Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістра
(освітній ступінь))

на тему: **Методи і засоби моніторингу та виявлення аномалій трафіку у локальних комп'ютерних мережах**

Виконав: студент (ка) 6 курсу, групи СІм-61
спеціальності 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

Демиденко А.О.
(підпис) (прізвище та ініціали)

Керівник Яцишин В.В.
(підпис) (прізвище та ініціали)

Нормоконтроль Тиш Є.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри Осухівська Г.М.
(підпис) (прізвище та ініціали)

Рецензент Деркач М.В.
(підпис) (прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
Кафедра комп'ютерних систем та мереж

ЗАТВЕРДЖУЮ
Завідувач кафедри *Осухівська Г.М.*

«*11*» *листопада* 2024 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *магістр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студенту *Демиденку Антону Олександровичу*
(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) *Методи і засоби моніторингу та виявлення аномалій трафіку у локальних комп'ютерних мережах*

Керівник проекту (роботи) *Яцишин Василь Володимирович, к.т.н., доц.*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «*29*» *листопада* 2024 року №*4/7-1144*

2. Термін подання студентом завершеної роботи

3. Вихідні дані до роботи *Принципи формування трафіку у комп'ютерних мережах, методи і засоби моніторингу пакетів у комп'ютерних мережах, методи штучного інтелекту для виявлення аномалій у даних*

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Аналіз характеристик трафіку комп'ютерних мереж і методів виявлення аномалій 2. Проектування архітектури апаратно-програмного засобу з інтегрованою функцією моніторингу і виявлення аномалій трафіку. 3. Апаратна і програмна реалізація засобів виявлення аномалій трафіку у комп'ютерних мережах 4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Актуальність і мета дослідження. 2. Задачі дослідження, об'єкт і предмет, наукова новизна і практична цінність дослідження. 3. Статистичні методи виявлення аномалій трафіку. 4. Архітектура маршрутизатора на концептуальному рівні 5. Організація сервісу виявлення аномалій трафіку 6. Алгоритми машинного навчання з учителем. 7. Результати виявлення аномалій трафіку. 8. Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та безпека в надзвичайних ситуаціях	*Осухівська Г.М., зав. каф. КС*		
	Стручок В.С., ст. викладач каф. ОХ		

7. Дата видачі завдання *11.11.2024 р.*

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	*Аналіз існуючих методів та засобів розв'язання науково-технічних проблем, що відповідають тематиці кваліфікаційної роботи.*	*20.11.2024 р.- 30.11.2024 р.*	*виконано*
2.	*Аналіз характеристик трафіку комп'ютерних мереж і методів виявлення аномалій*	*30.11.2024 р.- 04.12.2024 р.*	*виконано*
3.	*Проектування архітектури апаратно-програмного засобу з інтегрованою функцією моніторингу і виявлення аномалій трафіку*	*04.12.2024 р.- 09.12.2024 р.*	*виконано*
4.	*Апаратна і програмна реалізація засобів виявлення аномалій трафіку у комп'ютерних мережах*	*09.12.2024 р.- 15.12.2024 р.*	*виконано*
5.	*Охорона праці та безпека в надзвичайних ситуаціях*	*15.12.2024 р.- 17.12.2024 р.*	*виконано*
6.	*Оформлення пояснювальної записки і графічного матеріалу*	*17.12.2024 р.- 18.12.2024 р.*	*виконано*
7.	*Попередній захист кваліфікаційної роботи магістра*	*18.12.2024 р.*	*виконано*
8.	*Захист кваліфікаційної роботи магістра*		

Студент _____ *Демиденко А.О.*
(підпис) (прізвище та ініціали)

Керівник проекту (роботи) _____ *Яцишин В.В.*
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Демиденко А.О. Методи і засоби моніторингу та виявлення аномалій трафіку у локальних комп'ютерних мережах: робота на здобуття кваліфікаційного ступеня магістра: спец. 123 — комп’ютерна інженерія / наук. кер. В.В. Яцишин. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2024. — 89 с.

Ключові слова: метод, засіб, моніторинг, аномалії, трафік, комп’ютерна мережа.

У роботі проведено аналітичний огляд особливостей апаратно-програмної організації і принципів функціонування комп’ютерних мереж, як комунікаційної інфраструктури комп’ютерно-інформаційних систем в основі якого лежить IP-трафік. Запропоновано апаратно-програмне рішення щодо організації маршрутизатора на базі Raspberry PI 5 та прошивки OpenWrt з можливістю інтеграції наборів сервісів у вигляд Docker-контейнерів, що дає змогу імплементувати програмну модель виявлення аномалій трафіку та проведення його моніторингу з використання сервісу Snort практично у режимі реального часу.

Програмно реалізовано моделі машинного навчання для виявлення аномалій трафіку на основі алгоритмів навчання з учителем, у результаті аналізу яких обрано оптимальну за продуктивність і точністю модель, що базується на логістичній регресії.

ANNOTATION

Demydenko A.O. Methods and tools for monitoring and detecting traffic anomalies in local computer networks. Master’s Graduation Thesis: speciality 123 - Computer engineering/supervisor V.V. Yatsyshyn. Ternopil: Ternopil Ivan Puluj National Technical University, 2024. — 89 p.

Keywords: method, mean, monitoring, anomalies, traffic, computer network,

The paper provides an analytical review of the features of the hardware and software organization and the principles of functioning of computer networks as a communication infrastructure of computer and information systems based on IP traffic. This made it possible to identify the features and their importance in classifying traffic into classes "normal" and "anomalous".

A hardware and software solution for organizing a router based on Raspberry PI 5 and OpenWrt firmware with the ability to integrate service sets in the form of Docker containers is proposed, which makes it possible to implement a software model for detecting traffic anomalies and monitoring it using the Snort service in almost real time.

Machine learning models for detecting traffic anomalies based on supervised learning algorithms are implemented software, as a result of which the optimal model for performance and accuracy, based on logistic regression, was selected.

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ І СКОРОЧЕНЬ

ANN	Artificial Neuro Network
CNN	Convolutional Neuro Network
RNN	Recurent Neural Network
UML	Unified Modeling Language
КМ	Комп’ютерні Мережі

ВСТУП

Актуальність теми. Сучасні досягнення у сфері технічного, технологічного та інформаційного прогресу забезпечують можливість передачі різного типу та обсягу даних як на незначні, так і на великі відстані.

На сьогодні вже важко уявити ефективне функціонування будь-якої організації у якої відсутня комунікаційна інфраструктура. Мережі передачі даних забезпечують обмін інформації між відповідними структурними одиницями підприємства. Це дає змогу підвищити ефективність подальшого її опрацювання та проведення аналітичних досліджень. Організація доступу до мережі інтернет також покладається на мережеву інфраструктуру.

В залежності від сфери діяльності організації, всередині мережевої інфраструктури циркулюють різні за природою та вмістом потоки даних, які характеризуються певним об’ємом інформації, інтенсивністю пакетів, часом протягом якого відбувається обмін даними та іншими критеріями. Загалом інформаційні потоки у комп’ютерних мережах формують унікальний трафік для кожної організації.

Зважаючи на розвиток галузі інформаційних технологій, цінність та конфіденційність інформації, актуальними задачами сьогодення є аналіз трафіку на предмет аномалій. Це зумовлено великою кількістю загроз, які існують як всередині структурних підрозділів, так і ззовні. На параметри трафіку можуть впливати програми, які виконують зловмисні дії, наприклад, анонімно використовують ресурси обчислювальної техніки для майнінгу криптовалют, DDos атаки, надсилають дампи конфіденційних даних та ін.

У зв’язку з цим виникає необхідність розробки методів і засобів моніторингу і виявлення аномалій трафіку у локальних комп'ютерних мережах. Варто зазначити, що сьогодні розроблено багато підходів щодо побудови комп’ютерних мереж з високим рівнем надійності та безпеки, однак такі рішення потребують все ж вдосконалення з врахуванням досягнень у сфері штучного інтелекту.

Дослідженню принципів організації комп'ютерних мереж та можливостей щодо їх захисту присвячено ряд наукових публікацій українських вчених, зокрема, Виноградова М.А., Бурова Є.В., Журоковського Ю.Б., Митника М.М., Микитишина А.Г., Городецької О.С., Грайворонського М. В. та ін. Серед закордонних науковців і практиків, які зробили значний внесок у розвиток, забезпечення надійності та захисту комп'ютерних мереж варто відмітити праці Е. Таненбаума, Д. Уезерола, В. Оліфера, Р. Стівенсона, Д. Камера та ін.

Запропоновані підходи до моніторингу трафіку комп'ютерних мереж з можливістю виявлення аномалій досліджено не у повному обсязі, оскільки науковці основну увагу приділяли підвищенню ефективності організації мережі в цілому, а конкретні деталі потребують подальшого вивчення. Тому актуальною залишається розробка або вдосконалення існуючих методів і засобів моніторингу та виявлення аномалій трафіку.

Мета і задачі дослідження. Мета полягає у дослідженні методів і засобів моніторингу та виявлення аномалій трафіку у комп'ютерних мережах з використанням IoT пристроїв та моделі машинного навчання.

Для досягнення мети роботи необхідно розв'язати сукупність таких задач:

– аналіз підходів до проектування комп'ютерних мереж та організації сервісів моніторингу трафіку;

– виявлення та аналіз факторів, які впливають на формування трафіку;

– дослідження IoT пристроїв на предмет застосування в якості роутерів комп'ютерної мережі;

– аналіз підходів і моделей штучного інтелекту до виявлення аномалій даних;

– розробка апаратно-програмного рішення щодо забезпечення моніторингу та виявлення аномалій трафіку за критеріями гнучкості та надійності.

Об'єкт дослідження: процес моніторингу та виявлення аномалій трафіку у комп'ютерних мережах.

Предмет дослідження: методи та інструменти моніторингу трафіку, моделі і методи штучного інтелекту виявлення аномалій трафіку.

Методи дослідження: При розв’язанні поставлених задач кваліфікаційної роботи використано наступні методи:

– аналіз та узагальнення – при дослідженні характеристик трафіку комп’ютерних мереж і маршрутизаторів, а також при обгрунтуванні вибору підходу для виявлення аномалій трафіку;

– машинне навчання та математична статистика – при побудові моделі виявлення аномалій трафіку комп’ютерної мережі;

– проектування та програмування – при організації та налаштуванні маршрутизаторів комп’ютерної мережі та реалізації моделі засобами мови Python;

– експеримент – при оцінюванні якості моделі виявлення аномалій трафіку.

Наукова новизна отриманих результатів. Наукова новизна результатів дослідження полягає в наступному:

– уперше запропоновано апаратно-програмне рішення щодо організації маршрутизатора на базі Raspberry PI 5 та прошивки OpenWrt з можливістю інтеграції наборів сервісів у вигляд Docker-контейнерів, що дає змогу імплементувати програмну модель виявлення аномалій трафіку та проведення його моніторингу з використання сервісу Snort практично у режимі реального часу.

– набули подальшого розвитку, в контексті прикладного застосування, моделі машинного навчання бінарної класифікації з учителем, що дало змогу імплементувати у маршрутизаторі на базі мікроконтролера алгоритм виявлення аномалій трафіку та забезпечити моніторинг потоків даних, що проходять крізь нього.

Практичне значення одержаних результатів. Практична цінність одержаних під час виконання кваліфікаційної роботи результатів полягає у створенні маршрутизатора на базі Raspberry PI 5 з прошивкою OpenWRT та вбудованою моделлю виявлення аномалій трафіку, яка програмно реалізована мовою Python.

Публікації. Результати дипломної роботи апробовані на XIII міжнародній науково - технічній конференції молодих учених і студентів «Актуальні задачі сучасних технологій» (11-12 грудня 2024 р.) Тернопільського національного

технічного університету імені Івана Пулюя та на XII науково-технічній конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (18-19 грудня 2024 року) як тези конференцій.

1. Яцишин В.В., Демиденко А.О., Яцишин Вік.В. Підходи до виявлення аномалій трафіку у комп'ютерних мережах. Матеріали XIII міжнародної науково - технічної конференції молодих учених і студентів «Актуальні задачі сучасних технологій» (11-12 грудня 2024 р.) Тернопільського національного технічного університету імені Івана Пулюя. Тернопіль: ТНТУ. 2024. С. 520.

2. Яцишин В.В., Демиденко А.О., Яцишин Вік.В. Логістична регресія в задачах виявлення аномалій трафіку комп'ютерних мереж. Матеріали XII науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (18-19 грудня 2024 року). Тернопіль: ТНТУ. 2024. С. 149.

Структура роботи. Кваліфікаційна робота містить пояснювальну записку та графічний матеріал. До складу записки входить вступ, 4 розділи, загальні висновки, список використаних джерел і додатки. Обсяг роботи: пояснювальна записка – 88 арк. формату А4, графічна частина – 8 аркушів формату А1.

РОЗДІЛ 1
АНАЛІЗ ХАРАКТЕРИСТИК ТРАФІКУ КОМП'ЮТЕРНИХ МЕРЕЖ І МЕТОДІВ ВИЯВЛЕННЯ АНОМАЛІЙ

1.1. Аналіз характеристик трафіку

Сучасний стан розвитку IT-галузі забезпечує ефективність бізнес-процесів і постійно вдосконалюється на різних рівнях, починаючи від комунікаційної інфраструктури і закінчуючи рішеннями на базі інструментів штучного інтелекту. Забезпечення надійності і безпеки каналів передачі даних відіграє важливу роль у функціонуванні будь-якої комп'ютерної чи інформаційної системи і є актуальною задачею сьогодення. Це обумовлено рядом таких факторів, як необхідність автоматизації бізнес-сценаріїв і можливість забезпечити віддалене керування і доступ до даних. В основі комунікаційної інфраструктури більшості комп'ютерно-інформаційних систем лежать мережі передачі даних. Вони дають можливість об'єднати інформаційні компоненти системи в єдиний простір, забезпечивши при цьому рівні доступу і ролі конкретних користувачів.

За допомогою правильно організованої комп'ютерної мережі можна досягнути високої ефективності передачі та обміну даними з одночасним забезпеченням продуктивності апаратних складових при розв'язанні різного класу математичних задач. Важливість комунікаційної інфраструктури при провадженні освітнього процесу зумовлена необхідністю наявності доступу до електронних навчальних ресурсів, при проведенні відеоконференцій і перевірці знань студентів шляхом тестування. У виробничих умовах комп'ютерні мережі дають змогу забезпечити віддалений доступ до управління периферією, вбудованими компонентами, агрегатами, «розумними» верстатами та ін.

Створення єдиного інформаційного простору на основі комп'ютерних мереж володіє такими перевагами як [1]:

– ефективність і здатність оптимального та одночасного використання ресурсів апаратного забезпечення інформаційної системи багатьма користувачами;

– надійність і гнучкість побудови альтернативних шляхів передачі даних у випадку виникнення збою на вузлах комунікаційної інфраструктури;

– наявність засобів та інструментів для резервування баз даних і каналів зв’язку;

– підтримка різних способів віртуалізації окремих вузлів і цілих мереж на наявній апаратній інфраструктурі;

– наявність авторизаційних механізмів при організації доступу до апаратних засобів та інформаційних хостів.

На сьогодні найбільш поширеним є клас локальних комп’ютерних мереж. Належність комунікаційної інфраструктури до локальних комп’ютерних мереж визначається за наступними характеристиками:

– обмежена площа покриття і використання мережі, що, зазвичай, покриває територію в межах однією організації, а відстань між хостами становить не більше, ніж декілька кілометрів;

– наявність мультикористувацьких інтерфейсів при доступі до даних на вузлах мережі;

– керування доступом на основі правил, визначених адміністратором комунікаційної інфраструктури;

– стійкість, стабільність і працездатність сервісів, які передбачені і покладені на комп’ютерну мережу;

– фізична комунікація між визначеними вузлами на території розгортання комп’ютерної мережі.

Виходячи з наведених вище характеристик локальних КМ, комунікаційна інфраструктура на їх основі дає можливість оптимізувати затрати на накопичення, обробку і передачу даних. До переваг застосування комп’ютерних мереж відноситься і те, що значно скорочується частка непродуктивних втрат робочого часу працівниками. Як висновок, функціонування хостів у локальній КМ, дає можливість організувати гнучкість і масштабованість їхнього функціонування, а також оптимально розподілити ресурси, програми і дані за призначенням.

До важливих і, мабуть, визначальних в контексті оптимальності, характеристик будь-якої КМ належить спосіб її організації, тобто топологія. За даним критерієм можна оцінити ефективність зв'язків між компонентами мережі, а також їх структуру. Розрізняють два типи топології: фізична та логічна. Різниця між цими двома характеристиками мережі полягає у способі відображення комунікації між компонентами.

Фізична топологія покликана надавати об'єктивну інформацію про використовувані типи і способи фізичної комунікації наявних структурних елементів у КМ. На відміну від фізичної топології, логічна топологія зображає комунікаційну інфраструктуру на рівні логічної передачі даних між комопнентами мережі.

Обидві топології формують базис комунікаційної мережі, що максимально повно і широко відображає її важливі функціональні особливості. Володіючи інформацією про фізичну та логічну топології мережі з'являється можливість щодо проведення аналізу таких характеристик як пропускна здатність, тип трафіку та критерії завантаженості хостів, і як наслідок для підвищення безпеки з'являється можливість виявлення та моніторингу трафіку.

Тип, класифікацію і методи аналізу трафіку у КМ можна зобразити у вигляді, як продемонстровано на рис. 1.1.

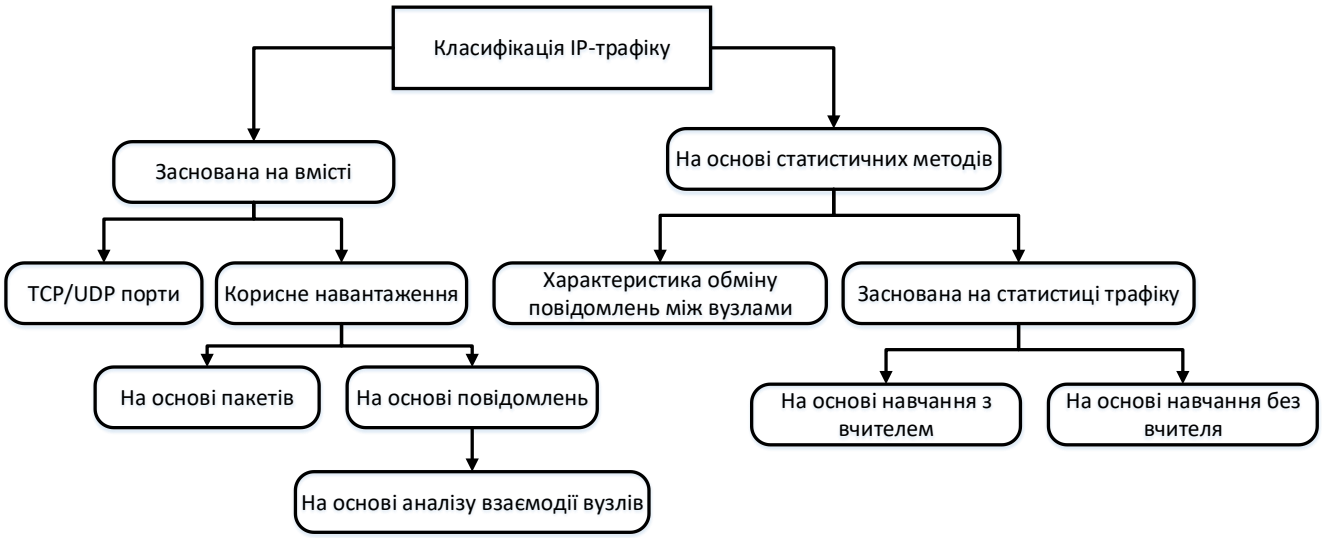

Рис. 1.1. Класифікація трафіку у комп'ютерних мережах

На класифікацію трафіку комп’ютерної мережі впливають також тип даних, які циркулюють у мережі і протоколи передачі, які застосовуються для обміну даних. Для проведення моніторингу та виявлення аномалій трафіку у КМ потрібно більш детально проаналізувати методи, які представлені на рис. 1.1.

1.2. Методи класифікації трафіку на основі аналізу вмісту пакетів

Особливість більшості програмного забезпечення, яке функціонує у комп’ютерній мережі, полягає у передачі чи обміні даними за допомогою вже відомих портів апаратних комунікаційних пристроїв. Враховуючи даний факт, задачу класифікації пакетів можна розглядати як задачу пошуку пакету (TCP SYN-packet) з метою виявлення серверної його частини. Наступний крок полягає в отриманні даних щодо доступності порту, який використовується для комунікації і функціонування конкретного програмного додатку, який формує запити чи отримує відповіді на цільовому порті. За подібним принципом виконується класифікація UDP-пакетів. Проте, варто відмітити, що з’єднання через порт не встановлюється.

Перевага такого методу класифікації трафіку полягає у простоті імплементації і забезпеченні високої продуктивності виконання операцій з визначення типу трафіку. До недоліків даного методу можна віднести наступні:

– суттєва залежність від типу програмного забезпечення, що супроводжується можливою відсутністю власних цільових портів у сервісі керування IP-адресним простором (IANA);

– наявність програмного забезпечення, що використовує нестандартні порти в рамках конкретної операційної системи для забезпечення визначеної функціональності, наприклад, для веб-сервера з сімейства ОС Unix застосування порта, що не відповідає порту 80;

– доволі висока імовірність появи помилок при виконанні шифрування пакетів, що може призвести до плутанини заголовків на рівні протоколів TCP і UDP.

Альтернативою методу аналізу портів при класифікації і виявленні типу трафіку в комп'ютерній мережі є підхід, що забезпечує отримання об'єктивних даних щодо використовуваного протоколу і базується на методах відновлення стану сеансу та аналізу вмісту кожного повідомлення. Це дає можливість зменшити залежність від портів.

Методи класифікації, в основі яких лежить аналіз корисного навантаження у випадку peer-to-peer трафіку, забезпечує опрацювання сигнатури пакетів прикладного рівня. Аналіз корисного навантаження дає можливість скоротити появу помилок до 5%. Це стосується, зокрема, помилок першого і другого роду.

Класифікація трафіку, що базується на аналізі корисного навантаження, може бути організована із застосуванням таких методів як:

– PBNS (Packet Based No State);
– PBFS (Packet Based Per Flow State);
– MBFS (Message Based Per Flow State);
– MBPS (Message Based Per Protocol State).

