23 – 24.05.23	
7	Аналіз виконання ТЗ, висновки	25.05.23 – 31.05.23	
8	Оформлення пояснювальної записки	01.06.23 – 15.06.23	
9	Попередній захист та доопрацювання БДР	16.06.23 – 19.06.23	
10	Представлення БДР до захисту	20.06.23 – 23.06.23	
11	Захист БДР	20.03.23 – 26.03.23	

Студент _____ Согур А.

Керівник роботи _____ Куперштейн Л.

АНОТАЦІЯ

Бакалаврська дипломна робота складається з 77 сторінок формату А4, на яких є 33 рисунків, перелік використаних джерел містить 42 найменування.

Дана дипломна робота присвячена розробці та реалізації засобу моніторингу бездротової мережі на базі мікрокомп'ютера. Головною метою дослідження є створення ефективного та зручного інструменту для збору та аналізу даних бездротових мереж з можливістю віддаленого керування. У роботі проведено аналіз основних методів та засобів, пов'язаних з моніторингом бездротових мереж. На базі мікрокомп'ютера Raspberry Pi було розроблено апаратну та програмну складові засобу моніторингу. Мікрокомп'ютер Raspberry Pi виступає як основний контролер, що керує процесом моніторингу бездротової мережі. Засіб використовує зовнішній Wi-Fi-адаптер, який дозволяє отримувати доступ до бездротової мережі у режимі моніторингу. Програмне забезпечення засобу включає в себе пакети для збору та аналізу пакетів бездротової мережі, розрахунку потужності сигналу, виявлення проблем та моніторингу трафіку. Для зручного взаємодії та керування розроблено telegram-бот, який надає користувачам зручний та інтуїтивно зрозумілий інтерфейс для отримання статистики, налаштувань та результатів моніторингу бездротової мережі.

Ключові слова: моніторинг бездротової мережі, мікрокомп’ютер, розробка та реалізація засобу моніторингу, telegram-бот.

ABSTRACT

The bachelor thesis consists of 77 pages of A4 format, on which there are 33 figures, the list of used sources contains 42 names.

This thesis is devoted to the development and implementation of a microcomputer-based wireless network monitoring tool. The main goal of the research is to create an effective and convenient tool for collecting and analyzing data of wireless networks with the possibility of remote management. The paper analyzes the main methods and tools associated with wireless network monitoring. The hardware and software components of the monitoring tool were developed on the basis of the Raspberry Pi microcomputer. The Raspberry Pi microcomputer acts as the main controller that controls the wireless network monitoring process. The device uses an external Wi-Fi adapter that allows you to access a wireless network in monitoring mode. The tool's software includes packages for collecting and analyzing wireless network packets, calculating signal strength, detecting problems, and monitoring traffic. Telegram bot has been developed for convenient interaction and management, which provides users with a convenient and intuitive interface for obtaining statistics, settings and results of wireless network monitoring.

Keywords: wireless network monitoring, microcomputer, development and implementation of a monitoring tool, telegram-bot.

ЗМІСТ

ВСТУП

У сучасному світі все більше використовують бездротові мережі для передачі даних та використання інтернету у всіх сферах та аспектах життя суспільства та держави. Разом з розповсюдженням бездротовий мереж також і збільшуються ризики та значно росте кількість випадків проведення кібератак на різноманітні об’єкти та мережі. Саме тому, зростає і потреба у засобах моніторингу та аналізу бездротових мереж для забезпечення безпеки та ефективного використання ресурсів.

Основними завданнями засобів моніторингу бездротових мереж є збір і аналіз інформації про доступні мережі, підключені пристрої, рівень сигналу, швидкість передачі даних та інші параметри бездротового зв'язку. Застосування таких засобів моніторингу може бути корисним у різних сферах, включаючи домашні мережі, бізнес-середовища, громадські місця, тощо.

Об’єктом бакалаврської дипломної роботи є процеси моніторингу бездротових мереж.

Предметом бакалаврської дипломної роботи є методи та засоби моніторингу бездротових мереж.

Метою бакалаврської дипломної роботи є підвищення захищеності бездротової мережі Wi-Fi за рахунок моніторингу програмно-апаратним засобом на базі мікрокомп’ютера.

Для виконання дипломної роботи необхідно виконати наступні задачі:

– проаналізувати методи та засоби моніторингу бездротових мереж;
– проаналізувати мікрокомп’ютерні засоби ;
– розробити структурну схему засобу;
– розробити програмний засіб для моніторингу мережі;
– провести тестування роботи засобу.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз архітектур бездротових мереж

Wi-Fi (скорочення від «Wireless Fidelity», «бездротова точність») – популярна у світі технологія і швидко розвивається в Україні, що забезпечує бездротове підключення мобільних користувачів до локальної мережі та Інтернету [1].

Під іменем «Wi-Fi» насправді ховається кілька стандартів, розроблених для бездротових мереж на основі випущеної ще у 1997 р специфікації IEEE 802.11.

Важливо відзначити, що у стандарті 802.11 передбачається використання тільки напівдуплексних приймальнопередатчиків, які не можуть одночасно передавати і приймати інформацію. Через це у бездротових мережах 802.11 станція в принципі не може виявити зіткнення під час передачі (оскільки в цей час не має можливості приймати дані). Тому в якості методу доступу до середовища у всіх стандартах використовується метод CSMA/CA (із запобіганням колізій), що дозволяє уникати зіткнень [2].

Основним недоліком мереж Wi-Fi на сьогодні є досить мала дальність передачі даних, що не перевищує для більшості пристроїв 150 м (максимум 300 м) на відкритому просторі або всього декількох десятків метрів – у приміщенні (таблиця 1.1).

WiMAX (Worldwide Interoperability for Microwave Access) є технологією бездротового зв'язку, що базується на стандарті IEEE 802.16. Вона появилась 2004 році, розроблена для надання широкого доступу до Інтернету на великих та рухомих умовах. WiMAX використовує радіосигнали для передачі даних через повітря, що дозволяє покривати велику територію без необхідності прокладання провідних ліній [4].

WiMAX можна використовувати як для створення широкого бездротового доступу до Інтернету для користувачів, так і для будівництва приватних мереж для підприємств, міської інфраструктури чи послуг провайдерів. За останні роки технологія WiMAX втратила популярність на інших бездротових стандартах, таких як LTE та 5G, які дають ще більшу швидкість і функціональні можливості [4].

Таблиця 1.1 – Найбільш важливі стандарти IEEE 802.11x

Стандарт	Середовище передачі	Швидкості передачі, Мбіт/с	Примітка
802.11	радіосигнал з частотою близько 2,4 ГГц або ІЧ-сигнал	1 або 2	Базовий стандарт, що визначає взаємодію на фізичному і канальному рівнях моделі OSI
802.11a	радіосигнал з частотою близько 5 ГГц	до 54	Несумісний на фізичному рівні зі стандартами 802.11b і g; в Україні не використовується
802.11b	радіосигнал з частотою 2,4-2,483 ГГц	до 11	Має відносно низьку швидкість та захищеність (захист шифруванням за технологією WEP – Wireless Equivalent Privacy). Забезпечує трохи більшу, порівняно з іншими стандартами, дальність передачі даних
802.11g	радіосигнал з частотою 2,4-2,483 ГГц	до 54	Забезпечує зворотну сумісність зі стандартом 802.11b, але характеризується більшою швидкістю та захищеністю (крім WEP, підтримується стандарт захисту WPA – Wi-Fi Protected Access)

Wireless USB — альтернатива Bluetooth, заснована на використанні нового стандарту ультраширокополосного бездротового зв’язку – UWB і забезпечує супершвидкісну (до 480 Мбіт/с, а у перспективі – і до 1 Гбіт/с) передачу даних на короткі відстані (до 10 м). Вона дозволяє реалізувати бездротове підключення периферійних пристроїв, аналогічне USB 2.0 [5].

Як і у Wi-Fi, у Bluetooth використовується радіосигнал з частотою 2,4 ГГц, однак ці стандарти між собою несумісні. Bluetooth характеризується досить низьким енергоспоживанням, що дозволяє з успіхом застосовувати цю технологію у переносних пристроях – ноутбуках, КПК і мобільних телефонах. До того ж Bluetooth практично не вимагає налаштування – цей стандарт дозволяє пристроям встановлювати взаємодію при мінімальній участі користувача. З іншого боку, у Bluetooth вельми низькі показники за дальністю передачі та пропускною здатністю– не більше 10 метрів і 400-700 Кбіт/с, - що різко обмежує можливості використання цієї технології в локальних мережах [6].

ZigBee. У технології ZigBee, що з’явилася завдяки зусиллям декількох великим комунікаційним компаніям (стандарт 802.15.4), показники ще

«скромніші» – її специфікація передбачає захищену передачу даних в радіусі 10-75 метрів і з максимальною швидкістю до 250 Кбіт/с . «Родзинкою» пристроїв ZigBee є їх наднизьке споживанням енергії і здатність переходити у «сплячий режим», коли передача даних не потрібна. Тому основною сферою використання ZigBee-пристроїв стануть не локальні мережі, а системи моніторингу та контролю апаратури, в тому числі мережевого обладнання [7].

Для підключення бездротових клієнтів слід зупинитися на технології Wi-Fi, причому вибирати потрібно пристрої, що підтримують останній стандарт 802.llg, - тільки в ньому забезпечується достатня швидкість передачі даних і, найголовніше, їх надійний захист.

1.2 Задачі моніторингу бездротової мережі

Моніторинг бездротової мережі є важливою складовою управління та оптимізації роботи, що дозволяє забезпечити захист бездротової мережі та ефективну передачу даних, виявляти проблеми та покращувати якість зв'язку.

Моніторинг бездротової мережі – це процес збору, аналізу та контролю параметрів і даних. Цей процес дозволяє забезпечити ефективну роботу бездротової мережі, виявити проблеми, забезпечення безперебійної роботи та контроль над станом мережі [8].

Моніторинг бездротової мережі включає в себе різноманітні задачі, спрямовані на забезпечення безпеки, оптимальної продуктивності та надійності мережі.

Найважливіших задач моніторингу бездротової мережі:

1. оптимізація розташування точок доступу;
2. моніторинг пропускної здатності;
3. виявлення і усунення перешкод;
4. моніторинг ресурсів мережі;
5. моніторинг доступу до мережі;
6. виявлення загроз та атак.

Ці задачі моніторингу бездротової мережі допомагають забезпечити безпеку, продуктивність та надійність мережі, а також вчасно виявляти й усувати проблеми, що можуть виникнути у процесі її функціонування.

Моніторинг допомагає визначити оптимальне розташування точок доступу для забезпечення належного охоплення сигналом і мінімізації зон тіні та перекриття сигналу. Це може включати аналіз сигналу, виявлення мережевих проблем та оптимізацію розташування та налаштування точок доступу[8].

Важливо відстежувати використання пропускної здатності мережі, щоб ідентифікувати можливі перевантаження або недостатню пропускну здатність. Моніторинг пропускної здатності дозволяє виявляти пікові навантаження, ідентифікувати проблеми з пропускною здатністю і приймати заходи для їх вирішення, такі як оптимізація налаштувань маршрутизації або додаткове розширення мережі.

Моніторинг бездротової мережі допомагає виявити перешкоди, які можуть впливати на якість сигналу і продуктивність мережі. Це можуть бути фізичні перешкоди, такі як стіни або металеві конструкції, або інші електронні пристрої, які створюють електромагнітну інтерференцію. Виявлення і усунення перешкод допомагає покращити якість зв'язку та продуктивність мережі. Дозволяє відстежувати використання ресурсів мережі, таких як пропускна здатність, пам'ять, процесор та інші системні ресурси. Це допомагає ідентифікувати проблеми з продуктивністю, виявляти навантаження і приймати заходи для оптимізації роботи мережі [9].

Моніторинг бездротової мережі дозволяє контролювати доступ до мережі та виявляти несанкціоновані підключення. Він включає в себе виявлення нових пристроїв, аутентифікацію та авторизацію підключених пристроїв, а також контроль доступу до різних ресурсів мережі.

Забезпечує безпеку мережевих пристроїв та стабільну роботу. Основні аспекти захисту, які пояснюють, як моніторинг забезпечує безпеку бездротових мереж [9]:

– Виявлення незвичайної активності. Моніторинг бездротової мережі дозволяє виявити незвичайну активність, таку як невідомі пристрої, незвичайний трафік або негативні дії.

– Виявлення несанкціонованих точок доступу. Моніторинг виявляє наявність несанкціонованих точок доступу до мережі. Це можуть бути маршрутизатори або точки доступу, які встановлюються без дозволу адміністратора мережі, що створює негативну вразливість і можливість для несанкціонованого доступу до мережі.

– Виявлення вторгнення. Моніторинг бездротової мережі виявляє незвичайну активність, що може свідчити про вторгнення в мережу. Це включає спроби несанкціонованого доступу до мережевих пристроїв, злам паролів, недавні спроби авторизації тощо. Моніторинг дозволяє швидше реагувати на такі вторгнення, блокувати несанкціонований доступ та захищати мережу від небажаних атак.

– Виявлення використання шифрувань. Моніторинг бездротової мережі дозволяє виявити використання несанкціонованих або слабких методів шифрування, які можна поставити під загрозу безпеці мережі. Це включає використання старих протоколів шифрування або неправильну конфігурацію безпеки мережевих пристроїв.

– Виявлення і усунення шкоди. Моніторинг бездротової мережі виявляє фізичні та електромагнітні шкоди, які можуть вплинути на якість зв'язку. Наприклад, можуть бути виявлені перешкоди у вигляді стіни, металевих об'єктів або інших бездротових пристроїв, які можуть впливати на сигнал.

– Захист від DoS-атаки. атаки DoS (відмова в обслуговуванні) можуть спрямовуватись до бездротової мережі, збільшуючи навантаження на неї та перешкоджаючи легітимному використанню ресурсів. Для виявлення від таких атак використовуватися моніторинг трафіку та механізм блокування небажаного або аномального трафіку.

– Звітування подій. Система моніторингу звітує та зберігає дані для кращого виявляти підозрілої активності бездротовій мережі. Реєстрація та аналіз подій може

допомогти вчасно виявити і реагувати на виявлені загрози та атаки, а також забезпечити відповідність стандартам безпеки.

– Виявлення пристроїв. Моніторинг дозволяє виявляти та ідентифікувати пристрої, які підключені до бездротової мережі, і збирати інформацію про них, таку як MAC-адресу, силу сигналу, швидкість передачі даних тощо [9].

– Аналіз пропускної здатності. Моніторинг бездротової мережі дозволяє відстежувати використання пропускної здатності і виявляти можливі перевантаження або недостатню пропускну здатність. Це допомагає ідентифікувати проблеми зі швидкістю передачі даних і приймати заходи для їх вирішення, наприклад, шляхом налаштування маршрутизації або оптимізації розташування точок доступу.

1.3 Методи та засоби моніторингу бездротових мереж

Існує два методи моніторингу: активний та пасивний методи. Активний та пасивний методи моніторингу бездротової мережі відрізняються способом, яким отримується інформація про мережу та пристрої.

Активний моніторинг. Активний моніторинг бездротової мережі включає активну взаємодію з мережею та пристроями шляхом відправлення спеціальних запитів або пакетів даних та аналізу відповідей. Після відправки запиту моніторинговий пристрій або програма отримує відповіді від бездротових пристроїв у мережі. Вони аналізують ці відповіді для отримання інформації про стан мережі [10].

Пасивний моніторинг. Пасивний моніторинг не включає активну взаємодію з мережею. У цій методиці моніторингові пристрої або програми аналізують трафік, який протікає в мережі без надсилання власних запитів. Пасивний моніторинг дозволяє отримати інформацію про стан мережі без додаткового завантаження або втручання [11].

Як правило, активний моніторинг вимагає спеціальних моніторингових пристроїв або програм, які взаємодіють з мережею, тоді як пасивний моніторинг може бути дійсним за допомогою програмних засобів, які прослуховують трафік у

мережі. Обидва методи мають свої переваги і застосовуються залежно від конкретних потреб моніторингу бездротової мережі.

Існує багато засобів моніторингу бездротової мережі такі як:

– програмні засоби;

– апаратні засоби;

– на основі мікрокомп’ютера;

1.3.1 Програмні засоби моніторинг бездротової мережі

Програмні засоби моніторингу бездротової мережі - це спеціальні програми або пакети програм, призначені для перегляду, аналізу та контролю бездротової мережі. Ці програмні засоби можуть надавати різноманітні функції, такі як виявлення несанкціонованих підключень, ідентифікація та відстеження пристроїв у мережі, аномальний трафік, аналіз подій, моніторинг рівня сигналу, швидкість передачі даних та інші [12].

Wireshark. Це один із найпопулярніших і потужних інструментів для аналізу та моніторингу мережі. Wireshark дозволяє перехоплювати та аналізувати пакети даних, які пересилаються по бездротовій мережі. Ви можете вивчати трафік, виявляти якісь проблеми та проводити розробку та тестування мережі [12].

Переваги програми:

– розширені можливості аналізу;

– підтримка різноманітних протоколів, включаючи Ethernet, Wi-Fi, TCP/IP, HTTP, DNS та багато інших;

– кросплатформенність: Wireshark доступний для різних операційних систем, включаючи Windows, macOS і Linux;

– складність для початківців.

Недоліки програми:

– потреба в навчанні;

– обмеження продуктивності системи.

Acrylic Wi-Fi Home. Ця програма надає інформацію про доступні бездротові мережі, їх сигнал і канали. Ви можете перевірити швидкість передачі даних, якість сигналу та інші характеристики мережі. Також Acrylic Wi-Fi Home має функцію

аналізу конфліктів каналів, яка допоможе вам знайти оптимальний канал для вашої мережі [13].

Переваги програми:

– візуалізація бездротової мережі;

– аналіз сигналу і каналів;

– простота використання;

– додаткові функції.

Недоліки програми:

– обмежений функціонал: У порівнянні з деякими іншими програмами;

– доступний лише для певних операційних систем;

– версія частково безплатна, повну версію потрібно придбати окремо.

NetCut. Ця програма дозволяє відстежувати та контролювати підключені пристрої до бездротової мережі. Ви можете переглядати список підключених пристроїв, блокувати небажані пристрої, а також вимічати або відключати доступ до Інтернету для конкретних пристроїв [13].

Переваги програми:

– відстеження та контроль підключених пристроїв;

– простота використання;

– безпека мережі;

– відновлення мережі;

– моніторинг швидкості та використання мережі.

Недоліки програми:

– обмежена функціональність;

– сумісність та платформи;

– вплив на деякі мережеві пристрої;

– сумісність із відсутністю аутентифікації;

PRTG Network Monitor. Це повнофункціональна система моніторингу мережі, яку можна використовувати для бездротової мережі. PRTG дозволяє вам контролювати стан мережі, контролювати сигнал Wi-Fi, швидкість передачі даних, виявляти перевантаження та проблеми з мережевими пристроями. Вона також підтримує аналіз трафіку та генерацію звітів для детального аналізу [14].

Переваги програми:

– комплексний моніторинг мережі;

– гнучкість та розширюваність;

– візуалізація та звіти;

– сповіщення та автоматизація;

– недоліки програми:

– вартість: PRTG Network Monitor є комерційним рішенням, і його вартість може бути високою;

– складність інсталяції та налаштувань;

– обмеження щодо масштабування;

– вимоги до обладнання та ресурсів.

Ekahau HeatMapper. Ця програма призначена для аналізу покриття Wi-Fi у бездротових мережах. Вона дозволяє створювати детальні картографічні зображення зонного покриття, виявляти мережеві проблеми, такі як слабкий сигнал або міжканальні перешкоди. Ви можете використовувати Ekahau HeatMapper для планування мережі, маршрутизаторів і точок доступу, щоб забезпечити оптимальне покриття бездротового сигналу [14].

Переваги програми:

– програма доступна безкоштовно для завантаження;

– простий у використанні;

– вимірювання сигналу Wi-Fi;

– аналіз покриття;

– візуалізація даних.

Недоліки програми:

– обмежені функціональні можливості;

– вимоги до обладнання;

– обмеження розміру зони покриття;

– відсутність регулярних оновлень;

– відсутність продвинутих функцій.

В таблиці 1.2 показано які протоколи та методи використовують програмі засоби для моніторингу бездротової мережі.

Таблиця 1.2 – Протоколи та методи які використовують програми моніторингу

Назва	Протоколи	Метод моніторингу
Wireshark	Wireshark підтримує широкий спектр мережевих протоколів, включаючи Ethernet, IP, TCP, UDP, HTTP, DNS, SSL/TLS та багато інших.	Пасивний метод
Acrylic Wi-Fi Home	Acrylic Wi-Fi Home підтримує основні бездротові протоколи, такі як 802.11a/b/g/n/ac.	Пасивний метод
NetCut	NetCut працює на рівні мережевого доступу і не обмежується підтримкою конкретних протоколів. Вона може взаємодіяти з будь-якими підключеними пристроями в мережі.	Пасивний метод
PRTG Network Monitor	PRTG Network Monitor підтримує багато мережевих протоколів, включаючи SNMP (Simple Network Management Protocol), WMI (Windows Management Instrumentation), ICMP (Internet Control Message Protocol), HTTP (Hypertext Transfer Protocol) та багато інших.	Пасивний метод
Ekahau HeatMapper	Ekahau HeatMapper не працює на рівні мережевих протоколів, але використовує дані Wi-Fi для аналізу покриття та якості сигналу.	Активний метод

1.3.2 Апаратні засоби моніторингу бездротової мережі

Апаратні засоби моніторингу бездротової мережі - це фізичні пристрої або компоненти, які використовують для перегляду, аналізу та контролю бездротової мережі. Вони надають можливість отримувати інформацію про стан мережі, пристроїв, трафіку та інших параметрів, для забезпечення ефективного функціонування та безпеки мережі. Апаратні засоби моніторингу можуть включати спеціалізовані мережеві пристрої, такі як мережеві аналізатори, спеціальні маршрутизатори та точки доступу з підтримкою моніторингу, спеціалізовані мережеві комутатори [15].

Бездротовий аналізатор спектру. Це пристрій, який вимірює сигнали в бездротовому спектрі та надає інформацію про наявність і потужність сигналів різних пристроїв у мережі. Він дозволяє виявити перешкоди, інтерференцію та зайнятість каналів, допомагаючи оптимізувати розташування точки доступу та налаштування мережі. Пристрій працює на основі активного методу моніторингу бездротової мережі. На рисунку 1.1 показано як виглядає аналізатор спектру MESA Deluxe [16].

Переваги:

– виявлення ідентифікації перешкод;

– виявлення несанкціонованого доступу;

– моніторинг використання каналів;

Недоліки:

– висока вартість;

– складність використання;

– обмежена мобільність;

– вплив на мережевий трафік.

Рисунок 1.1 – Аналізатор спектру бездротової мережі MESA Deluxe

Спеціальні маршрутизатори та точки доступу, які мають функції моніторингу мережі, є пристроями, розробленими для забезпечення розширених можливостей контролю, аналізу та моніторингу бездротових мереж. Вони зазвичай використовуються в бізнес-середовищах, де потрібен докладний моніторинг мережевих активів та оптимізація мережевої продуктивності. Спеціальні маршрутизатори та точки доступу моніторять бездротову мережу на основі пасивного методу. На рисунку 1.2 та 1.3 показано як виглядають спеціальний маршрутизатор та точки доступу [17].

Основні функції та можливості спеціальних маршрутизаторів та точок доступу для моніторингу мережі включають:

– моніторинг трафіку;

– виявлення проблем;

– захист мережі;

– керування мережею;

– генерація звітів;

Для прикладу візьмемо, маршрутизатор RUT950 є високопродуктивним промисловим маршрутизатором 4G LTE Wi-Fi, розроблений як основний / резервний інтернет-джерело і гарантує надійне інтернет-з'єднання з високою пропускною здатністю та надмірністю даних [17].

Сумісний із системою віддаленого керування Teltonika Networks (RMS) для простого та безпечного моніторингу. Він має чотири порти Ethernet, два зовнішніх власника SIM-карти, цифрове введення/виведення та світлодіодні індикатори стану рівня сигналу .

Пристрій працює на базі RutOS - уніфікованої операційної системи для всіх маршрутизаторів Teltonika Networks, відрізняється високим ступенем безпеки та можливістю налаштування. Спеціалізовані комутатори моніторять мережу на основі пасивного методу.

Вигляд маршрутизатора RUT950 наведено на рисунку 1.2. На рисунку 1.3 наведено вигляд спеціалізованої точки доступу.

Керований комутатор з підтримкою моніторингу (віддзеркалення) портів (функція, що дає змогу перенаправляти трафік з одних портів на певний порт комутатора) — ідеальний пристрій для мережевого моніторингу [17].

Рисунок 1.2 – Вигляд маршрутизатора RUT950

Рисунок 1.3 – Вигляд спеціалізованої точки доступу

Налаштування віддзеркалення портів залежать від моделі і виробника. На рисунку 1.4, 1.5 показано два типові варіанти з використанням з віддзеркалення портів.

Рисунок 1.4 – Варіант 1 використання віддзеркалення портів

Рисунок 1.5 – Варіант 2 використання віддзеркалення портів

У цьому варіанті головний комутатор має функцію віддзеркалення портів. ПК моніторингу підключений до «дзеркального» порту, на який переправляється весь трафік з локальних робочих станцій і маршрутизатора. Комутатор можна налаштувати на перенаправлення даних з одного або з кількох портів [18].

Для наглядного прикладу розглянемо комутатор FoxGate S6224-S2C2 - повністю гігабітний комутатор рівня L2 мережевої моделі OSI, що має розширені засоби мережевої безпеки, може працювати на рівні доступу або агрегації мереж інтернет-провайдерів, великих підприємств і підприємств малого та середнього бізнесу (SMB), а також в датацентрах. Гігабітний комутатор FoxGate S6224-S2C2-POE, підтримує стандарти 802.3af і 802.3at (рисунку 1.4, 1.5) [19].

Основні можливості:

– розширений функціонал по налаштуванню VLAN;

– підтримка як 802.1q і port based VLAN, так і MAC based VLAN;

– підтримка технології Stacking Vlan (подвійне тегування, QinQ, до 16000000 VLAN), що дозволяє застосовувати технологію VLAN per user;

– підтримка технології Vlan VPN (подвійне тегування, QinQ);

– Voice Vlan дозволяє простим способом додати IP Phone в Voice VLAN для забезпечення ефективної якості обслуговування QoS;

– висока надійність підтримка засобів цифрової діагностики та моніторингу несправності з'єднань по оптичному волокну і крученій парі VCT & DDM (Digital Diagnostic Monitoring;

– підтримка протоколів STP, RSTP, MSTP стандартів 802.1D, 802.1W, 802.1S;

– підтримка технології port aggregation стандарту 802.3ad.

На рисунку 1.6 наведено вигляд спеціалізованого мережевого комутатора FoxGate S6224-S2C2.

Рисунок 1.6 – Спеціалізований мережевий комутатор FoxGate S6224-S2C2.

1.3.3 Мобільні засоби моніторингу бездротової мережі

Wi-Fi Analyzer (доступно для Android) дозволяє вам відстежувати сигнали, виявляти перешкоди та визначати найкращі канали для вашої бездротової мережі [22]. Основні функції Wi-Fi Analyzer: сканування мереж, графіки сигналу, визначення найкращого каналу, Графічне відображення покриття мережі, підтримка фільтрів. Wi-Fi Analyzer є потужним інструментом для моніторингу бездротової мережі на Android-пристроях. Додаток сканує доступні бездротові мережі у вашому районі і відображає їх на графіку. Ви можете побачити назви мереж, сили сигналу (RSSI), канали та типи шифрування. Ви можете створити графічну карту покриття мережі у вашому приміщенні або на відкритій території. Це допомагає визначити місця зі слабим покриттям та вирішити проблеми зі зв'язком.

Fing - Network Scanner (доступно для Android та iOS) [23] дозволяє сканувати мережу, визначати підключені пристрої, аналізувати безпеку мережі та перевіряти

швидкість з'єднання. Основні можливості Fing - Network Scanner включають: сканування мережі, перевірка безпеки мережі, перевірка швидкості з'єднання, діагностика мережі, повний список пристроїв. Fing - Network Scanner є корисним інструментом для моніторингу та аналізу бездротових мереж.

Додаток сканує вашу локальну мережу та виявляє підключені до неї пристрої. Ви можете побачити список підключених пристроїв, їх IP-адреси, виробників, MAC-адреси та іншу інформацію. Fing дозволяє вам перевіряти безпеку своєї мережі та виявляти потенційні загрози. Ви можете перевірити, чи є відкриті порти, наявність потенційних ризиків безпеки та знайти проблеми, які можуть стати джерелом уразливості. Fing дозволяє вам проводити діагностику мережі та виявляти можливі проблеми. Ви можете виконати ping до певних пристроїв, перевірити наявність DNS-проблем, виявити перешкоди та інші аспекти, що впливають на якість зв'язку. Ви можете переглянути повний список пристроїв, які були раніше підключені до вашої мережі. Це допоможе вам виявити незрозумілі або небажані пристрої, що підключаються до вашої мережі.

iNet (доступно для iOS) [24] пропонує інструменти для аналізу мережі, сканування пристроїв, перевірки швидкості з'єднання та перевірки безпеки мережі. Цей додаток має наступні основні функції: сканування мережі, перевірка підключених пристроїв, аналіз мережі, пошук пристроїв у мережі, Інструменти безпеки мережі. iNet є потужним інструментом для моніторингу та аналізу бездротових мереж на пристроях iOS. iNet дозволяє сканувати бездротові мережі у вашому оточенні. Він виявляє доступні мережі, відображає їх інформацію, таку як назва мережі, сила сигналу (RSSI), канал та тип шифрування. Додаток відображає список пристроїв, які підключені до вашої мережі. Ви можете побачити їх IP-адреси, MAC-адреси та виробників. Додаток надає інструменти для перевірки безпеки вашої мережі. Ви можете виявляти вразливості, такі як слабкі паролі або незахищені налаштування мережі, та отримувати рекомендації щодо їх виправлення. На таблиці 1.3 наведено можливості та зображення мобільних засобів моніторингу бездротової мережі.

Таблиця 1.3 – Мобільні засоби моніторингу бездротової мережі

Назва	Зображення	Можливості
WiFi Analyzer		• WiFi Analyzer сканує оточуючі Wi-Fi мережі і надає інформацію про кожну з них; • Додаток відображає графіки та діаграми, що допомагають вам візуально зрозуміти розподіл Wi-Fi мереж; • WiFi Analyzer показує силу сигналу Wi-Fi мережі WiFi Analyzer може зберігати історію сканування та результати аналізу. в реальному часі.
Fing - Network Tools		• Fing дозволяє сканувати Wi-Fi мережі у вашому оточенні і надає детальну інформацію про кожну мережу; • Виявлення підключених пристроїв; • Якщо у вас виникають проблеми зі з'єднанням до Wi-Fi мережі, Fing допомагає виявити можливі причини та знайти рішення; • Налаштування мережі; • Історія та збереження результатів;
iNet		• Сканування мережі; • Перевірка підключених пристроїв; • Аналіз мережі; • Пошук пристроїв у мережі.

1.4 Різновиди та функціональні можливості мікрокомп'ютерів

Мікрокомп'ютери - це невеликі комп'ютери, які фактично мають невеликі розміри і обмежені обчислювальні можливості. Вони призначені для виконання простих завдань обробки даних і керування певними пристроями чи системами [25]. Мікрокомп’ютери стали дуже популярними останніми роками через зростання інтересу до Інтернету речей (IoT) та вбудованих систем. Їх можна знайти в різних пристроях, починаючи від побутових речей, таких як розумні датчики, освітлення, системи безпеки, і закінчуючи промисловими пристроями, такими як автоматизовані системи контролю та моніторингу [26].

Мікрокомп’ютери зазвичай мають такі переваги:

– Невеликі розміри. Вони мають компактний дизайн і низьку вартість виробництва, що дозволяє їх використовувати в пристроях з обмеженим простором.

– Енергоефективність. Мікрокомп’ютери регулярно працюють на низьких рівнях енергоспоживання. Це дозволяє їм працювати в автономному режимі на батареях або енергозберігаючих джерелах живлення.

– Здатність до підключення до мережі. Мікрокомп'ютери обов'язково мають вбудовані модулі Wi-Fi, Bluetooth або інші способи бездротового зв'язку. Це дозволяє їм підключатися до Інтернету або спілкуватися з іншими пристроями в мережі.

– Гнучкість програмування. Мікрокомп’ютери також підтримують різноманітні програмні мови та середовище, що дозволяє розробникам створювати різноманітні програми та забезпечувати взаємодію з іншими пристроями та системами.

– Розширюваність. Багато мікрокомп'ютерів мають можливість підключення додаткових модулів і розширювальних карт, таких як сенсори, актуатори, засоби зберігання даних, модулі зв'язку тощо. Це дає можливість розширити функціональність і можливості пристрою в залежності від конкретних вимог і потреб.

– Стабільність і надійність. Мікрокомп'ютери визначені відповідно до вимог до надійності і стабільності роботи. Вони можуть мати вбудовані механізми захисту від перенапруги, перегрівання, збоїв живлення та інших факторів, які можуть вплинути на їх роботу. Це робить їх надійними і стійкими до впливу зовнішніх чинників.

– Широкий спектр програм. Мікрокомп'ютори знаходять застосування в різних сферах, включаючи домашні автоматизовані системи, медичні пристрої, транспортні системи, промислові контролери, сільське господарство, робототехніку та багато інших галузей. Вони можуть виконувати різноманітні завдання, починаючи від збору даних і водіння до аналізу та передачі інформації.

– Відкритість і спільнота. Багато мікрокомп’ютерів базуються на відкритих стандартах та платформах, що сприяє розвитку активної спільноти розробників та ентузіастів. Це означає, що робочі відкриті документації, зразки програмного забезпечення та інструменти, які дозволяють широкому колу людей внести свій внесок у розробку та покращення функціональності цих мікрокомп’ютерів.

– Вартість. Мікрокомп'ютери мають доступну вартість, що робить їх привабливими для широкої аудиторії. Вони є менш дешевими в порівнянні з

традиційними комп'ютерами, що дозволяє використовувати їх у масових проектах або вбудовувати в інші пристрої без значного збільшення витрат.

– Інтероперабельність. Багато мікрокомп'ютерів підтримують стандарти та протоколи, які дозволяють їм співпрацювати з іншими пристроями та системами. Це забезпечує можливість інтеграції їх в екосистеми IoT, різні пристрої можуть обмінюватися даними і взаємодіяти між собою [].

Для прикладу розглянемо мікрокомп’ютерні платформи Raspberry Pi – це серія одноплатних комп'ютерів, розроблених Фондом Raspberry Pi. Ці комп'ютери мають невеликий розмір, але водночас можуть виконувати багато завдань. Є популярним мікрокомп'ютером, відкривають широкі можливості для розробників та всіх, хто бажає зайнятися програмуванням або створенням електронних пристроїв. Вони можуть бути використані для різних цілей, від створення простих проектів до розробки складних систем. [27].

Raspberry Pi заснований на архітектурі ARM і має універсальний набір вхідно-вихідних портів, що дозволяє підключати різноманітні пристрої і сенсори. Приклади декількох видів та моделей Raspberry Pi наведені у таблиці 1.4.

Таблиця 1.4 – Види та моделі Raspberry Pi

Види	моделі	зображення	характеристики
Raspberry Pi Zero	Zero W Zero WH.		• Процесор 1-ядерний ARM-процесор. • Оперативна пам'ять 512 МБ LPDDR2. • Вбудований бездротовий модуль: підтримка Wi-Fi та Bluetooth 4.1. • Відеовихід міні-HDMI • Звуковий вихід 3,5-мм аудіороз'єм. • Зовнішній інтерфейс 1x micro-USB порт для живлення та передачі даних. • Слот для карт пам'яті мікро-SD. • GPIO-роз'єми 40-контактний роз'єм GPIO.
Raspberry Pi 4	Model B з 2 ГБ Model B з 4 ГБ Model B з 8 ГБ		• Процесор: 4-ядерний ARM Cortex-A72 • Обсяг оперативної пам'яті: 8 ГБ LPDDR4 • Відеовихід: 2 мікро-HDMI • Зовнішній інтерфейс: 2x USB 3.0 порти, 2x USB 2.0 порти, 1x Gigabit Ethernet порт, 1x 3,5-мм аудіороз'єм. • Бездротові можливості: підтримка Wi-Fi 802.11ac і Bluetooth 5.0. • Зовнішній накопичувач: слот для карт пам'яті micro-SD. • GPIO-роз'єм: 40-контактний GPIO-роз'єм

Продовження таблиці 1.4

Raspberry Pi Compute Module	Module 3(CM3) Module 4(CM4) Module3Lite (CM3 Lite)		• Процесор: 4 ядра ARM Cortex-A72 • Оперативна пам'ять (RAM): 8ГБ LPDDR4. • Зберігання даних: не має вбудованої флеш-пам'яті. • Графічний процесор: Вбудований графічний процесор VideoCore VI, • Роз'єми: має роз'єми 2x USB 2.0, 2x MIPI DSI, 2x MIPI CSI, HDMI, Gigabit Ethernet, PCIe, UART, I2C, SPI та GPIO. • Операційна система: Raspbian • Розмір становить приблизно 55 мм\40 мм

Розглянемо вже готовий засіб моніторингу Wi-Fi мережі на базі мікрокомп’ютера «Wi-Fi Pineapple» – це пристрій, розроблений спеціально для моніторингу та аналізу бездротових мереж. Він базується на мікрокомп'ютері Alfa Network AWUS036H, який в свою чергу використовує чіпсет Realtek RTL8187L, і має вбудований Wi-Fi адаптер та спеціалізоване програмне забезпечення, яке надає розширені можливості для моніторингу, збору даних та виявлення потенційних проблем бездротової мережі (рисунок 1.4).

Рисунок 1.4 – Вигляд пристрою Wi-Fi Pineapple

Цей пристрій може бути використані для збору інформації про бездротову мережу, аналізу трафіку, виявлення потенційних проблем, таких як перенавантаження або зловживання, а також для здійснення моніторингу безпеки мережі. [28].

1.5 Формалізація вимог та постановка задачі

На даний час самою поширеною бездротовою мережею є Wi-Fi мережа, тому дана мережа є доволі вразливою для різних загроз.