Для перевірки результатів класифікації на основі аналізу корисного навантаження можна застосувати чотири метрики. В якості першої метрики можна використати таку, яка описує результат пошуку деякої сигнатури на прикладному рівні. Наприклад, структура HTTP-пакету описується після адреси URL і даних щодо версії протоколу, а інформація щодо корисного навантаження міститься у полях структури Edonkey-пакетів.

Інший підхід і метрики, що використовуються для перевірки пакетів, передбачають необхідність проведення синтаксичного аналізу повідомлень. Процедури верифікації сигнатури в першу чергу зорієнтовані на перевірку коректності відправлених даних з точки зору синтаксичної структури. Як приклад, можна навести HTTP-повідомлення, заголовки яких повинні бути наявними в корисному навантаженні. При такій верифікації виконується декодування всіх полів з повідомлення.

Третій тип моніторингу трафіку дуже тісно пов'язаний з контролем протоколу відповідності і базується на припущенні, що завжди буде наявна

відповідь від сервера на запит клієнта. При такій формі верифікації забезпечується більша точність, оскільки виконується перевірка поведінки протоколу у відповідності до його специфікації.

Четвертий тип метрик та відповідних процедур верифікації трафіку базується на семантичній перевірці даних. В основі даного підходу лежить процедура виявлення і визначення типу об'єкту, який транслюється у комп'ютерній мережі. Для прикладу, можна визначати тип об'єкту, що належить до певного класу даних: зображення, відеопотік або інший тип інформації. Сьогодні методи семантичного аналізу відносяться до найбільш не формалізованих підходів, проте їх перевагою є можливість визначення ефекту «тунелювання».

PBNS метод орієнтований на виконання перевірки сигнатури пакету за наперед встановленими полями, які наявні у будь-якому пакеті. Враховуючи простоту даного методу з точки зору складності обчислень та перевірки лише окремих полів пакету, то зберігання станів є не доцільним.

PBFS метод передбачає проведення аналізу таблиці сеансів, виходячи з того, що будь-який сеанс зберігає дані про ідентифікатор пакету та ідентифікатор протоколу на прикладному рівні. До складу ідентифікатора сеансу входить:

– IP-адреса хоста-відправника/хоста-отримувача;
– ідентифікатор протоколу транспортного рівня;
– порти хоста-відправника/хоста-отримувача.

Об'єм таблиці сеансів вимірюється кількома десятками байтів.

В основі MBFS методу лежить аналіз повідомлень на основі модуля, що забезпечує нормалізацію TCP/IP пакетів. Принцип роботи і застосування методу MBFS дуже схожий до методу PBFS, що використовується для моніторингу корисного навантаження. Однак перший метод працює на рівні повідомлень у трафіку, а другий – на рівні пакетів. У зв'язку з цим, при застосуванні MBFS необхідно використовувати більший об'єм пам'яті для зберігання даних про:

– сеанс;
– стан сеансу;
– буферизацію – при використанні нормалізаторів TCP/IP.

На параметри методів, базованих на аналізі повідомлень, впливають тип трафіку та фрагментація пакетів під час сеансу.

Методи засновані на аналізі повідомлень володіють високою точністю інтерпретації повідомлень при комунікації компонентів як в межах окремого програмного додатку, так і при обміні даними між додатками.

Модуль аналізу повідомлень, окрім семантичного опрацювання самого повідомлення, забезпечує також моніторинг етапів його передачі. Для таких підходів характерно зростання показників апаратних засобів, зокрема, об'єму пам'яті. Це обумовлено необхідністю моніторингу трафіку на прикладному і транспортному рівні. Проте, незважаючи на зростання вимог до апаратного забезпечення при аналізі трафіку, ефективність даного методу є найбільшою у порівнянні з іншими методами і дає можливість опрацьовувати дані на прикладному рівні для забезпечення точності при класифікації повідомлень.

1.3. Статистичні методи аналізу трафіку

Методи статистичного аналізу в процесі моніторингу трафіку умовно можна розділити на два підходи:

– поведінкові підходи до аналізу трафіку;
– підходи аналізу мережевого й транспортного рівнів.

В основі поведінкових алгоритмів аналізу трафіку лежить принцип виявлення програмного забезпечення, яке формує левову частку трафіку комп'ютерної мережі. Основними результатами такого аналізу є встановлення вузлів, які обмінюються даними та програмні додатки, що функціонують на вузлах. Для ідентифікації програмного забезпечення та в якості предмету дослідження використовуються статистичні характеристики трафіку, який генерують відповідні вузли комп'ютерної мережі. Однак, тут потрібно враховувати припущення, що кожен клас програмного забезпечення на мережевому рівні характеризується унікальними параметрами. Важливо також забезпечити можливість ідентифікацію конкретного програмного продукту в межах класу до якого він відноситься.

Загалом, застосування методів статистичного аналізу при дослідженні трафіку комп'ютерних мереж, базуються на ідеях машинного навчання при розв'язанні задач класифікації. При цьому можуть використовуватися як методи навчання з вчителем (supervised learning), так і без учителі (unsupervised learning).

Враховуючи специфіку домену комп'ютерних мереж при розв'язанні задач класифікації трафіку для виявлення аномалій, деякі поняття мають дещо інший семантичний відтінок у порівнянні з класичним трактуванням. Використовувана термінологія відштовхується від наступних понять:

– простий потік;
– двосторонній потік;
– повний потік.

Поняття простого потоку описується за допомогою деякої множини послідовних однонаправлених пакетів, що обов'язково володіють такими ознаками як:

– адреса хоста-відправника на транспортному рівні;
– адреса хоста-отримувача на транспортному рівні;
– номер порта відправника;
– номер порта отримувача;
– унікальне значення протоколу.

Двосторонній потік можна представити за допомогою двох простих потоків з різною направленістю між хостами, які комунікують між собою.

Під повним потоком слід розуміти двонаправлений потік, який триває під час сеансу комунікації між відправником і отримувачем.

У випадку розв'язання задачі виявлення аномального трафіку із застосуванням статистичних методів класифікації можна розглядати такий різновид, як бінарна класифікація (характеристики трафіку, що генерує відоме ПЗ або інший трафік). Так, на рис. 1.2 показано процедуру організації бінарної класифікації. Вона включає в себе деяку вхідну множину характеристик трафіку, з якої виділяється частина даних для навчальної і тестової вибірки, а далі виконується побудова моделі та розробка алгоритмів класифікації.

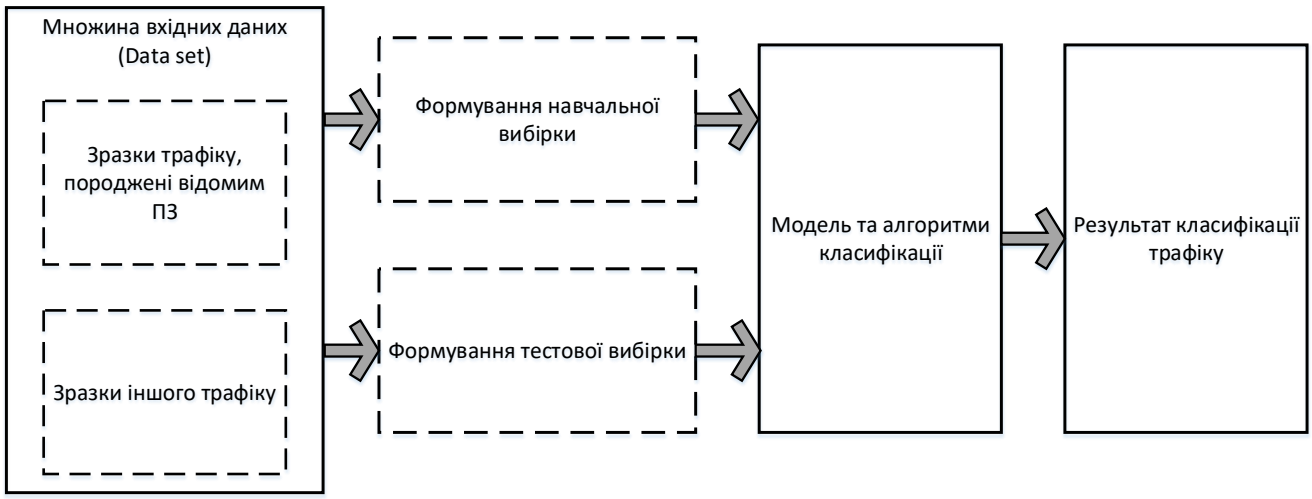

Рис. 1.2. Структура процесу бінарної класифікації трафіку у комп'ютерній мережі (класифікація з учителем)

При аналізі трафіку з метою моніторингу його характеристик та виявлення аномалій виконується ряд процедур, детальну структуру яких представлено на рис. 1.3. У випадку застосування підходу навчання з учителем обов'язково повинна бути наявна множина міток класів для двох видів трафіку – нормального та аномального.

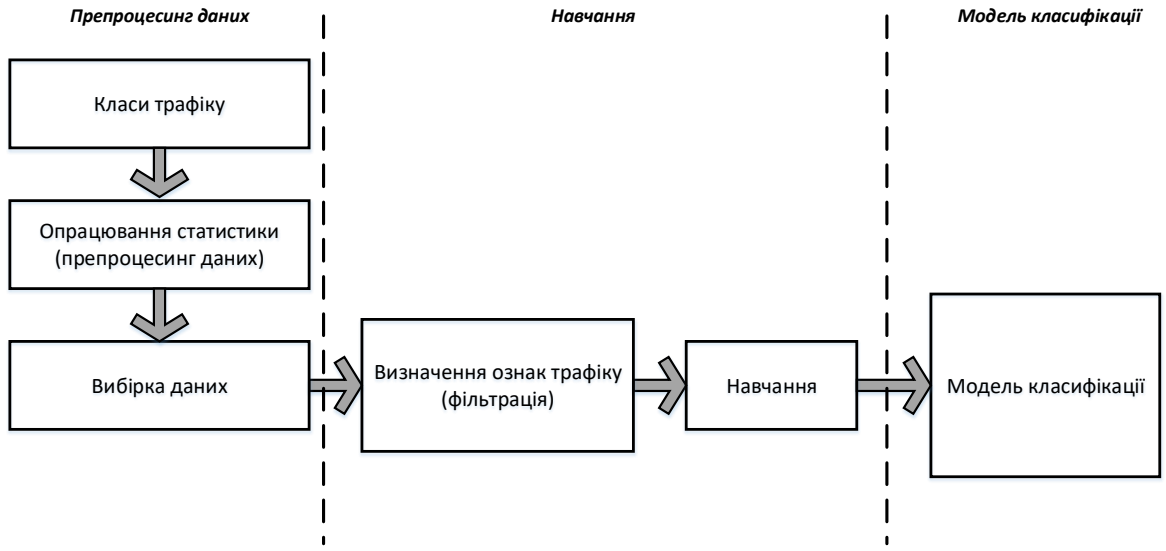

Рис. 1.3. Класифікатор з учителем

Послідовність виконання етапів алгоритму навчання з учителем при класифікації трафіку проілюстрована на рис. 1.4.

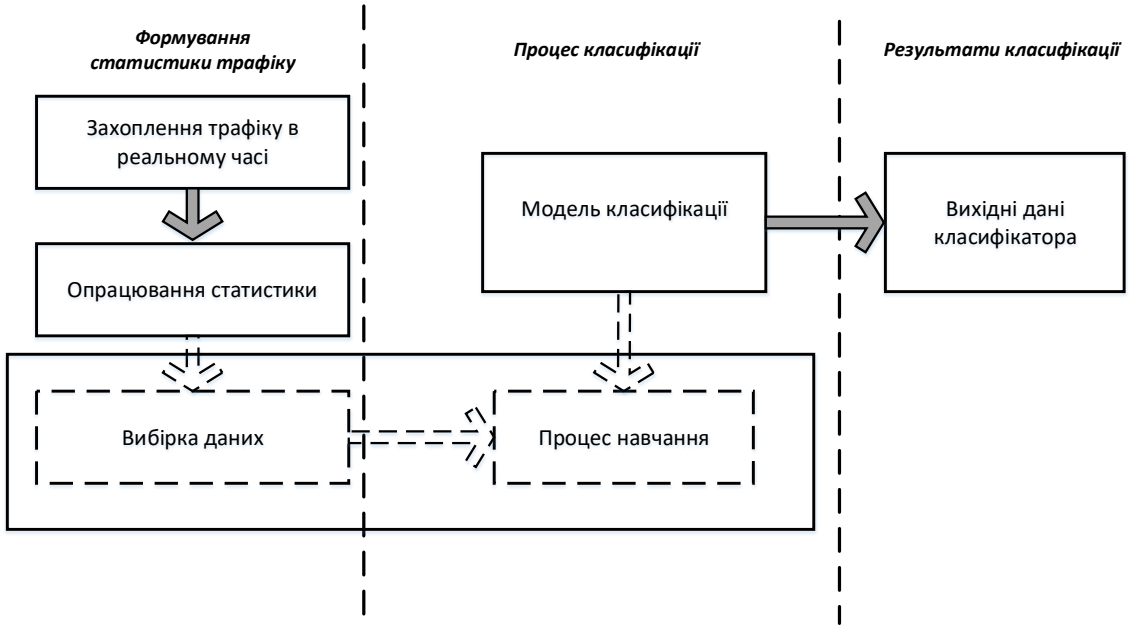

Рис. 1.4. Класифікація трафіку в режимі наближеному до реального часу

Алгоритм, представлений на рис. 1.4, на першому кроці передбачає формування статистичних ознак трафіку на основі вже функціонуючих цільових екземплярів програмних додатків. Такі ознаки становлять цільову змінну алгоритму класифікації. Окрім цього, до вхідних даних входять також ознаки трафіку, що формується відмінними від досліджуваних додатків, екземпляри іншого програмного забезпечення.

Ознаки трафіку, сформованого у режимі реального часу, є основою при виявленні ознак потоку, який безпосередньо подається на вхід алгоритму класифікації. Виходячи з цього, застосування підходу навчання з учителем є досить ефективним і дієвим інструментом при виявлення аномалій трафіку.

1.4. Аналіз сучасних інструментів аналізу трафіку у комп’ютерних мережах

На загальному концептуальному рівні, мережевий трафік можна означити як явище, що виникає у результаті передачі даних від джерела до деякого пункту призначення [1].

Моніторинг мережевого трафіку — це спостереження за вхідним і вихідним трафіком, що рухається по мережі. Аналіз мережевого трафіку є технікою

добування характеристик трафіку для розуміння його поведінки. Різні шаблони генеруються під час такого аналізу, які дають можливість формувати висновок про значущі ознаки його поведінки.

Однією з цілей і задач, які можна розв'язувати при аналізі трафіку є задача виявлення аномалій [2, 3]. Аномалії трафіку характеризуються невідомими до їх виникнення ознаками, а також значними відхиленнями від норми мережевого трафіку, що циркулює каналами комп'ютерної мережі.

Метод, який зазвичай застосовується при аналізі потоків даних при формуванні трафіку, базується на функції підрахунку кількості входжень таких параметрів, як кількість пакетів, кількість байтів багатовимірних часових рядів. Це дає змогу виявляти такі аномалії як «обурення мережі» (network outrage), «спалах натовпу» (flash crowds), «розповсюдження хробаків» (worm propagation) тощо.

Barford виконав аналіз трафіку, використовуючи дослідження сигналів, щоб виявити чотири категорії атак на трафік SNMP, використовуючи вейвлет-фільтри, як визначено в [6] та запропонував підхід під назвою k-ary sketch, який є модифікованою версією структури даних ескізу, що використовує менші об'єм пам'яті та має постійний запис тощо для підсумовування трафіку на різних рівнях, а потім його прогнозування за допомогою моделі авторегресійних ковзних середніх тощо для визначення значних помилок прогнозу.

Деякі з прикладів інструментів моніторингу та аналізу мережевого трафіку: wireshark, TCPDump, snort, bro, Xplico тощо. Це інструменти аналізу, і деякі з них також допомагають запобігти вторгненню. Існує широкий вибір майбутніх інструментів мережевого аналізу.

FlowScan — це інструмент аналізу та візуалізації мережевого потоку, який також використовується для створення звітів про мережевий трафік.

Iris — це техніка аналізу мережевого трафіку, яка може допомогти дослідникам у повторному розслідуванні вторгнень.

Зараз доступні численні інструменти збору та моніторингу мережевого трафіку. Деякі з них включають: wireshark, TcpDump, NfDump, PcapWT, Xplico, NetworkMinor, NetIntercept, Snort, PyFlag Iris, Bro

1.4.1. Аналізатор мережевого трафіку Wireshark

Wireshark – це найпоширеніший аналізатор мережевого трафіку. Він має можливість реалізувати захоплення мережевого трафіку в реальному часі у форматі libpcap. Як і багато інших інструментів, wireshark може досліджувати, перевіряти та аналізувати дані пакетів і виконувати аналіз.

Даний інструментальний засіб сумісний з кількома платформами, включаючи Windows, варіанти ОС Linux тощо. Він забезпечує фіксацію мережевого трафіку у формі пакетів і зберігає їх у буфері пакетів для подальшого вивчення та аналізу (рис. 1.5).

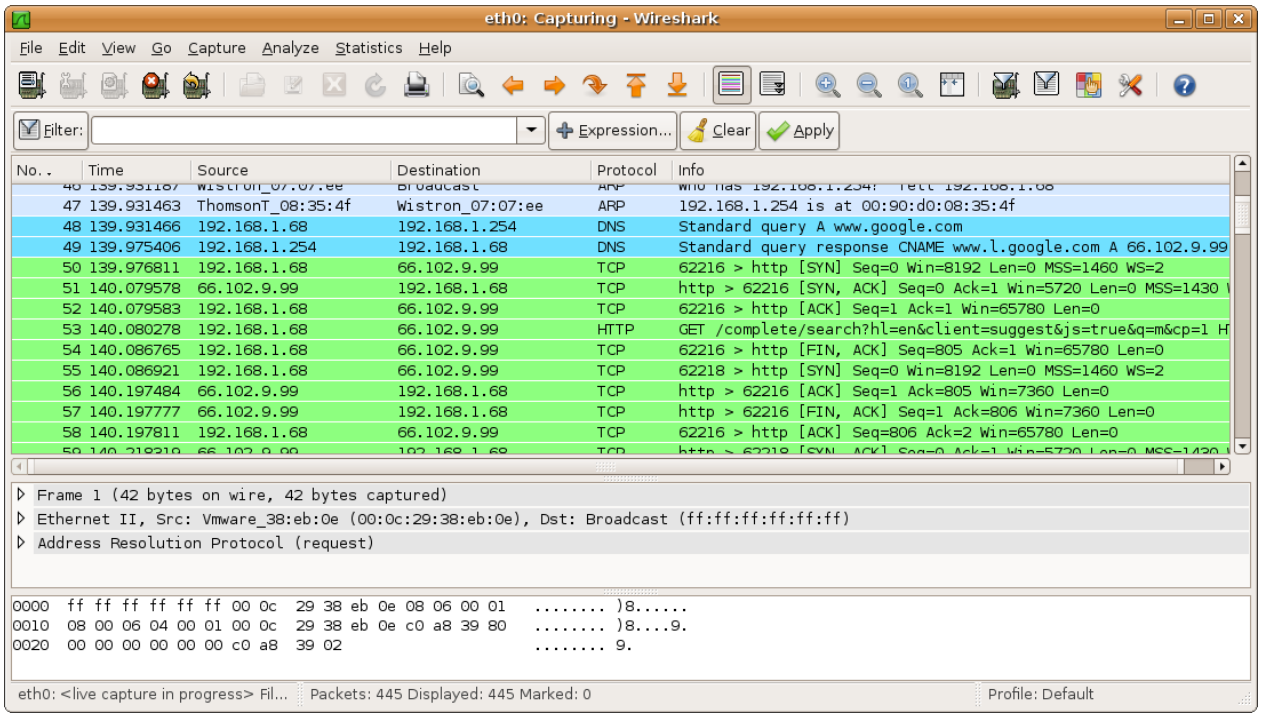

Рис. 1.5. Аналізатор мережевого трафіку Wireshark

До складу цього засобу аналізу трафіку входить багато попередньо налаштованих фільтрів, таких як: Wireshark Capture Filter, Wireshark Protocol Filters. Окрім цього, існує можливість створення нових фільтрів у відповідності до вимог. Користувач може переконатися, що вхідний мережевий трафік проходить через ці фільтри перед збереженням у буфері пакетів.

Окрім моніторингу мережевого трафіку, даний інструмент також допомагає аналізувати трафік для визначення проблем безпеки в мережі. Іншою особливістю цього інструменту є здатність відображати взаємодію між рівнями моделі OSI.

Wireshark забезпечує фільтри для перегляду номерів портів на рівні 4 та IP-адреси на рівні 3 моделі OSI. Трафік, пов'язаний із розмовою, можна зберігати та шукати за ключовими словами, де це необхідно. Список кінцевих точок із зони безпеки можна визначити за допомогою інструменту кореляції кінцевих точок для графічного їх відображення, які легко візуалізувати та зрозуміти.

Одним із обмежень wireshark, якщо його встановлено у відкритому середовищі, є витік паролів, тобто він чітко відображає ім'я користувача та пароль власників облікових записів, дані мережевого трафіку яких збираються. Він також неефективний при обробці великих об'ємних даних.

1.4.2. TcpDump

Походження TcpDump сягає 1990-х років у Національній лабораторії Лоуренса Берклі. Це типовий сніффер і аналізатор пакетів, які працюють у командному рядку. Він аналізує та забезпечує трансляцію вхідних/вихідних пакетів у мережі. Приклад команди для захоплення пакетів з поточного мережевого інтерфейсу показано на рис. 1.6.