У зв'язку з ростом загроз безпеки, бездротової мережі Wi-Fi, таких як несанкціонований доступ, поширення шкідливих програм, перехоплення даних та атаки з використанням великого потоку запитів або високого рівня трафіку, стає важливим мати ефективні методи моніторингу та захисту. [29].

Проаналізовано активний та пасивний методи моніторингу бездротових мереж. Проаналізовано програмні, мобільні та апаратні засоби моніторингу бездротових мереж. Програмні засоби для ПК, мають такі недоліки як: для використання потрібні навички та розуміння, деякі програми надаються з обмеженим функціоналом, так як це програми для ПК доволі незручно переносити з місця на місце, також деякі програми впливають на мережевий трафік. Мобільні засоби моніторингу мають ряд недоліків такі як: обмежена продуктивність, великі енергоспоживання, неможливість вільно використовувати смартфон під час моніторингу. Спеціальні апаратні засоби, які були розглянуті, мають такий ряд недоліків: обмежений функціонал, великі розміри, складність у використанні, обмежена мобільність, впливають на мережевий трафік.

З урахуванням цих обмежень і потреб користувачів, доцільно розробити новий засіб моніторингу на основі мікрокомп'ютера. Цей засіб повинен працювати надійно і без збоїв, постійно моніторити бездротову мережу, мати компактні розміри, бути легкодоступним, енергоефективним, мати зручний користувацький інтерфейс та просте використання.

Цей засіб може бути корисним для широкого спектру користувачів, зокрема:

– Системних адміністраторів, які бажають моніторити та діагностувати проблеми у своїй бездротовій інфраструктурі.
– Професіоналів з безпеки мережі, які виконують аудит безпеки та виявлення потенційних загроз у бездротових мережах.

– Домашніх користувачів, які бажають забезпечити безпеку своєї бездротової домашньої мережі та контролювати підключені пристрої.

Основний функціонал, який має бути реалізований у засобі на базі мікрокомп'ютера, включає:

– Перехоплення та аналіз пакетів даних у бездротовій мережі для виявлення потенційних загроз або несанкціонованої активності.
– Надання інформації про бездротову мережу, її параметри та рівень сигналу.
– Відстежування та контроль підключених пристроїв до мережі для виявлення недозволених або небажаних пристроїв.
– Виявлення перевантажень та проблем з мережевими пристроями, що дозволить реагувати на них своєчасно та забезпечити стабільну роботу мережі.

Розробка такого засобу на базі мікрокомп'ютера дозволить забезпечити ефективний моніторинг бездротової мережі Wi-Fi з простотою використання та широким функціоналом, що відповідає вимогам різних категорій користувачів.

Для розробки засобу моніторингу бездротової мережі, а саме Wi-Fi, на базі мікрокомп’ютера візьмемо Raspberry Pi. Він повинен бути енергоефективним, мати компактний розмір, бути легкодоступним, мати зручний користувацький інтерфейс. Засіб повинен надавати інформацію про Wi-Fi мережу, моніторити підключені пристрої, мати можливість віддаленого керування, аналізувати трафік, виявляти такі загрози як Dos-атака на мережу та ARP-spoofing.

2. ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Структура засобу моніторингу

Засіб моніторингу бездротової мережі на основі мікрокомп’ютера, повинен моніторити обрану мережу і надсилати дані створений телеграм-бот.

На рисунку 2.1 представлена структурна схема засобу моніторингу Wi-Fi мережі.

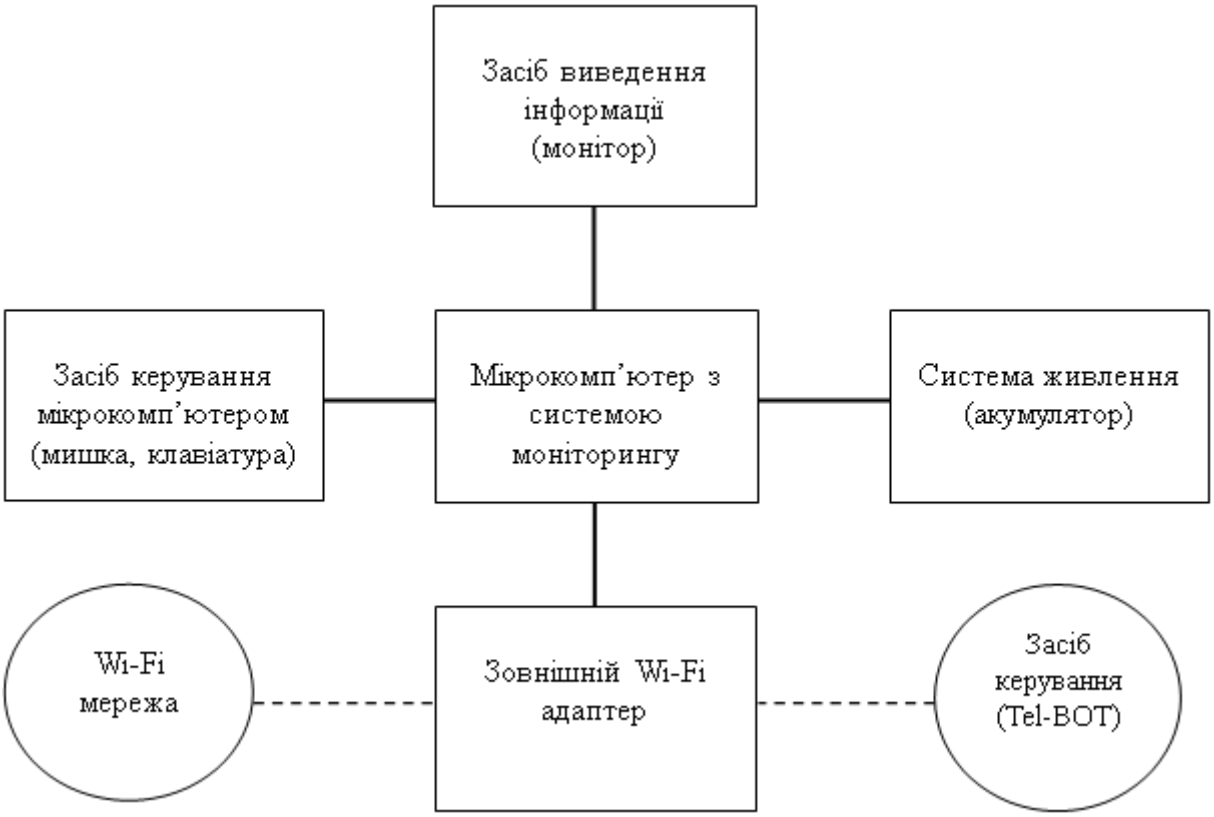


Рисунок 2.1 – Структурна схема засобу моніторингу

Засіб складається з таких частин:

1. мікрокомп’ютер Raspberry Pi;
2. система живлення не менше 5V та 2,5A;
3. засіб керування – мишка та клавіатура;
4. засіб віддаленого керування – телеграм-бот;
5. зовнішній Wi-Fi-адаптер який працює в режимі моніторингу;
6. засіб виведення інформації монітор або екран.

Raspberry Pi використовується як основний контролер, який керує процесом моніторингу Wi-Fi мережі. Це компактний одноплатний комп'ютер, який

використовується як основний контролер системи моніторингу. Raspberry Pi має достатньо потужний процесор та операційну пам'ять, щоб виконувати необхідні завдання збору та аналізу даних Wi-Fi мережі. Має мати програмний засіб для збору і аналізу даних, який зчитує дані, отримані від Wi-Fi-адаптера в режимі моніторингу. Це програмне забезпечення може включати в себе пакети для збору і аналізу пакетів Wi-Fi, розрахунку потужності сигналу, виявлення проблем в мережі, моніторингу трафіку тощо.

Засіб має мати зовнішній Wi-Fi-адаптер який підключається до Raspberry Pi і дозволяє отримувати доступ до Wi-Fi мережі. Цей адаптер переводиться в режим моніторингу, що дозволяє отримувати детальну інформацію про пакети, статистику та потужність сигналу в мережі.

Для зручного перегляду та керування моніторингом використовується телеграм-бот. Користувачі можуть отримувати статистику, налаштування та результати моніторингу через телеграм на підключеному пристрої. Телеграм-бот забезпечує зручний та інтуїтивно зрозумілий інтерфейс для взаємодії з системою моніторингу.

Для стабільної роботи мікрокомп'ютера Raspberry Pi потрібне живлення не менше 5V та 2,5A. Для цього можна використовувати звичайний зарядний блок та шнур для телефонних пристроїв з microUSB портами. Також можна використовувати зовнішні акумулятори, які надають достатню потужність для безперебійної роботи мікрокомп'ютера.

Загальними зусиллями ці складові працюють разом, дозволяючи ефективно моніторити Wi-Fi мережу, збирати та аналізувати дані, а також забезпечувати зручний інтерфейс користувача для взаємодії та керування системою моніторингу.

2.2 Алгоритм налаштування мікрокомп’ютера для моніторингу

1. Спочатку потрібно встановити підходящу операційну систему для Raspberry Pi на microSD. Найбільш підходяща ОС для Raspberry є система «Raspbian», розроблена спеціально для даного мікрокомп’ютера. Скачати образ на робочий ПК (ноутбук) який має порт для встановлення мікроSD карти та встановіть

ОС «Raspbian» на SD-карту обсягом не менше ніж 8ГБ. Після встановлення вставте карту в порт на Raspberry Pi та підключіть до живлення.

2. Після встановлення ОС та підключення SD-карит до Raspberry Pi підключіть до Raspberry Pi монітор, клавіатуру та мишу для подальшого налаштування.

Після запуску мікрокомп'ютера завантажується вікно ОС «Raspbian». При першому запуску Raspbian ми побачимо «Майстер привітання» (рисунок 2.).

Рисунок 2.2 – Вікно ОС при першому запуску Raspbian

3. Далі натиснувши кнопку "Next" вибираємо країну, мову та часову зону (рисунок 2.). Якщо використовується клавіатура з розкладкою для США (US-layout keyboard), то позначається галочкою відповідне поле, і потім натиснувши кнопку "Next" відкривається наступне меню з налаштуваннями.

Рисунок 2.3 – Вікно вибору країни мови та часового поясу

4. На наступному екрані потрібно встановити пароль користувача для безпеки. Створюється та водиться власний пароль у відповідне поле та потрібно повторити його для перевірки у наступному полі (рисунок 2.). Відзначивши чек-бокс "Hide characters" ("Сховати символи"), відобразиться введений пароль для того, щоб візуально перевірити його на збіг в обох полях. Потім знову натиснувши кнопку "Next" відкривається наступне вікно.

Рисунок 2.4 – Встановлення пароль користувача для безпеки

5. На наступному екрані потрібно вибрати бездротову мережу WiFi зі списку доступних. У списку, потрібно знайти та підключитись до вашої мережі. Натисніть "Next" для з'єднання з бездротовою мережею.

Рисунок 2.5 – Вибір бездротової мережі WiFi зі списку доступних

6. На наступному екрані запропоновано перевірити оновлення операційної системи та іншого програмного забезпечення, що встановлюється за замовчуванням, та встановити необхідні оновлення. ОС Raspbian регулярно перевіряє наявність оновлень і встановлює їх, оскільки це пов'язано зі стабільністю роботи системи, усунення вразливостей та більш стабільною роботою програмного забезпечення (рисунок 2. 6). Для встановлення необхідних оновлень натисніть "Next" або "Skip" для пропуску цього кроку. Встановлення оновлень займе кілька хвилин, дочекайтеся її закінчення. Після завершення встановлення оновлень система повідомить Вам "System is up to date" ("Система оновлена"). Просто натисніть "OK".

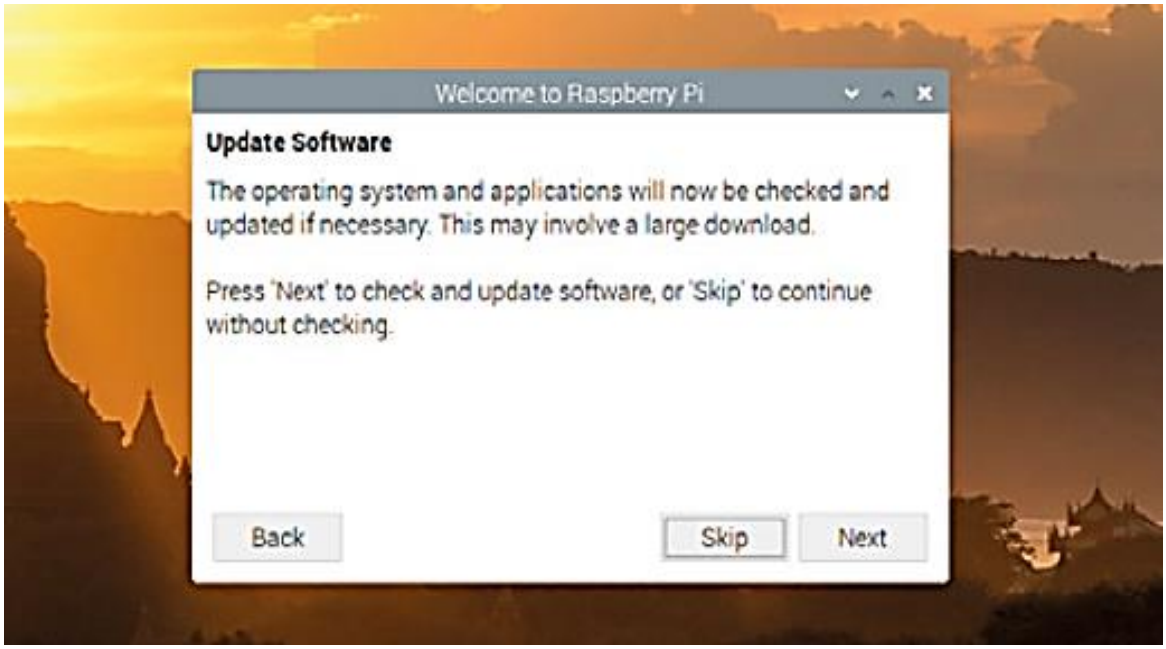


Рисунок 2.6 – Підтвердження перевірки оновлення операційної системи

7. На завершальному вікні «Майстра вітання» натиснувши кнопку «Restart» ми завершуємо налаштування та перезавантажуємо мікрокомп’ютер, оскільки всі зміни використовуються лише при перезавантаженні системи. За потреби можна продовжувати користуватися системою в поточному стані, натиснувши клавішу «Later» або повернутися назад для коригування попередніх кроків, натиснувши «Back» (рисунок 2.7).

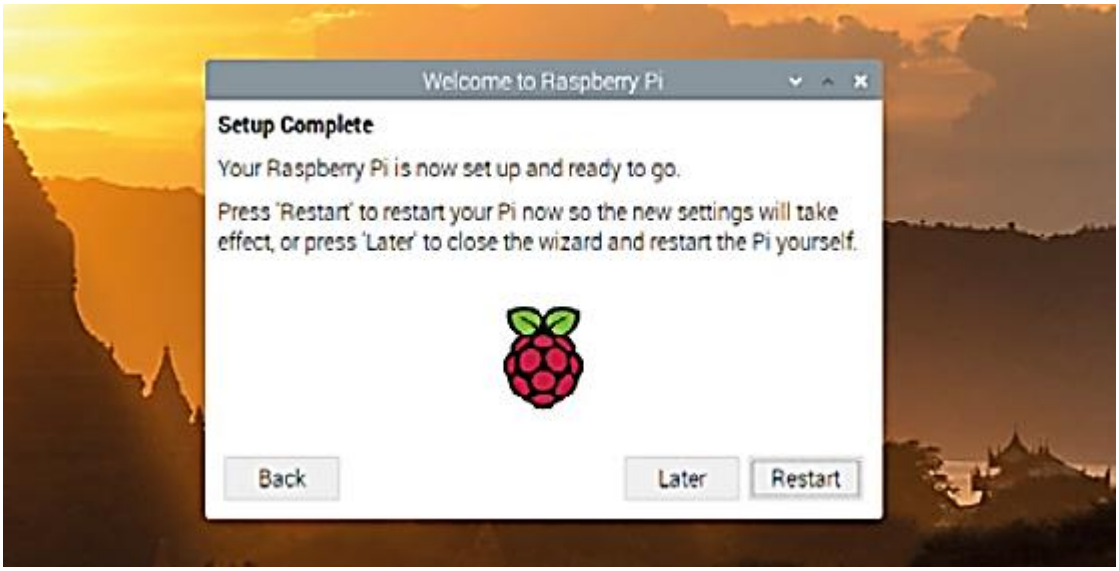


Рисунок 2.7 – завершальне вікно «Майстра вітання»

Після виконання всіх цих кроків можна повноцінно використовувати мікрокомп’ютер. Тепер можна підключитись до мережі та завантажити необхідні програми та драйвера.

8. Ознайомлення з файлами та папками в яких зберігаються основні дані (програми, тексти, відео, зображення тощо) розміщені у домашньому каталозі «home directory» (рисунок 2.8),. Щоб побачити домашній каталог, клацніть значок малинки для відкриття меню, виберіть «Accessories», потім клацніть «File Manager», щоб завантажити його.

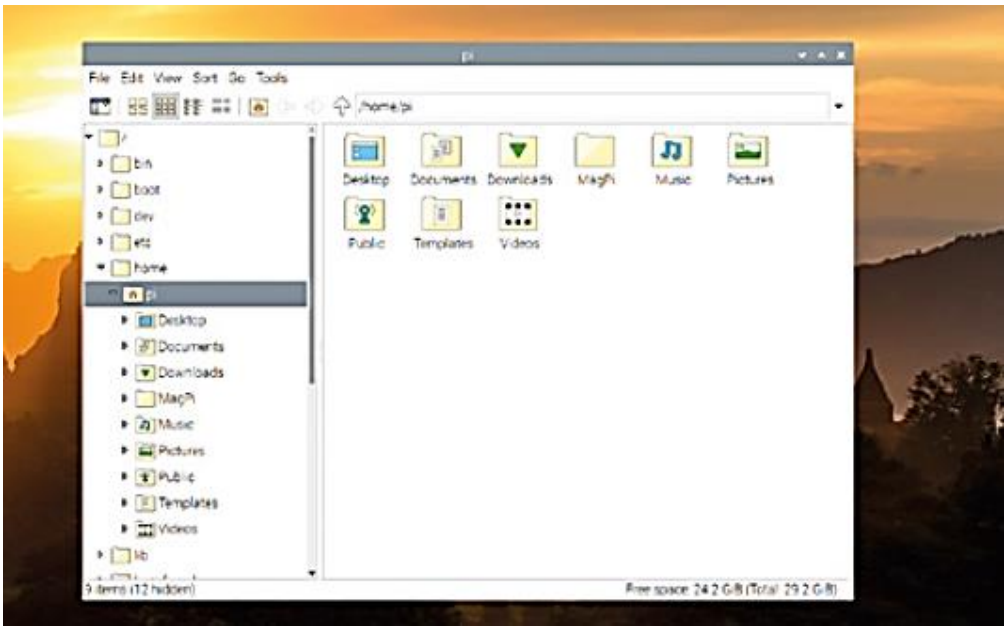


Рисунок 2.8 –Диспетчер файлів Raspberry

Диспетчер файлів дозволяє переглядати файли та папки, також відомі як каталоги, на карті microSD Raspberry Pi, а також на будь-яких знімних пристроях зберігання, таких як USB-флешки, які ви підключаєте до USB-портів Raspberry Pi. Коли ви вперше відкриваєте його, він автоматично потрапляє до вашого домашнього каталогу. Тут ви знайдете ряд інших папок, відомих як підкаталоги, які, як і меню, згруповані за категоріями.

9. Тепер потрібно мати Wi-Fi- адаптер з підтримкою рижиму моніторингу. Підключивши адаптер до мікрокомп’ютера потрібно перевірити чи має він потрібний на режим моніторингу, для цього потрібно відкрити консольне вікно на тиснувши на чорну іконку в лівому верхньому кутку. Щоб отримати дані про режими та інші данні Wi-Fi адаптер потрібно вести команду «iw list». Прокрутивши вивід команди вниз знаходимо розділ з назвою «Supported interface modes». Там буде перелік режимів, які підтримує адаптер. Перевіряємо , чи присутній у переліку режим "Monitor", якщо так, це означає, що Wi-Fi адаптер підтримує режим моніторингу.

2.3 Алгоритм функціонування засобу

Для легкого керування мікрокомп’ютером створюємо та налаштовуємо телеграм-бот який має надсилати команди та отримувати і виводити інформацію про мережу Wi-fi яку моніторить засіб.

Основні команди які має виконувати телеграм-бот:

1. При надсиланні команди «/start» у телеграм-бот, маємо отримати повідомлення з інформацію про Wi-Fi мережу у чат.
2. При надсиланні команди `/devices` у телеграм-бот, маємо отримати повідомлення зі список підключених пристроїв у чат.
3. При надсиланні команди `/dump` у телеграм-бот, через 1 хв маємо отримати повідомлення зі статистикою перехопленого трафіку, у чат.
4. Окрім команд телеграм-бот повинен отримувати повідомлення про можливе порушення безпеки мережі та надсилати у чат.

Для отримання команд від телеграм-бота та генерації повідомлення та відправки інформації, потрібно розробити програмний засіб який буде приймати команди та виконувати функції моніторингу та збору інформацію про Wi-fi мережу. Програма обов’язково має бути встановлена на мікрокомп’ютер та завантажуватись з найвищими правами доступу.

Програма має приймати вище сказані команди та виконувати такі функції:

1. При отримані команди «/start» від телеграм-бота, повинна виконуватися функція збору даних про Wi-Fi мережу та надсилати згенероване повідомлення з інформацією до бота.
2. При отримані команди «/devices» від телеграм-бота, повинна виконуватись функція аналізу та виявлення підключених пристроїв до мережі та згенерувати і відправити дані до бота.
3. При отримані команди «/dump» від телеграм-бота, повинна виконуватись функція перехоплення трафіку. Для збору даних та генерації статистики перехоплених пакетів потрібно, щоб аналіз виконувався певний час і потім надсилав повідомлення з інформацією до бота.
4. Програма моніторингу Wi-Fi мережі також повинна постійно перевіряти на наявність нових пристроїв у мережі та зберігати дані про них. Також програма повинна виявляти та надсилати повідомлення з попередженням про такі загрози як: ARP-spoofing та DOS-атаки на відмову системи.

Для безперервної роботи пристрою потрібно підключити акумулятор який має надавати постійну напругу 5V та силу струму не менше ніж 2,5A. Також підключемо до мікрокомп’ютера монітор або невеличкий екран для кращого контрою та виправлення можливих помилок у програмі.

При запуску програми на мікрокомп’ютері, повинно одразу виявлятися всі підключені пристрої до мережі, і надсилається повідомлення зі списком пристроїв.

Після збереження списку підключених пристроїв засіб починає аналізувати пакети які протікають по мережі для виявлення можливих атак. Засіб повинен постійно аналізувати нові підключені пристрої і перевіряти їх зі збереженим списком на схожість MAC-адрес та IP-адрес для виявлення можливої атаки ARP-spoofing.

Також засіб повинен постійно перевіряє пакети без команди та аналізувати: на який пристрій приходять пакети, скільки, і який розмір цих пакетів. Якщо засіб виявляє що на якийсь пристрій приходять велика кількість пакетів однакового розміру та на один конкретний ресурс то засіб надсилає повідомлення з попередженням про можливість DOS-атаки.

3. РОБОЧЕ ПРОЕКТУВАННЯ СИСТЕМИ

3.1 Обґрунтування вибору інструментальних засобів розробки

Для розробки засобу моніторингу бездротової мережі на базі мікрокомп’ютера є багато різних моделей. Розглянемо декілька моделей:

Raspberry Pi Zero – це стара та базова версія Raspberry Pi, забезпечує обмежені можливості та обчислювальні ресурси. Хоча вона компактна та доступна за ціною, її можливості є недостатніми для потреб моніторингу, особливо якщо потрібно обробляти великий обсяг даних або виконувати потужні алгоритми [30].

Raspberry Pi 1 Model A+/B+ – ці версії пропонують трохи кращі обчислювальні можливості, більше роз'ємів та деякі покращення в аспекті живлення та енергоефективності порівняно з Raspberry Pi Zero. Вони трохи більш потужні для завдань, але все ще обмежені в обчислювальних ресурсах [31].

Raspberry Pi 2 Model B – ця версія включає в себе потужніший 4-ядерний процесор і більше оперативної пам'яті, що дозволяє прискорити обчислення та оптимізувати роботу з мережевими протоколами. Raspberry Pi 2 Model B може бути корисною для моніторингу, особливо якщо потрібно обробляти середній обсяг даних та виконувати додаткові завдання але має не помірну ціну [32].

Raspberry Pi 3 Model A+ – ця версія пропонує значні покращення в порівнянні з попередніми моделями. Вона має 4-ядерний процесор, 512 МБ оперативної пам'яті, вбудований модуль Wi-Fi та Bluetooth, а також додаткові роз'єми. Raspberry Pi 3 Model A+ є потужною та енергоефективною платою з достатньою кількістю обчислювальних ресурсів та можливостей для моніторингу. Вона може легко обробляти дані, виконувати мережеві протоколи, зберігати дані та взаємодіяти зі зовнішніми пристроями а також має помірну ціну порівняно з іншими [33].

Raspberry Pi 4 Model B – це новий продукт у популярному ряді мікрокомп'ютерів Raspberry Pi. Ключовими характеристиками даного продукту є новий, високопродуктивний 64-розрядний чотириядерний процесор, підтримка двох дисплеїв з роздільною здатністю до 4k через пару micro-HDMI портів, апаратне декодування відео в 4Kp60, від 1-го до 4 х Гб оперативної пам'яті,

дводіапазонна бездротова мережа на 2,4 та 5,0 ГГц, Bluetooth 5.0, Gigabit Ethernet, два порти USB 3.0 та PoE. Двоканальна бездротова LAN і Bluetooth мають модульну сертифікацію, що дозволяє платі використовуватися в кінцевих продуктах зі значно спрощеним тестуванням на відповідність стандартам, покращуючи як вартість, так і час виходу на ринок. Ця модель набагато потужніща але має не зовсім компактний розмір а також має доволі високу ціну порівняно з всіма моделями [34].

З усіх згаданих плат Raspberry Pi 3 Model A+ (рисунок 3.1), має найкращу комбінацію обчислювальної потужності, енергоефективності та можливостей. Має один потр USB та HDMI яких достатньо для роботи засобу моніторингу. Крім того, є роз'єми які дозволяють підключати додаткові сенсори та пристрої для моніторингу. Ця плата має оптимальні можливості для розробки засобу моніторингу, забезпечуючи потрібну функціональність та продуктивність за доступною ціною.

Рисунок 3.1 – Зображення Raspberry Pi 3 Model A+

Для моніторингу, аналізу, та збору даних необхідний Wi-Fi–адаптер з підтримкою режиму монітору. Звичайний Wi-Fi адаптер, який не підтримує режим моніторингу, не дозволить здійснювати ефективний моніторинг Wi-Fi мережі. Такий адаптер буде обмежений у зборі інформації про мережу, не зможе

перехоплювати всі пакети та не зможе надати повну статистику та аналіз мережі. Отже, для успішної реалізації засобу моніторингу Wi-Fi мережі з віддаленим керуванням у вигляді телеграм-бота, необхідно мати адаптер, який підтримує режим монітору [35].

Режим моніторингу є необхідною функцією для Wi-Fi адаптера в системі моніторингу Wi-Fi мережі з кількох причин:

1. Збір даних: Режим моніторингу дозволяє адаптеру перехоплювати всі пакети, які передаються через Wi-Fi мережу, незалежно від їхнього призначення або адресата. Це включає пакети, які не призначені для самого адаптера. Завдяки цьому, засіб моніторингу може збирати повну інформацію про мережу, включаючи заголовки пакетів, потужність сигналу, MAC-адреси пристроїв тощо.

2. Аналіз мережі: Режим моніторингу дозволяє отримати детальну статистику та інформацію про Wi-Fi мережу. Це може включати інформацію про підключені пристрої, швидкість передачі даних, рівень сигналу, використання каналів, тип шифрування тощо. Ця інформація може бути корисною для аналізу потужності сигналу, виявлення перешкод, ідентифікації потенційних проблем у мережі та планування оптимальної конфігурації мережі.

3. Безпека мережі: Режим моніторингу є важливим інструментом для виявлення потенційних загроз безпеці Wi-Fi мережі. За допомогою програмного забезпечення моніторингу, можна виявляти атаки з перехоплення пакетів, підбору паролів, підробки MAC-адресів, дії зловмисників та інші загрози безпеці. Виявлення таких загроз дозволяє приймати заходи щодо забезпечення безпеки мережі та захисту пристроїв [35].

Для засобу вибрано зовнішній Wi-Fi- адаптер TL-WN722N від виробника TP-Link (рисунок 3.2). Цей адаптер має інтерфейс USB 2.0, що дозволяє підключати адаптер до будь-якого комп'ютера або пристрою з USB-портом, підтримує бездротове з'єднання стандарту 802.11n, забезпечуючи швидкість до 150 Мбіт/с., має зовнішню високоефективну антену, яка забезпечує кращий прийом та передачу сигналу Wi-Fi, підтримує режим моніторингу, що дозволяє перехоплювати та аналізувати всі пакети, що проходять через Wi-Fi мережу, сумісний з операційними системами Windows, Mac OS та Linux, що робить його універсальним і зручним для

використання з різними пристроями, підтримує різні методи шифрування, включаючи WEP, WPA/WPA2 та WPA-PSK/WPA2-PSK, що дозволяє забезпечити безпеку бездротового з'єднання [36].

Рисунок 3.2 – Зображення Wi-Fi- адаптер TL-WN722N від виробника TP-Link

На рисунку 3.3 зображено повністю зібрана апаратна частина засобу моніторингу бездротової мережі на базі мікрокомп’ютера.

Рисунок 3.3 – Схеми засобу – апартної частини

Для розробки програми для моніторингу бездротової мережі і зв’язку з використанням телеграм-бота, вибрано мову Python 3.9.

Python є однією з найпростіших мов програмування для вивчення та розробки. Вона має простий і зрозумілий синтаксис, що дозволяє швидко розпочати програмування без необхідності в глибокому розумінні складних концепцій. Це особливо важливо для початківців або тих, хто швидко хоче розпочати розробку свого проекту моніторингу Wi-Fi мережі.

Python має велику та активну спільноту розробників, що сприяє наявності багатого вибору бібліотек і модулів для розробки різноманітних додатків. У випадку моніторингу Wi-Fi мережі, ви можете скористатися такими бібліотеками, як scapy, pywifi, wifi, python-nmap та іншими, щоб отримати доступ до необхідної функціональності [37].

Python є крос-платформеною мовою програмування, можна розробляти свій засіб моніторингу Wi-Fi мережі на різних операційних системах, таких як Windows, macOS та Linux, з використанням одного і того ж коду. Це робить Python гнучким і зручним вибором для розробки на різних пристроях, включаючи Raspberry Pi.

Також для використання телеграм-бота та програмування його використовується мова Python. Загалом, Python є відмінним вибором для розробки засобу моніторингу Wi-Fi мережі, з'єднаного з Telegram-ботом, завдяки своїй простоті, широкому вибору бібліотек, крос-платформеній підтримці, багатофункціональності та наявності розширеної документації та підтримки спільноти [38].

3.2 Розробка алгоритмів функціонування програми

Програма для моніторингу Wi-Fi-мережі має приймати команди від телеграм бота та виконувати такі дії

1. При отримані команди «/start» від телеграм-бота, програма повинна виконувати збір даних про Wi-Fi мережу, перевіряти чи підключений до засобу адаптер, перевіряти чи на можливість отримання інформації, генерувати

повідомлення із отриманою інформацією та надсилати згенероване повідомлення з інформацією до бота. На рисунку 3.4 наведено схему обробки команди /start.

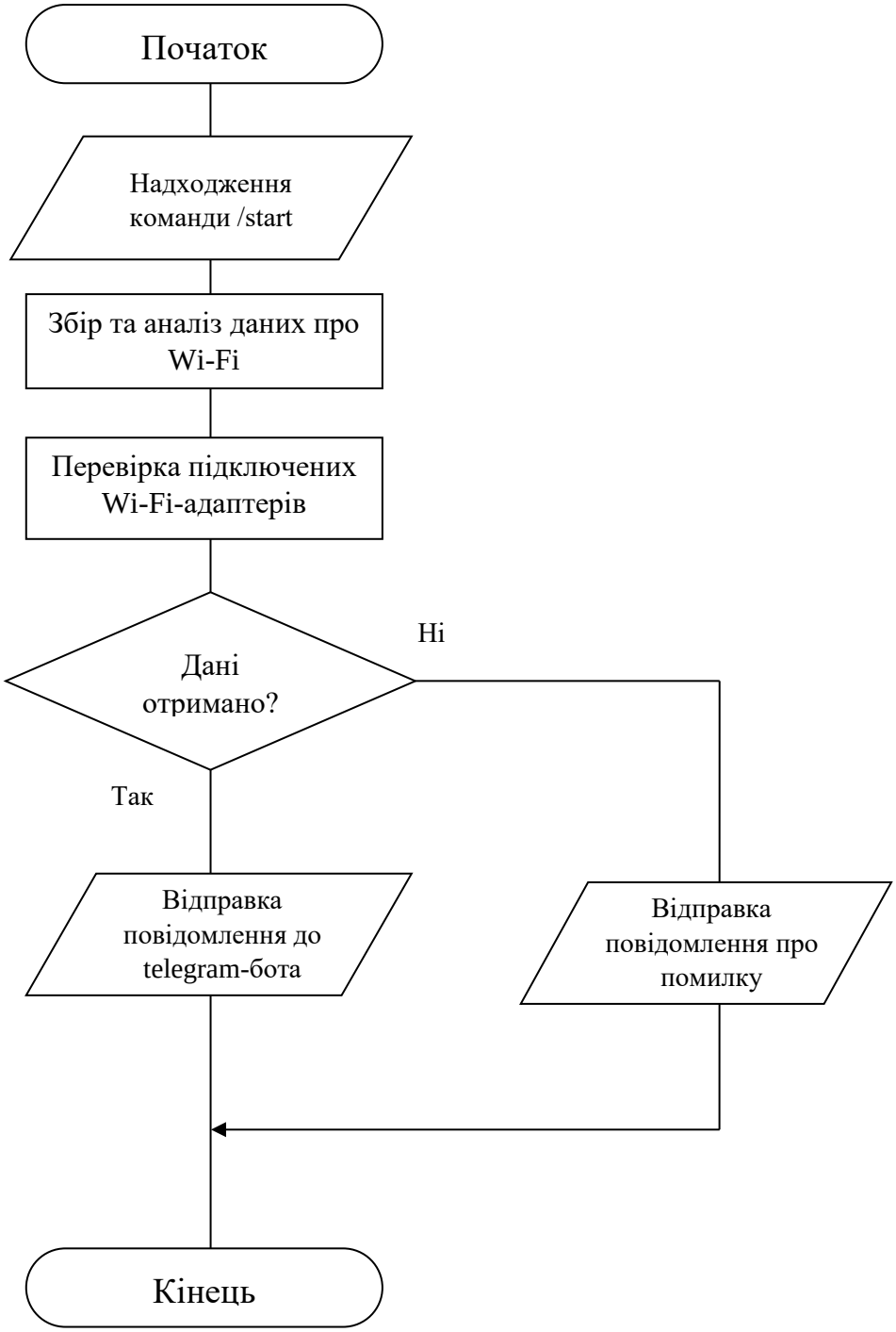


Рисунок 3.4 – Схема обробки команди /start

2. При отримані команди «/devices» від телеграм-бота, програма повинна виконувати аналіз та виявлення підключених пристроїв до мережі, генерування списку підключених пристрої, збереження цього списку для подальшого аналізу та

відправити дані до бота. На рисунку 3.5 наведено схему обробки команди «/devices».

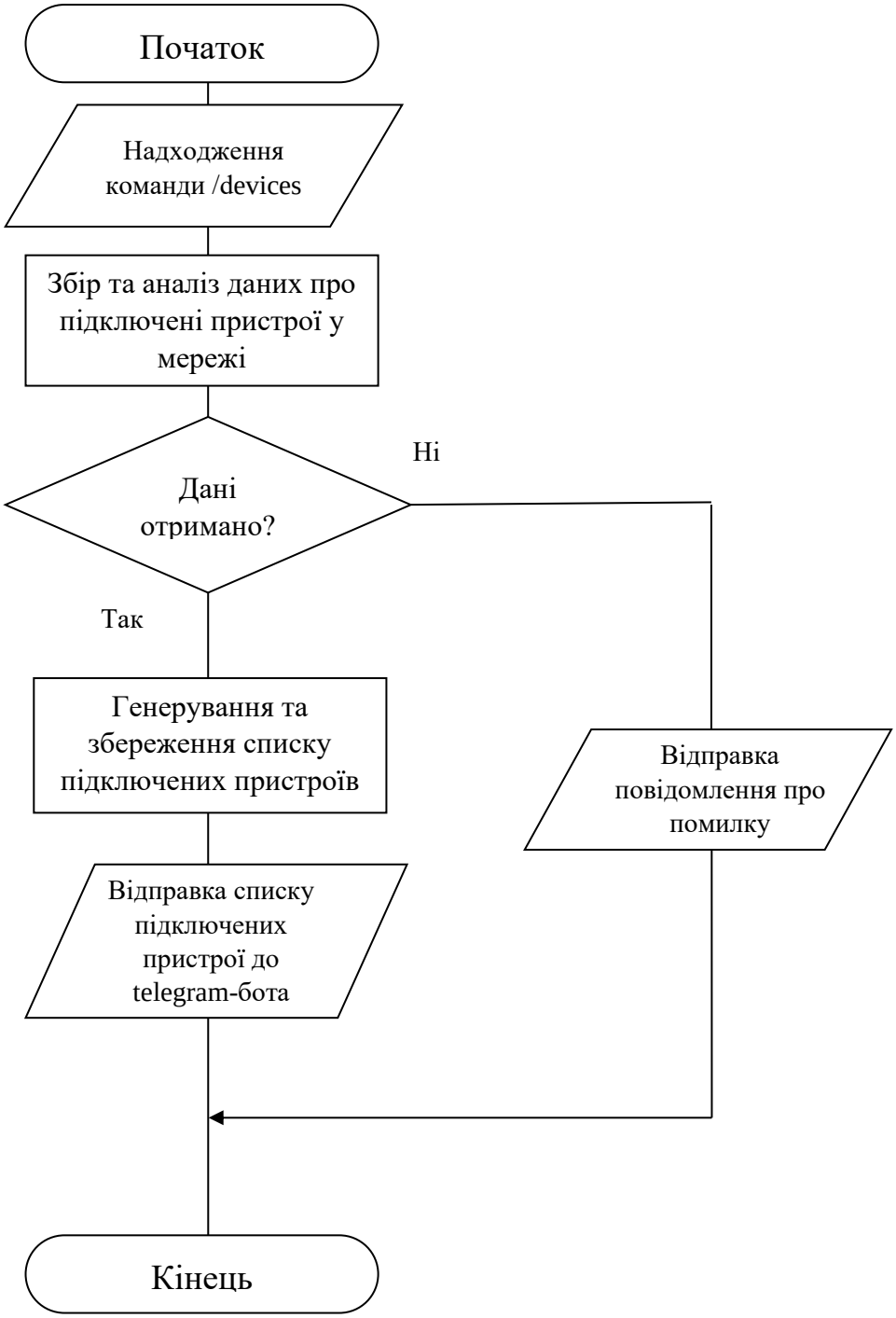


Рисунок 3.5 – Схема обробки команди «/devices»

3. При отримані команди «/dump» від телеграм-бота, програма повинна виконувати перехоплення трафіку, для збору даних та генерації статистики перехоплених пакетів. Потрібно, щоб аналіз виконувався певний час і потім

надсилав повідомлення з інформацією до бота. На рисунку 3.6 наведено схему обробки команди «/dump».