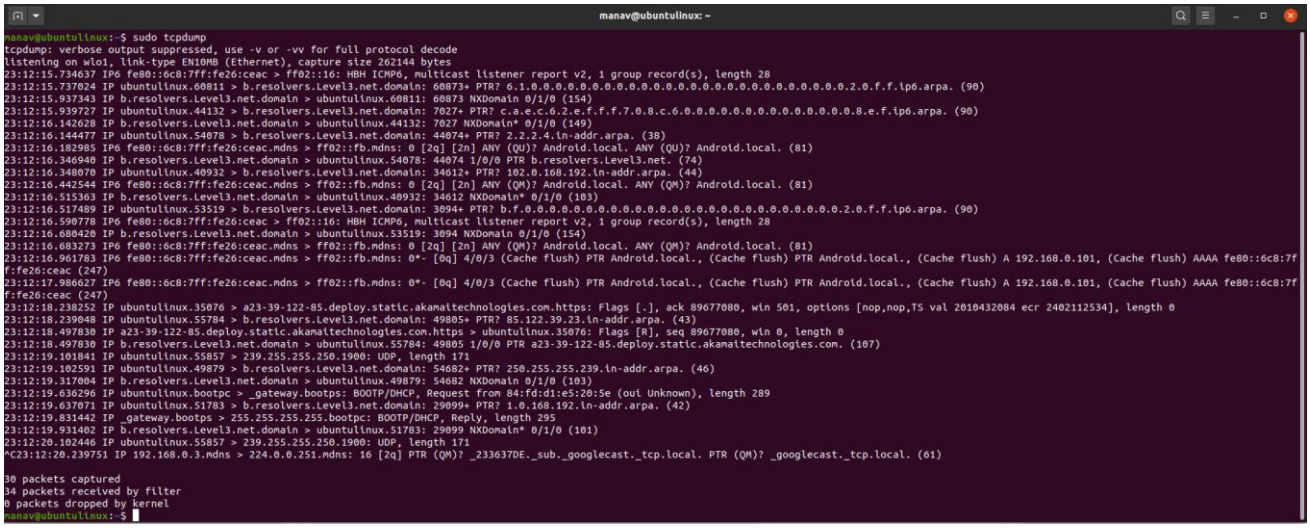

Рис. 1.6. Захоплення пакетів поточного мережевого інтерфейсу

TcpDump може відображати вміст пакета, наприклад мітку часу пакета, використаний протокол, адресу джерела, хости та порти призначення тощо. Даний засіб використовує CUI для кращої взаємодії користувача з системою та попередньо встановлено на Kali Linux [11]. Він використовується в основному, коли режим роботи є хаотичним [12][13].

Однією із сфер застосування tcpdump є створення брандмауера, наприклад, для McAfee і Juniper розгортають tcpdump у своєму наборі інструментів, щоб легко налагодити або повідомити про проблему [7].

TCPdump досягає даних швидше в порівнянні з інструментом wireshark [8]. Одним із обмежень Tcpdump є його неефективність обробки трасування великих пакетів. Tcpdump записує всі отримані дані або пакет на термінал, відображаючи мінімальну інформацію, включаючи тип пакета, наприклад, tcp, udp, icmp тощо. Щоб збільшити відображення інформації на вихідному терміналі, можна використовувати спеціальний тег під назвою verbose «-v», збільшивши його до трьох разів як «-vvv». Він навіть може отримувати дані з найнижчої лінії потоку. Одним із головних недоліків tcpdump є його нездатність транслювати дані прикладного рівня [8].

1.4.3. Xplico

Xplico – це свого роду інструмент сніфінгу, який фіксує мережевий трафік, опрацьовує його, нормалізуючи для подальшого використання [14]. Цей інструмент в основному використовується для добування аудіо сеансів із потоку [15]. Він використовується для реконструкції даних, згенерованих іншими інструментами збору мережевого трафіку, такими як wireshark, tcpdump тощо [16]. Графічний користувацький інтерфейс даного інструменту показано на рис. 1.7.

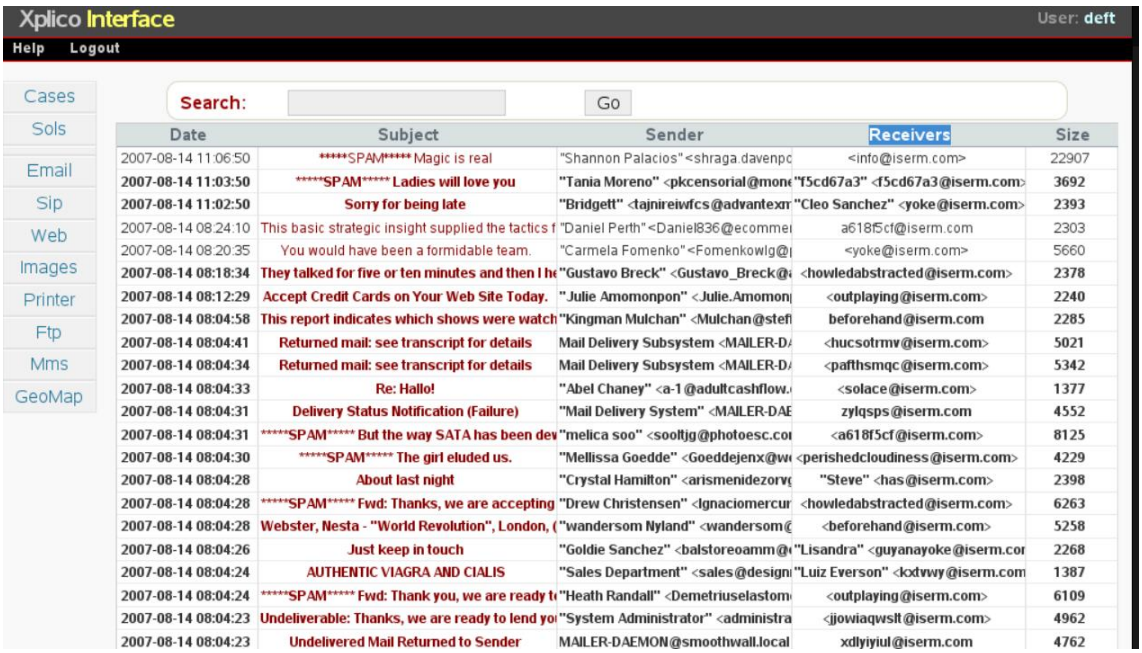

Рис. 1.7. Інтерфейс Xplico

Xplico може витягувати веб-сторінки з веб-даних, так само добувати, наприклад зображення, аудіо.

Даний засіб за замовчуванням доступний із дистрибутивом Linux Kali, який найкраще підходить для тестування на проникнення [17]. Одним з обмежень у Xplico є час доступу до жорсткого диска під час добування даних у реальному часі [18]. До плюсів можна віднести його здатність підтримувати багатокористувацьке середовище. Він також може підтримувати хмарний NFAT.

1.4.4. Snort

Snort – це мережевий інструмент, здатний виявляти вторгнення в мережу. Він також забезпечує механізми запобігання вторгненню в мережу та систему.

Даний засіб може виконувати реєстрацію пакетів і аналіз трафіку в реальному часі. Так само як і wireshark, snort також фіксує дані мережевого трафіку у форматі libpcap. Далі цей формат можна конвертувати в інші для подальшого аналізу [19].

Робота snort дещо схожа на tcpdump і відрізняється тим фактом, що перший виконує перевірку корисного навантаження пакетів. Форма налаштування сповіщень у Snort показана на рис. 1.8.

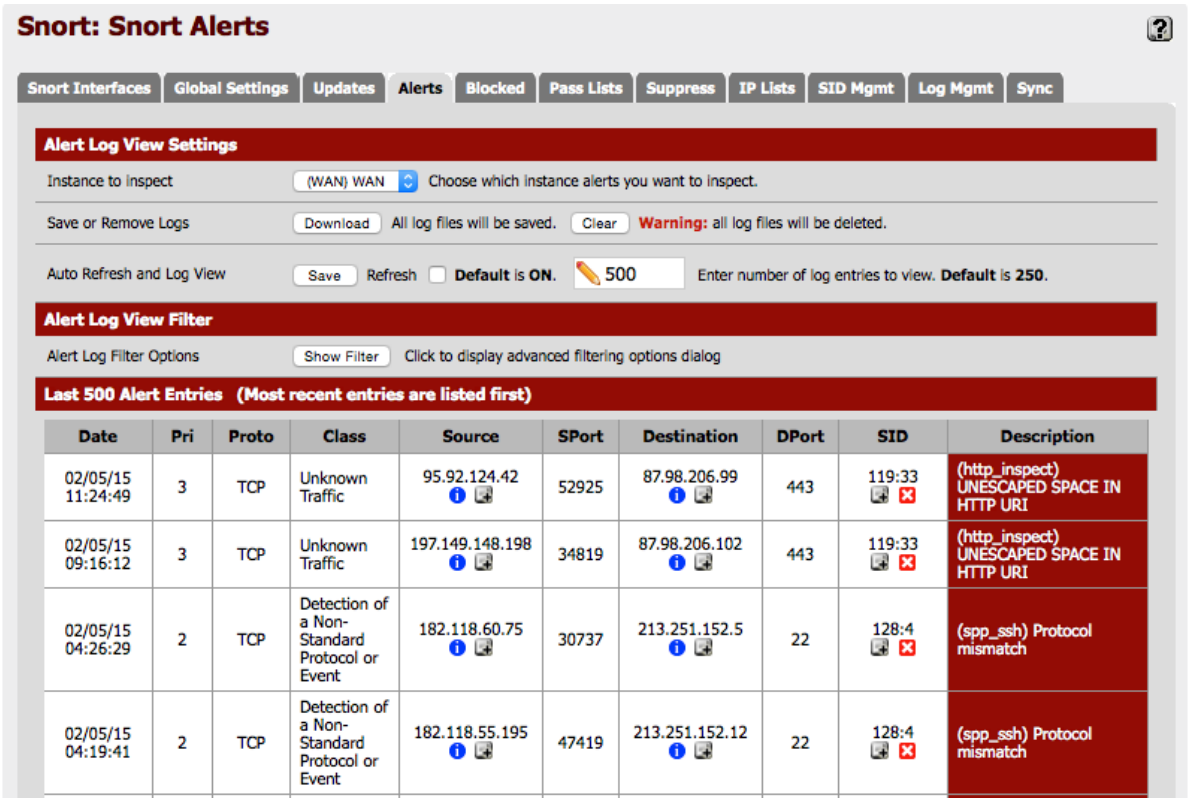

Рис. 1.8. Форма налаштування сповіщень

Одним із обмежень snort є те, що він не шукає імена хостів або портів під час роботи, оскільки швидко фокусується на максимальному зборі пакетів [20]. Плюси snort включають його здатність фільтрувати пакети на основі певної категорії, такої як тип протоколу, який переважно використовує команди фільтра пакетів Берклі (BPF) [19].

1.5. Висновки до розділу

1. Проведено аналітичний огляд особливостей апаратно-програмної організації і принципів функціонування комп'ютерних мереж, як комунікаційної інфраструктури комп'ютерно-інформаційних систем в основі якого лежить IP-

трафік. Це дало змогу виявити ознаки і їх важливість при класифікації трафіку за класами «нормальний» та «аномальний».

2. Проаналізовано сучасні методи класифікації трафіку на основі вмісту пакетів/повідомлень, а також статистичні методи і методи машинного навчання, що дало змогу обґрунтувати застосування моделей машинного навчання з алгоритмами класифікації з учителем для виявлення аномалій трафіку.

3. Проведено аналіз сучасних інструментів моніторингу трафіку у комп’ютерних мережах, що дало можливість обґрунтувати доцільність застосування відкритих сервісів, які можуть інтегруватись у будь-які апаратно-програмні засоби з підтримкою Unix-подібних операційних систем, зокрема, у маршрутизатори різних рівнів.

РОЗДІЛ 2
ПРОЕКТУВАННЯ АРХІТЕКТУРИ АПАРАТНО-ПРОГРАМНОГО ЗАСОБУ З ІНТЕГРОВАНОЮ ФУНКЦІЄЮ МОНІТОРИНГУ І ВИЯВЛЕННЯ АНОМАЛІЙ ТРАФІКУ

2.1. Аналіз функцій і структури вузлів комп’ютерної мережі, що забезпечують маршрутизацію пакетів

Ідея, яка висувається у кваліфікаційній роботі магістра, полягає у тому, щоб спроектувати пристрій на основі мікроконтролера та інших IoT пристроїв, які забезпечували б надійність функціонування при формуванні потоків трафіку та виявленні аномалій у ньому. Для цього, спершу, потрібно проаналізувати існуючі на ринку рішення реалізації маршрутизаторів, визначити їхню функціональність, переваги та недоліки

Маршрутизатор, або по-іншому роутер, покликаний забезпечити виконання таких базових функцій як:

- пересилання пакетів;
- організація маршрутів;
- трансляція адрес хостів;
- забезпечення безпеки;
- підтримка налаштувань якості обслуговування (QoS);
- здатність підтримки віртуальних приватних мереж (VPN);
- забезпечення управління смугою пропускання;
- наявність засобів моніторингу і діагностики;

Функція маршрутизатора щодо здатності пересилання пакетів полягає в тому, що при одержанні пакетів на вхідних портах, пристрій виконує перевірку їхніх заголовків, здійснює ряд визначених функцій, наприклад, перевіряє контрольну суму і лише після цього аналізує таблицю маршрутизації для знаходження встановленого вихідного порту для надсилання пакетів даних до адресата.

Функція організації та контролю маршрутів представляє собою процес оптимального вибору маршруту для доставки пакету чи повідомлення адресату за найкращим шляхом. Роутер забезпечує підтримку актуального стану таблиці маршрутизації, яка, зазвичай, формується з використанням різних алгоритмів, що підтримуються конкретним типом маршрутизатора.

Функція трансляції адрес хостів забезпечує взаємодію вузлів з різними діапазонами IP-адрес. Для цього використовується сервіс NAT, що дає змогу активним вузлам комп'ютерної мережі у VPN одержати можливість доступу до мережі інтернет з використанням єдиної публічної IP-адреси.

Функція чи сервіс забезпечення безпеки, який покладається на маршрутизатор, орієнтований на запобігання несанкціонованому доступу до ресурсів, виявленню зловмисних програм чи інших загроз. Зазвичай, функція налаштування параметрів безпеки виконується у брандмауерах або інших спеціалізованих сервісах.

Функція підтримки налаштувань якості обслуговування полягає у забезпеченні можливості встановлювати пріоритетність потоків мережевого трафіку з врахуванням типу даних. Цінність функції QoS полягає в тому, що критичне програмне забезпечення і служби завжди будуть одержувати достатній рівень пропускної здатності.

Функція здатності підтримки віртуальних приватних мереж забезпечує можливість віддаленим користувачам можливість безпечного під'єднання до КМ на основі VPN.

Функція управління смугою пропускання дає змогу контролювати об'єм даних, що мають дозвіл на передачу у мережі. Це запобігає надлишковому навантаженню і гарантує необхідну пропускну здатність для критичного програмного забезпечення.

Функції моніторингу і діагностики, які покладаються на роутери, дають можливість контролювати трафік і виявляти збої у роботі активних і пасивних вузлів комп'ютерної мережі з одночасним сповіщенням адміністраторів про виникнення проблем у комунікаційній інфраструктурі.

Структурно, типовий роутер, представлений на рис. 2.1.

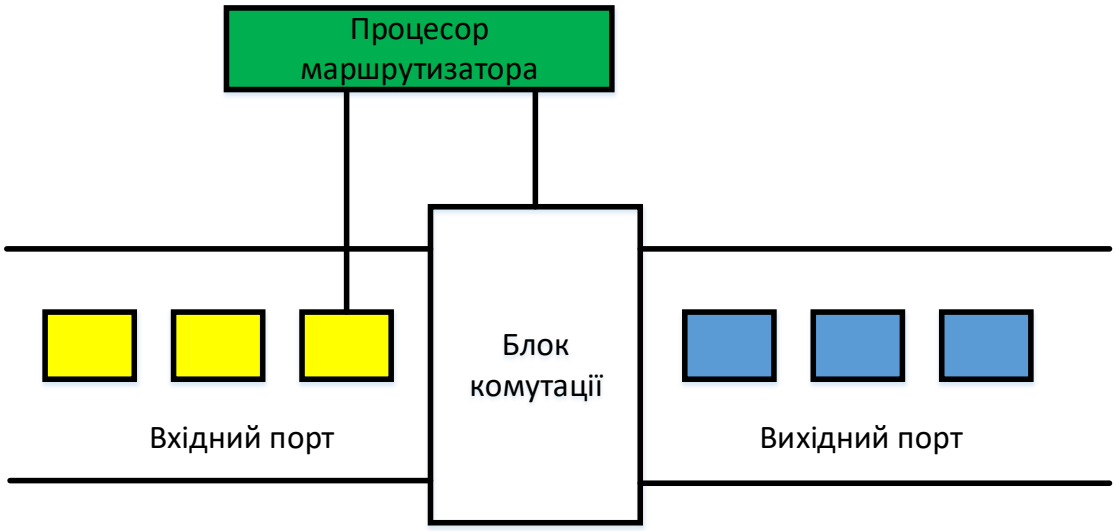

Рис. 2.1. Структура типового маршрутизатора

Вхідний порт представляє собою інтерфейс входу пакетів даних у маршрутизатор. На даний порт покладається така ключова функція, як припинення фізичної комунікації – виконується крайнім лівим блоком, як показано на рис. 2.1, середній блок виконує задачі забезпечення взаємодії з канальним рівнем шляхом декапсуляції пакету, а крайній правий блок виконує функції пошуку таблиці маршрутизації з метою визначення номера вихідного порту за адресою призначення.

Блок комутації є основною частиною будь-якого роутера, оскільки забезпечує зв’язок між вхідними і вихідними портами. Даний блок можна розглядати як свого роду мережу інтегровану у даний пристрій. При цьому реалізація комутаційної структури може виконуватися кількома способами.

Процесор маршрутизатора забезпечує копіювання пакетів, наявних на його вхідних портах, з подальшою передачею даних на вихідні порти. Даний структурний елемент маршрутизатора за принципом роботи подібний до традиційного центрального процесора з портами вводу/ виводу, які поводять себе аналогічно до пристроїв введення/виведення.

Окрім основних структурних елементів, на рис. 2.1 є шина, яка забезпечує з’єднання усіх вхідних портів з усіма вихідними. Алгоритм роботи при передачі даних з вхідного порту на вихідний передбачає виконання таких кроків:

– отримання пакету і визначення номера вихідного порту (адресата) вхідним портом;

– встановлення вхідним портом токена для вихідного порта;

– передача пакету даних до комунікаційної шини;

– доставка пакету даних до вихідного порту, що відповідає записаному токену;

– видалення токена (маркера) вихідним портом і доставка пакету адресату.

У маршрутизаторах може бути імплементована більш складніша мережа шин (роутер з багатьма шинами) для підвищення його продуктивності, що забезпечує з’єднання m вхідних та вихідних портів.

Ще одним важливим компонентом маршрутизатора є блок вихідних портів – сегмент, що відповідає за передачу пакетів з маршрутизатора до адресата. Функціями цього блоку є перегляд буферів черг, що можуть виникати у випадку, коли декілька пакетів використовують один і той самий вихідний порт, прийом пакетів з шини, виконання функцій канального рівня моделі OSI, передача пакетів у вихідний канал.

До алгоритмів, що можуть використовуватися процесором маршрутизатора, належать такі як: аналіз стану зв’язку, вектор відстані та ін. Застосування цих алгоритмів забезпечує підготовку таблиці маршрутизації дає змогу визначати маршрути та номер вихідних портів.

Проте, з точки зору безпеки, у маршрутизаторах існує кілька проблем щодо зовнішнього несанкціонованого втручання. До них належать:

1. Наявність вразливостей у системному програмному забезпеченні.

Системне програмне забезпечення (СПЗ) маршрутизатора повинно автоматично встановлюватися та оновлюватися для забезпечення його апаратної працездатності і покладених на нього функцій. СПЗ, тобто прошивка, може містити недоліки, які можна використовувати проти маршрутизатора. Виробники роутерів,

намагаються регулярно випускати оновлення для виправлення вразливостей, однак, у більшості випадків користувачі не спішать їх встановлювати, або частота випуску оновлень є дуже високою. Тому для кіберзлочинців існує можливість відстеження трафіку на маршрутизаторах, які не містять оновлень.

2. Атаки на мережеву інфраструктуру.

Сьогодні найбільш поширеними і загрозливим є DDoS атаки – розподілені атаки, що характеризуються відмовою в обслуговуванні мережі. Зазвичай, результат успішної атаки призводить до збоїв на мережевому рівні, спричиняючи надмірне навантаження на маршрутизатори для виведення їх з працездатного стану. Для захисту від DDoS-атак можна використовувати сервіс Cloudflare.

3. Слабкі налаштування профілів користувачів з правами адміністратора.

Для того, щоб забезпечувати необхідну функціональність будь-якого маршрутизатора потрібно володіти набором дозволів і прав, що відповідають обліковому запису з роллю адміністратора. За замовчуванням, значення параметрів облікових записів для доступу до пристрою маршрутизації відомі практично всім користувачам, які використовують відомі бренди. Так, найбільш популярними значенням за замовчуванням для адміністратора маршрутизатора є логін і пароль «admin».

Серед переваг використання маршрутизатора при побудові комп’ютерної мережі варто виділити:

– простоту інтеграції;
– наявність вбудованих служб безпеки;
– застосування NAT;
– наявність механізмів динамічної маршрутизації;
– можливість фільтрації повідомлень та пакетів.

Простота інтеграції маршрутизаторів у комп’ютерну мережу забезпечує можливість одночасного спільного доступу користувачів для визначених ресурсів та мережі інтернет. Окрім цього, роутери мають можливість комунікації з неоднорідними мережевими компонентами і конструкціями.

Наявність вбудованих служб безпеки по типу брандмауерів надає додатковий захист для вузлів мережі і потенційно знижує ризики проникнення зловмисних програмних продуктів чи хакерських атак. Окрім цього, маршрутизатор можна використовувати як деякий буфер, що розділяє середовище між кількома мережами. Однак, сучасні маршрутизатори не забезпечують функцій антивірусів чи типових брандмауерів операційних систем на хостах.

З метою інкапсуляції кількох приватних IP-адрес у єдину публічну, на рівні маршрутизаторів застосовується NAT. Це дає можливість підвищити якість підключення до глобальної мережі Інтернету та забезпечити ефективність потоків даних між наявними апаратними засобами, які входять до складу мережі.

Маршрутизатор, для забезпечення мережевої комунікації, володіє засобами підтримки динамічної маршрутизації. При цьому важливо, що оптимальність маршрутів при використанні і доступі до інтернету обирається власне на основі динамічної маршрутизації. Однак, це може призводити до утворення колізій і широкомовних доменів, а також зменшувати трафік у мережі.

Служби фільтрації забезпечують функції переходу між пакетами та безпосередньо їх фільтрування. За допомогою налаштовуваних наборів правил виконується аналіз дозволів або заборони певних видів трафіку на основі аналізу пакетів.

Недоліки маршрутизаторів:

– недостатня продуктивність;
– висока ціна;
– необхідність і складність налаштування параметрів;
– залежність і незручність як від частих, так і пізніх оновлень конфігурації;
– виникнення проблем, пов’язаних з якістю передачі даних;
– недостатність ширини пропускних каналів;

Самі по собі апаратні пристрої маршрутизації пакетів у КМ є досить повільними, що пов’язано з необхідністю аналізу інформації на кількох рівнях: починаючи з фізичного і завершуючи мережевим. Це значно сповільнює з’єднання.

Аналізуючи ринок засобів, які використовуються при адмініструванні комп’ютерних мереж, маршрутизатори характеризуються вищою ціною при доволі небагатому функціональному потенціалі. Вони є менш гнучкими в налаштуваннях, менше піддаються масштабованості та координації.

Для того, щоб забезпечити надійність функціонування пристроїв маршрутизації необхідно виконати доволі багато налаштувань у їхній конфігурації. У зв’язку з цим, чим складніші задачі покладаються на даний пристрій, тим складніше налаштування параметрів необхідно забезпечити.

Якість передачі даних при використанні маршрутизатора в якості безпровідної точки доступу характеризується частими збоями у роботі при переході з частоти 2,4 ГГц на 5ГГц і навпаки.

У результаті проведеного аналізу можна зробити висновок про те, що маршрутизатори мають багато переваг у порівнянні з іншими пристроями, що використовуються у комп’ютерних мережах, однак володіють і рядом недоліків.

Для вирішення задач виявлення та моніторингу аномального трафіку у комп’ютерних мережах пропонується скористатися не стандартними рішеннями його організації, а з використанням мікроконтролера Raspberry PI 5, прошивки OpenWRT та побудови власної моделі виявлення аномального трафіку, який також розгортається на даному пристрої.

2.2. Проектування архітектури та залежностей програмного забезпечення маршрутизатора на базі Raspberry PI 5

Для виявлення аномалій трафіку та його моніторингу у кваліфікаційній роботі запропоновано архітектурне рішення, що засноване на використанні мікроконтролера Raspberry PI 5. При цьому в якості системного програмного забезпечення використовується OpenWrt з вбудованою системою виявлення та запобігання атакам. Окрім цього, за допомогою технології Docker пропонується забезпечити інтеграцію додаткових сервісів, серед яких – модель на базі алгоритмів машинного навчання для визначення аномальних потоків трафіку.

На рис. 2.2 представлено схему архітектури маршрутизатора на базі Raspberry PI 5 на концептуальному рівні.

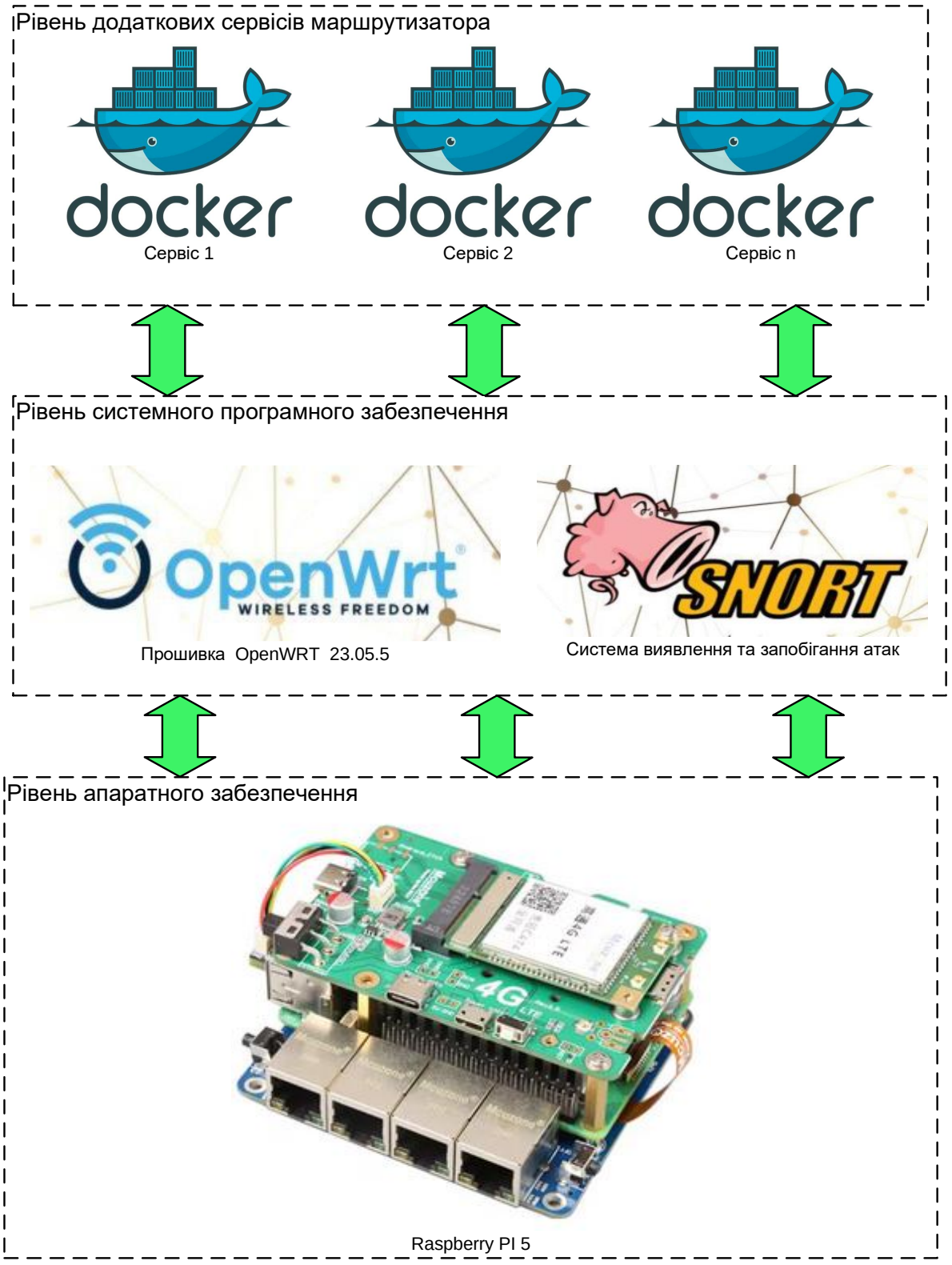

Рис. 2.2. Проект архітектури маршрутизатора на концептуальному рівні

В контексті виявлення аномального трафіку та проведення його моніторингу, запропоновано більш деталізовану схему цього процесу, яка проілюстрована на рис. 2.3.

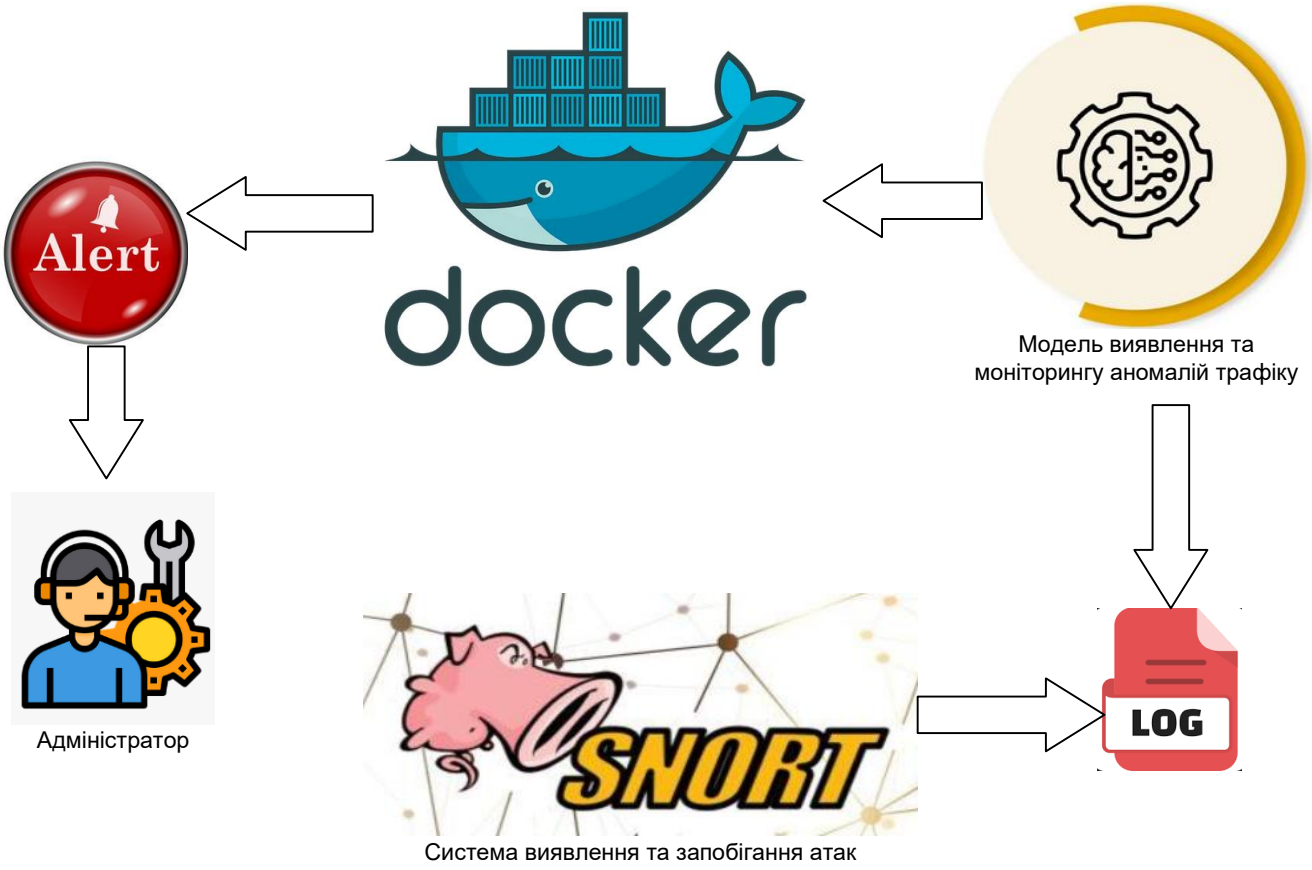

Рис. 2.3. Схема комунікації сервісів при виявленні аномального трафіку

Як видно з рис. 2.3, процес виявлення аномального трафіку починається з того, що сервіс Snort формує логи в процесі моніторингу трафіку, що проходить через маршрутизатор. Сам сервіс також може виконувати перевірку на аномальність трафіку, однак результати його роботи не в повній мірі і не завжди задовольняють, що пов’язано з неточністю прийняття рішень. Дані логів трафіку із застосуванням Snort можуть бути представлені у файлах різних форматів, зокрема, у CSV форматі. Це зручно для подальшого їх зчитування запропонованою моделлю виявлення та моніторингу аномалій трафіку.

Для того, щоб побудована модель виявлення аномалій ефективно працювала на базі прошивки OpenWrt, необхідно виконати її упаковку у docker-контейнер, що

забезпечить збереження залежностей на рівні бібліотек та використання необхідних апаратних ресурсів.

У випадку виявлення аномалій трафіку запропонованою моделлю виконується надсилання сповіщення на електронну пошту адміністратора комп’ютерної мережі.

Окрім моделі для виявлення аномалій трафіку у кваліфікаційній роботі варто організувати сервіси, структуру яких показано на рис. 2.4.

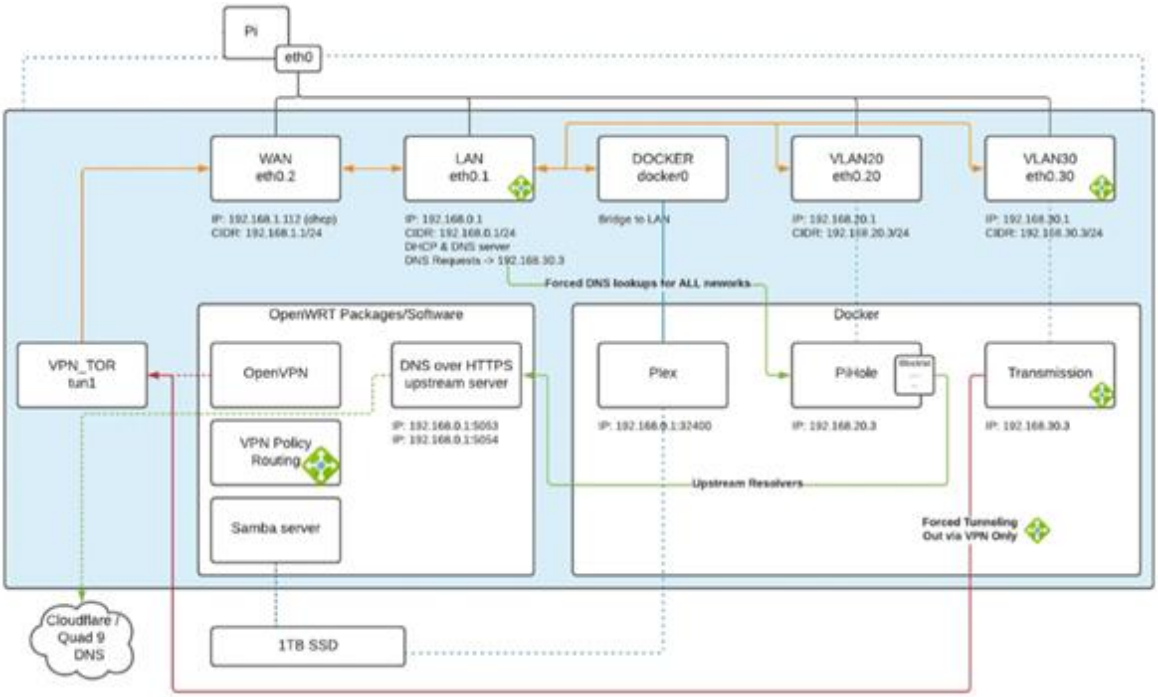

Рис. 2.4. Додаткові сервіси маршрутизатора з можливістю виявлення аномалій трафіку

Виходячи із структури сервісів, представлених на рис. 2.4, потрібно реалізувати наступні можливості:

– застосування одного порту у Raspberry PI 5 зі швидкість передачі даних на рівні 1 Гб/с з метою управління та моніторингу потоків трафіку;

– запуску Docker-контейнерів в OpenWRT;

– налаштування сервера Plex;

– налаштування сервісу PiHole при блокуванні реклами;

– налаштування вихідних серверів DNS через HTTPS.

При реалізації наведеної вище інфраструктури необхідно також виконати налаштування VLAN, а доступ до медіа вмісту на SD-карті повинен бути авторизованим. Сервіси PiHole, Plex та інші користувацькі служби і моделі повинні бути упаковані у Docker-контейнери та функціонувати на базі маршрутизатора з мікроконтролером Raspberry PI 5.

Таким чином, запропоновано архітектуру маршрутизатора та комбінацію сервісів, які дають змогу забезпечити виявлення та моніторинг трафіку у комп'ютерній мережі. Далі потрібно проаналізувати можливості Raspberry PI 5 і виконати практичні налаштування, що стосуються інсталяції OpenWrt та інших сервісів.

2.3. Моделі виявлення аномалій та аналізу трафіку на основі алгоритмів машинного навчання

Виявлення аномалій трафіку у комп'ютерних мережах можна звести до розв'язку задач бінарної класифікації при якому визначається приналежність об'єкту до класу об'єктів, що володіють аномальними властивостями чи поведінкою, або до класу об'єктів з нормальною поведінкою. Для цього, найбільш ефективним з точки зору практики та емпіричних тверджень, є підходи машинного навчання, що використовують алгоритми з учителем (supervised learning).

Алгоритми машинного навчання з вчителем або по-іншому алгоритми контрольованого машинного навчання відноситься до фундаментальних підходів штучного інтелекту. При застосуванні таких підходів дані мають мітки визначених класів і кожен вхід володіє відповідним правильним результатом. Процес навчання подібний до того, як учитель навчає учня, тобто відбувається «навчання під наглядом». До найбільш популярних алгоритмів машинного навчання з учителем відносять такі алгоритми:

– лінійна регресія;

– логістична регресія;

– дерева прийняття рішень;
– випадкові ліси;
– метод найближчих сусідів;
– наївний байєсівський класифікатор.

2.3.1. Модель виявлення аномалій на основі логістичної регресії

Логістична регресія відноситься до класу керованих алгоритмів машинного навчання, які використовується для задач класифікації. Основна ціль такого алгоритму полягає в тому, щоб спрогнозувати ймовірність приналежності екземпляра даних до одного чи іншого класу.

Логістична регресія доволі ефективно застосовується при бінарній класифікації з використанням сигмоїдної функції, яка приймає вхідні дані як незалежні змінні та генерує значення ймовірності в діапазоні від 0 до 1.

Наприклад, існує два класи (у даному випадку аномальний і нормальний трафік): клас 0 і клас 1. Якщо значення логістичної функції для вхідних даних перевищує 0,5 (порогове значення), тоді воно належить до класу 1, інакше воно належить до класу 0.

Логістична регресія прогнозує вихід категоріальної залежної змінної. Отже, результат повинен мати категоріальне або дискретне значення. Це може бути «Так» або «Ні», «0» або «1», «Істинно» або «Хибно» тощо, але замість того, щоб давати точні значення 0 або 1, алгоритм формує ймовірності, які знаходяться між 0 і 1.

У логістичній регресії замість «підгонки» лінії регресії підбирають логістичну функцію S-подібної формі, яка прогнозує два максимальні значення (0 або 1).

Логістична функція є сигмоїдною функцією, тобто математичною функцією, яка використовується для відображення прогнозованих значень у ймовірності. Сигмоїд перетворює будь-яке реальне значення на інше в діапазоні від 0 до 1. Значення логістичної регресії повинно бути між 0 і 1 і не може виходити за межі цього інтервалу, тому воно утворює криву, подібну до форми «S».

Крива S-подібної форми називається сигмоподібною функцією або логістичною функцією. У логістичній регресії використовується поняття порогового значення, яке визначає ймовірність 0 або 1. Наприклад, значення вище порогового значення прямує до 1, а значення нижче порогових значень прямують до 0.

На основі категорій логістичну регресію можна класифікувати на три типи:

– біноміальна або бінарна: у біноміальній логістичній регресії може бути лише два можливих типи залежних змінних, наприклад 0 або 1, «Пройшов» або «Не пройшов» тощо.

– мультиноміальна: у мультиноміальній логістичній регресії може бути 3 або більше можливих невпорядкованих типів залежної змінної, наприклад «кіт», «собаки» або «вівця».

– порядкова: у порядковій логістичній регресії може бути 3 або більше можливих упорядкованих типів залежних змінних, таких як «низький», «середній» або «високий».

Модель логістичної регресії перетворює неперервні вихідні дані функції лінійної регресії у категоріальні вихідні значення за допомогою сигмоїдної функції, яка відображає будь-який дійсний набір вхідних незалежних змінних у значення від 0 до 1.

Нехай для опису трафіку комп’ютерної мережі використовується множина наступних незалежних характеристик трафіку:

$$X = \begin{bmatrix} x_{11} & . & x_{m1} \\ . & . & . \\ x_{n1} & . & x_{nm} \end{bmatrix} \tag{2.1}$$

Залежна (цільова) змінна Y може набувати лише значення 0 або 1. При цьому 0 – інтерпретується як нормальний трафік, а 1 – як аномальний. Формально Y можна записати у вигляді:

$$Y = \begin{cases} 0, \text{якщо трафік нормальний} \\ 1, \text{якщо трафік аномальний} \end{cases} \quad (2.2)$$

Після цього застосовується мультилінійна функція до вхідних змінних (2.1):

$$z = (\sum_{i=1}^{n} w_i x_i) + b, \quad (2.3)$$

x_i – значення ознаки трафіку при і-тому спостереженні;

$w_i = [w_1, w_2, \dots w_n]$ – ваги або коефіцієнт ознаки x_i;

b – це зміщення кривої, або точка перетину її перетину з віссю.

Більш простіше мультилінійну функцію (2.3) можна представити як:

$$z = w \cdot X + b \quad (2.4)$$

На наступному етапі потрібно застосовувати сигмоїдну функцію, де вхідними даними будуть значення, одержані з формули (2.3). Це необхідно для знаходження імовірності, тобто прогнозування у.

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad (2.5)$$

На рис. 2.5 показано вигялд сигмоїдної функції.

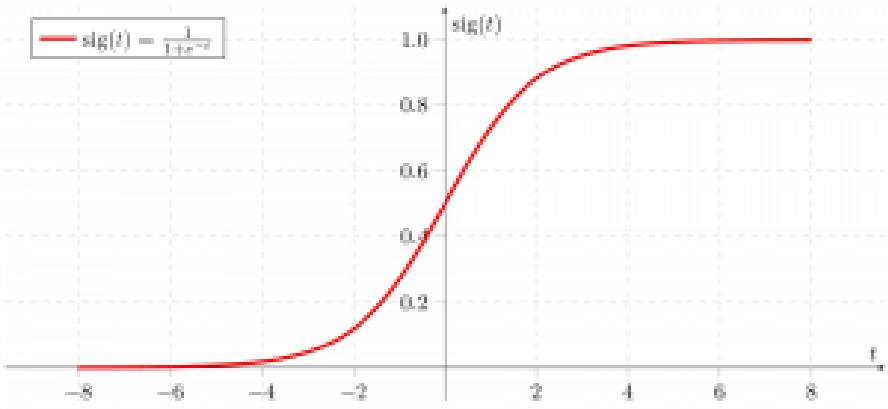

Рис. 2.5. Приклад сигмоїдної функції

Як показано на рис. 2.5, сигмоподібна функція перетворює дані неперервної змінної в імовірність, при цьому:

– $\sigma(z)$ – прямує до 1, коли z прямує до +∞;
– $\sigma(z)$ – прямує до 0, коли z прямує до -∞;
– область значень функції $\sigma(z)$ – інтервал від 0 до 1;

При обчисленні імовірності приналежності трафіку до класу нормального або аномального потоку можна скористатися наступною формулою:

$$\begin{aligned} P(y=1) &= \sigma(z) \\ P(y=0) &= 1-\sigma(z) \end{aligned} \tag{2.6}$$

Загалом рівняння логістичної регресії можна записати наступним чином:

$$\frac{p(x)}{1-p(x)} = e^{z} \tag{2.7}$$

Прологарифмувавши обидві частини рівняння (2.6) та виконавши спрощення, одержується рівняння логістичної регресії:

$$P(X;b;w) = \frac{1}{1+e^{-w\cdot X+b}} \tag{2.8}$$

Для оцінювання робастності, тобто стійкості алгоритму логістичної регресії, вводиться функція правдоподібності. Окрім цього, для керування параметрами логістичної регресії та оптимізації результатів можуть використовуватися деякі її різновиди.