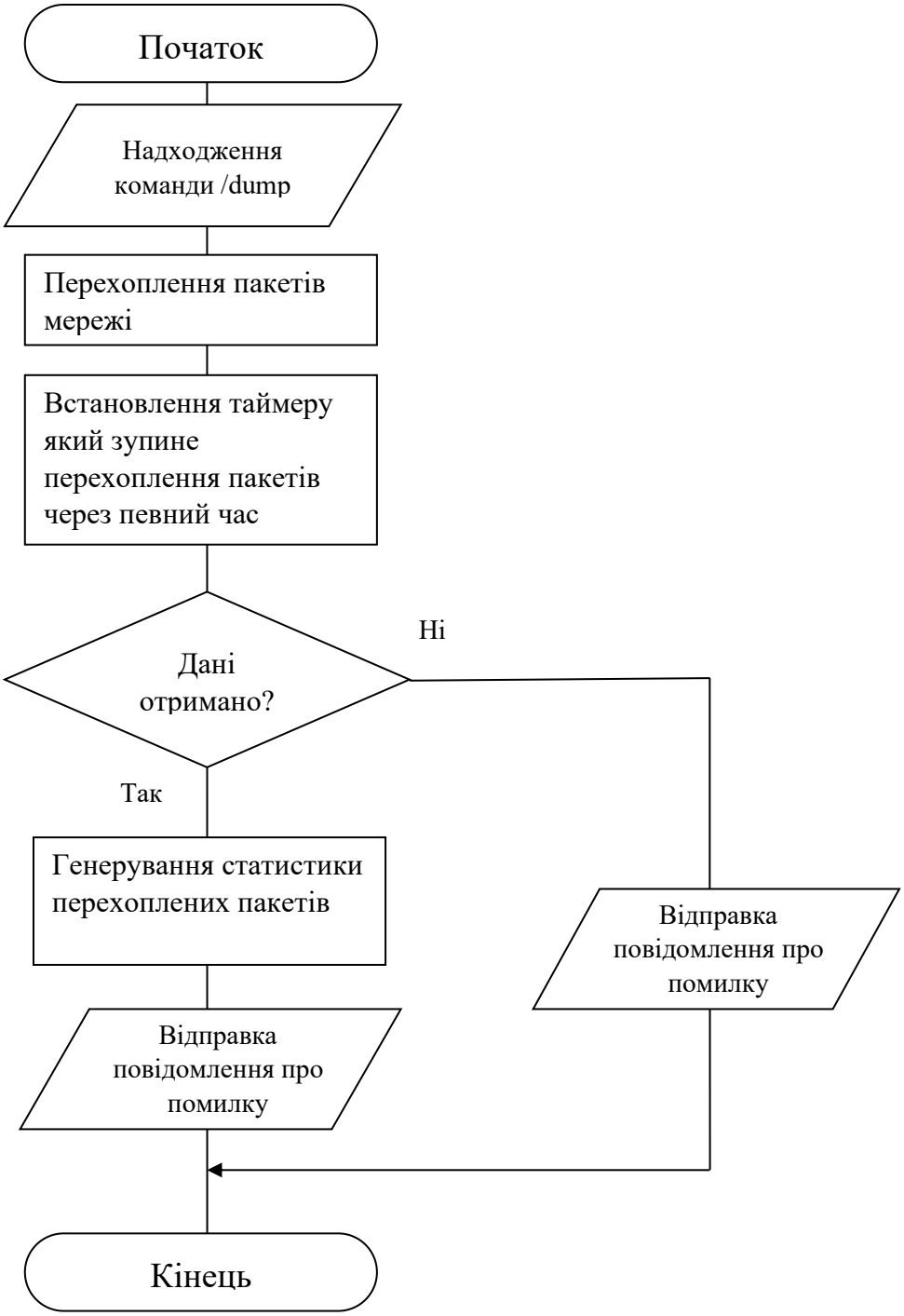


Рисунок 3.6 – Схема обробки команди «/dump»

4. При запуску програми на мікрокомп’ютері, повинно одразу виявлятися всі підключені пристрої до мережі, і надсилається повідомлення зі списком пристроїв. Програма також повинна постійно перевіряти на наявність нових пристроїв у

мережі та зберігати дані про них. На рисунку 3.7 наведено схему виявлення нових пристроїв які підключились до мережі.

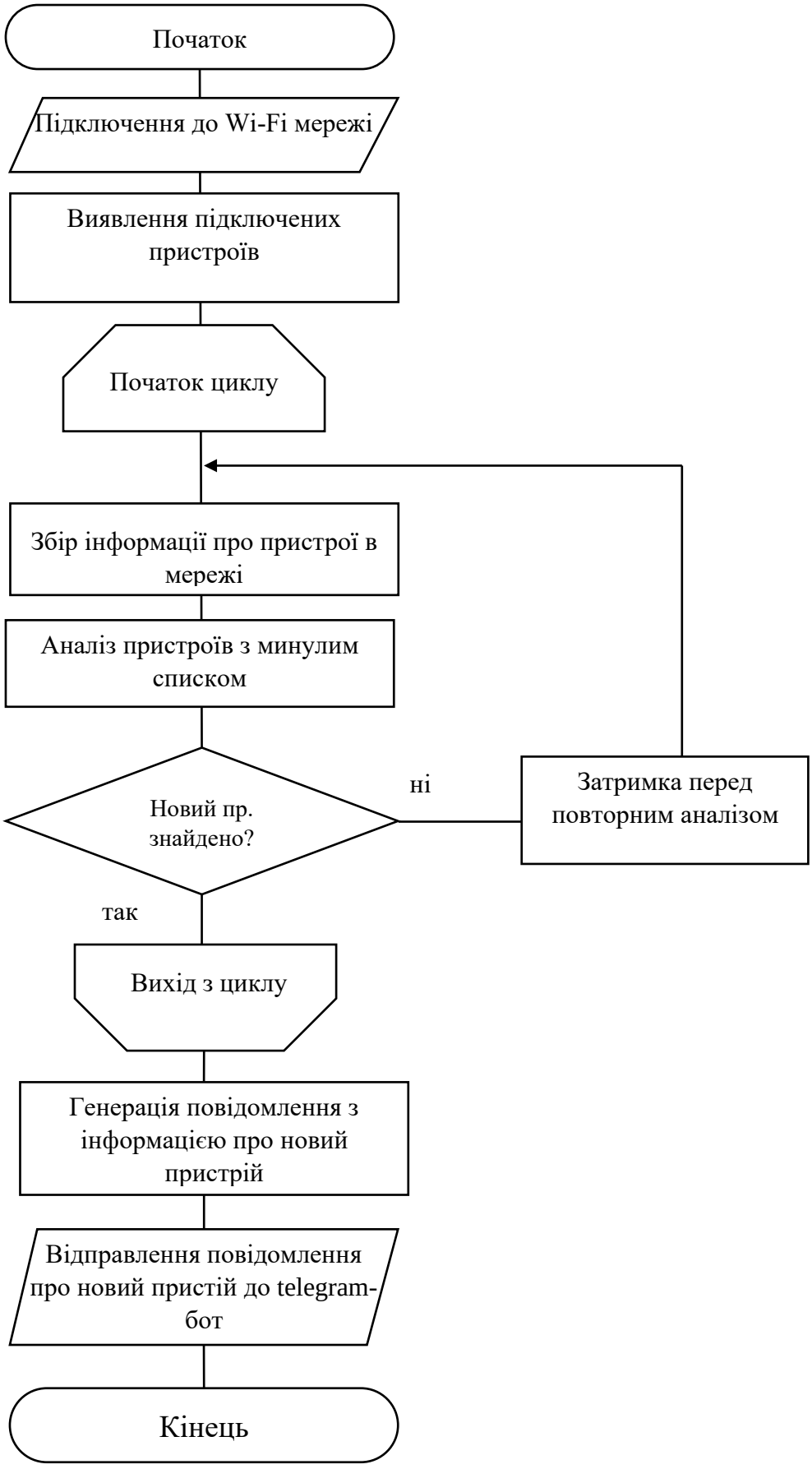


Рисунок 3.7 – Схема виявлення нового пристрою у мережі

5. Після збереження списку підключених пристроїв засіб починає аналізувати пакети які протікають по мережі для виявлення можливих атак. Засіб повинен постійно аналізувати нові підключені пристрої і перевіряти їх зі збереженим списком на схожість MAC-адрес та IP-адрес для виявлення можливої атаки ARP-spoofing Також програма повинна виявляти та надсилати повідомлення з попередженням про такі загрози як: ARP-spoofing та DOS-атаки на відмову системи. На рисунку 3.6 зображено схему виявлення ARP-spoofing. На рисунку 3.7 зображено схему виявлення DOS-атаки.

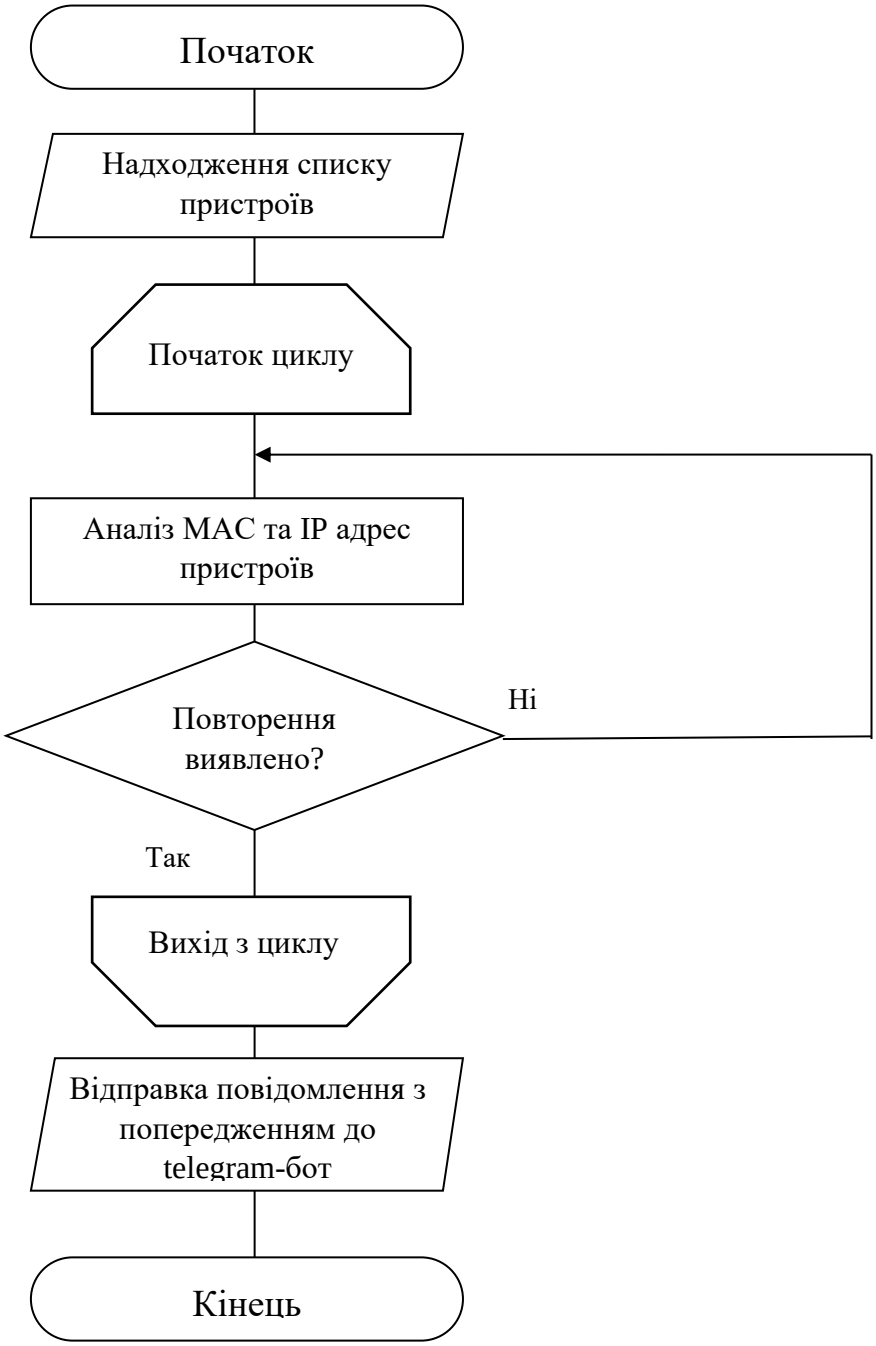


Рисунок 3.5 –Схема перевірки на ARP-spoofing

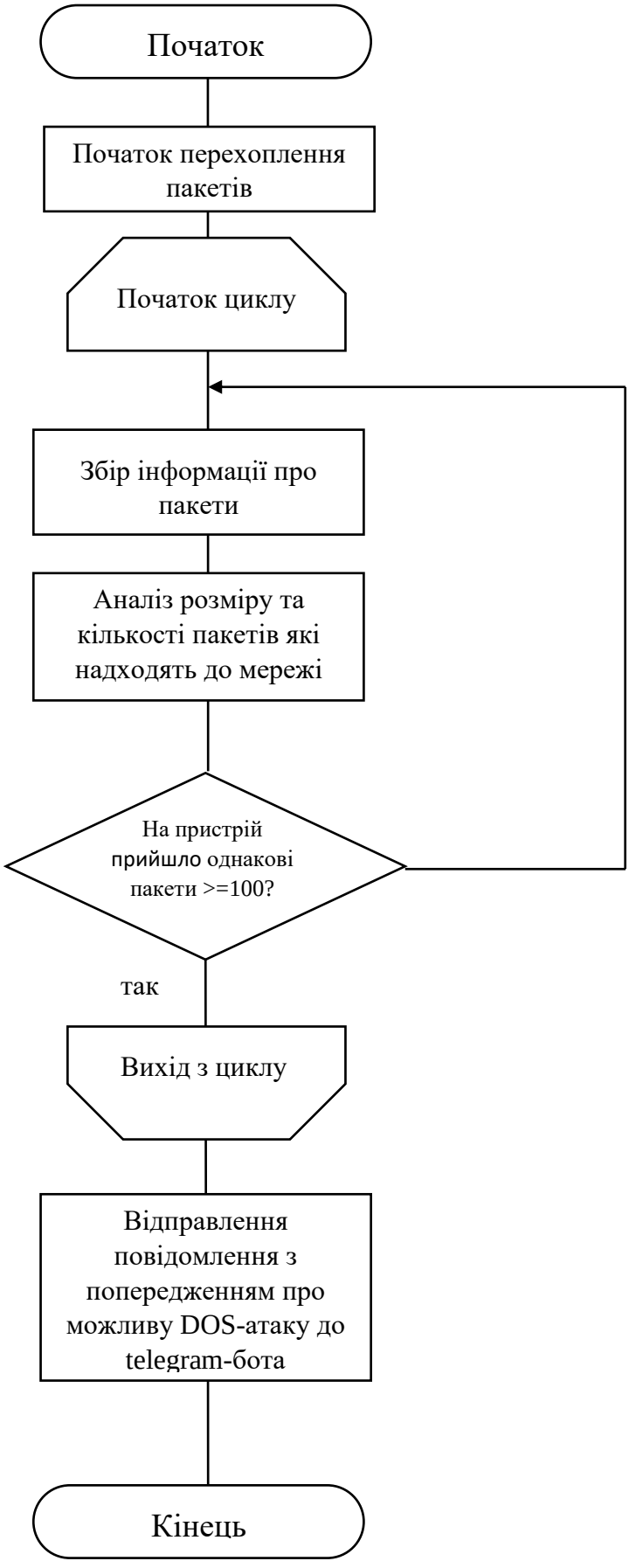


Рисунок 3.6 – Схема виявлення Dos-атаки

3.3 Програмна реалізація засобу моніторингу

Розглянемо основні частини програми:

Першим ділом при написані програми потрібно підключити певний набір бібліотек, які необхідні для подальшої реалізації функціоналу програми. При написані програми було імпортовано наступні бібліотеки:

— «subprocess» – використовується для взаємодії із зовнішніми команди;

— «os» – використовується для роботи з операційною системою мікрокомп’ютера;

— «re» – використовується для генерування регулярних виразів;

— «telebot» – використовується для роботи, взаємодії та з’єднання Telegram API щоб контролювати телеграм-бота;

— «threading» – використовується для виконання функцій у окремих потоках;

— «time» – використовується для роботи зі затримками програми та встановлення таймеру роботи певних функцій;

— «scapy.all» – використовується для аналізу, перехоплення та зчитування пакетів які надходять до мережі.

Для роботи з телеграм-ботом створюємо та отримуємо токен і chat_id для керування на взаємодії з телеграм ботом, вставляємо в bot_token = токен вашого телеграм-бота, та chat_id = ідентифікатор телеграм чату. Конфігурація телеграм-бота:

bot_token = «**********:**********»;

chat_id = «**********»

bot = telebot.TeleBot(bot_token) – створення бота.

Для виконання функції «get_wifi_info()» надсилаємо в телеграм боті команду / start, команда телеграм-бота «/start» звертається до функції «get_wifi_info()», а

функція надсилає інформацію про Wi-Fi мережу до телеграм-бота. Для отримання інформації про Wi-Fi мережу «get_wifi_info()» виконує команду «iwconfig». На рисунку 3.7 наведено код функції «get_wifi_info()».

```
# Функція для отримання даних про Wi-Fi мережу
def get_wifi_info():
    try:
        # Виконання команди для отримання інформації про Wi-Fi мережу
        output = subprocess.check_output(['iwconfig'])
        return output.decode('utf-8')
    except subprocess.CalledProcessError:
        return 'Не вдалося отримати інформацію про Wi-Fi мережу.'
```

Рисунок 3.7 – Код функції «get_wifi_info()»

Для виконання функції «get_connected_devices()» надсилаємо команду `/devices, команда звертається до функції «get_connected_devices()» а функція надсилає список підключених пристроїв. Для отримання списку підключених пристроїв функція «get_connected_devices()» виконує команду «arp-scan –localnet» для отримання списку підключених пристроїв та інформації про них.