2.3.2. Модель виявлення аномалій на основі методу k-найближчих сусідів

Алгоритм k-найближчих сусідів належить до контрольованих методів машинного навчання і може застосовуватися при вирішенні проблем класифікації і регресії.

Даний алгоритм машинного навчання є одним із найпростіших, однак не зважаючи на це показав свою ефективність у таких доменах як розпізнавання образів, аналіз даних і виявлення вторгнень.

Метод найближчих сусідів широко використовується в сценаріях реального життя, оскільки він непараметричний, тобто не робить жодних базових припущень щодо розподілу даних (на відміну від інших алгоритмів, таких як GMM, які припускають розподіл даних за Гаусом).

Як приклад, розглянемо наведений нижче рис. 2.6, який містить точки даних, що інтерпретують дві характеристики.

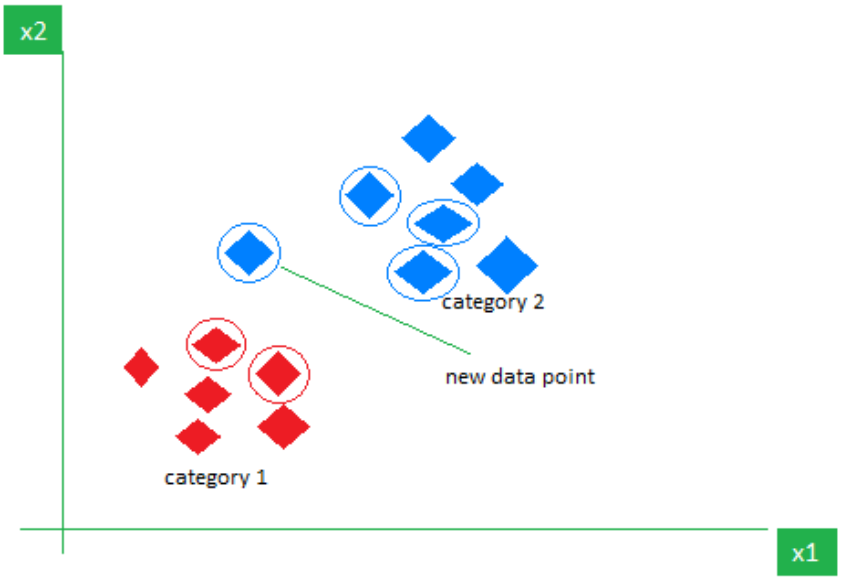

Рис. 2.6. Приклад набору даних при застосуванні методу k-найближчих сусідів

Алгоритм, на основі вхідних даних представлених на рис. 2.7, спрацьовує таким чином, що виконує пошук заданих k найближчих сусідів до вказаного об’єкту із застосуванням метрики відстані. Для визначення приналежності до класу можна використати підхід, що базується на більшості голосів або на середньому

значенню k-сусідів. Такий підхід дозволяє алгоритму адаптуватися до різних шаблонів і робити прогнози на основі локальної структури даних.

Відомо, що алгоритм KNN допомагає ідентифікувати найближчі точки або групи для деякого запиту. Але щоб визначити найближчі групи або найближчі точки, що відповідають запиту, потрібно використовувати певні метрики. Одна з можливих для використання метрик – Евклідова відстань.

Дана метрика інтерпретується як декартова відстань між двома точками, які знаходяться на площині або гіперплощині. Евклідову відстань також можна візуалізувати як довжину відрізка, який з'єднує дві точки.

Евклідову метрику розраховують за наступною формулою:

$$E\_distance(x, X_i) = \sqrt{\sum_{j=1}^{d}(x_i - X_{i_j})^2}, \tag{2.9}$$

де x – координата нового об'єкту;

X_i – координати і-го об'єкту з наявної групи;

Метрика Манхеттенської відстані, зазвичай, використовується, коли потрібно знайти загальну відстань пройдену об'єктом, а не його переміщення. Ця метрика обчислюється шляхом розрахунку абсолютної різниці між координатами точок у n-вимірному просторі:

$$M\_distance(x, y) = \sum_{i=1}^{n}|x_i - y_i| \tag{2.10}$$

де x – координати досліджуваного об'єкта у просторі;

y – координати наявного класифікованого об'єкта у просторі;

Відстань Мінковського є узагальненою метрикою двох попередніх, або дві попередні є частковим випадком метрики Мінковського. Дана метрика обчислюється за формулою:

$$distance(x, y) = (\sum_{i=1}^{n}(x_i - y_i)^P)^{\frac{1}{P}} \tag{2.11}$$

З наведеної вище формули (2.11) можна сказати, що коли $P = 2$, то це те саме, що формула для евклідової відстані, а коли $P = 1$, то отримуємо формулу для манхеттенської відстані.

Вищерозглянуті метрики найчастіше зустрічаються під час вирішення проблем машинного навчання, але є й інші метрики відстані, наприклад метрика Хеммінга, які стають у нагоді під час вирішення задач, що вимагають накладання порівнянь між двома векторами, вміст яких може бути як логічним, так і стрічковим (категоріальним).

2.3.3. Метод дерев прийняття рішень

Дерева прийняття рішень є популярним і потужним інструментом, який використовується в різних сферах, таких як машинне навчання, аналіз даних і статистика. Вони забезпечують чіткий та інтуїтивно зрозумілий спосіб прийняття рішень на основі даних шляхом моделювання зв’язків між різними змінними.

Дерево прийняття рішень представляє собою структуру, схожу на блок-схему, яка використовується при прийнятті рішень або прогнозів. Воно складається з наступних компонентів:

– вузлів – інтерпретують рішення або перевірку властивостей;

– гілок – представляють результати рішень;

– листків – кінцеві результати або прогнози.

Кожен внутрішній вузол відповідає тесту на атрибут, кожна гілка відповідає результату тесту, а кожен кінцевий вузол відповідає мітці класу або постійному значенню.

Структура дерева прийняття рішень:

– кореневий вузол – зображає весь набір даних і початкове рішення, яке потрібно прийняти;

– внутрішні вузли – представляють рішення або перевірку атрибутів, при цьому кожен внутрішній вузол має одну або кілька гілок.

– гілки – представляють результат рішення або перевірки, що веде до іншого вузла.

– листкові вузли – містять остаточне рішення або прогноз і більше не відбувається розщеплення.

Наочно структуру дерева прийняття рішень показано на рис. 2.7.

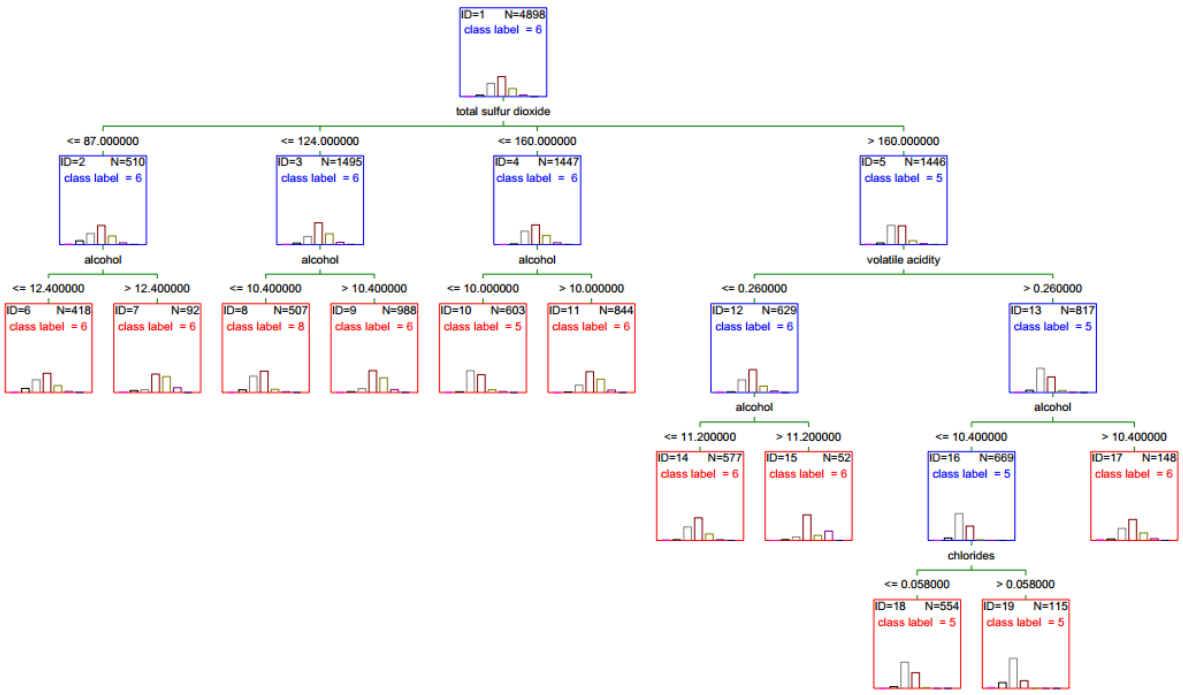

Рис. 2.7. Наочне представлення дерева прийняття рішень

Процес створення дерева рішень передбачає:

– вибір найкращого атрибута: за допомогою таких метрик, як Gini, ентропія або приріст інформації, вибирається найкращий атрибут для розділення даних.

– поділ набору даних: набір даних розбивається на підмножини на основі вибраного атрибута.

– повторення процесу: процес повторюється рекурсивно для кожної підмножини, створюючи новий внутрішній вузол або кінцевий вузол, доки не буде виконано критерій зупинки (наприклад, усі екземпляри у вузлі належать до одного класу або досягнуто попередньо визначеної глибини).

Метрика Gini: передбачає обчислення імовірності неправильної приналежності нових даних, якщо їх було випадково віднесено до відповідного класу у дата сеті та розраховується за формулою:

$$Gini = 1 - \sum_{i=1}^{n}(p_i)^2, \quad (2.12)$$

де p_i – імовірність класифікації даних за приналежністю до певного класу.

Метрика ентропії: вимірює величину невизначеності або шуми в наборі даних. Формула для обчислення ентропії має наступний вигляд:

$$Entropy = -\sum_{i=1}^{n} p_i \, log_2(p_i), \quad (2.13)$$

де p_i – це ймовірність класифікації екземпляра у певний клас.

Метрика приросту інформації забезпечує вимірювання величини зменшення ентропії або метрики Джині після того, як набір даних розділено на атрибути. Метрика приросту інформації розраховується за формулою:

$$I_{Gain} = Entropy - \sum_{i=1}^{n}(\frac{|D_i|}{|D|} * Entropy(D_i)), \quad (2.14)$$

де D_i є підмножиною D після розбиття за атрибутами.

До переваг дерев прийняття рішень належить:

– простота і можливість інтерпретації – дерева легко зрозуміти та інтерпретувати завдяки візуальному представленню, що точно відображає процеси прийняття рішень людиною;

– універсальність – можна використовувати як для завдань класифікації, так і для регресії;

– відсутність потреби в масштабуванні функцій – дерева рішень не потребують нормалізації чи масштабування даних;

– опрацювання нелінійних зв’язків: здатність фіксувати нелінійні зв’язки між функціями та цільовими змінними;

Серед недоліків дерев прийняття рішень можна виділити наступні:

– схильність до перенавчання – дерева прийняття рішень можуть легко перенавчатися на навчальній вибірці, особливо якщо вони «глибокі» з багатьма вузлами;

– нестабільність – невеликі варіації в даних можуть призвести до створення зовсім іншого дерева.

– упередженість до функцій із більшою кількістю рівнів – функції з більшою кількістю рівнів можуть домінувати в структурі дерева.

Окрім наведених вище алгоритмів, для реалізації функцій виявлення аномального трафіку можна використовувати контрольовані методи навчання на основі випадкових лісів, наївного байєсівського класифікатора, або застосовувати алгоритми unsupervised learning. Однак, виходячи з обмежень апаратної складової запропонованої архітектури маршрутизатора потрібно забезпечити оптимальний вибір моделі виявлення аномалій трафіку з врахуванням затрати мінімуму ресурсів пам’яті і процесорного часу та максимізації точності алгоритмів.

2.4. Висновки до розділу

1. Проведено аналіз функцій і структури типових маршрутизаторів, як пристроїв, що забезпечують та організовують шляхи передачі даних між хостами у комп’ютерній мережі, виявлено їх переваги та недоліки, що дало змогу встановити потенційні шляхи імплементації методів і засобів виявлення та моніторингу аномалій трафіку.

2. Запропоновано апаратно-програмне рішення щодо організації маршрутизатора на базі Raspberry PI 5 та прошивки OpenWrt з можливістю інтеграції наборів сервісів у вигляд Docker-контейнерів, що дає змогу імплементувати програмну модель виявлення аномалій трафіку та проведення його моніторингу з використання сервісу Snort практично у режимі реального часу.

3. Запропоновано та обгрунтовано застосування моделей виявлення аномалій трафіку у комп’ютерних мережах на основі алгоритмів класифікації з учителем, що дає змогу забезпечити продуктивність їх функціонування з врахуванням обмежень апаратних ресурсів.

РОЗДІЛ 3
АПАРАТНА І ПРОГРАМНА РЕАЛІЗАЦІЯ ЗАСОБІВ ВИЯВЛЕННЯ АНОМАЛІЙ ТА МОНІТОРИНГУ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

3.1. Інсталяція та налаштування системного програмного забезпечення та користувацьких сервісів на маршрутизатор

Першим кроком реалізації запропонованого у розділі 2 рішення є інсталяція системного програмного забезпечення у вигляді прошивки Open WRT. Для цього спершу необхідно завантажити підтримуваний тип і версію ядра. На рис. 3.1 показано обраний образ, який буде встановлено.

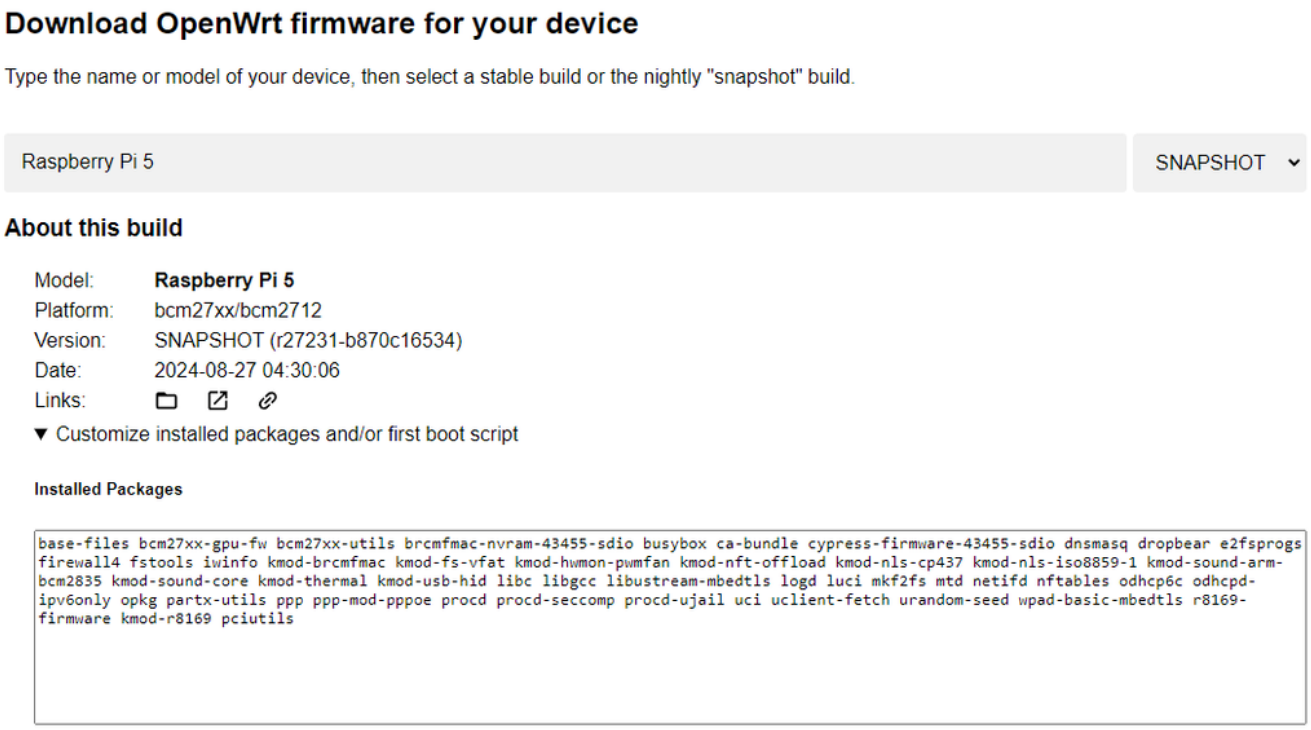

Рис. 3.1. Завантаження версії ядра прошивки OpenWrt для Raspberry PI 5

Після успішного завантаження образу на комп'ютер потрібно його записати на SD-картку, використовуючи спеціалізовані інструменти по типу RPi Imager або balenaEtcher. Сам запис є доволі тривіальною задачею і не потребує детального опису.

Оскільки, забезпечити автоматичне розширення файлової системи неможливо нативними засобами ОС Raspberry PI, то пропонується застосувати утиліту GParted. На рис. 3.2 показано вікно даної утиліти.

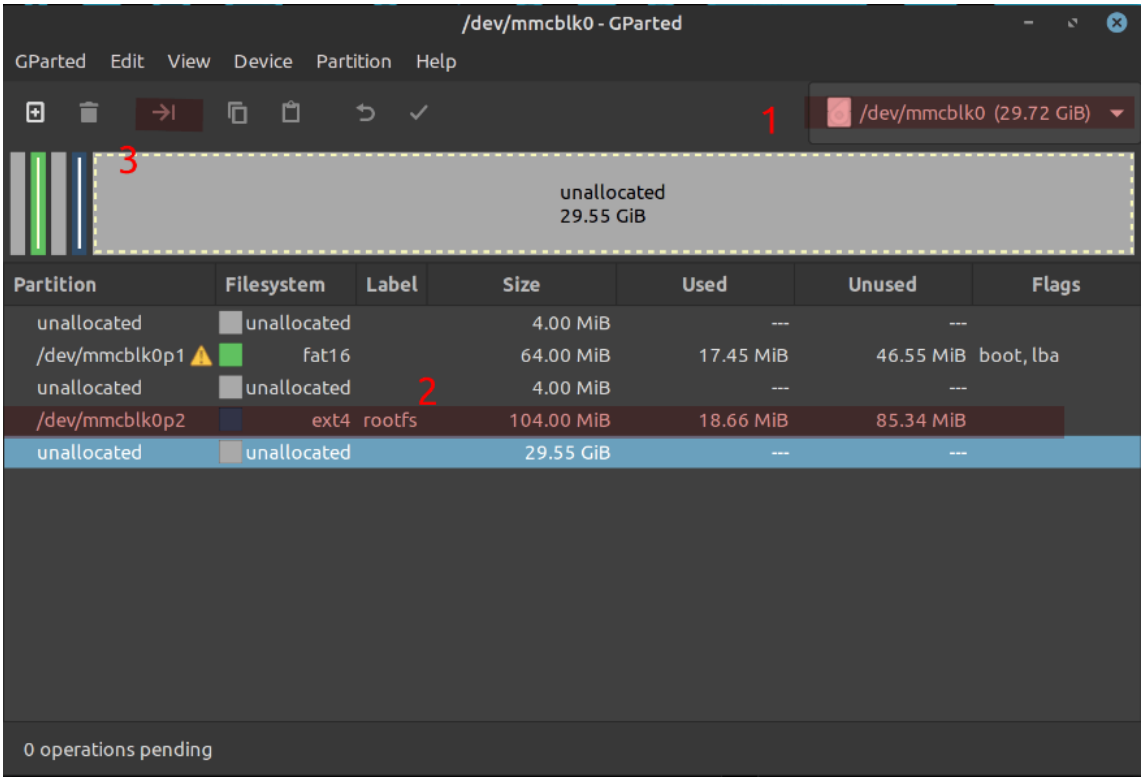

Рис. 3.2. Утиліта GParted

Представлений на рис. 3.2 сервіс є безкоштовним менеджером розділів, що дає можливість виконувати зміну розміру, копіювання і переміщення розділів без втрати даних. Для використання повноти усього функціоналу утиліти управління розміром диску можна скористатися завантажувальним образом GParted Live. Цей образ забезпечує можливість використання утиліти GParted у GNU/Linux і підтримується іншими операційними системами, зокрема, Windows та Mac OS X.

Збільшення розміру диску виконується шляхом переміщення повзунка вправо, як проілюстровано на рис. 3.3.

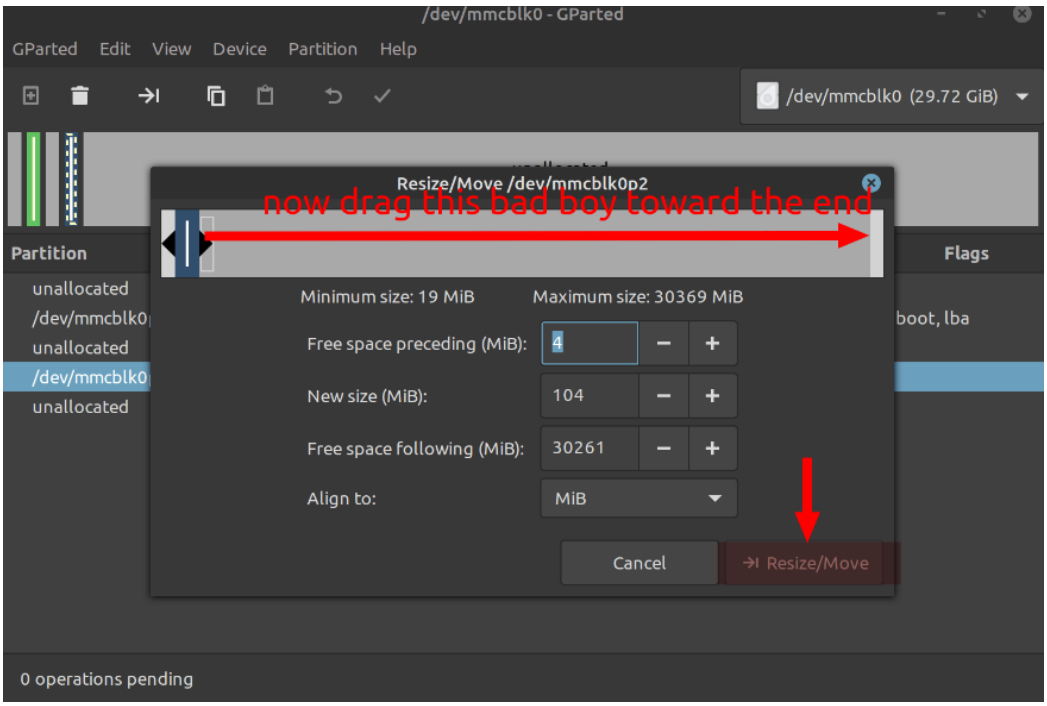

Рис. 3.3. Зміна налаштування об’єму диску для файлової системи

Встановивши потрібний розмір файлової системи з врахуванням об’єму образу Open WRT, далі потрібно виконати збереження налаштувань, як показано на рис. 3.4.

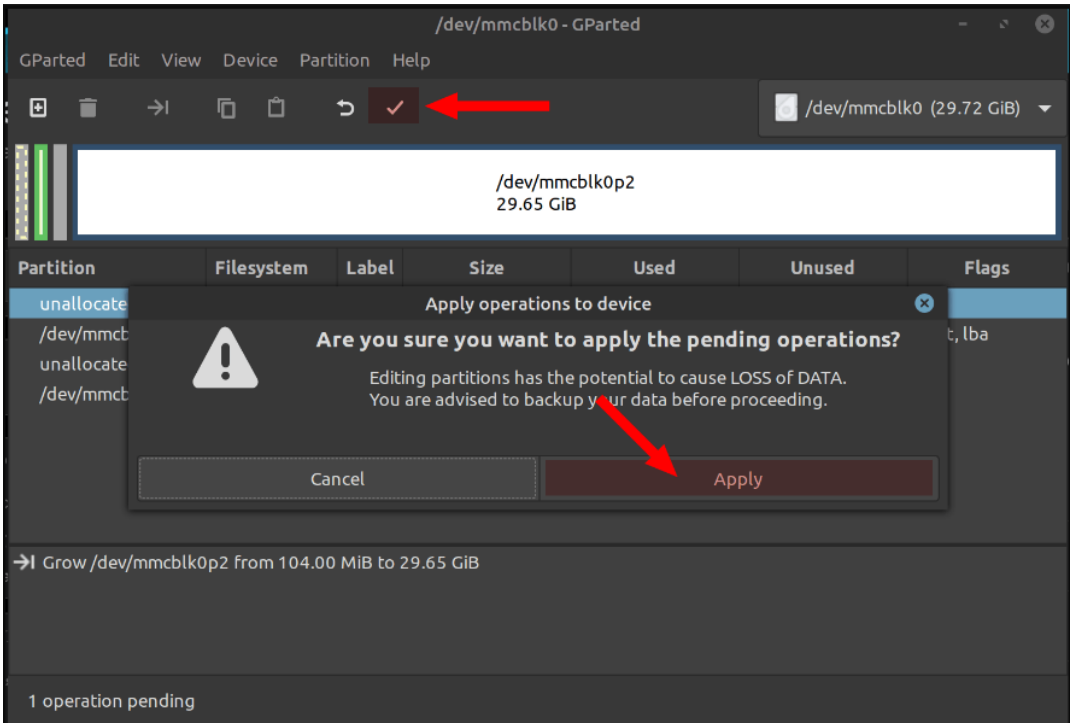

Рис. 3.4. Зберігання і застосування налаштувань розміру файлової системи

Створивши і записавши образ прошивки OpenWrt, наступний крок полягає у налаштуванні базових параметрів системного програмного забезпечення майбутнього маршрутизатора.

Встановивши SD-картку у порт Raspberry PI 5 та увімкнувши мікроконтролер, за замовчуванням буде встановлена IP-адреса 192.168.1.1. Для доступу до мікроконтролера можна скористатися зовнішнім ПК використавши кабельне з'єднання через Ethernet-порт.

Наступний етап налаштування передбачає призначення статичної IP-адреси, яка повинна знаходитись у діапазоні 192.168.1.0/24. Як приклад, такого налаштування може бути встановлена IP-адреса 192.168.1.13 та стандартна для цього випадку маска 255.255.255.0. Для того, щоб отримати доступ до RPi 5 за допомогою служби SSH необхідно виконати команду: «ssh root@192.168.1.1» без паролю.

У випадку, коли в якості зовнішнього пристрою використовується ПК з операційною системою Windows 10, підключення через SSH можна організувати за допомогою PowerShell, як показано на рис. 3.5.

```
ip a

# Sample command output:
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq master br-lan state UP group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
3: wlan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 00:00:00:00:00:00 ff:ff:ff:ff:ff:ff
5: br-lan: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.1/24 brd 192.168.1.255 scope global br-lan
       valid_lft forever preferred_lft forever
    inet6 fe80::dea6:32ff:feac:78c9/64 scope link
       valid_lft forever preferred_lft forever
6: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
```

Рис. 3.5. Успішне налаштування параметрів IP-адреси на Raspberry PI 5

Маршрутизатор, окрім функцій виявлення аномалій трафіку та його моніторингу, як було наведено у розділі 2 на рис. 2.4, повинен також реалізовувати

додаткові функцій, зокрема підтримувати два VLAN. Базові налаштування віртуальних мереж наведено у наступному підрозділі.

3.2. Організація віртуальних мереж на маршрутизаторі

Налаштування інтерфейсів віртуальних мереж з маркерами найпростіше виконувати шляхом редагування файлу конфігурації «/etc/config/network». Для внесення змін у цей файл можна скористатися доступом до Raspberry PI 5 через SSH або безпосередньо відмотувати SD-карту і підключити її до ПК. На рис. 3.6 наведено параметри, які необхідно налаштувати у конфігураційному файлі.

```
config interface 'loopback'
        option ifname 'lo'
        option proto 'static'
        option ipaddr '127.0.0.1'
        option netmask '255.0.0.0'

config interface 'lan'
        option ifname 'eth0.1'
        option proto 'static'
        option netmask '255.255.255.0'
        option ipaddr '192.168.0.1'

config interface 'wan'
        option ifname 'eth0.2'
        option proto 'dhcp'
```

Рис. 3.6. Параметри налаштування VLAN

З рис. 3.6, видно, що при таких параметрах конфігурації створюється інтерфейси двох віртуальних мереж:

– «LAN» – відповідає маркеру VLAN 1.
– «WAN» – відповідає маркеру VLAN 2.

У кожному конкретному випадку використання такого типу маршрутизатора необхідно перевірити і переналаштувати інтерфейс «LAN» у відповідності до наявних настройок локальної мережі.

Після того, як налаштовано інтерфейси віртуальних мереж можна виконувати підключення мікроконтролера Raspberry Pi 5 до визначеного порту комутатора, а також інших мережевих пристроїв. Це завершальний етап налаштування конфігурації комп'ютерної мережі та інтеграції Raspberry PI 5, як маршрутизатора.

У випадку, коли комп'ютерна мережа вже використовується і попередньо усі параметри налаштовані, можлива також і зміна інтерфейсу «WAN», що обумовлено типом і протоколом, який застосовує інтернет-провайдер для надання послуг. Додаткові налаштування, пов'язані з функціями і сервісами комп'ютерної мережі, що використовують VLAN, можна завершити з використанням веб-орієнтованого інтерфейсу OpenWRT.

3.3. Побудова і реалізація програмної моделі алгоритму виявлення аномалій та моніторингу трафіку у комп'ютерних мережах

Реалізацію програмної моделі алгоритму виявлення аномалій трафіку запропоновано імплементувати з використанням відкритих бібліотек для аналізу даних, мови програмування Python та загальнодоступного набору даних на платформі kaggle. Доступ до даних можна отримати перейшовши за посиланням https://www.kaggle.com/datasets/vidhikishorwaghela/synthetic-network-traffic.

Даний набір даних сформований на основі синтетичних даних про мережевий трафік з метою виявлення аномалій. Основне його призначення полягає у допомозі при розробці і тестуванні моделей машинного навчання в галузі безпеки мережевих інфраструктур і виявлення аномалій. Основні характеристики набору даних:

– кількість записів – 50 000;
– кількість параметрів трафіку – 10.

До складу набору даних входять такі базові поля, які описують характеристики трафіку:

– SourceIP – IP-адреса джерела мережевого трафіку;
– IP-адреса призначення – IP-адреса адресата мережевого трафіку;
– SourcePort – номер порту джерела;
– DestinationPort – номер порту адресата;
– Protocol – назва мережевого протоколу;
– BytesSent – кількість байтів, надісланих у мережі;
– BytesReceived – кількість байтів, отриманих під час мережевого зв’язку;
– PacketsSent – кількість надісланих пакетів;
– PacketsReceived – кількість отриманих пакетів;
– Duration – тривалість мережевої передачі даних.

Крім того, у дата сеті наявна бінарна цільова змінна – IsAnomaly, яка приймає значення 0 для типового мережевого трафіку та 1 – для аномального мережевого трафіку. Проте, більш інформативним для дослідження аномалій трафіку є набір даних, який містить 41 ознаку трафіку і відповідну мітку трафіку.

Програмну модель виявлення аномалій трафіку та його моніторингу на архітектурному запропоновано представити у вигляді, як показано на рис. 3.7.

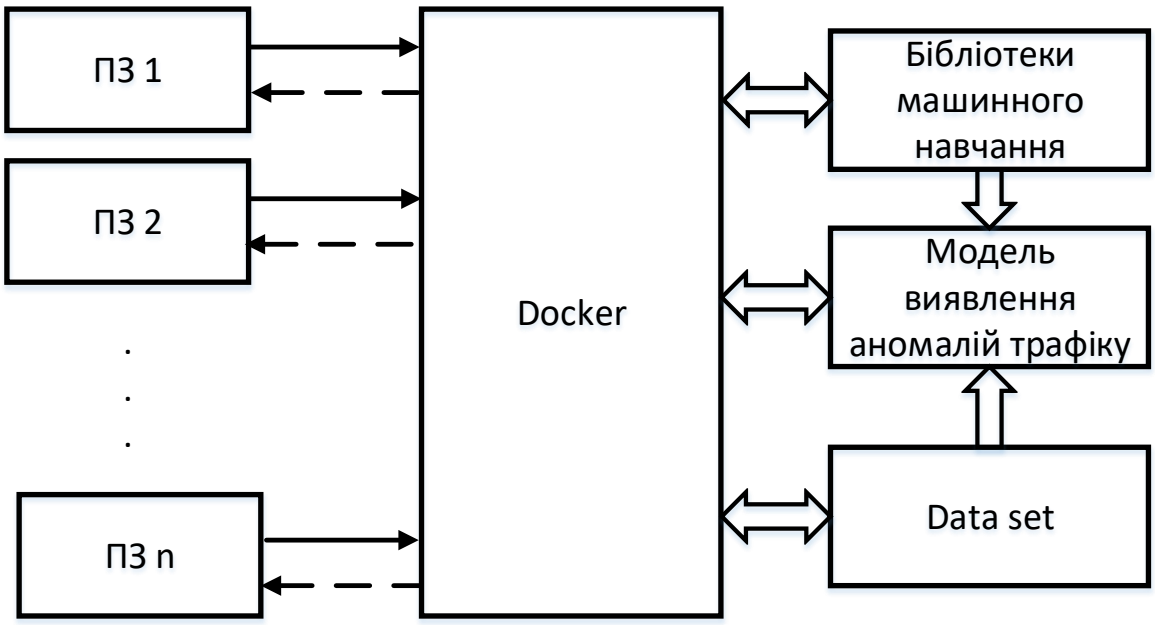

Рис. 3.7. Архітектура програмного забезпечення виявлення аномалій трафіку

Виходячи з представлення на рис. 3.7, модель виявлення аномалій трафіку з відповідними відкритими бібліотеками аналізу даних, а також вхідний набір даних у вигляді лог-файлу зі Snort запаковується у Docker-контейнер. Під зовнішнім програмним забезпеченням, у даному випадку, слід розуміти систему надсилання сповіщень адміністратору комп'ютерної мережі.

Модель виявлення аномалій трафіку передбачає реалізацію алгоритмів навчання з учителем з подальшим вибором оптимальної на основі критерії максимальної точності класифікації трафіку та використання мінімальних апаратних ресурсів.

3.3.1. Препроцесинг вхідного набору даних

Як було зазначено раніше, для імплементації моделі виявлення аномалій трафіку використовуються відкриті бібліотеки та мова програмування Python. Першим кроком при програмній реалізації моделі є підключення бібліотек, необхідних для проведення препроцесингу даних (рис. 3.8).

```
import numpy as np
import pandas as pd
```

Рис. 3.8. Підключення бібліотек препроцесингу даних

Зчитування даних з набору, що містить характеристики трафіку комп'ютерної мережі, виконується таким чином, як показано на рис. 3.9.

```
train = pd.read_csv('/kaggle/input/network-intrusion-detection/Train_data.csv')
# test set is not labelled, so we won't use it
# test = pd.read_csv('/kaggle/input/network-intrusion-detection/Test_data.csv')
train.info()
```

Рис. 3.9. Зчитування та відображення даних з дата сету

Фрагмент результату зчитування даних представлений на рис. 3.10.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25192 entries, 0 to 25191
Data columns (total 42 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   duration                     25192 non-null  int64
 1   protocol_type                25192 non-null  object
 2   service                      25192 non-null  object
 3   flag                         25192 non-null  object
 4   src_bytes                    25192 non-null  int64
 5   dst_bytes                    25192 non-null  int64
 6   land                         25192 non-null  int64
 7   wrong_fragment               25192 non-null  int64
 8   urgent                       25192 non-null  int64
 9   hot                          25192 non-null  int64
 10  num_failed_logins            25192 non-null  int64
 11  logged_in                    25192 non-null  int64
 12  num_compromised              25192 non-null  int64
 13  root_shell                   25192 non-null  int64
 14  su_attempted                 25192 non-null  int64
 15  num_root                     25192 non-null  int64
 16  num_file_creations           25192 non-null  int64
 17  num_shells                   25192 non-null  int64
 18  num_access_files             25192 non-null  int64
 19  num_outbound_cmds            25192 non-null  int64
 20  is_host_login                25192 non-null  int64
 21  is_guest_login               25192 non-null  int64
 22  count                        25192 non-null  int64
 23  srv_count                    25192 non-null  int64
 24  serror_rate                  25192 non-null  float64
 25  srv_serror_rate              25192 non-null  float64
```

Рис. 3.10. Структура і тип даних у дата сеті

Наступним кроком препроцесингу набору даних є відображення значень відповідних стовпців для розуміння їхньої області визначення (рис. 3.11).

In [4]:
```
train.describe()
```

Out[4]:

	duration	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	num_failed_logir
count	25192.000000	2.519200e+04	2.519200e+04	25192.000000	25192.000000	25192.00000	25192.000000	25192.000000
mean	305.054104	2.433063e+04	3.491847e+03	0.000079	0.023738	0.00004	0.198039	0.001191
std	2686.555640	2.410805e+06	8.883072e+04	0.008910	0.260221	0.00630	2.154202	0.045418
min	0.000000	0.000000e+00	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000
25%	0.000000	0.000000e+00	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000
50%	0.000000	4.400000e+01	0.000000e+00	0.000000	0.000000	0.00000	0.000000	0.000000
75%	0.000000	2.790000e+02	5.302500e+02	0.000000	0.000000	0.00000	0.000000	0.000000
max	42862.000000	3.817091e+08	5.151385e+06	1.000000	3.000000	1.00000	77.000000	4.000000

8 rows × 38 columns

In [5]:
```
train.describe(include='object')
```

Out[5]:

	protocol_type	service	flag	class
count	25192	25192	25192	25192
unique	3	66	11	2
top	tcp	http	SF	normal
freq	20526	8003	14973	13449

Рис. 3.11. Значення характеристик трафіку

Як видно із значень характеристик, для проведення процедур класифікації трафіку на аномальний та нормальний необхідно провести додаткову візуалізацію даних та забезпечити кодування категоріальних змінних, зокрема, типу використовуваного протоколу передачі даних у комп'ютерній мережі.

Перша за все варто розуміти кількість записів, наявних у наборі даних з мітками нормального та аномального трафіку. Для цього реалізовано фрагмент коду, який показано на рис. 3.12.

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

ax = sns.countplot(x=train["class"], palette=("coolwarm"))
abs_values = train["class"].value_counts(ascending=False).values
ax.bar_label(container=ax.containers[0], labels=abs_values)
ax.set_xticklabels(['normal', 'anomaly'])

plt.xlabel(None)
plt.show()
```

Рис. 3.12. Скрипт для візуалізації розподілу даних за класами «аномальний» та «нормальний» трафік

Результат виконання фрагменту програмного коду, представленого на рис. 3.12 показано на рис. 3.13.

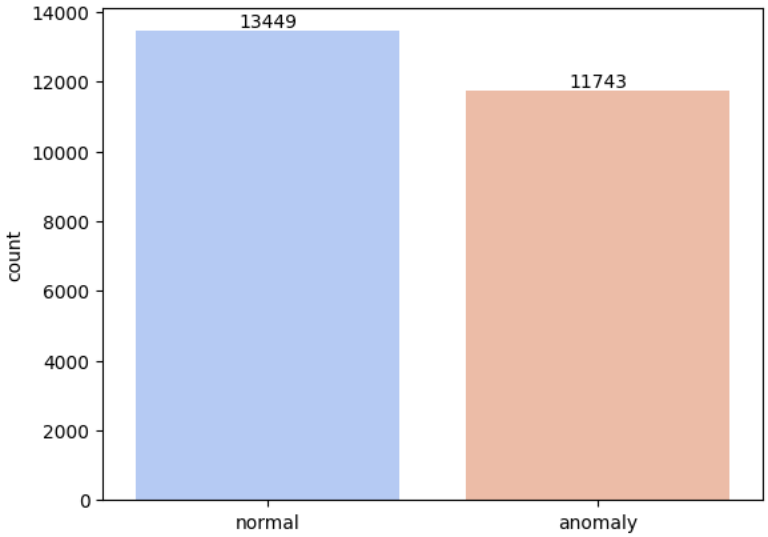

Рис. 3.13. Розподіл за кількістю даних щодо нормального та аномального трафіку

З рис. 3.13 видно, що у дата сеті практично рівномірно розподілені характеристики як аномального, так і нормального трафіку. Для категоріальних змінних, наявних у дата сеті, необхідно реалізувати скрипт, який би дозволив візуалізувати розподіл нормального та аномального трафіку за цими характеристиками. Для цього імплементовано програмний код, який наведено на рис. 3.14.

```
# categorical columns
for category in ['protocol_type', 'service', 'flag']:
    if category == 'service':
        plt.figure(figsize=(25, 5))
    sns.countplot(x=category, data=train, hue='class', palette=("coolwarm"))
    plt.title(f'Distribution of {category}')
    plt.xticks(rotation=45)
    plt.show()
```

Рис. 3.14. Програмний код візуалізації розподілу за категоріальними змінними

На рис. 3.15 наведено результат розподілу нормального та аномального трафіку за трьома основними протоколами: TCP, UDP та ICMP.

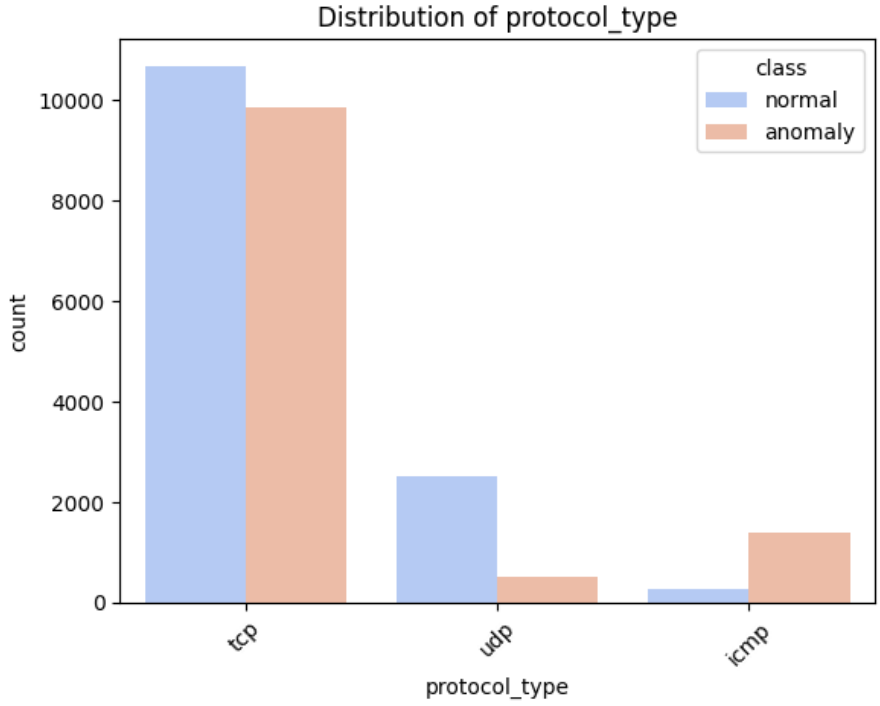

Рис. 3.15. Розподіл типового та аномального трафіку за протоколами

Для того, щоб забезпечити навчання моделей виявлення аномального трафіку, необхідно провести перетворення значень категоріальних змінних у числові, шляхом кодування міток, як проілюстровано на рис. 3.16.

```
from sklearn.preprocessing import LabelEncoder

def label_encoding(df):
    for col in df.columns:
        if df[col].dtype == 'object':
                label_encoder = LabelEncoder()
                df[col] = label_encoder.fit_transform(df[col])

label_encoding(train)

train.head()
```

Рис. 3.16. Реалізація кодування категоріальних змінних

У результаті виконання скрипту, що наведений на попередньому рисунку, одержано закодовані мітки для усіх категоріальних змінних (рис. 3.17).