На рисунку 3.8 наведено код функції «get_connected_devices()».

```
def get_connected_devices():
    try:
        output = subprocess.check_output(['arp-scan', '--localnet'])
        devices = output.decode('utf-8').split('\n')
        connected_devices = []
        for device in devices:
            device_info = device.strip().split('\t')
            if len(device_info) >= 2:
                ip_address = device_info[0]
                mac_address = device_info[1]
                device_name = device_info[2] if len(device_info) > 2 else 'Невідомо'
                connected_devices.append((ip_address, mac_address, device_name))

        devices_message = 'Список підключених пристроїв:\n\n'
        for device in connected_devices:
            ip_address, mac_address, device_name = device
            device_info = f'IP-адреса: {ip_address}\nMAC-адреса: {mac_address}\nНазва пристрою: {device_name}\n\n'
            devices_message += device_info

        return devices_message.strip()
    except subprocess.CalledProcessError:
        return 'Не вдалося отримати список підключених пристроїв.'
```

Рисунок 3.8 – Код функції «get_connected_devices()»

Для виконання аналізу та перехоплення пакетів мережі надсилаємо в телеграм-боті команду /dump, яка звертається до функції «packet_capture_thread()». Ця функція починає перехоплення пакетів за допомогою виконання команди «tcpdump» для перехоплення пакетів, аналіз пакетів відбувається протягом 1 хвилини. Потім функції «packet_capture_thread()» підраховує кількість пакетів та

адреси, та відправляє статистику до телеграм-боту. На рисунку 3.9 наведено код функції «packet_capture_thread()».

```
def packet_capture_thread():
    try:
        # Виконання команди для перехоплення пакетів з використанням tcpdump
        cmd = ['tcpdump']  # Замініть 'wlan0' на ваш інтерфейс Wi-Fi
        process = subprocess.Popen(cmd, stdout=subprocess.PIPE, stderr=subprocess.DEVNULL, universal_newlines=True)

        time.sleep(60)  # Затримка в 1 хвилину
        # Завершення процесу tcpdump
        process.terminate()
        # Отримання виводу з перехопленими пакетами
        output = process.stdout.read()
        # Підрахунок кількості пакетів та адрес
        input_packet_count = 0
        input_address_count = {}

        output_packet_count = 0
        output_address_count = {}
        for line in output.split('\n'):
            if line:
                if 'src' in line:
                    input_packet_count += 1
                    source_address = line.split()[2]
                    if source_address in input_address_count:
                        input_address_count[source_address] += 1
                    else:
                        input_adress_count[source_address] = 1
                elif 'dst' in line:
                    output_packet_count += 1
                    destination_address = line.split()[4]
                    if destination_address in output_address_count:
                        output_address_count[destination_address] += 1
                    else:
                        output_address_count[destination_address] = 1

                #if destination_address in address_count:
                    #address_count[destination_address] += 1
                #else:
                    #address_count[destination_address] = 1
        # Підготовка повідомлення зі статистикою
        stats_message = 'Статистика перехоплених пакетів:\n'
        stats_message += f'Загальна кількість пакетів input: {input_packet_count}\n'
        stats_message += f'Загальна кількість пакетів output: {output_packet_count}\n'
        stats_message += 'пакетів:\n'
        for address, count in input_address_count.items():
            stats_message += f'{address}: {count}/n'
        stats_message += 'address:\n'
        for address, count in output_address_count.items():
            stats_message += f'{address}: {count}\n'
        #if input_address_count:
            # = max(input_address_count, key=input_address_count.get)
            #stats_message += f'Найбільше input запитів до адреси: {most_common_input_address}\n'
        #if output_address_count:
            #most_common_output_address = max(output_address_count, key=output_address_count.get)
            #stats_message += f'Найбільше output запитів до адреси: {most_common_output_address}\n'
        # Відправка повідомлення зі статистикою
        bot.send_message(chat_id, stats_message)
    except Exception as e:
        print(f'Error in packet_capture_thread: {str(e)}')
```

Рисунок 3.9 – Код функції «packet_capture_thread()»

Функція «process_packet(packet)» є пасивною і виконується одразу після запуску програми, обробляє отримані пакети, підраховує кількість запитів до кожної адреси та зберігає пристрої, які з'явилися. На рисунку 3.10 наведено код функції «process_packet(packet)».

```
def process_packet(packet):
    global packet_count
    global address_count

    # Обробка отриманого пакета
    packet_count += 1

    # Отримання адреси, до якої здійснено запит
    destination_address = packet[ARP].pdst

    # Підрахунок кількості запитів до кожної адреси
    if destination_address in address_count:
        address_count[destination_address] += 1
    else:
        address_count[destination_address] = 1

    # Збереження пристроїв, що з'являються
    if packet[ARP].psrc not in [device[1] for device in device_list]:
        #device_name = get_device_name(packet[ARP].psrc)
        device_list.append((device_name, packet[ARP].psrc, packet[ARP].hwsrc))
```

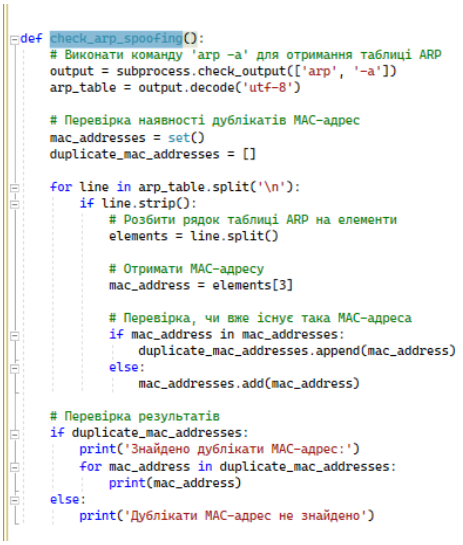
Рисунок 3.10 – Код функції «process_packet(packet)»

Функція «check_new_devices()» є пасивною і виконується одразу після запуску програми. Функція періодично виконує команду «arp-scan –localnet» для виявлення нових підключених пристроїв та у разі виявлення нових пристроїв надсилає повідомлення із інформацією про них. На рисунку 3.11 наведено код функції «check_new_devices()».

```
def check_new_devices():
    global device_list

    while True:
        output = subprocess.check_output(['arp-scan', '--localnet'])
        output_lines = output.decode('utf-8').split('\n')

        new_devices = []
        for line in output_lines:
            if line:
                elements = line.split('\t')
                if len(elements) >= 3:
                    ip_address = elements[0]
                    mac_address = elements[1]
                    device_name = elements[2]
                    device = (ip_address, mac_address, device_name)
                    if device not in device_list:
                        device_list.append(device)
                        new_devices.append(device)
        if new_devices:
            message = "New PC:\n"
            for device in new_devices:
                message += f"new-devices:\nName:{device[2]}\nIP adre:{device[0]}\nMAC adres:{device[1]}\n\n"
                bot.send_message(chat_id, message)
        time.sleep(60)

threading.Thread(target=check_new_devices).start()
```

Рисунок 3.11 – Код функції «check_new_devices()»

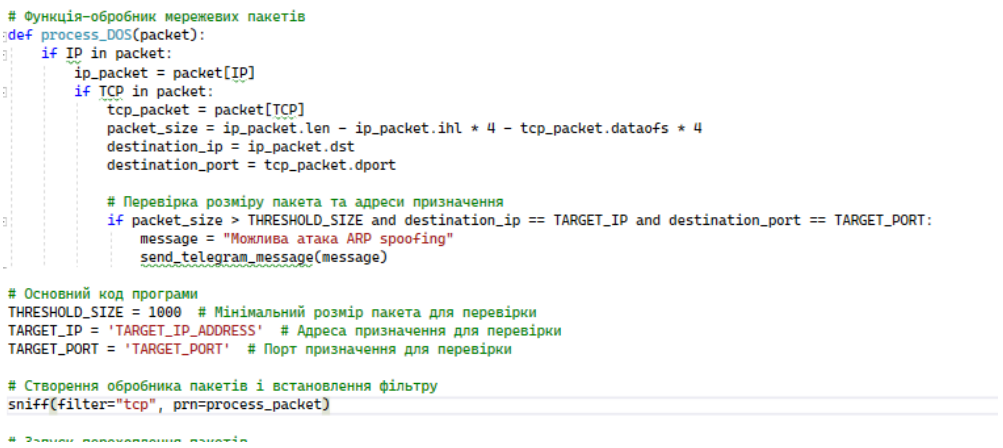
Функція «check_arp_spoofing()» запускається після отримання списку підключених пристроїв від функції «get_connected_devices()» та постійно перевіряє наявність ARP-spoofing шляхом перевірки дублікатних MAC-адрес у списку пристроїв. На рисунку 3.12 наведено код функції «check_arp_spoofing()».



```
def check_arp_spoofing():
    # Виконати команду 'arp -a' для отримання таблиці ARP
    output = subprocess.check_output(['arp', '-a'])
    arp_table = output.decode('utf-8')

    # Перевірка наявності дублікатів MAC-адрес
    mac_addresses = set()
    duplicate_mac_addresses = []

    for line in arp_table.split('\n'):
        if line.strip():
            # Розбити рядок таблиці ARP на елементи
            elements = line.split()

            # Отримати MAC-адресу
            mac_address = elements[3]

            # Перевірка, чи вже існує така MAC-адреса
            if mac_address in mac_addresses:
                duplicate_mac_addresses.append(mac_address)
            else:
                mac_addresses.add(mac_address)

    # Перевірка результатів
    if duplicate_mac_addresses:
        print('Знайдено дублікати MAC-адрес:')
        for mac_address in duplicate_mac_addresses:
            print(mac_address)
    else:
        print('Дублікати MAC-адрес не знайдено')
```

Рисунок 3.12 – Код функції «check_arp_spoofing()».

Функція «process_DOS()» запускається після запуску програми і працює постійно перевіряючи та аналізуючи перехоплені пакети. Після заппуску програми функція починає перехоплення пакетів та аналізує їх розмір, кількість та до якої адреси були надіслані. У разі виявлення великого надходження однакової кількості пакетів до одного і того ж ресурсу або пристрою надсилає повідомлення до телеграм-бота при виявлення можливої DOS-аткаки. На рисунку 3.13 забражено код функції «process_DOS()».



```
# Функція-обробник мережевих пакетів
def process_DOS(packet):
    if IP in packet:
        ip_packet = packet[IP]
        if TCP in packet:
            tcp_packet = packet[TCP]
            packet_size = ip_packet.len - ip_packet.ihl * 4 - tcp_packet.dataofs * 4
            destination_ip = ip_packet.dst
            destination_port = tcp_packet.dport

            # Перевірка розміру пакета та адреси призначення
            if packet_size > THRESHOLD_SIZE and destination_ip == TARGET_IP and destination_port == TARGET_PORT:
                message = "Можлива атака ARP spoofing"
                send_telegram_message(message)

# Основний код програми
THRESHOLD_SIZE = 1000  # Мінімальний розмір пакета для перевірки
TARGET_IP = 'TARGET_IP_ADDRESS'  # Адреса призначення для перевірки
TARGET_PORT = 'TARGET_PORT'  # Порт призначення для перевірки

# Створення обробника пакетів і встановлення фільтру
sniff(filter="tcp", prn=process_packet)
```

Рисунок 3.13 – Код функції «process_DOS()»

Функція «bot.polling()» запускає телеграм-бота для взаємодії програми та телеграм-бота, а також для отримання та обробки повідомлень.

Після запуску програми вона буде виконувати моніторинг мережі, перехоплювати пакети, а також надсилати повідомлення про інформацію про Wi-

Fi мережу, список підключених пристроїв та статистику перехоплених пакетів у вказаний телеграм-чат. Вона також буде періодично перевіряти нові підключені пристрої та виявляти можливі атаки.

3.4 Тестування роботи засобу

Щоб переконатися що засіб для моніторингу бездротової мережі на основі мікрокомп’ютера працює належним чином проведу довго тривале тестування.

Першим ділом підключаю до живлення і переконуюсь що мікрокомп’ютер завантажується та працює належним чином, а також перевіряю чи на екрані з’являється вікно операційної системи. Запускаю Raspberry Pi, підключаю Wi-Fi адаптер з підтримкою моніторингу, та підключаюсь до мережі Wi-Fi.

Першим ділом перевіряється, чи засіб для моніторингу з’єднався з мережею Wi-Fi, та чи має доступ до інтернету. Далі запускається програма для взаємодії телеграм-ботом та моніторингу Wi-Fi мережі. Перевіряється чи правильно виконується програма та чи не виникає помилок при запуску (рисунок 3.14).

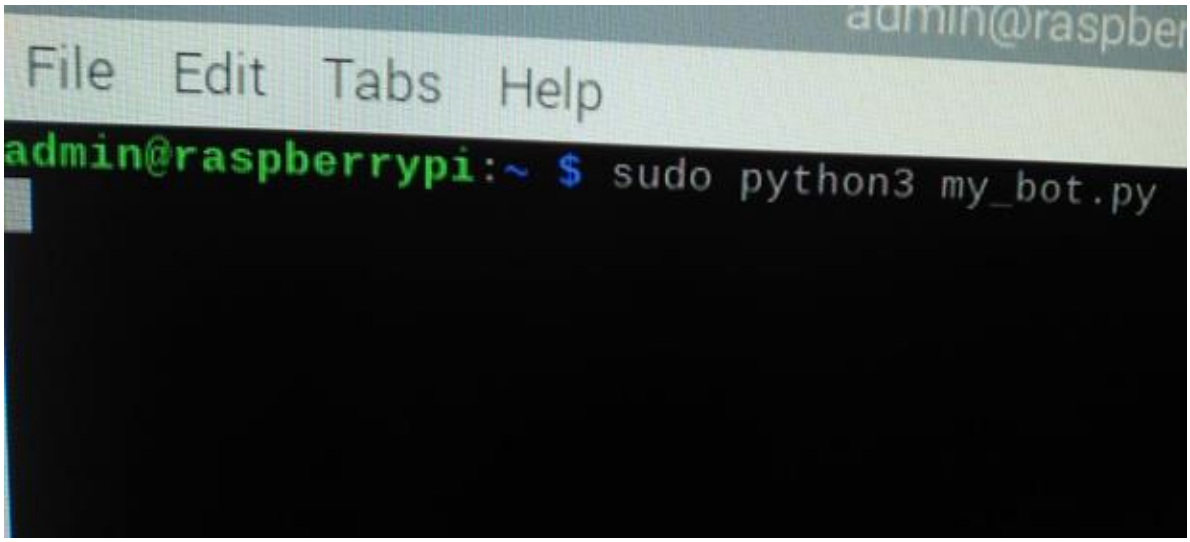


Рисунок 3.14 – запуск засобу моніторингу Wi-Fi мережі

При запуску програми помилок не було виявлено, тому далі перевіряється керування засобом через telegram-бот. Перевіряється, чи засіб реагує на команди

від телеграм бота. У telegram-боті надсилається команда /start та /devices для перевірки (рисунок 3.15, 3.16).

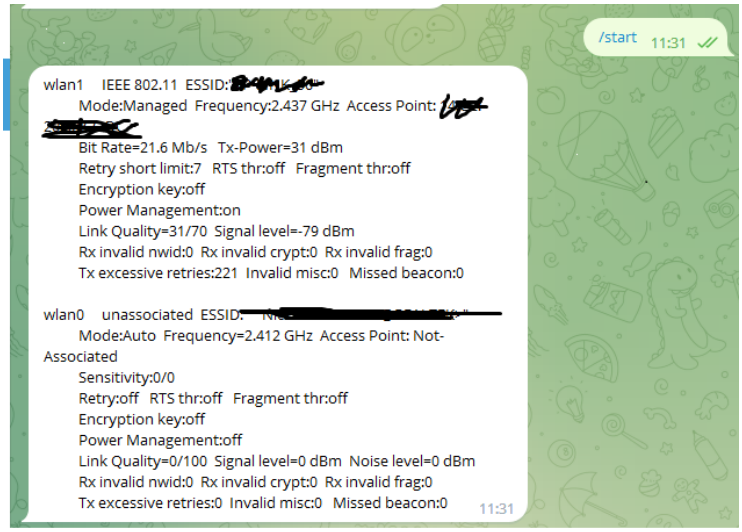


Рисунок 3.15 – Результат виконання команди /start

Отримується інформаці про наявність двох бездротових інтерфейсів: wlan1 та wlan0.

1. У інтерфейсі wlan1 ми бачемо таку інформацію:

ESSID: "**********" - це ідентифікатор бездротової мережі (SSID) до якої підключений адаптер.

Режим: Managed - це означає, що адаптер працює в режимі керованого з'єднання, де доступ до мережі здійснюється через точку доступу (Access Point).

Частота: 2.437 GHz - це частота, на якій працює бездротове з'єднання.

Access Point: **********- це MAC-адреса точки доступу, до якої адаптер підключений.

Швидкість передачі даних (Bit Rate): 21.6 Mb/s - це швидкість передачі даних, яку досягає адаптер.

Tx-Power: 31 dBm - це потужність передавача адаптера.

Link Quality: 31/70 - це показник якості зв'язку між адаптером та точкою доступу. Значення 31/70 вказує на помірну якість зв'язку.

Signal level: -79 dBm - це рівень сигналу, отриманий адаптером від точки доступу. Значення -79 dBm вказує на слабкий сигнал.

Power Management: on - це означає, що функція керування живленням увімкнена.

Ці дані показують, що адаптер wlan1, а саме TL-WN722N, підключений до бездротової мережі з ідентифікатором "TP-LINK_56" Адаптер wlan0 не підключений до жодної мережі.

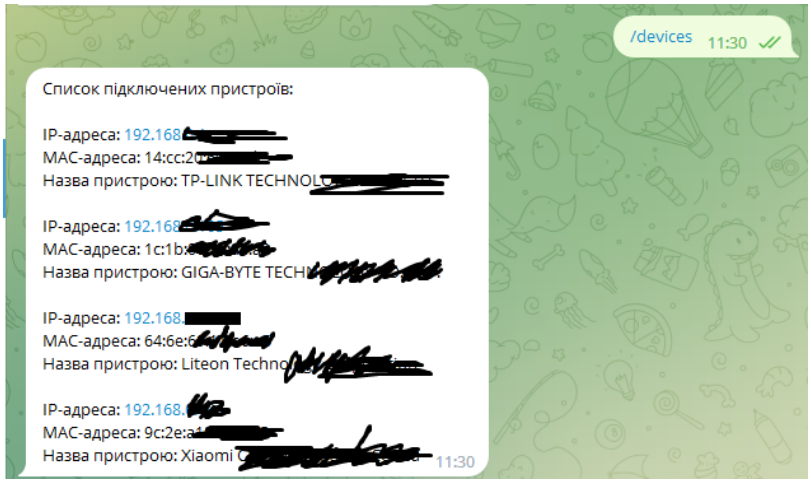


Рисунок 3.16 – Результат виконання команди /devices

Переконавшись, що програма правильно генерує та виводить інформацію про Wi-Fi мережу та підключені пристрої. Перевіряється розміщення розділювачів між різними пристроями та правильність виводу IP-адрес та MAC-адрес. Дані про пристрої вірні, повідомлення було надіслано правильно. У списку бачимо інформацію про пристрої таку як IP- адреса, MAC-адреса та назва пристрою

Далі перевіряється перехоплення та аналіз пакетів, для цього надсилається команда /dump у telegram-боті та перевіряється чи надходить повідомлення про початок аналізу. Очікується одна хвилина та перевіряється чи прийшло повідомлення зі статистикою переломлених пакетів. Перевіряється чи засіб правильно відображає статистику перевірки за одну хвилину (рисунок 3.17).