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_srv_count	dst_hos
0	0	1	19	9	491	0	0	0	0	0	...	25	0.17
1	0	2	41	9	146	0	0	0	0	0	...	1	0.00
2	0	1	46	5	0	0	0	0	0	0	...	26	0.10
3	0	1	22	9	232	8153	0	0	0	0	...	255	1.00
4	0	1	22	9	199	420	0	0	0	0	...	255	1.00

Рис. 3.17. Результат застосування LabelEncoding

Встановлення залежності чи незалежності змінних у наборі даних передбачає побудову і візуалізацію матриці кореляцій. Для цього реалізовано скрипт, що показаний на рис. 3.18.

```
plt.figure(figsize=(40, 30))
sns.heatmap(train.corr(), annot=True, fmt=".1f", cmap='coolwarm')
plt.title('Correlations')
plt.show()
```

Рис. 3.18. Побудова матриці кореляцій між характеристиками трафіку

Результат візуалізації матриці кореляцій між ознаками трафіку проілюстровано на рис. 3.19.

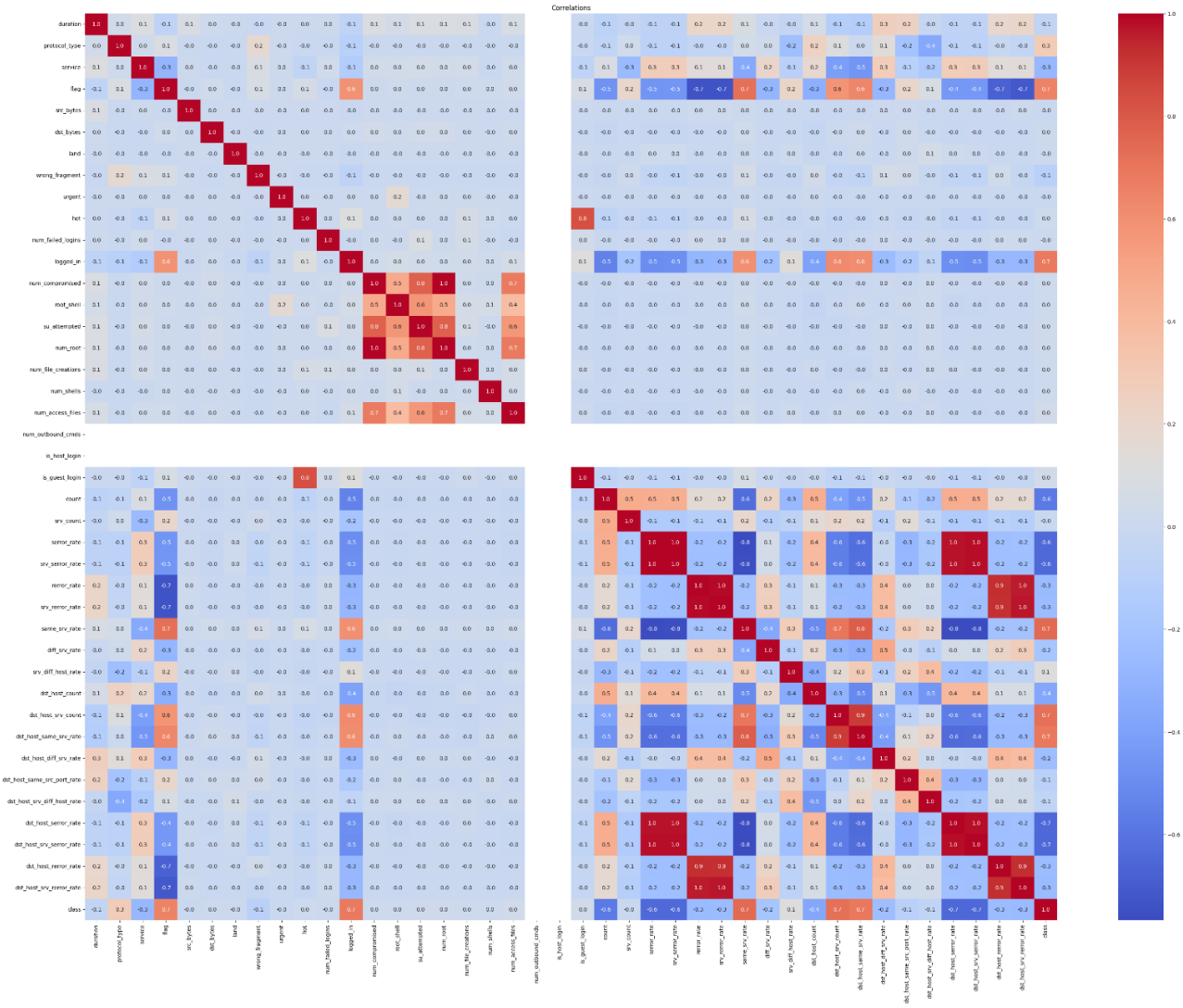

Рис. 3.19. Кореляційна теплова мапа залежностей між ознаками трафіку у комп’ютерній мережі

Як видно з рис. 3.19, багато змінних, що описують характеристики трафіку є незалежними, тому важливо з набору даних вибрати дані, які мають сильний кореляційний зв’язок, оскільки вони безпосередньо мають найбільший вплив на цільову змінну. Реалізація програмного коду для вибору характеристик із встановленою силою кореляційного зв’язку на рівні >=0.7 показано на рис. 3.20.

```python
# only strong correlations
corr_matrix = train.corr()
threshold = 0.7

# Create a mask for values above the threshold or below the negative threshold, excluding self-corr
elation
mask = np.abs(corr_matrix) >= threshold
np.fill_diagonal(mask.values, False)

# Filter the columns and rows based on the mask
filtered_columns = corr_matrix.columns[mask.any()]
filtered_corr = corr_matrix.loc[filtered_columns, filtered_columns]

print(filtered_columns)

# Plot the heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(filtered_corr, annot=True, fmt=".1f", cmap='coolwarm')
plt.title('Strong correlations')
plt.show()
```

Рис. 3.20. Вибір лише сильно корельованих змінних

У результаті виконання програмного коду, який показаний на рис. 3.20, одержано матрицю з меншою кількістю характеристик трафіку, які впливають на його нормальність чи аномальність (рис. 3.21.).

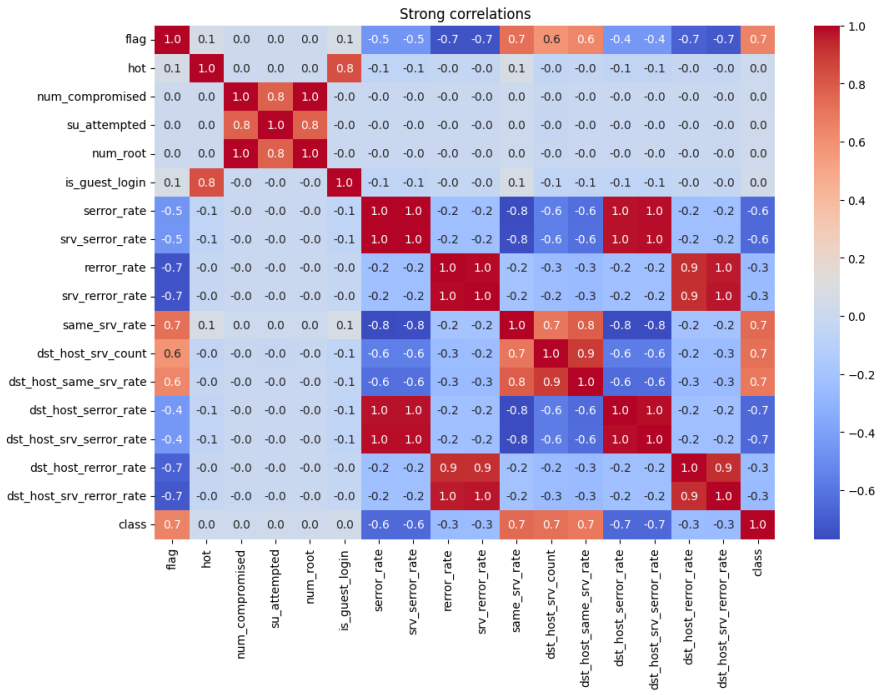

Рис. 3.21. Сильно корельовані характеристики тарфіку

Завершальним етапом препроцесингу даних у наборі з цільовими мітками нормального та аномального трафіку є видалення тих характеристик, які мають

сильну кореляцію на рівні 1, а також проведення масштабування значень, які є стійкими для викидів (рис. 3.22, рис. 3.23).

```
X = train.drop(['num_root', 'srv_serror_rate', 'srv_rerror_rate', 'dst_host_serror_rate', 'dst_ho
st_srv_serror_rate', 'dst_host_srv_rerror_rate', 'class'], axis=1)
y = train['class']
```

Рис. 3.22. Видалення даних з силою кореляційного зв’язку на рівні 1

```
from sklearn.preprocessing import RobustScaler

scaler = RobustScaler()
X = scaler.fit_transform(X)
```

Рис. 3.23. Масштабування значень даних для забезпечення стійкості до викидів

Наявний набір даних, як зазначалось раніше приблизно збалансований і підходить для навчання моделей в основі яких лежить алгоритм навчання з учителем. Однак в реальному світі набір даних часто дуже незбалансований. Отже, необхідно сформувати незбалансований дата сет із 5% аномалій. Для цього реалізовано програмний код, що представлення на рис. 3.24.

```
def shuffle_data(features, labels):
    indices = np.arange(features.shape[0])
    np.random.shuffle(indices)
    return features[indices], labels[indices]

normal = X[y == 1]
anomalies = X[y == 0]

# select random 5% of anomalies
num_anomalies = int(len(normal) * 0.05)
anomaly_indices = np.random.choice(anomalies.shape[0], num_anomalies, replace=False)
selected_anomalies = anomalies[anomaly_indices]

# combine
X_unbalanced = np.vstack([normal, selected_anomalies])
y_unbalanced = np.concatenate([np.full((len(normal)), 1), np.full((len(selected_anomalies)), 0)])

# shuffle
X_unbalanced, y_unbalanced = shuffle_data(X_unbalanced, y_unbalanced)
```

Рис. 3.24. Функція для формування не збалансованого набору даних

На цій ітерації завершується попередня підготовка і перетворення даних для застосування алгоритмів машинного навчання.

3.3.2. Реалізація алгоритмів машинного навчання для виявлення аномалій

Перед тим, як перейти до безпосередньої програмної реалізації алгоритмів виявлення (класифікації) типу трафіку у комп'ютерній мережі, необхідно задати метрики оцінювання та характеристики продуктивності алгоритмів. Для оцінювання якості функціонування алгоритмів пропонується скористатися наступними критеріями:

– час навчання алгоритму;
– час отримання результату роботи алгоритму;
– метрика точності Accuracy;
– метрика точності (кучності) Precision;
– метрика повноти;
– F1-метрика.

Програмне визначення метрик якості та функція формування результатів застосування алгоритму представлені на рис. 3.25.

```
# scores for cross-validation

from sklearn.model_selection import cross_validate
from sklearn.metrics import make_scorer, accuracy_score, precision_score, recall_score, f1_score

scoring = {
    'accuracy': make_scorer(accuracy_score),
    'precision': make_scorer(precision_score, average='binary'),
    'recall': make_scorer(recall_score, average='binary'),
    'f1_score': make_scorer(f1_score, average='binary')
}

def display_results(results):
    metrics = {
        "Metric": ["Fit Time", "Score Time", "Accuracy", "Precision", "Recall", "F1 Score"],
        "Value": [results[score_name].mean() for score_name in results]
    }
    df_metrics = pd.DataFrame(metrics)
    df_metrics.set_index("Metric", inplace=True)
    return df_metrics.T
```

Рис. 3.25. Програмне визначення метрик якості алгоритмів машинного навчання

Перший алгоритм машинного навчання з учителем, який реалізовано для виявлення аномального трафіку – байєсівський наївний класифікатор. Його програмна реалізація і результат представлено на рис. 3.26 та у табл 3.1.

```
from sklearn.naive_bayes import GaussianNB

NB = GaussianNB()
results = cross_validate(NB, X_unbalanced, y_unbalanced, cv=5, scoring=scoring, return_train_scor
e = False)
display_results(results) # custom function to display the results
```

Metric	Fit Time	Score Time	Accuracy	Precision	Recall	F1 Score
Value	0.00802	0.00986	0.832308	0.997405	0.826083	0.903628

Рис. 3.26. Реалізація наївного байєсівського класифікатора

Таблиця 3.1

Значення метрик якості алгоритму байєсівського наївного класифікатора

№	Метрика	Значення метрики
1	Час навчання алгоритму	0.00802
2	Час отримання результату роботи алгоритму	0.00986
3	Метрика точності (Accuracy)	0.832308
4	Метрика точності/кучності (Precision)	0.997405
5	Метрика повноти	0.826083
6	F1-метрика	0.903628

Як видно з результатів, представлених у табл. 3.1, алгоритм на основі наївного байєсівського класифікатора доволі швидкий і показує високі показники точності виявлення аномального трафіку.

Наступний алгоритм, який пропонується реалізувати при виявленні аномальності та моніторингу трафіку у комп’ютерній мережі – алгоритм логістичної регресії. Реалізація цього алгоритму представлена на рис. 3.27.

```
from sklearn.linear_model import LogisticRegression

# 'newton-cholesky' is the fastest
LR = LogisticRegression(max_iter=7000, solver='newton-cholesky')
results = cross_validate(LR, X_unbalanced, y_unbalanced, cv=5, scoring=scoring, return_train_score = False)
display_results(results)
```

Metric	Fit Time	Score Time	Accuracy	Precision	Recall	F1 Score
Value	0.069458	0.016787	0.991573	0.992974	0.998216	0.995588

Рис. 3.27. Реалізація логістичної регресії

Таблиця значень метрик якості для алгоритму логістичної регресії показана у табл. 3.2.

Таблиця 3.2

Значення метрик якості алгоритму логістичної регресії

№	Метрика	Значення метрики
1	Час навчання алгоритму	0.069458
2	Час отримання результату роботи алгоритму	0.016787
3	Метрика точності (Accuracy)	0.991573
4	Метрика точності/кучності (Precision)	0.992974
5	Метрика повноти	0.998216
6	F1-метрика	0.995588

Як видно з результатів, показаних у табл. 3.2, алгоритм класифікації аномального та нормального трафіку показує результати кращі у порівнянні з наївним байєсівським класифікатором, однак щодо продуктивності, то логістична прогресія незначно програє.

Наступний алгоритм навчання, який реалізовано у кваліфікаційній роботі – дерева прийняття рішень. Програмний код його реалізації проілюстровано на рис. 3.28.

```
from sklearn.tree import DecisionTreeClassifier

# 'gini' the slowest
# for criterion in ['gini', 'entropy', 'log_loss']:
#     DTC = DecisionTreeClassifier(criterion = criterion)
#     results = cross_validate(DTC, X_unbalanced, y_unbalanced, cv=5, scoring=scoring, return_train
_score = False)
#     display_results(results)

DTC = DecisionTreeClassifier(criterion = 'entropy')
results = cross_validate(DTC, X_unbalanced, y_unbalanced, cv=5, scoring=scoring, return_train_sco
re = False)
display_results(results)
```

Metric	Fit Time	Score Time	Accuracy	Precision	Recall	F1 Score
Value	0.049826	0.007962	0.997805	0.998811	0.998885	0.998848

Рис. 3.28. Алгоритм дерева прийняття рішень

Таблиця 3.3

Значення метрик якості алгоритму дерев прийняття рішень

№	Метрика	Значення метрики
1	Час навчання алгоритму	0.049826
2	Час отримання результату роботи алгоритму	0.007962
3	Метрика точності (Accuracy)	0.997805
4	Метрика точності/кучності (Precision)	0.998811
5	Метрика повноти	0.998885
6	F1-метрика	0.998848

Як видно з результатів, представлених у табл. 3.3, критерії якості алгоритму дерев прийняття рішень подібні до тих, які забезпечує логістична регресія і кращі за наївний баєйсівський класифікатор. Окрім, представлених алгоритмів у роботі програмно реалізовано алгоритм k-найближчих сусідів, алгоритм випадкових лісів, алгоритм на основі методу опорних векторів. Зведена таблиця значень критеріїв якості за усіма використовуваними алгоритмами представлена у табл. 3.4, а параметри продуктивності алгоритмів представлені у табл. 3.5.

Таблиця 3.4

Зведена таблиця значень критеріїв якості алгоритмів

№	Алгоритм	Accuracy	Precision	Recall	F1
1	Наївний байєсівський класифікатор	0.832308	0.997405	0.826083	0.903628
2	Логістична регресія	0.991573	0.992974	0.998216	0.995588
3	Дерева прийняття рішень	0.997805	0.998811	0.998885	0.998848
4	Випадкові ліси	0.998159	0.998515	0.999554	0.999034
5	Метод опорних векторів	0.990511	0.992092	0.997992	0.995033
6	Метод k-найближчих сусідів	0.99136	0.992537	0.998439	0.995478

Таблиця 3.5

Зведена таблиця значень критеріїв продуктивності алгоритмів

№	Алгоритм	Середній час навчання	Середній час роботи алгоритму
1	Наївний байєсівський класифікатор	0.008020	0.009860
2	Логістична регресія	0.069458	0.016787
3	Дерева прийняття рішень	0.049826	0.007962
4	Випадкові ліси	0.050114	0.009671
5	Метод опорних векторів	0.070016	0.016918
6	Метод k-найближчих сусідів	0.002396	0.306443

Виходячи з наведених результатів у табл. 3.4 і табл. 3.5, компромісним рішенням щодо оптимального алгоритму для виявлення та моніторингу аномалій трафіку у комп’ютерних мережах є логістична регресія, оскільки вона забезпечує параметри якості на рівні близько 99%, а за продуктивністю при зміні трафіку даний алгоритм є найбільш адаптивним.

3.4. Висновки до розділу

1. Проведено інсталяцію та налаштування прошивки OpenWrt на мікроконтролер Raspberry PI 5, що дало змогу забезпечити ефективність та надійність функціонування пристрою в якості маршрутизатора комп’ютерної мережі.

2. Налаштовано параметри портів та інтерфейсів віртуальних мереж, що дало можливість в подальшому аналізувати трафік та функціонування необхідних програмних сервісів, в тому числі і програмної моделі виявлення аномалій трафіку.

3. Програмно реалізовано моделі машинного навчання для виявлення аномалій трафіку на основі алгоритмів навчання з учителем, у результаті аналізу яких обрано оптимальну за продуктивність і точністю модель, що базується на логістичній регресії.

РОЗДІЛ 4
ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Кваліфікаційна робота магістра присвячена дослідженню методів і засобів моніторингу та виявляння аномалій трафіку у локальних комп'ютерних мережах. При роботі з апаратно-програмним забезпеченням спроектованої комп'ютерної системи потрібно забезпечити дотримання вимог з охорони праці, техніки безпеки та протипожежної безпеки.

Основними регламентуючими нормативними документом охорони праці при використанні комп'ютерної техніки є НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями».

При організації робочих місць передбачено наявність природного і штучного освітлення. Зазвичай, природне освітлення поступає у приміщення через вікна та світлові прорізи і забезпечує коефіцієнт освітленості на рівні не менше 1,5%. Орієнтація вікон – на північ або північний схід. Штучне освітлення забезпечують відповідні джерела, наприклад, люмінесцентні лампи.

Приміщення з комп'ютерною технікою не повинні межувати з будівлями, де рівень шуму чи вібрації перевищує визначені допустимі значення. Покриття підлоги повинне бути матовим з коефіцієнтом відбиття 0,3-0,5. Для внутрішнього оздоблення приміщень слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6 [23].

У приміщеннях, де організовано робочі місця користувачів ПК, повинні бути забезпечені аптечками першої медичної допомоги. Вологе прибирання у таких приміщеннях є обов'язковим кожного дня.

У відповідності до НПАОП 0.00-7.15-18, обладнання і організація робочого місця працюючих з ЕОМ мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного, розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності.

Висота робочого столу з ПК повинна бути виконана в діапазоні 680...800 мм, а ширина і глибина – 600...1400 мм і 800..1000 мм відповідно. Стіл також повинен мати достатній простір для ніг, що забезпечить зручну осанку користувача.

Стілець на робочому місці користувача ПК повинен бути підйомно-поворотним, регульованим за висотою, за кутом і за нахилом сидіння та спинки [26].

Екран комп'ютера повинен бути розміщений на відстані 600...700 мм від очей користувача. Розташування монітору має забезпечувати зручність зорового спостереження у вертикальній площині під кутом +30 градусів до нормальної лінії погляду працівника [23].

Електромережі штепсельних з'єднань та електророзеток для живлення ПК потрібно виконувати за магістральною схемою. При організації робочих місць електромережу штепсельних розеток для живлення ПК у центрі приміщення прокладають у каналах або під знімною підлогою в металевих трубах або гнучких металевих рукавах [24].

Щодо безпеки при роботі з ПК, щодня перед початком роботи необхідно очищати монітор від пилу та інших забруднень. Після закінчення роботи з ПК, він та периферійні пристрої повинні бути відключені від електричної мережі. У разі виникнення певної аварійної ситуації необхідно негайно відключити ПК від електричної мережі. Не допускається виконувати обслуговування, ремонт та налагодження ПК безпосередньо на робочому місці [23].

Щодо пожежної безпеки будівлі, де виконувався проект і приміщення його потенційної експлуатації, то дотримано вимог державних будівельних норм ДБН В.1.1-7-2016, а також правилами пожежної безпеки ДСТУ Б.В.1.1-36:2016.

Крім цього, на робочих місцях, обладнаних комп'ютерами і периферійною технікою забезпечено оптимальні значення параметрів мікроклімату: температури, руху повітря та відносної вологості, у відповідності до вимог нормативних документів.