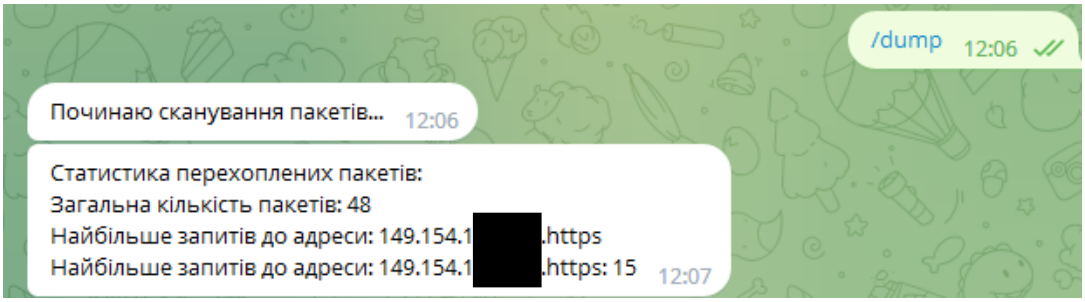


Рисунок 3.17 –Результат виконання команди /dump

Після запуску програми запускається перевірка но можливі атаки на мережу типу ARP-spoofing та DDOS-атака. При першому запуску програми засіб аналізує пристрої на ARP-spoofing та, якщо заміни не було виявлено повідомить про це телеграм бот (рис. 3.18)

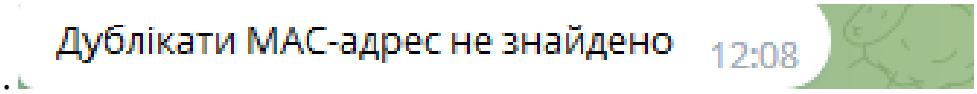


Рисунок 3.18 – Перевірка на ARP-spoofing після запуску засобу

Для перевірки чи засіб моніторингу виявляє можливі атаки на мережу потрібно виконати ці атаки на мережу. Виконавши ARP-spoofing атаку, перевіряю чи засіб виявить заміну MAC-адреси та чи надсилає повідомлення до телеграм-бота про виявлену атаку на мережу (рисунок 3.19).

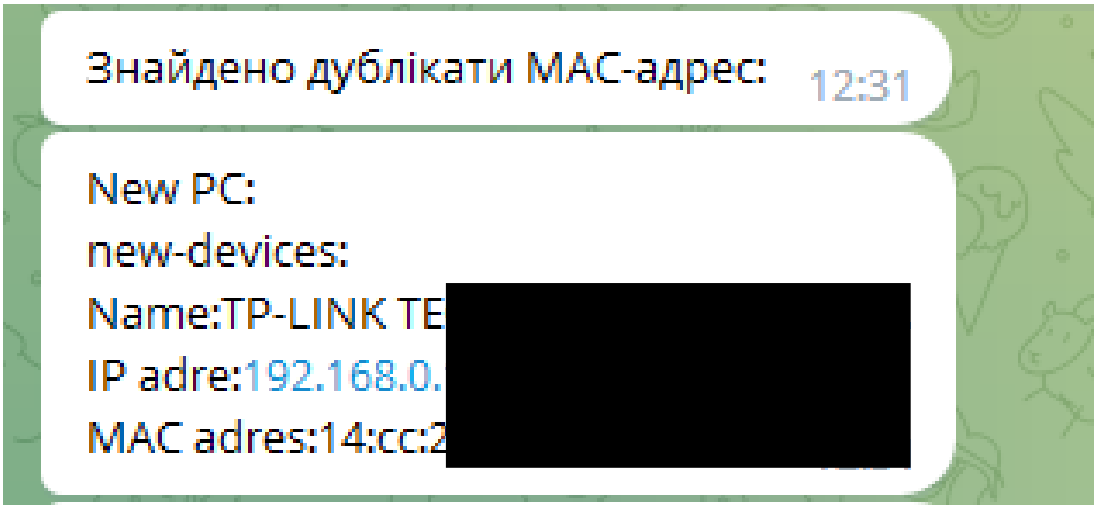


Рисунок 3.19 – Повідомлення про виявлену атаку ARP-spoofing

Для перевірки засобу моніторингу на виявлення DDOS-атаки необхідно виконати симуляцію даної атаки на мережу. Після виконання DDOS-атаки на мережу перевіряємо чи надсилається до telegram-бота повідомлення про виявлення даної атаки (рисунок 3.20).

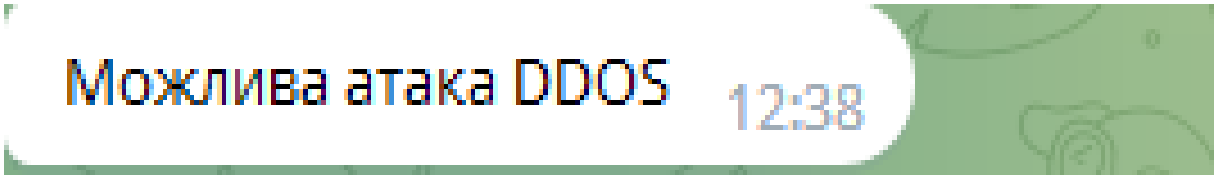


Рисунок 3.21 – Повідомлення про можливу DDOS-атаку на пристрій

Продовжуючи тривале тестування програми, переконуємось, що вона працює стабільно протягом тривалого часу та без відмов. Отже розроблений засіб моніторингу бездротової мережі на основі мікрокомп’ютера працює належним чином та без відмов.

ВИСНОВКИ

У цій роботі ми досліджували основні аспекти розробки таких засобів, зокрема збір і аналіз інформації про доступні мережі, підключені пристрої, рівень сигналу та швидкість передачі даних. Ми розглянули алгоритми та схеми функціонування для отримання цієї інформації і реалізували функції, які дозволяють виявляти нові пристрої та перевіряти наявність ARP-spoofing, DDOS-атак .

Засоби моніторингу бездротової мережі на мікрокомп'ютері Raspberry Pi забезпечують гнучкість і розширювані можливості. Завдяки можливості взаємодії з бездротовим інтерфейсом мікрокомп'ютера та використанню мови програмування Python, ми можемо легко налаштувати засіб моніторингу під свої потреби та додати додаткові функціональні можливості.

У процесі розробки таких засобів ми зазначили, що вони можуть бути застосовані в різних сферах, включаючи домашні мережі, бізнес-середовища, громадські місця тощо. Вони дозволяють забезпечити безпеку мережі, виявити нові пристрої, виявити можливі проблеми та забезпечити ефективне використання ресурсів бездротової інфраструктури.

Отже, засоби моніторингу бездротової мережі на основі мікрокомп'ютера забезпечують надійну інструментарій для контролю і управління бездротовими мережами. Вони відкривають нові можливості для покращення безпеки мережі, виявлення проблем та ефективного використання ресурсів. За допомогою таких засобів можна стежити за станом мережі в режимі реального часу і приймати вчасні заходи для запобігання можливим проблемам.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мережеві архітектури: веб-сайт. URL: https://sites.google.com/site/kmposibnyk/lekcii/lekcia-5
2. Архітектура адаптивної бездротової мережі: URL: http://zbirniksgm.kntu.kr.ua/pdf/50/28.pdf
3. Бездротові мережі: URL: https://ua5.org/lan/143-bezdrotov-merezh.html
4. Що таке бездротова мережа, типи та характеристики. URL:https://seguidores.online/uk/red-inalambrica/
5. WiMAX. Как это работает: веб-сайт. URL: https://habr.com/ua/articles/78743/
6. Що таке бездротова технологія BLUETOOTH?. URL: https://helpguide.sony.net/mdr/xb70bt/v1/uk/contents/TP0001177364.html
7. Технологія ZigBee. URL: https://salus.ua/2021/11/21/protokol-zigbee/
8. Моніторинг мережі. Протоколи, інструменти 2021. URL: https://eska.global/blog/setevoj-monitoring-protokoly-luchshie-praktiki-instrumenty-2020
9. Моніторинг та аналіз комп’ютерних мереж. URL: https://studfile.net/preview/9239701/
10. Моніторинг мережі. Протоколи, найкращі практики, інструменти 2021. URL: https://eska.global/blog/setevoj-monitoring-protokoly-luchshie-praktiki-instrumenty-2020
11. Презентація "Засоби моніторингу мережевого трафіку. URL: https://naurok.com.ua/prezentaciya-zasobi-monitoringu-merezhevogo-trafiku-190381.html
12. БЛОГ про все цікаве у світі телекомунікацій. URL: http://www.telesphera.net/blog/zabezpechennya-merezhevoi-bezpeki.html
13. Засоби моніторингу та аналізу мережі. URL: https://wiki.cuspu.edu.ua/index.php/ (дата звернення: 24.05.2023).
14. Програми для моніторингу локальних мереж і мереж вай фай. Що таке моніторинг. URL: https://texnogid.biz.ua/wi-fi/bezpeka/prohramy-dlia-monitorynhu-merezh.html

15. 16 найкращих безкоштовних програм для моніторингу мережі для Windows10. URL: https://techukraine.net/
16. Програми для моніторингу локальних мереж і мереж вай фай. URL: https://texnogid.biz.ua/wi-fi/bezpeka/prohramy-dlia-monitorynhu-merezh.html
17. Контролер бездротової мережі. URL: https://deps.ua/ua/knowegable-base/samples-of-the-technical-solutions/8009.htm
18. Мережеві аналізатори. URL: https://studfile.net/preview/7867665/page:4/
19. Що таке маршрутизатор?. URL: https://ntools.com.ua/uk/information/ faq/chto-takoe-marshrutizator-router
20. Моніторинг за допомогою комутаторів. URL: https://studfile.net/preview/9239701/page:13/
21. Базове мережеве обладнання: види і характеристики комутаторів. URL: https://secur.ua/articles/ua_bazove-merezheve-obladnannja-vidi-i-harakteristiki-komutatoriv.html
22. WiFi Analyzer. URL: https://play.google.com/store/apps/details?id=abdelrahman.wifianalyzerpro&hl
23. Fing - Network Tools. URL: https://play.google.com/store/apps/details?id=com.overlook.android.fing&hl=uk&gl=US
24. INET. URL: https://play.google.com/store/apps/details?id=com.inditex.inetmob&hl=uk&gl=US
25. Мікрокомп'ютери. URL: https://evo.net.ua/mikrokomputery/1088/
26. Мікрокомп’ютери, їх можливості та додаткові пристрої для них. URL: https://naurok.com.ua/mikrokomp-yuteri-h-mozhlivosti-ta-dodatkovi-pristro-dlya-nih-310786.html
27. 25 проектов на Raspberry Pi. URL: https://evo.net.ua/uk/25-proektov-na-raspberry-pi-4/
28. Wi-Fi Pineapple. URL: https://www.techtarget.com/searchsecurity/definition/Wi-Fi-Pineapple
29. Чим небезпечні безкоштовні Wi-Fi мережі. URL: https://news.finance.ua/ua/news/-/345683/chym-nebezpechni-bezkoshtovni-wi-fi-merezhi

30. Мікрокомп'ютер Raspberry Pi Zero W. URL: https://evo.net.ua/raspberry-pi-zero-w/
31. Мікрокомп'ютер Raspberry Pi 1. URL: https://evo.net.ua/raspberry-pi-1-model-b/
32. Raspberry Pi 2 Model B. URL: https://evo.net.ua/ru/raspberry-pi-2-model-b-v1.2/
33. Raspberry Pi 3 Model A+. URL: https://evo.net.ua/raspberry-pi-3-model-a/
34. Raspberry Pi 4 Model B. URL: https://evo.net.ua/mikrokomputer-raspberry-pi-4-model-b-8gb/
35. Мережевий адаптер для комп’ютера і ноутбука, підключення і настройка, поради з усунення неполадок. URL: http://radka.in.ua/poradi/merejevii-adapter-dlia-komputera-i-no.html
36. Wi-Fi- адаптер TL-WN722N. URL: https://www.tp-link.com/uk-ua/home-networking/high-gain-adapter/tl-wn722n/
37. Python™. URL: https://www.python.org/.
38. What is PyQt? URL: https://www.riverbankcomputing.com/software/pyqt/
39. Про основні засади забезпечення кібербезпеки України : Закон України від 1.12.2021 р. № 2163-VIII. URL: https://zakon.rada.gov.ua/laws/show/2163
40. Інформаційна та кібербезпека: соціотехнічний аспект : підручник / В.Л. Бурячок, В.Б. Толубко, В.О. Хорошко, С.В. Толюпа. Київ : ДУТ, 2015. 288 с.
41. Поняття безпеки бездротових мереж. URL: https://studfile.net/preview/9649992/
42. 25.Буров, Євген Вікторович. Комп'ютерні мережі / Є. Буров; За ред. В. Пасічника. - 2-е вид, оновлене і доп. - Львів: БаК, 2003. — 584 с.: іл. — ISBN 966-7065-41-3

ДОДАТКИ

Додаток А

ПРОТОКОЛ ПЕРЕВІРКИ БАКАЛАВРСЬКОЇ ДИПЛОМНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Засіб моніторингу бездротової мережі на базі мікрокомп'ютера

Автор роботи: Согур Антоній Андрійович

Тип роботи: бакалаврська дипломна робота .

Підрозділ кафедра захисту інформації ФІТКІ .

Показники звіту подібності Unicheck

Оригінальність – 91,6%. Схожість – 8,4%.

Аналіз звіту подібності (відмітити потрібне):

√ 1. Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

□ 2. Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.

□ 3. Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Каплун В. А.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ _____
(підпис) (прізвище, ініціали)

Керівник роботи _____ _____
(підпис) (прізвище, ініціали)

Додаток Б
Текст програми

```
import subprocess
import os
import re
import telebot
import threading
import time
from scapy.all import sniff, ARP

# Конфігурація телеграм-бота
bot_token = токен телеграм бота
chat_id = ваш chat_id телеграма

# Ініціалізація телеграм-бота
bot = telebot.TeleBot(bot_token)

packet_count = 0
address_count = {}
device_list = []

# Функція для отримання даних про Wi-Fi мережу
def get_wifi_info():
    try:
        # Виконання команди для отримання інформації про Wi-Fi мережу
        output = subprocess.check_output(['iwconfig'])
        return output.decode('utf-8')
    except subprocess.CalledProcessError:
        return 'Не вдалося отримати інформацію про Wi-Fi мережу.'


def get_connected_devices():
    try:
        output = subprocess.check_output(['arp-scan', '--localnet'])
        devices = output.decode('utf-8').split('\n')
        connected_devices = []
        for device in devices:
            device_info = device.strip().split('\t')
            if len(device_info) >= 2:
                ip_address = device_info[0]
                mac_address = device_info[1]
                device_name = device_info[2] if len(device_info) > 2 else 'Невідомо'
                connected_devices.append((ip_address, mac_address, device_name))

        devices_message = 'Список підключених пристроїв:\n\n'
        for device in connected_devices:
            ip_address, mac_address, device_name = device
            device_info  =  f'IP-адреса:  {ip_address}\nMAC-адреса:  {mac_address}\nНазва
пристрою: {device_name}\n\n'
            devices_message += device_info

        return devices_message.strip()
    except subprocess.CalledProcessError:
        return 'Не вдалося отримати список підключених пристроїв.'

def packet_capture_thread():
      try:
            # Виконання команди для перехоплення пакетів з використанням tcpdump
            cmd = ['tcpdump']  # Замініть 'wlan0' на ваш інтерфейс Wi-Fi
```

```
            process = subprocess.Popen(cmd, stdout=subprocess.PIPE,
stderr=subprocess.DEVNULL, universal_newlines=True)

            time.sleep(60)  # Затримка в 1 хвилину
            # Завершення процесу tcpdump
            process.terminate()
            # Отримання виводу з перехопленими пакетами
            output = process.stdout.read()
            # Підрахунок кількості пакетів та адрес
            input_packet_count = 0
            input_address_count = {}

            output_packet_count = 0
            output_address_count = {}
            for line in output.split('\n'):
                  if line:
                        if 'src' in line:
                              input_packet_count += 1
                              source_address = line.split()[2]
                              if source_address in input_address_count:
                                    input_address_count[source_address] += 1
                              else:
                                    input_adress_count[source_address] = 1
                        elif 'dst' in line:
                              output_packet_count += 1
                              destination_address = line.split()[4]
                              if destination_address in output_address_count:
                                    output_address_count[destination_address] += 1
                              else:
                                    output_address_count[destination_address] = 1

                        #if destination_address in address_count:
                              #address_count[destination_address] += 1
                        #else:
                              #address_count[destination_address] = 1
            # Підготовка повідомлення зі статистикою
            stats_message = 'Статистика перехоплених пакетів:\n'
            stats_message += f'Загальна кількість пакетів input: {input_packet_count}\n'
            stats_message += f'Загальна кількість пакетів output: {output_packet_count}\n'
            stats_message += 'пакетів:\n'
            for address, count in input_address_count.items():
                  stats_message += f'{address}: {count}/n'
            stats_message += 'address:\n'
            for address, count in output_address_count.items():
                  stats_message += f'{address}: {count}\n'
            #if input_address_count:
                  # = max(input_address_count, key=input_address_count.get)
                  #stats_message += f'Найбільше input запитів до адреси:
{most_common_input_address}\n'
            #if output_address_count:
                  #most_common_output_address = max(output_address_count,
key=output_address_count.get)
                  #stats_message += f'Найбільше output запитів до адреси:
{most_common_output_address}\n'
            # Відправка повідомлення зі статистикою
            bot.send_message(chat_id, stats_message)
      except Exception as e:
            print(f'Error in packet_capture_thread: {str(e)}')

# Обробка команди /start від користувача
@bot.message_handler(commands=['start'])
def handle_start(message):
    # Отримання даних про Wi-Fi мережу
```

```
    wifi_info = get_wifi_info()

    # Надсилання даних в телеграм
    bot.send_message(chat_id, wifi_info)


# Обробка команди /devices від користувача
@bot.message_handler(commands=['devices'])
def handle_devices(message):
    # Отримання списку підключених пристроїв
    connected_devices = get_connected_devices()

    # Формування повідомлення про підключені пристрої


    # Надсилання даних в телеграм
    bot.send_message(chat_id, connected_devices)


@bot.message_handler(commands=['dump'])
def start_packet_dump(message):
    global packet_count
    global address_count

    packet_count = 0
    address_count = {}

    bot.send_message(message.chat.id, 'Починаю сканування пакетів...')

    # Запуск потоку для сканування пакетів
    threading.Thread(target=packet_capture_thread).start()


def process_packet(packet):
    global packet_count
    global address_count

    # Обробка отриманого пакета
    packet_count += 1

    # Отримання адреси, до якої здійснено запит
    destination_address = packet[ARP].pdst

    # Підрахунок кількості запитів до кожної адреси
    if destination_address in address_count:
        address_count[destination_address] += 1
    else:
        address_count[destination_address] = 1

    # Збереження пристроїв, що з'являються
    if packet[ARP].psrc not in [device[1] for device in device_list]:
        #device_name = get_device_name(packet[ARP].psrc)
        device_list.append((device_name, packet[ARP].psrc, packet[ARP].hwsrc))

def check_duplicate_devices():
      duplicate_devices = []
      unique_devices = set()

      for device in device_list:
            if device[1] in unique_devices:
                  duplicate_devices.append(device)
            else:
                  unique_devices.add(device[1])
```

```
	return duplicate_devices

def check_new_devices():
	global device_list

	while True:
		output = subprocess.check_output(['arp-scan', '--localnet'])
		output_lines = output.decode('utf-8').split('\n')

		new_devices = []
		for line in output_lines:
			if line:
				elements = line.split('\t')
				if len(elements) >= 3:
					ip_address = elements[0]
					mac_address = elements[1]
					device_name = elements[2]
					device = (ip_address, mac_address, device_name)
					if device not in device_list:
						device_list.append(device)
						new_devices.append(device)
		if new_devices:
			message = "New PC:\n"
			for device in new_devices:
				message += f"new-devices:\nName:{device[2]}\nIP
adre:{device[0]}\nMAC adres:{device[1]}\n\n"
				bot.send_message(chat_id, message)
		time.sleep(60)


threading.Thread(target=check_new_devices).start()


def check_arp_spoofing():
    # Виконати команду 'arp -a' для отримання таблиці ARP
    output = subprocess.check_output(['arp', '-a'])
    arp_table = output.decode('utf-8')

    # Перевірка наявності дублікатів MAC-адрес
    mac_addresses = set()
    duplicate_mac_addresses = []

    for line in arp_table.split('\n'):
        if line.strip():
            # Розбити рядок таблиці ARP на елементи
            elements = line.split()

            # Отримати MAC-адресу
            mac_address = elements[3]

            # Перевірка, чи вже існує така MAC-адреса
            if mac_address in mac_addresses:
                duplicate_mac_addresses.append(mac_address)
            else:
                mac_addresses.add(mac_address)

    # Перевірка результатів
    if duplicate_mac_addresses:
        print('Знайдено дублікати MAC-адрес:')
        for mac_address in duplicate_mac_addresses:
            print(mac_address)
    else:
        print('Дублікати MAC-адрес не знайдено')
```

```
# Виклик функції для перевірки ARP-спуфінгу
check_arp_spoofing()

def packet_capture_thread():
    try:
        # Перехоплення пакетів з використанням scapy
        sniff(prn=process_packet, filter='arp', store=0)
    except Exception as e:
        print(f'Error in packet_capture_thread: {str(e)}')



# Функція-обробник мережевих пакетів
def process_DOS(packet):
    if IP in packet:
        ip_packet = packet[IP]
        if TCP in packet:
            tcp_packet = packet[TCP]
            packet_size = ip_packet.len - ip_packet.ihl * 4 - tcp_packet.dataofs * 4
            destination_ip = ip_packet.dst
            destination_port = tcp_packet.dport

            # Перевірка розміру пакета та адреси призначення
            if packet_size > THRESHOLD_SIZE and destination_ip == TARGET_IP and
destination_port == TARGET_PORT:
                message = "Можлива атака ARP spoofing"
                send_telegram_message(message)

# Основний код програми
THRESHOLD_SIZE = 1000  # Мінімальний розмір пакета для перевірки
TARGET_IP = 'TARGET_IP_ADDRESS'  # Адреса призначення для перевірки
TARGET_PORT = 'TARGET_PORT'  # Порт призначення для перевірки

# Створення обробника пакетів і встановлення фільтру
sniff(filter="tcp", prn=process_packet)

# Запуск перехоплення пакетів
def start_packet_capture():
    global packet_count
    global address_count

    packet_count = 0
    address_count = {}

    bot.send_message(chat_id, 'Починаю перехоплення пакетів...')

    threading.Thread(target=packet_capture_thread).start()

# Запуск телеграм-бота
bot.polling()
```