Щодо освітлення, то приміщення де експлуатуються ПК, повинно бути обладнаним джерелами штучного освітлення та мати природне освітлення.

Нормативний документ, який регламентує вимоги до рівнів природного і штучного освітлення – ДБН В.2.5-28-2018. Розрахунок коефіцієнта природного освітлення виконують згідно методики, яка наведена у ДБН В.2.5-28-2018.

Штучне освітлення у приміщеннях з ПК забезпечується за допомогою системи загального освітлення, переважно рівномірного. В якості штучного джерела світла застосовуються люмінесцентні лампи типу ЛБ.

При використанні ПК для розробки рішень щодо виявлення аномалій трафіку у комп'ютерних мережах було дотримано наступних вимог з техніки безпеки:

- не виконувався самостійний ремонт ПК і периферійних пристроїв;
- не вносились конструктивні чи інші зміни в апаратне забезпечення комп'ютера;
- використовувались тільки ті матеріали та предмети, які стосувались розробки комп'ютерної системи оптимізації потоку звернень користувачів.

Для забезпечення вимог щодо безпечної експлуатації інформаційних технологій та мереж дотримано вимог СТУ EN 60950-1:2015 «Обладнання інформаційних технологій. Безпека. Частина 1. Загальні вимоги» (ДСТУ EN 60950-1:2015).

Таким чином, дотримання вимог і норм з охорони праці, а також правил техніки безпеки дали можливість мінімізувати негативний вплив комп'ютерної техніки на здоров'я виконавця і забезпечити імплементацію методів і засобів моніторингу та виявлення аномалій трафіку у локальних комп'ютерних мережах.

4.2. Шум, вібрація, ультразвук, електромагнітні випромінювання у виробничих приміщеннях для роботи з ВДТ та захист від них

Під шумом розуміють набір багаточисельних звуків, які швидко змінюються за частотою, силою і складаються з ряду гармонік [25]. З фізичної точки зору звуки є механічними коливальними рухами частинок пружного середовища в діапазоні частот, що чує людина. Звукові гармоніки розповсюджуються у вигляді хвиль. Шум є загально-біологічним подразником, діє не тільки на органи слуху, але може

викликати порушення роботи серцево-судинної і нервової систем, зумовлювати професійні захворювання [25]. Основними характеристиками звукових коливань є інтенсивність (сила), частота і форма звукової хвилі. Інтенсивність визначається енергією, що переноситься за 1 с звуковою хвилею через поверхню площею 1 м^2, яка перпендикулярна напрямку розповсюдження звукової хвилі. Діапазон тисків, що сприймає вухо людини, дуже широкий (10-12Вт/м^2 – поріг больового відчуття, верхня межа) [25].

З розвитком промисловості все більший контингент людей підпадає під вплив вібрацій, які являють собою механічні коливання, що передаються тілу людини. Основні параметри вібрацій – частота та амплітуда коливань, але на відміну від шуму, при якому енергія механічних коливань передається через повітряне середовище, при дії вібрацій вона розповсюджується по тканинах і викликає їх коливання або тіла людини в цілому [25]. Найбільш небезпечна вібрація частотою 16-250 Гц, дія якої призводить до вібраційної хвороби. Нормування шуму здійснюється згідно з “Санітарними нормами допустимих рівнів шуму на робочих місцях”. В Україні застосовується принцип нормування шуму на основі граничних спектрів (гранично допустимих рівнів звукового тиску) в октавних смугах частот та еквівалентних рівнів звуку. Гранично-допустимі рівні шумів санітарними нормами встановлені для кожного класу [25]:

– для високочастотних шумів (вище 800 Гц) – 75-85 дБ;
– для середньо частотних шумів (300-800 Гц) – 85-90 дБ;
– для низькочастотних шумів (до 300 Гц) – 90-100 дБ.

Шумові явища мають якість кумуляції, накопичуючись в організмі, вони все більше і більше пригнічують нервову систему. Відомо, що після шумової дії інтенсивністю 120 дБ протягом однієї години потрібно 5 годин, щоб гострота слуху повернулась до норми. Стабільні широкосмугові шуми, які перевищують граничний рівень, викликають зниження темпу, ефективності й якості роботи операторів [28].

Ультразвук широко застосовують у технологічних процесах виготовлення радіоелектронної апаратури (промивка деталей, зварювання мініатюрних вузлів

тощо) з частотою вище 2220 кГц. При цьому густина енергії ультразвукових коливань у мільйони разів більша густини енергії звуків, які ми чуємо. Тому під його дією відбувається нагрівання тіла, а при дії коливань через рідкі і тверді середовища відбувається розривання і руйнування тканин [25].

Захист від ультразвуку, який діє через повітряне середовище, досягається шляхом звукоізоляції установок (листова сталь, дюралюміній, що обклеєні гумою або руберойдом, гетинакс) або розміщення їх в окремій звукоізолюючій кабіні. Ультразвукові установки повинні мати блокування, яке відключає генератор ультразвукових коливань в момент відкривання кришок або кожухів [25].

Для запобігання шкідливої дії шуму і вібрації на організм працюючих проводяться технічні, організаційні і медико-профілактичні заходи. Одним з основних технічних заходів є зменшення при експлуатації та на стадії проектування, конструювання обладнання причин шуму і вібрації в самому джерелі утворення. Досягають цього завдяки використанню раціональної конструкції обладнання, заміни ударної дії деталей і машин коливальною, з'єднання елементів гнучкими зв'язками, врівноважування обертових частин механізмів, заміни металевих деталей пластмасовими, забезпечення різних власних частот коливань механізму з частотою збуджуючої сили [25].

Якщо неможливо ізолювати чи знизити шум і вібрацію самого джерела, потрібно:

– ізолювати джерело шуму або вібрації від навколишнього середовища засобами вібро- та звукоізоляції;

– раціонально планувати виробничі приміщення, що мають інтенсивні джерела шуму;

– збільшувати звукопоглинання внутрішніх поверхонь приміщення шляхом звукопоглинальних покриттів.

Якщо не вдається зменшити рівень шуму і вібрації на робочому місці до нормативних значень та необхідно використовувати засоби індивідуального захисту: рукавиці, взуття, навушники, м'які шоломи, які зменшують рівень звукового тиску на 40-50 дБ.

У процесі виробництв, експлуатації і зберігання комп’ютерної і радіоелектронної апаратури можуть виникати механічні і динамічні дії, що характеризуються широким діапазоном частот коливань, а також амплітудою, прискоренням і часом дії [25].

При експлуатації високочастотного обладнання всередині виробничих приміщень зниження напруженості електромагнітного випромінювання досягається такими методами:

– захист часом – обмеження часу перебування людини в електромагнітному полі, що залежить від інтенсивності опромінення або напруженості ЕМП.

– захист відстанню застосовується при неможливості послабити інтенсивність опромінення в заданій зоні іншими методами: збільшують відстань між джерелом випромінювання і обслуговуючим персоналом;

– добре виконане екранування джерела і усунення нещільності у фланцевих з’єднаннях, фідерів, зазорів у обшивці корпусів, нещільних електричних контактів;

– проведення дистанційного контролю й управління роботою передавачів з екранованого приміщення;

– засобами індивідуального захисту.

В залежності від типу джерела випромінювання, його потужності, характеру технологічного процесу може застосовуватись один з вказаних методів або будь-яка їх комбінація.

ВИСНОВКИ

Основні наукові та практичні результати полягають в наступновсму.

1. Проведено аналітичний огляд особливостей апаратно-програмної організації і принципів функціонування комп’ютерних мереж, як комунікаційної інфраструктури комп’ютерно-інформаційних систем в основі якого лежить IP-трафік. Це дало змогу виявити ознаки і їх важливість при класифікації трафіку за класами «нормальний» та «аномальний».

2. Проаналізовано сучасні методи класифікації трафіку на основі вмісту пакетів/повідомлень, а також статистичні методи і методи машинного навчання, що дало змогу обґрунтувати застосування моделей машинного навчання з алгоритмами класифікації з учителем для виявлення аномалій трафіку.

3. Проведено аналіз сучасних інструментів моніторингу трафіку у комп’ютерних мережах, що дало можливість обґрунтувати доцільність застосування відкритих сервісів, які можуть інтегруватись у будь-які апаратно-програмні засоби з підтримкою Unix-подібних операційних систем, зокрема, у маршрутизатори різних рівнів.

4. Проведено аналіз функцій і структури типових маршрутизаторів, як пристроїв, що забезпечують та організовують шляхи передачі даних між хостами у комп’ютерній мережі, виявлено їх переваги та недоліки, що дало змогу встановити потенційні шляхи імплементації методів і засобів виявлення та моніторингу аномалій трафіку.

5. Запропоновано апаратно-програмне рішення щодо організації маршрутизатора на базі Raspberry PI 5 та прошивки OpenWrt з можливістю інтеграції наборів сервісів у вигляд Docker-контейнерів, що дає змогу імплементувати програмну модель виявлення аномалій трафіку та проведення його моніторингу з використання сервісу Snort практично у режимі реального часу.

6. Запропоновано та обгрунтовано застосування моделей виявлення аномалій трафіку у комп’ютерних мережах на основі алгоритмів класифікації з

учителем, що дає змогу забезпечити продуктивність їх функціонування з врахуванням обмежень апаратних ресурсів.

7. Проведено інсталяцію та налаштування прошивки OpenWrt на мікроконтролер Raspberry PI 5, що дало змогу забезпечити ефективність та надійність функціонування пристрою в якості маршрутизатора комп’ютерної мережі.

8. Налаштовано параметри портів та інтерфейсів віртуальних мереж, що дало можливість в подальшому аналізувати трафік та функціонування необхідних програмних сервісів, в тому числі і програмної моделі виявлення аномалій трафіку.

9. Програмно реалізовано моделі машинного навчання для виявлення аномалій трафіку на основі алгоритмів навчання з учителем, у результаті аналізу яких обрано оптимальну за продуктивність і точністю модель, що базується на логістичній регресії.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 1. Львів, «Магнолія 2006». 2013. 256 с.

2. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В Комп'ютерні мережі. Книга 2. Львів, «Магнолія 2006», 2014. 312 с.

3. Буров Є. Комп'ютерні мережі. 2-ге оновлене і доповн. Вид. Львів: Бак, 2003. – 584 с.

4. Осухівська Г., Лобур Т., БілостоцькийТ. Дослідження та моделювання інтернет-трафіку комп'ютерної мережі. Збірник тез доповідей XVI наукової конференції Тернопільського національного технічного університету імені Івана Пулюя. 2012. С. 58.

5. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Програма та методичні рекомендації з проходження практики за тематикою кваліфікаційної роботи для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. ТНТУ. 2024. 45 с.

6. Луцик Н. С., Луцків А. М., Осухівська Г. М., Тиш Є. В. Методичні рекомендації до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. ТНТУ. 2024. 44 с.

7. Варавін А.В., Лещишин Ю.З., Чайковський А.В. Методичні вказівки до виконання курсового проєкту з дисципліни «Дослідження і проєктування комп'ютерних систем та мереж» для здобувачів другого (магістерського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль. ТНТУ, 2024. 32 с.

8. Kharchenko A., Galay I., Yatcyshyn V. The method of quality management software. 2011 Proceedings of 7th International Conference on Perspective Technologies and Methods in MEMS Design, MEMSTECH 2011 . Polyana. 2011. pp. 82-84.

9. Harchenko A., Bodnarchuk I., Yatcyshyn V. The modeling and optimization of software engineering processes. Modern Problems of Radio Engineering, Telecommunications and Computer Science - Proceedings of the 11th International Conference, TCSET'2012. Lviv - Slavske , 2012. p. 326.

10. Yasniy O., Pastukh O., Didych I., Yatsyshyn V., Chykhira I. Application of machine learning for modeling of 6061-T651 aluminum alloy stress−strain diagram. Procedia Structural Integrity. 48. 2023. pp. 183–189.

11. Yatsyshyn V., Pastukh O., Palamar A., Zharovskyi R. Technology of relational database management systems performance evaluation during computer systems design. Scientific Journal of TNTU (Tern.). Vol. 109. No 1. 2023. pp. 54–65.

12. Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks. Vol. 61. N. 1. January 2015. pp. 85–117.

13. Fischer and C. Igel. Training Restricted Boltzmann Machines: An Introduction. Pattern Recognition. Vol. 47, N. 1. January 2014. pp. 25–39,

14. Pastukh O., Yatsyshyn V. Development of software for neuromarketing based on artificial intelligence and data science using high-performance computing and parallel programming technologies. Scientific Journal of TNTU. Tern.: TNTU, 2024. Vol 113. No 1. P. 143–149.

15. Яцишин В.В., Демиденко А.О., Яцишин Вік.В. Підходи до виявлення аномалій трафіку у комп’ютерних мережах. Матеріали XIII міжнародної науково - технічної конференції молодих учених і студентів «Актуальні задачі сучасних технологій» (11-12 грудня 2024 р.) Тернопільського національного технічного університету імені Івана Пулюя. Тернопіль: ТНТУ. 2024. С. 520.

16. Яцишин В.В., Демиденко А.О., Яцишин Вік.В. Логістична регресія в задачах виявлення аномалій трафіку комп’ютерних мереж. Матеріали XII науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (18-19 грудня 2024 року). Тернопіль: ТНТУ. 2024. С. 149.

17. Golovko, V.A. Deep learning: an overview and main paradigms. Optical Memory and Neural Networks (Information Optics). Vol. 26, № 1. 2017. pp. 1–17.

18. Liu, J.N., Hu, Y., You, J.J., Chan, P.W.: Deep neural network based feature representation for weather forecasting. In: International Conference on Artificial Intelligence (ICAI). 2014. p. 10.

19. Mosiy L., Kozbur H., Strutynska I., Mosiy O., Yatsyshyn V. Information technology to support the digital transformation of small and medium-sized businesses. CEUR Workshop Proceedings. The 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2024). Ternopil, Ukraine, June 12-14, 2024. pp. 150-165.

20. Yatsyshyn V., Pastukh O., Lutskiv A., Tsymbalistyy V. ,Martsenko N. A Risks management method based on the quality requirements communication method in Agile approaches. CEUR Workshop Proceedings. 2022. Vol. 3309. Information Technologies: Theoretical and Applied Problems (ITTAP 2022) : proc. of the 2nd Intern. Workshop,Ternopil, Ukraine, November 22-24, 2022. pp. 1-10.

21. Dalto, M.: Deep neural networks for time series prediction with applications in ultra-short-term wind forecasting. In: IEEE ICIT.2015. pp. 257-268.

22. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Київ. 2018.

23. Катренко Л.А., Катренко А.В. Охорона праці в галузі комп'ютинґу. Львів: Магнолія-2006. 2012. 544 с.

24. Желібо Е.Н. Безпека життєдіяльності: Навчальний посібник. Київ: «Каравела», Львів: «Новий світ - 2000». 2001. 320с.

25. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання «БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ» / В.С. Стручок –Тернопіль: ФОП Паляниця В. А.156 с.

Додаток А

Текст наукових публікацій кваліфікаційної роботи магістра

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний технічний університет імені Івана Пулюя (Україна)
Університет імені П'єра і Марії Кюрі (Франція)
Маріборський університет (Словенія)
Технічний університет у Кошице (Словаччина)
Вільнюський технічний університет ім. Гедимінаса (Литва)
Наукове товариство ім. Т.Шевченка

АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ

Збірник
тез доповідей

XIII Міжнародної науково-практичної конференції молодих учених та студентів
11-12 грудня 2024 року

УКРАЇНА
ТЕРНОПІЛЬ – 2024

Матеріали XIII Міжнародної науково-практичної конференції молодих учених та студентів
«АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ» – Тернопіль, 11-12 грудня 2024 року

УДК 004.492.3

Яцишин В.В. канд. техн. наук, доцент, Демиденко А.О., Яцишин Вік. В.
Тернопільський національний технічний університет імені Івана Пулюя

ПІДХОДИ ДО ВИЯВЛЕННЯ АНОМАЛІЙ ТРАФІКУ У КОМП'ЮТЕРНИХ МЕРЕЖАХ

Yatsyshyn V.V. PhD., Assoc. Prof., Demydenko A.O., Yatsyshyn Vic. V.

APPROACHES OF ANOMALY DETECTION IN COMPUTER NETWORKS

Тренд розвитку галузі інформаційних технології орієнтований на розробку та впровадження нових методів та інструментів опрацювання великих об'ємів інформацій з паралельною імплементацією технологій штучного інтелекту. При цьому комунікаційна інфраструктура є важливим компонентом і «скелетом» будь-якої комп'ютеризованої інформаційної системи перетворення інформації. Окрім прямих функціональних обов'язків щодо трансферу даних, комп'ютерна мережа повинна забезпечувати надання різних сервісів, які пов'язані з контролем типу інформації, що передається, конфіденційністю даних, захисту від потенційних вразливостей та ряду інших. Тому актуальними задачами на сьогодні є інтеграція розумних сервісів щодо виявлення аномалій трафіку у комп'ютерних мережах задля підвищення надійності її функціонування у випадку кіберзагроз.

Виявлення та аналіз аномалій трафіку є критичним процесом в контексті кібербезпеки комунікаційної інфраструктури, а саме поняття аналізу аномалій інтерпретується як здатність виявлення нетипової діяльності або активності, що може вказувати на кібератаку, збій мережі на системному рівні чи на рівні окремих компонентів, або інші загрози. Потрібно відмітити, що виявлення аномалій повинно забезпечувати виявлення нетипової поведінки об'єктів (хостів, пакетів даних, інтенсивності запитів і т.п.) у комп'ютерній мережі на ранніх стадіях. Це обумовлено необхідністю реагування на аномалії до того, як потенційна кіберзагроза завдасть значної шкоди інформаційній інфраструктурі. Виявлення аномалій допомагає захистити чутливу інформацію від несанкціонованого доступу, підвищити доступність і надійність сервісів до появи відмов.

Для підвищення ефективності виявлення аномалій трафіку необхідно проводити заходити регулярного оновлення програмного забезпечення на рівні маршрутизаторів та хостів мережі, формувати резервні копії даних, забезпечувати постійний моніторинг з використанням передових технологій виявлення та аналізу аномалій. Серед сучасних методів виявлення та аналізу аномалій трафіку у комп'ютерних мережах варто виділити:

– Статистичний аналіз – передбачає порівняння характеристик поточного трафіку з історичними даними і забезпечує виявлення відхилень від математичного сподівання, дисперсії та інших статистичних показників.

– Машинне навчання – використовує підхід визначення типу трафіку як нормальний або нетиповий (аномальний) з використанням нейромережевого підходу, дерев чи лісів прийняття рішень, або підходи anomaly detection чи novelity detection – в основі лежать спеціалізовані алгоритми визначення відхилень від типової поведінки у комп'ютерній мережі.

Поведінковий аналіз та експертні системи – базуються на моделюванні нормального стану трафіку з відповідним встановленням відхилення від норми, використовують базу знань експертів для побудови асоціативних правил щодо потенційних загроз мережі.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

М А Т Е Р І А Л И

XII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

18-19 грудня 2024 року

ТЕРНОПІЛЬ
2024

УДК 004.89

Василь Яцишин канд. техн. наук, доцент, Антон Демиденко, Вікторія Яцишин
Тернопільський національний технічний університет імені Івана Пулюя

ЛОГІСТИЧНА РЕГРЕСІЯ В ЗАДАЧАХ ВИЯВЛЕННЯ АНОМАЛІЙ ТРАФІКУ КОМП'ЮТЕРНИХ МЕРЕЖ

Vasyl Yatsyshyn PhD., Assoc. Prof., Anton Demydenko, Viktoria Yatsyshyn

LOGISTIC REGRESSION IN COMPUTER NETWORK TRAFFIC ANOMALIES DETECTION PROBLEMS

Логістична регресія відноситься до класу керованих алгоритмів машинного навчання, які використовується для задач класифікації. Основна ціль такого алгоритму полягає в тому, щоб спрогнозувати ймовірність приналежності екземпляра даних до одного чи іншого класу.

Логістична регресія доволі ефективно застосовується при бінарній класифікації з використанням сигмоїдної функції, яка приймає вхідні дані як незалежні змінні та генерує значення ймовірності в діапазоні від 0 до 1.

Наприклад, існує два класи (у даному випадку аномальний і нормальний трафік): клас 0 і клас 1. Якщо значення логістичної функції для вхідних даних перевищує 0,5 (порогове значення), тоді воно належить до класу 1, інакше воно належить до класу 0.

Логістична регресія прогнозує вихід категоріальної залежної змінної. Отже, результат повинен мати категоріальне або дискретне значення. Це може бути «Так» або «Ні», «0» або «1», «Істинно» або «Хибно» тощо, але замість того, щоб давати точні значення 0 або 1, алгоритм формує ймовірності, які знаходяться між 0 і 1.

У логістичній регресії замість «підгонки» лінії регресії підбирають логістичну функцію S-подібної форми, яка прогнозує два максимальні значення (0 або 1).

Логістична функція є сигмоїдною функцією, тобто математичною функцією, яка використовується для відображення прогнозованих значень у ймовірності. Сигмоїд перетворює будь-яке реальне значення на інше в діапазоні від 0 до 1. Значення логістичної регресії повинно бути між 0 і 1 і не може виходити за межі цього інтервалу, тому воно утворює криву, подібну до форми «S».

Крива S-подібної форми називається сигмоподібною функцією або логістичною функцією. У логістичній регресії використовується поняття порогового значення, яке визначає ймовірність 0 або 1. Наприклад, значення вище порогового значення прямує до 1, а значення нижче порогових значень прямують до 0.

На основі категорій логістичну регресію можна класифікувати на три типи:

– біноміальна або бінарна: у біноміальній логістичній регресії може бути лише два можливих типи залежних змінних, наприклад 0 або 1, «Пройшов» або «Не пройшов» тощо.

– мультиноміальна: у мультиноміальній логістичній регресії може бути 3 або більше можливих невпорядкованих типів залежної змінної, наприклад «кіт», «собаки» або «вівця».

– порядкова: у порядковій логістичній регресії може бути 3 або більше можливих упорядкованих типів залежних змінних, таких як «низький», «середній» або «високий».

Модель логістичної регресії перетворює неперервні вихідні дані функції лінійної регресії у категоріальні вихідні значення за допомогою сигмоїдної функції, яка відображає будь-який дійсний набір вхідних незалежних змінних у значення від 0 до 1.