71

ІЛЮСТРАТИВНА ЧАСТИНА
ЗАСІБ МОНІТОРИНГУ БЕЗДРОТОВОЇ МЕРЕЖІ НА БАЗІ МІКРОКОМП'ЮТЕРА

Виконав: студент 2 курсу групи 1БС-21мс
спеціальності 125 Кібербезпека
Согур А.А.
«19» червня 2023 р.

Керівник: к. т. н., доцент каф. ЗІ
Куперштейн Л. М.
«19» червня 2023 р.

Схеми засобу – апартної частини

Структурна схема засобу моніторингу

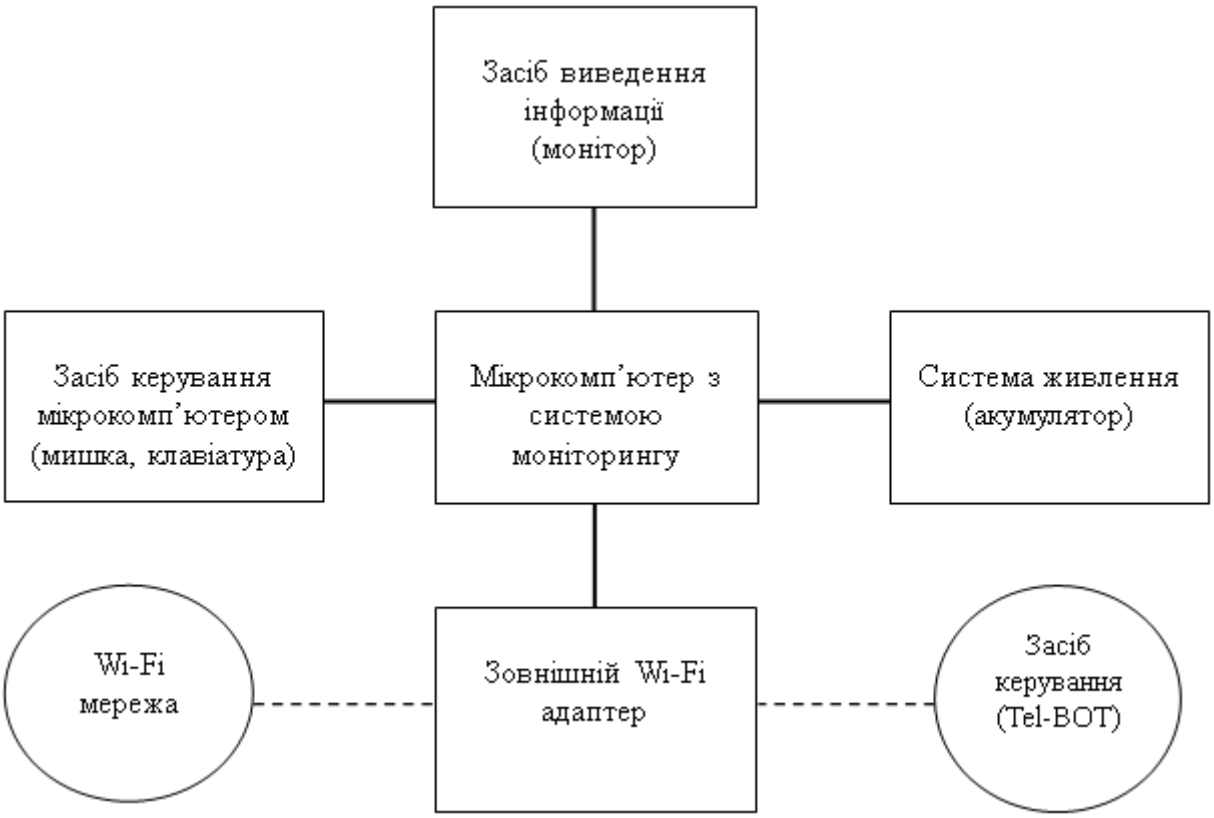


Схема обробки команди /start

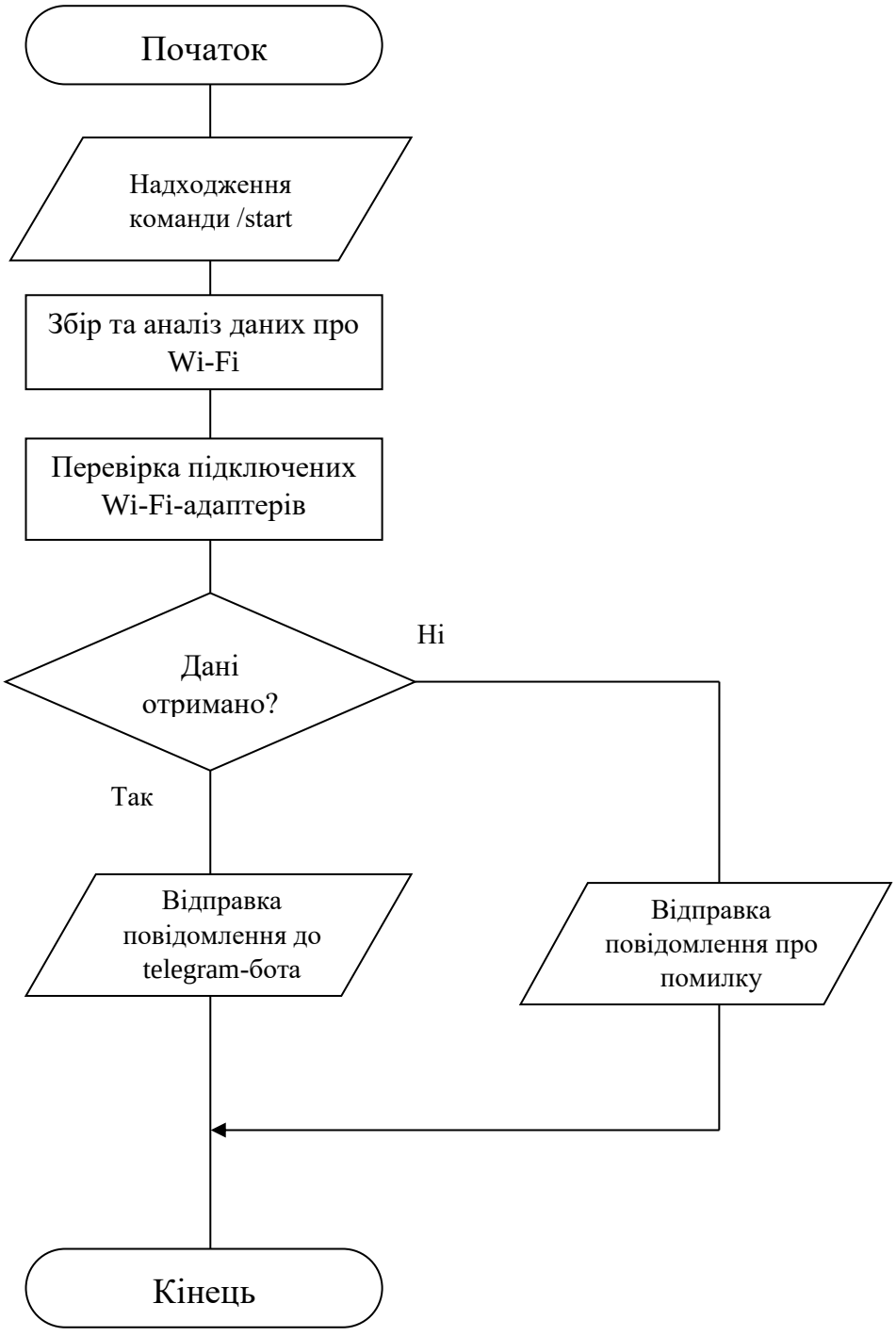


Схема обробки команди «/devices»

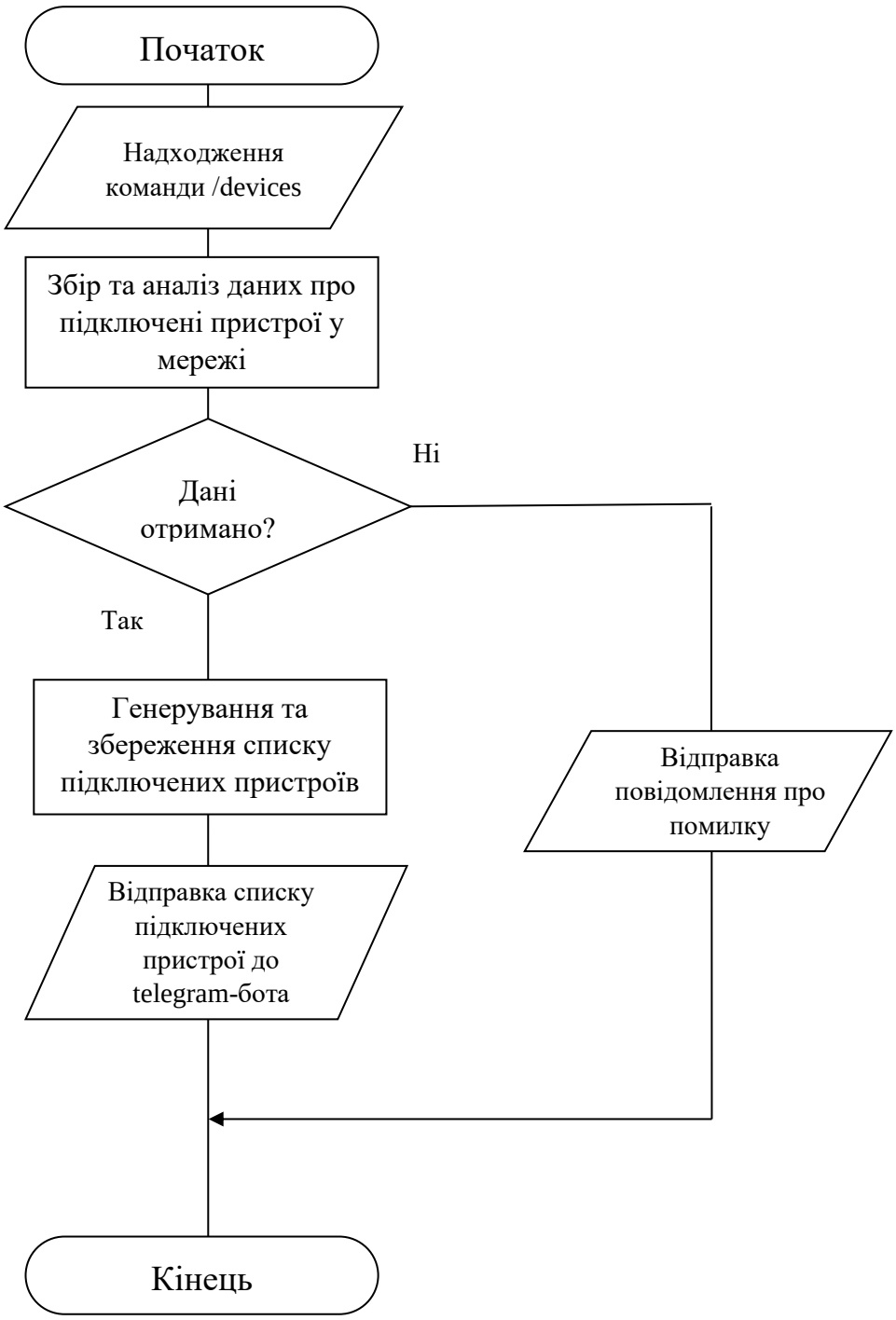


Схема обробки команди «/dump»

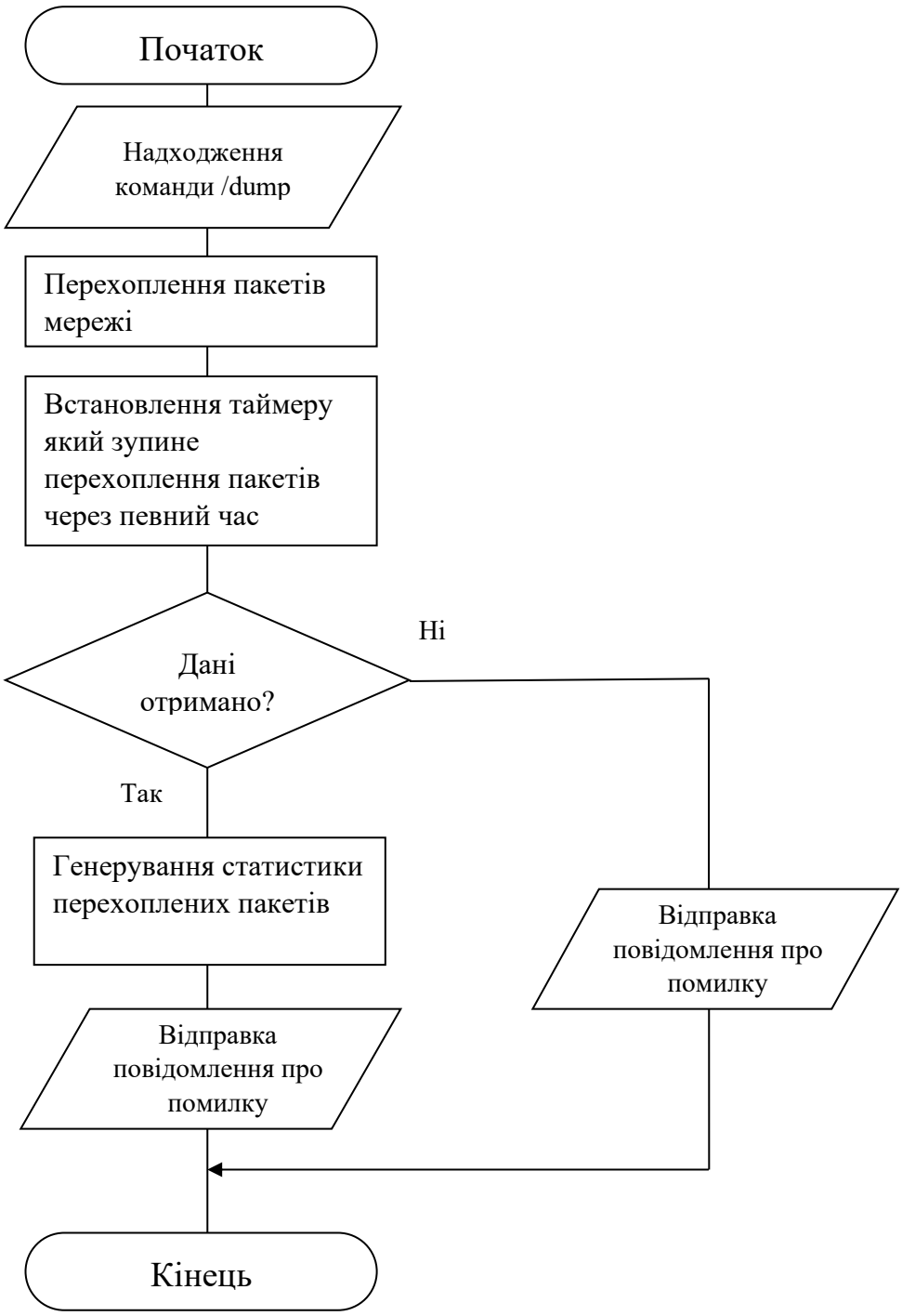


Схема виявлення Dos-атаки